# Extensions modales des logiques de ressources : expressivité et calculs

Pierre Kimmel

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

# Extensions modales des logiques de ressources : expressivité et calculs

## THÈSE

présentée et soutenue publiquement le 6 décembre 2018

pour l'obtention du

**Doctorat de l'Université de Lorraine**

**(mention informatique)**

par

Pierre Kimmel

**Composition du jury**

| | | |
|---|---|---|
| *Rapporteurs :* | Nicolas Olivetti | Professeur Université Aix-Marseille, LSIS, Marseille |
| | Serenella Cerrito | Professeur Université Evry Val d'Essonne, IBISC, Evry |
| *Examinateurs :* | David Pym | Professeur University College London, Londres |
| | Hans van Ditmarsch | Directeur de recherche CNRS, LORIA, Nancy |
| | Didier Galmiche | Professeur Université de Lorraine, LORIA, Nancy |
| | Dominique Larchey-Wendling | Chargé de recherche CNRS, LORIA, Nancy |

**Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503**

Mis en page avec la classe thesul.

# Résumé

Le développement de nouveaux formalismes logiques est au coeur de nombreuses problématiques de méthodes formelles. Ces formalismes doivent répondre à la fois à des impératifs de modélisation (ils doivent permettre de décrire certains systèmes) et de calcul (ils doivent fournir des méthodes de calcul correctes et complètes). Dans ce contexte, nous nous intéressons aux logiques de ressources, en particulier les logiques BI et BBI qui traitent du partage et de la séparation de ressources et qui ont conduit aux diverses logiques de séparation dont les applications à la vérification de programmes se sont développées fortement ces dernières années.

Nous proposons dans cette thèse d'étudier, à partir des logiques BI et BBI, des logiques de séparation modales et épistémiques en se focalisant sur leurs capacités de modélisation et leur expressivité mais aussi les nouveaux calculs de preuve pour ces logiques.

Une première étude a porté sur la modélisation de propriétés dynamiques de ressources au travers d'une nouvelle logique LTBI, qui est une logique de séparation temporelle, fondée sur la logique BI et des modalités temporelles. Cette logique offre notamment des perspectives intéressantes de modélisation temporelle branchante, permettant par exemple de caractériser les processus multi-thread.

Une étude complémentaire a porté sur la modélisation de l'accès par des agents à des propriétés sous conditions de posséder certaines ressources, au travers d'une nouvelle logique ERL, qui est une logique de séparation épistémique, fondée sur la logique BBI et des modalités épistémiques. Cette logique permet de nombreuses modélisations de systèmes de contrôle d'accès.

En vue d'étendre l'expressivité de telles logiques de séparation, comme la logique BBI et ses variantes, une étude sur l'internalisation des symboles de ressources dans la syntaxe de la logique a été développée au travers des nouvelles logiques HRL et HBBI (version hybride de BBI). L'internalisation permet à la fois d'étendre l'expressivité des logiques et d'axiomatiser la logique BBI et certaines de ses variantes.

Outre la conception de ces logiques, l'étude de leur sémantique et aussi de leurs capacités de modélisation, une partie de cette thèse a été consacrée à la définition de calculs de preuve, ici de tableaux, pour ces nouvelles logiques ainsi qu'à leurs preuves de correction et de complétude.

**Mots-clés:** modélisation, preuve, logiques de ressources, logiques de séparation, logiques temporelles, logiques épistémiques, logiques hybrides, calcul des tableaux, labels, extraction de contre-modèles

# Abstract

The design of new logical formalisms is at the heart of several problems in formal methods. Those formalisms must respond to requirements both concerning modelling (they must be able to describe certain systems) and computing (they must provide complete and sound calculus methods). In this context, we look at resource logics, and in particular BI and BBI logics, that deal with the separation and sharing of resources and have led to several separation logics whose applications to software verification have been widely developped recently.

We propose in this thesis, starting from BI and BBI logics, to study some modal and epistemic separation logics by focusing on their modelling capacities and their expresiveness, as well as on the new proof calculi for those logics.

A first study deals with the modelling of dynamic resource properties through new logic LTBI, which is a temporal separation logic, based on BI logic and temporal modalities. This logic notably offers interesting perspectives in temporal branching modelling, allowing for instance to characterize multi-thread processes.

A complementary study concerns the modelling of access by agents to properties under the conditions of posessing some resources, through a new logic ERL, which is an epistemic separation logic, based on BBI logic and epistemic modalities. This logic allows many modellings of access control systems.

In order to extend the expressivity of such separation logics, like BBI logic and its variants, a study on the internalization of resources symbols in the logic's syntax has been developed through the new logics HRL and HBBI (hybrid version of BBI). Internalization allows both the extension of the expressivity of logics and the axiomatisation of BBI logic and some of its variants.

In addition to the conception of those logics, the study of their semantics and their modelling capacities, a part of this thesis is dedicated to the definition of proof calculs, here tableaux calculus, for those new logics, as well as their proof of soundness and completeness.

**Keywords:** modelling, proof, resource logics, separation logics, temporal logics, epistemic logics, hybrid logics, tableaux calculus, labels, counter-model extraction

# Remerciements

Mes premiers remerciements vont à Didier Galmiche, pour avoir accepté de diriger ma thèse et m'avoir accompagné toutes ces années durant. Il a su me conseiller précisément à la fois sur le contenu technique de mes travaux et sur tout mon parcours professionnel de doctorant. Je dois en outre le saluer pour sa grande compréhension et son écoute attentive.

Je remercie Dominique Larchey-Wendling, co-encadrant de la thèse, et précieux dispensateur de conseils techniques. Je le remercie également pour sa présence discrète lors d'épreuves personnelles.

Je remercie Serenella Cerrito et Nicolas Olivetti d'avoir accepté de rapporter sur mon travail de thèse. Je remercie en outre Nicolas pour ses pistes de recherche lors de notre rencontre dans le cadre du projet TICAMORE.

Je remercie David J. Pym qui, outre sa présence au jury, a été le moteur pendant ses visites dans l'équipe TYPES d'une partie de cette thèse et a su susciter mon imagination tout en la recadrant quand il le fallait.

Je remercie Hans Van Ditmarsch qui a également accepté de faire partie de ce jury, apportant ses lumières épistémiques à mes travaux.

Je remercie le reste de l'équipe TYPES : Daniel Méry et son expertise sur les tableaux, ainsi que mes collègues Jean-René Courtault et Vincent Demange qui ont été des aides efficaces dans mes premières années de recherche.

Je me dois de remercier également toute la bande des doctorants du Loria : Hubert, Laurent, Eric, Svyatoslav, Ludovic, Jordi, Alicia, Simon, Anne et tant d'autres qui ont animé de nombreux repas par des débats aussi complexes qu'inutiles.

Je remercie aussi Constance et Priscille qui ont si souvent partagé un repas avec moi pendant ces années de thèse.

Finalement, il me faut remercier la foule silencieuse des soutiens de tous les jours : ma femme Lucile, ma mère, ma famille et mes amis. À tous ceux-là je demande pardon d'avoir dû être si souvent tellement vague sur le contenu et l'avancement de mes travaux, et je leur suis reconnaissant de m'avoir appuyé malgré cela.

Je me dois enfin d'avoir une pensée reconnaissante à mon père avec qui j'ai apprécié de partager ces années comme collègue de l'Université de Lorraine. Là où il est, je sais qu'il est fier de moi.

*À mon père.*

"This isn't magic- it's logic- a puzzle. A lot of the greatest wizards haven't got an ounce of logic, they'd be stuck in here forever."

*J.K Rowling - Harry Potter and the Philosopher's Stone*

# Sommaire

# Table des figures

# Chapitre 1

# Introduction

La taille et la complexité croissante des systèmes et des réseaux informatiques, notamment depuis l'explosion d'Internet et des systèmes distants comme le Cloud Computing, soulèvent des problématiques nouvelles. Une constatation de ces dernières décennies est la nécessité de nouveaux progrès quant à la robustesse des programmes et des systèmes. En effet, face à la prolifération des systèmes embarqués, au développement croissant des attaques cybernétiques et à la multiplication des données sensibles sur les réseaux numériques, les techniques usuelles de tests ad hoc ne semblent plus assez solides.

Face à ce besoin, une tendance s'est développée depuis le début du XXI$^{\text{ème}}$ siècle qui consiste à utiliser les méthodes formelles pour fournir une garantie de la robustesse et de la fiabilité des systèmes informatiques et automatiques, tant vis-à-vis de la correction que de la sécurité. Ces méthodes formelles s'appuient majoritairement sur des *cadres logiques* qui doivent répondre à deux impératifs : ils doivent d'abord permettre de *modéliser* les situations étudiées, afin de pouvoir analyser les problèmes dans le cadre du formalisme étudié. D'un autre côté, les formalismes utilisés doivent aussi permettre de *prouver* des propriétés, et proposer des calculs performants permettant une réponse efficace aux problèmes modélisés. Ces deux points, *modélisation* et *calcul*, sont au cœur de la démarche de cette thèse.

De nombreuses logiques traditionnelles peuvent être utilisées pour effectuer un tel travail de modélisation et de calcul : logiques classiques et non classiques, logiques du premier et second ordre. Si ces logiques produisent des résultats, ces derniers souffrent de certaines lacunes : d'un point de vue modélisation, les logiques génériques manquent d'expressivité, ce qui limite la lisibilité des modèles. D'un point de vue calcul, il a déjà été montré comment des logiques disposant de plus d'opérateurs appropriés présentent des résultats beaucoup plus satisfaisants en terme de décidabilité et de complexité que les logiques génériques (voir par exemple [93] pour une comparaison de la logique du premier ordre et des logiques modales pour la modélisation de phénomènes temporels). On doit donc innover dans le domaine des logiques formelles pour concevoir des formalismes à la fois plus adaptés à une modélisation lisible mais aussi avec des méthodes de preuve et de vérification plus performantes, qui gardent également des propriétés essentielles telles que la correction (ce qui est prouvable est valide) et la complétude (ce qui est valide est prouvable).

Une problématique particulière dans l'objectif que nous venons d'énoncer est celle de la modélisation de structures des programmes et systèmes manipulés : structure des mémoires informatiques, structures de stockage des données concernées, structures physiques des systèmes modélisés, structures hiérarchiques des agents impliqués, etc.

Les travaux passés ont fait émergé une notion générale pour cette question, celle de *ressource*. Une ressource est ici une entité qui a la particularité de pouvoir être à la fois décomposée en ressources plus petites et composée avec d'autres pour créer une ressource plus grande. Ce concept est en même temps assez large et assez précis pour permettre une modélisation acceptable des structures que nous avons mentionnées plus haut. Les ressources peuvent ainsi modéliser des composants divers (espaces mémoire, données, permissions, messages, ...) qui partagent des propriétés (consommation, production, localisation, mobilité, accès, ...).

De nombreuses logiques ont été proposées pour raisonner sur des ressources, qu'on dénommera donc par le terme de *logiques de ressources*. Une liste non exhaustive de ces logiques comprend la *logique linéaire* [53] qui modélise plus particulièrement la production et la consommation de ressources, la *logique des ambiants* [23] qui se concentre sur la localisation et la mobilité des ressources, la *logique propositionnelle dynamique avec stockage, récupération et composition parallèle* (PRSPDL [7,10], extension de la logique PDL [43,55,58]) travaillant sur des programmes décomposables en sous-programmes parallèles et enfin la logique *BI* (Bunched Implication Logic) et ses variantes [74,80] qui se focalisent sur le partage et la séparation de propriétés entre différentes ressources.

Pour être plus précis, BI est une logique sous-structurelles qui combine des opérateurs additifs et des opérateurs multiplicatifs. Les opérateurs *additifs* $\wedge$ et $\rightarrow$, qui sont ceux de la logique intuitionniste IL, possèdent les propriétés de contraction et d'affaiblissement (on a par exemple $A$ qui est équivalent à $A \wedge A$). Les opérateurs *multiplicatifs* $*$ et $-\!\!*$ ne possèdent pas ces propriétés.Cette cohabitation entre les deux familles d'opérateurs permet d'exprimer conjointement des propriétés *partagées* par des ressources (grâce aux opérateurs additifs) et des propriétés de ressources *séparées* (grâce aux opérateurs multiplicatifs). Ainsi, par exemple, dans $A \wedge B$, les propriétés $A$ et $B$ sont partagées par une même ressource ambiante, tandis que $A * B$ exprime que la ressource ambiante peut être séparée en deux sous-ressources vérifiant respectivement $A$ et $B$. Dans le calcul des séquents originellement proposés pour BI [74,80], la cohabitation des deux familles d'opérateurs est décrite par des structures appelées "bunches", d'où le nom de Bunched Implication Logic (BI).

Plusieurs sémantiques existent pour la logique BI fondées soit sur des monoïdes partiels ou complets, soit sur des structures relationnelles, ou des structures topologiques [80]. La logique *Boolean BI* ou *BBI* [63,80] est une variante particulière de BI obtenue en remplaçant les opérateurs intuitionnistes $\wedge$ et $\rightarrow$ par leurs équivalents classiques.

La logique BI est un point de départ particulièrement intéressant pour modéliser des propriétés sur les ressources, ce qui a conduit à d'autres logiques qui en spécialisent les modèles, par des extensions du formalisme pour décrire certains types de ressources en particulier. Par exemple, la *logique de pointeurs* [47,63,75,87] ajoute à BBI des opérateurs permettant de raisonner sur les espaces mémoire des ordinateurs. Cette logique est elle-même utilisée en combinaison avec un formalisme similaire à la logique de Hoare [61] pour créer la *logique de séparation* [87] qui est un outil puissant d'analyse formelle des programmes [21].

Dans le même esprit, de nombreuses autres logiques utilisent BI ou BBI comme noyau logique, notamment la *logique BI-Loc* [13] modélisant des arbres de ressources, la *logique MBI*

[31,82,83] capturant des processus manipulant des ressources, la *logique LGL* [28] modélisant des "layered graphs" qui sont des graphes pouvant être composés ou décomposés en couches, ainsi qu'une *logique de séparation simple* [60] permettant la composition et la décomposition de valuations de variables propositionnelles.

D'un point de vue calculatoire, BI a été originellement proposée avec un *calcul des séquents* sans labels (LBI) [74], qui utilise les "bunchs" que nous avons mentionnés plus haut pour décrire l'évolution des ressources. Un *calcul des tableaux* avec labels a aussi été proposé [49]. Dans ce calcul, les *labels* représentent des ressources et un ensemble de *contraintes* entre ces labels caractérise les relations entre les ressources. Ce calcul des tableaux présente l'avantage de fournir une méthode d'*extraction de contre-modèles*, puisque les labels sont en lien direct avec les ressources des sémantiques à la Kripke de BI. Ce lien est est au centre des preuves de correction (qui utilise la notion de réalisabilité d'une branche du tableau) et de complétude (par l'extraction de contre-modèles et la propriété des modèles finis). L'étude du calcul des tableaux pour BI a également permis de démontrer la *décidabilité* de BI ainsi que la propriété des modèles finis [49].

Quant à BBI, un calcul des tableaux similaire a été proposé [66]. Toutefois, dans ce calcul, la propriété des modèles finis n'étant plus valable, la preuve de complétude doit s'appuyer sur une autre technique, qui utilise un oracle et une stratégie équitable pour générer un contre-modèle à partir d'une branche non close. L'indécidabilité de BBI a notamment été démontrée pour les modèles non déterministes, ainsi que déterministes partiels ou totaux [69].

L'adaptation du calcul des tableaux à des logiques proches de BI ou s'en inspirant n'est pas une chose triviale. Notablement, le schéma de preuve pour la preuve de complétude de ce tableau comporte de nombreux points qui demandent une restructuration avancée selon le formalisme considéré. Dans les travaux cités ci-dessous et ceux présentés dans cette thèse, nous nous concentrons sur l'extension du calcul des tableaux aux nouveaux formalismes proposés, et nous n'abordons pas dans un premier temps l'étude de la décidabilité ou de la calculabilité de ces formalismes. En effet, cette étude, déjà complexe dans le cas de BI, est ardue pour ses extensions.

Outre ces logiques dédiées à l'étude d'un type de ressources en particulier, on peut également souhaiter étendre l'expressivité de BI (ou de BBI) avec des opérateurs empruntés à d'autres logiques modales.

Un exemple intéressant est la réflexion sur le dynamisme dans la logique BI. En effet, si comme nous l'avons dit BI est appropriée à la description de systèmes structurés (comme les zones mémoire), leur description purement statique n'apporte que peu d'intérêt. On est plus intéressé par des logiques permettant de décrire l'évolution de ces structures. L'idée est d'utiliser des opérateurs modaux pour décrire l'évolution des modèles dans tandis que les opérateurs de BI permettent d'exprimer les notions de partage et de séparation de la même manière que dans les modèles statiques. Pour ce faire, de nouvelles logiques ont été proposées récemment. On peut citer DBI (Dynamic BI) [34] qui ajoute à la syntaxe de BI les opérateurs de la logique modale S4 [70], DMBI (Dynamic Modular BI) [35] qui utilise des modalités paramétrées par des actions, à la manière de Hennessy-Milner [59] et LSM (Logic with Separating Modalities)

[36], qui paramétrise les modalités par des ressources.

Dans un même état d'esprit, des extensions épistémiques de BI et BBI ont été développées, pour utiliser simultanément les opérateurs de BI exprimant partage et séparation avec des opérateurs décrivant la connaissance ou les croyances d'agents particuliers. Cela se combine parfaitement avec le pouvoir structurel de BI, puisque les sommes de connaissances sont un cas particulier de ressources. En particulier, la logique ESL étend la logique BI par des modalités épistémiques [37] pour raisonner sur les connaissances respectives de différents agents. ESL utilise donc BI dans le but de combiner logiques épistémiques et ressources, comme cela avait déjà été proposé dans [9,57,73]. Une variante de ESL a également été définie qui ajoute un système d'annonces publiques permettant de caractériser la mise à jour des croyances des agents en fonction de leurs déclarations respectives [33].

Nous voyons donc comment l'ajout d'opérateurs modaux (ici dynamiques ou épistémiques) permet d'étendre le pouvoir expressif des logiques BI ou BBI. L'extension de la syntaxe par de nouveaux opérateurs exige une nouvelle définition de la sémantique pour caractériser les nouvelles dimensions modales introduites. Ces modifications se répercutent sur le calcul des tableaux de BI (ou BBI) qui peut être étendu pour ces nouvelles logiques par l'ajout de nouveaux labels et de nouvelles contraintes qui correspondent à cette nouvelle sémantique. Les preuves de correction et de complétude permettent de confirmer, et éventuellement d'ajuster les choix de labels et de contraintes pour qu'ils correspondent exactement à la sémantique. Si la preuve de correction est souvent simple, l'extraction de contre-modèles et par la suite la complétude peuvent demander un grand effort pour être adaptées aux nouveaux formalismes [66].

L'objectif de cette thèse et de proposer de nouvelles variations et extensions permettent d'étendre l'*expressivité* des variantes existantes de BI tout en proposant des *calculs* pour ces nouvelles logiques. En outre, on se propose de mener par ce travail une réflexion plus générale sur l'expressivité des logiques et aussi sur le calcul des tableaux.

Une première problématique que nous avons explorée est celle du dynamisme dans BI. Comme nous l'avons dit, des travaux se sont déjà penchés sur cette question. Toutefois, les solutions apportées à l'ajout d'aspects dynamiques dans BI s'appuient principalement sur des extensions modales, comme l'extension DBI [34] qui ajoute à BI les opérateurs modaux de la logique S4. Cette extension permet de modéliser de nombreuses dimensions en parallèle de la séparation offerte par BI, et on peut en particulier interpréter ces modalités comme une évolution dynamique. Toutefois, d'autres logiques que S4 proposent des opérateurs plus précisemment conçus pour modéliser l'évolution temporelle [25,77]. Se baser sur de telles logiques permettrait de modéliser des systèmes dynamiques de manière plus précise, en dégageant une sémantique capturant finement une conception du temps. En outre, le rapprochement avec les logiques temporelles usuelles ouvre la possibilité de profiter des résultats calculatoires de ces dernières et rapprochent des problèmes de vérification ("model checking") où ces logiques sont utilisées [65].

Nous proposons une logique de séparation temporelle, fondée sur BI (donc avec des opérateurs additifs intuitionnistes) et qui utilise au lieu des modalités de S4 les modalités $\Diamond$, $\Box$ et $\bigcirc$ de la logique temporelle linéaire temporelle LTL [77]. Nous dénommons cette logique *LTBI*, pour

Linear Temporal BI. Tout comme pour LTL dont nous nous inspirons, la sémantique temporelle de LTBI est une structure temporelle linéaire, ce qui veut dire que les instants sont totalement ordonnés et isomorphes aux entiers naturels pour l'ordre, ce qui permet notamment de désigner l'instant suivant d'un instant donné à l'aide de l'opérateur $\bigcirc$.

Cette logique permet ainsi efficacement de modéliser des systèmes dynamiques de consommation et de production de ressources ainsi que des systèmes dynamiques pouvant être décomposés en sous-systèmes distincts. En outre, une perspective prometteuse est la possibilité d'utiliser LTBI comme logique temporelle branchante, ce qui en fait un outil pour la modélisation de processus parallèles.

Notons qu'une extension similaire de la logique BI par les opérateurs de la logique LTL a été proposée et dénommée tBI (pour temporal BI) [64]. Toutefois, ce travail présente essentiellement un calcul à la Hilbert et un lien formel avec des extensions analogues de la logique linéaire. Nous apportons ici une sémantique différente, à la Kripke, avec un accent sur le pouvoir expressif de notre version, et un calcul des tableaux qui nous semble plus pertinent à développer.

Nous avons ainsi développé un calcul des tableaux pour LTBI, inspiré de celui pour DBI [34], avec deux ensembles de labels, marquant respectivement les *ressources* et les *instants temporels*. Contrairement à celui de DBI, le calcul pour LTBI doit intégrer les dimensions temporelles linéaires de notre sémantique, ce que nous réalisons par l'ajout de contraintes appropriées et de règles garantissant la linéarité des labels temporels. En revanche, l'isomorphisme pour l'ordre avec les entiers naturels, qui garantit l'existence d'un instant suivant pour tout instant donné, pose problème. Les travaux pour garantir cet isomorphisme n'ont pas complètement abouti mais nous avons résumé le résultat manquant à démontrer sous forme d'une conjecture dont nous expliquons les tenants et les aboutissants. Cette conjecture joue un rôle particulier dans l'adaptation de l'extraction de contre-modèles pour notre calcul des tableaux, et par suite dans la preuve de complétude qui en découle. La correction du calcul reste standard et son adaptation aux labels et aux contraintes temporelles ne présente pas de difficulté excessive.

De plus, nous avons esquissé un second calcul des tableaux pour LTBI, inspiré cette fois de celui pour LTL [65]. En introduisant la possibilité dans les tableaux d'obtenir des cycles (en obtenant un nœud dont le fils est aussi un de ses ascendants), ce second calcul permet de se passer de labels temporels et offre la possibilité de générer des modèles sous formes de graphes. Nous démontrons la correction de ce calcul en adaptant des techniques de preuve venant des calculs de DBI et LTL. De futurs travaux consisteront à prouver sa complétude, ce qui n'est pas trivial, ainsi que proposer une traduction systématique des preuves par tableaux pour LTBI d'un calcul vers l'autre. Le second calcul est en outre une occasion de discuter de la pertinence des labels dans les calculs des tableaux et la possibilité de remplacer de manière générale les labels par des structures spécifiques (ici, les cycles et certaines arêtes des graphes).

En second lieu, nous avons aussi voulu compléter le travail sur les extensions épistémiques de BBI. En particulier, nous avons voulu combiner la logique ESL d'une part [37], qui étend BBI avec des modalités d'accès épistémiques telles que $K_a\phi$ qui spécifie que l'agent $a$ connaît la propriété $\phi$, et la logique LSM d'autre part [36], qui ajoute des opérateurs modaux paramétrés par des ressources, tels que $\Diamond_s$ et $\Box_s$ qui décrivent l'accès à un monde modal sous la condition d'avoir accès à la ressource locale $s$. Le résultat est une logique dénommée *ERL* pour Epistemic Resource Logic. Elle étend BBI (donc avec des opérateurs additifs classiques) avec des

modalités ($\mathbf{L}^s_a$, $\mathbf{M}^s_a$, $\mathbf{N}^s_a$, $\widetilde{\mathbf{L}}^s_a$, $\widetilde{\mathbf{M}}^s_a$ et $\widetilde{\mathbf{N}}^s_a$) paramétrées à la fois par des ressources et des agents. Chacune d'entre elles signifie (à quelques nuances près) qu'un agent donné peut avoir accès à une propriété sachant qu'il a accès à une ressource locale donnée.

La sémantique de ERL utilise des relations d'équivalence paramétrées par les agents et compose de différentes manières la *ressource locale* servant de condition d'accès avec la *ressource ambiante*, ce qui permet de créer plusieurs opérateurs.

On voit comment de tels opérateurs peuvent facilement être utilisés pour modéliser des problèmes de contrôle d'accès. On a notamment modélisé avec ERL des problèmes d'accès restreint, d'accès concurrent ou d'accès conjoint. On a aussi montré comment ERL pouvait être intelligemment utilisé pour modéliser des systèmes par couches. Enfin, une perspective de développement est l'utilisation de ERL dans la modélisation de certains protocoles de communication.

Nous présentons un calcul des tableaux pour ERL dans l'esprit de celui pour ESL [37]. La principale modification apportée au calcul d'ESL est la prise en compte des ressources locales dans le calcul, avec l'exigence que chacune de ces ressources locales possède un label de ressources correspondant. Nous montrons que le calcul est correct et complet et nous donnons une méthode pour extraire des contre-modèles. Ces preuves doivent elles aussi prendre en compte cette dualité entre ressources locales et labels de ressources associés. Puisque des travaux similaires ont été proposés pour la logique ESL [33], une des perspectives que nous proposons est l'extension d'ERL par un système d'annonces publiques, ce qui permettrait par exemple de modéliser l'échange de messages entre agents dans une politique de sécurité.

Enfin, nous acons étudié l'expressivité de BBI en proposant l'*internalisation* des ressources dans la syntaxe, à la manière de ce qui a été fait à partir des logiques modales pour les logiques hybrides [14–16]. L'idée est d'ajouter aux opérateurs de BBI (donc avec des opérateurs additifs classiques) des symboles atomiques représentant chacun une ressource en particulier, les *nominaux*. Nous incluons aussi un opérateur @ qui force la validité d'une formule en une ressource. Par exemple, $@_i(\phi)$ est valide si $i$ valide $\phi$. Cet ajout étend l'expressivité de BBI en permettant d'énoncer directement des propriétés sur des ressources en particulier. En utilisant les nominaux comme variables, l'hybridation offre aussi une meilleure lisibilité lors de l'énoncé de propriétés complexes.

Cette construction permet enfin de transférer la sémantique de BBI depuis le cadre sémantique vers des axiomes. Ainsi, nous présentons une logique *HRL* (pour Hybrid Resource Logic) qui ne possède pas les propriétés structurelles de BBI mais qui peut être restreinte par des axiomes pour définir une logique *HBBI* (pour Hybrid BBI) qui est prouvée équivalente sémantiquement à BBI. Cette construction permet de facilement étendre ou restreindre les propriétés de BBI en rajoutant ou en enlevant des axiomes.

Nous proposons aussi un calcul des tableaux pour HRL qui conduit ensuite à un calcul pour HBBI par ajout de règles axiomatiques. Ce calcul ne comporte *pas de labels* puisque les nominaux sont utilisés à leur place. Malgré la présentation radicalement différente, la structure de ces tableaux est extrêmement proche du calcul pour BBI. Nous illustrerons cette proximité en partant des preuves de correction et de complétude de BBI [66] pour notre nouveau calcul avec le transfert des contraintes de labels vers des règles du calcul qui simplifie les preuves en supprimant les discussions sur les ensembles de contraintes, mais les étend par le plus grand

nombre de règles à considérer.

Enfin, nous étudierons plus en avant cette proximité entre les calculs dans le cas très particulier des formules de BBI. Comme nous l'avons esquissé, le calcul pour HBBI permet en particulier de raisonner sur les formules de BBI (puisque les deux logiques sont sémantiquement équivalentes). Nous formalisons cette équivalence en proposant une méthode systématique pour traduire un tableau du calcul de BBI vers un tableau du calcul de HBBI et vice-versa. Nous montrons également que le caractère clos des tableaux est préservé par cette transformation. Cette étude est un nouvel éclairage sur l'utilisation de labels dans les calculs des tableaux (dans la mesure où nous montrons une manière de s'en débarrasser dans ce cas précis).

Notons que l'hybridation de BBI a déjà été proposée sous la forme de la logique HyBBI [20] qui ajoute à BBI les nominaux et l'opérateur @. Si l'idée de base est similaire notre approche diffère en plusieurs points : tout d'abord dans la manière d'énoncer la sémantique, mais aussi dans l'étude plus poussée que nous faisons de l'expressivité d'une telle logique, dans la proposition que nous faisons d'utiliser l'hybridation comme moteur d'une axiomatisation de BBI et enfin par le calcul des tableaux inédit que nous proposons et l'étude comparative avec celui de BBI.

Des perspectives pour HRL et HBBI incluent le développement d'une version hybride de BI (intuitionniste) mais aussi la possibilité de plusieurs extensions axiomatiques de HRL pour capturer d'autres propriétés des ressources. On propose par exemple l'extension de HBBI pour les ressources inversibles.

Outre ces contributions, de nombreuses autres perspectives sont apportés par ces travaux, notamment une réflexion globale sur les labels dans les calculs des tableaux ainsi que la possibilité de proposer de nouvelles logiques qui combinent (au moins partiellement) les différentes modalités étudiées ici.

Le cœur de cette thèse est constitué bien entendu de la présentation de nouvelles logiques LTBI, ERL, HRLet HBBI. Toutefois, nous commençons dans le Chapitre 2 par présenter plus en détail les logiques BI et BBI qui sont à l'origine de nos travaux. Nous en présentons l'origine, la syntaxe, la sémantique et l'expressivité. Nous discutons aussi des différences entre BI et BBI. Nous présentons en détail le calcul des tableaux pour BBI ce qui sera une première approche pour les calculs de nos autres logiques. Enfin, nous présentons les différentes extensions modales et épistémiques existantes de BI et BBI, en particulier DBI et ESL.

Dans le Chapitre 3, nous présentons les réflexions qui ont amené à la création de la logique LTBI par la convergence de la logique BI et de la logique temporelle linéaire LTL. Nous exposons la syntaxe et la sémantique de LTBI et en montrons le pouvoir d'expressivité par la modélisation de systèmes dynamiques de consommation et de production de ressources, de systèmes dynamiques partitionnés et de logique temporelle branchante. Nous présentons ensuite un calcul des tableaux pour LTBI ainsi que sa correction, sa complétude et une méthode d'extraction de contre-modèles. Nous présentons aussi le calcul des c-tableaux pour LTBI à partir de l'étude du calcul de LTL qui utilise des graphes au lieu des labels temporels. Nous montrons

sa correction et discutons de sa complétude et de la traduction de tableaux en c-tableaux.

Dans le Chapitre 4, nous présentons la logique ERL qui utilise des modalités paramétrées par des agents et des ressources. Nous en présentons la syntaxe et la sémantique. Nous montrons ensuite en détail comment ERL peut être utilisée comme outil de modélisation de contrôle d'accès. Pour cela nous développons plusieurs exemples significatifs en modélisant tour à tour un système faillible à accès contrôlé, un système d'accès joint et un système d'accès concurrent par sémaphores. Nous proposons également un exemple de modélisation par couches utilisant ERL. Enfin, nous proposons un calcul des tableaux pour ERL et nous prouvons sa correction et sa complétude, avec une méthode d'extraction de contre-modèles.

Dans le Chapitre 5, nous montrons comment l'ajout de ressources internalisées dans la syntaxe de BBI, à la manière des logiques hybrides, offre une nouvelle présentation des logiques de ressources permettant une expressivité étendue. Nous présentons la syntaxe et la sémantique de la logique HRL qui combine les opérateurs de BBI et ceux des logiques hybrides, mais sans la structure sémantique des ressources de BBI. Nous montrons ensuite comment HRL peut être restreinte par des axiomes à une logique HBBI, équivalente à BBI. Nous illustrons par des exemples à la fois l'ajout d'expressivité de HBBI par rapport à BBI mais aussi la possibilité d'utiliser HRL avec d'autres axiomes pour modéliser d'autres logiques. Nous présentons un calcul des tableaux sans labels pour HRL dont nous montrons la correction et la complétude, avec une méthode d'extraction de contre-modèles. Nous montrons également comment étendre ce calcul pour HBBI par des règles axiomatiques, et nous montrons ces mêmes propriétés.

Dans le Chapitre 6, nous montrons comment nous pouvons traduire tout HBBI-tableau du Chapitre 5 en un BBI-tableau du Chapitre 2 et vice-versa. Nous en profitons pour discuter à nouveau de la pertinence des labels dans les calculs des tableaux.

Enfin, le Chapitre 7 conclut cette thèse en récapitulant ses contributions et ses perspectives, en ajoutant également quelques perspectives transversales.

# Chapitre 2

# Les logiques BI et BBI

Le travail de cette thèse continue le travail autour de la logique BI et sa variante BBI en en proposant de nouvelles extensions, dans la continuité de travaux existants. Les logiques sous-structurelles BI et BBI et les logiques de séparation dont elles sont le noyau logique permettent de modéliser des processus ou systèmes mettant en jeu des ressources, c'est à dire des entités pouvant être composées et décomposées.

Dans ce chapitre, nous voulons présenter ces logiques et un certain nombre de travaux sur lesquels se fondent notre approche et nos travaux. En premier lieu nous donnons un bref aperçu de ce que sont les logiques BI et BBI.

Nous établissons ensuite dans un cadre plus formel la syntaxe et la sémantique de ces logiques. Nous proposons deux sémantiques différentes pour BI (à base de monoïdes complets et partiels) et une pour BBI. Nous présentons aussi un calcul des tableaux pour BBI avec labels et contraintes qui sera une base pour des calculs similaires dans les chapitres suivants. Nous illustrons ensuite par deux exemples l'utilisation de BI et BBI comme outil de modélisation pour la consommation de ressources. Nous évoquons enfin les différents axes qui ont amenés dans d'autres travaux à des extensions de BI et BBI, notamment par des modalités dynamiques et épistémiques.

## 2.1 Les logiques de séparation

La *bunched implication logic* ou *logique BI* est une logique sous-structurelle, au même titre que les *logiques de relevance* [42,85,86] et la *logique linéaire* [53]. C'est à dire que le calcul des séquents sur laquelle elle est fondée présente une absence des règles de contraction et d'affaiblissement pour certains opérateurs. Concrètement, cela empêche la duplication ou la suppression de formules lors de preuves. Pour être plus précis, BI propose deux familles d'opérateurs, des *opérateurs additifs* (intuitionnistes) qui peuvent être contractés et affaiblis et les *opérateurs multiplicatifs* qui ne le peuvent pas. Cette cohabitation des deux familles d'opérateurs, additifs et multiplicatifs, permet à BI d'exprimer des propriétés de *séparation* et de *partage*.

Les opérateurs additifs de BI sont ceux de la logique intuitionniste IL, ses opérateurs multiplicatifs sont ceux du fragment multiplicatif de la logique linéaire intuitionniste MILL [12,53].

BI permet d'exprimer des propriétés sur des entités (appelées *ressources*) ainsi que sur des agrégats de ces entités, les ressources pouvant être composées entre elles. Cette logique est


FIGURE 2.1 – Représentation des opérateurs multiplicatifs de BI

donc plus particulièrement pertinente pour exprimer des *partages* de propriétés entre entités ou la *séparation* d'une entité en sous-entités ayant des propriétés différentes.

Les opérateurs multiplicatifs introduits par BI sont la *conjonction multiplicative* $*$ (prononcé "étoile" ou "star"), qui affirme la validité de deux propriétés dans deux ressources conjointes, et l'*implication multiplicative* $-\!\!*$ (prononcé "baguette magique" ou "magic wand"), qui affirme la réalisation d'une formule sous la condition de l'adjonction d'une ressource en vérifiant une autre.

La Figure 2.1 représente une idée intuitive du fonctionnement de ces deux opérateurs. Il existe plusieurs formalismes sémantiques pour décrire BI [49,68,69,72,80]. Nous en présenterons deux dans la section 2.2, les *monoïdes partiels* et les *monoïdes complets*.

En outre, il existe une variante très utilisée de BI dont la base additive est non pas la logique intuitionniste mais la logique classique. Cette variante est appelée *Boolean BI* ou *BBI* [63,80].

Comme nous l'avons vu, BI présente des similitudes avec la logique linéaire LL. La manière dont l'implication multiplicative $-\!\!*$ est définie, en particulier la rapproche de l'implication multiplicative de la logique linéaire, $\multimap$, tandis que l'implication additive $\to$ peut être définie dans la logique linéaire par l'égalité $A \to B \equiv !A \multimap B$.

Toutefois, BI est bien distincte de LL. La première différence est que le fragment additif de BI est exactement celui de la logique intuitionniste. Elle préserve donc les propriétés de cette logique, comme par exemple la distributivité des conjonctions sur les disjonctions, c'est à dire que dans BI $A \wedge (B \vee C)$ est sémantiquement équivalent à $(A \wedge B) \vee (A \wedge C)$, ce qui n'est pas le cas dans la logique LL et la logique linéaire intuitionniste MILL [12,53].

En outre, un point fort et essentiel de BI est que en son sein les deux familles d'opérateurs (additifs intuitionnistes et multiplicatifs) cohabitent et peuvent être directement combinés. En effet, la définition $A \to B \equiv !A \multimap B$ ne recouvre pas exactement la même expressivité que l'implication intuitionniste $\to$ définie dans BI. Ainsi, par son articulation avec $\to$, l'implication $-\!\!*$ se distingue fortement de $\multimap$. En particulier, on a $A \multimap B \vdash_{LL} A \to B$ et $A -\!\!* B \nvdash_{BI} A \to B$. On peut donc trouver des formules valides dans LL et pas dans BI et vice-versa. Par exemple, la formule $p \multimap ((p \to p) \to q) \to q$ n'est pas valide dans LL, mais son équivalent $p -\!\!* ((p \to p) \to q) \to q$ est valide dans BI.

Enfin d'un point de vue expressif, la logique BI est aussi différente de LL. BI et LL ont toutes les deux l'ambition de modéliser des *ressources*. Toutefois, pour LL, ce sont les propositions qui symbolisent des *ressources*, tandis que dans BI les propositions expriment des *propriétés sur les ressources*, ces dernières étant modélisées (par exemple) par une structure de Kripke. Ainsi l'expressivité de LL se centre sur la *consommation et la production* de ressources tandis que celle de BI est consacrée au *partage et à la séparation* de propriétés par les ressources.

Nous allons illustrer par un petit exemple abstrait comment les notions de partage et de séparation peuvent être exprimées avec BI à l'aide des deux conjonctions de cette logique.

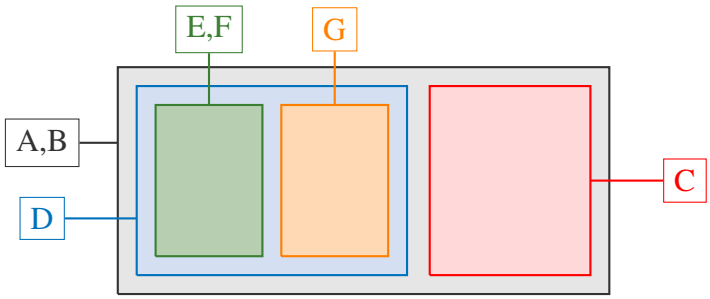



FIGURE 2.2 – Illustration de partage et de séparation de propriétés

La Figure 2.2 présente une représentation abstraite de *ressources* (sous forme d'aires colorées) étiquetées par des *propriétés* (sous forme de propositions $A, B, C, \ldots$). Cette représentation a un sens puisque les aires les plus grandes peuvent être décomposées en des aires plus petites : l'aire noire se décompose en l'aire rouge et l'aire bleue, qui elle-même se décompose en l'aire jaune et l'aire verte.

On a alors de nombreux cas de partage et de séparation exprimés par la structure de nos ressources. Par exemple, la zone noire possède la propriété $A$ et la propriété $B$. Elle *partage* donc ces deux propriétés. Du point de vue de notre formalisme logique, ceci est exprimé par une conjonction additive. On peut donc dire que la zone noire a la propriété $A \wedge B$.

A contrario, la zone noire exprime aussi une notion de séparation, puisqu'elle se décompose en deux parties distinctes, la zone bleue qui possède la propriété $D$ et la zone rouge qui possède la propriété $C$. La ressource noire se *sépare* donc en deux ressources, rouge et bleu, qui vérifient respectivement les propriétés $C$ et $D$. Ceci est exprimé par la conjonction multiplicative de BI et on peut dire que la zone noire possède la propriété $C * D$.

Remarquons que nous venons d'exprimer successivement deux propriétés différentes, toutes deux vérifiées par la zone noire, à savoir $A \wedge B$ et $C * D$. La zone noire partage donc ces deux propriétés, et en répétant notre raisonnement on peut donc dire que la zone noire possède la propriété $A \wedge B \wedge (C * D)$.

On peut réitérer ce processus de description pour la zone bleue, qui vérifie alors la propriété $D \wedge ((E \wedge F) * G)$. Enfin, en combinant toutes ces informations, on peut énoncer une proposition qui exprime toutes les propriétés de l'aire noire, ce qui résume toute la Figure 2.2 :

$$((((E \wedge F) * G) \wedge D) * C) \wedge A \wedge B$$

Bien entendu, ce petit exemple reste très abstrait. On voit cependant aisément comment il pourrait être adapté pour décrire de nombreux systèmes différents, selon que les zones colorées modélisent des zones de mémoire, des espaces physiques ou des sommes de connaissances. Cela illustre bien le fait que BI et BBI ont été à l'origine de logiques qui expriment des propriétés de séparation et de partage sur des zones mémoire [47,63,75,87], des zones de confiance [2,3], des arbres de ressources [13], des "layered graphs" [28], des processus manipulant les ressources [28,82,83], etc...

## 2.2 La logique BI

Comme nous l'avons déjà mentionné auparavant, la spécificité des logiques de séparation, et de BI en particulier, est de mélanger deux types de connecteurs logiques conjonctifs et disjonctifs : les connecteurs *additifs* sont ceux de la logique intuitionniste IL tandis que les connecteurs *multiplicatifs* viennent de la variante MILL de la logique linéaire [12,53].

Les connecteurs additifs sont la conjonction $\wedge$, la disjonction $\vee$ et l'implication $\rightarrow$. On y ajoute également le neutre pour la conjonction $\top$ et le neutre pour la disjonction $\bot$. Ces connecteurs permettent d'exprimer le *partage*. En effet, si l'on a par exemple $A \wedge B$ valide pour la ressource courante, cette dernière possédera à la fois la propriété $A$ et la propriété $B$. $A$ et $B$ sont alors partagées par la ressource courante.

Les connecteurs multiplicatifs sont la conjonction $*$ et l'implication $-\!\!*$. On y ajoute également le neutre pour la conjonction I. Ces connecteurs permettent d'exprimer la *séparation*. Si l'on a par exemple $A * B$ valide pour la ressource courante, cette dernière peut être décomposée en deux ressources, l'une validant $A$, l'autre validant $B$. $A$ et $B$ sont alors valides dans deux contextes séparés.

Enfin, on utilise des symboles propositionnels pour dénoter les formules atomiques de notre logique. On obtient donc la syntaxe suivante :

**Définition 2.1** (Syntaxe de BI)**.** *Soit Prop un ensemble de symbole propositionnels. Le langage $\mathcal{L}$ de la logique BI est donnée par la grammaire suivante pour tout $p \in Prop$ :*

$$X ::= p \mid \top \mid \bot \mid X \wedge X \mid X \vee X \mid X \rightarrow X \mid \mathrm{I} \mid X * X \mid X -\!\!* X$$

On introduit en outre la négation par l'égalité suivante : $\neg X ::= X \rightarrow \bot$.

Pour cette syntaxe, et pour toutes les autres syntaxes présentées dans cette thèse, on définit la taille d'une formule comme suit :

**Définition 2.2** (Taille d'une formule)**.** *Soit $\phi$ une formule dans une syntaxe logique donnée. On définit la taille $t(\phi) \in \mathbb{N}$ de $\phi$ par :*

- *Si $\phi$ est un opérateur d'arité* 0 *(c'est à dire soit un symbole propositionnel atomique, soit un opérateur atomique, typiquement $\bot$, $\top$ ou* I*), alors $t(\phi) = 1$.*
- *Si $\phi$ est de la forme $\phi = *(\psi)$ où $*$ est un opérateur unaire, alors $t(\phi) = t(\psi) + 1$.*
- *Si $\phi$ est de la forme $\phi = \psi_1 * \psi_2$ où $*$ est un opérateur binaire, alors $t(\phi) = max(t(\psi_1), t(\psi_2)) + 1$.*

Nous allons maintenant donner une sémantique formelle à la syntaxe de BI. De nombreuses sémantiques ont été proposées pour BI [49,68,69,72,80]. Nous développerons ici deux sémantiques de Kripke pour BI, très proches l'une de l'autre. Nous présentons tout d'abord la sémantique à base de *monoïdes de ressources partiels*.

### 2.2.1 Sémantique à monoïdes partiels

Jusqu'à la fin de cette thèse, pour toute fonction partielle, on dénotera par $\downarrow$ (c'est à dire $f(x)\downarrow$ ou $a \cdot b \downarrow$) une image définie et $\uparrow$ (c'est à dire $f(x)\uparrow$ ou $a \cdot b \uparrow$) une image indéfinie.

**Définition 2.3** (Monoïde partiel commutatif)**.** *On appelle* monoïde partiel commutatif *un triplet* $(E, e, \bullet)$ *tel que :*

- *$E$ est un ensemble*
- *$e \in E$*
- *$\bullet : E \times E \rightharpoonup E$ est une fonction partielle commutative, associative et possédant $e$ comme élément neutre. C'est à dire qu'on a pour tout $e_1, e_2, e_3 \in E$ :*
  - *si $e_1 \bullet e_2 \downarrow$, alors $e_2 \bullet e_1 \downarrow$ et $e_1 \bullet e_2 = e_2 \bullet e_1$ (commutativité)*
  - *si $(e_1 \bullet e_2) \bullet e_3 \downarrow$, alors $e_1 \bullet (e_2 \bullet e_3) \downarrow$ et $(e_1 \bullet e_2) \bullet e_3 = e_1 \bullet (e_2 \bullet e_3)$ (associativité)*
  - *$r_1 \bullet e \downarrow$ et $r_1 \bullet e = r_1$ (élément neutre)*

Pour obtenir un monoïde de ressource, nous allons munir un monoïde d'un pré-ordre, défini comme suit.

**Définition 2.4** (Pré-ordre)**.** *Soit $E$ un ensemble et $\sqsubseteq$ une relation sur $E$ ($\sqsubseteq \subseteq E \times E$). On dit que $\sqsubseteq$ est un* pré-ordre *si elle est transitive et réflexive, c'est à dire que pour $e_1, e_2, e_3 \in E$ on a :*

- *$e_1 \sqsubseteq e_1$ (réflexivité)*
- *si $e_1 \sqsubseteq e_2$ et $e_2 \sqsubseteq e_3$, alors $e_1 \sqsubseteq e_3$ (transitivité)*

**Définition 2.5** (Monoïde de ressources partiel)**.** *Un* monoïde de ressources partiel *est une structure $\mathcal{R} = (R, e, \bullet, \sqsubseteq)$ telle que :*

- *$(R, e, \bullet)$ est un monoïde partiel commutatif*
- *$\sqsubseteq$ est un pré-ordre de $R$*
- *$\bullet$ est compatible avec $\sqsubseteq$, i.e. pour tout $r_1, r_2, s \in R$, si $r_1 \sqsubseteq r_2$ et $r_2 \bullet s \downarrow$, alors $r_1 \bullet s \downarrow$ et $r_1 \bullet s \sqsubseteq r_2 \bullet s$*

Pour tout monoïde de ressource $\mathcal{R} = (R, e, \bullet, \sqsubseteq)$, nous appellerons *ressource* tout élément de l'ensemble $R$, *ressource unitaire* l'élément $e$ et *composition* la fonction $\bullet$.

Nous allons maintenant définir une interprétation, c'est à dire une fonction qui attribuera aux variables propositionnelles des valeurs de vérité. Pour tout ensemble $\mathcal{E}$, et jusqu'à la fin de cette thèse, on note $\mathbb{P}(\mathcal{E})$ l'ensemble des parties de $\mathcal{E}$, c'est à dire l'ensemble de tous les sous-ensembles de $\mathcal{E}$.

**Définition 2.6** (Interprétation)**.** *Soit $\mathcal{R} = (R, e, \bullet, \sqsubseteq)$ un monoïde de ressources partiel. Une* interprétation *de Prop pour $\mathcal{R}$ est une fonction $[\![\cdot]\!] : Prop \to \mathbb{P}(R)$ qui est monotone, c'est à dire qu'elle vérifie la propriété suivante :*

$$\forall p \in Prop, \forall r, r' \in R, \text{ si } r \sqsubseteq r' \text{ et } r \in [\![p]\!] \text{ alors } r' \in [\![p]\!] \text{ (monotonie)}$$

Notons que l'interprétation est intrinsèquement liée aux ressources. En effet, il ne s'agit pas de savoir si une proposition est valide dans l'absolu, mais plutôt si elle est valide dans le contexte donné, c'est à dire pour une ressource donnée. Ainsi, $r \in [\![p]\!]$ indique que la ressource $r$ valide la proposition $p$, et $[\![p]\!]$ est l'ensemble de toutes les ressources rendant $p$ valide.

**Définition 2.7** (Modèle)**.** *Un* modèle *de BI est un triplet* $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *où* $\mathcal{R} = (R, e, \bullet, \sqsubseteq)$ *est un monoïde de ressources partiel,* $[\![\cdot]\!]$ *est une interprétation de Prop pour* $\mathcal{R}$ *et* $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times R$ *une relation de forcing, définie récursivement par les relations suivantes pour tout* $p \in Prop, \phi, \psi \in \mathcal{L}$ *:*

- $r \vDash_{\mathcal{K}} p$ *ssi* $r \in [\![p]\!]$
- $r \vDash_{\mathcal{K}} \mathrm{I}$ *ssi* $e \sqsubseteq r$
- $r \vDash_{\mathcal{K}} \top$ *toujours*
- $r \vDash_{\mathcal{K}} \bot$ *jamais*
- $r \vDash_{\mathcal{K}} \phi \wedge \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *et* $r \vDash_{\mathcal{K}} \psi$
- $r \vDash_{\mathcal{K}} \phi \vee \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *ou* $r \vDash_{\mathcal{K}} \psi$
- $r \vDash_{\mathcal{K}} \phi \to \psi$ *ssi pour tout* $r' \in R$ *tel que* $r \sqsubseteq r'$ *et* $r' \vDash_{\mathcal{K}} \phi$*, on a* $r' \vDash_{\mathcal{K}} \psi$
- $r \vDash_{\mathcal{K}} \phi * \psi$ *ssi il existe* $r', r'' \in R$ *tels que* $r' \bullet r'' \downarrow$ *et* $r' \bullet r'' \sqsubseteq r$ *et* $r' \vDash_{\mathcal{K}} \phi$ *et* $r'' \vDash_{\mathcal{K}} \psi$
- $r \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ *ssi pour tout* $r' \in R$ *tel que* $r \bullet r' \downarrow$ *et* $r' \vDash_{\mathcal{K}} \phi$*, on a* $r \bullet r' \vDash_{\mathcal{K}} \psi$

Nous dirons que la ressource *r satisfait* la formule $\phi$ dans le modèle $\mathcal{K}$ lorsque $r \vDash_{\mathcal{K}} \phi$. Dans le cas contraire, on écrira $r \nvDash_{\mathcal{K}} \phi$.

Nous avions annoncé que les opérateurs additifs de BI (c'est à dire $\wedge$, $\vee$, $\to$, $\top$ et $\bot$) étaient ceux de la logique intuitionniste IL. Nous pouvons aller plus loin ici en observant que la sémantique des opérateurs additifs coïncident avec celle d'IL. On peut donc affirmer (même si nous ne le démontrerons pas formellement ici) que BI est une extension conservative d'IL, c'est à dire que la restriction de BI à ses opérateurs additifs a la même expressivité qu'IL (toute formule valide dans l'une l'est dans l'autre et vice-versa) [80].

**Définition 2.8** (Validité)**.** *On dit qu'une formule* $\phi$ *de* $\mathcal{L}$ *est valide dans BI et on note* $\vDash_{BI} \phi$ *(ou simplement* $\vDash \phi$ *si le contexte n'est pas ambigu) lorsque* $e \vDash_{\mathcal{K}} \phi$ *dans tout modèle* $\mathcal{K}$*.*

*On dit en outre que* $\psi$ *est une conséquence logique de* $\phi$ *dans BI et on note* $\phi \vDash_{BI} \psi$ *(ou* $\phi \vDash \psi$*) lorsque* $r \vDash_{\mathcal{K}} \phi$ *implique* $r \vDash_{\mathcal{K}} \psi$ *pour toute ressource r de tout modèle* $\mathcal{K}$*.*

Ainsi, une formule est valide si elle satisfaite par la ressource unitaire $e$. Comme nous l'illustrerons plus clairement plus tard, la ressource unitaire peut s'interpréter comme l'absence de ressource, d'information ou de contexte. Ainsi une formule est valide si elle est validée en l'absence de tout contexte particulier, sans besoin de ressource.

Enfin, voici un lemme dont nous illustrerons la pertinence plus loin.

**Lemme 2.1.** *Pour tout modèle* $\mathcal{K}$*, pour toutes ressources* $r, r'$ *de ce modèle et pour toute formule* $\phi$*, si* $r \vDash_{\mathcal{K}} \phi$ *et* $r \sqsubseteq r'$*, alors* $r' \vDash_{\mathcal{K}} \phi$*.*

### 2.2.2 Sémantique à monoïdes complets

Comme nous l'avons déjà mentionné, de nombreuses autres sémantiques ont été proposées pour BI. Nous n'en présenterons qu'une seule ici, qui est une légère variante de la précédente.

La première sémantique que nous avons exposée est basée sur des monoïdes partiels, qualifiés ainsi parce que la fonction de composition est partiellement définie. En effet, lorsque BI est

utilisée pour modéliser des systèmes, la composition entre ressources n'a de sens que pour certaines ressources, et il est donc nécessaire de traiter d'une manière ou d'une autre les ressources dont la composition n'est pas clairement définie dans le contexte.

La solution que nous avons utilisée jusque là est de pleinement assumer cette incomplétude en traitant, dans les définitions, uniquement les cas où la composition est définie. Toutefois, cette approche alourdit les définitions et les preuves en forçant à distinguer les cas où la composition est définie.

Une approche alternative consiste à ne pas traiter les cas de composition non définie et d'utiliser une ressource spécifique, dite *inconsistante*, qui collecte toutes les compositions non définies. On note cette ressource $\pi$. Les définitions deviennent alors les suivantes :

**Définition 2.9** (Monoïde commutatif)**.** *On appelle* monoïde commutatif *un quadruplet* $(E, e, \pi, \bullet)$ *tel que :*

- $E$ *est un ensemble*
- $e \in E$
- $\pi \in E$
- $\bullet : E \times E \rightharpoonup E$ *est une fonction commutative, associative, possédant* $e$ *comme élément neutre et* $\pi$ *comme élément absorbant. C'est à dire qu'on a pour tout* $e_1, e_2, e_3 \in E$ *:*
  - $e_1 \bullet e_2 = e_2 \bullet e_1$ *(commutativité)*
  - $(e_1 \bullet e_2) \bullet e_3 = e_1 \bullet (e_2 \bullet e_3)$ *(associativité)*
  - $r_1 \bullet e = r_1$ *(élément neutre)*
  - $r_1 \bullet \pi = \pi$ *(élément absorbant)*

**Définition 2.10** (Monoïde de ressources)**.** *Un* monoïde de ressources *est une structure* $\mathcal{R} = (R, e, \pi, \bullet, \sqsubseteq)$ *telle que :*

- $(R, e, \pi, \bullet)$ *est un monoïde commutatif*
- $\sqsubseteq$ *est un pré-ordre* $R$
- $\bullet$ *est compatible avec* $\sqsubseteq$*. C'est à dire que pour tout* $r_1, r_2, s \in R$*, si* $r_1 \sqsubseteq r_2$*, alors* $r_1 \bullet s \sqsubseteq r_2 \bullet s$
- $\pi$ *est le plus grand élément de* $R$ *par rapport à* $\sqsubseteq$*, c'est à dire que pour tout* $r \in R$*, on a* $r \sqsubseteq \pi$

**Définition 2.11** (Interprétation)**.** *Soit* $\mathcal{R} = (R, e, \pi, \bullet, \sqsubseteq)$ *un monoïde de ressources. Une* interprétation *de Prop pour* $\mathcal{R}$ *est une fonction* $[\![\cdot]\!] : Prop \to \mathbb{P}(R)$ *qui est monotone et inconsistante, c'est à dire qu'elle vérifie les propriétés suivantes :*

$$\forall p \in Prop, \forall r, r' \in R, \text{ si } r \sqsubseteq r' \text{ et } r \in [\![p]\!] \text{ alors } r' \in [\![p]\!] \text{ (monotonie)}$$

$$\forall p \in Prop, \forall r \in R, \text{ si } \pi \sqsubseteq r \text{ alors } r \in [\![p]\!] \text{ (inconsistance)}$$

La définition de modèle varie très légèrement de celle donnée en 2.7

**Définition 2.12** (Modèle complet)**.** *Un* modèle complet *de BI est un triplet* $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *où* $\mathcal{R} = (R, e, \bullet, \sqsubseteq)$ *est un monoïde de ressources,* $[\![\cdot]\!]$ *est une interprétation de Prop pour* $\mathcal{R}$ *(selon la Définition 2.11) et* $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times R$ *une relation de forcing, définie récursivement par les relations suivantes pour tout* $p \in Prop, \phi, \psi \in \mathcal{L}$ *:*

- $r \vDash_{\mathcal{K}} \bot$ *ssi* $\pi \sqsubseteq r$
- $r \vDash_{\mathcal{K}} \phi * \psi$ *ssi il existe* $r', r'' \in R$ *tels que* $r' \bullet r'' \sqsubseteq r$ *et* $r' \vDash_{\mathcal{K}} \phi$ *et* $r'' \vDash_{\mathcal{K}} \psi$
- $r \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ *ssi pour tout* $r' \in R$ *tel que* $r' \vDash_{\mathcal{K}} \phi$*, on a* $r \bullet r' \vDash_{\mathcal{K}} \psi$
- *le forcing est défini pour tous les autres opérateurs comme dans la Définition 2.7.*

La Définition 2.8 ainsi que le Lemme 2.1 sont toujours valables. On a en outre un nouveau lemme :

**Lemme 2.2.** *Pour tout modèle* $\mathcal{K}$ *et pour toute formule* $\phi$ *on a* $\pi \vDash_{\mathcal{K}} \phi$.

Ainsi, la ressource $\pi$ capture l'impossibilité d'une composition et absorbe toutes les compositions non définies. De la même manière que, dans la logique classique, on peut tout déduire d'une proposition fausse, la ressource inconsistante force toutes les propositions, ce qu'exprime le Lemme 2.2.

### 2.2.3 Calculs pour BI et propriétés des sémantiques

Plusieurs calculs ont été proposés pour BI. Le *calcul des séquents avec bunchs* [74,80] a été le premier introduit, mais il existe aussi un *calcul des tableaux* avec labels et contraintes [46,49,67,68,72], qui sera plus particulièrement au centre de notre étude, ainsi qu'une *méthode des connexions* [72] et un *calcul de display* [19].

Ces différents calculs sont corrects et complets vis-à-vis des deux formalismes sémantiques que nous avons introduits ci-dessus. Notons en outre que le deuxième formalisme, qui utilise la ressource absurde $\pi$, permet d'établie la *propriété des modèles finis* pour BI et ainsi de prouver sa *décidabilité* [49]. C'est donc cette sémantique que nous utiliserons pour les nouvelles logiques construites autour de BI (voir Chapitre 3).

L'autre formalisme, où la composition n'est pas définie pour toutes les ressources, est plus intuitive et sera la base de notre sémantique pour la variante classique BBI que nous allons maintenant expliciter.

## 2.3 La logique Boolean BI (BBI)

Il convient de présenter en particulier la logique *Boolean BI* ou BBI [63,80] qui est une variante de BI où les opérateurs additifs sont classiques et non intuitionnistes. La syntaxe de BBI est identique à celle de BI mais la sémantique diffère comme nous l'expliquons ici.

### 2.3.1 Sémantique de BBI

La sémantique de BBI est extrêmement proche de celle de BI. La différence principale réside dans la sémantique de l'implication qui est au cœur de la différence entre les logiques classique

et intuitionniste. Dans IL, le caractère particulier de l'implication (qui dénote la constructibilité plus que la conséquence) est exprimée, selon la méthode de Kripke, par la notion de mondes possibles. La relation transitive entre ces mondes est exprimée, dans le cas de BI, par la relation entre ressources que nous avons notée $\sqsubseteq$.

Pour revenir à une sémantique classique, il suffit de considérer tous les mondes comme équivalents en remplaçant la relation de pré-ordre par une relation d'équivalence. Nous allons formaliser cette opération en mettant l'accent sur les changements avec les définitions précédentes.

**Définition 2.13** (Relation d'équivalence)**.** *Soit $E$ un ensemble et $\sim$ une relation sur $E$ ($\sim \subseteq E \times E$). On dit que $\sim$ est une* relation d'équivalence *si elle est transitive, réflexive et symétrique, c'est à dire que pour $e_1, e_2, e_3 \in E$ on a :*

- *$e_1 \sim e_1$ (réflexivité)*
- *si $e_1 \sim e_2$ et $e_2 \sim e_3$, alors $e_1 \sim e_3$ (transitivité)*
- *si $e_1 \sim e_2$ alors $e_2 \sim e_1$ (symétrie)*

**Définition 2.14** (Monoïde de ressources partiel (pour BBI))**.** *Un* monoïde de ressources partiel *est une structure $\mathcal{R} = (R, e, \bullet, \sim)$ telle que :*

- *$(R, e, \bullet)$ est un monoïde partiel commutatif*
- *$\sim$ est une relation d'équivalence sur $R$*
- *$\bullet$ est compatible avec $\sim$. C'est à dire que pour tout $r_1, r_2, s \in R$, si $r_1 \sim r_2$ et $r_2 \bullet s \downarrow$, alors $r_1 \bullet s \downarrow$ et $r_1 \bullet s \sim r_2 \bullet s$*

La notion d'interprétation est la même que pour BI, avec la nuance que la monotonie porte sur la relation $\sim$ et non sur $\sqsubseteq$. On a donc :

$$\forall p \in Prop, \forall r, r' \in R, \text{ si } r \sim r' \text{ et } r \in [\![p]\!] \text{ alors } r' \in [\![p]\!] \text{ (monotonie)}$$

**Définition 2.15** (Modèle (pour BBI))**.** *Un* modèle *de BBI est un triplet $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où $\mathcal{R} = (R, e, \bullet, \sim)$ est un monoïde de ressources partiel, $[\![\cdot]\!]$ est une interprétation de Prop pour $\mathcal{R}$ et $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times R$ une relation de forcing, définie récursivement par les relations suivantes pour tout $p \in Prop, \phi, \psi \in \mathcal{L}$ :*

- *$r \vDash_{\mathcal{K}} \mathrm{I}$ ssi $e \sim r$*
- *$r \vDash_{\mathcal{K}} \phi \to \psi$ ssi $r \vDash_{\mathcal{K}} \phi$ implique $r \vDash_{\mathcal{K}} \psi$*
- *$r \vDash_{\mathcal{K}} \phi * \psi$ ssi il existe $r', r'' \in R$ tels que $r' \bullet r'' \downarrow$ et $r' \bullet r'' \sim r$ et $r' \vDash_{\mathcal{K}} \phi$ et $r'' \vDash_{\mathcal{K}} \psi$*
- *$\vDash_{\mathcal{K}}$ est défini pour tous les autres opérateurs comme dans la Définition 2.7.*

Notons que, si la négation logique est toujours définie par l'égalité $\neg X \equiv X \to \bot$, on peut toutefois se ramener à la définition sémantique plus intuitive de la négation.

**Lemme 2.3.** *Pour tout modèle $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ de BBI, et pour tout $r \in R$, on a :*

$$r \vDash_{\mathcal{K}} \neg\phi \text{ ssi } r \nvDash_{\mathcal{K}} \phi$$

*Démonstration.*
Supposons que $r \vDash_{\mathcal{K}} \neg\phi$. Par définition on a $r \vDash_{\mathcal{K}} \phi \to \bot$. Donc, par la Définition 2.15 on a $r \vDash_{\mathcal{K}} \phi$ implique $r \vDash_{\mathcal{K}} \bot$. Or, on a jamais $r \vDash_{\mathcal{K}} \bot$. Donc on a pas $r \vDash_{\mathcal{K}} \phi$, donc $r \nvDash_{\mathcal{K}} \phi$.

Inversement, si $r \nvDash_{\mathcal{K}} \phi$, alors $r \vDash_{\mathcal{K}} \phi$ est absurde. On a donc jamais $r \vDash_{\mathcal{K}} \phi$, ce qui veut dire que $r \vDash_{\mathcal{K}} \phi$ implique l'absurde, soit $r \vDash_{\mathcal{K}} \phi$ implique $r \vDash_{\mathcal{K}} \bot$, et on a donc $r \vDash_{\mathcal{K}} \phi \to \bot$ ce qui revient bien à $r \vDash_{\mathcal{K}} \neg\phi$. □

Tout comme BI était sémantiquement une extension conservative de IL, BBI est une extension conservative de la logique classique.

**Définition 2.16** (Validité (pour BBI))**.** *On dit qu'une formule $\phi$ de $\mathcal{L}$ est valide dans BBI et on note $\vDash_{BBI} \phi$ (ou simplement $\vDash \phi$ si le contexte n'est pas ambigu) lorsque $r \vDash_{\mathcal{K}} \phi$ pour toute ressource r dans tout modèle $\mathcal{K}$.*

*La conséquence logique est définie comme pour BI (Définition 2.8).*

**Lemme 2.4.** *Pour tout modèle $\mathcal{K}$, pour toutes ressources $r, r'$ de ce modèle et pour toute formule $\phi$, si $r \vDash_{\mathcal{K}} \phi$ et $r \sim r'$, alors $r' \vDash_{\mathcal{K}} \phi$.*

### 2.3.2 Calculs pour BBI et propriétés des sémantiques

Plusieurs calculs ont été proposés pour BBI, dans le prolongement de ceux proposés pour BI.

Pour les calculs des séquents, *calcul des séquents avec labels* [62] a été proposé. Le *calcul de display* [19] propose également une version pour BBI. Enfin, un *calcul des tableaux* avec labels et contraintes existe également pour BBI [66,67]. Nous présenterons ce dernier en détail dans la section 2.4.

Ces différents calculs sont corrects et complets vis-à-vis du formalisme sémantique que nous avons introduit ci-dessus. En outre, la calcul des tableaux a permis de montrer que BBI est indécidable pour les modèles non déterministes, ainsi que déterministes partiels ou totaux [69].

### 2.3.3 Différences avec BI (intuitionniste)

Nous avons posé les bases sémantiques de BI et de BBI, en mettant en lumière les différences inhérentes à leur nature première. Celles-ci peuvent être résumés en deux points majeurs :

- L'implication $\to$ est classique dans BI (elle traduit la conséquence) et intuitionniste dans BBI (elle traduit la constructibilité).
- Les ressources sont comparées par un pré-ordre dans BI et par une relation d'équivalence dans BBI.

Outre ces différences de nature, il convient d'examiner quelles sont les différences plus profondes entre BI et BBI. Il est bon de savoir, notamment, si ces deux logiques si proches l'une de l'autre se valent sur le plan de l'expressivité et sur le plan de l'intérêt formel qu'elles présentent.

Sur ce dernier point, il faut noter que le problème de la validité dans BBI est indécidable pour les modèles non déterministes, ainsi que déterministes partiels ou totaux [69]. Quant à BI, sa décidabilité a été démontrée ainsi que la propriété des modèles finis [49].

En outre, il est aisé de voir, en considérant les Définitions 2.4 et 2.13 que toute relation d'équivalence est un cas particulier de pré-ordre. Il semble donc assez naturel de considérer BBI comme un cas particulier de BI.

Toutefois, BI peut se révéler un outil peu intuitif de modélisation. Tout d'abord, il n'est pas toujours facile de saisir les particularités de l'implication intuitionniste.

Mais cette difficulté n'est pas la seule. En effet, si l'on considère la sémantique de $*$, la nuance entre BI et BBI est la suivante :

- $r \vDash_{BI} A * B$ signifie qu'il existe deux ressources validant l'une $A$ et l'autre $B$, telles que la composition des deux ressources est *une partie* de $r$.
- $r \vDash_{BBI} A * B$ signifie qu'il existe deux ressources validant l'une $A$ et l'autre $B$, telles que la composition des deux ressources est *exactement* $r$.

On voit tout de suite comment, dans le cas de BI, la définition est plus large et laisse une grande partie d'incertitude (à savoir la partie de $r$ qui ne participe pas à la validation de $A$ et $B$). Si cette description est plus flexible, elle peut souvent manquer de précision. Si l'on veut par exemple utiliser BI pour spécifier des espaces mémoire dans l'exécution d'un programme, on aura peut-être plus envie de d'utiliser BBI qui donnera une division explicite de l'espace mémoire en deux parties plutôt que BI qui laisse ouverte la possibilité d'une partie inexploitée (quitte à exprimer cette partie explicitement avec BBI).

Dans le même esprit, I a une signification plus directe dans BBI (la ressource courante *est* la ressource unitaire) que dans BI (la ressource courante *contient* la ressource unitaire).

Ainsi, si BI est décidable et plus générale, on préférera parfois utiliser BBI pour modéliser des problèmes concrets à cause de sa sémantique plus directe et compréhensible.

## 2.4 Un calcul des tableaux pour BBI

Nous allons maintenant présenter en détail le calcul des tableaux de BBI [66].

Les calculs des tableaux sont des méthodes formelles déjà utilisées pour de nombreuses logiques ( [6,44] pour le premier ordre) en particulier pour les logiques modales [54].

Ces calculs consistent à décomposer la négation d'une formule que l'on veut prouver en utilisant des *règles* traduisant la sémantique de la logique en question. la procédure produit un arbre appelé *tableau*, construit selon ces règles, et qui contient l'ensemble des formules produites par décomposition. Pour prouver une formule, on cherchera à *clore* chacune des branches de cet arbre en exhibant dans chacune de ces branches une formules et son opposé logique. Une branche close correspond donc à une contradiction logique, et une clôture complète du tableau amène donc, par un raisonnement ab absurdo, à la validité de la formule initiale.

BI possède d'ors et déjà un calcul des tableaux qui a la particularité d'utiliser des *labels de ressources*. Il utilise des *formules étiquetées* à la fois par des labels de vérité et des labels de ressources, ainsi qu'un système de *contraintes de ressources* introduites à travers les règles par des *affirmations* (assertions), puis utilisées par d'autres règles dans des *exigences* (requirements).

Nous présentons ici une variante adaptée à BBI [66] avec des notations spécifiques que nous utiliserons pour nos propres calculs des tableaux, présentés dans les Chapitres 3, 4 et 5.

Aucune preuve ne sera donnée dans cette présentation puisqu'il s'agit d'une présentation de travaux existants. Le lecteur pourra reconstruire les preuves en s'appuyant de preuves similaires

ou identiques dans [49,66], dans les autres chapitres de cette thèse ou dans les références qui y sont mentionnées.

### 2.4.1 Labels et contraintes

Comme nous l'avons mentionné, le calcul des tableaux pour BBI utilise des *labels* qui seront utilisés comme des marqueurs pour les ressources de la sémantique de BBI. Nous commençons par introduire formellement ces labels.

Puisque les labels doivent symboliser les ressources, ils doivent hériter de la structure de celles-ci (à savoir la composition interne). On pourra pour cela introduire une composition entre labels similaire à celle entre ressources. Dans la version présentée ici, on préférera présenter les labels comme des mots dont les lettres sont les labels atomiques.

**Définition 2.17** (Labels)**.** *On considère un ensemble de constantes $\gamma_r$ ( usuellement noté $\gamma_r = \{c_1, c_2, \ldots\}$.*

*Un* label de ressource *est un mot construit sur $\gamma_r$ où l'ordre des lettres n'est pas pris en compte (c'est à dire un multi-ensemble de constantes). On note $\varepsilon$ le mot vide.*

*On dit qu'un label $x$ est un* sous-label *de $y$ s'il existe $z$ tel que $y = xz$. L'ensemble des sous-labels de $x$ est noté $\mathcal{E}(x)$.*

On voit ainsi comment la structure de mots permet de retrouver les propriétés structurelles des ressources, à savoir la commutativité et l'associativité (puisque l'ordre des lettres n'importe pas) ainsi que la présence d'un élément neutre (en l'occurrence, le mot vide $\varepsilon$).

Il nous faut également retranscrire le fait que les ressources des modèles de BBI peuvent être mis en relation par la relation d'équivalence $\sim$. Pour cela, on introduit la notion de *contrainte*. Une contrainte représentera une telle équivalence entre ressources [1].

**Définition 2.18** (Contraintes)**.** *Une* contrainte de ressources *est une expression de la forme $x \simeq y$, où $x$ et $y$ sont des labels de ressources.*

*Un* ensemble de contraintes *est un ensemble $\mathcal{C}$ qui contient des contraintes de ressources.*

*Soit $\mathcal{C}$ un ensemble de contraintes. Le* domaine *de $\mathcal{C}$, noté $\mathcal{D}_r(\mathcal{C})$ est l'ensemble de tous les sous-labels de ressource apparaissant dans $\mathcal{C}$. On a donc :*

$$\mathcal{D}_r(\mathcal{C}) = \bigcup_{x \simeq y \in \mathcal{C}} (\mathcal{E}(x) \cup \mathcal{E}(y))$$

*L'*alphabet *de $\mathcal{C}$, noté $\mathcal{A}_r(\mathcal{C})$ est l'ensemble de toutes les constantes de ressources apparaissant dans $\mathcal{C}$, c'est à dire :*

$$\mathcal{A}_r(\mathcal{C}) = \gamma_r \cap \mathcal{D}_r(\mathcal{C})$$

Enfin, pour que les propriétés de $\sim$ soient correctement capturées par son analogue $\simeq$, on introduit la notion de clôture d'un ensemble de contraintes.

**Définition 2.19** (Clôture de contraintes)**.** *Soit $\mathcal{C}$ un ensemble de contraintes. La clôture de $\mathcal{C}$, notée $\overline{\mathcal{C}}$, est le plus petit ensemble clos par les règles de la Figure 2.3 et tel que $\mathcal{C} \subseteq \overline{\mathcal{C}}$.*

1. Dans le cas de BI, les contraintes seront plutôt une traduction de la relation de pré-ordre $\sqsubseteq$.

$$\frac{}{\varepsilon \simeq \varepsilon}\ \langle \varepsilon \rangle \qquad \frac{x \simeq y}{y \simeq x}\ \langle s_r \rangle \qquad \frac{xy \simeq xy}{x \simeq x}\ \langle d_r \rangle$$

$$\frac{x \simeq y \qquad yk \simeq yk}{xk \simeq yk}\ \langle c_r \rangle \qquad \frac{x \simeq y \qquad y \simeq z}{x \simeq z}\ \langle t_r \rangle$$

FIGURE 2.3 – Règles de clôture des contraintes

On voit comment les règles $\langle \varepsilon \rangle$ et $\langle d_r \rangle$ traduisent la réflexivité de $\simeq$, la règle $\langle s_r \rangle$ en traduit la symétrie, la règle $\langle t_r \rangle$ traduit la transitivité et enfin la règle $\langle c_r \rangle$ traduit la composition à droite. On retrouve bien toutes les propriétés de $\sim$.

On peut également introduire des règles alternatives, déduites des précédentes, pour faciliter le calcul de la clôture d'un ensemble de contraintes.

**Proposition 2.1.** *Les règles suivantes peuvent être déduites des règles de la Figure 2.3*

$$\frac{xk \simeq y}{x \simeq x}\ \langle p_l \rangle \qquad \frac{x \simeq yk}{y \simeq y}\ \langle p_r \rangle$$

On a en outre quelques propriétés sur les ensembles de contraintes. Celles-ci sont particulièrement utiles pour raisonner sur les tableaux de BBI, par exemple pour montrer leur correction et leur complétude (ce que nous ne ferons pas ici).

**Corollaire 2.1.** *Soit $\mathcal{C}$ un ensemble de contraintes.*

1. *$x \in \mathcal{D}_r(\overline{\mathcal{C}})$ ssi $x \simeq x \in \overline{\mathcal{C}}$.*
2. *Si $xy \in \mathcal{D}_r(\overline{\mathcal{C}})$, $x' \simeq x \in \overline{\mathcal{C}}$, et $y' \simeq y \in \overline{\mathcal{C}}$, alors $xy \simeq x'y' \in \overline{\mathcal{C}}$.*

**Proposition 2.2.** *Soit $\mathcal{C}$ un ensemble de contraintes. On a $\mathcal{A}_r(\mathcal{C}) = \mathcal{A}_r(\overline{\mathcal{C}})$.*

**Proposition 2.3.** *Soit $\mathcal{C}$, $\mathcal{C}'$ deux ensembles de contraintes tels que $\mathcal{C} \subseteq \mathcal{C}'$. Si $x \simeq y \in \overline{\mathcal{C}}$, alors $x \simeq y \in \overline{\mathcal{C}'}$.*

**Lemme 2.5.** *Soit $\mathcal{C}$ un ensemble de contraintes (possiblement infini). Si $x \simeq y \in \overline{\mathcal{C}}$, alors il existe un ensemble fini $\mathcal{C}_f$ tel que $\mathcal{C}_f \subseteq \mathcal{C}$ et $x \simeq y \in \overline{\mathcal{C}_f}$.*

Nous pouvons maintenant formaliser ce que sont les tableaux pour BBI.

### 2.4.2 Règles et tableaux

Tout d'abord, nous allons présenter la structure de *formule étiquetée*. Celle-ci nous permet d'associer une formule de BBI à un label de ressource (représentant la ressource pouvant potentiellement valider la formule) et à une étiquette de vérité (indiquant que la formule est potentiellement valide ou invalide).

On définit en outre une structure d'ensemble d'énoncés contraints, ou *CSS*, qui regroupe un ensemble de formule étiquetées et de contraintes. Ce sont ces ensembles qui vont former le tableau.

**Définition 2.20** (Formule étiquetée, CSS)**.** *Une* formule étiquetée *est un triplet* $(S,\phi,x)$ *tel que* $S \in \{\mathbb{T},\mathbb{F}\}$, $\phi \in \mathcal{L}$ *est une formule de BBI et* $x \in L_r$ *est un label de ressource. On note une telle formule* $\mathbb{S}\phi : x$.

*Un* ensemble d'énoncés contraints *(noté CSS pour Constrained Set of Statement) est une paire* $\langle \mathcal{F}, \mathcal{C} \rangle$*, où* $\mathcal{F}$ *est un ensemble de formules étiquetées et* $\mathcal{C}$ *est un ensemble de contraintes, et qui satisfait la propriété* $P_{CSS}$ *définie par :*

$$si\ \mathbb{S}\phi : x \in \mathcal{F}\text{, alors } x \simeq x \in \overline{\mathcal{C}}\ \ (P_{CSS}).$$

*Un CSS* $\langle \mathcal{F}, \mathcal{C} \rangle$ *est* fini *si* $\mathcal{F}$ *et* $\mathcal{C}$ *sont finis.*
*La relation* $\preccurlyeq$ *est définie par* $\langle \mathcal{F}, \mathcal{C} \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}' \rangle$ *si* $\mathcal{F} \subseteq \mathcal{F}'$ *et* $\mathcal{C} \subseteq \mathcal{C}'$.
*On écrit* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C} \rangle$ *lorsque* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle \preccurlyeq \langle \mathcal{F}, \mathcal{C} \rangle$ *et que* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle$ *est fini.*

Intuitivement, si une formule $\mathbb{T}\phi : x$ appartient à notre tableau, c'est qu'on doit avoir un modèle tel que $x \vDash \phi$. Cette intuition est formalisée dans la preuve de correction (par la notion de réalisation d'un CSS) et dans la méthode d'extraction de contre-modèles.

Les CSS sont potentiellement infinis, notamment quand aux ensembles de contraintes, ce qui peut être problématiques dans les différentes preuves. Toutefois, on a la propriété suivante.

**Proposition 2.4.** *Pour tout* $\langle \mathcal{F}_f, \mathcal{C} \rangle$*, où* $\mathcal{F}_f$ *est fini, il existe* $\mathcal{C}_f \subseteq \mathcal{C}$ *tel que* $\mathcal{C}_f$ *est fini et* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle$ *est un CSS.*

Nous pouvons maintenant énoncer les règles qui guideront notre calcul des tableaux. Ces règles sont rassemblées dans la Figure 2.4.

Dans cette figure, la note '$c_i$ et $c_j$ sont de nouvelles constantes de label' signifie que $c_i \neq c_j \in \gamma_r$.

Jusqu'à la fin de cette thèse, on note $\oplus$ la concaténation entre deux listes d'éléments. Par exemple, on a $[e_1,e_2,e_3] \oplus [e_4,e_5] = [e_1,e_2,e_3,e_4,e_5]$.

L'utilisation formelle des règles, et la construction d'un tableau sont alors dictées par la définition suivante :

**Définition 2.21** (Tableau pour BBI)**.** *Soit* $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ *un CSS fini. Un* tableau *pour* $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ *est une liste de CSS, appelés* branches*, construite de manière inductive de la manière suivante :*

1. *La liste à une branche* $[\langle \mathcal{F}_0, \mathcal{C}_0 \rangle]$ *est un tableau pour* $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ *;*
2. *Si la liste* $\mathcal{T}_m \oplus [\langle \mathcal{F}, \mathcal{C} \rangle] \oplus \mathcal{T}_n$ *est un tableau pour* $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ *et*

$$\frac{cond(\langle \mathcal{F}, \mathcal{C} \rangle)}{\langle \mathcal{F}_1, \mathcal{C}_1 \rangle \mid \ldots \mid \langle \mathcal{F}_k, \mathcal{C}_k \rangle}$$

*est une instance d'une règle de la Figure 2.4 pour laquelle la condition* $cond(\langle \mathcal{F}, \mathcal{C} \rangle)$ *est remplie, alors la liste* $\mathcal{T}_m \oplus [\langle \mathcal{F} \cup \mathcal{F}_1, \mathcal{C} \cup \mathcal{C}_1 \rangle; \ldots; \langle \mathcal{F} \cup \mathcal{F}_k, \mathcal{C} \cup \mathcal{C}_k \rangle] \oplus \mathcal{T}_n$ *est un tableau pour* $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$.

*Un tableau pour la formule* $\phi$ *est un tableau pour* $\langle \{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\} \rangle$.

Un tableau est donc un ensemble de CSS qui est successivement étendu par les règles de la Figure 2.4. Cela veut dire que si la formule et le cas échéant la contrainte indiqué dans les prémisses de la règle sont présents dans le CSS, on l'étend par les formules et contraintes

$$\frac{\mathbb{T}\mathrm{I} : x \in \mathcal{F}}{\langle \emptyset, \{x \simeq \varepsilon\} \rangle} \ \langle \mathbb{T}\mathrm{I} \rangle$$

$$\frac{\mathbb{T}\neg\phi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset \rangle} \ \langle \mathbb{T}\neg \rangle \qquad \frac{\mathbb{F}\neg\phi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x\}, \emptyset \rangle} \ \langle \mathbb{F}\neg \rangle$$

$$\frac{\mathbb{T}\phi \wedge \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x, \mathbb{T}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{T}\wedge \rangle \qquad \frac{\mathbb{F}\phi \wedge \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset \rangle \mid \langle \{\mathbb{F}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{F}\wedge \rangle$$

$$\frac{\mathbb{T}\phi \vee \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x\}, \emptyset \rangle \mid \langle \{\mathbb{T}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{T}\vee \rangle \qquad \frac{\mathbb{F}\phi \vee \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x, \mathbb{F}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{F}\vee \rangle$$

$$\frac{\mathbb{T}\phi \to \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset \rangle \mid \langle \{\mathbb{T}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{T}\to \rangle \qquad \frac{\mathbb{F}\phi \to \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x, \mathbb{F}\psi : x\}, \emptyset \rangle} \ \langle \mathbb{F}\to \rangle$$

$$\frac{\mathbb{T}\phi * \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i, \mathbb{T}\psi : c_j\}, \{x \simeq c_i c_j\} \rangle} \ \langle \mathbb{T}* \rangle \qquad \frac{\mathbb{F}\phi * \psi : x \in \mathcal{F} \text{ et } x \simeq yz \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\}, \emptyset \rangle \mid \langle \{\mathbb{F}\psi : z\}, \emptyset \rangle} \ \langle \mathbb{F}* \rangle$$

$$\frac{\mathbb{T}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F} \text{ et } xy \simeq xy \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\}, \emptyset \rangle \mid \langle \{\mathbb{T}\psi : xy\}, \emptyset \rangle} \ \langle \mathbb{T}\mathrel{-\!\!*} \rangle \qquad \frac{\mathbb{F}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i, \mathbb{F}\psi : xc_i\}, \{xc_i \simeq xc_i\} \rangle} \ \langle \mathbb{F}\mathrel{-\!\!*} \rangle$$

Note : $c_i$ et $c_j$ sont des nouvelles constantes de label.

FIGURE 2.4 – Règles pour le calcul des tableaux pour BBI

présentées dans la conclusion. Si la règle présente deux conclusions distinctes (séparées par $|$), deux nouveaux CSS sont générés.

On remarque que toutes ces règles correspondent en réalité à la définition sémantique des différents opérateurs, selon l'intuition qu'il y a un lien entre la validité d'une formule et ses étiquettes de vérité dans les CSS. Pour être plus précis, si la prémisse d'une règle est vérifiée, alors la Définition 2.15 assure que l'une des conclusions l'est aussi.

On montrera facilement que le CSS de départ pour une formule vérifie la propriété $P_{CSS}$ de la Définition 2.20, et également que toutes les règles de la Figure 2.4 conservent cette propriété.

Parmi les règles de la Figure 2.4, certaines introduisent de nouveaux labels et de nouvelles contraintes ($\langle \mathbb{T}\mathbf{I} \rangle$, $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}-\!\!* \rangle$).

D'autres règles requièrent la présence de contraintes spécifiques ($\langle \mathbb{F}* \rangle$ et $\langle \mathbb{T}-\!\!* \rangle$).

De manière assez logique, nous devons donc respecter un *ordre d'application* des règles. Les règles de la première catégorie (qui produisent labels et contraintes) doivent être autant que faire ce peut appliquées avant celles de la deuxième (qui en consomment). Les autres règles peuvent être appliquées indifféremment, mais on préférera toujours appliquer la deuxième catégorie le plus tard possible.

Définissons maintenant les tableaux clos.

**Définition 2.22** (Conditions de clôture)**.** *Un CSS $\langle \mathcal{F}, \mathcal{C} \rangle$ est* clos *si l'une des conditions suivantes tient, pour n'importe quelle formule $\phi \in \mathcal{L}$, et n'importe quels labels de ressource $x, y$ :*

1. $\mathbb{T}\phi : x \in \mathcal{F}$, $\mathbb{F}\phi : y \in \mathcal{F}$ *et* $x \simeq y \in \overline{\mathcal{C}}$ *;*
2. $\mathbb{F}\mathbf{I} : x \in \mathcal{F}$ *et* $x \simeq \varepsilon \in \overline{\mathcal{C}}$ *;*
3. $\mathbb{F}\top : x \in \mathcal{F}$ *;*
4. $\mathbb{T}\bot : x \in \mathcal{F}$.

*Un CSS est* ouvert *s'il n'est pas clos. Un tableau est* clos *si toutes ses branches sont closes. Une* preuve *pour une formule $\phi$ est un tableau clos pour $\phi$.*

D'après l'intuition que nous avons donnée plus haut (à savoir qu'une formule $\mathbb{T}x : \phi$ appartient à notre tableau s'il y a un modèle tel que $x \vDash \phi$), ces quatre conditions correspondent donc à des cas absurdes. Ainsi un CSS clos est intuitivement un CSS qui n'admet pas de modèle.

Si tous les CSS d'un tableau sont clos, alors la formule initiale de ce dernier ne peut avoir de modèle. Comme le CSS initial pour une preuve de $\phi$ est $\langle \{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\} \rangle$, on a donc que $\mathbb{F}\phi : c_1$ n'a pas de modèle et donc par suite que $\phi$ est valide. Ceci justifie qu'une preuve par ce calcul permet d'établir la validité d'une formule de BBI.

### 2.4.3 Un exemple de BBI-tableau

Nous allons enfin présenter un exemple de BBI-tableau pour illustrer le calcul. La compréhension de cette exemple aidera le lecteur à se familiariser avec les calculs similaires présentés dans les chapitres suivants.

On se propose de dresser le tableau pour la formule $(A * (A -\!\!* B)) \to B$. On doit donc, selon la Définition 2.21, commencer par le CSS suivant :

$$\langle \{\mathbb{F}(A * (A -\!\!* B)) \to B : c_1\}, \{c_1 \simeq c_1\} \rangle$$

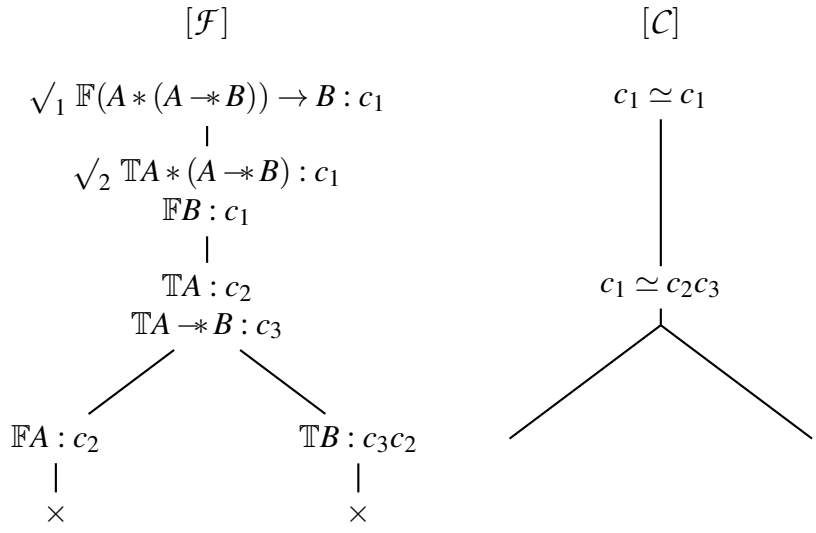



FIGURE 2.5 – BBI-Tableau pour $(A*(A\mathrel{-\!\!*}B))\to B$

Pour plus de commodité, on représentera les CSS sous forme de deux arbres évoluant simultanément : l'un contenant les formules étiquetées, l'autre les contraintes de ressource. Toute extension du CSS générera un nouveau nœud de l'arbre, et dans le cas d'une règle qui génère plusieurs CSS, on représentera par deux branches distinctes les parties qui diffèrent entre les deux CSS.

Ainsi, le CSS initial sera représenté comme suit :

| $[\mathcal{F}]$ | $[\mathcal{C}]$ |
|---|---|
| $\mathbb{F}(A*(A\mathrel{-\!\!*}B))\to B : c_1$ | $c_1 \simeq c_1$ |

On doit ensuite, pour développer le tableau, appliquer l'une des règles de la Figure 2.4 sur ce CSS initial. Pour suivre l'évolution du tableau, on marque par un $\surd$ les formules déjà développée par une règle et on ajoute un indice désignant l'ordre de ce développement. La Figure 2.5 présente l'intégralité du tableau développé en suivant les règles du calcul des tableaux.

Les étapes de construction du tableau sont les suivantes :

1. Pour le moment, seule la règle $\langle\mathbb{F}\to\rangle$ peut être appliquée. On l'applique donc, ce qui produit deux nouvelles formules au sein du même CSS.
2. On doit ensuite utiliser la règle $\langle\mathbb{T}*\rangle$, ce qui produit à nouveau deux formules dans notre CSS, mais aussi une nouvelle contrainte.
3. On doit maintenant appliquer la règle $\langle\mathbb{T}\to\rangle$ sur la formule $\mathbb{T}A\mathrel{-\!\!*}B : c_3$. Pour cela, il faut trouver un labels de ressource $y$ tel que $c_3y \simeq c_3y \in \overline{\mathcal{C}}$. Or, comme $c_1 \simeq c_2c_3 \in \mathcal{C}$, par le Corollaire 2 on peut déduire que $c_3c_2 \simeq c_3c_2 \in \overline{\mathcal{C}}$. On peut donc appliquer la règle avec $y = c_2$, ce qui produit deux CSS distincts.

À ce stade, on peut clore chacune des branches produites. En effet, dans la branche de gauche on a $\mathbb{T}A : c_2 \in \mathcal{F}$ et $\mathbb{F}A : c_2 \in \mathcal{F}$. En outre, par $P_{CSS}$ on a $c_2 \simeq c_2 \in \overline{\mathcal{C}}$. Donc cette

branche est close par la condition 1 de la Définition 2.22. De même, la branche de droite présente $\mathbb{T}B : c_3c_2 \in \mathcal{F}$, $\mathbb{F}B : c_1 \in \mathcal{F}$ et $c_1 \simeq c_3c_2 \in \overline{C}$. Donc par le même raisonnement cette branche est aussi close.

Finalement, toutes nos branches sont closes et on a une preuve pour notre formule.

### 2.4.4 Propriétés du calcul des tableaux de BBI

Les propriétés de ce calcul ont été démontrées dans [66]. Elles consistent principalement en la correction et la complétude du calcul des tableaux, formalisées par les théorèmes suivants :

**Théorème 2.1** (Correction)**.** *S'il existe une preuve pour une formule $\phi$ de BBI,alors elle est valide.*

Ce premier théorème est prouvé par la notion de *réalisabilité* d'une branche du tableau. Une branche est réalisable si on peut trouver un modèle qui en traduise les propriétés, c'est à dire que si $\mathbb{T}\phi : x$ est dans le tableau, on doit avoir $x \vDash \phi$ et si $\mathbb{F}\phi : x$ est dans le tableau, on doit avoir $x \nvDash \phi$. On montre que les branches closes ne sont pas réalisables et que l'extension par les règles de la Figure 2.4 préserve la réalisabilité d'une branche. Dès lors, un bref raisonnement montre que le CSS initial $\langle\{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\}\rangle$ d'un tableau ne peut être réalisable, et par contraposée que la formule $\phi$ qu'il contient doit être valide.

**Théorème 2.2** (Complétude)**.** *S'il une formule $\phi$ de BBI est valide, alors il en existe une preuve.*

La propriété de complétude est bien plus ardue à montrer. La preuve traditionnelle consiste à extraire un contre-modèle d'une branche non close (montrant ainsi qu'une formule n'ayant pas de preuve n'est pas valide). Le problème est que la génération de contraintes par les règles $\langle\mathbb{T}\mathrm{I}\rangle$, $\langle\mathbb{T}*\rangle$ et $\langle\mathbb{F}-\!\!*\rangle$ peut aboutir à la génération de branches à la fois non closes et infinies.

Il faut donc une manière de générer un contre-modèle à partir de branches non closes, potentiellement infinies. Le problème principal avec cet objectif est que l'introduction des labels et des contraintes n'est pas contrôlée, il est donc impossible de généraliser la manière de construire un contre-modèle d'une branche infinie. Pour résoudre ce problème, on reconstruit toute branche non close grâce à deux structures :

- une *stratégie équitable* qui est une liste de formules et/ou de contraintes où chaque élément se répète un nombre infini de fois ;
- un *oracle* qui est un ensemble de CSS clos pour l'inclusion, saturés et finis contenant tout les CSS ne menant pas à une preuve.

Avec ces deux outils, on peut générer des formules (par la stratégie) et sélectionner celles qui conduisent à un tableau non clos (par l'oracle). Ceci permet de reconstruire de manière contrôlée une branche non close à partir d'une formule donnée, et ainsi d'obtenir un contre-modèle.

Cette manière de produire la preuve de complétude est efficace mais très complexe. Un des défis principaux des différentes extensions qui ont été développées à partir de BI et BBI (y compris celles présentées dans cette thèse) est de réussir à appliquer ce schéma de preuve aux nouveaux formalismes introduits.

## 2.5 Modélisation de systèmes avec les logiques BBI et BI : un exemple

Comme nous l'expliciterons plus tard, BI et BBI ont un large pouvoir d'expression qui permet de modéliser de nombreux types de systèmes [2,3,21,28,63,75,87,87].

Nous voulons ici illustrer ces possibilités par un exemple de modélisation de consommation de ressources. Cet exemple est assez trivial a pas pour but de rendre plus claire la sémantique des opérateurs multiplicatifs propres à BI et BBI et souligner les différences entre BI et BBI.

L'exemple de cette section est largement inspiré de ceux présentés dans [32].

### 2.5.1 Ressources et consommation avec BBI

Nous avons explicité comment dans BBI comme dans BI on a $A \nvdash A * A$ et $A * B \nvdash A$, ce qui exprime respectivement qu'on ne peut pas dupliquer ni supprimer une formule d'une conjonction multiplicative.

Concrètement, ces propriétés permettent de modéliser aisément des systèmes où la consommation et la production sont centrale. Prenons un exemple simple de vente pour expliciter ce phénomène.

Commençons par montrer comment les opérateurs additifs ne sont pas suffisants pour exprimer, par exemple, les propriétés de la consommation d'argent. Si la proposition $E$ signifie "je possède 1 euro", alors l'utilisation de connecteurs additifs ne permet de modéliser aucune notion d'économie. En effet, si l'on veut modéliser le fait de posséder 2 euros, et ce uniquement avec les additifs, on se tourne vers la proposition $E \wedge E$ (j'ai 1 euro et 1 autre euro). Or, on a $E \vDash E \wedge E$, puisque les opérateurs additifs permettent la contraction. Ceci n'est pas ce que nous attendions puisque (malheureusement) 1 euro n'est pas équivalent à 2 euros.

La conjonction multiplicative permet, elle d'obtenir le résultat escompté. Comme mentionné plus haut, $E \nvdash E * E$, et l'opérateur $*$ nous permet ainsi de modéliser l'existence de deux entité distinctes partageant la même propriété.

Nous allons développer un exemple plus précis avec BBI pour illustrer plus en avant ce fait. Ce sera également l'occasion pour rendre plus claire la sémantique de cette logique.

Si, dans le paragraphe ci-dessus, on a modélisé la possession d'une somme d'argent par une proposition, on préférera avec BBI utiliser pour cela des ressources (contrairement à ce qui peut être fait par exemple, en logique linéaire, voir la Section 2.1).

Pour être plus précis, nous avons dans BBI et sa sémantique deux types d'entités, les ressources et les propositions (celles-ci pouvant être des propositions atomiques de l'ensemble $Prop$ ou des formules composées à partir de celles-ci). Pour modéliser l'achat d'objets, on se propose d'utiliser les ressources pour modéliser l'argent et les propositions pour les objets à acheter.

Commençons par définir notre ensemble de ressources. Les ressources atomiques modéliseront différentes pièces utilisées pour l'achat. On considère l'ensemble suivant :

$$Res = \{e, p_1, p_2, c_{50}, c_{20}, c_{10}\}$$

Dans cet ensemble, on retrouve tout d'abord la ressource unitaire $e$ qui, étant l'élément neutre de notre composition, marque l'absence d'argent ou, si l'on veut, une pièce sans valeur.

Les ressources $p_1$ et $p_2$ représentent des pièces de 1 et 2 euros tandis que les $c_{50}$, $c_{20}$, $c_{10}$ représentent des pièces de 50, 20 et 10 cents.

Une ressource simple (ou atomique) est donc une pièce, une ressource composée sera une somme d'argent composée de plusieurs pièces. La composition $\bullet$ représente donc simplement l'association de plusieurs ensembles de pièces en un ensemble plus grand. Par exemple, la ressource $p_1 \bullet c_{50} \bullet c_{50} \bullet c_{20}$ est une somme composée d'une pièce de 1 euro, d'une pièce de 20 cents et de deux pièces de 50 cents.

Pour capturer l'ensemble des ressources constructibles, on va simplement clore notre ensemble *Res* de ressources unitaires par la composition. On a donc :

$$R = \{r \bullet r' | r, r' \in Res\}$$

On notera que $Res \subset R$ puisqu'on a pour tout $r \in Res$ que $r \bullet e = r$.

Deux ressources sont équivalentes si elles représentent la même somme d'argent. Pour formaliser ce fait, nous allons introduire une fonction intermédiaire qui capturera le coût d'une ressource. On définit donc une fonction $val : R \to \mathbb{R}$ qui donnera la valeur d'une ressources en euros. Plus précisément, on a :

- $val(e) = 0$ ;
- pour tout $i$, $val(p_i) = i$ ;
- pour tout $i$, $val(c_i) = 0.1 \times i$ ;
- pour tout $r, r' \in R$, $val(r \bullet r') = val(r) + val(r')$.

Enfin, on peut définir notre relation d'équivalence $\sim$ par :

$$r \sim r' \text{ ssi } val(r) = val(r')$$

Nous avons donc un monoïde de ressources partiel $\mathcal{R} = (R, e, \bullet, \sim)$ correctement défini qui modélise un ensemble de pièces et toutes les combinaisons réalisables à partir de lui.

Nous allons maintenant spécifier des propositions atomiques. Si les ressources modélisent en général des objets, les propositions modélisent des propriétés sur ces objets. C'est ce que nous allons faire ici. Plus précisément, nos propositions décriront la possibilité d'acheter des objets. On se donne l'ensemble de propositions atomiques suivant :

$$Prop = \{Obj_{(0.30)}, Obj_{(1.70)}, Obj_{(2)}\}$$

Dans cet ensemble, chaque proposition $Obj_{(x)}$ désigne la possibilité d'acheter exactement un objet qui coûte $x$ euros (pour simplifier le problème, on considérera que posséder plus que la somme requise ne convient pas). Pour formaliser cela, nous définissons une fonction d'interprétation qui liera nos ressources à nos propositions. Il nous suffit d'utiliser notre fonction de valeur *val* définie ci-dessus de la manière suivante :

$$\text{Pour tout } x, [\![Obj_{(x)}]\!] = \{r \in R / val(r) = x\}$$

Par exemple, $[\![Obj_{(0.30)}]\!] = \{c_{20} \bullet c_{10}, c_{10} \bullet c_{10} \bullet c_{10}\}$, ce qui signifie que pour acheter un objet à 30 cents, je peux payer avec une pièce de 20 et une de 10 cents ou avec trois pièces de 10 cents. Notez que nous ne différencions pas $c_{20} \bullet c_{10}$ et $c_{10} \bullet c_{20}$ puisque, selon notre définition d'un modèle, $\bullet$ est commutative et les deux sont donc équivalent.

Enfin, nous complétons notre modèle avec la relation de forcing $\vDash_{\mathcal{K}}$ définie comme indiqué dans la Définition 2.15. Cette relation nous permet de saisir plus en avant la signification de nos opérateurs, dans le cadre très particulier de ce modèle.

- Pour tout $p \in Prop$, $r \vDash_{\mathcal{K}} p$ si $r \in [\![p]\!]$. Comme les éléments de $Prop$ sont les $Obj_{(x)}$, et vu notre définition de $[\![\cdot]\!]$, on en fait $r \vDash_{\mathcal{K}} Obj_{(x)}$ si $val(r) = x$, ce qui nous dit bien que la ressource $r$ permet d'acheter l'objet $Obj_{(x)}$. De manière plus générale, on peut d'ors et déjà inférer que $r \vDash \phi$ signifiera que la somme représentée par l'ensemble de pièces $r$ permet d'effectuer l'opération décrite par la proposition $\phi$.
- $r \vDash_{\mathcal{K}} \mathrm{I}$ si $r \sim e$. C'est à dire que I représente un objet "gratuit" puisqu'on peut l'obtenir avec $e$, représentant l'absence de pièce. De manière plus large, $e$ signifie souvent l'absence de ressource ou d'information, et I est donc la proposition qu'on peut obtenir sans aucune ressource.
- $r \vDash_{\mathcal{K}} \top$ toujours. $\top$ représente donc une opération pouvant être effectuée avec n'importe quelle somme. La nuance entre $\top$ et I est subtile, car ce sont tous deux des opérateurs neutres, mais le premier l'est pour $\wedge$ et le deuxième pour $*$. D'un point de vue calculatoire, $\top$ peut donc être effacé de toute séquence de $\wedge$ et I de toute séquence de $*$.
- $r \vDash_{\mathcal{K}} \bot$ jamais. $\bot$ représente donc un objet qui ne peut être acheté.
- $r \vDash_{\mathcal{K}} \phi \wedge \psi$ si $r \vDash_{\mathcal{K}} \phi$ et $r \vDash_{\mathcal{K}} \psi$, c'est à dire que la valeur de $r$ permet d'effectuer à la fois l'opération $\phi$ et l'opération $\psi$. C'est là toutefois que le langage naturel est quelque peu limitant. L'intuition de "effectuer $\phi$ et $\psi$" tend à faire croire que la somme représentée par $r$ peut être divisée en deux sommes pour effectuer les deux opérations. C'est en réalité le sens de l'opérateur $*$, comme nous le verrons plus loin. Ici il s'agit plutôt d'effectuer $\phi$ ou $\psi$, dans le sens où $r$ permet d'effectuer exactement les deux opérations en entier. Ainsi, de manière contre-intuitive, $\wedge$ prend plutôt le sens d'un "ou" dans notre contexte, mais est distincte du "ou" de $\vee$ comme nous allons l'expliquer maintenant.
- $r \vDash_{\mathcal{K}} \phi \vee \psi$ si $r \vDash_{\mathcal{K}} \phi$ ou $r \vDash_{\mathcal{K}} \psi$, c'est à dire que la valeur de $r$ permet d'effectuer à la fois l'opération $\phi$ ou l'opération $\phi$. Toutefois, contrairement au cas précédent, le "ou" énoncé ici est un ou logique. On affirme que l'une des deux opérations au moins est réalisable, mais sans certitude que les deux le sont (comme dans le cas $\wedge$).
- $r \vDash_{\mathcal{K}} \phi \to \psi$ si chaque fois que $r \vDash_{\mathcal{K}} \phi$, alors $r \vDash_{\mathcal{K}} \psi$, c'est à dire que si la valeur de $r$ permet d'effectuer l'opération $\phi$, alors elle permet aussi d'effectuer l'opération $\phi$.
- $r \vDash_{\mathcal{K}} \phi * \psi$ s'il existe $r_1, r_2$ tels que $r \sim r_1 \bullet r_2$, $r_1 \vDash_{\mathcal{K}} \phi$ et $r_2 \vDash_{\mathcal{K}} \psi$, c'est à dire que la somme représentée par $r$ peut être divisée en deux sommes, l'une permettant d'effectuer $\phi$, l'autre permettant d'effectuer $\psi$. On retrouve le sens plus intuitif de "$r$ permet d'effectuer $\phi$ et $\phi$". Ce "et" semble donc le plus naturel dans ce contexte financier.
- $r \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ si pour tout $r'$ tel que $r' \vDash_{\mathcal{K}} \phi$ on a $r \bullet r' \vDash_{\mathcal{K}} \psi$. Ici, on se trouve dans le cas de la modélisation d'une transaction payante : la "transformation" de $\phi$ en $\psi$ nécessite l'apport de $r$. Ainsi, si une somme permet d'effectuer $\phi$, alors cette somme augmentée de la valeur de $r$ permet d'effectuer $\psi$.

En combinant ces opérateurs, on peut alors modéliser des situations complexes. Par exemple, on peut établir la formule suivante :

$$c_{10} \bullet c_{10} \bullet c_{10} \vDash_{\mathcal{K}} Obj_{1.70} \mathrel{-\!\!*} (Obj_2 \wedge (Obj_{1.70} * Obj_{0.30}))$$

Cette formule est valide, en la développant on constate qu'elle exprime que 3 pièces de 10 centimes, additionnées à des pièces permettant d'acheter un objet à 1,70 euros permettent à la fois d'acheter un objet à 2 euros et deux objets l'un à 1,70 euros, l'autre à 30 centimes.

Nous revisiterons cet exemple (ou sa variation intuitionniste, présentée ci-dessous) dans les chapitres suivants pour voir en quoi nos extensions et variantes de BI et BBI permettent d'en étendre l'expressivité. Dans le chapitre 3, nous ajouterons des opérateurs temporels pour décrire un système d'achat évoluant dans le temps. Dans le chapitre 4, nous ajouterons des opérateurs épistémique permettant de décrire l'accès d'agents à une opération moyennant une ressource. Enfin, dans le chapitre 5, nous verrons comment l'internalisation des ressources étend l'expressivité en permettant de raisonner au même niveau sur les propositions et les ressources.

### 2.5.2 Ressources et consommation avec BI

Pour mettre en lumière les différences entre BI et BBI, on se propose de reprendre l'exemple précédent en le modifiant légèrement pour utiliser BI à la place de BBI comme logique de modélisation.

La différence de structures entre BI et BBI réside dans le fait que BI utilise une relation de pré-ordre là où BBI utilisait une équivalence. Par conséquent, nous devons adapter notre modèle pour trouver une relation d'ordre entre les ressources.

Cet objectif n'est pas très complexe. En effet, nous avons déjà vu dans l'exemple précédent comment la fonction *val* nous permettait d'associer à chaque ressource un nombre réel représentant la somme d'argent que valent les ressources (c'est à dire les pièces) qui la composent. Il nous suffit alors d'utiliser la relation d'ordre usuelle sur les nombres réels pour ordonner nos ressources. Formellement, on introduit notre relation d'ordre de la manière suivante :

$$\text{Pour tout } r, r' \in R, r \sqsubseteq r' \text{ si } val(r) \leq val(r')$$

L'interprétation de la validité d'une formule est légèrement biaisée dans notre modélisation avec BI. En effet, avec BBI, $r \vDash \phi$ signifiait que la somme représentée par $r$ permettait d'effectuer exactement $\phi$. Dans notre cas actuel, nous devons modifier légèrement cette affirmation pour rester en cohérence à la fois avec l'interprétation de l'implication et la monotonie de l'interprétation.

Pour rappel, la monotonicité de l'interprétation signifie que notre interprétation $[\![\cdot]\!]$ doit vérifier la propriété suivante :

$$\text{Pour tout } p \in Prop, \text{et pour tout } r, r' \in R, \text{ si } r \sqsubseteq r' \text{ et } r \in [\![p]\!], \text{ alors } r' \in [\![p]\!]$$

Avec la modélisation que nous avons choisie, cela signifie donc :

$$\text{Si } val(r) \leq val(r') \text{ et } r \in [\![p]\!], \text{ alors } r' \in [\![p]\!]$$

Dans l'exemple précédent, on avait $r \in [\![p]\!]$ lorsque $val(r)$ permettait d'acheter exactement l'objet $p$. Or, avec cette contrainte de monotonie, cette définition de l'interprétation ne convient plus. En effet, si $val(r) \leq val(r')$, alors si $r$ permet d'acheter $p$, $r'$ ne le permet pas (la somme peut être trop grande).

Cela dit, on peut résoudre facilement ce problème : il suffit de dire que $r \in [\![p]\!]$ lorsque $val(r)$ permet d'acheter *au moins* l'objet $p$. Alors, si $r$ permet au moins d'acheter $p$ et que $r'$ est

plus grande, alors elle permet aussi d'acheter au moins $p$. Par extension, on aura alors $r \vDash \phi$ si $r$ permet d'effectuer *au moins* $\phi$.

Voyons maintenant comment les différents opérateurs s'expriment dans cette nouvelle forme de modèle.

- $r \vDash_{\mathcal{K}} \mathrm{I}$ si $r \sqsubseteq e$. C'est à dire que I représente un objet "gratuit" puisqu'on peut l'obtenir avec une somme plus petite que $e$, représentant l'absence de pièce. On retrouve donc sensiblement la même sémantique que dans le cas de BBI.
- $r \vDash_{\mathcal{K}} \phi \rightarrow \psi$, pour toute ressource $r'$ telle que $r \sqsubseteq r'$, si $r' \vDash_{\mathcal{K}} \phi$, alors $r' \vDash_{\mathcal{K}} \psi$, c'est à dire que toute valeur plus grande que $r$, si elle permet d'effectuer l'opération $\phi$, permet aussi d'effectuer l'opération $\phi$. Ainsi une ressource qui permet d'effectuer $\phi \rightarrow \psi$ est un peu la somme minimum pour passer de $\phi$ à $\psi$.
- $r \vDash_{\mathcal{K}} \phi * \psi$ s'il existe $r_1, r_2$ tels que $r_1 \bullet r_2 \sqsubseteq r$, $r_1 \vDash_{\mathcal{K}} \phi$ et $r_2 \vDash_{\mathcal{K}} \psi$, c'est à dire que la somme représentée par $r$ contient au moins deux sommes, l'une permettant d'effectuer $\phi$, l'autre permettant d'effectuer $\psi$.Cette fois-ci, contrairement au cas de BBI, cette séparation de $r$ en deux sommes n'est pas stricte : une partie de la ressource peut ne pas être exploitée. C'est cette petite différence qui en même temps permet de décrire des cas plus généraux, mais apporte aussi un degré d'incertitude (puisque cette somme complémentaire ne peut être quantifiée).
- Les autres opérateurs modélisent les même opérations que dans le cas de BBI.

On peut alors modéliser des situations diverses. Considérons par exemple la formule que nous avions proposée dans l'exemple précédent.

$$c_{10} \bullet c_{10} \bullet c_{10} \vDash_{\mathcal{K}} Obj_{1.70} \mathrel{-\!\!*} (Obj_2 \wedge (Obj_{1.70} * Obj_{0.30}))$$

Cette affirmation reste correcte dans notre modèle avec BI. En effet, la somme représentée par $c_{10} \bullet c_{10} \bullet c_{10}$, c'est à dire 30 centimes, est suffisante, combinée avec le prix d'un objet à 1,70 euros pour obtenir soit le prix d'un objet à 2 euros, soit une somme qui séparée en deux autres sommes, l'une permettant d'acheter au moins un objet à 1,70 euros, l'autre un objet à 30 centimes.

Cela traduit le fait que les modèles de BBI sont un cas particuliers des modèles de BI (étant donné qu'une relation d'équivalence est un cas particulier de relation d'ordre). Toutefois, on peut aussi utiliser le fait que notre nouvelle modélisation d'interprétation est plus générale que dans le cas de BBI. Par exemple, on peut affirmer :

$$c_{50} \vDash_{\mathcal{K}} Obj_{1.70} \mathrel{-\!\!*} (Obj_2 \wedge (Obj_{1.70} * Obj_{0.30}))$$

En effet, une pièce de 50 centimes, additionnée à la valeur d'un objet de 1,70 euros représentent une somme de 2,20 euros, ce qui est suffisant pour acheter un objet à 2 euros et peut aussi être décomposé en deux sommes achetant l'une un objet à 1,70 euros, l'autre à 30 centimes. Par exemples, les sommes 1,80 et 0,40 conviennent.

On voit ainsi comment les modélisations avec BI sont plus générales (ici, on traduit la possibilité d'acheter un objet de manière générale, pas seulement de posséder la somme exacte) mais amènent aussi un grand degré d'incertitude (vu qu'on ne contrôle pas la différence entre les sommes possédées et les sommes requises).

## 2.6 Extensions modales et épistémiques de BI et BBI

Comme nous venons de le voir, BI et BBI sont des outils de modélisation puissants et disposent en outre de calculs performants et clairs.

BI et BBI peuvent ainsi exprimer des propriétés de séparation et de partage sur des zones mémoire [47,63,75,87], des zones de confiance [2,3], des arbres de ressources [13], des "layered graphs" [28], des processus manipulant les ressources [28,82,83].

L'expressivité de BI et de BBI a aussi abouti à de nombreux résultats, notamment concernant la modélisation de pointeurs [63,75,87] et la sécurité [2,3,28]. Pour la modélisation des pointeurs, la *logique de séparation* [87] qui utilise des opérateurs de BBI dans des assertions de Hoare [61] a produit d'excellents résultats dans le domaine de la vérification de programmes, par exemple l'outil Infer de Facebook [21].

Toutefois, pour étendre encore plus l'expressivité de ces logiques, des travaux ont cherché à les étendre par d'autres opérateurs modaux et/ou épistémiques.

### 2.6.1 Dynamisme et séparation : la logique DBI

Au vu de l'utilisation potentielle et effective de BI et BBI pour modéliser des structures de ressources [4,30], notamment des systèmes de consommation et des espaces mémoires (notablement via la logique de séparation [87]), l'ajout d'une dimension dynamique à ces logiques est une évidence.

En effet, l'étude statique est insuffisante à la majorité des travaux, et c'est souvent l'évolution des systèmes plutôt que leur structure qui importe. Par exemple, l'agencement d'un système de mémoire à un instant donné n'est utile à un procédé de vérification que si son évolution peut être suivie au cours de l'exécution d'un programme.

Cette dimension dynamique est d'ailleurs souvent implicitement déjà présente dans les modélisations statiques utilisant BI ou BBI. En effet, l'énonciation d'une formule contenant les opérateurs $\to$, $*$ ou $-\!\!*$ modélisent des implications ou des décompositions qui supposent des actions de transformation de l'espace des ressources ou des propositions qu'on considère. Par exemple $A \to B$ exprime que la validité de $A$ implique celle de $B$, d'où une pré-existence de $A$ par rapport à $B$, c'est-à-dire une forme d'ordonnancement temporel entre les propositions.

Mais bien entendu ces dynamiques implicites ne suffisent pas à décrire des notions plus complexes. Il faut expliciter la dimension temporelle de nos modèles. Une manière de le faire est d'utiliser les ressources comme marques d'états temporels. On pourra ainsi interpréter la composition comme une succession d'états, on aura $r_1 \bullet r_2$ si $r_2$ représente un état qui succède $r_1$. Évidemment, cela présuppose que la composition $\bullet$ ne soit pas commutative dans ce cas précis.

Cet exemple de l'utilisation des ressources comme marqueurs de temps peut être trouvé dans l'exemple des "layered graphs" [28]. Ces structures sont constituées de plusieurs graphes reliés par des arrêtes spéciales, chaque graphe étant appelé couche. Les différentes couches peuvent ainsi représenter plusieurs états successifs d'un même graphe au cours du temps. Cela dit, cette approche a un inconvénient majeur : puisque les ressources sont utilisées pour modéliser le temps, on ne peut plus utiliser pour modéliser des structures comme l'espace mémoire. On peut séparer les ressources en deux catégories (l'une exploitant le temps, l'autre l'espace), mais cela

est extrêmement complexe. Il est plus intéressant d'avoir une structure extérieure qui porte la dimension dynamique orthogonalement aux ressources.

Cette forme de dynamisme lui-aussi existe déjà dans d'autres travaux. Tout d'abord, la logique de séparation elle-même [87] possède une forme de temporalité induite, puisque les opérateurs de BI y sont inclus dans des structures évolutives à la manière de la logique de Hoare [61] avec des pré et post-conditions qui décrivent effectivement l'évolution de l'espace mémoire.

Mais là encore, on reste dans une forme implicite de temporalité. D'autres travaux ont exprimé plus clairement les relations entre BI et le dynamisme. On peut citer DBI (Dynamic BI) [34] qui ajoute à la syntaxe de BI les opérateurs de la logique modale S4 [70], DMBI (Dynamic Modular BI) [35] qui utilise des modalités paramétrées par des actions, à la manière de Hennessy-Milner [59] et LSM (Logic with Separating Modalities) [36], qui paramétrise les modalités par des ressources (voir le chapitre suivant pour plus de détails).

Nous nous intéresserons ici plus particulièrement à DBI. Cette extension utilise donc les opérateurs des logiques modales, en particulier dans sa variante S4 intuitionniste, tels que définis dans les travaux de Lewis et Langford [70]. Ces opérateurs sont traditionnellement notés $\Diamond$ et $\Box$. Étant donnée une structure de pré-ordre $R$ sur un ensemble d'états, $\Diamond\phi$ est valide à un état $s$ s'il existe un état $s'$ tel que $sRs'$ et $\phi$ est valide en $s'$, tandis que $\Box\phi$ est valide à un état $s$ si pour tout état $s'$ tel que $sRs'$, $\phi$ est valide en $s'$.

La logique DBI [34] combine donc les opérateurs de BI et de S4 de manière orthogonale. Par exemple, on aura $r, s \vDash \Diamond(A * B)$ s'il existe $r_1$, $r_2$ et $s'$ tels que $r_1 \bullet r_2 \sqsubseteq r$, $sRs'$ avec $r_1, s' \vDash A$ et $r_2, s' \vDash B$. On voit comment cette logique, comme son titre l'indique, permet d'introduire une dimension dynamique à BI. Si la relation $R$ est interprétée comme un ordre temporel, alors les opérateurs $\Diamond$ et $\Box$ permettent de décrire l'évolution d'un système dépeint par les ressources.

Comme dans toute logique modale, les modalités de DBI ne sont pas restreintes à cette sémantique temporelle. On peut les interpréter comme des marqueurs de localité, de possibilité, d'intention, de croyance, d'obligation ou de connaissance (un point qui a déjà été exploité dans d'autres domaines [37,45]). Toutefois, l'interprétation temporelle est la plus intuitive. D'ailleurs, on dira souvent que $s'$ est l'état suivant $s$ lorsque $sRs'$.

Notons enfin que DBI possède un calcul des tableaux étendant celui de BI. Dans ce calcul, une deuxième famille de labels, les *labels d'état*, marquent les états modaux et sont ordonnés par des *contraintes d'état*. Notons également que les preuves de correction et de complétude sont adaptées au cas du calcul pour DBI.

### 2.6.2 Connaissance et séparation : la logique ESL

Comme nous venons de le présenter, la logique DBI étend BI avec les modalités simples de la logique S4 dans son cadre intuitionniste [34]. Pour utiliser toute la puissance des logiques modales, toutefois, il est intéressant d'explorer des formalismes plus aboutis.

Une direction possible d'extension est l'ajout de modalités paramétrées par des actions, à la manière du calcul de Hennessy-Milner [59]. Cette autre extension a été proposée sous forme de la logique MDBI, présentée dans [35]. Cette autre logique propose une autre forme de dynamisme, plus adaptée à décrire l'évolution des ressources elles-mêmes.

D'un autre côté, une autre extension modale intéressante de BBI a été LSM [36] qui a vu l'introduction de modalités paramétrées par une ressource ($\Diamond_s$ et $\Box_s$, où $s$ est une ressource).

Une telle modalité décrit l'accès à un autre monde modal, mais à la condition que la ressource illustrée en indice soit présente dans le contexte initial (ou selon les cas dans le contexte final).

Mais l'une des directions des plus intéressantes concernant l'extension modale de BI est l'ajout de modalités épistémiques. Pour rappel, les logiques épistémiques sont une famille de logiques traitant de la connaissance [92]. Des opérateurs typiques de ces formalismes permettent de modéliser ce qu'un agent [2] connaît, mais aussi ce qu'il croit ou ce qu'il croit que les autres connaissent. Des raffinements impliquent aussi des systèmes d'annonces où les agents peuvent rendre publique une partie de leurs connaissances (notamment dans la logique d'annonces publiques PAL [8,76]).

De nombreux travaux ont été proposés pour rapprocher les logiques épistémiques des notions de ressources [9,57,73]. En effet, la notion de ressource est très pertinente pour modéliser des connaissances. On voit en particulier comment les logiques de séparation comme BI sont appropriées. Cela est dû en partie à l'utilisation de la composition de ressources $\bullet$ comme méthode d'agrégation des connaissances. Deux informations étant a priori distinctes, un opérateur multiplicatif est beaucoup plus pertinent qu'un additif.

Étant donné ce lien thématique fort entre ressources et connaissances, il paraissait intéressant de considérer des extensions de BI avec des modalités épistémiques. Cela a été réalisé par la création de la logique de séparation épistémique ESL [37]. Cette logique étend la logique classique BBI par l'opérateur épistémique $K_a$, permettant donc d'exprimer la connaissance d'agents en plus des propriétés de séparation propres à BBI. Plus précisément, $K_a\phi$ indique que l'agent $a$ connaît la propriété $\phi$. Dans cet opérateur, les ressources remplacent les mondes modaux des logiques épistémiques, et l'opérateur caractérise donc la connaissance d'une propriété d'une ressource.

Notons enfin que ESL possède un calcul des tableaux étendant celui de BBI. Dans ce calcul, l'accès épistémique est caractérisé par des *contraintes d'agents* étiquetées par les agents du système. Notons également que les preuves de correction et de complétude sont adaptées au cas du calcul pour ESL.

# Conclusion

Nous avons présenté dans ce chapitre les logiques BI et BBI introduites à l'origine dans [74,80]. Ces logiques manipulent des ressources, c'est à dire des entités pouvant être composées et décomposées, ce qui les rend idéales pour modéliser divers systèmes (espaces mémoires, zones de confiance, arbres de ressources, layered graphs, etc...).

Nous avons introduit une sémantique à base de monoïdes complets pour BI et une sémantique à base de monoïdes partiels pour BBI qui seront à la base des logiques présentées dans cette thèse. Nous avons présenté un calcul des tableaux pour BBI [66] que nous exploiterons par des extensions dans les chapitres suivants. Nous avons illustré par un exemple comment BI et BBI peuvent notamment être utilisées pour modéliser un système de consommation de ressources. Enfin, nous avons présenté des extensions dynamiques et épistémiques de BI et BBI, notamment la logique DBI [34] et la logique ESL [37].

2. On utilisera le terme *agent* pour désigner une entité qui a accès à une connaissance. Dans le cadre d'une modélisation concrète, souvent ces agents représenteront des personnes, mais il peut aussi s'agir d'objets plus abstraits comme des systèmes informatiques.

Dans le Chapitre 3, nous explorons une nouvelle logique, LTBI, qui allie l'expressivité de BI et celle de la logique temporelle linéaire LTL [65,77] afin de proposer une logique de ressources dynamiques dans la lignée de DBI, mais avec des spécificités de modélisation et de calcul propres à la logique temporelle linéaire. Après en avoir exposé la syntaxe et la sémantique, nous étudions son expressivité autour d'exemples. Nous proposons également un calcul des tableaux inspiré par celui de DBI mais avec des labels doubles et des contraintes adaptés à la spécificité temporelle linéaire de LTBI. Nous en montrons la correction, la complétude tout en proposant une méthode d'extraction de contre-modèles. Nous proposons également un deuxième calcul des tableaux, inspiré de celui de LTL, qui ne possède pas de labels temporels mais utilise des tableaux potentiellement cycliques et des arrêtes spécifiques pour traduire la dimension temporelle. Nous en montrons la correction et discutons de sa complétude ainsi que des liens entre les deux calculs.

Dans le Chapitre 4, nous envisageons l'ajout de ressources à l'opérateur épistémique de ESL, de la même manière que LSM le faisait avec les opérateurs de S4. On obtient alors une logique de ressources épistémique ERL qui permet de caractériser l'accès par un agent à une propriété conditionné par une ressource. L'expressivité de cette logique en fait un excellent outil de modélisation de problèmes de contrôle d'accès. Nous illustrons ce fait par plusieurs exemples de modélisation. Nous proposons un calcul des tableaux avec labels pour ERL et nous en étudions la correction, la complétude, en proposant également une méthode d'extraction de contre-modèles.

Dans le Chapitre 5, nous proposons une manière d'étendre l'expressivité de BBI en représentant les ressources dans la syntaxe de la logique, par un procédé appelé hybridation. Nous présentons une logique hybride de ressources HRL où la composition de ressource est sans propriétés, ainsi qu'une restriction de HRL par des axiomes nommée HBBI qui est sémantiquement équivalente à BBI. Nous proposons également un calcul des tableaux sans labels pour HRL et HBBI inspiré du calcul de BBI. Nous étudions également dans le Chapitre 6 le lien entre ce calcul et le calcul de BBI pour le cas des formules de BBI.

# Chapitre 3

# Une extension temporelle pour BI

La direction la plus naturelle pour étendre l'expressivité des logiques de la famille de BI est l'ajout de dynamisme. En effet, comme nous l'avons exposé dans le chapitre précédent, la richesse de BI et de BBI quant à la manipulation de ressources, notamment l'expression de leur partage et de leur séparation, en font des outils particulièrement pertinents pour modéliser des systèmes divers [4,30] et notamment des espaces mémoires, preuve en est le succès de la logique de séparation [87]. Or, si la richesse de ces logiques se prêtent bien à une analyse statique de tels systèmes, il devient presque essentiel de pouvoir capturer aussi des aspects dynamiques.

Des propositions ont déjà été faites quant à l'extension de BI ou de BBI dans ce sens, notamment la logique DBI (Dynamic BI) [34], qui prolonge BI et que nous avons exposée dans le Chapitre 2 (Section 2.6.1). Toutefois, cette logique utilise les opérateurs modaux de la logique S4 qui, s'ils peuvent être utilisés pour décrire le temps, ne sont spécifiquement conçus pour modéliser une évolution temporelle.

Dans ce chapitre, nous exposons notre propre extension temporelle de BI, appelée LTBI (pour Linear Temporal BI), dans le même esprit que DBI, mais qui utilise les opérateurs $\Diamond$, $\Box$ et $\circ$ de la logique temporelle linéaire LTL [77] pour ajouter une dimension dynamique à la logique BI. Ceci permet de modéliser plus finement les aspects temporels de systèmes. En outre, la logique LTL sur laquelle nous nous basons possède de nombreux résultats calculatoires et est utilisée largement dans les problèmes de vérification ("model checking") [65] ce qui laisse espérer des perspectives intéressantes pour LTBI en terme de modélisation et de calcul.

Dans un premier temps, nous exposons une analyse de la temporalité dans le cadre des logiques de séparation et les choix fixés après celle-ci.

Nous présentons ensuite la syntaxe et la sémantique de la nouvelle logique LTBI avant de donner plusieurs exemples de son expressivité.

Puis, nous définissons un calcul des tableaux pour LTBI qui, comme le caclul de DBI [34], utilise des labels de ressources et des labels d'états pour traduire la sémantique de LTBI. Toutefois, notre calcul comporte des règles supplémentaires pour garantir la linéarité des états temporels. Dans les tableaux non finis, il est également nécessaire de garantir un isomorphisme d'ordre entre les labels d'états et $\mathbb{N}$, afin que chaque état puisse avoir un état suivant. Ce point fait l'objet d'une conjecture qui est actuellement non résolue et dont nous expliquons les tenants et les aboutissants.

Nous fournissons ensuite des exemples de tableaux, une preuve de correction et de complétude ainsi qu'une méthode d'extraction de contre-modèles, méthode dans laquelle la conjecture

sur l'ordre des labels d'états jouera un rôle essentiel (pour garantir des états suivants).

Enfin, nous proposons un deuxième calcul pour LTBI, inspiré cette fois du calcul des tableaux de LTL, et qui n'utilise pas de labels d'états mais s'appuie sur des tableaux potentiellement cycliques où la structure temporelle est portée par certaines arrêtes. Nous l'avons nommé calcul des c-tableaux (pour "cyclic tableaux", ou tableaux cycliques). Nous exposons pour ce deuxième calcul un exemple et une preuve de correction. La complétude fait lieu d'une discussion ainsi que les perspectives de ce calcul, notamment en terme de modélisation par des automates.

## 3.1 Temporalité dans les logiques de séparation

Nous avons exposé dans le Chapitre 2 comment la logique DBI [34], extension de BI par des modalités de la logique S4, permettait d'intégrer du dynamisme dans les logiques de séparation, en utilisant la relation $R$ entre états modaux comme une modélisation de l'écoulement du temps.

Si cette interprétation des modalités de DBI est naturelle, elle n'est pas en revanche la plus adaptée. Comme nous l'avons dit, la relation $R$ est un pré-ordre. En représentant les états selon un graphe, cela veut dire qu'on peut avoir un cycle dans un tel graphe. Si l'interprétation circulaire du temps a été étudiée philosophiquement, elle n'est pas ordinairement utilisée en modélisation. On préférera des états temporels ordonnés ou totalement ordonnés (voir plus loin). Ainsi, certains modèles de DBI conviennent à la modélisation de systèmes dynamiques (ceux qui sont ordonnés ou totalement ordonnés) mais pas tous.

L'intention dans le développement de LTBI est donc de reprendre le travail sur DBI mais de modifier la sémantique pour rendre les modalités spécifiques à une interprétation temporelle. Comme nous l'avons déjà dit, il reste un choix possible. Les instants temporels peuvent être ordonnés ou totalement ordonnés. Cela correspond a deux visions bien connues de la notion de temps : le *temps linéaire* et le *temps branchant* (voir [11] pour plus de détails sur les fondations philosophiques et logiques de la temporalité).


FIGURE 3.1 – Temps linéaire

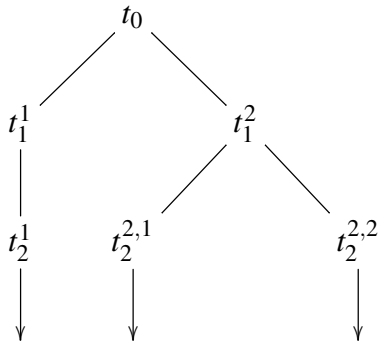



FIGURE 3.2 – Temps branchant

Dans le temps linéaire, les instants temporels se suivent les uns derrière les autres, ils sont ordonnés totalement (Figure 3.1). Ainsi, il n'y a qu'un futur possible et qu'un passé possible. Ce formalisme, le plus simple, a été utilisé dès 1977 sous la forme de la logique temporelle linéaire LTL [77]. Plus que l'ordre total, LTL fonde ses instants temporels sur les entiers naturels, ce

qui permet d'introduire en plus des opérateurs modaux initiaux $\Diamond$ et $\Box$ (notés parfois **F** et **G**) un opérateur $\circ$ (noté parfois **X**) énonçant qu'une formule est valide à l'instant suivant.

Dans le temps branchant, chaque instant dispose potentiellement de plusieurs instants futurs, créant pour ainsi dire plusieurs futurs alternatifs (voir Figure 3.2). Les instants sont alors simplement ordonnés. Cette approche a été étudiée dans une extension de LTL donnée par Clarke et Emerson, dénommée CTL (Computation Tree Logic) [25]. Aux opérateurs **F**, **G**, et **X** de LTL, CTL ajoute deux opérateurs **A** et **E** qui permettent respectivement de quantifier universellement et existentiellement sur les futurs possibles. Ainsi **EF**$\phi$ signifie qu'il existe au moins un avenir et au moins un instant dans cet avenir tel que $\phi$ soit réalisée.

Entre ces deux formalismes temporels, nous avons préféré fonder LTBI sur LTL. Ce choix est d'abord pratique, puisque la sémantique de LTL est plus simple que celle de CTL, mais aussi vise à la meilleure utilisation de LTBI. En effet, LTL est plus ancienne, plus étudiée, et la base de la plupart des méthodes de model checking actuelles (même si CTL et ses variantes y sont aussi bien représentées). On pourra se référer à [93] pour en savoir plus sur l'évolution des logiques temporelles et sur LTL en particulier, notamment sur sa complexité et sa décidabilité.

En outre, comme nous l'évoquons dans la section 3.3.3, on peut en intégrant LTL à BI recomposer une logique temporelle branchante analogue à CTL, et même potentiellement plus expressive.

Nous présentons donc LTBI, une extension temporelle de BI, analogue à l'extension modale DBI, mais où les opérateurs modaux $\Diamond$ et $\Box$ sont restreints à un contexte temporel linéaire inspiré de LTL. Cela nous permet également l'ajout d'un opérateur $\circ$.

Notons que, tout comme DBI dont elle s'inspire, la logique LTBI s'appuie sur la logique intuitionniste BI (et non sur sa contre-partie classique BBI). Comme nous l'avons explicité dans la Section 2.3.3, le fait de s'appuyer sur la logique BI permet de bénéficier de la décidabilité de cette dernière (décidabilité qui s'étend sans doute à LTBI, bien que nous ayons choisi de ne pas aborder cette problématique ici). Bien entendu, il est envisageable de développer une version classique de LTBI qui s'appuierait cette fois sur BBI, ce qui permettrait notamment de bénéficier d'une sémantique plus compréhensible pour la modélisation (voir Section 2.3.3).

Notons enfin que la logique *tBI* (temporal BI) proposée dans [64] présente une extension similaire de BI par les opérateurs temporels de LTL (notés **F**, **G** et **X**). Si les sémantiques de tBI et LTBI sont équivalentes, l'approche que nous présentons ici est différente puisque cet article présente une extension du calcul des séquents avec "bunchs" de Gentzen pour BI [81] et une traduction de tBI dans une extension temporelle de la logique linéaire [53]. Notre approche, basée sur une sémantique à la Kripke, est plus explicite et essentiellement axée sur les capacités d'expressivité de cette logique, notamment pour la modélisation, et sur la construction d'un calcul des tableaux avec labels dans l'esprit de celle de DBI [34].

Ce calcul des tableaux, présenté dans la Section 3.4, utilise des labels d'états et des contraintes temporelles pour identifier les instants temporels. Des règles spécifiques permettent d'assurer la linéarité de ces états. En outre, il est nécessaire d'assurer dans les branches non finies que chaque label possède un label suivant. Ce problème fait l'objet d'une conjecture que nous laissons inachevée et dont nous exposerons plus loin les tenants et aboutissants.

## 3.2 La logique LTBI

### 3.2.1 Syntaxe de LTBI

La nouvelle logique que nous voulons construire se veut être une extension conservative de BI, de même que BI étend IL [74]. Ainsi, la restriction de notre logique aux opérateurs additifs et multiplicatifs doit coïncider avec BI. Nous allons donc utiliser les opérateurs de BI pour définir notre nouvelle syntaxe.

Dans le même esprit, la partie temporelle de notre nouvelle logique s'inspire de la logique temporelle linéaire, LTL [77]. Toutefois, LTBI n'est pas exactement une extension conservative de LTL puisque ses opérateurs sont intuitionnistes. On peut cependant retrouver une extension conservative de LTL soit en construisant une variante classique de LTBI (basée sur BBI), soit en considérant une version intuitionniste de LTL [71].

Les opérateurs de LTL sont nombreux. On utilise traditionnellement **F** (un jour), **G** (toujours), **X** (à l'état suivant), **U** (jusqu'à ce que) et **V** (à partir de). Dans notre approche, par esprit de simplicité et par similitude avec les logiques modales étudiées dans DBI [34], nous nous sommes concentrés sur les opérateurs **F**, **G** et **X**. Toujours par similitude avec les logiques modales, nous avons décidé de noter $\Diamond$, $\Box$ et $\circ$ les opérateurs **F**, **G** et **X** (ce qui est une convention usuelle).

On obtient donc une nouvelle logique que nous dénommons LTBI, pour Linear Temporal Bunched Implication logic, et dont la syntaxe précise est la suivante :

**Définition 3.1** (Syntaxe de LTBI)**.** *Soit Prop un ensemble de symbole propositionnels. Le langage $\mathcal{L}$ de la logique LTBI est donnée par la grammaire suivante pour tout $p \in Prop$ :*

$$X ::= p \mid \top \mid \bot \mid X \wedge X \mid X \vee X \mid X \to X \mid \mathrm{I} \mid X * X \mid X \mathrel{-\!\!*} X \mid \Diamond X \mid \Box X \mid \circ X$$

Dans l'intégralité de ce chapitre, $\mathcal{L}$ désignera le langage de LTBI sauf indication contraire. On introduit en outre la négation par l'égalité suivante : $\neg X ::= X \to \bot$.

Intuitivement, nos nouveaux opérateurs ont la signification suivante :

- $\Diamond\phi$ est valide si au moins un instant futur (possiblement l'instant présent) valide $\phi$
- $\Box\phi$ est valide si tous les instants futurs (y compris l'instant présent) valident $\phi$
- $\circ\phi$ est valide si l'instant suivant valide $\phi$

Nous allons maintenant formaliser ces intuitions par la définition de notre sémantique.

### 3.2.2 Sémantique de LTBI

LTBI étant basée sur deux prémisses, le partage des ressources et la dimension temporelle, nous devons concevoir des structures formelles qui décrivent ces deux dimensions. Nous fondons la partie concernant les ressources sur les monoïdes complets que nous avons exposés dans la section 2.2.

Quant à la partie temporelle, il est superflu de développer une structure pour la décrire. En effet, un sous-ensemble de l'ensemble $\mathbb{N}$ des entiers naturels capture parfaitement les propriétés que nous souhaitons pour nos instants temporels : il est totalement ordonné, dénombrable et non dense pour l'ordre.

**Définition 3.2** (Monoïde linéaire de ressources)**.** *Un* monoïde linéaire de ressources *est une structure* $\mathcal{R} = (R, e, \bullet, \sqsubseteq, S)$*, où* $(R, e, \bullet, \sqsubseteq)$ *est un monoïde de ressources (tel que défini dans la Définition 2.10) et* $S \subseteq \mathbb{N}$ *est un ensemble d'entiers naturels (possiblement infini) appelés* états *ou* instants*.*

*Pour tout* $i \in S$*, l'instant* $i+1$ *est appelé* instant suivant *i.*

*Le plus petit élément de S, noté* $s_0$*, est appelé* instant initial*.*

L'existence de $s_0$ est assurée par le fait que $S$ est une partie de $\mathbb{N}$.

**Définition 3.3** (Interprétation linéaire)**.** *Soit* $\mathcal{R} = (R, e, \pi, \bullet, \sqsubseteq, S)$ *un monoïde de ressources linéaire. Une* interprétation *de Prop pour* $\mathcal{R}$ *est une fonction* $[\![\cdot]\!] : Prop \to \mathbb{P}(R \times S)$ *qui est monotone et inconsistante, c'est à dire qu'elle vérifie les propriétés suivantes :*

$$\forall p \in Prop, \forall i \in S, \forall r, r' \in R, \text{ si } r \sqsubseteq r' \text{ et } (r,i) \in [\![p]\!] \text{ alors } (r',i) \in [\![p]\!] \text{ (monotonie)}$$

$$\forall p \in Prop, \forall i \in S, \forall r \in R, \text{ si } \pi \sqsubseteq r \text{ alors } (r,i) \in [\![p]\!] \text{ (inconsistance)}$$

Nous pouvons maintenant définir une structure de Kripke pour interpréter notre syntaxe.

**Définition 3.4** (Modèle linéaire de ressources)**.** *Un* modèle linéaire de ressources *pour LTBI est un triplet* $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *où* $\mathcal{R} = (R, e, \bullet, \sqsubseteq, S)$ *est un monoïde linéaire de ressources,* $[\![\cdot]\!]$ *est une interprétation linéaire de Prop pour* $\mathcal{R}$ *et* $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times R \times S$ *une relation de forcing, définie récursivement par les relations suivantes pour tout* $p \in Prop, i \in S, \phi, \psi \in \mathcal{L}$ *:*

- $(r,i) \vDash_{\mathcal{K}} p$ *ssi* $(r,i) \in [\![p]\!]$
- $(r,i) \vDash_{\mathcal{K}} \mathrm{I}$ *ssi* $e \sqsubseteq r$
- $(r,i) \vDash_{\mathcal{K}} \top$ *toujours*
- $(r,i) \vDash_{\mathcal{K}} \bot$ *ssi* $\pi \sqsubseteq r$
- $(r,i) \vDash_{\mathcal{K}} \phi \wedge \psi$ *ssi* $(r,i) \vDash_{\mathcal{K}} \phi$ *et* $(r,i) \vDash_{\mathcal{K}} \psi$
- $(r,i) \vDash_{\mathcal{K}} \phi \vee \psi$ *ssi* $(r,i) \vDash_{\mathcal{K}} \phi$ *ou* $(r,i) \vDash_{\mathcal{K}} \psi$
- $(r,i) \vDash_{\mathcal{K}} \phi \to \psi$ *ssi pour tout* $r' \in R$ *tel que* $r \sqsubseteq r'$ *et* $(r',i) \vDash_{\mathcal{K}} \phi$*, on a* $(r',i) \vDash_{\mathcal{K}} \psi$
- $(r,i) \vDash_{\mathcal{K}} \phi * \psi$ *ssi il existe* $r', r'' \in R$ *tels que* $r' \bullet r'' \sqsubseteq r$ *et* $(r',i) \vDash_{\mathcal{K}} \phi$ *et* $(r'',i) \vDash_{\mathcal{K}} \psi$
- $(r,i) \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ *ssi pour tout* $r' \in R$ *tel que* $(r',i) \vDash_{\mathcal{K}} \phi$*, on a* $(r \bullet r', i) \vDash_{\mathcal{K}} \psi$
- $(r,i) \vDash_{\mathcal{K}} \Diamond\phi$ *ssi il existe* $j \in S$ *tel que* $j \geq i$ *et* $(r,j) \vDash_{\mathcal{K}} \phi$
- $(r,i) \vDash_{\mathcal{K}} \Box\phi$ *ssi pour tout* $j \in S$ *tel que* $j \geq i$*, on a* $(r,j) \vDash_{\mathcal{K}} \phi$
- $(r,i) \vDash_{\mathcal{K}} \bigcirc\phi$ *ssi* $i+1 \in S$ *et* $(r,i+1) \vDash_{\mathcal{K}} \phi$

Notons que les définitions sémantiques des opérateurs temporels $\Diamond$, $\Box$ et $\bigcirc$ correspondent aux définitions intuitives que nous avions données à la fin de la section précédente.

**Définition 3.5** (Validité)**.** *On dit qu'une formule* $\phi$ *de* $\mathcal{L}$ *est* valide *dans LTBI et on note* $\vDash_{LTBI} \phi$ *lorsque* $(e, s_0) \vDash_{\mathcal{K}} \phi$ *dans tout modèle* $\mathcal{K}$*.*

*On dit en outre que* $\psi$ *est une* conséquence logique *de* $\phi$ *dans LTBI et on note* $\phi \vDash_{LTBI} \psi$ *lorsque* $(r, s_0) \vDash_{\mathcal{K}} \phi$ *implique* $(r, s_0) \vDash_{\mathcal{K}} \psi$ *pour toute ressource r de tout modèle* $\mathcal{K}$*.*

*On dit que deux formules* $\psi$ *et* $\phi$ *sont* équivalentes *dans LTBI et on note* $\phi \equiv_{LTBI} \psi$ *lorsqu'on a* $\phi \vDash_{LTBI} \psi$ *et* $\psi \vDash_{LTBI} \phi$*.*

Lorsque le contexte n'est pas ambigü, on omettra les indices et on notera $\vDash \phi$, $\phi \vDash \psi$ ou $\phi \equiv \psi$.

Le théorème suivant permettra de démontrer aisément la conséquence logique.

**Théorème 3.1** (Conséquence logique)**.** *Soit deux formules $\phi$ et $\psi$. On a $\phi \vDash \psi$ ssi $\vDash \phi \mathrel{-\!\!*} \psi$.*

*Démonstration.*

- On suppose $\phi \vDash \psi$.
  Soit $\mathcal{K}$ un modèle. Soit $r \in R$ tel que $(r, s_0) \vDash_{\mathcal{K}} \phi$. Comme $\phi \vDash \psi$, on a donc également $(r, s_0) \vDash_{\mathcal{K}} \psi$. Or, comme $e$ est un élément neutre pour la composition, on a $r \bullet e = r$, donc $(r \bullet e, s_0) \vDash_{\mathcal{K}} \psi$. On a montré que $(r \bullet e, s_0) \vDash_{\mathcal{K}} \psi$ pour tout $r$ tel que $(r, s_0) \vDash_{\mathcal{K}} \phi$, donc $(e, s_0) \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ et ce pour tout modèle $\mathcal{K}$. Finalement, $\vDash \phi \mathrel{-\!\!*} \psi$
- On suppose $\vDash \phi \mathrel{-\!\!*} \psi$. Soit $\mathcal{K}$ un modèle. Soit $r \in R$ tel que $(r, s_0) \vDash_{\mathcal{K}} \phi$. Puisque $\vDash \phi \mathrel{-\!\!*} \psi$, on a $(e, s_0) \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$. Par définition, et comme $(r, s_0) \vDash_{\mathcal{K}} \phi$, on obtient que $(e \bullet r, s_0) \vDash_{\mathcal{K}} \psi$. Or $r \bullet e = r$. Donc $(r, s_0) \vDash_{\mathcal{K}} \psi$. Comme cela est valable pour toute ressource $r$ de tout modèle $\mathcal{K}$, on a donc $\phi \vDash \psi$.

□

Nous retrouvons les lemmes de monotonie et d'inconsistance pour notre nouvelle logique, et nous en donnons ici une brève démonstration.

**Lemme 3.1.** *Pour tout modèle $\mathcal{K}$, pour toute ressource $r, r'$ de ce modèle, pour tout instant $i \in S$ et pour toute formule $\phi$, si $(r, i) \vDash_{\mathcal{K}} \phi$ et $r \sqsubseteq r'$, alors $(r', i) \vDash_{\mathcal{K}} \phi$.*

*Démonstration.*
Soit $\mathcal{K}$ un modèle. On démontre la propriété par récurrence sur la taille de $\phi$.

- Cas de base
  On considère deux ressources $r, r' \in R$ telles que $r \sqsubseteq r'$ et un instant $i \in S$. On montre que le lemme est vrai pour toutes les formules de taille 1.
  - Cas $\phi = p \in Prop$
    On suppose que $(r, i) \vDash_{\mathcal{K}} \phi$. On a donc $(r, i) \vDash_{\mathcal{K}} p$. Par définition, $(r, i) \in [\![p]\!]$. Or, comme $[\![\cdot]\!]$ est une interprétation linéaire et que $r \sqsubseteq r'$, on a par monotonie que $(r', i) \in [\![p]\!]$ (Définition 3.3) et on a donc $(r', i) \vDash_{\mathcal{K}} p$. Le lemme est donc vrai dans ce cas.
  - Cas $\phi = \mathrm{I}$
    On suppose que $(r, i) \vDash_{\mathcal{K}} \phi$. On a donc $e \sqsubseteq r$. Comme $r \sqsubseteq r'$ et que $\sqsubseteq$ est transitive (Définition 2.10), on a $(e, i) \sqsubseteq r'$, ainsi $(r', i) \vDash_{\mathcal{K}} \mathrm{I}$. Le lemme est donc vrai dans ce cas.
  - Cas $\phi = \top$
    $(r, i) \vDash_{\mathcal{K}} \top$ est toujours valide, et $(r', i) \vDash_{\mathcal{K}} \top$ est également toujours valide. Par conséquent le lemme est vrai dans ce cas.
  - Cas $\phi = \bot$
    On suppose que $(r, i) \vDash_{\mathcal{K}} \phi$. On a donc $\pi \sqsubseteq r$. Par transitivité, comme $r \sqsubseteq r'$, on a $\pi \sqsubseteq r'$ et donc $(r', i) \vDash_{\mathcal{K}} \bot$. Le lemme est donc vrai dans ce cas.

- Hérédité
  On suppose que le lemme est vrai pour tout couple de ressources $(r, r')$, pour tout instant $i \in S$ pour toutes les formules de taille inférieure à $n-1$ (hypothèse de récurrence (HR)). On montre que le lemme est vrai pour les formules de taille $n$. On considère deux ressources $r, r' \in R$ telles que $r \sqsubseteq r'$ et un instant $i \in S$.

  - Cas $\phi = \psi \wedge \psi'$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \psi \wedge \psi'$, donc $(r,i) \vDash_{\mathcal{K}} \psi$ et $(r,i) \vDash_{\mathcal{K}} \psi'$. Comme $\psi$ et $\psi'$ sont de taille $n-1$, on a par (HR) que $(r',i) \vDash_{\mathcal{K}} \psi$ et $(r',i) \vDash_{\mathcal{K}} \psi'$, donc $(r',i) \vDash_{\mathcal{K}} \psi \wedge (r',i) \vDash_{\mathcal{K}} \psi'$, c'est à dire $(r',i) \vDash_{\mathcal{K}} \phi$. Ainsi le lemme est vérifié dans ce cas.
  - Cas $\phi = \psi \vee \psi'$
    Cas similaire au précédent.
  - Cas $\phi = \psi \rightarrow \psi'$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \psi \rightarrow \psi'$. Soit $r'' \in R$ tel que $r' \sqsubseteq r''$ et $(r'',i) \vDash_{\mathcal{K}} \psi$. Alors par transitivité on a aussi $r \sqsubseteq r''$. Donc par la Définition 3.4, on a $(r'',i) \vDash_{\mathcal{K}} \psi'$, et ce pour tout $r'' \in R$ tel que $r' \sqsubseteq r''$ et $(r'',i) \vDash_{\mathcal{K}} \psi$. On a donc $(r',i) \vDash_{\mathcal{K}} \psi \rightarrow \psi'$ et le lemme est vérifié dans ce cas.
  - Cas $\phi = \psi * \psi'$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \psi * \psi'$. Il existe donc $r_1, r_2 \in R$ tels que $r_1 \bullet r_2 \sqsubseteq r$ et $(r_1,i) \vDash_{\mathcal{K}} \psi$ et $(r_2,i) \vDash_{\mathcal{K}} \psi'$. Par transitivité, on a aussi $r_1 \bullet r_2 \sqsubseteq r'$ et par conséquent on a $(r',i) \vDash_{\mathcal{K}} \psi * \psi'$. Le lemme est donc vérifié dans ce cas.
  - Cas $\phi = \psi \mathrel{-\!\!*} \psi'$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \psi \mathrel{-\!\!*} \psi'$. Soit $r'' \in R$ tel que $(r'',i) \vDash_{\mathcal{K}} \psi$. Par la Définition 3.4 on obtient donc $(r \bullet r'',i) \vDash_{\mathcal{K}} \psi'$. Comme $\sqsubseteq$ est compatible avec $\bullet$ (Définition 2.10), $r \sqsubseteq r'$ entraîne $r \bullet r'' \sqsubseteq r' \bullet r''$. Alors, comme $\psi'$ est de taille $n-1$, on a par (HI) que $(r' \bullet r'',i) \vDash_{\mathcal{K}} \psi'$. Donc par la Définition 3.4 on obtient $(r',i) \vDash_{\mathcal{K}} \psi \mathrel{-\!\!*} \psi'$. Le lemme est donc vérifié dans ce cas.
  - Cas $\phi = \Diamond \psi$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \Diamond \psi$. Il existe donc $j \in S$ tel que $j \geq i$ et $(r,j) \vDash_{\mathcal{K}} \psi$. Comme $\psi$ est de taille $n-1$, on a par (HR) que $(r',j) \vDash_{\mathcal{K}} \psi$. On a donc trouvé $j$ tel que $j \geq i$ et $(r',j) \vDash_{\mathcal{K}} \psi$, donc $(r',i) \vDash_{\mathcal{K}} \Diamond \psi$. Le lemme est donc vérifié dans ce cas.
  - Cas $\phi = \Box \psi$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \Box \psi$. Soit $j \in S$ tel que $j \geq i$ et $(r,j) \vDash_{\mathcal{K}} \psi$. Comme $\psi$ est de taille $n-1$, on a par (HR) que $(r',j) \vDash_{\mathcal{K}} \psi$. On a donc pour tout $j$ tel que $j \geq i$ que $(r',j) \vDash_{\mathcal{K}} \psi$, donc $(r',i) \vDash_{\mathcal{K}} \Box \psi$. Le lemme est donc vérifié dans ce cas.
  - Cas $\phi = \bigcirc \psi$
    On suppose que $(r,i) \vDash_{\mathcal{K}} \phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \bigcirc \psi$, donc $(r,i+1) \vDash_{\mathcal{K}} \psi$. Comme $\psi$ est de taille $n-1$, on a par (HR) que $(r',i+1) \vDash_{\mathcal{K}} \psi$. On a donc $(r',i) \vDash_{\mathcal{K}} \bigcirc \psi$. Le lemme est donc vérifié dans ce cas.

  $\square$

**Lemme 3.2.** *Pour tout modèle $\mathcal{K}$, pour tout instant $i \in S$ et pour toute formule $\phi$ on a $(\pi, i) \vDash_{\mathcal{K}} \phi$.*

*Démonstration.*
Soit $\mathcal{K}$ un modèle. On démontre la propriété par récurrence sur la taille de $\phi$.

- Cas de base
  On considère un instant $i \in S$. On montre que le lemme est vrai pour toutes les formules de taille 1.
  - Cas $\phi = p \in Prop$
    La relation $\sqsubseteq$ est réflexive, on a donc en particulier $\pi \sqsubseteq \pi$. Or, par la Définition 3.3, $[\![\cdot]\!]$ est inconsistante, et on a donc en particulier $(\pi, i) \in [\![p]\!]$. Ainsi, $(\pi, i) \vDash_{\mathcal{K}} p$ et le lemme est vrai dans ce cas.
  - Cas $\phi = \mathrm{I}$
    $\pi$ est le plus grand élément de $R$, donc pour tout $r \in R$ on a $r \sqsubseteq \pi$ et en particulier $e \sqsubseteq \pi$. Ainsi, $(\pi, i) \vDash_{\mathcal{K}} \mathrm{I}$ et le lemme est vrai dans ce cas.
  - Cas $\phi = \top$
    $(r, i) \vDash_{\mathcal{K}} \top$ est toujours valide, par conséquent $(\pi, i) \vDash_{\mathcal{K}} \top$ et le lemme est vrai dans ce cas.
  - Cas $\phi = \bot$
    La relation $\sqsubseteq$ est réflexive, on a donc en particulier $\pi \sqsubseteq \pi$, donc $(\pi, i) \vDash_{\mathcal{K}} \bot$. Le lemme est donc vrai dans ce cas.
- Hérédité
  On suppose que le lemme est vrai pour tout instant $i \in S$ pour toutes les formules de taille inférieure à $n-1$ (hypothèse de récurrence (HR)). On montre que le lemme est vrai pour les formules de taille $n$. On considère un instant $i \in S$.
  - Cas $\phi = \psi \wedge \psi'$
    $\psi$ et $\psi'$ sont de taille inférieure à $n$, donc par (HR) on a $(\pi, i) \vDash_{\mathcal{K}} \psi$ et $(\pi, i) \vDash_{\mathcal{K}} \psi'$, donc $(\pi, i) \vDash_{\mathcal{K}} \psi \wedge \psi'$. Ainsi le lemme est vérifié dans ce cas.
  - Cas $\phi = \psi \vee \psi'$
    Cas similaire au précédent.
  - Cas $\phi = \psi \to \psi'$
    Soit $r \in R$ tel que $\pi \sqsubseteq r$ et $(r, i) \vDash_{\mathcal{K}} \psi$. Comme $\pi$ est le plus grand élément de $r$, on a nécessairement $\pi = r$. Or, par (HR), $(\pi, i) \vDash_{\mathcal{K}} \psi'$, donc en particulier $(r, i) \vDash_{\mathcal{K}} \psi'$. On a donc $(\pi, i) \vDash_{\mathcal{K}} \psi \to \psi'$
  - Cas $\phi = \psi * \psi'$
    Par (HR) on a $(\pi, i) \vDash_{\mathcal{K}} \psi$ et $(\pi, i) \vDash_{\mathcal{K}} \psi'$. Or, comme $\pi$ est un élément absorbant pour $\bullet$ (Définition 2.10), on a $\pi \bullet \pi = pi$, et comme $\sqsubseteq$ est réflexive, $\pi \sqsubseteq \pi$, ainsi $\pi \bullet \pi \sqsubseteq \pi$. On a donc que $(\pi, i) \vDash_{\mathcal{K}} \psi * \psi'$ et le lemme est vérifié dans ce cas.
  - Cas $\phi = \psi \mathrel{-\!\!*} \psi'$
    Soit $r \in R$ tel que $(r, i) \vDash_{\mathcal{K}} \psi$. Comme $\pi$ est un élément absorbant pour $\bullet$, on a $r \bullet \pi = \pi$. Or, comme $\psi'$ est de taille $n-1$, on a par (HR) que $(\pi, i) \vDash_{\mathcal{K}} \psi'$, c'est à dire $(r \bullet \pi, i) \vDash_{\mathcal{K}} \psi'$. Donc $(\pi, i) \vDash_{\mathcal{K}} \psi \mathrel{-\!\!*} \psi'$ et le lemme est vérifié dans ce cas.

- Cas $\phi = \Diamond\psi$
  Comme $\psi$ est de taille $n-1$, on a par (HR) que $(\pi, i) \vDash_{\mathcal{K}} \psi$. Or on a $i \geq i$, donc $(\pi, i) \vDash_{\mathcal{K}} \Diamond\psi$ et le lemme est vérifié dans ce cas.
- Cas $\phi = \Box\psi$
  Soit $j \in S$ tel que $j \geq i$. Par (HR), comme $\psi$ est de taille $n-1$, on a $(\pi, j) \vDash_{\mathcal{K}} \psi$. Donc $(\pi, i) \vDash_{\mathcal{K}} \Box\psi$ et le lemme est vérifié dans ce cas.
- Cas $\phi = \bigcirc\psi$
  Par (HR), comme $\psi$ est de taille $n-1$, on a $(\pi, i+1) \vDash_{\mathcal{K}} \psi$. Donc $(\pi, i) \vDash_{\mathcal{K}} \bigcirc\psi$ et le lemme est vérifié dans ce cas.

□

Nous allons exprimer une dernière propriété qui donnera lieu à des perspectives intéressantes que nous explorerons dans la section 3.6.

**Lemme 3.3.** *Les équivalences logiques suivantes sont vérifiées pour toute formule $\phi$ :*

1. $\Diamond\phi \equiv \phi \vee \bigcirc\Diamond\phi$
2. $\Box\phi \equiv \phi \wedge \bigcirc\Box\phi$

*Démonstration.*
Soit $\phi$ une formule, $\mathcal{K}$ un modèle, $r \in R$ une ressource et $i \in S$ un instant.

1. Sens direct
   On suppose que $(r,i) \vDash_{\mathcal{K}} \Diamond\phi$. Il existe alors $j \in S$ tel que $j \geq i$ et $(r,j) \vDash_{\mathcal{K}} \phi$. Supposons que $j = i$. On a alors $(r,i) \vDash_{\mathcal{K}} \phi$. Dans le cas contraire, on a $j > i$, c'est à dire $j \geq i+1$. Par conséquent, $(r, i+1) \vDash_{\mathcal{K}} \Diamond\phi$, et par suite $(r,i) \vDash_{\mathcal{K}} \bigcirc\Diamond\phi$. Ainsi, on a soit $(r,i) \vDash_{\mathcal{K}} \phi$, soit $(r,i) \vDash_{\mathcal{K}} \bigcirc\Diamond\phi$, donc $(r,i) \vDash_{\mathcal{K}} \phi \vee \bigcirc\Diamond\phi$. Donc $\Diamond\phi \vDash \phi \vee \bigcirc\Diamond\phi$
   Sens réciproque
   On suppose que $(r,i) \vDash_{\mathcal{K}} \phi \vee \bigcirc\Diamond\phi$. On a donc soit $(r,i) \vDash_{\mathcal{K}} \phi$, soit $(r,i) \vDash_{\mathcal{K}} \bigcirc\Diamond\phi$. Si $(r,i) \vDash_{\mathcal{K}} \phi$, comme $i \geq i$ on a aussi $(r,i) \vDash_{\mathcal{K}} \Diamond\phi$. Si $(r,i) \vDash_{\mathcal{K}} \bigcirc\Diamond\phi$, alors $(r, i+1) \vDash_{\mathcal{K}} \Diamond\phi$, donc il existe $j \geq i+1$ tel que $(r,j) \vDash_{\mathcal{K}} \phi$. Comme $j \geq i+1$, on a en particulier $j \geq i$ et donc $(r,i) \vDash_{\mathcal{K}} \Diamond\phi$. Donc $\phi \vee \bigcirc\Diamond\phi \vDash \Diamond\phi$
2. Sens direct
   On suppose que $(r,i) \vDash_{\mathcal{K}} \Box\phi$. On a alors $(r,j) \vDash_{\mathcal{K}} \phi$ pour tout $j \in S$ tel que $j \geq i$. Comme $i \geq i$, on a donc en particulier $(r,i) \vDash_{\mathcal{K}} \phi$. Soit $k \in S$ tel que $k \geq i+1$. Alors on a également $k \geq i$, et par conséquent $(r,k) \vDash_{\mathcal{K}} \phi$, et ce pour tout $k \geq i+1$. Donc $(r, i+1) \vDash_{\mathcal{K}} \Box\phi$ et donc $(r,i) \vDash_{\mathcal{K}} \bigcirc\Box\phi$. Finalement, on a montré que $(r,i) \vDash_{\mathcal{K}} \phi \wedge \bigcirc\Box\phi$. Donc $\Box\phi \vDash \phi \wedge \bigcirc\Box\phi$.
   Sens réciproque
   On suppose que $(r,i) \vDash_{\mathcal{K}} \phi \wedge \bigcirc\Box\phi$. On a donc $(r,i) \vDash_{\mathcal{K}} \phi$ (1) et $(r,i) \vDash_{\mathcal{K}} \bigcirc\Box\phi$ (2). Soit $j \in S$ tel que $j \geq i$. Si $i = j$, alors on a $(r,j) \vDash_{\mathcal{K}} \phi$ par (1). Sinon, $j > i$, et donc $j \geq i+1$. De (2) on tire $(r, i+1) \vDash_{\mathcal{K}} \Box\phi$ et donc, $(r,j) \vDash_{\mathcal{K}} \phi$. Ainsi, dans tous les cas on a $(r,j) \vDash_{\mathcal{K}} \phi$ et ce pour tout $j \geq i$, donc $(r,i) \vDash_{\mathcal{K}} \Box\phi$. Donc $\phi \wedge \bigcirc\Box\phi \vDash \Box\phi$.

□

Ces équivalences sont parfois utilisées, dans LTL, comme base pour la définition d'opérateurs. En effet, si on a auparavant défini l'opérateur $\circ$, on peut définir $\Diamond$ comme l'opérateur $op$ satisfaisant l'égalité suivante pour toute formule $\phi$ :

$$op(\phi) \stackrel{\text{def}}{=} \phi \vee \circ op(\phi)$$

Et de même $\Box$ est la solution de :

$$op(\phi) \stackrel{\text{def}}{=} \phi \wedge \circ op(\phi)$$

Cette vision des choses, où les opérateurs temporels sont en quelque sorte des points fixes fonctionnels, est à l'origine d'un calcul des tableaux pour LTL ayant des liens forts avec les automates de Büchi [65]. La conception d'un calcul des tableaux pour LTBI inspirée par cette pratique, est explorée dans la section 3.6.

## 3.3 Expressivité de LTBI

Nous allons maintenant présenter quelques exemples concrets de l'application possible de LTBI à la modélisation de systèmes.

### 3.3.1 Consommation retardée de ressources

En premier lieu, on veut illustrer l'intérêt de l'ajout d'un dynamisme temporel en ajoutant un tel dynamisme à un exemple statique utilisant BI. Ce premier exemple est fortement inspiré de l'exemple pour DBI trouvé dans [32].

On se propose de reconsidérer l'exemple que nous avions développé en second dans le Chapitre 2. Pour rappel, il s'agit de considérer un ensemble de ressources élémentaires $Res = \{e, p_1, p_2, c_{50}, c_{20}, c_{10}\}$ modélisant des pièces d'euros et de centimes d'euro, qu'on étend par la composition $\bullet$ en un ensemble $R$. On définit une fonction *sum* qui associe à une ressource la valeur monétaire associée (par exemple, $sum(p_1 \bullet p_2 \bullet c_{20}) = 3.20$). On donne aussi un ensemble de propositions $Prop = \{Obj_{(0.30)}, Obj_{(1.70)}, Obj_{(2)}\}$ où chacun des $Obj_{(x)}$ représente la capacité d'acheter au moins un objet à $x$ euros, et on convient intuitivement que $r \vDash \phi$ signifie que la somme d'argent représentée par $r$ permet d'effectuer au moins l'opération décrite par $\phi$. Par exemple on peut énoncer la chose suivante :

$$p_2 \vDash Obj_{(1)} \wedge (Obj_{(0.30)} * Obj_{(1.70)})$$

qui signifie qu'avec la somme représentée par une pièce de 2 euros, on peut à la fois acheter un objet à 1 euro ou bien diviser la somme en deux parties, chacune permettant d'acheter un objet à 30 centimes et un objet à 1,70 euros.

Considérons maintenant que le prix des objets que l'on représente évolue dans le temps. On représente par exemple par le tableau suivant (Figure 3.3) le prix de trois objets sur plusieurs années.

Comme précédemment, on représente par une proposition atomique le fait de pouvoir acheter un objet de ce tableau. On note $Obj_A$, $Obj_B$ et $Obj_C$ ces trois propositions. L'espace des

| | 2018 | 2019 | 2020 |
|---|---|---|---|
| Objet A | 1 euro | 1,10 euros | 1,20 euros |
| Objet B | 0,30 euros | 0,25 euros | 0,35 euros |
| Objet C | 1,70 euros | 1,80 euros | 1,50 euros |

FIGURE 3.3 – Prix de trois objets évoluant dans le temps

états temporels qu'on considère est simplement le triplet $S = \{2018, 2019, 2020\}$. On a alors assez naturellement que pour tout $r \in R$ et tout $s \in S$, si la valeur $x$ est dans la case $(Objet\ X, s)$ du tableau de la Figure 3.3 :

$$(r,s) \vDash Obj_X \text{ ssi } sum(r) \geq x$$

Par exemple, on aura $(p_1, 2018) \vDash Obj_A$ car on peut acheter l'objet A avec 1 euro en 2018. On a aussi $(p_1, 2020) \vDash Obj_B$ même si dans ce cas la valeur monétaire est plus importante que le prix de l'objet. En revanche, on a $(c_{50}, 2019) \nvDash Obj_B$ car 50 cents sont insuffisants pour acheter B en 2019.

L'ajout de nos nouveaux opérateurs permet alors d'exprimer ce qui suit :

- $(r,i) \vDash_{\mathcal{K}} \Diamond\phi$ s'il existe une année postérieure à $i$ telle que l'opération $\phi$ soit réalisable cette année là avec la somme $r$ ;
- $(r,i) \vDash_{\mathcal{K}} \Box\phi$ si l'opération $\phi$ est réalisable toutes les années postérieures à $i$ avec la somme $r$ ;
- $(r,i) \vDash_{\mathcal{K}} \bigcirc\phi$ si l'opération $\phi$ est réalisable l'année $i+1$ avec la somme $r$.

On peut alors formuler des propositions plus complexes, comme la formule suivante :

$$(p_2, 2018) \vDash (\Box Obj_B) * ((\Diamond Obj_C) \wedge (Obj_A * (\bigcirc Obj_B)))$$

Cette formule atteste que je peux, en 2018, diviser la somme représentée par $p_2$ en deux autres sommes, l'une permettant de toujours acheter l'objet B, l'autre permettant à la fois d'acheter un jour l'objet C et d'obtenir deux autres sommes, l'une suffisante pour acheter cette année l'objet A, l'autre pour acheter l'année prochaine l'objet B.

Vérifions que cette formule est bien valide :

1. $(p_1, 2018) \vDash Obj_A$ par définition.
2. $(c_{20} \bullet c_{10}, 2019) \vDash Obj_B$ par définition.
3. Par 2., $(c_{20} \bullet c_{10}, 2018) \vDash \bigcirc Obj_B$.
4. On a $(c_{20} \bullet c_{10}) \bullet p_1 \sqsubseteq (p_1 \bullet c_{50})$. Alors, par 1. et 3., $(p_1 \bullet c_{50}, 2018) \vDash Obj_A * (\bigcirc Obj_B)$.
5. $(p_1 \bullet c_{50}, 2020) \vDash Obj_C$ par définition.
6. Par 5., on a $(p_1 \bullet c_{50}, 2018) \vDash \Diamond Obj_C$.
7. Par 4. et 6. on a $(p_1 \bullet c_{50}, 2018) \vDash ((\Diamond Obj_C) \wedge (Obj_A * (\bigcirc Obj_B))$
8. $(c_{50}, 2018) \vDash Obj_B$, $(c_{50}, 2019) \vDash Obj_B$ et $(c_{50}, 2020) \vDash Obj_B$ par définition.

9. Par 8., on a $(c_{50}, 2018) \vDash \Box Obj_B$
10. On a $(p_1 \bullet c_{50}) \bullet c_{50} \sqsubseteq p_2$. Alors, par 7. et 9., $(p_2, 2018) \vDash (\Box Obj_B) * ((\Diamond Obj_C) \wedge (Obj_A * (\bigcirc Obj_B)))$.

Notons que chaque étape de ce raisonnement a un sens simple dans notre modèle :

1. 1 euro permet d'acheter l'objet A en 2018.
2. 30 cents permettent d'acheter l'objet B en 2019.
3. Par 2., 30 cents permettent d'acheter l'objet B l'année prochaine en 2018.
4. Comme 1 euro et 30 cents se composent en moins de 1,50 euros, par 1. et 3. on a que 1,50 euros permettent à la fois d'acheter l'objet A et d'acheter l'objet B l'année prochaine en 2018.
5. 1,50 euros permet d'acheter l'objet C en 2020.
6. Par 5., 1,50 euros permettra d'acheter l'objet C un jour en 2018.
7. Par 4. et 6., 1,50 euros permettent d'acheter à la fois l'objet A et l'objet B l'année prochaine, mais aussi d'acheter un jour l'objet C en 2018.
8. 50 cents permettent d'acheter l'objet B à la fois en 2018, en 2019 et en 2020.
9. Par 8., 50 cents permettent toujours d'acheter l'objet B à partir de 2018.

On voit ainsi comment l'ajout des opérateurs de LTBI permet de décrire des situations de consommation de ressources qui évoluent dans le temps.

### 3.3.2 Description évolutive d'un système cloisonné

Nous venons de voir comment la dimension temporelle de LTBI permettait d'étendre des modélisations effectuées sur des systèmes statiques avec BI.

A contrario, les logiques temporelles comme LTL dont on s'inspire pour nos opérateurs temporels ont aussi été beaucoup utilisées pour modéliser des systèmes. Dans de tels modélisations, l'opérateur $\bigcirc$ est particulièrement utile pour illustrer la transition d'un système d'un état vers un autre. Les opérateurs $\Diamond$ et $\Box$ peuvent alors être utilisées pour énoncer des propriétés (imposées par le système ou souhaitées) sur l'évolution du système dans le temps.

L'apport de BI à de telles modélisations est l'ouverture à la possibilité de structurer des états distincts d'un système. Bien évidemment, l'exemple typique est la modélisation d'espaces mémoires, mais en guise d'exemple plus simple et expressif, on se propose de modéliser la gestion d'un ensemble de salles et de personnel.

On considère un ensemble $R$ de ressources représentant différents lieux d'une entreprise (bâtiments, salles, sites, etc...). La relation entre ressources $\sqsubseteq$ symbolisera l'inclusion d'un tel lieu dans un autre. Par exemple, si la salle $s$ est dans le bâtiment $b$, on écrira $s \sqsubseteq b$. On construit en outre une loi interne $\bullet$ désignant simplement les multi-ensembles de lieux, par exemple $s_1 \bullet s_2 \sqsubseteq b$ signifie que les salles $s_1$ et $s_2$, mises ensembles, forment un nouveau lieu faisant partie du bâtiment $b$. On retrouve l'idée propre à BI que $\bullet$ indique la séparation stricte de deux entités.

On considère également un ensemble d'instants $S \subseteq \mathbb{N}$ ainsi qu'un ensemble de formules propositionnelles $A, B, \ldots$ qui représenteront chacune la faculté d'une personne d'occuper un

lieu en un instant donné. On aura par exemple que $b,t \vDash A$ si la personne $A$ est dans le bâtiment $b$ à l'instant $t$.

Notons que cette définition respecte bien le principe de monotonie énoncé dans le Lemme 3.1. En effet, si $(s,t) \vDash A$ et $s \sqsubseteq b$, cela veut dire que $A$ est dans la salle $s$ à l'instant $t$ et que $s$ est incluse dans le bâtiment $b$. On a donc bien que $A$ est aussi dans le bâtiment $b$ et on peut écrire $(b,t) \vDash A$.

On considère à présent la formule suivante :

$$\phi = (A \wedge B \wedge \circ A) * (\circ B)$$

Si on a $b,t \vDash \phi$, alors le bâtiment $b$ comporte deux salles $s_1$ et $s_2$ avec $s_1 \bullet s_2 \sqsubseteq b$ et on a d'une part $s_1,t \vDash A$, $s_1,t \vDash B$ et $s_1,t \vDash \circ A$ et d'autre part $s_2,t \vDash \circ B$. C'est à dire qu'on a d'une part $s_1,t \vDash A$, $s_1,t \vDash B$ et $s_1,t+1 \vDash A$ et d'autre part $s_2,t+1 \vDash B$. On vient ainsi de décrire une transition de notre système où la personne $B$ quittait une salle, initialement occupée par $A$ et elle-même, pour une autre.

On peut ainsi avec de telles formules modéliser les évolutions possibles de notre système. On peut aussi énoncer des propriétés du système. Par exemple, la propriété suivante assure que la personne $A$ n'est jamais à deux endroits en même temps :

$$\neg(\Diamond(A * A))$$

De manière similaire, on peut énoncer des propriétés qu'on souhaite vérifier. Par exemple, étant donné un ensemble de transitions et de propriétés du système, on peut chercher à démontrer que $A$ se retrouve à un moment dans la même salle que $B$, en montrant qu'on peut déduire des formules décrivant transitions et propriétés la formule suivante :

$$\Diamond(A * B)$$

Cette approche de modélisation est fortement inspirée des techniques de model-checking [26,27,84]. Un autre exemple de modélisation avec un telle démarche est celle du langage B, lui aussi basé sur les logiques temporelles [1].

### 3.3.3 Vers une logique temporelle branchante

Nous avons expliqué dans la section 3.1 comment nous avions choisi la logique temporelle linéaire LTL comme base pour notre extension temporelle de BI. Toutefois, comme nous avons pu l'esquisser, les logiques temporelles branchantes comme CTL [25] ont un pouvoir d'expression bien plus large et ont été beaucoup utilisées dans le model checking [26,65]. Nous revenons ici sur ce choix et nous montrons comment LTBI peut en réalité être exploitée comme une forme de logique temporelle branchante.

Pour rappel, aux opérateurs $\Diamond$, $\Box$ et $\circ$ (notés **F**, **G**, et **X**) de LTL, CTL ajoute deux opérateurs **A** et **E** qui permettent respectivement de quantifier universellement et existentiellement sur les futurs possibles. Ainsi **EF**$\phi$ signifie qu'il existe au moins un avenir et au moins un instant dans cet avenir tel que $\phi$ soit réalisée.

La différence d'expressivité entre les logiques temporelles linéaires et branchantes se perçoit non seulement dans la sémantique des logiques (on rappelle que le temps branchant suppose

des instants ordonnés tandis que le linéaire suppose des instants totalement ordonnés) mais aussi dans les formules qu'elles permettent d'énoncer. Une formule discriminante entre les deux formalismes est la suivante :

$$\mathcal{T} \equiv \Diamond A \wedge \Diamond B \rightarrow \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$$

Intuitivement, cette formule exprime que si $A$ est promis un jour et que $B$ l'est aussi, alors on doit avoir $A$ un jour puis $B$ plus tard, ou $B$ un jour et $A$ plus tard. Cette formule est valide dans un cadre sémantique linéaire (comme LTL) mais pas dans cadre branchant (comme CTL). En effet, dans ce dernier, on peut avoir $A$ et $B$ vrais dans deux futurs possibles différents, sans relation l'un avec l'autre, et on a alors aucun contrôle sur l'ordre temporel entre $A$ et $B$.

Puisque LTBI s'appuie sur un formalisme temporel linéaire, comme LTL, il est assez évident de montrer que la formule $\mathcal{T}$ est aussi valide dans LTBI. Toutefois, on peut considérer la version légèrement altérée suivante :

$$\mathcal{T}' \equiv \Diamond A * \Diamond B \rightarrow \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$$

$\mathcal{T}'$ n'est pas valide dans LTBI, mais en analysant sa sémantique on se rend compte que le sens de cette formule rejoint les réflexions que nous avions sur les logiques branchantes. En effet, $\mathcal{T}'$ n'est pas valide car la partie gauche de l'équivalence signifie que $A$ évolue avec une ressource tandis que $B$ évolue avec une autre, elles ne peuvent donc pas se retrouver avec la même ressource pour satisfaire une des parties de la disjonction à droite. Une interprétation de ce phénomène est que $A$ et $B$ évoluent dans deux lignes temporelles différentes, créées par la séparation de l'opérateur $*$.

On a ainsi recréé avec LTBI une forme de temps branchant où l'on spécifie explicitement le nombre et la nature des branches temporelles considérées, au contraire de CTL où les branches temporelles étaient quantifiées existentiellement ou universellement. On pourrait en outre recouvrer les opérateurs **E** et **A** de CTL en étendant la sémantique de BI par de nouveaux opérateurs [3].

On peut ainsi, à l'aide de LTBI, exprimer des propriétés temporelles branchantes différentes de celles usuellement décrites par les logiques temporelles branchantes, probablement tout en recouvrant l'expressivité de ces dernières. Ces travaux ne sont qu'à l'état d'ébauche mais présentent du potentiel, notamment dans l'esprit de la vérification de programmes. En effet, comme nous l'avons déjà mentionné, le succès de la logique de séparation [87] place BI et sa famille au centre de la modélisation d'espaces mémoire. Dans ce contexte, utiliser LTBI permettrait de modéliser des exécutions parallèles de threads ou de programmes dans des systèmes multiprocesseurs [90].

## 3.4 Un calcul des tableaux pour LTBI

Nous allons maintenant présenter un calcul pour notre nouvelle logique. C'est un calcul des tableaux directement inspirée de celui de DBI [34]. Comme ce dernier, il présente deux séries de labels, l'une pour gérer les ressources, l'autre pour les états modaux, ici temporels. La différence avec le calcul de DBI est que les contraintes d'états doivent ici traduire la spécificité

3. Il se peut que composer les opérateurs actuels de BI suffise à recouvrer la sémantique de CTL dans cet esprit. Ce travail n'est pas encore exploré.

des instants temporels de LTBI, à savoir un ordre total et un isomorphisme avec les entiers naturels.

Le premier point est assuré par la présence de *règles temporelles* qui permettent d'ordonner totalement les labels d'état au fur et à mesure de leur introduction. Le deuxième point, en revanche, est problématique. Garantir l'isomorphisme d'ordre avec $\mathbb{N}$ n'est pas direct, il faut garantir l'impossibilité d'avoir une infinité de labels entre deux autres. Si nous avons des pistes pour parvenir à ce résultat, le travail n'est pas complètement abouti. Nous formulons donc ce problème sous la forme de la Conjecture 3.1 ci-après et nous discutons des difficultés rencontrées pour la prouver ainsi que des pistes envisagées pour le faire.

### 3.4.1 Labels et contraintes

Comme pour les calculs autour de BI, nous utilisons des *labels* pour représenter les ressources pouvant possiblement valider une formule et des relations entre ces labels, appelées *contraintes*, pour marquer les relations nécessaires entre ces ressources. Comme LTBI est basé sur une sémantique à deux dimensions (séparation et temporel), il est nécessaire d'utiliser deux familles de labels, chacun codant pour l'une des dimensions.

**Définition 3.6** (Labels et contraintes de ressources)**.** *On note $L_r$ un ensemble de* labels de ressources *construit sur une constante $1_r$, un ensemble infini dénombrable $\gamma_r = \{c_1, c_2, ...\}$ et une fonction interne notée $\circ$ par la grammaire suivante :*

$$X ::= 1_r \mid c_i \mid X \circ X$$

*où $c_i \in \gamma_r$. On impose de plus que $\circ$ soit une fonction sur $L_r$ qui est associative, commutative et a $1_r$ comme élément neutre.*

*Une* contrainte de ressources *est une expression de la forme $x \leqslant y$ où $x$ et $y$ sont des labels de ressources.*

De plus, on dit que $x$ est un *sous-label* de $y$ ssi il existe $z \in L_r$ tel que $x \circ z = y$. L'ensemble des sous-labels d'un label $x$ est noté $\mathcal{E}(x)$. Par commodité d'écriture, on omettra $\circ$ dès que possible et on écrira $xy$ à la place de $x \circ y$.

Notons que, contrairement à ce que nous avons présenté pour BBI dans le Chapitre 2 où nous avions utilisé la structure de mots (c'est à dire de multi-ensembles), nous définissons ici explicitement la composition entre labels de ressources, même si nous l'omettons par commodité dans les exemples pratiques.

Définissons maintenant des notations similaires pour encoder les états temporels.

**Définition 3.7** (Labels et contraintes d'états)**.** *On appelle* label d'états *tout élément d'un ensemble infini dénombrable $L_s = \{l_1, l_2, \ldots\}$.*

*Une* contrainte d'états *est une expression d'une des quatre formes suivantes :*

$$x \lhd y \qquad x \blacktriangleleft y \qquad x \simeq y \qquad x \not\simeq y$$

*où $x$ et $y$ sont des labels d'états. On utilisera la notation $x \star y$ pour dénoter l'une de ces quatre relations arbitrairement.*

Ces quatre types de contraintes nous permettent d'exprimer les liens entre labels d'états qui font écho aux liens entre entiers naturels (qui dans notre sémantique représentent les instants temporels). Lors de l'extraction de contre-modèles présentée dans la section 3.5.2, nous établirons formellement les liens entre ces contraintes et l'ordre de $\mathbb{N}$. Toutefois, nous pouvons en donner ici une idée intuitive :

- $x \lhd y$ signifie que $y$ est un état après $x$ ;
- $x \blacktriangleleft y$ signifie que $y$ est l'état suivant $x$ ;
- $x \simeq y$ signifie que $x$ et $y$ sont équivalents ;
- $x \not\simeq y$ signifie que $x$ et $y$ ne sont pas équivalents ;

Nous allons maintenant pouvoir utiliser des ensembles de contraintes de ressources et de contraintes d'états pour caractériser nos labels dans nos tableaux.

**Définition 3.8** (Domaine et alphabet)**.** *Soit $\mathcal{C}_r$ un ensemble de contraintes de ressources. Le* domaine *de $\mathcal{C}_r$, noté $\mathcal{D}_r(\mathcal{C}_r)$, est l'ensemble de tous les sous-labels apparaissant dans $\mathcal{C}_r$. En particulier, $\mathcal{D}_r(\mathcal{C}_r) = \bigcup_{x \leqslant y \in \mathcal{C}_r} (\mathcal{E}(x) \cup \mathcal{E}(y))$.*

*L'*alphabet *d'un ensemble de contraintes de ressources $\mathcal{C}_r$(resp. un ensemble de contraintes d'états $\mathcal{C}_s$) est l'ensemble de tous les constantes de labels apparaissant $\mathcal{C}_r$ (resp. $\mathcal{C}_s$).*

*On a en particulier $\mathcal{A}_r(\mathcal{C}_r) = \gamma_r \cap \mathcal{D}_r(\mathcal{C}_r)$ et $\mathcal{A}_s(\mathcal{C}_s) = \cup_{x \star y} \{x, y\}$.*

Comme nous l'avons indiqué, l'objectif est de capturer par les contraintes les propriétés de la relation $\sqsubseteq$ Or, cette relation est réflexive, transitive et compatible. Pour transmettre ces propriétés à nos contraintes, nous clôturons nos ensemble de contraintes par une série de règles.

**Définition 3.9** (Clôture d'ensemble de contraintes de ressources)**.** *Soit $\mathcal{C}_r$ un ensemble de contraintes de ressources. La* clôture *de $\mathcal{C}_r$ (noté $\overline{\mathcal{C}_r}$) est le plus petit ensemble tel que $\mathcal{C}_r \subseteq \overline{\mathcal{C}_r}$ et qui est clos par l'ensemble de règles suivant :*

$$\frac{x \leqslant y \qquad y \leqslant z}{x \leqslant z}\ \langle t_r \rangle \qquad \frac{xy \leqslant xy}{x \leqslant x}\ \langle d_r \rangle$$

$$\frac{ky \leqslant ky \qquad x \leqslant y}{kx \leqslant ky}\ \langle c_r \rangle \qquad \frac{x \leqslant y}{x \leqslant x}\ \langle l_r \rangle \qquad \frac{x \leqslant y}{y \leqslant y}\ \langle r_r \rangle$$

On notera que, comme ces règles n'introduisent pas de nouvelles constantes de labels de ressources, on a $\mathcal{A}_r(\mathcal{C}_r) = \mathcal{A}_r(\overline{\mathcal{C}_r})$.

Comme nous le cherchions, ces règles reflètent les propriétés de $\sqsubseteq$. $\langle t_r \rangle$ traduit la transitivité, $\langle c_r \rangle$ et $\langle d_r \rangle$ la compatibilité, $\langle l_r \rangle$ et $\langle r_r \rangle$ la réflexivité.

**Proposition 3.1.** *Soit $\mathcal{C}_r$ un ensemble de contraintes de ressources :*

1. *Si $kx \leqslant y \in \overline{\mathcal{C}_r}$, alors $x \leqslant x \in \overline{\mathcal{C}_r}$*
2. *Si $x \leqslant ky \in \overline{\mathcal{C}_r}$, alors $y \leqslant y \in \overline{\mathcal{C}_r}$*

*Démonstration.*

1. Supposons que $kx \leqslant y \in \overline{\mathcal{C}_r}$. On peut établir l'arbre de déduction suivant :

$$\dfrac{\dfrac{\dfrac{kx \leqslant y}{kx \leqslant kx}\ \langle l_r \rangle}{xk \leqslant xk}\ kx = xk}{x \leqslant x}\ \langle d_r \rangle$$

   Par conséquent, $x \leqslant x \in \overline{\mathcal{C}_r}$.

2. Supposons que $x \leqslant ky \in \overline{\mathcal{C}_r}$. On peut établir l'arbre de déduction suivant :

$$\dfrac{\dfrac{\dfrac{x \leqslant ky}{ky \leqslant ky}\ \langle r_r \rangle}{yk \leqslant yk}\ ky = yk}{y \leqslant y}\ \langle d_r \rangle$$

   Par conséquent, $y \leqslant y \in \overline{\mathcal{C}_r}$.

□

**Corollaire 3.1.** *Soit $\mathcal{C}_r$ un ensemble de contraintes de ressources. $x \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ ssi $x \leqslant x \in \overline{\mathcal{C}_r}$.*

*Démonstration.* Montrons le sens direct.

Soit $x \in \mathcal{D}_r(\overline{\mathcal{C}_r})$. Comme $\mathcal{D}_r(\overline{\mathcal{C}_r}) = \bigcup_{a \leqslant b \in \mathcal{D}_r(\overline{\mathcal{C}_r})}(\mathcal{E}(a) \cup \mathcal{E}(b))$, il existe $a \leqslant b \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ tel que $x \in \mathcal{E}(a)$ ou $x \in \mathcal{E}(b)$.

Supposons que $x \in \mathcal{E}(a)$. Cela signifie que $x$ est un sous-label de $a$. Donc il existe $k \in L_r$ tel que $a = kx$. On a alors $kx \leqslant b \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ et par la Proposition 3.1 on a $x \leqslant x \in \overline{\mathcal{C}_r}$.

De même, si $x \in \mathcal{E}(b)$, il existe $k$ tel que $b = kx$, $a \leqslant kx \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ et par la même proposition $x \leqslant x \in \overline{\mathcal{C}_r}$.

Le sens réciproque est trivial, par définition de $\mathcal{D}_r()$ . □

Il nous faut maintenant définir une clôture similaire pour les labels d'états. Comme nous l'avons mentionné plus haut, $\lhd$ doit refléter l'ordre entre entiers, $\blacktriangleleft$ la succession entre entiers, $\simeq$ l'équivalence et $\not\simeq$ la non équivalence.

**Définition 3.10** (Clôture d'ensemble de contraintes d'états)**.** *Soit $\mathcal{C}_s$ un ensemble de contraintes d'états. La* clôture *de $\mathcal{C}_s$ (notée $\overline{\mathcal{C}_s}$) est le plus petit ensemble tel que $\mathcal{C}_s \subseteq \overline{\mathcal{C}_s}$ et qui est clos par l'ensemble de règles suivant :*

$$\frac{x \lhd y \qquad y \lhd z}{x \lhd z}\ \langle t_s \rangle \qquad \frac{x \lhd y \qquad y \lhd x}{x \simeq y}\ \langle a_s \rangle$$

$$\frac{x \simeq y}{y \simeq x}\ \langle s_e \rangle \qquad \frac{x \not\simeq y}{y \not\simeq x}\ \langle s_{ne} \rangle$$

$$\frac{x \blacktriangleleft y}{x \not\simeq y \qquad x \lhd y}\ \langle e_n \rangle \qquad \frac{x \blacktriangleleft y \qquad y \lhd z}{x \not\simeq z}\ \langle t_n \rangle$$

$$\frac{x \blacktriangleleft y \qquad x \blacktriangleleft z}{y \simeq z}\ \langle u_n \rangle \qquad \frac{x \blacktriangleleft y \qquad x \lhd z \qquad z \lhd y \qquad z \not\simeq y}{x \simeq z}\ \langle i_n \rangle$$

$$\frac{x \star y \qquad y \simeq z}{x \star z}\ \langle r_r \rangle \qquad \frac{x \star y \qquad x \simeq z}{z \star y}\ \langle r_l \rangle \qquad \frac{x \simeq y}{x \lhd y}\ \langle w_s \rangle$$

$$\frac{x \star y}{x \lhd x}\ \langle e_l \rangle \qquad \frac{x \star y}{y \lhd y}\ \langle e_r \rangle \qquad \frac{x \lhd x}{x \simeq x}\ \langle g_e \rangle$$

Dans les règles de remplacement ($\langle r_l \rangle$ et $\langle r_r \rangle$) et d'extraction ($\langle e_l \rangle$ et $\langle e_r \rangle$), $\star$ peut être n'importe quelle des quatre relations ($\leqslant$, $\blacktriangleleft$, $\simeq$, $\not\simeq$). Toutefois, la règle concernée doit être la même dans les prémisses et dans les conclusions de la règle.

Les règles $\langle t_s \rangle$, $\langle e_l \rangle$, $\langle e_r \rangle$ et $\langle a_s \rangle$ traduisent respectivement la transitivité, la réflexivité et l'asymétrie de $\lhd$. $\langle w_s \rangle$ exprime le fait que $\simeq$ est un cas particulier de $\lhd$. $\langle s_e \rangle$ et $\langle s_{ne} \rangle$ traduisent la symétrie de $\simeq$ et $\not\simeq$. $\langle e_n \rangle$ exprime qu'un successeur est plus grand et différent. $\langle t_n \rangle$ traduit qu'un label plus grand qu'un successeur est également différent de l'élément d'origine. $\langle u_n \rangle$ traduit l'unicité du successeur. $\langle i_n \rangle$ exprime qu'il ne peut y avoir d'élément entre un élément et son successeur. $\langle r_r \rangle$ et $\langle r_l \rangle$ expriment qu'on peut remplacer un label par son équivalent. Elles expriment également la transitivité de $\simeq$. Enfin, $\langle g_e \rangle$ traduit le fait que la réflexivité de $\lhd$ et de $\simeq$ sont équivalentes.

Tout comme pour les contraintes de ressources, aucune de ces règles n'introduit de nouvelles constantes de labels d'états, et on a donc $\mathcal{A}_s(C_s) = \mathcal{A}_s(\overline{C_s})$.

Nous présentons maintenant quelques propriétés intrinsèques aux ensembles de contraintes.

**Proposition 3.2.** *Soient $C_r$ et $C'_r$ deux ensembles de contraintes de ressources, $C_s$ et $C'_s$ deux ensembles de contraintes d'états.*

- *Si $C_r \subseteq C'_r$ et $x \leqslant y \in \overline{C_r}$, alors $x \leqslant y \in \overline{C'_r}$*
- *Si $C_s \subseteq C'_s$ et $x \star y \in \overline{C_s}$, alors $x \star y \in \overline{C'_s}$*

*Démonstration.*
On montre le résultat pour les contraintes de ressources, la preuve est similaire pour les contraintes d'états. On démontre par récurrence sur la taille de l'arbre permettant d'obtenir la contrainte.

- Cas de base
  Soit $x \leqslant y$ une contrainte obtenue par un arbre de taille 0. On a alors $x \leqslant y \in C_r$, et comme $C_r \subseteq C'_r$, on a $x \leqslant y \in C'_r$.

- Hérédité
  On suppose que la propriété est vraie pour toutes les contraintes obtenue par un arbre de taille $n$. On considère une contrainte $x \leqslant y \in \overline{\mathcal{C}_r}$ obtenue par un arbre de taille $n+1$. Alors, quelque soit la dernière règle utilisée pour obtenir $x \leqslant y$, ses prémisses sont de taille $n$. Par hypothèse de récurrence, ces prémisses sont toutes dans $\overline{\mathcal{C}'_r}$. Par clôture par la même règle dans $\overline{\mathcal{C}'_r}$, on montre alors que $x \leqslant y \in \overline{\mathcal{C}'_r}$

□

**Lemme 3.4** (Compacité)**.** *Soit $\mathcal{C}_r$ (resp. $\mathcal{C}_s$) un ensemble de contraintes de ressources (resp. d'états) possiblement infini. Si $x \leqslant y \in \overline{\mathcal{C}_r}$ (resp. $x \star y \in \overline{\mathcal{C}_s}$) alors il existe un ensemble de contraintes fini $\mathcal{C}_f$ tel que $\mathcal{C}_f \subseteq \mathcal{C}_r$ (resp. $\mathcal{C}_f \subseteq \mathcal{C}_s$) et $x \leqslant y \in \overline{\mathcal{C}_f}$ (resp. $x \star y \in \overline{\mathcal{C}_s}$).*

*Démonstration.*
On donne la preuve pour les labels de ressources, la preuve est similaire pour les labels d'états. La preuve se fait par récurrence sur la taille de l'arbre ayant permis d'obtenir $x \leqslant y$.

- Cas de base
  Soit $x \leqslant y \in \overline{\mathcal{C}_r}$ une contrainte obtenue par un arbre de taille 0. On a $x \leqslant y \in \mathcal{C}_r$. On pose alors $\mathcal{C}_f = \{x \leqslant y\}$. $\mathcal{C}_f$ est évidemment de caractère fini. En outre, on a bien $x \leqslant y \in \mathcal{C}_f$, par conséquent $x \leqslant y \in \overline{\mathcal{C}f}$ par définition de la clôture. Enfin, comme $x \leqslant y \in \mathcal{C}_r$, $\mathcal{C}_f \subseteq \mathcal{C}_r$. On a donc bien trouvé un ensemble $\mathcal{C}_f$ qui remplit les conditions posées par le lemme.
- Hérédité
  On suppose que le lemme est vrai pour toutes les contraintes obtenues par des arbres de taille $n$. Soit $x \leqslant y \in \overline{\mathcal{C}_r}$ une contrainte obtenue par un arbre de taille $n+1$. On discute selon la règle ayant permis d'obtenir $x \leqslant y$ :
  - Si $x \leqslant y$ a été obtenue par la règle $\langle t_r \rangle$, il existe deux contraintes $x \leqslant z \in \overline{\mathcal{C}_r}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$ qui ont été utilisées comme prémisses et donc l'arbre de déduction est de taille $n$. Par l'hypothèse de récurrence, il existe deux ensembles de contraintes finis $\mathcal{C}_1$ et $\mathcal{C}_2$ tels que $\mathcal{C}_1 \subseteq \mathcal{C}_r$ , $\mathcal{C}_2 \subseteq \mathcal{C}_r$ et $x \leqslant z \in \overline{\mathcal{C}_1}$ et $z \leqslant y \in \overline{\mathcal{C}_2}$.
    On pose $\mathcal{C}_f = \mathcal{C}_1 \cup \mathcal{C}_2$. La réunion de deux ensembles finis est fini, donc $\mathcal{C}_f$ est fini. Comme $\mathcal{C}_1 \subseteq \mathcal{C}_r$ et $\mathcal{C}_2 \subseteq \mathcal{C}_r$, par la Proposition 3.2, on a $\mathcal{C}_f \subseteq \mathcal{C}_r$. Enfin, comme $\mathcal{C}_1 \subseteq \mathcal{C}_f$ et $\mathcal{C}_2 \subseteq \mathcal{C}_f$, on a $x \leqslant z \in \overline{\mathcal{C}_f}$ et $z \leqslant y \in \overline{\mathcal{C}_f}$. Alors, par clôture de $\mathcal{C}_f$ par la règle $\langle t_r \rangle$, on a bien $x \leqslant y \in \overline{\mathcal{C}_f}$ et on a trouvé un ensemble $\mathcal{C}_f$ satisfaisant.
  - La preuve est similaire pour tous les autres cas.

□

### 3.4.2 Pseudo-tableaux pour LTBI

Nous disposons maintenant de structures de labels satisfaisantes qui nous permettent de capturer d'un côté les propriétés des ressources pour les labels de ressources, de l'autre les propriétés des instants temporels pour les labels d'états.

Avant de pouvoir construire des tableaux, nous devons introduire une structure plus faible, que nous appelons *pseudo-tableau*. La différence notable entre tableaux et pseudo-tableaux est que ces derniers n'interdisent pas la densité pour l'ordre des labels d'états, un point qui est le sujet de la Conjecture 3.1 dont nous discuterons dans la section suivante.

Nous décrivons maintenant les structures nécessaires pour associer des labels aux formules.

**Définition 3.11** (Formule étiquetée/ CTSS)**.** *Une* formule étiquetée *est un quadruplé* $(\mathbb{S}, \phi, x, u) \in \{\mathbb{T}, \mathbb{F}\} \times \mathcal{L} \times L_r \times L_s$ *que l'on note* $\mathbb{S}\phi : (x, u)$.

*Un* CTSS *(pour Constrained Temporal Set of Statements, ou ensemble temporel d'énoncés contraints) est un triplet noté* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$*, où* $\mathcal{F}$ *est un ensemble de formules étiquetées,* $\mathcal{C}_r$ *est un ensemble de contraintes de ressources et* $\mathcal{C}_s$ *est un ensemble de contraintes d'états, tels que la propriété suivante, notée* $(P_{CSS})$*, est vérifiée :*

$$\textit{Si } \mathbb{S}\phi : (x, u) \in \mathcal{F}\textit{, alors } x \leqslant x \in \overline{\mathcal{C}_r} \textit{ et } u \lhd u \in \overline{\mathcal{C}_s}$$

Un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est la représentation d'une branche dans laquelle les formules étiquetées sont les éléments de $\mathcal{F}$ et les contraintes sur les labels sont les éléments de $\mathcal{C}_r$ et $\mathcal{C}_s$.

Un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est *fini* si $\mathcal{F}$, $\mathcal{C}_r$ et $\mathcal{C}_s$ sont tous les trois finis.

On définit la relation $\preccurlyeq$ par : $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$ si $\mathcal{F} \subseteq \mathcal{F}'$ et $\mathcal{C}_r \subseteq \mathcal{C}'_r$ et $\mathcal{C}_s \subseteq \mathcal{C}'_s$. De plus on écrit $\langle \mathcal{F}_f, \mathcal{C}_{r_f}, \mathcal{C}_{s_f} \rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ lorsque $\langle \mathcal{F}_f, \mathcal{C}_{r_f}, \mathcal{C}_{s_f} \rangle \preccurlyeq \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ et $\langle \mathcal{F}_f, \mathcal{C}_{r_f}, \mathcal{C}_{s_f} \rangle$ est fini.

**Proposition 3.3.** *Pour tout CTSS* $\langle \mathcal{F}_f, \mathcal{C}_r, \mathcal{C}_s \rangle$ *où* $\mathcal{F}_f$ *est fini, il existe* $\mathcal{C}_{r_f} \subseteq \mathcal{C}_r$ *et* $\mathcal{C}_{s_f} \subseteq \mathcal{C}_s$ *tels que* $\mathcal{C}_{r_f}$ *et* $\mathcal{C}_{s_f}$ *sont finis et* $\langle \mathcal{F}_f, \mathcal{C}_{r_f}, \mathcal{C}_{s_f} \rangle$ *est un CTSS.*

*Démonstration.*
Soit $F = \mathbb{S}\phi : (x, u)$ une formule signée de $\mathcal{F}_f$. Comme $\langle \mathcal{F}_f, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS, la propriété $(P_{CSS})$ nous dit que $x \leqslant x \in \overline{\mathcal{C}_r}$ et $u \lhd u \in \overline{\mathcal{C}_s}$. Le Lemme 3.4 nous indique alors qu'il existe deux ensembles finis $\mathcal{C}_{r_f}(F)$ et $\mathcal{C}_{s_f}(F)$ tels que $\mathcal{C}_{r_f}(F) \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s_f}(F) \subseteq \mathcal{C}_s$ et $x \leqslant x \in \overline{\mathcal{C}_{r_f}(F)}$ et $u \lhd u \in \overline{\mathcal{C}_{s_f}(F)}$.

On définit alors $\mathcal{C}_{r_f} = \bigcup_{F \in \mathcal{F}_f} \mathcal{C}_{r_f}(F)$ et $\mathcal{C}_{s_f} = \bigcup_{F \in \mathcal{F}_f} \mathcal{C}_{s_f}(F)$. Comme $\mathcal{F}_f$ est fini, ces deux unions sont des unions finies et donc $\mathcal{C}_{r_f}$ et $\mathcal{C}_{s_f}$ sont finis. En outre, comme pour tout $F \in \mathcal{F}_f$ on a $\mathcal{C}_{r_f}(F) \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s_f}(F) \subseteq \mathcal{C}_s$, par l'union on obtient $\mathcal{C}_{r_f} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s_f} \subseteq \mathcal{C}_s$. Enfin, pour tout $F = \mathbb{S}\phi : (x, u) \in \mathcal{F}_f$, par construction on a $x \leqslant x \in \overline{\mathcal{C}_{r_f}(F)}$ et $\mathcal{C}_{r_f}(F) \subseteq \mathcal{C}_{r_f}$, et donc par la Proposition 3.2, on a $x \leqslant x \in \overline{\mathcal{C}_{r_f}}$. On montre de même que $u \lhd u \in \overline{\mathcal{C}_{s_f}}$. Ainsi, la propriété $(P_{CSS})$ est vérifiée et $\langle \mathcal{F}_f, \mathcal{C}_{r_f}, \mathcal{C}_{s_f} \rangle$ est un CTSS. □

Nous avons maintenant besoin de définir la notion de label inconsistant qui traduit, dans le monde des labels, la notion de ressources inconsistante.

**Définition 3.12** (Label inconsistant)**.** *Soit* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *un CTSS et x un label de ressources. x est dit* inconsistant *s'il existe deux labels de ressources y et z et un label d'états u tels que* $yz \leqslant x \in \overline{\mathcal{C}_r}$ *et* $\mathbb{T}\bot : (y, u) \in \mathcal{F}$.

*Un label est* consistant *s'il n'est pas inconsistant.*

On précise les propriétés suivantes des labels consistants :

**Proposition 3.4.** *Soit* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *un CTSS. On a les propriétés suivantes :*

1. *Si* $y \leqslant x \in \overline{\mathcal{C}_r}$ *et x est un label consistant, alors y est un label consistant.*

2. *Si $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ est un label consistant, x et y sont des labels consistants.*

*Démonstration.*

1. On raisonne par l'absurde. On suppose que $y$ est inconsistant. Il existe alors deux labels de ressources $z$ et $t$ et un label d'états $u$ tels que $zt \leqslant y \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (z,u) \in \mathcal{F}$. Comme on a aussi $y \leqslant x \in \overline{\mathcal{C}_r}$, par clôture par la règle $\langle t_r \rangle$, on obtient $zt \leqslant x \in \overline{\mathcal{C}_r}$. Par conséquent, $x$ est inconsistant, ce qui est contradictoire. Donc $y$ est consistant.
2. On raisonne par l'absurde et on suppose que $x$ est inconsistant . Il existe alors deux labels de ressources $z$ et $t$ et un label d'états $u$ tels que $zt \leqslant x \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (z,u) \in \mathcal{F}$. Comme $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r})$, par le Corollaire 3.1, on a aussi que $xy \leqslant xy \in \overline{\mathcal{C}_r}$. Par clôture de $\mathcal{C}_r$ par la règle $\langle c_r \rangle$, on a alors $zty \leqslant xy \in \overline{\mathcal{C}_r}$. On a donc $z(ty) \leqslant xy \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (z,u) \in \mathcal{F}$, donc $xy$ est inconsistant, ce qui est absurde. On a le même raisonnement si on suppose que $y$ est inconsistant (le problème est symétrique). Finalement, $x$ et $y$ sont consistants.

□

On peut maintenant écrire les règles de notre calcul des tableaux pour LTBI. Elles sont regroupées dans la Figure 3.4. La manière formelle d'utiliser ces règles sera données plus loin dans la définition d'un pseudo-tableau. La mention "$c_i$, $c_j$ et $l_i$ sont des nouvelles constantes" signifie que $c_i \neq c_j \in \gamma_r \backslash \mathcal{A}_r(\mathcal{C}_r)$ et $l_i \in L_s \backslash \mathcal{A}_s(\mathcal{C}_s)$.

On retrouve dans ces règles les règles additives et multiplicatives qui sont celles de BI. Les règles temporelles sont une traduction directe de la sémantique que nous avions donnée aux opérateurs dans la section 3.2.2.

Enfin, nous insistons sur la présence de deux nouvelles règles, que nous avons intitulées règles linéaires. En effet, si nous avons fait en sorte dans la Définition 3.10 que $\lhd$ traduise une relation ordonnée (transitive, antisymétrique et réflexive), cela ne suffit pas à traduire l'ordre de $\mathbb{N}$. Les entiers formant un ensemble totalement ordonné, il est nécessaire de s'assurer que cette propriété est aussi présente entre nos labels d'états. C'est à dire que, pour tout $u, v \in \mathcal{A}_s(\overline{\mathcal{C}_s})$ on doit avoir $u \lhd v \in \overline{\mathcal{C}_s}$ ou $u \lhd v \in \overline{\mathcal{C}_s}$.

C'est cet impératif que traduisent les règles $\langle \lhd \rangle$ et $\langle \blacktriangleleft \rangle$. Dans la règle $\langle \lhd \rangle$, chaque fois deux labels $v$ et $w$ introduits à partir d'un label $u$, on génère deux branches, l'une dans laquelle $v$ est avant $w$, l'une dans laquelle c'est l'inverse. La règle $\langle \blacktriangleleft \rangle$ opère de la même façon dans le cas où $v$ est le successeur de $u$. A ce moment là, dans la branche où $w$ est avant $v$, on a nécessairement $u \simeq w$ et dans l'autre on a $w$ qui est strictement supérieur à $u$.

Toutefois, l'ordre total entre label d'états garantis par ces nouvelles règles ne suffit toujours pas à avoir un isomorphisme avec $\mathbb{N}$ quant à l'ordre. En effet, les labels peuvent être denses pour l'ordre, ce qui peut empêcher d'associer chaque label d'états à un entier naturel tout en préservant l'ordre indiqué par les contraintes d'états. Nous reportons ce problème particulier à la section suivante et commençons par définir la notion de pseudo-tableau qui ne tient pas compte de cette contrainte.

**Définition 3.13** (LTBI-pseudo-tableau)**.** *Un* LTBI-pseudo-tableau *pour un CTSS fini $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$ est une liste de CTSS (appelés* branches*), construit de manière inductive en utilisant les règles suivantes :*

1. *La liste à une branche $[\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle]$ est un LTBI-pseudo-tableau pour $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$*

**Règles additives**

$$\frac{\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{T}\wedge\rangle \qquad \frac{\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{F}\phi : (x,u)\}, \emptyset, \emptyset\rangle \mid \langle \{\mathbb{F}\psi : (x,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{F}\wedge\rangle$$

$$\frac{\mathbb{T}\phi \vee \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (x,u)\}, \emptyset, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : (x,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{T}\vee\rangle \qquad \frac{\mathbb{F}\phi \vee \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{F}\phi : (x,u), \mathbb{F}\psi : (x,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{F}\vee\rangle$$

$$\frac{\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F} \text{ et } x \leqslant y \in \overline{C_r}}{\langle \{\mathbb{F}\phi : (y,u)\}, \emptyset, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : (y,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{T}\to\rangle \qquad \frac{\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (c_i,u), \mathbb{F}\psi : (c_i,u)\}, \{x \leqslant c_i\}, \emptyset\rangle} \langle \mathbb{F}\to\rangle$$

**Règle multiplicatives**

$$\frac{\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}}{\langle \emptyset, \{1_r \leqslant x\}, \emptyset\rangle} \langle \mathbb{T}\mathrm{I}\rangle$$

$$\frac{\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (c_i,u), \mathbb{T}\psi : (c_j,u)\}, \{c_i c_j \leqslant x\}, \emptyset\rangle} \langle \mathbb{T}*\rangle \qquad \frac{\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F} \text{ et } yz \leqslant x \in \overline{C_r}}{\langle \{\mathbb{F}\phi : (y,u)\}, \emptyset, \emptyset\rangle \mid \langle \{\mathbb{F}\psi : (z,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{F}*\rangle$$

$$\frac{\mathbb{T}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F} \text{ et } xy \leqslant xy \in \overline{C_r}}{\langle \{\mathbb{F}\phi : (y,u)\}, \emptyset, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : (xy,u)\}, \emptyset, \emptyset\rangle} \langle \mathbb{T}\mathrel{-\!\!*}\rangle \qquad \frac{\mathbb{F}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (c_i,u), \mathbb{F}\psi : (xc_i,u)\}, \{xc_i \leqslant xc_i\}, \emptyset\rangle} \langle \mathbb{F}\mathrel{-\!\!*}\rangle$$

**Règles temporelles**

$$\frac{\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}}{\langle \mathbb{T}\phi : (x,l_i), \emptyset, \{u \lhd l_i\}\rangle} \langle \mathbb{T}\Diamond\rangle \qquad \frac{\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F} \text{ et } u \lhd v \in \overline{C_s}}{\langle \{\mathbb{F}\phi : (x,v)\}, \emptyset, \emptyset\rangle} \langle \mathbb{F}\Diamond\rangle$$

$$\frac{\mathbb{T}\Box\phi : (x,u) \in \mathcal{F} \text{ et } u \lhd v \in \overline{C_s}}{\langle \{\mathbb{T}\phi : (x,v)\}, \emptyset, \emptyset\rangle} \langle \mathbb{T}\Box\rangle \qquad \frac{\mathbb{F}\Box\phi : (x,u) \in \mathcal{F}}{\langle \mathbb{F}\phi : (x,l_i), \emptyset, \{u \lhd l_i\}\rangle} \langle \mathbb{F}\Box\rangle$$

$$\frac{\mathbb{T}\bigcirc\phi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{T}\phi : (x,l_i)\}, \emptyset, \{u \blacktriangleleft l_i\}\rangle} \langle \mathbb{T}\bigcirc\rangle \qquad \frac{\mathbb{F}\bigcirc\phi : (x,u) \in \mathcal{F}}{\langle \{\mathbb{F}\phi : (x,l_i)\}, \emptyset, \{u \blacktriangleleft l_i\}\rangle} \langle \mathbb{F}\bigcirc\rangle$$

**Règles linéaires**

$$\frac{u \lhd v \in \overline{C_s} \text{ et } u \lhd w \in \overline{C_s}}{\langle \{\emptyset, \emptyset, \{v \lhd w\}\rangle \mid \langle \{\emptyset, \emptyset, \{w \lhd v\}\rangle} \langle \lhd\rangle \qquad \frac{u \blacktriangleleft v \in \overline{C_s} \text{ et } u \lhd w \in \overline{C_s}}{\langle \{\emptyset, \emptyset, \{v \lhd w, u \not\simeq w\}\rangle \mid \langle \{\emptyset, \emptyset, \{u \simeq w\}\rangle} \langle \blacktriangleleft\rangle$$

Note : $c_i$, $c_j$ et $l_i$ sont des nouvelles constantes.

FIGURE 3.4 – Règles de pseudo-tableaux / tableaux pour LTBI

2. *Si la liste* $\mathcal{T}_m \oplus [\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle] \oplus \mathcal{T}_n$ *est un LTBI-pseudo-tableau pour* $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$ *et*

$$\frac{cond(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle)}{\langle \mathcal{F}_1, \mathcal{C}_{r_1}, \mathcal{C}_{s_1} \rangle \mid ... \mid \langle \mathcal{F}_k, \mathcal{C}_{r_k}, \mathcal{C}_{s_k} \rangle}$$

*est une instance de la Figure 3.4 pour laquelle la condition* $cond(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle)$ *est remplie, alors la liste*

$$\mathcal{T}_m \oplus [\langle \mathcal{F} \cup \mathcal{F}_1, \mathcal{C}_r \cup \mathcal{C}_{r_1}, \mathcal{C}_s \cup \mathcal{C}_{s_1} \rangle; ...; \langle \mathcal{F} \cup \mathcal{F}_k, \mathcal{C}_r \cup \mathcal{C}_{r_k}, \mathcal{C}_s \cup \mathcal{C}_{s_k} \rangle] \oplus \mathcal{T}_n$$

*est un LTBI-pseudo-tableau pour* $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$.

*Un LTBI-pseudo-tableau pour une formule* $\phi$ *est un LTBI-pseudo-tableau pour :*

$$\langle \{\mathbb{F}\phi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\} \rangle$$

On remarque que $(P_{CSS})$ est bien vérifiée pour $\langle \{\mathbb{F}\phi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\} \rangle$. En outre, on peut vérifier que chaque application d'une règle de la Figure 3.4 conserve la propriété $(P_{CSS})$.

En observant les règles, on peut distinguer trois sortes de règles :

1. Les règles $\langle \mathbb{T}\mathrm{I} \rangle$, $\langle \mathbb{F} \rightarrow \rangle$, $\langle \mathbb{T} * \rangle$, $\langle \mathbb{F} \mathrel{-\!\!*} \rangle$, $\langle \mathbb{T}\Diamond \rangle$, $\langle \mathbb{F}\Box \rangle$, $\langle \mathbb{T}\mathrm{O} \rangle$ et $\langle \mathbb{F}\mathrm{O} \rangle$ introduisent de nouvelles contraintes et de nouvelles constantes ($c_i$, $c_j$ et $l_i$), sauf la règle $\langle \mathbb{T}\mathrm{I} \rangle$ qui n'introduit qu'une contrainte.

   Par exemple, illustrons avec une application de la règle $\langle \mathbb{T}\mathrm{O} \rangle$. Si l'on souhaite appliquer cette règle sur la formule étiquetée $\mathbb{T}\mathrm{O}\phi : (c_1, l_3)$ appartenant à un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$, on génère une nouvelle constante. Supposons que $l_7 \notin \mathcal{A}_s(\mathcal{C}_s)$. On obtient alors le CTSS $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1, l_7)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{l_3 \blacktriangleleft l_7\} \rangle$, et on a introduit une nouvelle contrainte $l_3 \blacktriangleleft l_7$ dans notre branche

2. Les règles $\langle \lhd \rangle$ et $\langle \blacktriangleleft \rangle$ ont des conditions qui imposent de sélectionner des contraintes de $\overline{\mathcal{C}_s}$ mais introduisent également de nouvelles contraintes.

   Par exemple, si nous reprenons l'exemple précédent où nous avons obtenu le CTSS $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1, l_7)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{l_3 \blacktriangleleft l_7\} \rangle$, si l'on suppose que $l_3 \lhd l_5 \in \mathcal{C}_s$, on peut appliquer la règle $\langle \blacktriangleleft \rangle$. On génère alors deux branches, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1, l_7)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{l_3 \blacktriangleleft l_7\} \cup \{l_7 \lhd l_5, l_3 \not\simeq l_5\} \rangle$ et $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1, l_7)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{l_3 \blacktriangleleft l_7\} \cup \{l_3 \simeq l_5\} \rangle$, chacune contenant de nouvelles contraintes, $\{l_7 \lhd l_5, l_3 \not\simeq l_5\}$ d'une part et $\{l_3 \simeq l_5\}$ d'autre part.

3. Les règles $\langle \mathbb{T} \rightarrow \rangle$, $\langle \mathbb{F} * \rangle$, $\langle \mathbb{T} \mathrel{-\!\!*} \rangle$, $\langle \mathbb{F}\Diamond \rangle$ et $\langle \mathbb{T}\Box \rangle$ ont des conditions qui imposent de choisir des contraintes déjà présentes dans $\overline{\mathcal{C}_r}$ ou $\overline{\mathcal{C}_s}$.

   Illustrons avec la règle $\langle \mathbb{T}\Box \rangle$. Si l'on souhaite appliquer cette règle sur la formule étiquetée $\mathbb{T}\Box\phi : (c_1, l_2)$ appartenant à un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$, on doit sélectionner un label $l \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $l_2 \lhd l \in \overline{\mathcal{C}_s}$. Supposons qu'il y ait un label $l_4$ tel que $l_2 \lhd l_4 \in \overline{\mathcal{C}_s}$. On peut alors appliquer la règle et on génère une nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1, l_4)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

Pour établir une preuve d'une formule, il convient d'observer un ordre dans l'application des règles. Les règles de la catégorie 1 doivent être appliquées d'abord, puis celle de la catégorie

2 et finalement celles de la catégorie 3. Ainsi, les contraintes générées par l'application des règles 1 peuvent être utilisées pour appliquer les règles 2, et celles générées par les règles 1 et 2 peuvent être utilisées pour appliquer les règles 3.

Maintenant que nous avons défini formellement comment un pseudo-tableau était construit, regardons plus en détail ce qu'il manque aux pseudo-tableaux pour exactement retranscrire la sémantique de LTBI.

### 3.4.3 CTSS normaux et tableaux pour LTBI

Comme nous venons de l'évoquer, la définition de pseudo-tableaux (Définition 3.13) ne parvient pas à traduire correctement la sémantique de LTBI. L'introduction des règles $\langle\lhd\rangle$ et $\langle\blacktriangleleft\rangle$ est un pas dans ce sens. En effet, comme l'indique le Lemme 3.9, dans une branche saturée, on a un ordre total entre tous les labels d'état d'un tableau pour la relation $\lhd$. Toutefois ce n'est pas suffisant. La sémantique de LTBI utilise des sous-ensembles de $\mathbb{N}$ comme instants temporels. Quant à nos labels, puisqu'ils sont ordonnés totalement, ils sont isomorphiques pour l'ordre à un sous-ensemble de $\mathbb{Q}$ (d'après un théorème de Cantor dans [22]).

La nuance est dans la densité des labels pour l'ordre. S'ils sont isomorphiques à $\mathbb{Q}$, les labels peuvent être denses pour l'ordre (c'est à dire qu'entre deux labels, on peut toujours en trouver un troisième). Bien évidemment, ce cas ne peut pas apparaître si l'ensemble des labels d'états est fini. Dans le cas où on introduit de nouveaux labels d'état indéfiniment après le label le plus grand, on a bien une infinité de labels, mais on parvient à maintenir l'isomorphisme avec $\mathbb{N}$.

Le cas problématique est celui où l'on introduit des labels indéfiniment entre deux labels préexistants. Ce type de situation est bien entendu à éviter, puisque la sémantique de LTBI impose des labels énumérés. Pour le formaliser, on pose la définition suivante.

**Définition 3.14** (CTSS normal)**.** *Un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est dit* normal *si pour tout $l_i, l_j \in \mathcal{A}_s(\mathcal{C}_s)$, $\{l_k \in \mathcal{A}_s(\mathcal{C}_s) | l_i \lhd l_k \in \overline{\mathcal{C}_s}$ et $l_k \lhd l_j \in \overline{\mathcal{C}_s}\}$ est un ensemble fini.*

Le but est donc de s'assurer que tous les CTSS de nos tableaux sont bien normaux. Toutefois, des recherches avancées n'ont pas permis de trouver une solution simple à ce problème. Nous devons donc nous contenter de la conjecture suivante.

**Conjecture 3.1.** *Il existe des conditions dans l'application des règles de la Figure 3.4 de sorte que tout CTSS obtenu par une règle soit un CTSS normal.*

Avant de poursuivre, on souhaite discuter des raisons pour lesquelles cette conjecture n'est pas prouvée. En premier lieu, on veut examiner de plus près les cas problématiques qui pourraient infirmer la Conjecture 3.1.

Pour comprendre comment un tel cas peut se produire, il faut réfléchir sur la structure même des tableaux. Toutes les règles de la Figure 3.4 respectent la règle de la sous-formule, c'est à dire qu'aucune des formules étiquetées produites par une règle n'introduit de nouvelles formules de LTBI. La décomposition des formules ne peut donc pas conduire à des branches infinies (puisque la formule d'origine du tableau est finie).

En revanche, certaines règles introduisent de nouveaux labels et de nouvelles contraintes. Cela pourrait ne pas être un problème, puisque ce qui détermine l'application des règles est avant tout l'opérateur principal de la formule. Toutefois, rien dans la définition des tableaux

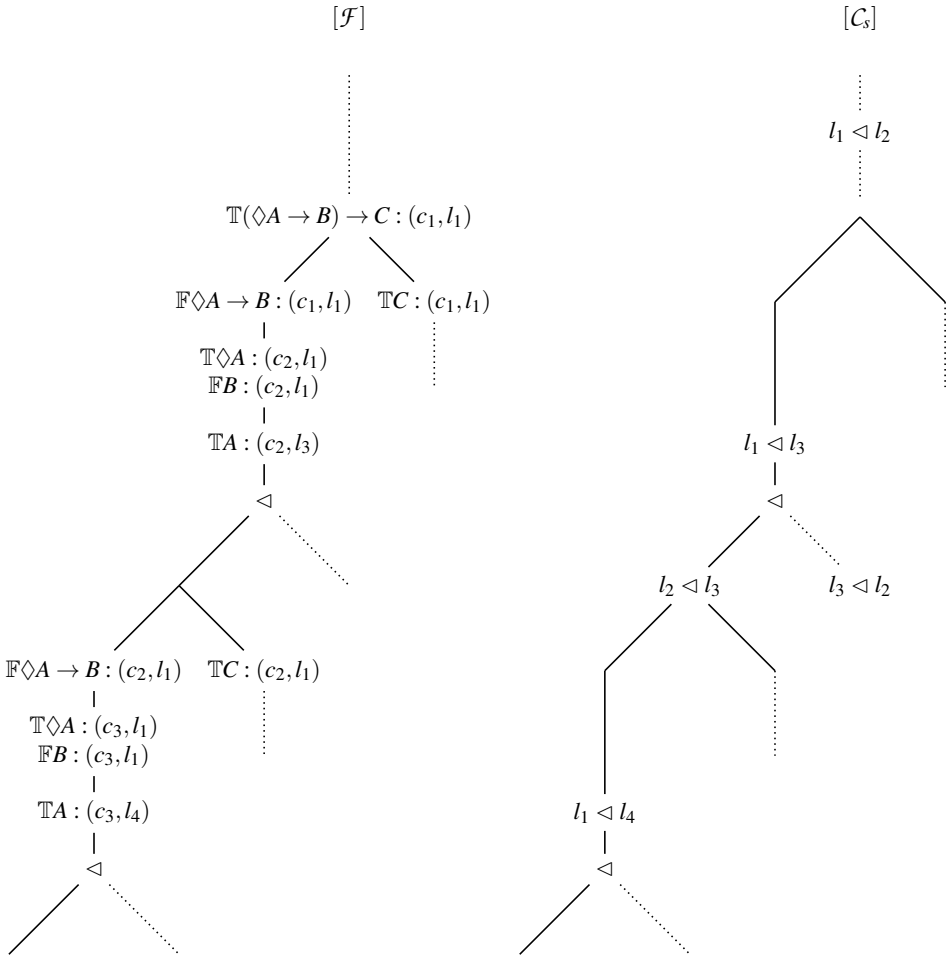



FIGURE 3.5 – Développement d'une branche infinie violant la Conjecture 3.1

n'interdit d'appliquer plusieurs fois une même règle, tant que la condition de sa prémisse peut toujours être validée.

C'est une opération courante dans le calcul, et parfois même essentielle, que d'utiliser une règle avec une condition sur les contraintes de labels plusieurs fois, avec plusieurs contraintes différentes. Si l'utilisation d'une nouvelle contrainte entraîne la création d'une autre, on peut alors générer une branche infinie en utilisant systématiquement la contraintes venant d'être créée, ce qui en crée à son tour une nouvelle.

Ainsi, toute partie du tableau qui successivement consomme une contrainte (de ressources ou d'états) et en crée une de même nature peut générer une boucle infinie. Si cette boucle contient également la création d'un label temporel, alors on peut se retrouver dans le cas précis que la Conjecture 3.1 cherche à éviter.

La Figure 3.5 donne l'exemple d'un tel cas. On se place dans le cas d'un tableau où un CTSS contient la formule $\mathbb{F}(\Diamond A \to B) \to C : (c_1, l_1)$ ainsi que la contrainte d'états $l_1 \lhd l_2$. On peut alors utiliser la règle $\langle \mathbb{F} \to \rangle$ avec la contrainte $c_1 \leqslant c_1$ (qui est dans la clôture par définition

d'un CTSS). Cela génère un nouveau label de ressources $c_2$ tel que $c_1 \leqslant c_2$ ainsi qu'un label d'états $l_3$ tel que $l_1 \lhd l_3$.

Une application de la règle $\lhd$ crée une branche (en l'occurrence celle de gauche) où l'on a $l_1 \lhd l_3 \lhd l_2$. Cette branche ne peut être close, et on peut alors considérer une deuxième fois la formule $\mathbb{F}(\Diamond A \to B) \to C : (c_1, l_1)$, cette fois avec la contrainte $c_1 \leqslant c_2$, ce qui crée un label $c_3$ tel que $c_2 \leqslant c_3$ (et donc par clôture transitive, $c_1 \leqslant c_3$) et un label $l_4$ tel que $l_1 \lhd l_4$.

Encore une fois, des applications de $\lhd$ permettent d'obtenir au moins une branche avec $l_1 \lhd l_3 \lhd l_4 \lhd l_2$, et la nouvelle constante $c_3$ permet de réitérer le processus. On voit donc comment une telle branche permet de construire une branche où les labels d'états forment le schéma $l_1 \lhd l_3 \lhd l_4 \lhd \ldots \lhd l_n \lhd \ldots \lhd l_2$, qui est précisément celui qu'on cherche à éviter.

Une chose que l'on remarque en étudiant cet exemple est que les nouveaux labels introduits (et qui créent la branche infinie) ne sont pas réellement significatifs dans notre tableau. Essentiellement, lorsqu'on crée une boucle infinie de la sorte, on réintroduit systématiquement un label qui remplit la même fonction que ses prédécesseurs. Ainsi, dans l'exemple exposé ci-dessus, les labels $c_2, c_3, \ldots$ ont la même signification et sont en réalité redondants. De même, les labels $l_3, l_,4, \ldots$ sont aussi redondants.

Dès lors, il serait aisé de satisfaire la Conjecture 3.1 si l'on parvient à caractériser formellement ces labels redondants : si l'on empêche d'appliquer toute règle qui produit des labels redondants, alors une branche infinie n'est plus constructible et tous les CSS sont normaux. Une telle démarche a déjà été effectuée pour les tableaux de BI dans [46].

La réelle difficulté dans notre cas est la caractérisation des labels redondants. En effet, dans le cadre de LTBI, la présence des labels de ressources et des labels d'états en même temps rend délicate la détection de redondance : non seulement un label d'états redondant peut être introduit à partir d'un autre label d'états, mais aussi à partir d'un label de ressources, lui même potentiellement redondant. Par exemple, dans notre illustration précédente, le couple $(c_3, l_4)$ a le même rôle que $(c_2, l_3)$. La complexité de caractérisation des labels redondants vient donc du fait les deux familles de labels (états et ressources) sont introduits orthogonalement et que la redondance doit donc porter sur les deux labels simultanément.

De futurs travaux permettront de caractériser formellement de telles redondances et supprimer ainsi les chaînes infinies de labels d'états. En attendant, on peut souligner que tous les pseudo-tableaux où les labels d'états sont soit en nombres finis, soit infinis mais non bornés pour l'ordre vérifient naturellement la Conjecture 3.1. Cela implique que la majorité des cas d'application de notre calcul des tableaux n'est pas concerné par ce problème. En outre, même pour les tableaux concernés, l'absence de cette conjecture n'impacte pas sur la correction de notre calcul, seulement sur la méthode d'extraction de contre-modèles, et par suite sur la complétude du calcul (voir la section 3.5.2).

Quoi qu'il en soit, avec cet outil hypothétique, nous pouvons définir proprement nos tableaux :

**Définition 3.15** (LTBI-tableau)**.** *Un* LTBI-tableau *pour un CTSS fini $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$ est un LTBI-pseudo-tableau pour $\langle \mathcal{F}_0, \mathcal{C}_{r_0}, \mathcal{C}_{s_0} \rangle$ construit de telle sorte que toute application de règle de la Figure 3.12 respecte la Conjecture 3.1.*

*Un LTBI-tableau pour une formule $\phi$ est un LTBI-pseudo-tableau pour $\phi$, c'est à dire un LTBI-tableau pour :*

$$\langle \{\mathbb{F}\phi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\} \rangle$$

Il nous reste à énoncer les conditions de clôture de nos tableaux.

**Définition 3.16** (Conditions de clôture)**.** *Un CTSS* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *est* clos *si l'une des conditions suivantes est vraie :*

1. $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, $\mathbb{F}\phi : (y,v) \in \mathcal{F}$ *et* $x \leqslant y \in \overline{\mathcal{C}_r}$ *et* $u \simeq v \in \overline{\mathcal{C}_s}$
2. $\mathbb{F}\mathrm{I} : (x,u) \in \mathcal{F}$ *et* $1_r \leqslant x \in \overline{\mathcal{C}_r}$
3. $\mathbb{F}\top : (x,u) \in \mathcal{F}$
4. $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ *et* $x$ *est inconsistant*
5. $u \simeq v \in \overline{\mathcal{C}_s}$ *et* $u \not\simeq v \in \overline{\mathcal{C}_s}$

*Un CTSS est* ouvert *s'il n'est pas clos. Un tableau est clos si toutes ses branches sont closes.*

La première condition correspond au cas le plus classique : la présence dans une branche d'une formule à la fois étiquetée vraie ou fausse (d'où une contradiction). La deuxième condition est la même mais dans le cas particulier de l'opérateur I. Les conditions 3 et 4 gèrent les opérateurs $\top$ et $\bot$. Enfin, la condition 5 traitent du cas contradictoire où deux labels d'états sont à la fois équivalents et non équivalents.

On peut simplifier la première condition dans certains cas particuliers récurrents.

**Lemme 3.5.** *Soit* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *un CTSS. Si l'une des conditions suivantes est remplie, alors le CTSS est clos :*

1. $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ *et* $x \leqslant y \in \overline{\mathcal{C}_r}$
2. $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, $\mathbb{F}\phi : (x,v) \in \mathcal{F}$ *et* $u \simeq v \in \overline{\mathcal{C}_s}$
3. $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, $\mathbb{F}\phi : (x,u) \in \mathcal{F}$

*Démonstration.*
Comme tout CTSS vérifie $(P_{css})$, on a $x \leqslant x \in \overline{\mathcal{C}_r}$ et $u \leqslant u \in \overline{\mathcal{C}_s}$. De plus, en appliquant la règle $\langle g_e \rangle$ de la Définition 3.10, on a $u \simeq u \in \overline{\mathcal{C}_s}$.

Par suite, en utilisant ces résultats avec la condition 1 de la Définition 3.16, on obtient les résultats indiqués.

□

Enfin, nous pouvons définir la notion de preuve.

**Définition 3.17** (LTBI-preuve)**.** *Une* LTBI-preuve *pour une formule* $\phi$ *est un LTBI-tableau défini pour* $\phi$ *qui est clos.*

### 3.4.4 Exemples de LTBI-tableaux

Nous allons maintenant donner deux exemples d'utilisation de notre calcul des tableaux.

**Une preuve par les tableaux de $\Diamond\phi \vDash \phi \vee \bigcirc\Diamond\phi$**

Nous avons établi plus tôt dans ce chapitre la propriété $\Diamond\phi \vDash \phi \vee \bigcirc\Diamond\phi$. On se propose ici d'en faire une preuve par le calcul des tableaux. Pour cela, on va dresser un tableau pour la formule $\Diamond A \multimap (A \vee \bigcirc\Diamond A)$ et montrer qu'il est clos.

Pour commencer, nous créons un tableau à partir de la formule. Comme indiqué dans la Définition 3.15, notre tableau initial est le CTSS suivant :

$$\langle \{\mathbb{F}\Diamond A \multimap (A \vee \bigcirc\Diamond A) : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\} \rangle$$

Pour plus de lisibilité, nous allons représenter nos CTSS non pas comme des listes, mais comme des graphes. Chaque formule et chaque contrainte est représentée sur un arbre et seules les parties qui diffèrent entre deux CTSS parents seront écrites sur des branches différentes de notre graphe.

Nous établissons donc les racines de trois graphes, un pour le formules et un pour chacun des ensembles de contraintes.

| $[\mathcal{F}]$ | $[\mathcal{C}_r]$ | $[\mathcal{C}_s]$ |
|---|---|---|
| $\mathbb{F}\Diamond A \multimap (A \vee \bigcirc\Diamond A) : (1_r, l_1)$ | $1_r \leqslant 1_r$ | $\{l_1\}$ |

Notre CTSS ne contient pour l'instant qu'une seule formule $\mathbb{F}\Diamond A \multimap (A \vee \bigcirc\Diamond A) : (1_r, l_1)$. Nous ne pouvons donc appliquer qu'une seule règle, $\langle \multimap \rangle$. On génère alors une nouvelle constante de label de ressources $c_1$, deux nouvelles formules étiquetées $\mathbb{F}\Diamond A : (c_1, l_1)$ et $\mathbb{T}A \vee \bigcirc\Diamond A : (c_1, l_1)$ ainsi qu'une nouvelle contrainte $1_r c_1 \leqslant 1_r c_1$. Comme $1_r$ est un élément neutre pour la composition des labels de ressources, on peut simplifier la contrainte en $c_1 \leqslant c_1$. Nous représentons tout cela dans nos graphes. En outre, nous marquons la formule que nous avons développé en premier par la marque $\surd_1$.

| $[\mathcal{F}]$ | $[\mathcal{C}_r]$ | $[\mathcal{C}_s]$ |
|---|---|---|
| $\surd_1\ \mathbb{F}\Diamond A \multimap (A \vee \bigcirc\Diamond A) : (1, l_1)$ | $1_r \leqslant 1_r$ | $l_1 \lhd l_1$ |
| $\mathbb{T}\Diamond A : (c_1, l_1)$ | $c_1 \leqslant c_1$ | |
| $\mathbb{F}A \vee \bigcirc\Diamond A : (c_1, l_1)$ | | |

On continue à appliquer les règles pour développer notre tableau. On finit par obtenir la Figure 3.6. Chaque nouvelle application de règle est marquée par un nouveau $\surd$.

Regardons de près chaque étape :

1. La première étape est celle que nous avons décrite plus haut.
2. Nous avons ici plusieurs règles applicables : $\langle \mathbb{T}\Diamond \rangle$ et $\langle \mathbb{F}\vee \rangle$. La règle $\langle \mathbb{F}\vee \rangle$ n'a pas de place particulière dans l'ordre de priorité que nous avons défini dans la section précédente, il n'y a donc pas de risque à la développer en premier. On développe $\vee$ ce qui nous donne deux nouvelles formules, avec les mêmes labels et les mêmes contraintes.

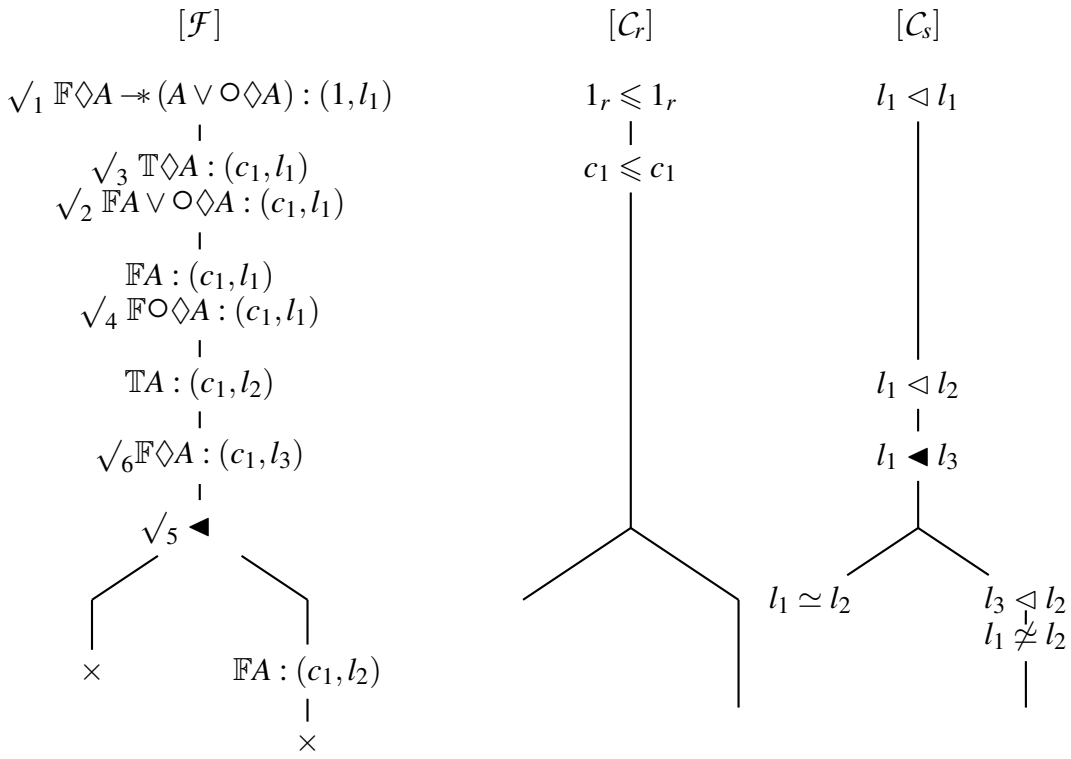



FIGURE 3.6 – Tableau pour $\Diamond A \multimap (A \vee \bigcirc\Diamond A)$

3. En prenant en compte les règles de priorité que nous avons énoncées, il nous faut d'abord utiliser la règle $\langle \mathbb{T}\Diamond \rangle$. Nous introduisons $l_2$ et la contrainte $l_1 \lhd l_2$.

4. On peut maintenant décomposer $\bigcirc$. Cela nous donne un nouveau label $l_3$ et une nouvelle contrainte $l_1 \blacktriangleleft l_3$.

5. On pourrait décomposer $\mathbb{F}\Diamond A$, mais nous remarquons que nous avons dans notre ensemble de contraintes d'états les éléments nécessaire pour appliquer $\langle \blacktriangleleft \rangle$, et selon notre ordre de priorité nous devons l'appliquer d'abord. Cela crée deux nouvelles branches avec des contraintes d'états distinctes.

   On peut remarquer que la branche de gauche est déjà close. En effet, on a $\mathbb{F}A : (c_1, l_1, \in)\mathcal{F}$, $\mathbb{T}A : (c_1, l_2, \in)\mathcal{F}$ et $l_1 \simeq l_2 \in \mathcal{C}_s$, et donc selon le Lemme 3.5, on peut clore la branche.

6. On applique finalement $\langle \mathbb{F}\Diamond \rangle$ dans la branche restante. Il nous faut un label $v$ tel que $l_3 \lhd v$, et nous pouvons choisir $v = l_2$ dans cette branche. Cela nous donne la clôture, puisque nous avons $\mathbb{F}A : (c_1, l_2)$ et $\mathbb{T}A : (c_1, l_2)$ dans cette branche.

Finalement, nous avons clos toutes les branches et notre tableau est clos.

Nous montrerons plus tard le lien entre validité et prouvabilité, mais nous pouvons déjà affirmer que l'existence d'une preuve démontre la validité de la formule. Alors le Théorème 3.1 nous indique que $\Diamond A \vDash (A \vee \bigcirc\Diamond A)$, ce qui est le résultat que nous avions déjà annoncé.

**Une illustration de la spécificité linéaire de LTBI**

Nous nous proposons maintenant d'illustrer comment LTBI se base sur un temps linéaire, ce qui la distingue fondamentalement de l'extension modale DBI [34], et comment cette spécificité est portée dans notre calcul des tableaux par les règles linéaires.

Nous utilisons pour cela une des formules traditionnellement utilisée (dans le cadre de LTL notamment) pour distinguer le temps linéaire du temps branchant, comme nous l'avons présenté dans la section 3.3.3.

$$\Diamond A \wedge \Diamond B \rightarrow \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$$

Cette formule postule que, si deux formules sont valides chacune à un instant futur différent, alors l'une doit arriver d'abord et l'autre ensuite. Ceci est valide si le temps est linéaire, mais pas s'il est branchant : on pourrait avoir deux futurs possibles chacun dans une branche temporelle différente.

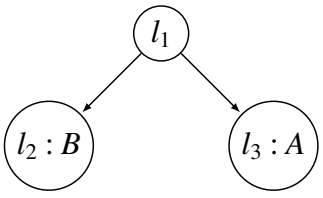
Nous pouvons tout d'abord tenter d'établir un arbre de preuve pour cette formule en nous interdisant d'utiliser les règles linéaires. On obtient alors le tableau de la Figure 3.7 (nous avons omis de représenter $\mathcal{C}_r$, son évolution étant sans rapport avec le problème étudié).

Comme le lecteur peut le constater, à l'étape 6 nous avons deux formules restantes, $\mathbb{T}A : (c_1, l_2)$ et $\mathbb{T}B : (c_1, l_3)$ que nous essayons de faire correspondre aux deux formules plus complexes $\mathbb{F}\Diamond(A \wedge \Diamond B) : (c_1, l_1)$ et $\mathbb{F}\Diamond(B \wedge \Diamond A) : (c_1, l_1)$. Comme $l_2$ et $l_3$ sont liées à $l_1$ dans $\mathcal{C}_s$, on doit choisir l'un ou l'autre pour développer les deux dernières formules.

Notre première approche est d'utiliser $l_2$ pour développer $\mathbb{F}\Diamond(A \wedge \Diamond B) : (c_1, l_1)$. Cela nous donne une clôture rapide avec $\mathbb{F}A : (c_1, l_2)$, mais nous laisse avec $\mathbb{F}\Diamond B : (c_1, l_2)$ que nous ne pouvons développer qu'avec $l_2$ puisqu'aucun autre label n'est lié à lui dans $\overline{\mathcal{C}_s}$. Cela nous amène à créer $\mathbb{F}B : (c_1, l_2)$ qui ne permet pas la clôture.

Le développement de $\mathbb{F}\Diamond(B \wedge \Diamond A) : (c_1, l_1)$ soulève un problème similaire. Choisir d'autres labels ne résout pas le problème et on ne parvient pas à clore le tableau [4].

Il semble donc que nous ne puissions pas établir de preuve de notre formule. On peut même considérer le même tableau dans la logique DBI (qui est similaire quant au traitement de $\Diamond$ si on exclut la linéarité) et en extraire un contre-modèle. On obtient alors le contre-modèle suivant :




Cela est bien le contre-exemple non linéaire que nous attendions : un temps branchant avec un futur qui vérifie $A$ et l'autre qui vérifie $B$.

Voyons maintenant ce qui se passe dans LTBI sans restriction. La Figure 3.8 représente la LTBI-preuve de notre formule. Encore une fois, $\mathcal{C}_r$ n'est pas représenté.

La preuve commence de la même manière que précédemment, mais comme nous générons $l_1 \lhd l_2$ et $l_1 \lhd l_3$, nous devons appliquer notre règle $\langle \lhd \rangle$ , et cela nous donne deux branches avec

4. Ce tableau n'est pas tout à fait complet, il faudrait réutiliser $\mathbb{F}\Diamond(A \wedge \Diamond B) : (c_1, l_1)$ et $\mathbb{F}\Diamond(B \wedge \Diamond A) : (c_1, l_1)$, en considérant $l_1$ comme une possible relation à $l_1$ lui-même, mais cela ne marche pas non plus de manière assez évidente.

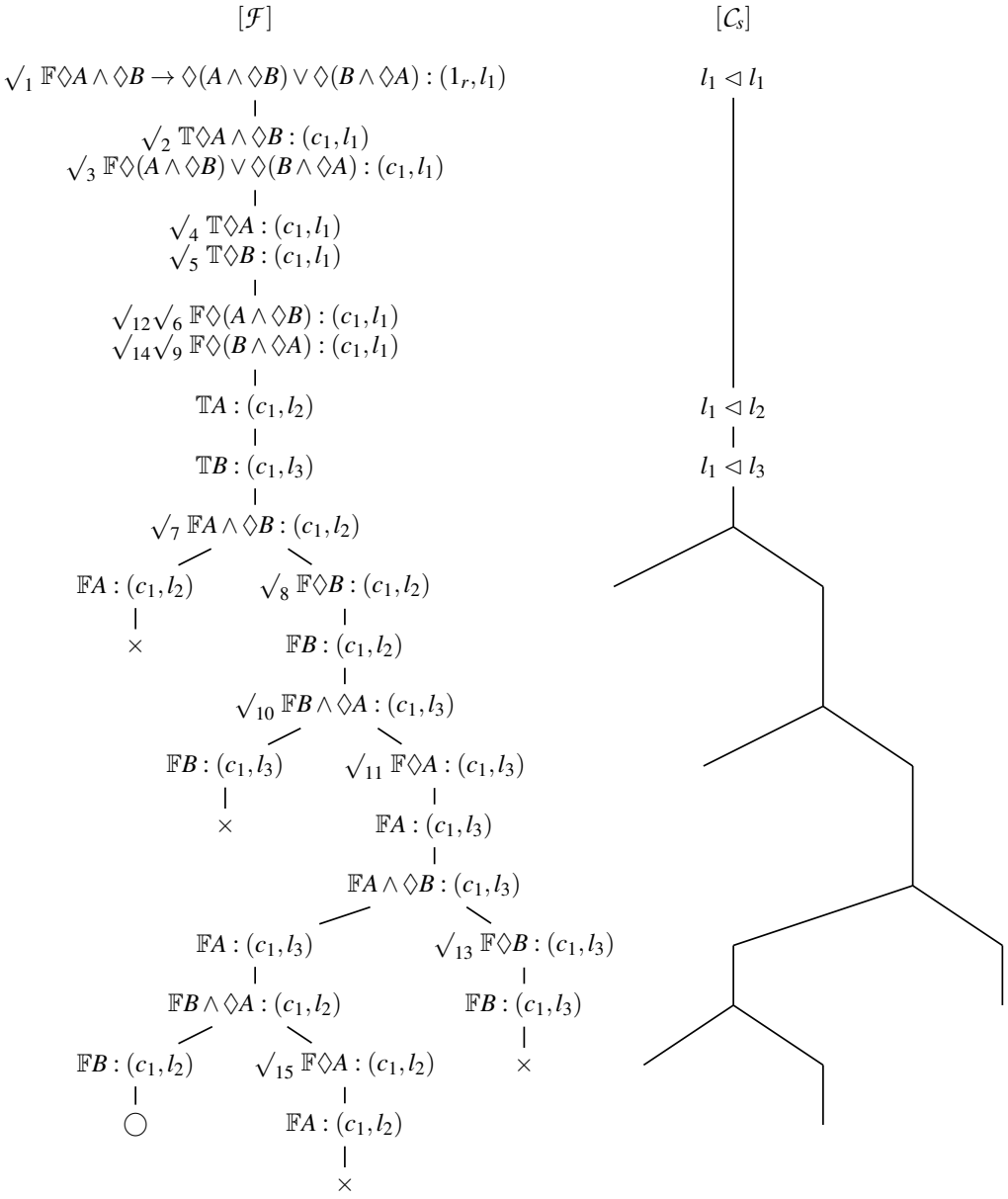



FIGURE 3.7 – LTBI-tableau pour $\Diamond A \wedge \Diamond B \to \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$ sans règles linéaires

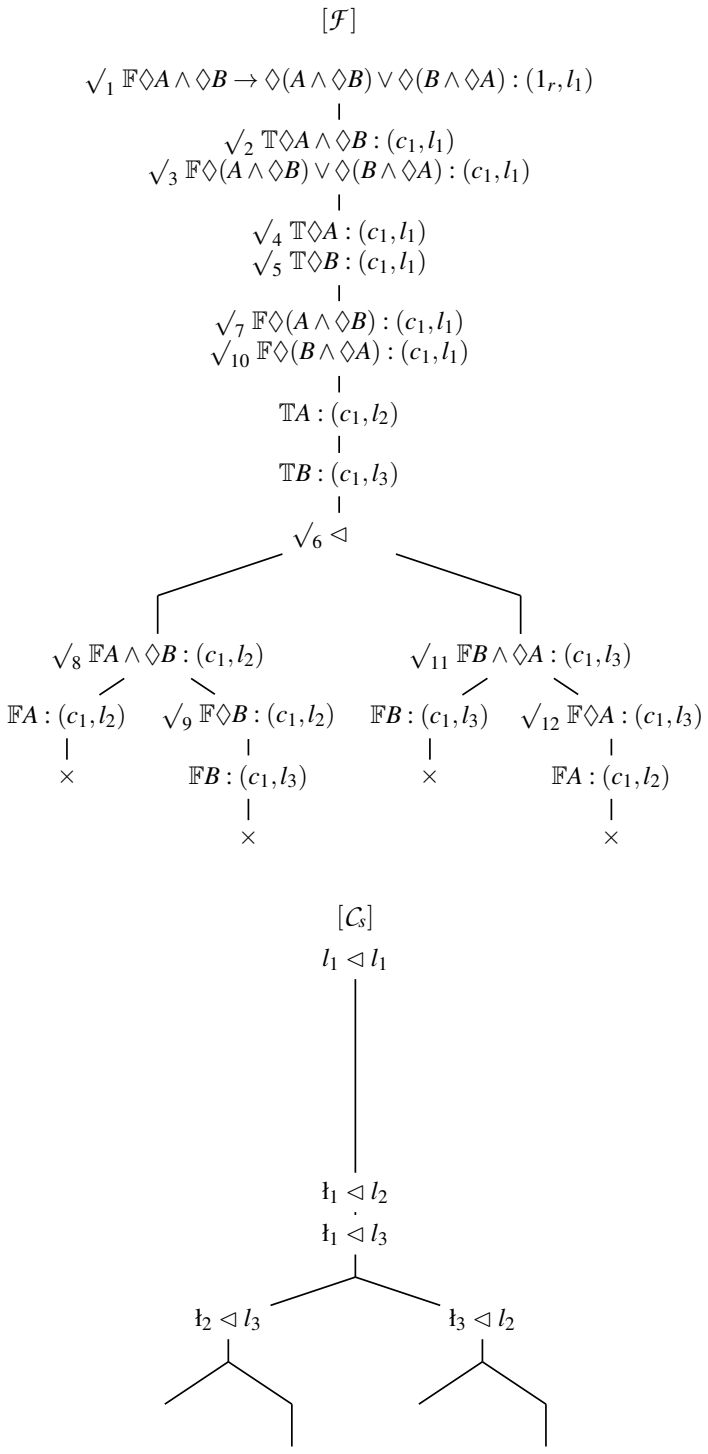



FIGURE 3.8 – LTBI-tableau pour $\Diamond A \wedge \Diamond B \to \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$

respectivement $l_2 \lhd l_3$ et $l_3 \lhd l_2$. Ces deux contraintes supplémentaires vont nous permettre de résoudre les problèmes que nous avions dans la version précédente. En effet, en développant $\mathbb{F}\Diamond(A \wedge \Diamond B) : (c_1, l_1)$, on peut toujours avoir une clôture facile avec $\mathbb{F}A : (c_1, l_2)$, mais nous pouvons aussi cette fois développer $\mathbb{F}\Diamond B : (c_1, l_2)$ en utilisant $l_3$ au lieu de $l_2$,ce qui nous donne la clôture, et de même pour l'autre branche et $\mathbb{F}\Diamond(B \wedge \Diamond A) : (c_1, l_1)$.

Finalement, nous pouvons clore toutes les branches, ce qui nous donne le résultat que, dans ce cas, cette formule a une preuve. Cet exemple est une bonne illustration de comment l'utilisation de $\langle\lhd\rangle$ garantit un ordre total entre les labels d'états, ce qui permet d'assurer la linéarité de nos modèles. En effet, c'est l'application de la règle $\langle\lhd\rangle$ qui permet dans notre exemple de générer une contrainte entre $l_2$ et $l_3$ et ainsi ordonne totalement les labels d'états.

## 3.5 Propriétés du calcul des tableaux pour LTBI

Nous allons maintenant présenter les propriétés de notre calcul, à savoir sa correction et sa complétude vis-à-vis de la sémantique de LTBI. Nous présenterons aussi une méthode d'extraction de contre-modèles à partir d'un tableau non clos et saturé, ce qui sera une base pour notre preuve de complétude. Ces preuves et méthodes s'appuient sur les travaux similaires sur DBI dans [34].

On verra comment les règles temporelles de la Figure 3.4 ainsi que la Conjecture 3.1 permettent d'obtenir des contre-modèles en accord avec la sémantique de LTBI, avec des états isomorphes à $\mathbb{N}$ pour l'ordre. Passée cette étape, les preuves de correction et de complétude s'adaptent directement depuis leur développement pour DBI.

### 3.5.1 Correction du calcul

La preuve de correction de notre calcul est basée sur la notion de *réalisabilité* d'un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$, ce qui signifie qu'il existe un modèle linéaire de ressources $\mathcal{K}$ et des plongements de l'ensemble des labels de ressources vers l'ensemble des ressources ($\lfloor\cdot\rfloor$) et de l'ensemble des labels d'états vers l'ensemble des états ($\lceil\cdot\rceil$) de $\mathcal{K}$ tels que, si $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ alors $(\lfloor x\rfloor, \lceil u\rceil) \vDash_{\mathcal{K}} \phi$ et si $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ alors $(\lfloor x\rfloor, \lceil u\rceil) \nvDash_{\mathcal{K}} \phi$. Ces plongements permettent de construire un raisonnement par l'absurde qui montre que chaque formule ayant une preuve doit être valide.

Commençons par définir formellement la notion de réalisation.

**Définition 3.18** (Réalisation)**.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS. Une* réalisation *de ce CTSS est un triplet $(\mathcal{K}, \lfloor.\rfloor, \lceil.\rceil)$ tel que $\mathcal{K} = (\mathcal{M}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est un modèle linéaire de ressources avec $\mathcal{M} = (R, \bullet, e, \pi, \sqsubseteq, S)$, $\lfloor.\rfloor : \mathcal{D}_r(\overline{\mathcal{C}_r}) \to R$ et $\lceil.\rceil : \mathcal{A}_s(\mathcal{C}_s) \to S$, tels que :*

- $\lfloor 1_r \rfloor = e$
- $\lfloor x \circ y \rfloor = \lfloor x \rfloor \bullet \lfloor y \rfloor$
- *Si* $\mathbb{T}\phi : (x,u) \in \mathcal{F}$*, alors* $(\lfloor x\rfloor, \lceil u\rceil) \vDash_{\mathcal{K}} \phi$
- *Si* $\mathbb{F}\phi : (x,u) \in \mathcal{F}$*, alors* $(\lfloor x\rfloor, \lceil u\rceil) \nvDash_{\mathcal{K}} \phi$
- *Si* $x \leqslant y \in \mathcal{C}_r$*, alors* $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$
- *Si* $u \lhd v \in \mathcal{C}_s$*, alors* $\lceil u \rceil \leq \lceil v \rceil$
- *Si* $u \blacktriangleleft v \in \mathcal{C}_s$*, alors* $\lceil v \rceil = \lceil u \rceil + 1$

- *Si* $u \simeq v \in \mathcal{C}_s$, *alors* $\lceil v \rceil = \lceil u \rceil$
- *Si* $u \not\simeq v \in \mathcal{C}_s$, *alors* $\lceil v \rceil \neq \lceil u \rceil$

*On dit qu'un CTSS (ou branche) est* réalisable *s'il en existe une réalisation.*
*On dit qu'un tableau est* réalisable *si au moins une de ses branches est réalisable.*

Notons que la notion de réalisabilité s'étend aux clôture des ensembles de contrainte, comme le spécifie le lemme suivant.

**Lemme 3.6.** *Soit* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *un CTSS et* $\mathfrak{R} = (\mathcal{K}, \lfloor . \rfloor, \lceil . \rceil)$ *une réalisation de ce CTSS.* $\mathfrak{R}$ *est aussi une réalisation de* $\langle \mathcal{F}, \overline{\mathcal{C}_r}, \overline{\mathcal{C}_s} \rangle$. *En d'autres mots :*

- *Pour tout* $x \leqslant y \in \overline{\mathcal{C}_r}$, $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$
- *Pour tout* $u \lhd v \in \overline{\mathcal{C}_s}$, $\lceil u \rceil \leq \lceil v \rceil$
- *Pour tout* $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$, $\lceil v \rceil = \lceil u \rceil + 1$
- *Pour tout* $u \simeq v \in \overline{\mathcal{C}_s}$, $\lceil v \rceil = \lceil u \rceil$
- *Pour tout* $u \not\simeq v \in \overline{\mathcal{C}_s}$, $\lceil v \rceil \neq \lceil u \rceil$

*Démonstration.*
On prouve le lemme pour les contraintes de ressources, la preuve pour les contraintes d'états est similaire. Pour tout $x \leqslant y \in \overline{\mathcal{C}_r}$, on montre que $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$ par récurrence sur la taille $n$ de l'arbre de preuve de $x \leqslant y$.

- Cas de base ($n = 0$)
  Dans ce cas $x \leqslant y \in \overline{\mathcal{C}_r}$ parce que $x \leqslant y \in \mathcal{C}_r$. Donc, par définition de la réalisation on a $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$.

- Hérédité
  On suppose que le lemme est vrai pour toutes les contraintes de ressources ayant un arbre de preuve de taille au plus égale à $n$ (HR). On prouve le lemme pour une contrainte d'états de $\overline{\mathcal{C}_r}$ ayant un arbre de preuve de taille $n+1$. On discute selon la règle ayant permis d'obtenir cette contrainte :

  - $\langle t_r \rangle$ :
    L'arbre de preuve est de la forme $\dfrac{\dfrac{...}{x \leqslant y} \quad \dfrac{...}{y \leqslant z}}{x \leqslant z}\ \langle t_r \rangle$ . Par (HR), on a $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$ et $\lfloor y \rfloor \sqsubseteq \lfloor z \rfloor$. Par transitivité de $\sqsubseteq$, on a $\lfloor x \rfloor \sqsubseteq \lfloor z \rfloor$.
  - $\langle d_r \rangle$ :
    L'arbre de preuve est de la forme $\dfrac{\dfrac{...}{xy \leqslant xy}}{x \leqslant x}\ \langle d_r \rangle$ . Alors $x \leqslant x \in \overline{\mathcal{C}_r}$ et $x \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ par le Corollaire 3.1. Comme $\lfloor . \rfloor : \mathcal{D}_r(\overline{\mathcal{C}_r}) \to R$, $\lfloor x \rfloor$ est défini. Par conséquent $\lfloor x \rfloor \sqsubseteq \lfloor x \rfloor$, par réflexivité de $\sqsubseteq$.
  - $\langle c_r \rangle$ :
    L'arbre de preuve est de la forme $\dfrac{\dfrac{...}{ky \leqslant ky} \quad \dfrac{...}{x \leqslant y}}{kx \leqslant ky}\ \langle c_r \rangle$ . Par (HR), $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$. Comme $kx \leqslant ky \in \overline{\mathcal{C}_r}$, on a $k, x, y, kx, ky \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ par le Corollaire 3.1. Donc $\lfloor k \rfloor$, $\lfloor x \rfloor$, $\lfloor y \rfloor$, $\lfloor kx \rfloor$ et $\lfloor ky \rfloor$ sont définis. Alors par compatibilité, $\lfloor k \rfloor \bullet \lfloor x \rfloor \sqsubseteq \lfloor k \rfloor \bullet \lfloor y \rfloor$. Ainsi on a, $\lfloor kx \rfloor \sqsubseteq \lfloor ky \rfloor$ par définition de la réalisation.

- Les autres cas sont similaires.

□

On va à présent énoncer les deux lemmes clés de notre preuve par l'absurde de la correction. Le premier énonce que les tableaux clos ne sont pas réalisables, l'autre que la réalisabilité est conservée par extension d'un tableau par une règle de la Figure 3.4.

**Lemme 3.7.** *Les LTBI-tableaux clos ne sont pas réalisables.*

*Démonstration.* Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS. On suppose qu'il existe une réalisation $(\mathcal{K}, \lfloor . \rfloor, \lceil . \rceil)$ de ce CTSS. On suppose que ce CTSS est clos. Il y a cinq cas :

- $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, $\mathbb{F}\phi : (y,v) \in \mathcal{F}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$ et $u \simeq v \in \overline{\mathcal{C}_s}$. Alors par la Définition 3.18, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ et $(\lfloor y \rfloor, \lceil v \rceil) \nvDash_{\mathcal{K}} \phi$. En outre, par le Lemme 3.6, $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$, donc, par monotonicité (Lemme 3.1), $(\lfloor y \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$. Toujours par le Lemme 3.6, $\lceil u \rceil = \lceil v \rceil$, donc $(\lfloor y \rfloor, \lceil v \rceil) \vDash_{\mathcal{K}} \phi$, ce qui est contradictoire.
- $\mathbb{F}\mathrm{I} : (x,u) \in \mathcal{F}$ et $1_r \leqslant x \in \overline{\mathcal{C}_r}$. Alors $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \mathrm{I}$. Donc par définition de la relation de forcing, $e \not\sqsubseteq \lfloor x \rfloor$. Par le Lemme 3.6, comme $1_r \leqslant x \in \overline{\mathcal{C}_r}$, on a $e \sqsubseteq \lfloor x \rfloor$, ce qui est contradictoire.
- $\mathbb{F}\top : (x,u) \in \mathcal{F}$. Alors $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \top$ ce qui est contradictoire.
- $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ et $x$ est inconsistant. Alors $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ et il existe deux labels $y$ et $z$ tels que $yz \leqslant x \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (y,v) \in \mathcal{F}$. Par le Lemme 3.6, $\lfloor y \rfloor \bullet \lfloor z \rfloor \sqsubseteq \lfloor x \rfloor$ et par la Définition 3.18, $(\lfloor y \rfloor, \lceil v \rceil) \vDash_{\mathcal{K}} \bot$. Donc $\pi \sqsubseteq \lfloor y \rfloor$. Par compatibilité, $\pi \sqsubseteq \lfloor y \rfloor \bullet \lfloor z \rfloor$ et par transitivité, $\pi \sqsubseteq \lfloor x \rfloor$. Donc, par le Lemme 3.2, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ ce qui est absurde.
- $u \simeq v \in \overline{\mathcal{C}_s}$ et $u \not\simeq v \in \overline{\mathcal{C}_s}$. Alors, par le Lemme 3.6, on doit avoir $\lceil u \rceil = \lceil v \rceil$ et $\lceil u \rceil \neq \lceil v \rceil$, ce qui est absurde.

Ainsi un tableau réalisable ne peut pas être clos, et donc par contraposée tout tableau clos n'est pas réalisable. □

**Lemme 3.8.** *Les règles d'extensions (Figure 3.4) préservent la réalisabilité.*

*Démonstration.*
On considère un tableau réalisable et on montre que son extension par une règle reste réalisable, en discutant selon la règle d'extension utilisée.

La preuve complète est en Annexe A. □

Enfin, on peut énoncer le théorème de correction.

**Théorème 3.2** (Correction)**.** *Soit $\phi$ une LTBI-formule. S'il existe une LTBI-preuve de $\phi$,alors $\phi$ est valide.*

*Démonstration.*
Soit $\mathcal{T}$ une LTBI-preuve de $\phi$. On suppose que $\phi$ n'est pas valide. Alors il existe un modèle linéaire de ressources $\mathcal{K}$ tel que $(e, s_0) \nvDash_{\mathcal{K}} \phi$. Si on considère $\lfloor 1_r \rfloor = e$ et $\lceil l_1 \rceil = s_0$ on obtient une réalisation $(\mathcal{K}, \lfloor . \rfloor, \lceil . \rceil)$ du CTSS initial $\langle \{\mathbb{F}\phi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1\} \rangle$. Par le Lemme 3.8, $\mathcal{T}$ est réalisable. Mais par le Lemme 3.7, cela est contradictoire car puisque $\mathcal{T}$ est une preuve, $\mathcal{T}$ est clos. Donc $\phi$ est valide. □

Nous allons à présent montrer que notre calcul est également complet, c'est à dire que toute formule valide peut être prouvée. Auparavant, nous devons pour cela formaliser une manière d'extraire des contre-modèles à partir de branches non closes.

### 3.5.2 Méthode d'extraction de contre-modèles

Dans cette section, nous adaptons la méthode d'extraction de contre-modèle de DBI [34]. L'idée est de transformer les contraintes d'états et de ressources en un monoïde de ressources linéaire à partir d'une branche $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ qui ne peut être close.

Tout d'abord, pour obtenir un modèle, nous devons construire un ensemble de ressources et un ensemble d'états, déduits des labels et contraintes de notre branche. En particulier, la Conjecture 3.1 nous permet de déterminer pour chaque label d'états un *label suivant*, ce qui permet la construction d'un sous-ensemble de $\mathbb{N}$ vérifiant les contraintes d'états.

Ensuite, pour obtenir un contre-modèle, notre modèle doit remplir deux propriétés :

- : si $\mathbb{T}\phi : (x,u) \in \mathcal{F}$, alors $x, u \vDash \phi$.
- : si $\mathbb{F}\phi : (x,u) \in \mathcal{F}$, alors $x, u \nvDash \phi$.

Pour remplir ces conditions, nous devons *saturer* les formules de notre CTSS pour obtenir un *CTSS de Hintikka*. Cela signifie, par exemple, que si $\mathbb{T}\Box\phi : (x,u) \in \mathcal{F}$, nous voulons avoir pour tout label $v$ tel que $u \lhd v \in \overline{\mathcal{C}_s}$, $\mathbb{T}\phi : (x,v) \in \mathcal{F}$ doit être vérifié. C'est pour cela qu'on parle de saturation : on fait en sorte qu'aucune règle ne puisse plus être appliquée sur la formule en question. Une branche sera saturée lorsque toutes les formules seront saturées.

Par rapport aux branches de Hintikka de DBI, on doit rajouter les conditions pour saturer la branche par rapport aux règles $\langle \mathbb{T}\bigcirc \rangle$, $\langle \mathbb{F}\bigcirc \rangle$, $\langle \lhd \rangle$ et $\langle \blacktriangleleft \rangle$, ainsi que la condition assurant qu'il ne peut y avoir de clôture par la présence de $u \simeq v$ et $u \not\simeq v$.

**Définition 3.19** (CTSS de Hintikka)**.** *Un CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un* CTSS de Hintikka *ssi pour toute formule $\phi, \psi \in \mathcal{L}$ et pour tous labels $x, y \in L_r$ et $u, v, w \in L_s$, on a :*

1. $\mathbb{T}\phi : (x,u) \notin \mathcal{F}$ *ou* $\mathbb{F}\phi : (y,v) \notin \mathcal{F}$ *ou* $x \leqslant y \notin \overline{\mathcal{C}_r}$ *ou* $u \simeq v \notin \overline{\mathcal{C}_s}$
2. $\mathbb{F}\mathrm{I} : (x,u) \notin \mathcal{F}$ *ou* $1_r \leqslant x \notin \overline{\mathcal{C}_r}$
3. $\mathbb{F}\top : (x,u) \notin \mathcal{F}$
4. $\mathbb{F}\phi : (x,u) \notin \mathcal{F}$ *ou* $x$ *est consistant*
5. $u \simeq v \notin \overline{\mathcal{C}_s}$ *ou* $u \not\simeq v \notin \overline{\mathcal{C}_s}$
6. *Si* $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}$ *alors* $1 \leqslant x \in \overline{\mathcal{C}_r}$
7. *Si* $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$ *alors* $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ *et* $\mathbb{T}\psi : (x,u) \in \mathcal{F}$
8. *Si* $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$ *alors* $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ *ou* $\mathbb{F}\psi : (x,u) \in \mathcal{F}$
9. *Si* $\mathbb{T}\phi \vee \psi : (x,u) \in \mathcal{F}$ *alors* $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ *ou* $\mathbb{T}\psi : (x,u) \in \mathcal{F}$
10. *Si* $\mathbb{F}\phi \vee \psi : (x,u) \in \mathcal{F}$ *alors* $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ *et* $\mathbb{F}\psi : (x,u) \in \mathcal{F}$
11. *Si* $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$ *alors* $\forall y \in L_r$, $x \leqslant y \in \overline{\mathcal{C}_r} \Rightarrow \mathbb{F}\phi : (y,u) \in \mathcal{F}$ *ou* $\mathbb{T}\psi : (y,u) \in \mathcal{F}$
12. *Si* $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$ *alors* $\exists y \in L_r$, $x \leqslant y \in \overline{\mathcal{C}_r}$ *et* $\mathbb{T}\phi : (y,u) \in \mathcal{F}$ *et* $\mathbb{F}\psi : (y,u) \in \mathcal{F}$
13. *Si* $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$ *alors* $\exists y, z \in L_r$, $y \circ z \leqslant x \in \overline{\mathcal{C}_r}$ *et* $\mathbb{T}\phi : (y,u) \in \mathcal{F}$ *et* $\mathbb{T}\psi : (z,u) \in \mathcal{F}$

14. *Si* $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}$ *alors* $\forall y,z \in L_r$, $y \circ z \leqslant x \in \overline{\mathcal{C}_r} \Rightarrow \mathbb{F}\phi : (y,u) \in \mathcal{F}$ *ou* $\mathbb{F}\psi : (z,u) \in \mathcal{F}$
15. *Si* $\mathbb{T}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}$ *alors* $\forall y \in L_r$, $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r}) \Rightarrow \mathbb{F}\phi : (y,u) \in \mathcal{F}$ *ou* $\mathbb{T}\psi : (xy,u) \in \mathcal{F}$
16. *Si* $\mathbb{F}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}$ *alors* $\exists y \in L_r$, $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ *et* $\mathbb{T}\phi : (y,u) \in \mathcal{F}$ *et* $\mathbb{F}\psi : (xy,u) \in \mathcal{F}$
17. *Si* $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$ *alors* $\exists v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s}$ *et* $\mathbb{T}\phi : (x,v) \in \mathcal{F}$
18. *Si* $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$ *alors* $\forall v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s} \Rightarrow \mathbb{F}\phi : (x,v) \in \mathcal{F}$
19. *Si* $\mathbb{T}\Box\phi : (x,u) \in \mathcal{F}$ *alors* $\forall v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s} \Rightarrow \mathbb{T}\phi : (x,v) \in \mathcal{F}$
20. *Si* $\mathbb{F}\Box\phi : (x,u) \in \mathcal{F}$ *alors* $\exists v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s}$ *et* $\mathbb{F}\phi : (x,v) \in \mathcal{F}$
21. *Si* $\mathbb{T}\bigcirc\phi : (x,u) \in \mathcal{F}$ *alors* $\exists v \in L_s$, $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ *et* $\mathbb{T}\phi : (x,v) \in \mathcal{F}$
22. *Si* $\mathbb{F}\bigcirc\phi : (x,u) \in \mathcal{F}$ *alors* $\exists v \in L_s$, $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ *et* $\mathbb{F}\phi : (x,v) \in \mathcal{F}$
23. *Si* $u \lhd v \in \overline{\mathcal{C}_s}$ *et* $u \lhd w \in \overline{\mathcal{C}_s}$, *alors* $v \lhd w \in \mathcal{C}_s$ *ou* $w \lhd v \in \mathcal{C}_s$
24. *Si* $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ *et* $u \lhd w \in \overline{\mathcal{C}_s}$, *alors* $v \simeq w \in \mathcal{C}_s$ *ou* $v \lhd w \in \mathcal{C}_s$ *et* $u \not\simeq w \in \mathcal{C}_s$

Les conditions 1 à 5 de cette définition nous assurent qu'un CTSS de Hintikka n'est pas clos. Les autres conditions certifient que toutes les formules d'un CTSS de Hintikka sont saturées. Nous allons maintenant définir une fonction $\Omega$ nous permettant de transformer un CTSS de Hintikka en modèle. Tout d'abord, nous devons mettre en place une fonction auxiliaire pour traiter les labels d'états. En effet, dans nos modèles, les instants sont représentés par des entiers. Nous devons donc trouver un moyen d'énumérer les labels d'états tout en conservant les conditions imposées par les contraintes. Nous commençons par remarquer que, dans une branche saturée, nos labels sont totalement ordonnés par rapport à l'ordre de nos contraintes de labels.

**Lemme 3.9.** *Soit* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *un CTSS de Hintikka issu d'une construction de tableau par les règles de la Figure 3.4. Soit* $u,v \in \mathcal{A}_s(\mathcal{C}_s)$ *deux labels d'états. Au moins une des deux propositions suivantes est vérifiée :*

- $u \lhd v \in \overline{\mathcal{C}_s}$
- $v \lhd u \in \overline{\mathcal{C}_s}$

*Démonstration.*
Notons les labels d'états dans l'ordre où ils sont apparus dans le CTSS. On note $L_s = \{l_i\}_{i \geq 1}$ l'ensemble des labels (possiblement infini) et on prouve le lemme par récurrence sur la taille de $L_s$.

- Cas de base
  Dans ce cas, $L_s = \{l_1\}$, et comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS, on doit avoir $l_1 \lhd l_1 \in \mathcal{C}_s$, ce qui remplit les deux propositions.
- Hérédité
  On suppose (HR) vraie pour $L_s = \{l_1, \ldots, l_{n-1}\}$ (c'est à dire que tout couple de cet ensemble vérifie une des deux propositions annoncées) et on montre qu'elle est toujours vraie après l'ajout d'un label $l_n$.
  Soit $u$ et $v$ deux labels de $L_s = \{l_1, \ldots, l_n\}$. Si $u$ et $v$ sont dans $\{l_1, \ldots, l_{n-1}\}$, on a immédiatement le résultat par (HR). Considérons maintenant le cas où $u = l_n$ ou $v = l_n$. Si $u = v = l_n$, alors de même manière que dans le cas de base on a les deux propositions vérifiées en même temps. Considérons le cas où $u = l_n$ et $v \neq l_n$ (le cas inverse est similaire,

le problème étant symétrique en $u$ et $v$). Dans ce cas, il existe $k \in \{1,\dots,n-1\}$ tel que $v = l_k$. On va montrer $l_k \lhd l_n \in \overline{\mathcal{C}_s}$ ou $l_n \lhd l_k \in \overline{\mathcal{C}_s}$.

$l_n$ a été ajouté dans notre CTSS par une règle de la Figure 3.4. Les seules règles qui introduisent des labels d'états sont $\langle \mathbb{T}\Diamond \rangle$, $\langle \mathbb{F}\Box \rangle$, $\langle \mathbb{T}\bigcirc \rangle$ et $\langle \mathbb{F}\bigcirc \rangle$, $l_n$ provient donc de l'utilisation de l'une de ces règles. Dans toutes ces règles $l_n$ ne peut avoir été introduit qu'à partir d'un label $l_j \in \{l_1,\dots,l_{n-1}\}$ et on a soit $l_j \lhd l_n$ (si $l_n$ vient de $\langle \mathbb{T}\Diamond \rangle$ ou $\langle \mathbb{F}\Box \rangle$) ou bien $l_j \blacktriangleleft l_n$ (si $l_n$ vient de $\langle \mathbb{T}\bigcirc \rangle$ ou $\langle \mathbb{F}\bigcirc \rangle$).

Dans ce second cas, par clôture par la règle $\langle e_n \rangle$, on a aussi $l_j \lhd l_n \in \overline{\mathcal{C}_s}$ et donc on peut dire que dans tous les cas on doit avoir $l_j \lhd l_n \in \overline{\mathcal{C}_s}$.

Si $k = j$, alors on a $l_k \lhd l_n \in \overline{\mathcal{C}_s}$ et la preuve est complète.

Considérons maintenant le cas où $k \neq j$. Par (HR), comme $k$ et $j$ sont tous les deux dans $\{1,\dots,n-1\}$, on doit avoir l'un des cas suivants :

- $l_k \lhd l_j \in \overline{\mathcal{C}_s}$
  Dans ce cas, on a aussi $l_j \lhd l_n \in \overline{\mathcal{C}_s}$, par clôture par la règle $\langle t_s \rangle$, on conclut que $l_k \lhd l_n \in \overline{\mathcal{C}_s}$ et notre preuve est complète.
- $l_j \lhd l_k \in \overline{\mathcal{C}_s}$
  Dans ce cas, on a aussi $l_j \lhd l_n \in \overline{\mathcal{C}_s}$, et comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS de Hintikka, par la condition (23) de la Définition 3.19 on a $l_k \lhd l_n \in \overline{\mathcal{C}_s}$ ou bien $l_n \lhd l_k \in \overline{\mathcal{C}_s}$, ce qui complète notre preuve.
- $l_j \simeq l_k \in \overline{\mathcal{C}_s}$
  Dans ce cas, on a aussi $l_j \lhd l_n \in \overline{\mathcal{C}_s}$, par clôture par la règle $\langle r_l \rangle$ appliquée avec $\lhd$, on conclue que $l_k \lhd l_n \in \overline{\mathcal{C}_s}$ et notre preuve est complète.

Finalement on a montré que pour tout $u, v \in \mathcal{A}_s(\mathcal{C}_s)$, on a $u \lhd v \in \overline{\mathcal{C}_s}$ ou $v \lhd u \in \overline{\mathcal{C}_s}$. □

Ainsi, nos labels d'états sont totalement ordonnés. En outre, on peut définir un plus petit élément pour cette relation.

**Lemme 3.10.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS construit par les règles de la Figure 3.4. Soit $l_1$ le premier label introduit, dans l'ordre de construction du tableau. On a pour tout $u \in \mathcal{A}_s(\mathcal{C}_s)$ :*

1. $l_1 \lhd u \in \overline{\mathcal{C}_s}$
2. *si $u \lhd l_1 \in \overline{\mathcal{C}_s}$, alors $u \simeq l_1 \in \overline{\mathcal{C}_s}$*

*Démonstration.*

1. Notons les labels d'états dans l'ordre où ils sont apparus dans le CTSS. On note $L_s = \{l_i\}_{i \geq 1}$ l'ensemble des labels (possiblement infini) et on prouve que la propriété est vraie par récurrence sur la taille de $L_s$.
   - Cas de base
     Dans ce cas, $L_s = \{l_1\}$, et comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS, on doit avoir $l_1 \lhd l_1 \in \mathcal{C}_s$, ce qui accomplit la propriété qu'on cherche à montrer.

- Hérédité
  On suppose (HR) vraie pour $L_s = \{l_1, \ldots, l_{n-1}\}$ (c'est à dire que tout label de cet ensemble est plus grand que $l_1$) et on montre qu'elle est toujours vraie après l'ajout d'un label $l_n$.
  Soit $u \in \{l_1, \ldots, l_n\}$.
  Si $u \in \{l_1, \ldots, l_{n-1}\}$, alors (HR) nous montre le résultat. Si $u = l_n$, $l_n$ a été ajouté dans notre CTSS par une règle de la Figure 3.4. Les seules règles qui introduisent des labels d'états sont $\langle \mathbb{T}\Diamond \rangle$, $\langle \mathbb{F}\Box \rangle$, $\langle \mathbb{T}\bigcirc \rangle$ et $\langle \mathbb{F}\bigcirc \rangle$, $l_n$ provient donc de l'utilisation de l'une de ces règles. Dans toutes ces règles $l_n$ ne peut avoir été introduit qu'à partir d'un label $l_j \in \{l_1, \ldots, l_{n-1}\}$ et on a soit $l_j \lhd l_n$ (si $l_n$ vient de $\langle \mathbb{T}\Diamond \rangle$ ou $\langle \mathbb{F}\Box \rangle$) ou bien $l_j \blacktriangleleft l_n$ (si $l_n$ vient de $\langle \mathbb{T}\bigcirc \rangle$ ou $\langle \mathbb{F}\bigcirc \rangle$).
  Dans ce second cas, par clôture par la règle $\langle e_n \rangle$, on a aussi $l_j \lhd l_n \in \overline{\mathcal{C}_s}$ et donc on peut dire que dans tous les cas on doit avoir $l_j \lhd l_n \in \overline{\mathcal{C}_s}$.
  Or, comme $l_j \in \{l_1, \ldots, l_{n-1}\}$, par (HR), on a $l_1 \lhd l_j \in \mathcal{C}_s$. Donc, par clôture par la règle $\langle t_s \rangle$, on a $l_1 \lhd l_n \in \mathcal{C}_s$, ce qui montre la propriété.

2. Si $u \lhd l_1 \in \overline{\mathcal{C}_s}$, alors par 1. on a aussi $l_1 \lhd u \in \overline{\mathcal{C}_s}$ et donc par clôture par la règle $\langle a_s \rangle$, on a $l_1 \simeq u \in \mathcal{C}_s$, ce qui montre la propriété.

□

Enfin, nous pouvons définir un moyen de trouver le label suivant dans notre suite de labels. Pour cela, nous utilisons des classes d'équivalences de labels plutôt que les labels eux-mêmes. En effet, plusieurs labels qui sont équivalents par la relation $\simeq$ seront représentés par un même entier lors de l'extraction de contre-modèles. Il est donc plus pertinent de travailler sur les classes d'équivalence.

En outre, la Conjecture 3.1 nous assure que nous pouvons trouver toujours un label minimum parmi les labels plus grands qu'un label d'états donné, ce qui permet de toujours pouvoir trouver un label d'états suivant un autre.

**Lemme 3.11.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka construit par les règles de la Figure 3.4.*

*Pour tout $u \in \mathcal{A}_s(\mathcal{C}_s)$, on note $[u]$ la classe d'équivalence de u par rapport à $\simeq$, c'est à dire $[u] = \{v \in \mathcal{A}_s(\mathcal{C}_s) | u \simeq v \in \overline{\mathcal{C}_s}\}$. On note $[\mathcal{C}_s]$ l'ensemble des classes de labels d'états.*

*Soit $u \in \mathcal{A}_s(\mathcal{C}_s)$. Soit $Sup(u)$ l'ensemble des classes de labels défini par $Sup(u) = \{[v] | u \lhd v \in \overline{\mathcal{C}_s}\}$.*

1. *Si $Sup(u) \neq \emptyset$, alors il existe une unique classe $[v_0] \in Sup(u)$ telle que pour tout $[v] \in Sup(u)$ on a $v_0 \lhd v \in \overline{\mathcal{C}_s}$. On note $\sigma(u)$ cette classe.*
2. *S'il existe un label $v \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$, alors $[v] = \sigma(u)$.*

*Démonstration.*

1. Si l'ensemble $Sup(u)$ est fini, le résultat est trivial (il suffit d'ordonner les labels totalement, comme cela est possible par le Lemme 3.9, puis de prendre la classe du premier de cette liste).

Si $Sup(u)$ est infini, on résonne par l'absurde : supposons qu'il n'existe aucun $[v_0] \in Sup(u)$ tel que pour tout $[v] \in Sup(u)$ on a $v_0 \lhd v \in \overline{\mathcal{C}_s}$. Alors pour tout $[v] \in Sup(u)$, il existe $[v'] \in Sup(u)$ tel que $v \lhd v' \notin \overline{\mathcal{C}_s}$, c'est à dire que $v' \lhd v \in \overline{\mathcal{C}_s}$ (par le Lemme 3.9).

Soit $[v_0] \in Sup(u)$. On vient de montrer qu'il existe $[v_1] \in Sup(u)$ tel que $v_1 \lhd v_0 \in \overline{\mathcal{C}_s}$. En réitérant ce procédé, on peut construire une suite $(v_i)_{i\in\mathbb{N}}$ tel que pour tout $i$ on ait $[v_i] \in Sup(u)$ et $v_{i+1} \lhd v_i \in \overline{\mathcal{C}_s}$. Or par définition de $Sup(u)$, on a $u \lhd v_i \in \overline{\mathcal{C}_s}$ pour tout $i$. En outre, par applications successives de la règle $\langle t_s \rangle$ on montre que $v_i \lhd v_0 \in \overline{\mathcal{C}_s}$ pour tout $i$. On a donc construit un nombre infini de labels $v_i$ tels que $u \lhd v_i \in \overline{\mathcal{C}_s}$ et $v_i \lhd v_0 \in \overline{\mathcal{C}_s}$, ce qui est contraire à la Conjecture 3.1.

Ainsi il doit donc exister $[v_0] \in Sup(u)$ tel que pour tout $[v] \in Sup(u)$ on a $v_0 \lhd v \in \overline{\mathcal{C}_s}$.

Montrons maintenant l'unicité d'une telle classe. Soit $[v_0], [v_1] \in Sup(u)$ qui vérifient la propriété. Comme $[v_1] \in Sup(u)$ et par définition de $\sigma$, on doit avoir $v_0 \lhd v_1 \in \overline{\mathcal{C}_s}$. De manière symétrique, comme $[v_0] \in Sup(u)$ et que $v_1$ vérifie $\sigma$, on doit avoir $v_1 \lhd v_0 \in \overline{\mathcal{C}_s}$. Alors par la règle $\langle a_s \rangle$ on a $v_0 \simeq v_1 \in \overline{\mathcal{C}_s}$ et donc $[v_0] = [v_1]$.

2. Par la règle $\langle e_n \rangle$, on a $u \lhd v \in \overline{\mathcal{C}_s}$, donc $[v] \in Sup(u)$. Soit $[v'] \in Sup(u)$. Alors $u \lhd v' \in \overline{\mathcal{C}_s}$. Comme on est dans un CTSS de Hintikka, par la condition 24 de la Définition 3.19, on a $v \simeq v' \in \overline{\mathcal{C}_s}$ ou bien $v \lhd v' \in \overline{\mathcal{C}_s}$. Dans le cas où $v \simeq v' \in \overline{\mathcal{C}_s}$, par la règle $\langle w_s \rangle$, on a aussi $v \lhd v' \in \overline{\mathcal{C}_s}$. Comme cela est vrai pour tout $[v'] \in Sup(u)$, on a bien $[v] = \sigma(u)$.

□

Bien que ces lemmes soient un peu techniques, ils formalisent une manière simple de dénombrer les labels d'états : le Lemme 3.10 nous indique que le label $l_1$ (ou sa classe) est le plus petit, le Lemme 3.11 nous indique que pour chaque label qui a des labels plus grands, il y en a au moins un qui est le plus petit d'entre eux et qui est donc le label suivant dans la suite. On peut ainsi mettre en forme tout ensemble de labels comme une suite de classes de labels ordonnées.

Encore une fois, la Conjecture 3.1 permet de s'assurer qu'une telle suite est finie.

Pour tout ensemble $\mathcal{E}$, et jusqu'à la fin de cette thèse, on note $|\mathcal{E}|$ la cardinalité de ce ensemble (c'est à dire son nombre d'éléments).

**Lemme 3.12.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka construit par les règles de la Figure 3.4. Soit $u, v \in \mathcal{A}_s(\mathcal{C}_s)$ tels que $u \lhd v \in \overline{\mathcal{C}_s}$. On note $[u;v]$ l'ensemble $[u;v] = \{[w] \in \mathcal{A}_s(\mathcal{C}_s) | u \lhd w \in \overline{\mathcal{C}_s}$ et $w \lhd v \in \overline{\mathcal{C}_s}\}$.*

1. *$[u;v]$ est fini ;*
2. *On note $k = |[u;v]|$. Il existe une suite $([u_i])_{i<k}$ telle que :*
   - *$[u_0] = [u]$ ;*
   - *$[u_{k-1}] = [v]$ ;*
   - *si $k > 1$, pour tout $i$, $[u_{i+1}] = \sigma(u_i)$.*

*Démonstration.*

1. Ce point découle directement de la Conjecture 3.1.
2. On démontre par récurrence sur la taille de $k$.

- Cas de base
  Au minimum on a $k = 1$. Dans ce cas là on a $[u] = [v]$ et $[u;v] = \{[u]\}$. Alors la suite $\{[u_0]\} = \{[u]\}$ convient.
- Hérédité
  On suppose que la propriété est vraie pour tous les couples de labels tels que $k = n$. On démontre qu'elle reste vraie pour $n+1$.
  Soient $u, v$ tels que $|[u;v]| = n+1$. Comme $u \lhd v \in \overline{\mathcal{C}_s}$, $Sup(u)$ n'est pas vide. On peut donc définir $[u'] = \sigma(u)$. On considère l'ensemble $[u';v]$. On a $|[u';v]| = n$. Par hypothèse de récurrence, il existe une suite $(v_i)_{i \leq n}$ telle que $[v_0] = [u']$, $[v_{n-1}] = [v]$ et pour tout $i$, $[v_{i+1}] = \sigma(v_i)$. On pose alors $[u_0] = [u]$ et pour tout $0 < i \leq n+1$, $[u_i] = [v_{i-1}]$. Alors $[u_0] = [u]$ et $[u_n] = [v_{n-1}] = [v]$ par construction. En outre, pour tout $i \neq 0$ on a $[u_{i+1}] = [v_i]$ et $[u_i] = [v_{i-1}]$, donc $[u_{i+1}] = \sigma(u_i)$. Enfin par définition $[u_1] = [v_0] = [u'] = \sigma([u]) = \sigma([u_0])$. Finalement la propriété est vraie par récurrence pour tous les couples $u, v$.

□

Avec cette structure, on peut alors facilement numéroter nos labels :

**Définition 3.20** (Fonction ω). *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka.*
*On définit une fonction $\omega : [\mathcal{C}_s] \to \mathbb{N}$ par :*

- $\omega([l_1]) = 0$ ;
- *Si* $\omega([u]) = n$, *alors* $\omega(\sigma(u)) = n+1$.

Vérifions que cette définition de notre ordonnancement sur les labels d'états remplit les propriétés qu'on attend d'elle.

**Lemme 3.13.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka. La fonction ω définie précédemment vérifie les propriétés suivantes :*

1. *Pour tout $u \in \mathcal{A}_s(\mathcal{C}_s)$, il existe $n \in \mathbb{N}$ tel que $\omega([u]) = n$.*
2. *Pour tout $u, v \in \mathcal{A}_s(\mathcal{C}_s)$, $u \simeq v \in \overline{\mathcal{C}_s}$ ssi $\omega([u]) = \omega([v])$.*
3. *Pour tout $u, v \in \mathcal{A}_s(\mathcal{C}_s)$, si $u \not\simeq v \in \overline{\mathcal{C}_s}$ alors $\omega([u]) \neq \omega([v])$.*
4. *Pour tout $u, v \in \mathcal{A}_s(\mathcal{C}_s)$, $u \lhd v \in \overline{\mathcal{C}_s}$ ssi $\omega([u]) \leq \omega([v])$.*
5. *Pour tout $u, v \in \mathcal{A}_s(\mathcal{C}_s)$, si $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ alors $\omega([v]) = \omega([u]) + 1$.*

*Démonstration.*

1. Soit $u \in \mathcal{A}_s(\mathcal{C}_s)$. On considère l'ensemble $[l_1;u]$ tel que défini dans le Lemme 3.12. On a alors une suite $([u_i])_{i<k}$ telle que $[u_0] = [l_1]$, $[u_{k-1}] = [u]$ et pour tout $i$, $[u_{i+1}] = \sigma(u_i)$. On démontre que tous les éléments de $([u_i])$ ont une image par ω par récurrence. Le cas de base est $[u_0] = [l_1]$. Par définition $\omega([l_1]) = 0$. Supposons que $\omega([u_i]) = n$ existe. Si $i = k-1$, $\omega(u)$ existe et la preuve est finie. Sinon, $\sigma(u_i)$ existe et on a $[u_{i+1}] = \sigma(u_i)$. Par suite, $\omega(u_{i+1}) = n+1$ existe. Une simple récurrence donne alors que $\omega(u)$ existe.
2. $u \simeq v \in \overline{\mathcal{C}_s}$ ssi $[u] = [v]$ ssi $\omega([u]) = \omega([v])$ (ω est trivialement injective par sa définition).

3. On suppose que $u \not\simeq v \in \overline{\mathcal{C}_s}$. Alors par le point 5 de la Définition 3.19, on a $u \simeq v \notin \overline{\mathcal{C}_s}$ et donc $[u] \neq [v]$. Par suite, $\omega([u]) \neq \omega([v])$.
4. - Sens direct
   On suppose que $u \lhd v \in \overline{\mathcal{C}_s}$.
   On étudie d'abord le cas où $[u] \neq [v]$. On considère l'ensemble $[u;v]$ tel que défini dans le Lemme 3.12. On a alors une suite $([u_i])_{i<k}$ telle que $[u_0] = [u]$, $[u_{k-1}] = [v]$ et, comme $k > 1$, pour tout $i$, $[u_{i+1}] = \sigma(u_i)$. Alors, par définition de $\omega$, on a pour tout $i$ que $\omega([u_{i+1}]) = \omega([u_i]) + 1$. C'est à dire que pour tout $i$, $\omega([u_{i+1}]) > \omega([u_i])$ pour tout $i$, et donc en particulier $\omega([u_{k-1}]) > \omega([u_0])$, c'est à dire $\omega([v]) > \omega([u])$.
   Dans le cas où $[u] = [v]$, on a alors que $\omega([u]) = \omega([v])$. Ainsi, en toute généralité, on a $\omega([u]) \leq \omega([v])$
   - Sens réciproque
   On suppose que $\omega([u]) \leq \omega([v])$. On considère d'abord le cas où $\omega([u]) < \omega([v])$. Dans ce cas, on raisonne par l'absurde. On suppose que $u \lhd v \notin \overline{\mathcal{C}_s}$. Alors, par le Lemme 3.9, on a $v \lhd u \in \overline{\mathcal{C}_s}$. On peut utiliser le sens direct, ce qui nous donne que $\omega([v]) \leq \omega([u])$. Comme on a supposé que $\omega([u]) < \omega([v])$, c'est absurde. Donc $u \lhd v \in \overline{\mathcal{C}_s}$.
   Dans le cas où $\omega([u]) = \omega([v])$, alors par 2. on a $u \simeq v \in \overline{\mathcal{C}_s}$ et par la règle $\langle w_s \rangle$, $u \lhd v \in \overline{\mathcal{C}_s}$.
5. On suppose que $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$. D'après le point 2 du Lemme 3.11, on a alors $[v] = \sigma(u)$, donc par définition $\omega([v]) = 1 + \omega([u])$.

□

Nous pouvons maintenant définir la fonction qui transforme nos branches saturées en modèles.

**Définition 3.21** (Fonction $\Omega$)**.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka. Soit $\overline{\mathcal{C}_{r\omega}}$ la restriction de $\overline{\mathcal{C}_r}$ aux contraintes qui n'incluent que des labels de ressources consistants (voir Définition 3.12). La fonction $\Omega$ associée à $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un triplet $\Omega(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle) = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où $\mathcal{R} = (R, \bullet, e, \pi, \sqsubseteq, S)$, tel que :*

- $R = \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}}) \cup \{\pi\}$, *avec* $\pi \notin \mathcal{D}_r(\overline{\mathcal{C}_r})$.
- $e = 1_r$.
- $\bullet$ *est définie par*

$$\forall r_1, r_2 \in R \begin{cases} r_1 \bullet r_2 = r_1 \circ r_2 & \text{si } r_1 \circ r_2 \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}}) \\ r_1 \bullet r_2 = \pi & \text{sinon.} \end{cases}$$

- $r_1 \sqsubseteq r_2$ *ssi* $r_1 \leqslant r_2 \in \overline{\mathcal{C}_{r\omega}}$ *ou* $r_2 = \pi$.
- $S = \{\omega([u]) | u \in \mathcal{A}_s(\mathcal{C}_s)\}$.
- $\forall p \in Prop$, $\forall r \in R$, $\forall \omega([u]) \in S$, $(r, \omega([u])) \in [\![p]\!]$ *ssi* $(r = \pi)$ *ou* $(\exists r' \in R, r' \sqsubseteq r$ *tel que* $\mathbb{T}p : (r', u) \in \mathcal{F})$.
- $\vDash_{\mathcal{K}}$ *est définie par induction structurelle à partir de* $[\![\cdot]\!]$ *de la même manière que dans la Définition 3.4.*

Vérifions maintenant que cette opération nous donne bien un modèle linéaire.

**Lemme 3.14.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka tel que $1_r \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ et $1_r$ est un label consistant. Soit $\Omega(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle) = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où $\mathcal{R} = (R, \bullet, e, \pi, \sqsubseteq, S)$. $(\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est un modèle linéaire de ressources.*

*Démonstration.*
On montre tout d'abord que $\mathcal{R} = (R, \bullet, e, \pi, \sqsubseteq, S)$ est un monoïde linéaire de ressources.

On doit montrer que $(R, \bullet, e, \pi, \sqsubseteq)$ est un monoïde de ressources :

- $R$ est un ensemble.
- $\pi \in R$ par construction.
- On a supposé que $1_r$ était consistant et que $1_r \in \mathcal{D}_r(\overline{\mathcal{C}_r})$, donc $1_r \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, c'est à dire $e \in R$.
- Montrons que la composition $\bullet$ vérifie les propriétés recherchées :
  - $\bullet$ est définie pour tout $r_1, r_2 \in R$, elle est donc totale. En outre, pour tout $r_1, r_2 \in R$, $r_1 \bullet r_2 \in R$, donc $\bullet$ est bien interne.
  - Soit $r \in R$. Comme $\pi \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, $r \circ \pi \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, donc $r \bullet \pi = \pi$. Ainsi, $\pi$ est bien un élément absorbant.
  - Soit $r \in R$. Si $r \neq \pi$, alors $r \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, donc $r = r \circ 1_r \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$. Alors $r \bullet e = r \circ 1_r = r$. Dans le cas où $r = \pi$, d'après le point précédent $r \bullet e = \pi \bullet e = \pi = r$. Dans tous les cas, $r \bullet e = r$ et $e$ est bien un élément neutre.
  - Soient $r_1, r_2 \in R$. Si $r_1 \circ r_2 \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, alors $r_1 \bullet r_2 = \pi$. Dans ce cas, on doit aussi avoir $r_2 \circ r_1 \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$ (autrement on a une absurdité puisque $\circ$ est associative), et on a $r_2 \bullet r_1 = \pi$, donc $r_1 \bullet r_2 = r_2 \bullet r_1$. Dans le cas où $r_1 \circ r_2 \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, alors par la commutativité de $\circ$ on a $r_1 \bullet r_2 = r_1 \circ r_2 = r_2 \circ r_1 = r_2 \bullet r_1$. Dans tous les cas $r_1 \bullet r_2 = r_1 \bullet r_1$ et $\bullet$ est bien commutative.
  - Soient $r_1, r_2, r_3 \in R$. Si $r_1 \circ r_2 \circ r_3 \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, on a $r_1 \bullet (r_2 \bullet r_3) = \pi$ et $(r_1 \bullet r_2) \bullet r_3 = \pi$, donc $r_1 \bullet (r_2 \bullet r_3) = (r_1 \bullet r_2) \bullet r_3$. Dans le cas contraire, par la Proposition 3.4, on a $r_1 \circ r_2 \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$ et $r_2 \circ r_3 \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$. Alors, comme $\circ$ est associative, on obtient $r_1 \bullet (r_2 \bullet r_3) = r_1 \circ (r_2 \circ r_3) = (r_1 \circ r_2) \circ r_3 = (r_1 \bullet r_2) \bullet r_3$. Ainsi dans tous les cas on a $r_1 \bullet (r_2 \bullet r_3) = (r_1 \bullet r_2) \bullet r_3$ et $\bullet$ est associative.
- Montrons maintenant que $\sqsubseteq$ est un pré-ordre compatible avec $\bullet$ et ayant $\pi$ comme plus grand élément.
  - Soit $r_1, r_2, r_3 \in R$ tels que $r_1 \sqsubseteq r_2$ et $r_2 \sqsubseteq r_3$. Notons qu'on ne peut alors pas avoir $r_1 = \pi$ ou $r_2 = \pi$ car $\sqsubseteq$ n'est pas défini pour le cas où l'opérande de gauche est $\pi$. On a donc $r_1 \leqslant r_2 \in \overline{\mathcal{C}_{r\omega}}$ et il y a deux cas : si $r_3 = \pi$, alors par définition on a $r_1 \sqsubseteq r_3$. Sinon, on a $r_2 \leqslant r_3 \in \overline{\mathcal{C}_{r\omega}}$ et, par la règle $\langle t_r \rangle$ on a aussi $r_1 \leqslant r_3 \in \overline{\mathcal{C}_{r\omega}}$, donc $r_1 \sqsubseteq r_3$. Ainsi dans tous les cas $r_1 \sqsubseteq r_3$ et $\sqsubseteq$ est transitive.
  - Soit $r \in R$. Si $r = \pi$, alors par définition $r \sqsubseteq r$. Dans le cas contraire, $r \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$. Alors par le Corollaire 3.1, on a $r \leqslant r \in \overline{\mathcal{C}_{r\omega}}$, et donc $r \sqsubseteq r$. Ainsi, dans tous les cas $r \sqsubseteq r$ et $\sqsubseteq$ est réflexive. Comme $\sqsubseteq$ est aussi transitive, c'est un pré-ordre.

- Soit $r_1, r_2, s \in R$ tels que $r_1 \sqsubseteq r_2$. Si $r_2 \circ s \notin \mathcal{D}_r(\overline{\mathcal{C}_r})$, alors $r_2 \bullet s = \pi$ et $r_1 \bullet s \sqsubseteq r_2 \bullet s$. Si $r_2 \circ s \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, alors par le Corollaire 3.1, on a $r_2 s \leqslant r_2 s \in \overline{\mathcal{C}_{r\omega}}$. En outre, comme $r_2 \circ s \in \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$, nécessairement $r_2 \neq \pi$ et donc comme $r_1 \sqsubseteq r_2$, on a $r_1 \leqslant r_2 \in \overline{\mathcal{C}_{r\omega}}$. Alors par la règle $\langle c_r \rangle$, on a $r_1 s \leqslant r_2 s \in \overline{\mathcal{C}_{r\omega}}$ et par suite $r_1 \bullet s \sqsubseteq r_2 \bullet s$. Ainsi, dans tous les cas on a $r_1 \bullet s \sqsubseteq r_2 \bullet s$ et donc $\sqsubseteq$ est compatible avec $\bullet$.
- Soit $r \in R$. Par définition on a $r \sqsubseteq \pi$, donc $\pi$ est bien le plus grand élément pour $\sqsubseteq$.

On a ainsi montré par tous ces points que $(R, \bullet, e, \pi, \sqsubseteq)$ est un monoïde de ressources. En outre, par définition de $\omega$, on a trivialement que $S \subseteq \mathbb{N}$, et donc $\mathcal{R}$ est un monoïde linéaire de ressources.

On doit maintenant montrer que $[\![\cdot]\!]$ est bien une interprétation linéaire, c'est à dire qu'elle est bien monotone et inconsistante.

- Soit $p \in Prop$, $i \in S$, $r, r' \in R$ tels que $r \sqsubseteq r'$ et $(r, i) \in [\![p]\!]$. On a deux cas :
  - $r = \pi$. Dans ce cas, comme $r \sqsubseteq r'$, on a nécessairement $r' = \pi$, et donc $(r', i) \in [\![p]\!]$.
  - on a $i = \omega([u])$ et il existe $r'' \in R$ tel que $r'' \sqsubseteq r$ et $\mathbb{T}p : (r'', u) \in \mathcal{F}$. Alors, comme $\sqsubseteq$ est transitive, on a alors $r'' \sqsubseteq r'$ et donc par définition $(r', i) \in [\![p]\!]$.

  Dans les deux cas, $(r', i) \in [\![p]\!]$ et donc $[\![\cdot]\!]$ est monotone.
- Soit $p \in Prop$, $i \in S$, $r \in R$ tels que $\pi \sqsubseteq r$. On ne peut pas avoir $\pi \leqslant r \in \overline{\mathcal{C}_{r\omega}}$, donc nécessairement on a $r = \pi$ et donc $(r, i) \in [\![p]\!]$. Ainsi, $[\![\cdot]\!]$ est inconsistante.

Finalement, $[\![\cdot]\!]$ est une interprétation. Enfin, comme $\vDash_{\mathcal{K}}$ est définie comme dans la Définition 3.4, $(\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est bien un modèle linéaire de ressources.

□

Notre fonction $\Omega$ produit donc bien des modèles. Il faut en outre que les modèles produits traduisent correctement les étiquettes de vérité des formules étiquetées d'un tableau en la validité ou non-validité des formules correspondantes dans le modèle.

**Lemme 3.15.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ un CTSS de Hintikka. Soit $\Omega(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle) = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où $\mathcal{R} = (R, \bullet, e, \pi, \sqsubseteq, S)$. On a le résultat suivant pour toute formule $\phi$, pour tout label de ressources $r$ et pour tout label d'états $u$ :*

*(a)* $(\pi, \omega([u])) \vDash_{\mathcal{K}} \phi$.

*(b) Si* $\mathbb{F}\phi : (r, u) \in \mathcal{F}$ *et $r$ est consistant, alors* $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi$ .

*(c) Si* $\mathbb{T}\phi : (r, u) \in \mathcal{F}$ *et $r$ est consistant, alors* $(r, \omega([u])) \vDash_{\mathcal{K}} \phi$.

*Démonstration.*
Par le Lemme 3.14, $(\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est un modèle linéaire de ressources. La propriété (a) est validée par le Lemme 3.2. On prouve les deux autres propriétés simultanément par récurrence sur la taille de $\phi$.

La preuve complète est en Annexe A. □

On peut donc conclure sur notre extraction de contre-modèles :

**Lemme 3.16.** *Si $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS de Hintikka tel que $\mathbb{F}\phi : (1_r, l_1) \in \mathcal{F}$, alors $\phi$ n'est pas valide.*

*Démonstration.*
Si $1_r$ est inconsistant, on a une contradiction par la condition 4 de la Définition 3.19, puisque $\mathbb{F}\phi : (1_r, l_1) \in \mathcal{F}$. Donc $1_r$ est consistant. En outre, puisque $\mathbb{F}\phi : (1_r, l_1) \in \mathcal{F}$, on a $1_r \in \mathcal{D}_r(\overline{C_r})$ par ($P_{CSS}$ et le Corollaire 3.1. On peut donc appliquer le Lemme 3.14 et $\Omega(\langle \mathcal{F}, C_r, C_s \rangle)$ est un modèle linéaire de ressources. Par le Lemme 3.15, et la définition de $\Omega$, $(e, \omega([l_1])) \not\Vdash_{\mathcal{K}} \phi$ dans ce modèle. Comme en outre $\omega([l_1]) = 0$, $\Omega(\langle \mathcal{F}, C_r, C_s \rangle)$ est un contre-modèle de $\phi$ et $\phi$ n'est pas valide. □

### 3.5.3 Un exemple d'extraction de contre-modèles

Nous allons maintenant illustrer notre méthode d'extraction de contre-modèle par un exemple. On considère la formule suivante :

$$(\Diamond A * \circ B) \to (\Diamond B * \circ A)$$

En appliquant le calcul des tableaux de LTBI à cette formule, on obtient le tableau représenté dans les figures 3.9, 3.10 et 3.11.

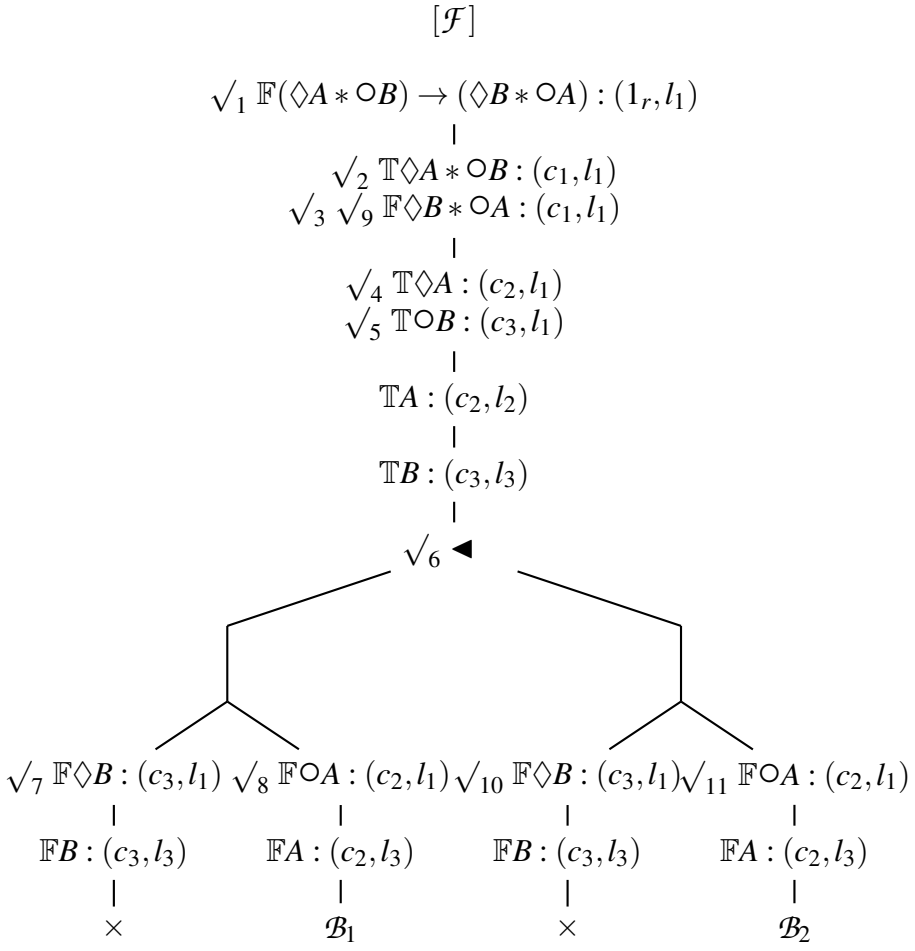



FIGURE 3.9 – Formules du LTBI-tableau pour $(\Diamond A * \circ B) \to (\Diamond B * \circ A)$

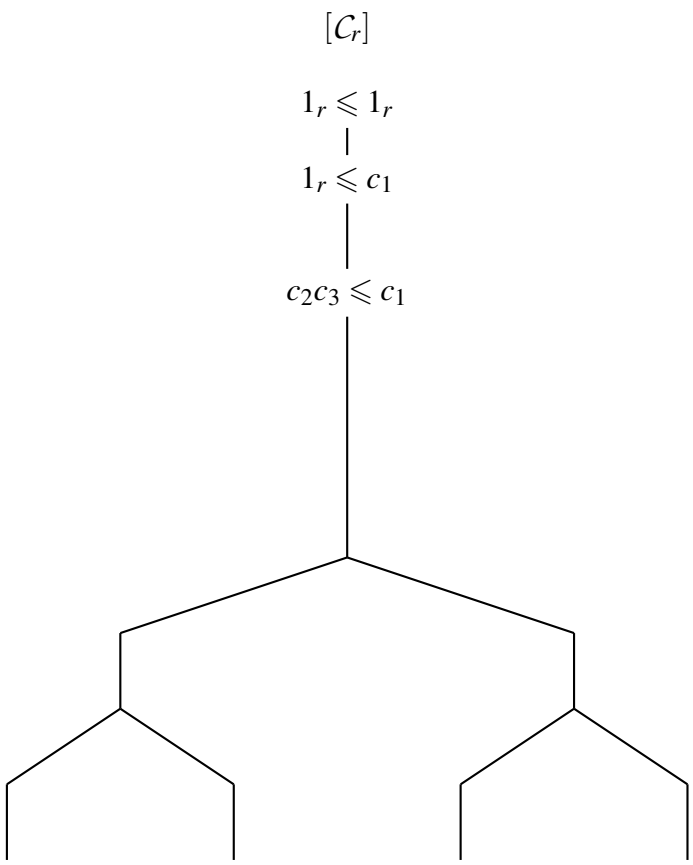
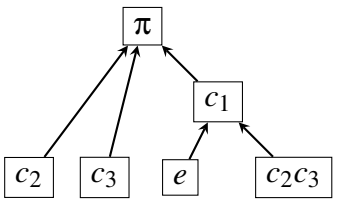



FIGURE 3.10 – Contraintes de ressources pour $(\Diamond A * \circ B) \to (\Diamond B * \circ A)$

Ce LTBI-tableau contient deux branches closes et deux branches ouvertes (notées $\mathcal{B}_1$ et $\mathcal{B}_2$) qui sont des CTSS de Hintikka. Par le Lemme 3.16, $(\Diamond A * \circ B) \to (\Diamond B * \circ A)$ n'est pas valide. De plus, $\Omega(\mathcal{B}_1)$ et $\Omega(\mathcal{B}_1)$ sont des contre-modèles de cette formule. Nous allons extraire ces contre-modèles en utilisant la Définition 3.21.

Commençons par $\Omega(\mathcal{B}_1)$. On a $\Omega(\mathcal{B}_1) = (\mathcal{M}_1, [\![\cdot]\!]_1, \vDash_{\mathcal{K}})$ avec $\mathcal{M}_1 = (R, \bullet, e, \pi, \sqsubseteq, S_1)$ de sorte que :

- $R = \{e, c_1, c_2, c_3, c_2c_3, \pi\}$
- La relation $\sqsubseteq$ est définie par la figure suivante (la clôture réflexive n'est pas représentée) :




- La composition de ressources $\bullet$ est définie par :

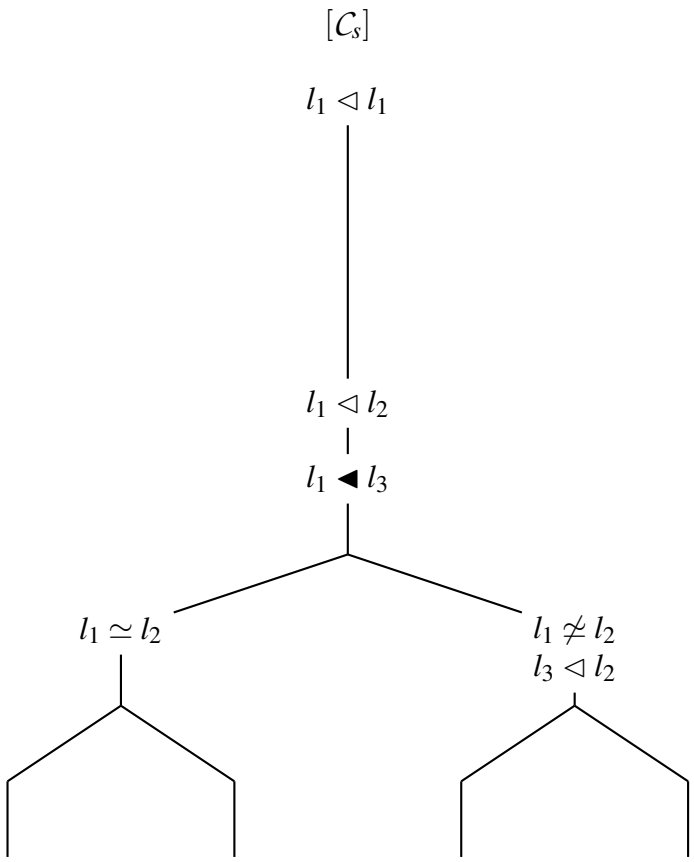



FIGURE 3.11 – Contraintes d'états pour $(\Diamond A * \circ B) \to (\Diamond B * \circ A)$

| $\bullet$ | $e$ | $c_1$ | $c_2$ | $c_3$ | $c_2c_3$ | $\pi$ |
|---|---|---|---|---|---|---|
| $e$ | $e$ | $c_1$ | $c_2$ | $c_3$ | $c_2c_3$ | $\pi$ |
| $c_1$ | $c_1$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ |
| $c_2$ | $c_2$ | $\pi$ | $\pi$ | $c_2c_3$ | $\pi$ | $\pi$ |
| $c_3$ | $c_3$ | $\pi$ | $c_2c_3$ | $\pi$ | $\pi$ | $\pi$ |
| $c_2c_3$ | $c_2c_3$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ |
| $\pi$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ | $\pi$ |

- Pour composer $S_1$, nous devons tout d'abord calculer la classe de chaque label d'états. On a :
  - $[l_1] = [l_2] = \{l_1, l_2\}$
  - $[l_3] = \{l_3\}$

  Par la Définition 3.20, on a $\omega([l_1]) = \omega([l_2]) = 0$. En outre, par le point 5 du Lemme 3.13, puisque $l_1 \blacktriangleleft l_3 \in \overline{\mathcal{C}_s}$ on doit avoir $\omega([l_3]) = \omega([l_1]) + 1$, donc $\omega([l_3]) = 1$. Donc $S_1 = \{0, 1\}$.
- Enfin l'interprétation $[\![\cdot]\!]_1$ est définie par :
  - $[\![A]\!]_1 = \{(\pi, 0), (\pi, 1), (c_2, 0)\}$
  - $[\![B]\!]_1 = \{(\pi, 0), (\pi, 1), (c_3, 1)\}$

On peut vérifier qu'on obtient bien un contre-modèle de $(\Diamond A * \circ B) \to (\Diamond B * \circ A)$ :

1. $(c_3,1) \in [\![B]\!]_1$, donc $(c_3,1) \vDash_{\mathcal{K}} B$
2. Par (1.) on a $(c_3,0) \vDash_{\mathcal{K}} \circ B$
3. $(c_2,0) \in [\![A]\!]_1$, donc $(c_2,0) \vDash_{\mathcal{K}} A$
4. Par (3.) et comme $0 \leq 0$, on a $(c_2,0) \vDash_{\mathcal{K}} \Diamond A$
5. Par (2.) et (4.) et comme $c_2 c_3 \sqsubseteq c_1$, on a $(c_1,0) \vDash_{\mathcal{K}} \Diamond A * \circ B$
6. Pour tout $r \in R$, $(r,1) \notin [\![A]\!]_1$, donc $(r,1) \nvDash_{\mathcal{K}} A$, donc $(r,0) \nvDash_{\mathcal{K}} \circ A$ ;
7. Par (6.), pour tout couple $(r,r') \in R$ tel que $r \bullet r' \sqsubseteq c_1$, on a $(r,0) \nvDash_{\mathcal{K}} \circ A$.
8. Par (7.), on a $(c_1,0) \nvDash_{\mathcal{K}} \Diamond B * \circ A$
9. Par (5.) et (8.), et comme $e \sqsubseteq c_1$, on peut conclure que $(e,0) \nvDash_{\mathcal{K}} (\Diamond A * \circ B) \to (\Diamond B * \circ A)$

Concernant la branche $\mathcal{B}_2$, étant donné qu'elle présente les mêmes contraintes de ressources, on peut y utiliser les mêmes $R$, $\bullet$, $e$ et $\sqsubseteq$. On a donc $\Omega(\mathcal{B}_2) = (\mathcal{M}_2, [\![\cdot]\!]_2, \vDash_{\mathcal{K}})$ avec $\mathcal{M}_2 = (R, \bullet, e, \pi, \sqsubseteq, S_2)$, et $S_2$ est construit comme suit.

Pour composer $S_2$, nous devons à nouveau calculer la classe de chaque label d'états. On a cette fois :

- $[l_1] = \{l_1\}$
- $[l_2] = \{l_2\}$
- $[l_3] = \{l_3\}$

Par la Définition 3.20, on a $\omega([l_1]) = 0$. Par suite, on a $\sigma(l_1) = [l_3]$ puisque $l_3 \lhd l_2 \in \overline{\mathcal{C}_s}$, et donc $\sigma(l_3) = l_2$. Ainsi on obtient $\omega([l_3]) = 1$, $\omega([l_2]) = 2$, et enfin $S_2 = \{0,1,2\}$.

On a ensuite l'interprétation suivante :

- $[\![A]\!]_2 = \{(\pi,0),(\pi,1),(c_2,2)\}$
- $[\![B]\!]_2 = \{(\pi,0),(\pi,1),(c_3,1)\}$

On vérifie encore qu'on a bien obtenu un contre-modèle :

1. $(c_3,1) \in [\![B]\!]_1$, donc $(c_3,1) \vDash_{\mathcal{K}} B$
2. Par (1.) on a $(c_3,0) \vDash_{\mathcal{K}} \circ B$
3. $(c_2,2) \in [\![A]\!]_1$, donc $(c_2,2) \vDash_{\mathcal{K}} A$
4. Par (3.) et comme $0 \leq 2$, on a $(c_2,0) \vDash_{\mathcal{K}} \Diamond A$
5. Par (2.) et (4.) et comme $c_2 c_3 \sqsubseteq c_1$, on a $(c_1,0) \vDash_{\mathcal{K}} \Diamond A * \circ B$
6. Pour tout $r \in R$, $(r,1) \notin [\![A]\!]_1$, donc $(r,1) \nvDash_{\mathcal{K}} A$, donc $(r,0) \nvDash_{\mathcal{K}} \circ A$ ;
7. Par (6.), pour tout couple $(r,r') \in R$ tel que $r \bullet r' \sqsubseteq c_1$, on a $(r,0) \nvDash_{\mathcal{K}} \circ A$.
8. Par (7.), on a $(c_1,0) \nvDash_{\mathcal{K}} \Diamond B * \circ A$
9. Par (5.) et (8.), et comme $e \sqsubseteq c_1$, on peut conclure que $(e,0) \nvDash_{\mathcal{K}} (\Diamond A * \circ B) \to (\Diamond B * \circ A)$

### 3.5.4 Complétude du calcul

Nous allons maintenant détailler la preuve de complétude. C'est une adaptation de la preuve de DBI [34]. Elle consiste en l'obtention, à partir d'un CTSS non clos, d'un CTSS de Hintikka. Pour cela, nous utiliserons une stratégie équitable, c'est à dire une suite où chaque formule ou contrainte pouvant être sélectionnés par une règle apparaît un nombre infini de fois, et un oracle contenant tous les CTSS *consistants* (non finis mais saturés).

Par rapport à la preuve de DBI, cette preuve doit prendre en compte la possibilité de sélectionner une contrainte pour appliquer les règles $\langle\lhd\rangle$ ou $\langle\blacktriangleleft\rangle$. Ainsi, notre stratégie équitable doit contenir toutes les contraintes d'états possibles. On définit d'abord les stratégies équitables.

**Définition 3.22** (Stratégie Équitable)**.** *Une* stratégie équitable *est une suite de formules et de contraintes* $(S_i)_{i\in\mathbb{N}}$ *dans* $(\{\mathbb{T},\mathbb{F}\}\times\mathcal{L}\times L_r\times L_s)\cup(L_s\times\{\lhd,\blacktriangleleft,\simeq,\not\simeq\}\times L_s)$ *telle que chaque formule et chaque contrainte apparaît un nombre infini de fois dans cette suite, c'est à dire que* $\{i\in\mathbb{N}\mid S_i\equiv\mathbb{S}F:(x,u)\}$ *est infini pour tout* $\mathbb{S}F:(x,u)\in\{\mathbb{T},\mathbb{F}\}\times\mathcal{L}\times L_r\times L_s$ *et* $\{i\in\mathbb{N}\mid S_i\equiv u\star v\}$ *est infini pour tout* $u\star v\in(L_s\times\{\lhd,\blacktriangleleft,\simeq,\not\simeq\}\times L_s)$.

On s'assure qu'un tel objet existe bien.

**Proposition 3.5.** *Il existe une stratégie équitable.*

*Démonstration.*
Soit $X=(\{\mathbb{T},\mathbb{F}\}\times\mathcal{L}\times L_r\times L_s)\cup(L_s\times\{\lhd,\blacktriangleleft,\simeq,\not\simeq\}\times Ls)$ l'ensemble de toutes les formules étiquetées et toutes les contraintes d'états. Comme *Prop* est dénombrable, $\mathcal{L}$ l'est aussi. En outre, $L_r$ et $L_s$ sont dénombrables (puisque $\gamma_r$ est dénombrable). Finalement, $\{\lhd,\blacktriangleleft,\simeq,\not\simeq\}$ est évidemment dénombrable. Par conséquent, $X$ est dénombrable.

Donc $\mathbb{N}\times X$ est dénombrable et il existe une fonction surjective $\varphi:\mathbb{N}\longrightarrow\mathbb{N}\times X$. Soit $p:\mathbb{N}\times X\longrightarrow X$ définie par $p(i,x)=x$ et $u=p\circ\varphi$. On montre que $u$ est une stratégie équitable en montrant que pour tout $x\in X$, $u^{-1}(\{x\})$ est infini. Soit $x\in X$. $u^{-1}(\{x\})=\varphi^{-1}(p^{-1}(\{x\}))$. Mais $p^{-1}(\{x\})=\{(i,x)|i\in\mathbb{N}\}$ donc $p^{-1}(x)$ est infini. Comme $\varphi$ est surjective, $\varphi^{-1}(p^{-1}(\{x\}))$ est aussi infini. □

On définit ensuite certaines propriétés d'un ensemble de CTSS, qui seront utilisées pour définir un oracle.

**Définition 3.23** (Ensemble de CTSS clos, finis, saturés)**.** *Soit* $\mathcal{P}$ *un ensemble de CTSS.*

1. $\mathcal{P}$ *est* $\preccurlyeq$-clos *si on a* $\langle\mathcal{F},\mathcal{C}_r,\mathcal{C}_s\rangle\in\mathcal{P}$ *chaque fois qu'on a* $\langle\mathcal{F},\mathcal{C}_r,\mathcal{C}_s\rangle\preccurlyeq\langle\mathcal{F}',\mathcal{C}'_r,\mathcal{C}'_s\rangle$ *et* $\langle\mathcal{F}',\mathcal{C}'_r,\mathcal{C}'_s\rangle\in\mathcal{P}$.
2. $\mathcal{P}$ *est de* caractère fini *si on a* $\langle\mathcal{F},\mathcal{C}_r,\mathcal{C}_s\rangle\in\mathcal{P}$ *chaque fois qu'on a* $\langle\mathcal{F}_f,\mathcal{C}_{rf},\mathcal{C}_{sf}\rangle\in\mathcal{P}$ *pour tout* $\langle\mathcal{F}_f,\mathcal{C}_{rf},\mathcal{C}_{sf}\rangle\preccurlyeq_f\langle\mathcal{F},\mathcal{C}_r,\mathcal{C}_s\rangle$.
3. $\mathcal{P}$ *est* saturé *si pour tout* $\langle\mathcal{F},\mathcal{C}_r,\mathcal{C}_s\rangle\in\mathcal{P}$ *et pour toute instance*

$$\frac{cond(\mathcal{F},\mathcal{C}_r,\mathcal{C}_s)}{\langle\mathcal{F}_1,\mathcal{C}_{r1},\mathcal{C}_{s1}\rangle\ \mid\ \ldots\ \mid\ \langle\mathcal{F}_k,\mathcal{C}_{rk},\mathcal{C}_{sk}\rangle}$$

*d'une règle de la Figure 3.4, si* $cond(\mathcal{F},\mathcal{C}_r,\mathcal{C}_s)$ *est remplie, alors* $\langle\mathcal{F}\cup\mathcal{F}_i,\mathcal{C}_r\cup\mathcal{C}_{ri},\mathcal{C}_s\cup\mathcal{C}_{si}\rangle\in\mathcal{P}$ *pour au moins un* $i\in\{1,\ldots,k\}$.

On peut alors définir un oracle.

**Définition 3.24** (Oracle)**.** *Un* oracle *est un ensemble de CTSS non-clos qui est $\preccurlyeq$-clos, de caractère fini et saturé.*

On s'intéresse à une classe particulière d'oracle décrite par le lemme suivant.

**Lemme 3.17.** *Il existe un oracle qui contient tous les CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos.*

*Démonstration.*
La preuve de ce lemme est donnée dans l'Annexe A. □

On va commencer la preuve de complétude en supposant qu'il n'existe pas de LTBI-preuve d'une formule $\phi$.

Montrer que $\phi$ n'est pas valide montrera, par l'absurde, que le calcul est complet. Soit $\mathcal{T}_0$ un tableau initial pour $\phi$. On a donc :

1. $\mathcal{T}_0 = [\langle\{\mathbb{F}\varphi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\}\rangle]$
2. $\mathcal{T}_0$ ne peut être clos (puisqu'il n'y a pas de preuve de $\phi$).

Par le Lemme 3.17, il existe un oracle qui contient tous les CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos. Soit $\mathcal{P}$ un tel oracle.

Par hypothèse, $\langle\{\mathbb{F}\varphi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\}\rangle \in \mathcal{P}$.

D'autre part, par la Proposition 3.5, il existe une stratégie équitable. On appelle $\mathcal{S}$ une telle stratégie. On note $S_i$ la $i^{\text{ème}}$ formule ou contrainte de $\mathcal{S}$.

On construit une suite $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle_{0\leqslant i}$ comme suit :

- $\langle \mathcal{F}_0, \mathcal{C}_{r0}, \mathcal{C}_{s0}\rangle = \langle\{\mathbb{F}\varphi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\}\rangle$
- Si $S_i$ est une formule étiquetée de la forme $\mathbb{S}F : (x,u)$, alors :
  - Si $\langle \mathcal{F}_i \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle \notin \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle$
  - Si $\langle \mathcal{F}_i \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle \in \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1}\rangle = \langle \mathcal{F}_i \cup \{\mathbb{S}F : (x,u)\} \cup F_e, \mathcal{C}_{ri} \cup \mathcal{C}_{re}, \mathcal{C}_{si} \cup \mathcal{C}_{se}\rangle$ tel que $F_e$, $\mathcal{C}_{re}$ et $\mathcal{C}_{se}$ sont déterminés par :

| $\mathbb{S}_i$ | $F$ | $F_e$ | $\mathcal{C}_{re}$ | $\mathcal{C}_{se}$ |
|---|---|---|---|---|
| $\mathbb{F}$ | $\phi \to \psi$ | $\{\mathbb{T}\phi : (a,u), \mathbb{F}\psi : (a,u)\}$ | $\{x \leqslant a\}$ | $\emptyset$ |
| $\mathbb{T}$ | $\phi * \psi$ | $\{\mathbb{T}\phi : (a,u), \mathbb{T}\psi : (b,u)\}$ | $\{ab \leqslant x\}$ | $\emptyset$ |
| $\mathbb{F}$ | $\phi \mathrel{-\!\!*} \psi$ | $\{\mathbb{T}\phi : (a,u), \mathbb{F}\psi : (xa,u)\}$ | $\{xa \leqslant xa\}$ | $\emptyset$ |
| $\mathbb{T}$ | I | $\emptyset$ | $\{1_r \leqslant x\}$ | $\emptyset$ |
| $\mathbb{T}$ | $\Diamond\phi$ | $\{\mathbb{T}\phi : (x,c)\}$ | $\emptyset$ | $\{u \lhd c\}$ |
| $\mathbb{F}$ | $\Box\phi$ | $\{\mathbb{F}\phi : (x,c)\}$ | $\emptyset$ | $\{u \lhd c\}$ |
| $\mathbb{T}$ | $\bigcirc\phi$ | $\{\mathbb{T}\phi : (x,c)\}$ | $\emptyset$ | $\{u \blacktriangleleft c\}$ |
| $\mathbb{F}$ | $\bigcirc\phi$ | $\{\mathbb{F}\phi : (x,c)\}$ | $\emptyset$ | $\{u \blacktriangleleft c\}$ |
| Sinon | | $\emptyset$ | $\emptyset$ | $\emptyset$ |

avec $a = c_{2i+1}$, $b = c_{2i+2}$ et $c = l_{i+2}$.

- Si $S_i$ est une contrainte d'états de la forme $u \star v$ :
  - Si $\{u,v\} \not\subseteq \{l_1, \ldots, l_{i+1}\}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle$

- Si $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\}\rangle \notin \mathcal{P}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle$
- Si $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\}\rangle \in \mathcal{P}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\}\rangle$

Cette suite de CTSS présente plusieurs propriétés.

**Proposition 3.6.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathbb{F}\varphi : (1_r, l_1) \in \mathcal{F}_i$, $1_r \leqslant 1_r \in \mathcal{C}_{ri}$ *et* $l_1 \lhd l_1 \in \mathcal{C}_{si}$
2. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$, $\mathcal{C}_{ri} \subseteq \mathcal{C}_{ri+1}$ *et* $\mathcal{C}_{si} \subseteq \mathcal{C}_{si+1}$
3. $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle_{0 \leqslant i} \in \mathcal{P}$
4. $\mathcal{A}_r(\mathcal{C}_{ri}) \subseteq \{1_r, c_1, c_2, ..., c_{2i}\}$
5. $\mathcal{A}_s(\mathcal{C}_{si}) \subseteq \{l_1, l_2, ..., l_{i+1}\}$

*Démonstration.*
La preuve de cette proposition est donnée dans l'Annexe A. □

On considère à présent le CTSS limite $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$ de la suite $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle_{0 \leqslant i}$ défini par :

$$\mathcal{F}_\infty = \bigcup_i \mathcal{F}_i \quad \text{et} \quad \mathcal{C}_{r\infty} = \bigcup_i \mathcal{C}_{ri} \quad \text{et} \quad \mathcal{C}_{s\infty} = \bigcup_i \mathcal{C}_{si}$$

**Proposition 3.7.** *Les propriétés suivantes sont vérifiées :*

1. $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle \in \mathcal{P}$
2. *Pour toute formule étiquetée* $\mathbb{S}\phi : (x,u)$, *si* $\langle \mathcal{F}_\infty \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle \in \mathcal{P}$ *alors* $\mathbb{S}\phi : (x,u) \in \mathcal{F}_\infty$
3. *Pour toute contrainte d'états* $u \star v$, *si* $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{u \star v\}\rangle \in \mathcal{P}$ *alors* $u \star v \in \mathcal{C}_{s\infty}$

*Démonstration.*
On prouve d'abord que $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$ est un CTSS. Soit $\mathbb{S}\phi : (x,u) \in \mathcal{F}_\infty$. Par construction, et par le point 2 de la Proposition 3.6, il existe $i$ tel que $\mathbb{S}\phi : (x,u) \in \mathcal{F}_i$. Par le point 3 de la Proposition 3.6, $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle \in \mathcal{P}$. Donc $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle$ est un CTSS et donc $x \leqslant x \in \overline{\mathcal{C}_{ri}}$ et $u \lhd u \in \overline{\mathcal{C}_{si}}$. Alors, par définition, $x \leqslant x \in \overline{\mathcal{C}_{r\infty}}$ et $u \lhd u \in \overline{\mathcal{C}_{s\infty}}$. Donc, $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$ est un CTSS.
On montre maintenant les propriétés de la proposition :

1. Soit $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$. Comme $\mathcal{F}_f$, $\mathcal{C}_{rf}$ et $\mathcal{C}_{sf}$ sont finis, et comme la suite $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si}\rangle_{0 \leqslant i}$ est croissante par le point 2 de la Proposition 3.6, il existe $j \in \mathbb{N}$ tel que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{rf}\rangle \preccurlyeq \langle \mathcal{F}_j, \mathcal{C}_{rj}, \mathcal{C}_{sj}\rangle$. Par le point 3 de la Proposition 3.6, $\langle \mathcal{F}_j, \mathcal{C}_{rj}, \mathcal{C}_{sj}\rangle \in \mathcal{P}$. Alors, comme $\mathcal{P}$ est $\preccurlyeq$-clos, $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{rf}\rangle \in \mathcal{P}$. Donc pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{rf}\rangle \preccurlyeq_f \langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{rf}\rangle \in \mathcal{P}$. Donc, comme $\mathcal{P}$ est de caractère fini, on a $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle \in \mathcal{P}$.
2. Soit $\mathbb{S}\phi : (x,u)$ tel que $\langle \mathcal{F}_\infty \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle \in \mathcal{P}$. Par définition $x \leqslant x \in \overline{\mathcal{C}_{r\infty}}$ et $u \lhd u \in \overline{\mathcal{C}_{s\infty}}$. Par compacité (Lemme 3.4), il existe $\mathcal{C}_{rf} \subseteq \mathcal{C}_{r\infty}$ et $\mathcal{C}_{sf} \subseteq \mathcal{C}_{s\infty}$ tels que $\mathcal{C}_{rf}$ et $\mathcal{C}_{sf}$ sont finis et $x \leqslant x \in \overline{\mathcal{C}_{rf}}$ et $u \lhd u \in \overline{\mathcal{C}_{sf}}$. Comme la suite et croissante, il existe $j \in \mathbb{N}$ tel que $\mathcal{C}_{rf} \subseteq \mathcal{C}_{rj}$ et $\mathcal{C}_{sf} \subseteq \mathcal{C}_{sj}$. Comme $\mathbb{S}\phi : (x,u)$ apparaît un nombre infini de fois dans notre stratégie équitable, il existe $k \geqslant j$ tel que $\mathcal{S}_k = \mathbb{S}\phi : (x,u)$. De plus, $\mathcal{C}_{rj} \subseteq \mathcal{C}_{rk}$ et $\mathcal{C}_{sj} \subseteq \mathcal{C}_{sk}$. Donc $x \leqslant x \in \overline{\mathcal{C}_{rk}}$ et $u \lhd u \in \overline{\mathcal{C}_{sk}}$. Comme $\langle \mathcal{F}_k \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{rk}, \mathcal{C}_{sk}\rangle$ est un CTSS, $\langle \mathcal{F}_k \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{rk}, \mathcal{C}_{sk}\rangle \preccurlyeq \langle \mathcal{F}_\infty \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle$, par définition du CTSS limite. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, alors $\langle \mathcal{F}_k \cup \{\mathbb{S}\phi : (x,u)\}, \mathcal{C}_{rk}, \mathcal{C}_{sk}\rangle \in \mathcal{P}$. Par construction de $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1}\rangle$, $\mathbb{S}\phi : (x,u) \in \mathcal{F}_{k+1}$. Donc $\mathbb{S}\phi : (x,u) \in \mathcal{F}_\infty$.

3. Soit $u \star v$ tel que $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{u \star v\}\rangle \in \mathcal{P}$. Soit $j = max\{i \in \mathbb{N} | l_i \in \{u, v\}\}$. Comme $u \star v$ apparaît un nombre infini de fois dans notre stratégie équitable, il existe $k \geq j$ tel que $\mathcal{S}_k = u \star v$. On a $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \cup \{u \star v\}\rangle \preccurlyeq \langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{u \star v\}\rangle$ par définition du CTSS limite. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \cup \{u \star v\}\rangle \in \mathcal{P}$. De plus, comme $L_s \cap \{u, v\} \subseteq \{l_1, \ldots, l_{k+1}\}$, par construction de $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1}\rangle$, on a $u \star v \in \mathcal{C}_{sk+1}$. Donc $u \star v \in \mathcal{C}_{s\infty}$.

□

Ces résultats permettent de conclure le résultat clé suivant :

**Lemme 3.18.** *Le CTSS limite est un CTSS de Hintikka.*

*Démonstration.*
Par la propriété 1 de la Proposition 3.7, $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty}\rangle \in \mathcal{P}$. On vérifie que toutes les conditions de la Définition 3.19.

La preuve complète est donnée dans l'Annexe A.

□

Enfin on peut conclure quant à la complétude :

**Théorème 3.3** (Complétude)**.** *Soit $\phi$ une formule. Si $\phi$ est valide, alors il existe une LTBI-preuve de $\phi$.*

*Démonstration.*
On suppose qu'il n'existe pas de LTBI-preuve de $\phi$. On montre que $\phi$ n'est pas valide. La méthode que nous avons présentée permet de construire un CTSS limite, qui est un CTSS de Hintikka par le Lemme 3.18. Par la propriété 1 de la Proposition 3.6, $\mathbb{F}\varphi : (1_r, l_1) \in \mathcal{F}_i$ pour tout $i$. Donc par construction, $\mathbb{F}\varphi : (1_r, l_1) \in \mathcal{F}_\infty$. Par le Lemme 3.16, $\phi$ n'est pas valide. □

Nous avons ainsi présenté une adaptation du calcul des tableaux pour DBI à notre logique LTBI ainsi que des preuves de correction et de complétude (sous réserve que la Conjecture 3.1 soit vérifiée). Ce calcul est suffisant pour travailler sur LTBI, mais si cette logique hérite de DBI, elle est aussi inspirée de la logique LTL qui possède ses propres calculs des tableaux. Dans la section suivante, nous envisageons un autre calcul qui adapte ceux de LTL à la logique LTBI.

## 3.6 Un deuxième calcul des tableaux pour LTBI

Jusqu'à présent, nous avons présenté un calcul des tableaux inspiré des travaux sur BBI, BI et DBI [34,49,66].

Toutefois, comme nous l'avons expliqué dans le début de ce chapitre, la logique LTBI émerge d'une double parenté, puisque sa dimension multiplicative est inspirée par BI tandis que sa dimension temporelle est puisée dans LTL. Or il se trouve que des calculs des tableaux ont également été développés pour LTL et ses variantes [65].

Contrairement à la famille des calculs des tableaux que nous avons explorée jusque maintenant, les tableaux de LTL ne font pas usage de labels. Au lieu des labels de vérité $\mathbb{T}$ et $\mathbb{F}$, ces tableaux utilisent deux ensembles de formules séparées, à la manière du calcul des séquents.

Quant aux opérateurs temporels, ces tableaux pour LTL les développe en utilisant les égalités que nous avions mentionnées dans le Lemme 3.3. On voit comment un tel remplacement crée des redondances, puisqu'on remplace une formule par une autre plus complexe contenant la formule d'origine. Toutefois, en autorisant les tableaux à être des graphes au lieu des arbres (c'est à dire avec des cycles), on parvient à maintenir le caractère fini des tableaux. Les conditions de clôture doivent être modifiées en conséquence. On verra dans le détail de la formalisation de ce calcul comment ce processus se déroule.

On se propose donc de développer, à partir de ces calculs pour LTL, un deuxième calcul des tableaux pour LTBI qui n'utilise qu'un seul ensemble de labels (pour gérer les ressources). On distinguera ce calcul du précédent en appelant ces nouveaux tableaux des *c-tableaux* (où le c signifie cyclique ou cyclic).

Il y a plusieurs avantages à développer le calcul des c-tableaux. Tout d'abord, il est intéressant de multiplier les approches de notre logique LTBI. On discutera brièvement à la fin de cette section de l'intérêt respectif de chaque calcul. Dans une optique plus générale, l'utilisation de calculs sans labels ouvre des questions sur les calculs des tableaux en général et la pertinence de l'utilisation de labels. Ce point sera discuté ici, et une réflexion supplémentaire sera donnée dans le Chapitre 5.

Enfin, le développement des c-tableaux ouvre des perspectives fascinantes étant donné le matériel de base dont ils s'inspirent. En effet, comme expliqué dans [65], les tableaux pour LTL présentent la particularité de pouvoir être transformés de manière systématique en modèle sous la forme d'automates de Büchi généralisés [52]. Une telle génération de modèles sous la forme d'automates est une ouverture très intéressante pour LTBI. En effet, dans l'esprit du "model checking", on peut vérifier des propriétés sur des systèmes beaucoup plus facilement en manipulant les automates de Büchi correspondant (l'acceptation d'un mot par un automate se décidant en temps linéaire [65],).

Les propriétés et les possibilités d'exploitation du calcul des c-tableaux pour LTBI n'ont pas pu être explorées complètement, aussi nous ne donnerons ici que les définitions nécessaires pour construire un tel tableau ainsi qu'un exemple et une preuve de correction, puis nous discuterons de possibles perspectives.

### 3.6.1 Le calcul des c-tableaux pour LTBI

Bien que le calcul des c-tableaux n'ait pas de labels de vérité et de labels d'états, comme exposé plus haut, nous avons toutefois besoin d'utiliser des labels de ressources et les contraintes associées, comme dans le calcul des tableaux pour LTBI.

On reprend donc la Définition 3.6 ainsi que la notion de sous-label exposée juste après, la Définition 3.8 pour la partie qui concerne les labels de ressources et la Définition 3.9. La Proposition 3.1, le Corollaire 3.1 ainsi que la Proposition 3.2 et le Lemme 3.4 pour ce qui concerne les contraintes de ressources restent tous valides.

On redéfinit la notion de formule étiquetée ainsi qu'un nouveau type de structures, analogue aux CTSS, les triplets positifs-négatifs ou TPN.

**Définition 3.25** (Formule étiquetée simple / TPN)**.** *Une* formule étiquetée simple *est un couple* $(\phi, x) \in \mathcal{L} \times L_r$ *qu'on écrit* $\phi : x$.

*Un* triplet positif/négatif *(noté TPN) est un triplet* $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$*, où* $\mathcal{F}^+$ *et* $\mathcal{F}^-$ *sont deux ensembles de formules étiquetées simples et* $\mathcal{C}_r$ *est un ensemble de contraintes de ressources, tels que la propriété suivante, appelée* $(P_{TPN})$*, est vérifiée : si* $\phi : x \in \mathcal{F}^+ \cup \mathcal{F}^-$ *alors* $x \leqslant x \in \overline{\mathcal{C}_r}$*.*
*Le TPN vide* $\langle \{\}, \{\}, \{\} \rangle$ *est appelé* TPN absurde*, on le note* ■*.*

Un TPN $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ est la représentation d'un nœud de notre tableau dans laquelle les formules positives sont les formules étiquetées de $\mathcal{F}^+$, les formules négatives sont les formules étiquetées de $\mathcal{F}^-$ et les contraintes de labels sont les éléments de $\mathcal{C}_r$.

Nos formules étiquetées ont bien entendu perdu leurs labels d'états (c'est tout l'objectif du calcul) et également leur signe puisque le calcul manipulera deux ensembles de formules.

Notons dès à présent que les TPN n'ont plus la même signification que les CTSS vis-à-vis du tableau final : les CTSS représentaient des branches du tableau (qui était un arbre), les TPN sont des nœuds du c-tableau qui est un graphe.

Le TPN absurde est un outil pour simplifier la clôture des tableaux : c'est simplement un marqueur pour les branches closes.

**Définition 3.26** (Label inconsistant)**.** *Soit* $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ *un TPN et x un label de ressources. x est* inconsistant *s'il existe deux labels de ressources y et z tels que* $yz \leqslant x \in \overline{\mathcal{C}_r}$ *et* $\bot : y \in \mathcal{F}^+$*. Un label est* consistant *s'il n'est pas inconsistant.*

La notion de label inconsistant est analogue à celle du calcul des tableau. On a d'ailleurs les mêmes propriétés :

**Proposition 3.8.** *Soit* $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ *un TPN. On a les propriétés suivantes :*

1. *Si* $y \leqslant x \in \overline{\mathcal{C}_r}$ *et x est un label consistant, alors y est un label consistant.*
2. *Si* $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ *est un label consistant alors x et y sont des labels consistants.*

*Démonstration.*

1. On suppose que $y \leqslant x \in \overline{\mathcal{C}_r}$ et que $x$ est un label consistant. On suppose que $y$ n'est pas consistant. Alors il existe deux labels $z$ et $t$ tels que $zt \leqslant y \in \overline{\mathcal{C}_r}$ et $\bot : z \in \mathcal{F}^+$. Alors par la règle $\langle t_r \rangle$ on a $zt \leqslant x \in \overline{\mathcal{C}_r}$. Donc $x$ est inconsistant. C'est absurde. Donc $y$ est consistant.
2. On suppose que $xy \in \mathcal{D}_r(\overline{\mathcal{C}_r})$ est un label consistant. Supposons que $x$ est inconsistant. il existe deux labels $z$ et $t$ tels que $zt \leqslant x \in \overline{\mathcal{C}_r}$ et $\bot : z \in \mathcal{F}^+$. Par le Corollaire 3.1, on a $xy \leqslant xy \in \overline{\mathcal{C}_r}$. Par la règle $\langle c_r \rangle$, on a alors $zty \leqslant xy \in \overline{\mathcal{C}_r}$. Alors $xy$ est inconsistant, ce qui est absurde. Donc $x$ est consistant. On raisonne de même pour $y$ (le problème est symétrique).

□

Nous pouvons maintenant énoncer les règles de notre calcul. La Figure 3.12 présente les règles pour le calcul des c-tableaux de LTBI.

Parmi ces règles, les règles additives et multiplicatives sont essentiellement identiques à celle du calcul précédent (voir Figure 3.4) avec la différence de la présentation par TPN. Les règles de clôture traduisent les conditions de clôture, également identiques au calcul précédent (voir Définition 3.16). Les règles pour $\Diamond$ et $\Box$ ne font que retranscrire les égalités montrées dans le Lemme 3.3.

**Règles de clôture**

$$\frac{\phi : x \in \mathcal{F}^- \text{ et } x \text{ est inconsistant}}{\blacksquare}\ \langle\bot\rangle \qquad \frac{\top : x \in \mathcal{F}^-}{\blacksquare}\ \langle\top\rangle$$

$$\frac{\mathrm{I} : x \in \mathcal{F}^- \text{ et } 1 \leqslant x \in \overline{\mathcal{C}_r}}{\blacksquare}\ \langle\mathrm{I}^-\rangle \qquad \frac{A : x \in \mathcal{F}^+ \text{ et } A : y \in \mathcal{F}^- \text{ et } x \leqslant y \in \overline{\mathcal{C}_r}}{\blacksquare}\ \langle\blacksquare\rangle$$

**Règles additives**

$$\frac{A \wedge B : x \in \mathcal{F}^+}{\langle \{A : x, B : x\}, \emptyset, \emptyset \rangle}\ \langle\wedge^+\rangle \qquad \frac{A \wedge B : x \in \mathcal{F}^-}{\langle \emptyset, \{A : x\}, \emptyset \rangle \mid \langle \emptyset, \{B : x\}, \emptyset \rangle}\ \langle\wedge^-\rangle$$

$$\frac{A \vee B : x \in \mathcal{F}^+}{\langle \{A : x\}, \emptyset, \emptyset \rangle \mid \langle \{B : x\}, \emptyset, \emptyset \rangle}\ \langle\vee^+\rangle \qquad \frac{A \vee B : x \in \mathcal{F}^-}{\langle \emptyset, \{A : x, B : x\}, \emptyset \rangle}\ \langle\vee^-\rangle$$

$$\frac{A \to B : x \in \mathcal{F}^+ \text{ et } x \leqslant y \in \overline{\mathcal{C}_r}}{\langle \emptyset, \{A : y\}, \emptyset \rangle \mid \langle \{B : y\}, \emptyset, \emptyset \rangle}\ \langle\to^+\rangle \qquad \frac{A \to B : x \in \mathcal{F}^-}{\langle \{A : c_i\}, \{B : c_i\}, \{x \leqslant c_i\} \rangle}\ \langle\to^-\rangle$$

**Règle multiplicatives**

$$\frac{\mathrm{I} : x \in \mathcal{F}^+}{\langle \emptyset, \emptyset, \{1_r \leqslant x\} \rangle}\ \langle\mathrm{I}^+\rangle$$

$$\frac{A * B : x \in \mathcal{F}^+}{\langle \{A : c_i, B : c_j\}, \emptyset, \{c_i c_j \leqslant x\} \rangle}\ \langle*^+\rangle \qquad \frac{A * B : x \in \mathcal{F}^- \text{ et } yz \leqslant x \in \overline{\mathcal{C}_r}}{\langle \emptyset, \{A : y\}, \emptyset \rangle \mid \langle \emptyset, \{B : z\}, \emptyset \rangle}\ \langle*^-\rangle$$

$$\frac{A \mathrel{-\!\!*} B : x \in \mathcal{F}^+ \text{ et } xy \leqslant xy \in \overline{\mathcal{C}_r}}{\langle \emptyset, \{A : y\}, \emptyset \rangle \mid \langle \{B : xy\}, \emptyset, \emptyset \rangle}\ \langle\mathrel{-\!\!*}^+\rangle \qquad \frac{A \mathrel{-\!\!*} B : x \in \mathcal{F}^-}{\langle \{A : c_i\}, \{B : xc_i\}, \{xc_i \leqslant xc_i\} \rangle}\ \langle\mathrel{-\!\!*}^-\rangle$$

**Règles temporelles**

$$\frac{\Box A : x \in \mathcal{F}^+}{\langle \{A : x, \bigcirc\Box A : x\}, \emptyset, \emptyset \rangle}\ \langle\Box^+\rangle \qquad \frac{\Box A : x \in \mathcal{F}^-}{\langle \emptyset, \{A : x\}, \emptyset \rangle \mid \langle \emptyset, \{\bigcirc\Box A : x\}, \emptyset \rangle}\ \langle\Box^-\rangle$$

$$\frac{\Diamond A : x \in \mathcal{F}^+}{\langle \{A : x\}, \emptyset, \emptyset \rangle \mid \langle \{\bigcirc\Diamond A : x\}, \emptyset, \emptyset \rangle}\ \langle\Diamond^+\rangle \qquad \frac{\Diamond A : x \in \mathcal{F}^-}{\langle \emptyset, \{A : x, \bigcirc\Diamond A : x\}, \emptyset \rangle}\ \langle\Diamond^-\rangle$$

$$\frac{\mathcal{F}^+ = \bigcup\{\bigcirc A_i : x_i\} \cup \bigcup\{p_j : x_j\} \text{ et } \mathcal{F}^- = \bigcup\{\bigcirc B_i : x_i\} \cup \bigcup\{q_j : x_j\}}{\langle \bigcup\{A_i : x_i\}, \bigcup\{B_i : x_i\}, \emptyset \rangle}\ \langle\bigcirc\rangle$$

Note : $c_i$ et $c_j$ sont des nouvelles constantes.
Les $p_j$ et les $q_j$ sont des formules atomiques.

FIGURE 3.12 – Règles des c-tableaux pour LTBI

Enfin, la règle pour $\circ$ peut paraître complexe dans sa présentation mais est en réalité très simple à appliquer. La condition de la règle précise qu'elle ne peut être appliquée que si les ensembles $\mathcal{F}^+$ et $\mathcal{F}^-$ sont composés exclusivement de formules atomiques et de formules $\circ$. Le TPN résultant correspond à l'effacement de toutes les formules atomiques et tous les symboles $\circ$. De manière informelle, cette règle consiste à passer à l'état suivant : toutes les propositions valides à l'état précédent sont ignorées, et les formules annoncées par $\circ$ sont développées.

Formalisons maintenant ce qu'est un c-tableau.

**Définition 3.27** (LTBI-c-tableau)**.** *Un* LTBI-c-tableau *pour un TPN fini* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$ *est un graphe* $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ *construit par induction en suivant les règles suivantes :*

1. *Le graphe* $(\{\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle\}, \emptyset)$ *est un LTBI-c-tableau pour* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$
2. *Si le graphe* $(\mathcal{V}, \mathcal{E})$*, tel que* $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle \in \mathcal{V}$ *est un LTBI-c-tableau pour* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$ *et*

$$\frac{cond(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle)}{\blacksquare}$$

*est une instance d'une des règles de la Figure 3.12 tel que* $cond(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle)$ *est vérifiée, alors le graphe* $(\mathcal{V}', \mathcal{E}')$ *est un LTBI-c-tableau pour* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$*, avec :*

$$\mathcal{V}' = \mathcal{V} \cup \{\blacksquare\}$$

$$\mathcal{E}' = \mathcal{E} \cup \{(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle, \blacksquare) \mid 1 \leq i \leq k\}$$

3. *Si le graphe* $(\mathcal{V}, \mathcal{E})$*, tel que* $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle \in \mathcal{V}$ *est un LTBI-c-tableau pour* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$ *et*

$$\frac{cond(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle)}{\langle \mathcal{F}_1^+, \mathcal{F}_1^-, \mathcal{C}_{r_1} \rangle \mid ... \mid \langle \mathcal{F}_k^+, \mathcal{F}_k^-, \mathcal{C}_{r_k} \rangle}$$

*est une instance d'une des règles de la Figure 3.12 tel que* $cond(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle)$ *est vérifiée, on définit un ensemble de TPN* $(\mathcal{C}_i)_{1 \leq i \leq k}$ *par :*

$$\mathcal{C}_i = \langle \mathcal{F}^+ \cup \mathcal{F}_i^+ \setminus \{F^+\}, \mathcal{F}^- \cup \mathcal{F}_i^- \setminus \{F^-\}, \mathcal{C}_r \cup \mathcal{C}_{r_i} \rangle$$

*où* $F^+$ *est toute formule apparaissant dans une condition de type* $F^+ \in \mathcal{F}^+$ *et* $F^-$ *toute formule apparaissant dans une condition de type* $F^- \in \mathcal{F}^-$ [5]*.*
*Alors le graphe* $(\mathcal{V}', \mathcal{E}')$ *est un LTBI-c-tableau pour* $\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$*, avec :*

$$\mathcal{V}' = \mathcal{V} \cup \{\mathcal{C}_i \mid 1 \leq i \leq k\}$$

$$\mathcal{E}' = \mathcal{E} \cup \{(\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle, \mathcal{C}_i)\}$$

$\langle \mathcal{F}_0^+, \mathcal{F}_0^-, \mathcal{C}_{r_0} \rangle$ *est appelée la* racine *de* $\mathcal{T}$*.*
*Un LTBI-c-tableau pour une formule* $\phi$ *est un LTBI-tableau pour :*

$$\langle \emptyset, \{\phi : 1_r\}, \{1_r \leqslant 1_r\} \rangle$$

5. Dans le cas de la règle $\langle \circ \rangle$, toutes les formules du TPN de la prémisse sont donc supprimées puisque l'ensemble de $\mathcal{F}^+$ et $\mathcal{F}^-$ sont spécifiés par la condition de la règle.

Cette définition emprunte au vocabulaire des graphes puisque nos tableaux sont maintenant des graphes. $\mathcal{V}$ est donc un ensemble de TPN qui constitue les nœuds du c-tableaux et $\mathcal{E}$ est un ensemble de couples de TPN qui constituent les arcs. De manière intuitive, ce formalisme signifie que pour chaque règle les TPN correspondant aux conclusions deviennent les fils (dans le graphe) du TPN de la prémisse.

À chaque application d'une règle, on génère ainsi un nouveau nœud pour chaque conclusion de la règle, en supprimant de $\mathcal{F}^+$ ou $\mathcal{F}^-$ la formule utilisée dans la prémisse et en rajoutant les formules (et les contraintes) présentes dans la conclusion. Ce nouveau nœud devient alors un fils du nœud courant dans le graphe.

On note que comme $\mathcal{V}$ est un ensemble, si un TPN déjà existant doit être ajouté, il ne l'est pas et seul l'arc correspondant l'est. Ceci permet la création des fameux cycles que nous avons mentionnés dans l'introduction de cette section.

Remarquons enfin que dans toutes ces règles, la propriété $(P_{TPN})$ de la Définition 3.25 est conservée.

Comme pour le calcul des tableaux précédent, il convient de respecter un ordre dans l'application des règles :

1. Il est préférable d'appliquer en premier lieu les règles de clôture ($\langle \mathrm{I}^- \rangle$, $\langle \top \rangle$, $\langle \bot \rangle$ et $\langle \blacksquare \rangle$). Cela n'est pas une obligation formelle, mais cela garantit d'avoir le tableau le plus direct possible.
2. Nous devons d'abord utiliser les règles qui introduisent des contraintes de ressources ($\langle \rightarrow^- \rangle$, $\langle -\!\!* ^- \rangle$ et $\langle *^+ \rangle$) afin de pouvoir les consommer plus tard.
3. On appliquera ensuite les règles qui ont des conditions sur les contraintes ($\langle \rightarrow^+ \rangle$, $\langle -\!\!* ^+ \rangle$, $\langle *^- \rangle$ et $\langle \mathrm{I}^+ \rangle$)
4. Enfin, il est *impératif* d'appliquer la règle $\langle \bigcirc \rangle$ en dernier, comme cela est impliqué par la condition qui lui est associée (il ne doit rester que des formules propositionnelles en dehors des formules $\bigcirc$).

**Définition 3.28** (Chemin)**.** *Soit $C$ et $C'$ deux TPN non absurdes d'un c-tableau $\mathcal{T} = (\mathcal{V}, \mathcal{E})$. On appelle chemin entre $C$ et $C'$ une suite de TPN $\{C_0, \ldots, C_n\}$ telle que :*

- $C_0 = C$
- $C_n = C'$
- $\forall i,\ 0 \leq i \leq n-1,\ C_i \in \mathcal{V}$
- $\forall i,\ 0 \leq i \leq n-1,\ (C_i, C_{i+1}) \in \mathcal{E}$

Cette notion de chemin, assez naturelle vu la nature de graphe des c-tableau, permet de définir les c-tableaux clos.

**Définition 3.29** (Conditions de clôture)**.** *On dit qu'un TPN $C = \langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ d'un c-tableau $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ est* clos *dans les cas suivants :*

1. *$C = \blacksquare$*
2. *Pour tout $C_i$ tel que $(C, C_i) \in \mathcal{E}$, $C_i$ est clos (tous les fils de $C$ sont clos)*
3. *Si $A$ est une formule telle que $\Diamond A : x \in \mathcal{F}^+$ et que pour tout nœud $C' = \langle \mathcal{F}^{+\prime}, \mathcal{F}^{-\prime}, \mathcal{C}'_r \rangle$ tel que $A : x \in \mathcal{F}^{+\prime}$ tout chemin entre $C$ et $C'$ contient un TPN clos, alors $C$ est clos.*

4. *Si A est une formule telle que* $\Box A : x \in \mathcal{F}^-$ *et que pour tout nœud* $C' = \langle \mathcal{F}^{+\prime}, \mathcal{F}^{-\prime}, \mathcal{C}_r' \rangle$ *tel que* $A : x \in \mathcal{F}^{-\prime}$ *tout chemin entre C et* $C'$ *contient un TPN clos, alors C est clos.*

*Un LTBI-c-tableau est clos si sa racine est close.*

Parmi ces conditions, la condition 1 reprend les conditions de clôture de la Définition 3.16 qui sont reflétée dans les règles de clôture de la Figure 3.12. La condition 2 permet de clore un c-tableau. On a bien une correspondance intuitive avec le calcul précédent. Enfin, les conditions 3 et 4 gèrent les clôtures dues aux opérateurs temporels. Par exemple, $\Diamond A$ promet la réalisation de $A$. Si sur le chemin entre $\Diamond A$ et $A$ on rencontre un nœud clos, alors ce chemin ne peut être emprunté sans rencontrer d'absurdité et le nœud contenant $\Diamond A$ est lui-même clos.

**Définition 3.30** (LTBI-c-preuve)**.** *Une* LTBI-c-preuve *pour une formule* $\phi$ *est un LTBI-c-tableau pour* $\phi$ *qui est clos.*

On représentera les tableaux de ce calcul comme des graphes, traditionnellement avec la racine en haut du graphe. Pour des questions de commodité, on représentera plusieurs nœuds ■, un pour chaque règle où il apparaît.

### 3.6.2 Un exemple de c-tableaux pour LTBI

Pour illustrer ce seconde calcul, on se propose de dresser le tableau pour la formule $(\Diamond \circ A * \Box B) \mathrel{-\!\!*} \Diamond (A * B)$.

Avant de considérer le c-tableau pour cette formule, nous allons en dresser le tableau avec le calcul précédente à titre de comparaison. La Figure 3.13 présente le LTBI-tableau pour notre formule. On observe que ce tableau est clos dans ses deux branches, on a donc une preuve de la formule. Ainsi, le calcul des c-tableaux devrait nous permettre également de dresser une preuve.

En nous référant à la Définition 3.27, nous savons que notre c-tableau pour $\Diamond \circ A * \Box B) \mathrel{-\!\!*} \Diamond (A * B)$ doit commencer par le TPN suivant (qui sera la racine du c-tableau) :

$$\langle \emptyset, \{(\Diamond \circ A * \Box B) \mathrel{-\!\!*} \Diamond (A * B) : 1_r\}, \{1_r \leqslant 1_r\} \rangle$$

Tout comme dans le calcul précédent, nous allons représenter nos c-tableaux différemment pour gagner en lisibilité. Tout d'abord, nous allons séparer les ensembles de formules $\mathcal{F}^+$ et $\mathcal{F}^-$ des contraintes de ressources $\mathcal{C}_r$. En outre, nous allons numéroter chaque nœud de notre c-tableau par un entier . Notre c-tableau a donc l'aspect suivant :

On doit ensuite développer ce TPN racine en utilisant les règles de la Figure 3.12. Pour le moment, nous n'avons pas le choix et devons utiliser la règle $\langle \mathrel{-\!\!*}^- \rangle$. Cela crée un nouveau TPN dans lequel la formule $\mathrel{-\!\!*}$ est supprimée et deux nouvelles formules sont introduites. En outre, l'ensemble $\mathcal{C}_r$ est mis à jour. Encore une fois, pour des raisons de commodité, nous ne représenterons que les mises à jour de $\mathcal{C}_r$ et non l'ensemble entier à chaque étape. On obtient donc le graphe à deux sommets suivant :

On continue ainsi d'appliquer les règles de la Figure 3.12 jusqu'à obtenir le c-tableau illustré dans la Figure 3.14. À chaque étape, on souligne la formule sur laquelle la règle est appliquée.

Observons maintenant en détail les différentes étapes de cette construction :

1. On choisit de développer d'abord la formule $\Diamond \circ A * \Box B$ par la règle $\langle *^+ \rangle$. Cela génère un nouveau sommet avec deux nouvelles formules, ainsi qu'une nouvelle contrainte de ressources.

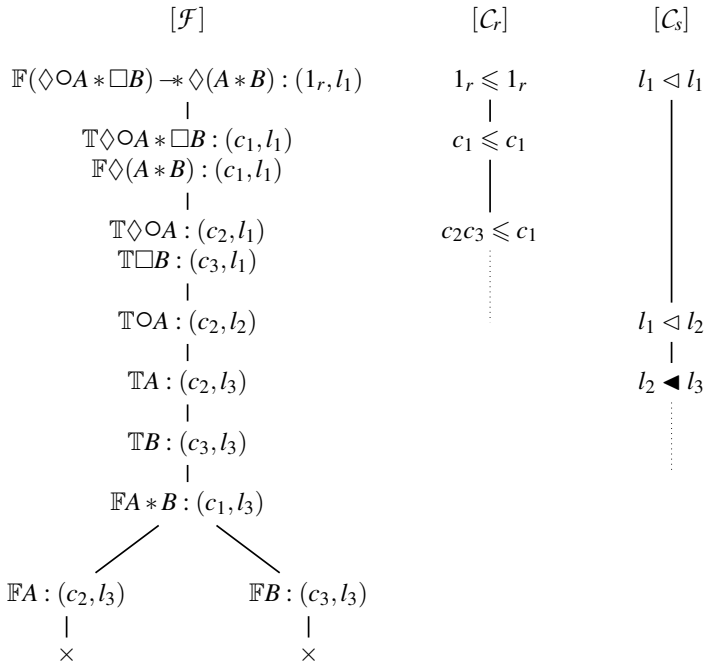



FIGURE 3.13 – LTBI-preuve pour $(\Diamond\circ A * \Box B) \multimap\!\!* \Diamond(A * B)$

$[n : \mathcal{F}^+\mathcal{F}^-]$ $\qquad$ $[n : \mathcal{C}_r]$

$0 : \{\}\{\underline{(\Diamond\circ A * \Box B) \multimap\!\!* \Diamond(A * B) : 1_r}\}$ $\qquad$ $0 : 1_r \leqslant 1_r$

2. On décompose ensuite la formule $\Diamond(A * B)$ par la règle $\langle\Diamond^-\rangle$. Cela génère un nouveau sommet avec deux nouvelles formules.
3. On décompose ensuite la formule $\Box B$ par la règle $\langle\Box^+\rangle$. Cela génère un nouveau sommet avec deux nouvelles formules.
4. On choisit ensuite de décomposer la formule $A * B$. Cela nécessite de trouver deux labels de ressources $x$,$y$ tels que $xy \leqslant c_1 \in \overline{\mathcal{C}_r}$. Les labels $c_2$ et $c_3$ conviennent. Un peu d'intuition nous indique qu'il est plus malin d'associer $c_2$ à $A$ et $c_3$ à $B$. On génère alors deux nouvelles branches, les nœuds 5 et 7, qui contiennent chacun une nouvelle formule.
5. Sur ce sommet, on peut appliquer la règle $\langle\blacksquare\rangle$. Cela génère un TPN absurde.
6. Cette branche est close.
7. Dans cette branche, on peut encore décomposer la formule $\Diamond\circ A$ avec la règle $\langle\Diamond^+\rangle$. On crée à nouveau deux branches, les nœuds 8 et 9, qui contiennent chacun une nouvelle formule.
8. Dans ce nœud, toutes les formules sont soit des formules atomiques propositionnelles, soit des formules $\circ$. On peut donc appliquer la règle $\langle\circ\rangle$. Toutes les formules atomiques sont éliminées, et les formules $\circ$ sont débarrassées de leur opérateur. Le nœud obtenu est exactement le nœud 2. On ne crée donc pas de nouveau nœud, mais on crée un arc qui relie le nœud 8 au nœud 2.

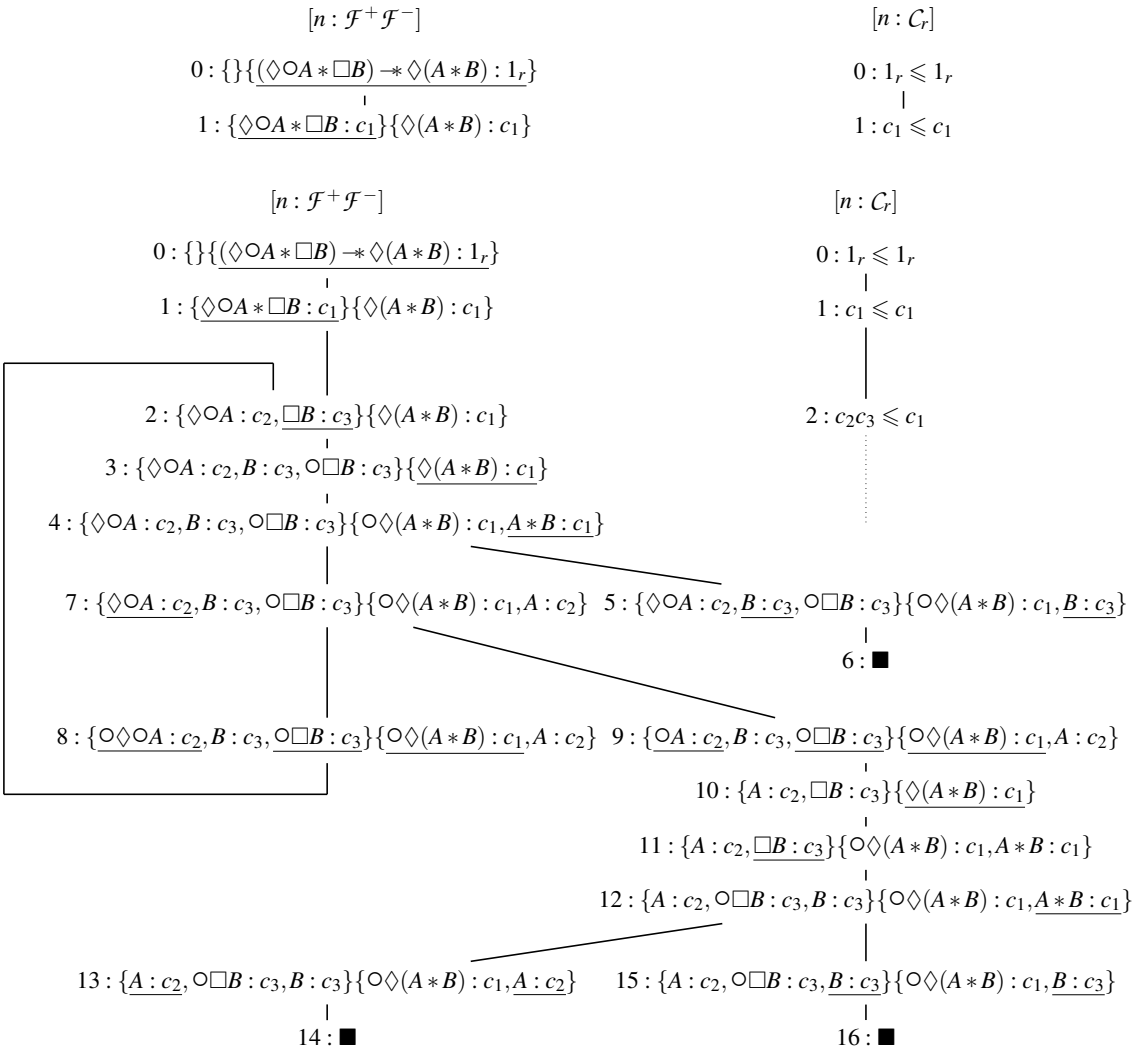



FIGURE 3.14 – LTBI-c-preuve pour $(\Diamond\bigcirc A * \Box B) \multimap \Diamond(A * B)$

9. Encore une fois, on se trouve dans une situation qui permet d'appliquer la règle $\langle\bigcirc\rangle$, ce qui génère un nouveau sommet avec trois nouvelles formules.
10. On développe à nouveau $\Diamond(A * B)$ par la règle $\langle\Diamond^-\rangle$. Cela génère un nouveau sommet avec deux nouvelles formules.
11. On décompose la formule $\Box B$ par la règle $\langle\Box^+\rangle$. Cela génère un nouveau sommet avec deux nouvelles formules.
12. On décompose la formule $A * B$. On choisit à nouveau les labels $c_2$ et $c_3$. On génère deux nouvelles branches, les nœuds 13 et 15, qui contiennent chacun une nouvelle formule.
13. On peut appliquer la règle $\langle\blacksquare\rangle$ avec $A : c_2$, cela génère un TPN absurde.
14. Cette branche est close.
15. On peut appliquer la règle $\langle\blacksquare\rangle$ avec $B : c_3$, cela génère un TPN absurde.
16. Cette branche est close.

On ne peut plus développer le c-tableau d'avantage. Contrairement au calcul des tableaux précédent, cela ne finit pas notre recherche, toutefois, car nous devons encore déterminer si le c-tableau est clos.

On se réfère à la Définition 3.29.

- Les nœuds 6, 14 et 16 sont clos puisqu'ils sont des nœuds de type ■ (point 1 de la définition).
- Par le point 3 de la définition, les nœuds 5, 13 et 15 sont aussi clos (tous leurs fils sont clos).
- Par le même point, le nœud 12 qui a comme fils les nœuds 13 et 15, est clos.
- Toujours par ce point, on peut successivement déterminer que les nœuds 11, 10, puis 9 sont clos.
- Pour appliquer les points 4 et 5 de la définition, on recherche des formules de type $\Diamond\phi$ dans $\mathcal{F}^+$ ou $\Box\psi$ dans $\mathcal{F}^-$. On considère la formule $\Diamond\bigcirc A : c_2$ qui est présente dans l'ensemble $\mathcal{F}^+$ du nœud 2. D'après la condition de clôture, on recherche les nœuds tels que $\bigcirc A : c_2 \in \mathcal{F}^+$. Le nœud 9 est le seul qui convient. Or ce nœud est clos d'après le point précédent. Donc le chemin entre le nœud 2 et le nœud 9 contient un nœud clos. Alors le nœud 2 est clos.
- Par le point 3 de la règle, le nœud 1, puis le nœud 0 sont clos.
- La racine du c-tableau étant close, le c-tableau est clos, et on a bien un c-preuve de $\Diamond\bigcirc A * \Box B) \multimap \Diamond(A * B)$.

### 3.6.3 Correction du calcul des c-tableaux

On se propose de prouver la correction du calcul des c-tableaux pour LTBI par une approche similaire à celle utilisée pour le calcul des tableaux dans la section 3.5.1. On va introduire la notion de réalisabilité pour les c-tableaux puis démontrer les lemmes nécessaires à la preuve de correction.

**Définition 3.31** (Réalisation des TPN)**.** *Soit $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ un TPN. Une* réalisation *de ce CTSS est un triplet $(\mathcal{K}, \lfloor . \rfloor, s)$ tel que $\mathcal{K} = (\mathcal{M}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est un modèle linéaire de ressources avec $\mathcal{M} = (R, \bullet, e, \pi, \sqsubseteq, S)$, $\lfloor . \rfloor : \mathcal{D}_r(\overline{\mathcal{C}_r}) \to R$ et $s \in S$, tels que :*

- $\lfloor 1_r \rfloor = e$
- $\lfloor x \circ y \rfloor = \lfloor x \rfloor \bullet \lfloor y \rfloor$
- *Si $\phi : x \in \mathcal{F}^+$, alors $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \phi$*
- *Si $\phi : x \in \mathcal{F}^-$, alors $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} \phi$*
- *Si $x \leqslant y \in \mathcal{C}_r$, alors $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$*

*On dit qu'un TPN est* réalisable *s'il en existe une réalisation.*

**Lemme 3.19.** *Soit $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ un TPN et $\mathfrak{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ une réalisation de ce TPN. $\mathfrak{R}$ est aussi une réalisation de $\langle \mathcal{F}^+, \mathcal{F}^-, \overline{\mathcal{C}_r} \rangle$. En d'autres mots, pour tout $x \leqslant y \in \overline{\mathcal{C}_r}$, $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$.*

*Démonstration.*
La preuve est identique à celle du Lemme 3.6. □

**Lemme 3.20.** *Les TPN ayant pour fils le TPN absurde (■) ne sont pas réalisables.*

*Démonstration.*
Soit $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ ayant pour fils le TPN absurde (■). Alors on a appliqué sur $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ une des règles de clôture de la Figure 3.12. On suppose que $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ est réalisable et on note $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ sa réalisation. On discute selon la règle appliquée sur le TPN pour produire ■ :

- $\langle \mathrm{I}^- \rangle$
  Dans ce cas $\mathrm{I} : x \in \mathcal{F}^-$ et $1_r \leqslant x \in \overline{\mathcal{C}_r}$. Alors par définition, et par le Lemme 3.21 on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \mathrm{I}$ et $e \sqsubseteq \lfloor x \rfloor$. C'est absurde.
- $\langle \bot \rangle$
  Dans ce cas $\phi : x \in \mathcal{F}^-$ et $x$ est inconsistant. Comme $x$ est inconsistant, il existe $y$ et $z$ tels que $yz \leqslant x \in \overline{\mathcal{C}_r}$ et $\bot : y \in \mathcal{F}^+$. Alors par définition et par le Lemme 3.19, on a $\lfloor y \rfloor \bullet \lfloor z \rfloor \sqsubseteq \lfloor x \rfloor$ et $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} \bot$. Alors par la Définition 3.4, on a $\pi \sqsubseteq \lfloor y \rfloor$. Par compatibilité, on a alors $\pi \sqsubseteq \lfloor y \rfloor \bullet \lfloor z \rfloor$ et par transitivité $\pi \sqsubseteq \lfloor x \rfloor$. Alors par inconsistance (Lemme 3.2), on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \phi$. C'est absurde.
- $\langle \top \rangle$
  Dans ce cas $\top : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \top$. C'est absurde.
- $\langle ■ \rangle$
  Dans ce cas $A : x \in \mathcal{F}^+$,$A : y \in \mathcal{F}^-$ et $x \leqslant y \in \overline{\mathcal{C}_r}$. Alors par définition, et par le Lemme 3.21, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A$, $(\lfloor y \rfloor, s) \nVdash_{\mathcal{K}} B$ et $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$. Alors par monotonie (Lemme 3.1) on a aussi $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} A$. C'est absurde.

Tous les cas sont absurdes, donc $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ ne peut pas être réalisable. □

**Lemme 3.21.** *Soit $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ un c-tableau. Soit $C \in \mathcal{V}$ un TPN réalisable. Parmi tous les TPN $C'$ tels que $(C, C') \in \mathcal{E}$ (c'est à dire les fils de C), au moins un est réalisable et le modèle de sa réalisation étend celui de C.*

*Démonstration.*
La preuve de ce lemme est donnée dans l'Annexe A. □

**Définition 3.32** (Chemin réalisable)**.** *Soit $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ un c-tableau complètement développé. Soit C un TPN réalisable de $\mathcal{T}$. On appelle* chemin réalisable *partant de C une suite, potentiellement infinie, $(C_0, C, \ldots)$ de TPN de $\mathcal{T}$ construit comme suit :*

- *$C_0 = C$ ;*
- *Si $C_i$ a pour fils ■ dans $\mathcal{T}$, la suite est finie et $C_i$ est son dernier élément ;*
- *Si $C_i$ n'a qu'un fils $C'$, alors $C_{i+1} = C'$ ;*
- *Si $C_i$ a deux fils, d'après le Lemme 3.21, un des deux, $C'$, est réalisable, et on pose $C_{i+1} = C'$.*

**Lemme 3.22.** *Soit $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ un c-tableau complètement développé. Soit C un TPN réalisable de $\mathcal{T}$. Soit $(C_i)$ le chemin réalisable partant de C.*

1. *$(C_i)$ est infinie ;*

2. *Tous les éléments de* $(C_i)$ *sont réalisables ;*
3. *Pour tout i, si* $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ *est une réalisation de* $C_i$ *et* $\mathcal{R}' = (\mathcal{K}', \lfloor . \rfloor', s')$ *une réalisation de* $C_{i+1}$*, alors* $s' = s$ *ou* $s' = s+1$.
4. *Il existe une infinité d'applications de la règle* $\langle \bigcirc \rangle$ *dans* $(C_i)$.
5. *Si* $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ *est une réalisation de* $C_0$*, pour tout* $k \in \mathbb{N}$*, il existe au moins un* $C_i$ *tel que* $\mathcal{R}' = (\mathcal{K}', \lfloor . \rfloor', s+k)$ *soit une réalisation de* $C_i$.

*Démonstration.*

1.-2. On montre par récurrence sur les éléments de $(C_i)$ que tous les éléments de $(C_i)$ sont réalisable et qu'il existe toujours un élément suivant.

- Cas de base
  Comme $C$ est réalisable, $C_0 = C$ est réalisable.
- Hérédité
  On suppose que $C_k$ est réalisable. Supposons que $C_{k+1}$ n'existe pas. Alors par définition, $C_k$ a pour fils ■. Par le Lemme 3.20, $C_k$ n'est pas réalisable. C'est absurde, donc $C_{k+1}$ existe. Il y a deux cas : si $C_k$ n'a qu'un fils, $C_{k+1}$ est ce fils, et par le Lemme 3.21, il est réalisable. Si $C_k$ a deux fils, $C_{k+1}$ est un fils réalisable par définition.

Ainsi tous les éléments de $(C_i)$ sont réalisables, et chacun d'eux a un successeur, donc $(C_i)$ est infinie.

3. En reprenant les arguments de la preuve du Lemme 3.21, on a que $s' = s+1$ si la règle $\langle \bigcirc \rangle$ a donné $C_{i+1}$ à partir de $C_i$, $s' = s$ dans tous les autres cas.

4. Puisque $(C_i)$ est infinie, elle doit contenir un cycle du graphe $\mathcal{T}$. On remarque que de toutes les règles, seule $\langle \bigcirc \rangle$ a potentiellement pour fils un noeud déjà existant. Donc tout cycle de $\mathcal{T}$ contient une application de la règle $\langle \bigcirc \rangle$, et en particulier $(C_i)$ contient une infinité d'applications de la règle $\langle \bigcirc \rangle$.

5. Par les points 1., 3. et 4. il existe une infinité de $i \in \mathbb{N}$ tels que $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ est une réalisation de $C_i$ et $\mathcal{R}' = (\mathcal{K}', \lfloor . \rfloor', s')$ une réalisation de $C_{i+1}$ et $s' = s+1$, d'où le résultat. □

Il nous reste à montrer que tous les TPN clos sont non réalisables. Le cas du TPN absurde est à part puisqu'il ne peut pas être réalisé par définition. Les autres cas sont un TPN dont tous les fils sont clos (cas traité rapidement par le Lemme 3.21) et enfin le cas des formules $\Diamond A \in \mathcal{F}^+$ et $\Box A \in \mathcal{F}^-$. Dans ces cas, on adapte une technique héritée des calculs des tableaux de LTL. Il s'agit de remarque qu'un TPN réalisable tel que $\Diamond A \in \mathcal{F}^+$ "promet" l'apparition d'une formule $A \in Fp$ dans un autre TPN réalisable (par définition sémantique de $\Diamond$). Or par définition de la clôture, cette condition implique que le chemin entre les deux TPN contient un TPN clos et donc n'est pas entièrement réalisable. Cette idée est formalisée dans la preuve du lemme suivant.

**Lemme 3.23.** *Les TPN clos autres que* ■ *ne sont pas réalisables.*

*Démonstration.*
On prouve par récurrence sur la taille de l'ensemble des TPN clos autres que ■.

- Cas de base
  S'il n'y a qu'un TPN clos autre que ■, alors il l'est par la condition 2 de la Définition 3.29 et a pour fils ■. Alors par le Lemme 3.20, ce TPN n'est pas réalisable.

- Hérédité
  On suppose que les $k$ TPN clos autres que ■ sont clos (HR1). On discute selon la nature du $k+1$-ème TPN clos autre que ■. On note $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ ce TPN. On a 3 cas :

  - Tous les fils de $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ sont clos. Alors par (HR1), tous les fils de $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ sont non réalisable. Alors par contraposée du Lemme 3.21, $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ est clos aussi.

  - $A$ est une formule telle que $\Diamond A : x \in \mathcal{F}^+$ et pour tout nœud $C' = \langle \mathcal{F}^{+\prime}, \mathcal{F}^{-\prime}, \mathcal{C}'_r \rangle$ tel que $A : x \in \mathcal{F}^{+\prime}$ tout chemin $(C, C')$ contient un TPN clos (C).

    On suppose que $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ est réalisable. Soit $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ une réalisation de ce TPN. Comme $\Diamond A : x \in \mathcal{F}^+$, par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \Diamond A : x$. Donc par la Définition 3.4, il existe $j \in S$ tel que $j \geq s$ et $(\lfloor x \rfloor, j) \vDash_{\mathcal{K}} A$. On considère le chemin réalisable $(C_i)$ partant de $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$. Par le Lemme 3.22, il existe au moins un $i$ tel que $C_i$ ait pour réalisation $\mathcal{R}' = (\mathcal{K}', \lfloor . \rfloor', j)$ (en prenant $j = s + k$ dans le lemme). Soit $i_0$ le plus petit indice qui vérifie cette propriété. Alors $C_{i_0-1}$ est réalisé par $j-1$ et $C_{i_0}$ est obtenu par la règle $\langle \bigcirc \rangle$.

    On considère la sous-suite $(C_0, \ldots, C_{i_0})$. On montre par récurrence sur cette suite que pour tout $C_k$ on a $\Diamond A : x \in \mathcal{F}^+$ ou bien $\bigcirc \Diamond A : x \in \mathcal{F}^+$. Le cas initial est vrai par hypothèse. Supposons que cette propriété est vraie pour un certain $k$ (HR2). On montre qu'elle tient pour $k+1$. Par (HR2), $\Diamond A : x \in \mathcal{F}^+$ ou bien $\bigcirc \Diamond A : x \in \mathcal{F}^+$ dans $C_k$.

    - Si $\Diamond A : x \in \mathcal{F}^+$ dans $C_k$, si une autre règle que $\langle \Diamond^+ \rangle$ est appliquée, on a toujours $\Diamond A : x \in \mathcal{F}^+$ dans $C_{k+1}$. Si $\langle \Diamond^+ \rangle$ est appliquée, on ne peut pas choisir le fils de gauche de la règle, autrement on a $A : x \in \mathcal{F}^+$ dans $C_{k+1}$, et par (C) $(C_0, \ldots, C_{k+1})$ contient un noeud clos qui par (HR1) est non réalisable, ce qui est absurde par le Lemme 3.22. Donc $C_{k+1}$ est nécessairement le fils droit et $\bigcirc \Diamond A : x \in \mathcal{F}^+$ dans $C_{k+1}$.

    - Si $\bigcirc \Diamond A : x \in \mathcal{F}^+$ dans $C_k$, si une autre règle que $\langle \bigcirc \rangle$ est appliquée, on a toujours $\bigcirc \Diamond A : x \in \mathcal{F}^+$ dans $C_{k+1}$. Si $\langle \bigcirc \rangle$ est appliquée, alors $\Diamond A : x \in \mathcal{F}^+$ dans $C_{k+1}$.

    Finalement, pour tous les TPN de $(C_0, \ldots, C_{i_0})$ on a $\Diamond A : x \in \mathcal{F}^+$ ou bien $\bigcirc \Diamond A : x \in \mathcal{F}^+$.

    Comme la règle $\langle \bigcirc \rangle$ est appliquée sur $C_{i_0-1}$, on a nécessairement $\bigcirc \Diamond A : x \in \mathcal{F}^+$ dans $C_{i_0-1}$ (autrement on ne peut pas appliquer $\langle \bigcirc \rangle$). Donc $\Diamond A : x \in \mathcal{F}^+$ dans $C_{i_0}$. Alors il existe $l > i_0$ tel que la règle $\langle \Diamond \rangle$ soit appliquée sur $C_l$ et qu'on ait toujours $j$ qui réalise $C_l$ (autrement on aurait du appliquer $\langle \bigcirc \rangle$ avant, ce qui n'est pas possible vu les conditions de la règle). Alors comme $(\lfloor x \rfloor, j) \vDash_{\mathcal{K}} A$, le fils gauche de $C_l$ est réalisable et on peut le choisir comme $C_{l+1}$.

    Finalement, la suite $(C_0, \ldots, C_{l+1})$ est un chemin de TPN, tous réalisables et $A : x \in \mathcal{F}^+$ dans $C_{l+1}$. Par (HR1) aucun de ces nœuds n'est donc clos. C'est absurde vu (C). Donc $\langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$ n'est pas réalisable.

- $A$ est une formule telle que $\Box A : x \in \mathcal{F}^-$ et pour tout nœud $C' = \langle \mathcal{F}^{+\prime}, \mathcal{F}^{-\prime}, \mathcal{C}'_r \rangle$ tel que $A : x \in \mathcal{F}^{-\prime}$ tout chemin $(C, C')$ contient un TPN clos. Ce cas se prouve comme le précédent.

□

Enfin, on peut énoncer le théorème de correction.

**Théorème 3.4** (Correction des c-tableaux)**.** *Soit $\phi$ une LTBI-formule. S'il existe une LTBI-c-preuve de $\phi$,alors $\phi$ est valide.*

*Démonstration.*
Soit $\mathcal{T}$ une LTBI-c-preuve de $\phi$. On suppose que $\phi$ n'est pas valide. Alors il existe un modèle linéaire de ressources $\mathcal{K}$ tel que $e, s_0 \not\Vdash_{\mathcal{K}} \phi$. Si on considère $\lfloor 1_r \rfloor = e$, on obtient une réalisation $(\mathcal{K}, \lfloor . \rfloor, s_0)$ de la racine $\langle \{\}, \{\phi : 1_r\}, \{1_r \leqslant 1_r\} \rangle$. Comme c'est une c-preuve, sa racine est close. Donc par le Lemme 3.23, elle n'est pas réalisable ce qui est absurde. Donc $\phi$ est valide. □

Nous n'avons pas présenté de preuve de la complétude du calcul des c-tableaux pour LTBI. En effet, contrairement à la preuve de correction ci-dessus, il est très ardu d'adapter la preuve utilisée pour des versions et extensions de BI au calcul des c-tableaux, vue la structure fondamentalement différente de ces derniers.

De même, les preuves de complétude pour les calculs de LTL [65] sont ardues à adapter, s'appuyant sur le caractère fini des tableaux pour LTL ce qui n'est plus forcément le cas avec l'ajout des labels de ressources de BI.

Une piste intéressante pour démontrer la complétude de manière élégante serait de parvenir à transformer systématiquement un tableau pour une LTBI-formule (tels que présentés dans la section 3.4) en un c-tableau pour cette même formule, avec conservation du caractère clos. Alors, la complétude du calcul des c-tableaux serait une conséquence de celle des tableaux.

Cette solution non plus n'est pas aisée. La difficulté de la transformation réside dans le traitement fondamentalement différent des dimensions temporelles : les c-tableaux détaillent explicitement tous les états potentiels par des nœuds successifs, en utilisant les cycles dans le graphe pour modéliser les chemins infinis.

Les tableaux, eux, utilisent les labels d'états comme des variables symbolisant des états potentiels distincts et (dans la majorité des cas) finis. Il faut donc pour passer de l'un à l'autre déplier les cycles des c-tableaux et en extraire les états significatifs pour en faire des labels d'états distincts.

Ainsi la transformation d'un tableau en un c-tableau (ou inversement) reste à l'état de projet pour le moment, et par suite la preuve de complétude qu'elle porte en germe fait partie des travaux futurs.

### 3.6.4 Comparaisons des calculs

Puisque nous avons soulevé la traduction d'un calcul à l'autre, il est intéressant de constater comment les c-tableaux et les tableaux se comportent les uns par rapport aux autres.

La première chose qui frappe est que les c-tableaux sont plus ardus à écrire. En effet, là où les tableaux ne font écrire, à chaque étape, que les formules nouvellement générées, les c-tableaux forcent à recopier à chaque étape l'intégralité des formules non concernées par l'application

d'une règle. Cette contrainte est essentielle pour faire apparaître les cycles nécessaires à la clôture des c-tableaux.

Dans le même ordre d'esprit, on observe que la condition de clôture des c-tableaux est plus complexe que celle des tableaux simples. Les cas 3 et 4 de la Définition 3.29 demandent une recherche approfondie dans les c-tableaux, là où les règles de clôture de la Définition 3.16 étaient facile à appliquer.

Ainsi plusieurs points rendent les c-tableaux plus complexes. Toutefois, un point important est à noter : quant à l'application des règles temporelles, les c-tableaux ne font pas appel à la sélection d'une contrainte comme les tableaux. L'application de ces règles est donc plus systématique. De même, les points délicats de la Définition 3.29 mentionnés plus haut sont systématiques et ne demandent pas de choix judicieux.

Ainsi, on peut remarquer que les c-tableaux sont plus difficiles à construire à la main mais plus systématiques à développer et à clore. Les tableaux, eux, demandent plus de choix dans leur construction pour être établis convenablement (choix dus aux contraintes à sélectionner dans l'application des règles).

### 3.6.5 Vers des modèles fondés sur les automates

L'une des perspectives très intéressantes du calcul des tableaux de LTL est la possibilité de transformer un tableau pour une formule en un automate qui est un modèle de la formule considérée. Cette méthode est décrite de manière extensive dans [65].

On ne donnera ici qu'une idée générale de cette transformation. Le calcul des tableaux pour LTL ressemble fortement au calcul des c-tableaux que nous venons de présenter (à l'exception des règles multiplicatives).

La transformation en automate isole en particulier les applications de la règle similaire à $\langle \circ \rangle$. Les arêtes produites par une telle application correspondent au passage à un état suivant, ce seront donc les transitions de notre automate. Tous les nœuds en amont et en aval d'une telle transition produisent un état différent dans l'automate. Les états (ou dans certaines versions, les transitions) sont étiquetés par les formules atomiques valides dans les noeuds correspondants du tableau initial. Enfin, les états initiaux de l'automate sont déterminés par la racine du tableau tandis que les états finaux sont calculés à partir des "promesses" des formules $\Diamond$ et $\Box$ (pour plus de détails, consulter [51,52,65]).

Cette transformation produit un automate de Büchi généralisé (c'est à dire un automate qui accepte des mots infinis, avec plusieurs états finaux) qui peut être transformé à son tour en un automate de Büchi (avec un seul état final). Plusieurs optimisations de cette transformation ont été proposées, comme celles de [51,52].

Idéalement, on souhaiterait pouvoir transposer une telle transformation dans le cas de LTBI. Cette transposition n'a pas été effectuée. On voit bien comment la partie temporelle s'adapte très bien de LTL à LTBI, vu que les c-tableaux et les tableaux de LTL sont très proches.

Le challenge est de faire figurer la structure des ressources sur les automates produits par la traduction. Cela semble possible, car l'organisation des contraintes de ressources selon un graphe est une pratique courante dans les calculs des tableaux de la logique BI (une description exhaustive de cette pratique peut être trouvée dans [48]). Le travail est donc d'intégrer ces graphes de ressources dans les automates de manière à traduire dans le modèle-automate les contraintes sur les ressources décrites par le c-tableau.

Si ce travail aboutit, il permettrait donc de générer sous la forme d'automates des modèles de LTBI ce qui est un avancement à la fois pour illustrer des modélisations, et pour utiliser LTBI comme une base pour de nouvelles formes de model checking [26,27,84].

### 3.6.6 Quelques perspectives sur les calculs des tableaux

Enfin, nous voulons conclure ces discussions par une brève mention de ce que le développement du calcul des c-tableaux offre comme perspective vis-à-vis des logiques dérivées de BI, et des calculs des tableaux en général.

Nous avons vu comment dans le calcul des tableaux de LTL (que nous avons adapté pour LTBI), certaines arêtes du tableau (en l'occurrence celles issues de la règle $\langle \bigcirc \rangle$) illustrent la structure temporelle des modèles (ou contre-modèles) des formules considérées. Ainsi, le rôle joué dans le calcul des tableaux de la section 3.4 par les labels d'états est ici joué par certains arcs du tableau.

On peut donc se demander si cela pourrait être possible dans le cadre d'autres logiques qui utilisent traditionnellement un calcul des tableaux avec des labels. C'est le cas de BI et BBI [49,66]. Une piste de recherche est donc l'établissement d'un nouveau calcul des tableaux pour BI ou BBI, dans lequel les habituelles contraintes sur les labels seraient remplacées par des règles d'une forme différente, potentiellement avec différents types d'arcs dans le tableau.

Une telle démarche rejoint à la fois l'esprit de la conception primordiale de BI avec un calcul des séquents avec plusieurs types de multi-ensembles [74], mais aussi l'extraction de contre-modèles sous forme de graphes, comme dans [48]. En effet, faire porter les relations entre ressources par la structure des tableaux reviendrait à intégrer les graphes de ressources dans les tableaux.

En outre, la réflexion autour de cette nouvelle forme potentielle de tableaux pose la question plus générale de la pertinence des labels dans les calculs de tableaux. Il a été montré que dans certains cas, l'utilisation de labels accélère la recherche de preuves [56], mais la pertinence de leur utilisation est encore débattue. Avoir un calcul alternatif pour BI ou BBI et leurs variantes sans utiliser de labels, avec une traduction de tableaux d'une version vers l'autre nourrira le débat sur la pertinence des labels. Notons qu'un autre calcul sans labels pour BBI est présenté dans le Chapitre 5 de cette thèse, avec une discussion sur ce point.

Enfin, on peut noter que le développement d'un calcul sans label de ressources, avec des arêtes traduisant la structure des ressources, n'est pas sans rappeler le calcul des séquents à bunches intialement développé pour BI [74,80]. En effet, dans les bunches, les connecteurs , et ; jouent le même rôle que les labels et les contraintes du calcul des tableaux. L'utilisation des TPN dans la méthode des c-tableaux, qui groupe les formules en un ensemble de formules vraies et un autre de formules fausses rapproche également ce calcul des séquents où la structure est similaire.

# Conclusion

Nous avons présenté une nouvelle logique, LTBI, qui est une extension de la logique BI par des opérateurs modaux temporels ($\Diamond$, $\Box$ et $\bigcirc$) dont la sémantique est inspirée par la logique temporelle linéaire LTL. Ce travail poursuit celui des logiques de ressources dynamique, notamment DBI mais en tirant part des spécificités des logiques purement temporelles et de LTL en particulier.

Nous avons montré comment cette logique permettait des modélisations intéressantes de systèmes dynamiques cloisonnés ou de consommation de ressources. En outre, nous avons proposé un développement futur de LTBI comme une logique temporelle branchante d'une nouvelle forme, avec le potentiel de résonner sur la parallélisation de programmes.

Nous avons présenté un calcul des tableaux pour LTBI, inspiré par celui de DBI, que nous avons illustré par des exemples, et dont nous avons fourni une preuve de correction, de complétude ainsi qu'une méthode d'extraction de contre-modèles. Toutefois, l'exactitude de ce calcul dans tous les cas repose sur une conjecture complexe que nous n'avons pas montrée. Ce point faible n'impacte pas le calcul dans la majorité des cas pratiques, puisque seul les tableaux générant une infinité de labels d'états coincés entre deux autres pose problème.

Enfin, nous avons présenté le début d'un travail sur un autre calcul, appelé calcul des c-tableaux, qui utilise une structure de graphe pour décrire l'évolution temporelle des formules. Ce calcul offre des perspectives intéressantes, notamment quant à la génération de modèles sous forme d'automates, mais également quant à la discussion générale sur la pertinence des labels dans les calculs des tableaux.

# Chapitre 4

# Une extension modale et épistémique de BBI

Notre chapitre précédent présentait une extension temporelle de la logique BI en utilisant les modalités de la logique temporelle LTL. Dans ce chapitre, nous allons effectuer un travail similaire avec les modalités épistémiques, tout en ajoutant des ressources à la syntaxe pour conditionner les accès épistémiques par des ressources. Cette fois, dans l'optique de modéliser des systèmes de contrôle d'accès, nous choisissons de prendre comme base la logique classique BBI qui est plus appropriée à la modélisation.

La logique ESL [37] a proposé une extension de BBI par un opérateur épistémique $K_a$ modélisant la connaissance d'un agent $a$. Toutefois, cette connaissance n'est pas dépendante de facteurs extérieurs. D'un autre côté, la logique LSM [36] étendait BBI par des opérateurs modaux $\Diamond_r$ et $\Box_r$ paramétrés par des ressources. Un traitement similaire de l'opérateur épistémique de ESL permet de conditionner l'accès d'un agent à une propriété. L'accès à des ressources pour certains agents seulement (offert par l'extension épistémique) et à la condition de l'accès préalable à d'autres ressources (offert par la paramétrisation par une ressource locale) pose les bases d'une modélisation des problèmes de contrôle d'accès en général (voir Section 2.3.3).

Dans cette optique, après avoir présenté la syntaxe et la sémantique de notre nouvelle logique ERL, nous développons des exemples où cette logique est utilisée pour modéliser des problèmes traditionnels de contrôle d'accès (accès faillible à un espace restreint, accès conjoint à une ressource critique, accès concurrent à une ressource critique, etc.).

Enfin, nous présentons un calcul des tableaux pour raisonner sur notre logique, inspiré de celui pour ESL. De nouvelles contraintes permettent de traduire les relations de connaissance des agents, tandis que des labels particuliers assurent la bonne représentation des ressources apparaissant dans les opérateurs. Nous prouvons la correction et la complétude de notre calcul, tout en proposant une méthode d'extraction de contre-modèles. Dans ces preuves, la correspondance entre ressources et labels doit être particulièrement examinée pour les ressources apparaissant dans les opérateurs.

Notons que la logique ERL présentée dans ce chapitre a été introduite dans [45].

## 4.1 De l'extension épistémique au contrôle d'accès

On a présenté dans le Chapitre 2 la logique ESL [37] qui étend BBI par des opérateurs épistémiques. D'autres logiques comme LSM [36] ont intégré des ressources à des opérateurs pour paramétriser des accès modaux. On se propose de reprendre les opérateurs épistémiques de ESL et d'y rajouter une paramétrisation par une ressource.

On voit facilement comment ce type de modalités pourrait être intéressant pour modéliser des problèmes de sécurité, notamment de contrôle d'accès, ce pourquoi les logiques épistémiques ont d'ors et déjà été considérées [40,91]. Il paraît donc fortement avantageux de combiner ces deux types de modalités (épistémiques et paramétrisées par des ressources), notamment dans le but de modéliser des problèmes de contrôle d'accès.

Bien entendu, de nombreux autres formalismes existants peuvent aussi être utilisés pour modéliser de telles situations. En particulier, plusieurs cadres logiques ont été présentés pour modéliser le contrôle d'accès dans certaines situations spécifiques. Par exemple, la réécriture de multi-ensembles a été utilisée pour caractériser des protocoles de sécurité [24]. Notre but ici, toutefois, est de fournir un cadre plus général qui peut être un outil de modélisation dans plusieurs situations plutôt qu'un outil ad-hoc créé dans un contexte spécifique.

Cette sorte de modélisation générique a aussi été abordée par des extensions de la logique linéaire [50]. Nous avons déjà mentionné dans le Chapitre 2 les distinctions entre LL et BBI qui font de BBI un outil plus riche pour la modélisation. En outre, notre approche du contrôle d'accès avec BBI est plus subtile. En effet, puisque la sémantique de BBI est centrée sur les ressources et leurs propriétés, on peut directement utiliser ces ressources comme des jetons pour modéliser, en particulier dans la portée de nos opérateurs. Ainsi, la ressource locale $s$ dans $r \vDash \mathbf{L}_a^s$ par exemple est de la même nature que la ressource ambiante $r$ (même si elle n'a pas le même rôle). Ceci permet une intégration simple des nouveaux acteurs d'un système dans la modélisation en utilisant ERL et évite la création de nouveaux éléments formels d'une nature plus ad-hoc.

Notre nouvelle logique ERL prend donc le formalisme de ESL, mais en intégrant des ressources indiçant les opérateurs modaux, comme dans LSM. Comme ces deux logiques, ERL sera une extension de la logique BBI, dont la partie additive est classique, ce qui est plus approprié pour la modélisation (voir Section 2.3.3).

Les nouveaux opérateurs de ERL sont en premier lieu inspirés des opérateurs épistémiques, en particulier le plus simple d'entre eux, $K_a$. Dans la logique épistémique, si $a$ est un agent, $\phi$ une formule et $w$ un monde modal, alors on a $w \vDash K_a\phi$ si, pour tout monde $w'$ tel que $w \sim_a w'$ on a $w' \vDash \phi$, où $\sim_a$ est une relation d'équivalence entre les mondes, propre à $a$, qui modélise la connaissance de $a$. Ainsi, on dit que $a$ sait $\phi$ s'il a accès à un monde où $\phi$ est valide.

Notre première idée est d'utiliser au lieu des mondes modaux les ressources de la logique BBI (comme cela a été fait pour ESL dans [37]), on aura ainsi $r \sim_a r'$ au lieu de $w \sim_a w'$. Ensuite, dans l'esprit de [37], on veut que la sémantique de nos nouveaux opérateurs incorporent la paramétrisation par une ressource locale $s$. Cela nous donne trois possibilités. En effet, on peut vouloir composer la ressource $r$ avec $s$, ou bien la ressource quantifiée $r'$, ou bien encore les deux.

ERL propose ainsi trois nouveaux opérateurs, que nous noterons $\mathbf{L}$, $\mathbf{M}$ et $\mathbf{N}$, et dont la sémantique est la suivante :

- $r \models_{\mathcal{K}} \mathbf{L}^s_a\phi$ ssi si $r \bullet s \downarrow$ alors pour tout $r' \in R$ tel que $r \bullet s \sim_a r'$, on a $r' \models_{\mathcal{K}} \phi$
- $r \models_{\mathcal{K}} \mathbf{M}^s_a\phi$ ssi pour tout $r' \in R$ tel que $r' \bullet s \downarrow$ et $r \sim_a r' \bullet s$ , on a $r' \bullet s \models_{\mathcal{K}} \phi$
- $r \models_{\mathcal{K}} \mathbf{N}^s_a\phi$ ssi si $r \bullet s \downarrow$ alors pour tout $r' \in R$ tel que $r' \bullet s \downarrow$ et $r \bullet s \sim_a r' \bullet s$, on a $r' \bullet s \models_{\mathcal{K}} \phi$

On voit comment ces trois opérateurs traduisent, avec des nuances, le fait que l'agent $a$ peut accéder à la formule $\phi$ moyennant la ressource $s$. Ces modalités paraissent adaptées pour modéliser des contrôles d'accès, puisque l'accès à $\phi$ est à la fois propre à l'agent $a$ et conditionné par la présence de la ressource $s$.

Dans ce chapitre, nous allons tout d'abord détailler la syntaxe et la sémantique de ERL. Nous proposons notamment une légère extension ERL$^*$ où la composition $\bullet$ et l'équivalence pour les agents $\sim_a$ sont compatibles. Nous montrons aussi comment ERL est une extension conservative à la fois de BBI et de la logique épistémique EL.

Nous présentons ensuite des exemples d'applications de ERL à des problèmes de contrôle d'accès : tout d'abord un exemple d'accès conditionné à un espace restreint, avec modélisation d'une faille de sécurité, puis un accès conjoint à une ressource critique, puis un accès concurrent à une ressource critique (avec des sémaphores), enfin un exemple pour illustrer la modélisation par couches.

Après cela, nous proposons un calcul des tableaux pour ERL inspiré par celui de ESL. La différence fondamentale est l'incorporation dans le calcul des ressources portées par les opérateurs : il faut s'assurer que chacune des ressources apparaissant dans les opérateurs possède un représentant unique parmi les labels de ressources. Cela s'effectue en distinguant un ensemble de labels de même taille que l'ensemble des ressources apparaissant dans les opérateurs, et en dressant un isomorphisme entre ces deux ensembles. Nous fournirons une preuve de *correction*, de *complétude* et une méthode d'extraction de *contre-modèles*.

## 4.2 Syntaxe de la logique ERL

De la même manière que pour LTBI, nous voulons que notre nouvelle logique, qui est à la fois une extension de la logique épistémique et de la logique BI, soit conservative par rapport à ces deux logiques. Nous allons donc naturellement reprendre les opérateurs de BI, mais cette fois en leur ajoutant une toute nouvelle série d'opérateurs.

Comme nous l'avons explicité dans la section précédente, l'enjeu de notre nouvelle logique est de décrire l'accessibilité conditionné par les ressources de notre structure sémantique. Il nous faut donc avoir accès aux ressources directement dans notre syntaxe. Pour plus de simplicité, nous partons du principe que l'ensemble des ressources que nous utilisons dans la syntaxe est fini.

Nous devons également refléter dans ce sous-ensemble de ressources la composition qui sous-tend notre sémantique ainsi que la ressource unitaire $e$. Cette opération n'est pas fondamentalement obligatoire mais elle nous permet une plus grande expressivité, en paramétrant nos opérateurs par des ressources composées ou par la ressource unitaire. Nous en verrons l'utilité plus en avant dans les exemples illustratifs de la section 4.4.

Voici donc la syntaxe de notre nouvelle logique que nous dénommons ERL pour Epistemic Resource Logic :

**Définition 4.1** (Syntaxe de ERL)**.** *Soit A un ensemble fini de symboles d'agents. Soit Res un ensemble fini non vide de symboles de ressource, muni d'une loi de composition interne partielle* $\cdot$ *(*$\cdot : Res \times Res \rightharpoonup Res$*) et dont un des éléments est noté e. Soit Prop un ensemble de symbole propositionnels. Le langage* $\mathcal{L}$ *de la logique ERL est donnée par la grammaire suivante pour tout* $p \in Prop$*, pour tout* $a \in A$ *et tout* $s \in Res$ *:*

$$X ::= p \mid \top \mid \bot \mid \neg X \mid X \wedge X \mid X \vee X \mid X \to X \mid \mathrm{I} \mid X * X \mid X \mathrel{-\!\!*} X \mid \mathbf{L}_a^s X \mid \mathbf{M}_a^s X$$

Dans l'intégralité de ce chapitre, $\mathcal{L}$ désignera le langage de ERL sauf indication contraire. Par simplification, on omettra chaque fois que ce n'est pas ambigu l'écriture de $\cdot$. Ainsi, on écrira $rs$ au lieu de $r \cdot s$ et $\mathbf{L}_a^{rs}$ au lieu de $\mathbf{L}_a^{r \cdot s}$.

Notons que, contrairement à LTBI, la négation $\neg$ est explicitement donnée dans la syntaxe au lieu d'être définie à partir de l'implication. Cela est justifié par le fait que nous allons choisir une sémantique classique pour ERL, calquée sur celle de BBI. Ainsi, comme on l'a montré dans le Lemme 2.3, on peut définir sémantiquement la négation comme on le fait pour les logiques classiques.

On introduit en outre les opérateurs auxiliaires suivants :

$$\mathbf{N}_a^s X \equiv \mathbf{L}_a^s(\mathbf{M}_a^s X) \qquad \widetilde{\mathbf{L}}_a^s X \equiv \neg \mathbf{L}_a^s \neg X \qquad \widetilde{\mathbf{M}}_a^s X \equiv \neg \mathbf{M}_a^s \neg X \qquad \widetilde{\mathbf{N}}_a^s X \equiv \neg \mathbf{N}_a^s \neg X$$

Avant de donner une sémantique formelle de cette syntaxe, nous voulons donner ici une idée informelle de la signification des nouveaux opérateurs. Tous les opérateurs que nous avons introduits ($\mathbf{L}_a^s$, $\mathbf{M}_a^s$, $\mathbf{N}_a^s$, $\widetilde{\mathbf{L}}_a^s$, $\widetilde{\mathbf{M}}_a^s$ et $\widetilde{\mathbf{N}}_a^s$) évoquent le fait que l'agent $a$ peut avoir accès, moyennant la ressource $s$, à la formule encapsulée dans l'opérateur.

Notons que deux familles se distinguent facilement : les opérateurs simples que sont $\mathbf{L}_a^s$, $\mathbf{M}_a^s$ et $\mathbf{N}_a^s$ dénotent une propriété valable pour tous les mondes accessibles, tandis que les opérateurs alternatifs $\widetilde{\mathbf{L}}_a^s$, $\widetilde{\mathbf{M}}_a^s$ et $\widetilde{\mathbf{N}}_a^s$ dénotent l'existence d'un monde accessible vérifiant une propriété.

Enfin, chaque famille comporte trois opérateurs distincts dont voici la signification :

- $\mathbf{L}_a^s\phi$ signifie que la formule $\phi$ est valide pour toutes les ressources qui contiennent $s$ et qui sont équivalentes à la ressource courante du point de vue de $a$.
- $\mathbf{M}_a^s\phi$ signifie que la formule $\phi$ est valide pour toutes les ressources qui sont équivalentes à la ressource courante à laquelle on ajoute $s$ du point de vue de $a$.
- $\mathbf{N}_a^s\phi$ signifie que la formule $\phi$ est valide pour toutes les ressources qui contiennent $s$ et qui sont équivalentes à la ressource courante à laquelle on ajoute $s$ du point de vue de $a$.

Voyons maintenant comment ces points de vue se traduisent formellement.

## 4.3 Sémantique de la logique ERL

Nous avons fait le choix de baser notre logique sur BBI comme la logique ESL dont elle s'inspire.

ERL est encore une fois une logique qui a une double paternité : la dimension séparative de BBI et la dimension epistémique des logiques du même nom. Nous devons donc conserver

la structure de ressources de BBI et les relations d'équivalence au centre des logiques épistémiques.

Toutefois, notre structure doit aussi incorporer la particularité de cette logique, à savoir l'inclusion des ressources présentes dans les opérateurs aux conditions d'accès. Comme nous utilisons les ressources de la structure propre à BBI, il n'est pas nécessaire d'introduire de nouveaux éléments, mais il faut toutefois s'assurer que la structure des ressources apparaissant dans la syntaxe coïncide avec celle définie dans notre sémantique.

### 4.3.1 Structure de Kripke

Nous introduisons donc une nouvelle forme de monoïde, très similaire à ceux de BBI, mais qui prennent en compte les ressources internes de la syntaxe.

**Définition 4.2** (Monoïde de ressources partiel (pour ERL))**.** *Un* monoïde de ressources partiel *est une structure $\mathcal{R} = (R, \bullet, \sim)$ telle que :*

- *$R$ est un ensemble dénombrable de ressources tel que $Res \subseteq R$ (donc en particulier $e \in R$)*
- *$(R, e, \bullet)$ est un monoïde partiel commutatif, tel que défini dans la Définition 2.3*
- *$\bullet$ étend $\cdot$, c'est à dire que pour tout $r_1, r_2 \in Res$, si $r_1 \cdot r_2 \downarrow$, alors $r_1 \bullet r_2 \downarrow$ et $r_1 \bullet r_2 = r_1 \cdot r_2$*
- *$\sim$ est une relation d'équivalence sur $R$*
- *$\bullet$ est compatible avec $\sim$. C'est à dire que pour tout $r_1, r_2, s \in R$, si $r_1 \sim r_2$ et $r_2 \bullet s \downarrow$, alors $r_1 \bullet s \downarrow$ et $r_1 \bullet s \sim r_2 \bullet s$*

La différence avec la structure de BBI réside en deux points : d'abord le fait que $\bullet$ prolonge la composition de *Res*, ensuite que l'élément neutre $e$ est celui de *Res*. Cela nous assure que toute ressource composée ou unitaire utilisée dans un opérateur de ERL coïncide bien avec la composition de notre sémantique. Définissons maintenant une interprétation pour ERL (qui sera la même que dans BBI).

**Définition 4.3** (Interprétation)**.** *Soit $\mathcal{R} = (R, \bullet, \sim)$ un monoïde de ressources partiel (selon la Définition 4.2). Une* interprétation *de Prop pour $\mathcal{R}$ est une fonction $[\![\cdot]\!] : Prop \to \mathbb{P}(R)$ qui est monotone, c'est à dire qu'elle vérifie la propriété suivante :*

$$\forall p \in Prop, \forall r, r' \in R, \text{ si } r \sim r' \text{ et } r \in [\![p]\!] \text{ alors } r' \in [\![p]\!] \text{ (monotonie)}$$

Enfin, on peut définir nos modèles.

**Définition 4.4** (Modèle)**.** *Un* modèle *de ERL est un quadruplet $\mathcal{K} = (\mathcal{R}, \{\sim_a\}_{a \in A}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où :*

- *$\mathcal{R} = (R, \bullet, \sim)$ est un monoïde de ressources partiel (comme défini dans la Définition 4.2)*
- *pour tout $a \in A$, $\sim_a$ est une relation d'équivalence sur $R$ ($\sim_a \subseteq R \times R$)*
- *pour tout $a \in A$, $\sim_a$ est en outre compatible avec $\sim$ (c'est à dire que si $r \sim r'$ et $r \sim_a r''$, alors $r' \sim_a r''$)*
- *$[\![\cdot]\!]$ est une interprétation de Prop pour $\mathcal{R}$ (comme défini dans la Définition 4.3)*

- $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times R$ *une relation de forcing, définie récursivement par les relations suivantes pour tout* $p \in Prop$, *pour tout* $s \in Res$, *pour tout* $a \in A$ *et pour tout* $\phi, \psi \in \mathcal{L}$ :
  - $r \vDash_{\mathcal{K}} p$ *ssi* $r \in [\![p]\!]$
  - $r \vDash_{\mathcal{K}} \mathrm{I}$ *ssi* $e \sim r$
  - $r \vDash_{\mathcal{K}} \top$ *toujours*
  - $r \vDash_{\mathcal{K}} \bot$ *jamais*
  - $r \vDash_{\mathcal{K}} \neg\phi$ *ssi* $r \nVdash_{\mathcal{K}} \phi$
  - $r \vDash_{\mathcal{K}} \phi \wedge \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *et* $r \vDash_{\mathcal{K}} \psi$
  - $r \vDash_{\mathcal{K}} \phi \vee \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *ou* $r \vDash_{\mathcal{K}} \psi$
  - $r \vDash_{\mathcal{K}} \phi \to \psi$ *ssi si* $r \vDash_{\mathcal{K}} \phi$, *alors* $r \vDash_{\mathcal{K}} \psi$
  - $r \vDash_{\mathcal{K}} \phi * \psi$ *ssi il existe* $r', r'' \in R$ *tels que* $r' \bullet r'' \downarrow$ *et* $r' \bullet r'' \sim r$ *et* $r' \vDash_{\mathcal{K}} \phi$ *et* $r'' \vDash_{\mathcal{K}} \psi$
  - $r \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ *ssi pour tout* $r' \in R$ *tel que* $r \bullet r' \downarrow$ *et* $r' \vDash_{\mathcal{K}} \phi$, *on a* $r \bullet r' \vDash_{\mathcal{K}} \psi$
  - $r \vDash_{\mathcal{K}} \mathbf{L}^s_a\phi$ *ssi si* $r \bullet s \downarrow$ *alors pour tout* $r' \in R$ *tel que* $r \bullet s \sim_a r'$, *on a* $r' \vDash_{\mathcal{K}} \phi$
  - $r \vDash_{\mathcal{K}} \mathbf{M}^s_a\phi$ *ssi pour tout* $r' \in R$ *tel que* $r' \bullet s \downarrow$ *et* $r \sim_a r' \bullet s$ , *on a* $r' \bullet s \vDash_{\mathcal{K}} \phi$

Les relations $\sim_a$ sont une représentation de la connaissance (ou de la croyance) des agents. Deux ressources sont liées par $\sim_a$ si elles sont équivalentes aux yeux de $a$.

On peut donner la sémantique suivante aux opérateurs auxiliaires :

**Proposition 4.1** (Sémantique des opérateurs auxiliaires)**.** *Soit* $\mathcal{K} = (\mathcal{R}, \{\sim_a\}_{a\in A}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *un modèle et soit* $r \in R$. *On a :*

1. $r \vDash_{\mathcal{K}} \mathbf{N}^s_a\phi$ *ssi si* $r \bullet s \downarrow$ *alors pour tout* $r' \in R$ *tel que* $r' \bullet s \downarrow$ *et* $r \bullet s \sim_a r' \bullet s$, *on a* $r' \bullet s \vDash_{\mathcal{K}} \phi$
2. $r \vDash_{\mathcal{K}} \widetilde{\mathbf{L}}^s_a\phi$ *ssi* $r \bullet s \downarrow$ *et il existe* $r' \in R$ *tel que* $r \bullet s \sim_a r'$ *et* $r' \vDash_{\mathcal{K}} \phi$
3. $r \vDash_{\mathcal{K}} \widetilde{\mathbf{M}}^s_a\phi$ *ssi il existe* $r' \in R$ *tel que* $r' \bullet s \downarrow$ *et* $r \sim_a r' \bullet s$ *et* $r' \bullet s \vDash_{\mathcal{K}} \phi$
4. $r \vDash_{\mathcal{K}} \widetilde{\mathbf{N}}^s_a\phi$ *ssi* $r \bullet s \downarrow$ *et il existe* $r' \in R$ *tel que* $r' \bullet s \downarrow$ *et* $r \bullet s \sim_a r' \bullet s$ *et* $r' \bullet s \vDash_{\mathcal{K}} \phi$

*Démonstration.*

1. On a $\mathbf{N}^s_a\phi \equiv \mathbf{L}^s_a(\mathbf{M}^s_a\phi)$.

   $\Rightarrow$ On suppose que $r \vDash_{\mathcal{K}} \mathbf{N}^s_a\phi$. On a $r \vDash_{\mathcal{K}} \mathbf{L}^s_a(\mathbf{M}^s_a\phi)$. Soit $r' \in R$ tel que $r \bullet s \downarrow$ et $r' \bullet s \downarrow$ et $r \bullet s \sim_a r' \bullet s$. Alors par définition de la sémantique de $\mathbf{L}$, on a $r' \bullet s \vDash_{\mathcal{K}} \mathbf{M}^s_a\phi$. Or, $\sim_a$ est une relation d'équivalence, donc on a $r' \bullet s \sim_a r' \bullet s$ par réflexivité. Alors, la définition de la sémantique de $\mathbf{M}$ nous indique que $r' \bullet s \vDash_{\mathcal{K}} \phi$.

   $\Leftarrow$ On suppose que pour tout $r' \in R$ tel que $r \bullet s \downarrow$ et $r' \bullet s \downarrow$ et $r \bullet s \sim_a r' \bullet s$, on a $r' \bullet s \vDash_{\mathcal{K}} \phi$. On doit montrer que $r \vDash_{\mathcal{K}} \mathbf{N}^s_a\phi$, c'est à dire que $r \vDash_{\mathcal{K}} \mathbf{L}^s_a(\mathbf{M}^s_a\phi)$. Soit $r''$ tel que $r \bullet s \sim_a r''$. On va montrer que $r'' \vDash \mathbf{M}^s_a\phi$. Soit $r'''$ tel que $r''' \bullet s \downarrow$ et $r'' \sim_a r''' \bullet s$. Alors par transitivité, on a $r \bullet s \sim_a r''' \bullet s$. Donc, par notre hypothèse, on a $r''' \bullet s \vDash_{\mathcal{K}} \phi$. Comme cela est vrai pour tout $r'''$ vérifiant nos conditions, on a bien $r'' \vDash \mathbf{M}^s_a\phi$. Comme cela est vrai pour tout $r''$ vérifiant nos conditions, on a bien $r \vDash_{\mathcal{K}} \mathbf{L}^s_a(\mathbf{M}^s_a\phi)$, c'est à dire $r \vDash_{\mathcal{K}} \mathbf{N}^s_a\phi$.

2. On a $\widetilde{\mathbf{L}}^s_a\phi \equiv \neg\mathbf{L}^s_a(\neg\phi)$.
   Donc $r \vDash_{\mathcal{K}} \widetilde{\mathbf{L}}^s_a\phi$ ssi $r \vDash_{\mathcal{K}} \neg\mathbf{L}^s_a(\neg\phi))$, ssi $r \nvDash \mathbf{L}^s_a(\neg\phi)$,
   ssi $r \bullet s \downarrow$ et il existe $r' \in R$ tel que $r \bullet s \sim_a r'$ et $r' \nvDash_{\mathcal{K}} \neg\phi$,
   ssi $r \bullet s \downarrow$ et il existe $r' \in R$ tel que $r \bullet s \sim_a r'$ et $r' \vDash_{\mathcal{K}} \phi$.
3. Similaire à 2.
4. Similaire à 2. et en utilisant le résultat 1.

□

Notez que nous avons introduit trois opérateurs $\mathbf{L}$, $\mathbf{M}$ et $\mathbf{N}$ où des ressources sont introduites universellement puis des opérateurs alternatifs $\widetilde{\mathbf{L}}$, $\widetilde{\mathbf{M}}$ et $\widetilde{\mathbf{N}}$ où elles sont introduites existentiellement. Cette manière de construire les opérateurs, qui a été notre première approche, semble toutefois ne pas être la plus élégante.

De récentes recherches nous ont orientés vers une autre approche, qui consiste à introduire en premier lieu les opérateurs $\mathbf{L}$ (universel) et $\widetilde{\mathbf{M}}$ (existentiel), puis de proposer les quatre autres opérateurs comme des dérivés de ces deux-là. Cette approche a l'avantage de respecter la dualité universelle - existentielle présente dans la majorité des logiques modales (l'exemple premier étant les opérateurs $\Diamond$ et $\Box$ de la logique modale S4 [70]). Mais ce choix a aussi une justification d'un point de vue de l'expressivité :

- $\widetilde{\mathbf{L}}$ présente peu d'intérêt, vu que l'existence de $r'$ dans sa définition sémantique est toujours garantie par la réflexivité de $\sim_a$ (on a $r \bullet s \sim_a r \bullet s$). En revanche, la version universelle $\mathbf{L}$ permet d'atteindre *toutes* les ressources correspondant à la ressource ambiante $r$ composée avec la ressource locale $s$.
- Similairement, seule la version existentielle $\widetilde{\mathbf{M}}$ présente de l'intérêt puisqu'elle assure que la ressource ambiante $r$ *peut* être décomposée de manière à inclure la ressource locale $s$ (par la relation $r \sim_a r' \bullet s$). Le fait que toutes les décomposition vérifient la propriété est moins intéressante.

Notons d'ailleurs que $\mathbf{L}$ et $\widetilde{\mathbf{M}}$ sont les opérateurs majoritairement utilisés dans les exemples que nous présentons dans la Section 4.4. Enfin, cette manière de mettre en valeur $\mathbf{L}$ et $\widetilde{\mathbf{M}}$ permet un rapprochement avec les deux opérateurs multiplicatifs $*$ et $-\!\!*$. En effet, dans $\mathbf{L}$ comme dans $-\!\!*$, la ressource ambiante est composée par une autre ressource et l'on a l'introduction d'une ressource annexe universellement. De même, dans $\widetilde{\mathbf{M}}$ comme dans $*$, la ressource ambiante est décomposée en sous-ressources, de manière existentielle.

Cette approche différente d'ERL, où $\mathbf{L}$ et $\widetilde{\mathbf{M}}$ sont mis en valeur, sera développée dans de futures publications.

Notons que par le troisième point de la Définition 4.2, comme $\bullet$ étend $\cdot$, toutes les propriétés de $\bullet$ s'étendent à $\cdot$ également ($\cdot$ est commutative, associative et admet $e$ comme élément neutre). Ainsi on a les résultats suivants :

- pour tout $r, s \in Res$ et pour tout $\phi$, $\mathbf{L}^{rs}_a\phi \equiv \mathbf{L}^{sr}_a\phi$,
- pour tout $r \in Res$ et pour tout $\phi$, $\mathbf{L}^{re}_a\phi \equiv \mathbf{L}^{r}_a\phi$,
- pour tout $r, s, t \in Res$ et pour tout $\phi$, $\mathbf{L}^{(rs)t}_a\phi \equiv \mathbf{L}^{r(st)}_a\phi$.

Bien évidemment ces équivalence se retrouvent pour tous les autres opérateurs ayant des ressources internes ($\mathbf{M}^s_a$, $\mathbf{N}^s_a$, $\widetilde{\mathbf{L}}^s_a$, $\widetilde{\mathbf{M}}^s_a$ et $\widetilde{\mathbf{N}}^s_a$).

Enfin, on redonne une notion de validité et de conséquence logique pour notre logique :

**Définition 4.5** (Validité)**.** *On dit qu'une formule $\phi$ de $\mathcal{L}$ est* valide *dans ERL et on note $\vDash_{ERL} \phi$ (ou simplement $\vDash \phi$ si le contexte n'est pas ambigu) lorsque $r \vDash_{\mathcal{K}} \phi$ pour toute ressource r dans tout modèle $\mathcal{K}$.*

*On dit en outre que $\psi$ est une* conséquence logique *de $\phi$ dans ERL et on note $\phi \vDash_{ERL} \psi$ (ou $\phi \vDash \psi$) lorsque $r \vDash_{\mathcal{K}} \phi$ implique $r \vDash_{\mathcal{K}} \psi$ pour toute ressource r de tout modèle $\mathcal{K}$.*

*On dit que deux formules $\psi$ et $\phi$ sont* équivalentes *dans ERL et on note $\phi \equiv_{ERL} \psi$ (ou $\phi \equiv \psi$) lorsqu'on a $\phi \vDash_{ERL} \psi$ et $\psi \vDash_{ERL} \phi$.*

De même que pour LTBI, on peut définir une manière plus directe de prouver la conséquence logique. Toutefois, dans LTBI qui était intuitionniste, cette méthode s'appuyait sur $-\!\!*$. Ici, comme on est dans un cadre classique, $\rightarrow$ remplira ce rôle.

**Théorème 4.1** (Conséquence logique)**.** *Soit deux formules $\phi$ et $\psi$. On a $\phi \vDash \psi$ ssi $\vDash \phi \rightarrow \psi$.*

*Démonstration.*

- On suppose $\phi \vDash \psi$.
  Soit $\mathcal{K}$ un modèle. Soit $r \in R$ tel que $r \vDash_{\mathcal{K}} \phi$. Comme $\phi \vDash \psi$, on a donc également $r \vDash_{\mathcal{K}} \psi$. On a donc $r \vDash_{\mathcal{K}} \phi \rightarrow \psi$ pour cette ressource. Comme cela est vrai pour toute ressource de tout modèle, on a finalement $\vDash \phi \rightarrow \psi$.
- On suppose $\vDash \phi \rightarrow \psi$. Soit $\mathcal{K}$ un modèle. Soit $r \in R$ tel que $r \vDash_{\mathcal{K}} \phi$. Puisque $\vDash \phi \rightarrow \psi$, on a $r \vDash_{\mathcal{K}} \phi \rightarrow \psi$. Par définition, et comme $r \vDash_{\mathcal{K}} \phi$, on obtient que $r \vDash_{\mathcal{K}} \psi$. Comme cela est valable pour toute ressource $r$ de tout modèle $\mathcal{K}$, on a donc $\phi \vDash \psi$.

□

Enfin, on a annoncé en début de section que ERL se voulait une extension des logiques qui lui ont donné le jour. On souhaite ici formaliser ce résultat.

**Proposition 4.2** (ERL extension conservative de BBI et de EL)**.**

1. *On considère le langage $\mathcal{L}_{BBI}$ obtenu en retirant les opérateurs $\mathbf{L}^s_a$, $\mathbf{M}^s_a$, $\mathbf{N}^s_a$, $\widetilde{\mathbf{L}}^s_a$, $\widetilde{\mathbf{M}}^s_a$ et $\widetilde{\mathbf{N}}^s_a$ de $\mathcal{L}$*[6]*. On considère le fragment $ERL_{BBI}$ de ERL obtenu en imposant $A = \emptyset$ et en se restreignant aux opérateurs de $\mathcal{L}_{BBI}$. Si dans tout modèle l'élément neutre pour la composition est l'élément $e \in Res$, alors $ERL_{BBI}$ est équivalent sémantiquement à BBI (c'est à dire que toute formule valide dans l'une est valide dans l'autre et réciproquement).*
2. *On considère le langage $\mathcal{L}_{EL}$ obtenu en retirant les opérateurs $-\!\!*$, $*$ et $\mathbf{I}$ de $\mathcal{L}$ et en remplaçant $\mathbf{L}^s_a$, $\mathbf{M}^s_a$, $\mathbf{N}^s_a$, $\widetilde{\mathbf{L}}^s_a$, $\widetilde{\mathbf{M}}^s_a$ et $\widetilde{\mathbf{N}}^s_a$ par l'opérateur $K_a$ défini par $K_a\phi = \mathbf{L}^e_a\phi = \mathbf{M}^e_a\phi = \mathbf{N}^e_a\phi$. On considère le fragment $ERL_{EL}$ de ERL obtenu en imposant $Res = \{e\}$ et en se restreignant aux opérateurs de $\mathcal{L}_{EL}$. Si les ensembles d'agents sont identiques, alors $ERL_{EL}$ est équivalent sémantiquement à EL (la logique épistémique, dans sa définition traditionnelle comme dans [92]).*

*Démonstration.*
La preuve est triviale étant donné les restrictions données dans la proposition et les définitions respectives de ERL, BBI et EL. □

6. On obtient donc exactement le langage de BBI tel que défini dans le Chapitre 2.

### 4.3.2 ERL* : une variante de la logique ERL

Si la structure d'ERL correspond à ce que nous avons envisagé dans son développement et permet de traduire l'intuition que nous avions des nouveaux opérateurs, il lui manque une propriété fondamentale pour qu'elle génère des propriétés intéressantes auxquelles nous nous attendons. Plutôt que de forcer cette propriété dans notre structure, nous allons définir une légère variante qui sera plus utile pour développer l'expressivité de ERL et que nous nommons ERL$^*$.

**Définition 4.6** (ERL$^*$)**.** *La logique* ERL$^*$ *est une variante de ERL définie par la même syntaxe et la même sémantique à l'exception du point suivant.*

*Pour tout modèle $\mathcal{K}$ de* ERL$^*$*, pour tout $a \in A$, $\sim_a$ est compatible avec $\bullet$. C'est à dire que pour toutes ressources $r, r', s \in R$, si $r \bullet s \downarrow$ et $r \sim_a r'$, alors $r' \bullet s \downarrow$ et $r \bullet s \sim_a r' \bullet s$.*

Dans ce cadre, par la simple application des définitions sémantiques, on peut déduire quelques équivalences simples permettant de manipuler nos nouveaux opérateurs :

**Lemme 4.1.** *Soit $a \in A$, $s,t \in Res$ et $\phi \in \mathcal{L}$ une formule de* ERL$^*$*. On a les égalités suivantes :*

1. $\mathbf{L}^s_a(\mathbf{L}^t_a\phi) \equiv \mathbf{L}^{st}_a\phi$
2. $\mathbf{M}^s_a\phi \vDash \mathbf{M}^t_a(\mathbf{M}^s_a\phi)$
3. $\mathbf{L}^s_a\phi \vDash \mathbf{M}^t_a(\mathbf{L}^s_a\phi)$
4. $\mathbf{L}^e_a\phi \equiv \mathbf{M}^e_a\phi \equiv \mathbf{N}^e_a\phi$
5. $\widetilde{\mathbf{L}}^s_a(\widetilde{\mathbf{L}}^t_a\phi) \equiv \widetilde{\mathbf{L}}^{st}_a\phi$
6. $\widetilde{\mathbf{M}}^t_a(\widetilde{\mathbf{M}}^s_a\phi) \vDash \widetilde{\mathbf{M}}^s_a\phi$
7. $\widetilde{\mathbf{M}}^t_a(\widetilde{\mathbf{L}}^s_a\phi) \vDash \widetilde{\mathbf{L}}^s_a\phi$
8. $\widetilde{\mathbf{L}}^e_a\phi \equiv \widetilde{\mathbf{M}}^e_a\phi \equiv \widetilde{\mathbf{N}}^e_a\phi$

*Démonstration.*

1. Soit $\mathcal{K}$ un modèle et $r \in R$.

   $\Rightarrow$ Supposons que $r \vDash_{\mathcal{K}} \mathbf{L}^s_a(\mathbf{L}^t_a\phi)$. Alors $r \bullet s \downarrow$ et pour tout $r' \in R$ tel que $r \bullet s \sim_a r'$ on a $r' \vDash_{\mathcal{K}} \mathbf{L}^t_a\phi$. C'est à dire que $r \bullet s \downarrow$ et pour tout $r' \in R$ tel que $r \bullet s \sim_a r'$, $r' \bullet t \downarrow$ et pour tout $r'' \in R$ tel que $r' \bullet t \sim_a r''$ on a $r'' \vDash_{\mathcal{K}} \phi$. Soit $r''' \in R$ tel que $r \bullet s \bullet t \sim_a r'''$. Alors par réflexivité on a $r \bullet s \sim_a r \bullet s$. Donc, par ce qu'on vient d'énoncer, en prenant $r' = r \bullet s$ et $r'' = r'''$, on a $r \bullet s \bullet t \downarrow$ et $r''' \vDash_{\mathcal{K}} \phi$. Donc $r \vDash_{\mathcal{K}} \mathbf{L}^{st}_a\phi$. Comme cela est vrai pour tout modèle et toute ressource, on a $\mathbf{L}^s_a(\mathbf{L}^t_a\phi) \vDash \mathbf{L}^{st}_a\phi$.

   $\Leftarrow$ Supposons que $r \vDash_{\mathcal{K}} \mathbf{L}^{st}_a\phi$. Alors $r \bullet s \bullet t \downarrow$ et pour tout $r'''$ tel que $r \bullet s \bullet t \sim_a r'''$, on a $r''' \vDash_{\mathcal{K}} \phi$. Comme $r \bullet s \bullet t \downarrow$, on a $r \bullet s \downarrow$. Soit $r' \in R$ tel que $r \bullet s \sim_a r'$. Alors, comme on est dans ERL$^*$, on a par compatibilité que $r' \bullet t \downarrow$ et $r \bullet s \bullet t \sim_a r' \bullet t$. Soit $r''$ tel que $r' \bullet t \sim_a r''$. Alors par transitivité on a $r \bullet s \bullet t \sim_a r''$. Donc, en prenant $r''' = r''$, on a $r'' \vDash_{\mathcal{K}} \phi$. On a donc montré que $r \bullet s \downarrow$ et pour tout $r' \in R$ tel que $r \bullet s \sim_a r'$, $r' \bullet t \downarrow$ et pour tout $r'' \in R$ tel que $r' \bullet t \sim_a r''$ on a $r'' \vDash_{\mathcal{K}} \phi$. Cela montre que $r \vDash_{\mathcal{K}} \mathbf{L}^s_a(\mathbf{L}^t_a\phi)$. Comme cela est vrai pour tout modèle et toute ressource, on a $\mathbf{L}^{st}_a\phi \vDash \mathbf{L}^s_a(\mathbf{L}^t_a\phi)$.

   Finalement on a $\mathbf{L}^s_a(\mathbf{L}^t_a\phi) \equiv \mathbf{L}^{st}_a\phi$

2. Soit $\mathcal{K}$ un modèle et $r \in R$. On suppose $r \vDash_{\mathcal{K}} \mathbf{M}^s_a\phi$. On a donc pour tout $r'$ tel que $r' \bullet s \downarrow$ et $r \sim_a r' \bullet s$ que $r' \bullet s \vDash_{\mathcal{K}} \phi$. Soit $r''$ tel que $r'' \bullet t \downarrow$ et $r \sim_a r'' \bullet t$. Soit $r'''$ tel que $r''' \bullet s \downarrow$ et $r'' \bullet t \sim_a r''' \bullet s$. Alors par transitivité on a $r \sim_a r''' \bullet s$ et en prenant $r' = r'''$ on a $r''' \bullet s \vDash_{\mathcal{K}} \phi$. Comme cela est vrai pour tout $r'''$ tel que $r''' \bullet s \downarrow$ et $r'' \bullet t \sim_a r''' \bullet s$, on a $r'' \vDash_{\mathcal{K}} \mathbf{M}^s_a\phi$.

Comme cela est vrai pour tout $r''$ tel que $r'' \bullet t \downarrow$ et $r \sim_a r'' \bullet t$, on a $r \vDash_{\mathcal{K}} \mathbf{M}_a^t(\mathbf{M}_a^s\phi)$. Comme cela est vrai pour toute ressource $r$ de tout modèle $\mathcal{K}$, on a bien $\mathbf{M}_a^s\phi \vDash \mathbf{M}_a^t(\mathbf{M}_a^s\phi)$.

Notons que l'on ne peut pas avoir la conséquence logique dans l'autre sens ($\mathbf{M}_a^t(\mathbf{M}_a^s\phi) \nvDash \mathbf{M}_a^s\phi$). En effet, si $r \vDash_{\mathcal{K}} \mathbf{M}_a^t(\mathbf{M}_a^s\phi)$, $\phi$ est validé par tous les $r'' \bullet s$ tels que $r \sim_a r' \bullet t$ et $r' \bullet t \sim_a r'' \bullet s$. Or, pour que $r \vDash_{\mathcal{K}} \mathbf{M}_a^s\phi$, on doit avoir $r''' \bullet s \vDash \phi$ pour tous les $r'''$ tels que $r \sim_a r''' \bullet s$, pas seulement ceux dont l'équivalence par $\sim_a$ est construite par l'intermédiaire de $t$.

3. Soit $\mathcal{K}$ un modèle et $r \in R$. On suppose $r \vDash_{\mathcal{K}} \mathbf{L}_a^s\phi$. On a donc $r \bullet s \downarrow$ et pour tout $r'$ tel que $r \bullet s \sim_a r'$ on a $r' \vDash_{\mathcal{K}} \phi$. Soit $r''$ tel que $r'' \bullet t \downarrow$ et $r \sim_a r'' \bullet t$. Comme $r \bullet s \downarrow$, on a donc par compatibilité de $\bullet$ et $\sim_a$ (Définition 4.6) que $r'' \bullet t \bullet s \downarrow$ et que $r \bullet s \sim_a r'' \bullet t \bullet s$. Soit $r'''$ tel que $r'' \bullet t \bullet s \sim_a r'''$. Par transitivité de $\sim_a$, on a alors $r \bullet s \sim_a r'''$. Alors par hypothèse, on a $r''' \vDash_{\mathcal{K}} \phi$. Par conséquent, on a montré que $r'' \bullet t \vDash_{\mathcal{K}} \mathbf{L}_a^s\phi$ et par suite que $r \vDash_{\mathcal{K}} \mathbf{M}_a^t(\mathbf{L}_a^s\phi)$. Comme cela est vrai pour toute ressource $r$ de tout modèle $\mathcal{K}$, on a bien $\mathbf{L}_a^s\phi \vDash \mathbf{M}_a^t(\mathbf{L}_a^s\phi)$

Pour des raisons similaires à 2., la conséquence logique inverse n'est pas vraie.

4. Soit $\mathcal{K}$ un modèle et $r \in R$. $r \vDash_{\mathcal{K}} \mathbf{L}_a^e\phi$ ssi $r \bullet e \downarrow$ et pour tout $r' \in R$ tel que $r \bullet e \sim_a r'$, $r' \vDash_{\mathcal{K}} \phi$. Or $r \bullet e$ est toujours défini et $r \bullet e \sim r$. Par compatibilité de $\sim_a$ et $\sim$, on a donc $r \vDash_{\mathcal{K}} \mathbf{L}_a^e\phi$ ssi pour tout $r' \in R$ tel que $r \sim_a r'$, $r' \vDash_{\mathcal{K}} \phi$. Par un raisonnement similaire, on montre que $r \vDash_{\mathcal{K}} \mathbf{M}_a^e\phi$ ssi pour tout $r' \in R$ tel que $r \sim_a r'$, $r' \vDash_{\mathcal{K}} \phi$ et que $r \vDash_{\mathcal{K}} \mathbf{N}_a^e\phi$ ssi pour tout $r' \in R$ tel que $r \sim_a r'$, $r' \vDash_{\mathcal{K}} \phi$. Finalement on a $r \vDash_{\mathcal{K}} \mathbf{L}_a^e\phi$ ssi $r \vDash_{\mathcal{K}} \mathbf{M}_a^e\phi$ ssi $r \vDash_{\mathcal{K}} \mathbf{N}_a^e\phi$, d'où le résultat.

5. On développe la définition de $\widetilde{\mathbf{L}}$ : $\widetilde{\mathbf{L}}_a^s(\widetilde{\mathbf{L}}_a^t\phi) \equiv \neg\mathbf{L}_a^s(\neg\widetilde{\mathbf{L}}_a^t\phi) \equiv \neg\mathbf{L}_a^s(\neg\neg\mathbf{L}_a^t\neg\phi)$. On utilise alors l'égalité $\neg\neg\phi \equiv \phi$ que nous n'avons pas montrée formellement ici, mais qui est triviale vu nos définitions. On a donc $\widetilde{\mathbf{L}}$ : $\widetilde{\mathbf{L}}_a^s(\widetilde{\mathbf{L}}_a^t\phi) \equiv \neg\mathbf{L}_a^s(\mathbf{L}_a^t\neg\phi)$. Par l'égalité 1., on a alors $\widetilde{\mathbf{L}}_a^s(\widetilde{\mathbf{L}}_a^t\phi) \equiv \neg\mathbf{L}_a^{st}\neg\phi$, et donc par définition $\widetilde{\mathbf{L}}_a^s(\widetilde{\mathbf{L}}_a^t\phi) \equiv \widetilde{\mathbf{L}}_a^{st}\phi$.

6. On commence par déplier l'opérateur comme en 4. On a $\widetilde{\mathbf{M}}_a^t(\widetilde{\mathbf{M}}_a^s\phi) \vDash \neg\mathbf{M}_a^t(\neg\widetilde{\mathbf{M}}_a^s\phi) \vDash \neg\mathbf{M}_a^t(\neg\neg\mathbf{M}_a^s\neg\phi) \vDash \neg\mathbf{M}_a^t(\mathbf{M}_a^s\neg\phi)$. Soit $\mathcal{K}$ un modèle et $r \in R$. On suppose que $r \vDash_{\mathcal{K}} \widetilde{\mathbf{M}}_a^t(\widetilde{\mathbf{M}}_a^s\phi)$. Alors, comme on vient de montrer, on a $r \vDash_{\mathcal{K}} \neg\mathbf{M}_a^t(\mathbf{M}_a^s\neg\phi)$. Donc $r \nvDash_{\mathcal{K}} \mathbf{M}_a^t(\mathbf{M}_a^s\neg\phi)$. Par 2., on a alors que $r \nvDash_{\mathcal{K}} \mathbf{M}_a^s\neg\phi$, donc $r \vDash_{\mathcal{K}} \neg\mathbf{M}_a^s\neg\phi$, c'est à dire que $r \vDash_{\mathcal{K}} \widetilde{\mathbf{M}}_a^s\phi$. Finalement, comme cela est vrai pour toute ressource $r$ de tout modèle $\mathcal{K}$, on a $\widetilde{\mathbf{M}}_a^t(\widetilde{\mathbf{M}}_a^s\phi) \vDash \widetilde{\mathbf{M}}_a^s\phi$.

Encore une fois, la conséquence inverse est fausse, cela est encore plus compréhensible ici. En effet, s'il existe $r'$ tel que $r \sim_a r' \bullet s$, cela ne garanti par l'existence d'un $r''$ intermédiaire tel que $r \sim_a r'' \bullet t \sim_a r' \bullet s$.

7. Preuve similaire à 6., en utilisant les définitions des opérateurs et la relation 3.

8. De manière similaire à 5., on a $\widetilde{\mathbf{L}}_a^e\phi \equiv \neg\mathbf{L}_a^e\neg\phi$, puis par 4. $\widetilde{\mathbf{L}}_a^e\phi \equiv \neg\mathbf{M}_a^e\neg\phi$ et enfin $\widetilde{\mathbf{L}}_a^e\phi \equiv \widetilde{\mathbf{M}}_a^e\phi$. L'autre égalité se prouve de la même manière.

□

Ce lemme nous indique quelques propriétés intéressantes des opérateurs propres à ERL. En particulier, le résultat 1. (et le 5. qui lui est associé) met en lumière un phénomène qu'on était en droit d'attendre : l'accès successif à travers la modalité $\mathbf{L}$ par une ressource $s$ puis une ressource $t$ est équivalent à un accès direct par la ressource composée $st$.

Les résultats 2. et 6. sont plus étonnants. Ils semblent indiquer que lors d'une composition de plusieurs opérateurs **M**, seul le plus interne est pertinent. Toutefois, dans chacun des cas un seul des sens de l'équivalence logique est vrai. Cela est dû au fait que le sens inverse nécessite l'introduction ex nihilo d'une ressource interagissant avec $t$, ce qui n'est pas possible.

Enfin, les égalités 4. et 8. sont assez évidentes, elles mettent en lumière la place particulière de $e$ comme ressource neutre : quel que soit le mode d'accès, utiliser $e$ comme ressource indique qu'aucune ressource particulière n'est nécessaire, donc les 3 opérateurs deviennent équivalent. Notons d'ailleurs que c'est $e$ qui est utilisée pour signifier le simple accès épistémique lorsque l'on veut transformer ERL en EL.

## 4.4 Expressivité des logiques ERL et ERL*

Nous allons maintenant explorer plusieurs manières d'exploiter ERL pour modéliser des problèmes concrets. Cela permettra également de mieux cerner l'usage de chacun des opérateurs. Nous commençons par développer un long exemple dans lequel nous illustrons plusieurs utilisations d'ERL$^*$ avant de compléter cette exploration par quelques autre exemples plus court où nous traiterons quelques problèmes traditionnels du contrôle d'accès.

### 4.4.1 Problème de la porte de Schneier [88]

Dans ce premier exemple, nous allons utiliser le problème de la porte de Schneier [88] pour exposer l'emploi des opérateurs d'ERL$^*$, épistémiques comme multiplicatifs.

#### Présentation du problème

Le problème de la porte de Schneier est un problème concret très simple qui met en lumière la perte de pertinence du raisonnement sur un système de sécurité quand des détails physiques viennent complètement contredire la politique de sécurité théorique mise en place.

Plus concrètement, l'exemple que nous prenons est celui d'un espace dont l'accès est réservé à un certain nombre de personnes (les employés d'une entreprise par exemple). Pour pénétrer dans cet espace, les utilisateurs doivent utiliser un système de contrôle d'accès (nous parlerons ici de jeton, mais il s'agira concrètement d'une clé, d'une carte d'accès ou d'un code) pour ouvrir une barrière. Ainsi, théoriquement la politique de sécurité de ce système interdit qu'une personne non autorisée (et donc ne possédant pas le jeton) ne soit présent dans l'enceinte de l'espace réservé.

Toutefois, nous nous intéressons au cas où la barrière elle-même peut être facilement contournée. Par exemple, une barrière située sur une route entourée de terre peut être contournée si les usagers quittent la route en question (que ce soit avec de mauvaises intentions ou juste pour rendre leur entrée plus rapide et moins compliquée), comme illustré dans la Figure 4.1.

On est donc devant un exemple simple et flagrant d'un système de sécurité qui, bien que parfaitement acceptable sur le plan théorique, est rendu faillible par son implémentation sur le terrain.

Nous allons maintenant voir comment ERL$^*$ peut non seulement être un outil pour décrire des systèmes d'accès comme celui-ci, mais encore nous permet de faire la distinction entre les


FIGURE 4.1 – Illustration humoristique du problème de la porte de Schneier

spécifications théoriques du système et son implémentation physique.

**Modélisation avec ERL**

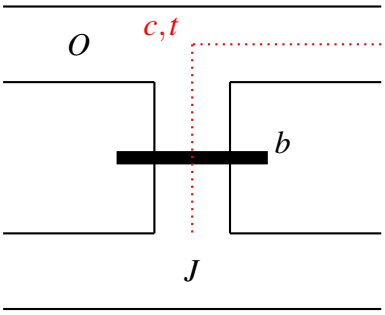



FIGURE 4.2 – Problème de la barrière, cas de base

Nous allons commencer par décrire un modèle simple, puis nous allons progressivement l'enrichir pour le raffiner. Notre modèle de base, tel qu'illustré par la Figure 4.2, consiste à simplement décrire l'entrée d'un véhicule par la barrière à l'aide d'un jeton (qui, dans le cas concret, peut être n'importe quel dispositif de contrôle d'accès). Dans un premier, nous ne distinguerons pas un usager du système d'un autre, et nous ne considérerons pas non plus le défaut du système décrit dans le paragraphe précédent.

Pour bénéficier de l'expressivité supplémentaire qu'elle offre (notamment les égalités décrites dans le Lemme 4.1), on préfèrera employer la logique ERL$^*$ pour modéliser le système. Nous utiliserons les ressources pour décrire les différentes entités du système et les propositions atomiques pour décrire des propriétés de ce système. On considère l'ensemble de ressources

syntaxiques $Res = \{e,t,b\}$ : $t$ représentera le jeton d'accès, $b$ nous servira à marquer la présence et le bon usage de la barrière, et enfin $e$ est la ressource unité. Comme nous ne distinguons pour le moment aucun usager d'un autre, nous nous contenterons d'un ensemble d'agents $A = \{a\}$ où $a$ représente un agent arbitraire. Enfin, nous considérons deux propositions $O$ et $J$ ($O,J \in Prop$) qui représenteront respectivement le fait de se trouver à l'extérieur ou à l'intérieur de la zone sécurisée, ainsi si une ressource $c$ modélise une voiture de notre système, $c \vDash O$ signifiera que la voiture est à l'extérieur et $c \vDash J$ qu'elle est à l'intérieur.

D'un point de vue modélisation, les ressources que nous avons proposées ici sont de natures différentes : $t$ est un jeton matériel (une clé, une carte ou un code par exemple), $c$ est une voiture et $b$ est un simple marqueur qui indique la présence d'une barrière fonctionnelle. Cette diversité soulève notamment la question de la nature de la ressource unitaire $e$. Nous contournons ce problème en acceptant le fait que les ressources recouvrent une variété d'objets différents, mais nous pouvons également utiliser la nature épistémique de notre logique pour considérer que les ressources ne représentent pas réellement des objets eux-mêmes mais plutôt la connaissance qu'un objet est présent dans notre système.

Par exemple, $c$ peut être vu comme un marqueur abstrait qui indique la présence d'une voiture et $t$ la présence d'un dispositif d'accès dans cette voiture. Ainsi les ressources agissent comme un niveau d'abstraction de notre système. Dans cet optique, il est aisé de voir $e$ comme l'absence d'information.

On établit la propriété suivante :

$$O \to \mathbf{L}_a^{bt} J$$

Selon notre sémantique, $c \vDash O \to \mathbf{L}_a^{bt} J$ dans le cas où, si $c \vDash O$, alors pour tout $c'$ tel que $c \bullet b \bullet t \sim_a c'$, $c' \vDash J$. Ainsi, si une voiture est à l'extérieur avec le jeton d'accès et le marqueur de la barrière, alors toute ressource accessible est à l'intérieur. La combinaison de nos deux jetons ($b$ et $t$) garantit ainsi l'accès à l'intérieur. L'usage du jeton $b$ pour marquer la présence de la barrière nous permet de modéliser aisément une situation où la barrière est défectueuse ou complètement verrouillée. Notons que les formules $O \to \mathbf{L}_a^t J$, $O \to \mathbf{L}_a^b J$, et $O \to \mathbf{L}_a^e J$ ne sont pas valides car nous ne pouvons accéder à l'espace sécurisé si la barrière est verrouillée, si l'on ne possède pas de jeton d'accès ou les deux.

L'utilisation de l'opérateur $\mathbf{L}$ dans cette situation est significative. Considérons ce qu'impliquerait l'usage d'un des autres opérateurs à la place. Si nous affirmons $O \to \mathbf{M}_a^{bt} J$, cela signifierait que quiconque est à l'extérieur peut rentrer (sans condition) et acquérir les deux jetons d'accès. Ce n'est évidemment pas ce que nous attendions. D'un autre côté, utiliser $\mathbf{N}$ a un effet plus intéressant. $O \to \mathbf{N}_a^{bt} J$ exige non seulement que l'agent entrant ait les jetons attendus pour entrer, mais aussi que ces jetons restent présent ou actif une fois l'agent à l'intérieur. C'est une approche légèrement différente de notre première, dans laquelle nous n'avions aucune information quant à l'état des jetons une fois l'agent entré.

Nous pouvons aussi discuter la pertinence de l'emploi de l'implication additive $\to$ dans notre modélisation comparé à celui de l'implication multiplicative $-\!\!*$. Pour une première approche, $\to$ semble suffisant. En effet, si on affirme que $O \to \mathbf{L}_a^{bt} J$ est valide, alors toute ressource la satisfait. Donc dès qu'on a une ressource $c$ qui satisfait $O$, alors on a $c \vDash O \to \mathbf{L}_a^{bt} J$ et on obtient $c \vDash \mathbf{L}_a^{bt} J$ que l'on attendait. Toutefois, si l'on considère des propriétés plus complexes, la

situation est différente.

Imaginons par exemple que l'environnement soit composé non seulement de $c$ mais d'autres informations ou éléments que nous regrouperons sous une ressource $o$. Notre contexte épistémique serait alors $o \bullet c$. Alors, dans le cas où $c \vDash O$, l'utilisation de la propriété $O \to \mathbf{L}_a^{bt} J$ nous amènerait à nouveau $c \vDash \mathbf{L}_a^{bt} J$. On ne peut inclure le contexte $o$ dans cette proposition car on n'a pas $o \bullet c \vDash O$. L'usage de $\to$ nous interdit ainsi de travailler sur une partie du contexte épistémique seulement tout en incluant les autres ressources.

A contrario, si l'on utilise $O \mathrel{-\!\!*} \mathbf{L}_a^{bt} J$, qu'on admet valide, alors on a notamment $o \vDash O \mathrel{-\!\!*} \mathbf{L}_a^{bt} J$, ce qui, combiné avec la proposition $c \vDash O$ nous amène bien $o \bullet c \vDash \mathbf{L}_a^{bt} J$. Ainsi l'utilisation de l'opérateur multiplicatif $\mathrel{-\!\!*}$ est plus pertinent dans des systèmes complexes, puisqu'il nous permet d'agir sur une partie du contexte seulement tout en conservant les autres ressources dans notre champ d'action.

### Ajouts d'agents

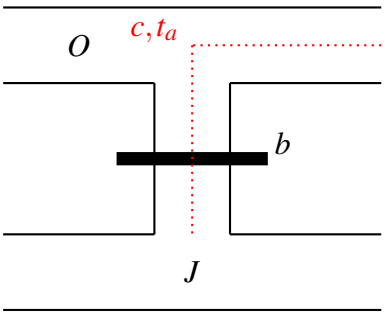



FIGURE 4.3 – Problème de la barrière avec agents

Nous souhaitons maintenant introduire plus de précision dans notre modèle en introduisant plusieurs agents (comme illustré par la Figure 4.3). En effet, on peut constater que dans notre première approche nous n'avions fait aucune distinction entre les différents utilisateurs du système : nous disposons d'une unique ressource $t$ pour symboliser les dispositifs d'accès de tous les utilisateurs, ce qui revient à considérer qu'ils sont tous équivalents (par exemple le même code pour tous les usagers).

Cette simplification n'est pas habile, d'autant qu'une des volontés d'un système de contrôle d'accès est de donner la possibilité de distinguer plusieurs politiques d'accès en fonction des usagers. En outre, notre logique étant construite autour d'un accès dépendant d'agents, elle semble toute appropriée pour développer une modélisation plus précise allant dans ce sens.

Nous allons incorporer trois agents distincts au système (ce nombre est arbitraire, mais on comprendra aisément comment la démarche peut être étendue à un nombre arbitraire d'agents). Nous redéfinissons donc l'ensemble des agents $A$ par $A = \{\alpha, \beta, \gamma\}$. Chacun de ces agents, modélisant l'un des utilisateurs du système, doit posséder son dispositif d'accès propre qu'il est le seul à pouvoir utiliser. Pour cela, nous modifions notre ensemble de ressources internes *Res*

de la manière suivante : $Res = \{e, b, t_\alpha, t_\beta, t_\gamma\}$. Ainsi, nous n'avons plus un seul jeton d'accès $t$ mais un pour chaque agent.

Il nous faut maintenant réécrire notre formule exprimant l'accès à l'espace sécurisé en incluant ces nouveautés. On postule que la formule $O \rightarrow \mathbf{L}_a^{bt_a} J$ est valide pour tout agent $a \in A$. Ainsi, par exemple, $O \rightarrow \mathbf{L}_\alpha^{bt_\alpha} J$ est valide, ce qui exprime que $\alpha$ peut rentrer avec son jeton $t_\alpha$, mais $O \rightarrow \mathbf{L}_\alpha^{bt_\beta} J$ ne l'est pas, ce qui veut dire que $\alpha$ ne peut pas utiliser le jeton de $\beta$.

### Agents frauduleux

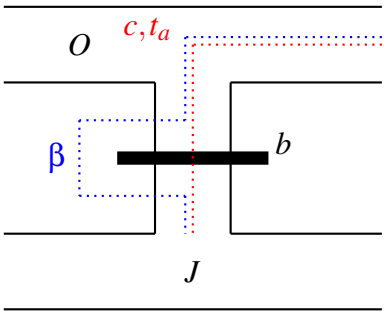



FIGURE 4.4 – Problème de la barrière avec contournement

Nous avons maintenant une représentation acceptable de notre système de base : nous avons modélisé le fait que chaque agent peut accéder à l'espace sécurisé moyennant qu'il dispose du jeton d'accès approprié. Nous allons à présent nous recentrer sur le problème initial que nous voulions aborder, celui de la porte de Schneier. Nous voulons modéliser le fait que le système est potentiellement défaillant et qu'une faille physique (comme le possible contournement de la barrière) le remet en cause.

Contourner la barrière revient simplement à considérer qu'on accède à l'espace sécurisé sans avoir aucun jeton. On peut modéliser plus finement en considérant que certains agents seulement connaissent le raccourci (ou osent l'utiliser) et que les autres ne peuvent pas l'utiliser (comme illustré dans la Figure 4.4).

Supposons par exemple que l'agent $\beta$ du paragraphe précédent soit un tel agent frauduleux. Notre nouvel ensemble de propriétés devient le suivant :

$$\begin{cases} O \rightarrow \mathbf{L}_a^{bt_a} J \text{ (pour tout } a \in A) \\ O \rightarrow \mathbf{L}_\beta^{e} J \end{cases}$$

Ici, la ressource $e$ exprime un accès direct (sans aucune ressource nécessaire). On peut mettre l'accent sur le fait que l'usage d'agents nous permet d'exprimer aisément différentes politiques de sécurité de notre modèle, qu'elles soient intentionnelles ou inhérentes à l'implémentation physique.

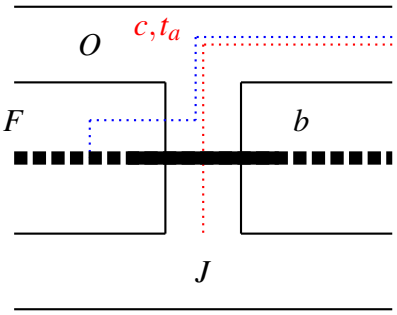



FIGURE 4.5 – Problème de la barrière avec une clôture

### Ajout d'une clôture

On peut raisonnablement supposer qu'un système avec un tel défaut serait rapidement corrigé, par exemple en installant une clôture qui empêcherait de contourner la barrière (voir Figure 4.5). On peut, évidemment, modéliser ceci simplement en retirant notre dernier ajout et revenir à la politique souhaitée, mais il parait plus intéressant de le représenter par une formule. On pourrait alors aussi décrire un défaut dans cette clôture (ou sa suppression). Pour cela, on peut simplement ajouter une proposition atomique $F$ qui est valide pour n'importe quelle ressource à la condition qu'il y ait une barrière empêchant le passage des agents frauduleux. Notre système devient alors :

$$\begin{cases} O \to \mathbf{L}_a^{bt_a} J \text{ (pour tout } a \in A) \\ O \wedge \neg F \to \mathbf{L}_\beta^e J \end{cases}$$

### Preuve de propriétés

Maintenant que nous avons établi un système de formules qui décrit notre situation assez clairement, nous pouvons rechercher des propriétés du modèle. L'idée est d'établir des propriétés du système qui sont plus que des conséquences directes de sa définition. Par exemple, on peut vouloir vérifier que chaque agent à l'intérieur de l'espace sécurisé a en sa possession un jeton d'accès. Cela veut dire qu'on doit vérifier qu'on a pour tout agent $a \in A$ la propriété suivante :

$$J \to \widetilde{\mathbf{M}}_a^{bt_a} J$$

Effectivement, si $c \vDash J \to \widetilde{\mathbf{M}}_a^{bt_a} J$ cela veut dire que si $c \vDash J$, alors il existe $c' \in R$ tel que $c \sim_a c' \bullet b \bullet t_a$ et $c' \bullet b \bullet t_a \vDash J$, ce qui exprime que chaque ressource représentant une voiture qui est à l'intérieur doit être équivalente, pour un certain agent $a$, à une ressource qui est à l'intérieur *et* qui est composée à la fois du jeton approprié $t_a$ et du jeton de la barrière $b$. C'est exactement ce que nous voulions capturer.

Remarquons que cette propriété en particulier n'est *pas* vérifiée dans le système que nous avons décrit dans le dernier paragraphe. En effet, comme nous avons remarqué plus tôt, spécifier l'entrée avec $r \vDash O \to \mathbf{L}_a^{bt_a} J$ fait que $J$ est vérifiée par n'importe quelle ressource $r'$ telle que

$r \bullet b \bullet t_a \sim_a r'$. On voit qu'il n'existe aucune ressource $r''$ telle que $r' \sim_a r'' \bullet b \bullet t_a$ et $r'' \bullet b \bullet t_a \vDash J$ ($r$ vérifie la première propriété, mais pas la deuxième).

Toutefois, l'usage de $\mathbf{N}$ résout ce problème : si $r \vDash O \to \mathbf{N}_a^{bt_a} J$, alors pour tout $r'$ tel que $r \bullet b \bullet t_a \sim_a r' \bullet b \bullet t_a$ on a $r' \bullet b \bullet t_a \vDash J$ et on satisfait bien la propriété demandée.

**Pertinence des logiques de séparation**

Jusque maintenant, nous avons seulement considéré une situation simple, c'est à dire une seule voiture traversant la barrière dans diverses situations. Bien entendu, nous pouvons souhaiter considérer des modèles plus complexes et établir des propriétés similaires.

Par exemple, nous pouvons vouloir voir ce qui arrive lorsque plusieurs voitures sont modélisées ensemble dans le système. Nous avons d'ors et déjà l'ensemble des propriétés sous la forme des implications que nous avons énoncées auparavant. Pour dire qu'une voiture est dans le système, nous avons simplement à indiquer que la proposition $O$ est valide. Alors, en observant la sémantique de nos formules, nous créons une ressource $c$ qui satisfait cette formule.

Pour avoir plusieurs voitures, on peut être d'abord tentés d'écrire quelque chose comme $O \wedge O \wedge O$ (pour trois voitures). Toutefois, comme nous l'avons déjà exposé dans le Chapitre 2, l'usage d'opérateurs additifs ne nous permet pas de caractériser correctement des objets distincts. En effet, d'après notre sémantique, on a $O \wedge O \wedge O \equiv O$, ce qui est ennuyant pour notre modélisation.

Nous devons à la place utiliser des opérateurs additifs et utiliser la formule $O * O * O$. Alors, pour satisfaire cette formule, nous avons effectivement besoin de trois ressources $c_1, c_2, c_3$ et nous avons $c_1 \bullet c_2 \bullet c_3 \vDash O * O * O$. Alors, en utilisant $\mathrel{-\!\!*}$ comme nous avons déjà exposé auparavant, nous pouvons caractériser l'entrée des voitures autorisées à l'intérieur. Ainsi, l'usage des opérateurs des logiques de séparation et de $*$ en particulier permet de modéliser plusieurs instances d'un même objet.

### 4.4.2 Accès conjoint

L'un des problèmes les plus fréquents pour le contrôle d'accès est celui des accès joints [31]. La situation typique est celle d'une ressource ou d'un système partagé par plusieurs utilisateurs et dont l'accès ne peut être réalisé que si chacun des utilisateurs utilise un dispositif d'accès qui lui est propre.

Pour illustrer cela, on se propose de modéliser un exemple très simple. La toile de fond de cet exemple est une situation trouvée dans de nombreux films de l'époque de la guerre froide : un système critique (typiquement le lancement d'un appareil nucléaire) est sécurisé par deux clés différentes, chacune détenue par une opérateur différent. Pour que le système se déverrouille, il est nécessaire que les deux opérateurs activent leurs clés simultanément.

Certains choix de modélisation sont assez évidents : nous avons besoin de deux agents et de deux ressources représentant leurs clés. On pose donc $A = \{a_1, a_2\}$ et $Res = \{k_1, k_2, e\}$. De manière intuitive, $k_i$ est la clé associée à l'agent $a_i$, mais cela sera spécifié plus formellement par les formules que nous allons établir.

Nous devons exprimer le fait que chaque agent utilise sa clé. Bien entendu, le but de cet exemple est d'illustrer comment deux accès séparés mais simultanés déverrouillent le système,

FIGURE 4.6 – Exemple d'accès joint dans le film *Wargames* (1983)

ainsi chaque utilisation d'une clé doit être modélisé par une formule différente. On propose les formules suivantes :

$$\widetilde{\mathbf{M}}_{a_1}^{k_1} \top$$

$$\widetilde{\mathbf{M}}_{a_2}^{k_2} \top$$

On utilise la formule atomique $\top$ puisque nous n'avons pas besoin d'un accès à une propriété en particulier, seulement de mettre à jour les mondes accessibles à $a_1$ et $a_2$ pour signifier que $k_1$ et $k_2$ sont maintenant accessibles. Prenons $\widetilde{\mathbf{M}}_{a_1}^{k_1} \top$ par exemple. Si $r \vDash \widetilde{\mathbf{M}}_{a_1}^{k_1} \top$, alors il existe une ressource $r'$ telle que $r \sim_{a_1} r' \bullet k_1$ et $r' \bullet k_1 \vDash \top$. Cette dernière affirmation étant toujours valide, il nous reste seulement qu'il existe $r'$ telle que $r \sim_{a_1} r' \bullet k_1$. Nous avons donc établi une formule qui exprime que $a_1$ peut atteindre un état où $k_1$ est activée. La seconde formule établit la même chose pour $a_2$ et $k_2$.

À présent, nous devons dire que lorsque les deux clés sont présentes, le système peut être débloqué. On peut penser à une formule comme $\mathbf{M}_{a_1}^{k_1 k_2} D$ où $D$ est une formule atomique qui exprime que le système est déverrouillé. Toutefois, nous voyons que ce choix est problématique. En effet, cette formule dépend de $a_1$, mais le principe même de l'accès joint est que aucun des deux agents impliqués n'est responsable lui-même de l'activation du dispositif. En outre, si l'on utilise cette formule, cela ne marche pas vraiment puisque $k_2$ est amené dans le système par $a_2$ et que la formule ne concerne que $a_1$. Bien évidemment, utiliser $a_2$ à la place de $a_1$ entraîne un problème similaire (le modèle est symétrique).

Ainsi, il semble que notre modélisation nécessite au moins un agent supplémentaire. Nous introduisons donc un agent omnipotent $o$ (on a donc $A = \{a_1, a_2, o\}$). L'idée est d'avoir un agent qui puisse voir et utiliser tout ce à quoi $a_1$ et $a_2$ ont accès, sans que les deux ne partagent de connaissance ou de potentiel d'action. Cet agent peut être interprété soit comme une autorité globale, soit plus simplement comme une modélisation du système matériel lui-même (par exemple l'ordinateur qui accepte les clés et déverrouille l'appareil). Dès lors, avec cet agent supplémentaire, $\mathbf{M}_{o}^{k_1 k_2} D$ semble être un candidat acceptable pour modéliser le déverrouillage du système. Cela veut dire que tout état atteignable par $o$ qui contient $k_1$ et $k_2$ déclenche le déblocage.

Toutefois, nous devons encore exprimer les capacités de $o$, notamment en ce qui concerne son omnipotence. Plus précisément, nous devons expliquer comment toute propriété affectant la connaissance ou le potentiel d'un agent doit aussi affecter $o$. Pour cela nous introduisons une famille de formule comme suit :

$$\forall a \in A,\ \forall s \in Res,\ \forall \phi \in \mathcal{L},\ \widetilde{\mathbf{M}}^s_a \phi \rightarrow \widetilde{\mathbf{M}}^s_o \phi$$

On a bien décrit le fait que toute propriété d'un agent est transférée vers $o$. Seules les propriétés utilisant l'opérateur $\widetilde{\mathbf{M}}$ sont concernées ici. Bien évidemment on pourrait généraliser à d'autres types d'opérateurs, mais dans le cas présent seul celui-ci est utilisé.

Enfin, pour que notre système fonctionne, nous devons activer les deux clés simultanément. Une première approche serait de concaténer les deux formules d'activation par un $\wedge$ : $\widetilde{\mathbf{M}}^{k_1}_{a_1}\top \wedge \widetilde{\mathbf{M}}^{k_2}_{a_2}\top$. Cela ne produit pas le bon résultat. En effet, si $r \vDash \widetilde{\mathbf{M}}^{k_1}_{a_1}\top \wedge \widetilde{\mathbf{M}}^{k_2}_{a_2}\top$, on a $r \sim_a r' \bullet k_1$ et $r \sim_b r'' \bullet k_2$ et nous ne faisons pas apparaître la combinaison de $k_1$ et $k_2$ que nous attendions. En réalité, la bonne manière d'obtenir ce que nous voulons est d'utiliser $r \vDash \widetilde{\mathbf{M}}^{k_1}_{a_1}\top * \widetilde{\mathbf{M}}^{k_2}_{a_2}\top$. Ainsi les deux contextes faisant apparaître $k_1$ et $k_2$ seront connectés par le $\bullet$ généré par $*$. Non seulement cette formule nous permet d'obtenir le résultat attendu, mais elle confirme l'usage intuitif de l'opérateur multiplicatif $*$ : ce dernier est adapté pour la modélisation de deux évènements indépendants et concourant.

Nous avons donc modélisé notre situation avec le système suivant :

1. $\forall a \in A,\ \forall s \in Res,\ \forall \phi \in \mathcal{L},\ \widetilde{\mathbf{M}}^s_a \phi \rightarrow \widetilde{\mathbf{M}}^s_o \phi$
2. $\widetilde{\mathbf{M}}^{k_1}_{a_1}\top * \widetilde{\mathbf{M}}^{k_2}_{a_2}\top$
3. $\mathbf{M}^{k_1 k_2}_o U$

Nous pouvons vérifier que cet enchaînement de formules modélise bien la situation souhaitée. Soit $r$ une ressource qui vérifie 2. et 3. En développant 3., on obtient :

$$\forall r' \text{ tel que } r \sim_o r' \bullet k_1 \bullet k_2,\ r' \bullet k_1 \bullet k_2 \vDash U$$

D'autre part, le développement de 2. produit :

$$\exists r_1, r_2 \text{ tel que } r = r_1 \bullet r_2 \text{ et } r_1 \vDash \widetilde{\mathbf{M}}^{k_1}_{a_1}\top \text{ et } r_2 \vDash \widetilde{\mathbf{M}}^{k_2}_{a_2}\top$$

On peut alors instancier deux fois 1., avec $a = a_1$, $s = k_1$ et $\phi = \top$ puis avec $a = a_2$, $s = k_2$ et $\phi = \top$. On obtient :

$$\exists r_1, r_2 \text{ tel que } r = r_1 \bullet r_2 \text{ et } r_1 \vDash \widetilde{\mathbf{M}}^{k_1}_{o}\top \text{ et } r_2 \vDash \widetilde{\mathbf{M}}^{k_2}_{o}\top$$

De quoi on peut tirer, en développant :

$$\exists r_1, r_2, r'_1, r'_2 \text{ tel que } r = r_1 \bullet r_2 \text{ et } r_1 \sim_o r'_1 \bullet k_1 \text{ et } r_2 \sim_o r'_2 \bullet k_2$$

Alors, puisqu'on travaille dans ERL$^*$, par applications successives de la compatibilité de $\bullet$ et $\sim_o$, on obtient que $r \sim_o r'_1 \bullet k_1 \bullet r_2$ puis $r \sim_o r'_1 \bullet k_1 \bullet r'_2 \bullet k_2$ ce qui veut dire, par commutativité, $r \sim_o r'_1 \bullet r'_2 \bullet k_1 \bullet k_2$. On a alors $r'_1 \bullet r'_2 \bullet k_1 \bullet k_2 \vDash U$ par réutilisation de 3., ce qui est ce que nous cherchions à montrer.

On a donc réussi à modéliser l'accès joint à un système, c'est à dire son activation par la présence simultanée de deux agents avec leurs clés respectives. Cet exemple met particulièrement en avant à la fois l'utilisation des opérateurs multiplicatifs pour la description de contextes séparés réunis, mais aussi la notion d'héritage de propriété d'un agent à un autre (ici, les agents particuliers $a_1$ et $a_2$ et l'agent omnipotent $o$). Cette dernière propriété sera développée plus en avant dans l'exemple 1.4.5 qui traite de la modélisation par couche.

### 4.4.3 Sémaphores

Un autre problème très commun dans l'étude du contrôle d'accès est celui des accès concurrents [31]. De même manière que dans l'exemple précédent, il s'agit de systèmes accessibles par plusieurs agents, mais on veut cette fois, au lieu de forcer l'intervention concourante des agents, l'interdire. On souhaite, en effet, traiter des ressources qui ne peuvent être accessible qu'à un agent à la fois et gérer la potentielle concurrence entre agents.

De nombreux exemples de cette situation sont rencontrés dans tous les systèmes d'information, particulièrement de nos jours où le partage d'information est devenu crucial. Une instance particulière du problème est le partage de ressources (processeur, espace mémoire, périphériques, etc...) par des programmes lancés en parallèle sur un système informatique. Cet exemple est particulièrement pertinent dans la mesure où, comme nous l'avons expliqué plus tôt, la logique de séparation, qui est basée sur BBI, est un outil puissant pour raisonner sur les programmes et leur manière de gérer la mémoire [47,75,87]. Utiliser ERL pour modéliser une concurrence d'accès mémoire semble donc tout approprié.

Le problème est donc de modéliser un multi-processeur (ou une série de systèmes en réseau) dans lequel deux programmes souhaitent accéder à une même ressource. On utilise pour gérer l'accès concurrent le dispositif très simple des *sémaphores*, introduit par Dijkstra [39] : une variable marque la disponibilité ou non de la ressource en question. Un programme qui réserve la ressource vérifie que la variable est à 1, la fait passer à 0, puis de nouveau à 1 lorsqu'il la libère. La section de programme qui nécessite l'accès à la ressource partagée est appelée *section critique*.

On modélise simplement cette situation avec ERL$^*$ : l'ensemble des ressources $R$ modélise la mémoire du système, $Res$ étant un sous-ensemble de la mémoire spécifié pour chaque problème. Comme d'habitude, $e$ dénote un ensemble vide d'informations, soit un espace mémoire vide. L'ensemble des agents $A$ représente les différents threads ou processus qui effectuent les tâches. Deux parties $m$ et $m'$ de la mémoire seront équivalente par la relation $\sim_p$ si l'accès à $m$ est équivalent à l'accès à $m'$ pour le processus $p$. Enfin, on utilise les proposition de ERL$^*$ pour modéliser le contenu des programmes exécutés par les processus. Ainsi, on écrira $m \vDash P$ pour dire que les informations stockées dans la mémoire $m$ sont utilisées pour exécuter le programme $P$.

On utilise un ensemble arbitraire d'agents $A$ et les ressources $Res = \{e,t\}$ où $t$ est un marqueur (token) permettant de distinguer la présence d'un programme en section critique. On utilise également deux variables propositionnelles $C$ et $NC$ qui décrivent respectivement la section critique et la section non critique d'un même programme.

On considère les formules suivantes pour un certain processus $p \in A$ :

1. pour tout $p', p'' \in A, p' \neq p''$, on a : $Guard \equiv \widetilde{\mathbf{L}}^t_{p'} \top \rightarrow \neg \widetilde{\mathbf{L}}^t_{p''} \top$

2. $In \equiv NC \rightarrow \mathbf{L}^t_p C$
3. $Out \equiv C \rightarrow ((\neg \widetilde{\mathbf{M}}^t_p \top) \wedge \widetilde{\mathbf{M}}^e_p NC)$

La formule *Guard*, qui est vraie pour toute paire de processus différents $p'$ et $p''$, assure que deux programmes ne peuvent pas entrer en section critique en même temps. En effet, si on a $m \vDash Guard$, alors s'il existe $m'$ tel que $m \bullet t \sim_{p'} m'$ alors il n'y a aucun $m''$ tel que $m \bullet t \sim_{p''} m''$. C'est à dire que pour tout processus $p'$ qui a accès à $t$ en mémoire, aucun autre $p''$ ne peut récupérer ce token.

La formule *In* spécifie que le programme $p$ entre en section critique. Si on a $m \vDash In$, alors si $m \vDash NC$ on a pour tout $m'$ tel que $m \bullet t \sim_p m'$ que $m' \vDash C$. C'est à dire que si le processus est en train d'exécuter la section non critique, l'ajout du token $t$ suffit à lui faire atteindre un état de mémoire pouvant exécuter la section critique .

De manière symétrique, la formule *Out* exprime la sortie de $p$ de la section critique. Si $m \vDash Out$, alors si $m \vDash C$, on a pour tout $m'$ tel que $m \sim_p m' \bullet t$ que $m' \bullet t \nvDash \top$. C'est à dire qu'il n'y a aucun $m'$ tel que $m \sim_p m' \bullet t$. Cela nous permet d'effacer $t$ de la mémoire accessible à $p$. L'autre partie de la formule, $\widetilde{\mathbf{M}}^e_p NC)$, affirme qu'il y a un état $m''$ tel que $m \sim_p m''$ et $m'' \vDash NC$, c'est à dire que $p$ retourne en section critique. Ainsi, aucun état de mémoire qui satisfait $NC$ après que $C$ ait été exécuté ne peut inclure $t$. Par conséquent, une fois cette formule prise en compte, soit $p$ continue à exécuter $C$, soit il passe dans $NC$ et relâche le token $t$.

On peut maintenant vérifier que la garde que nous avons proposée est suffisante pour nous garantir que deux programmes ne rentrent pas en section critique. Pour cela, une manière très naïve de procéder est d'introduire une nouvelle formule qui signifie le lancement de deux programmes (initialement en section non critique). La formule que nous proposons est la suivante :

$$NC * NC$$

Notons, comme dans l'exemple précédent, l'utilisation de l'opérateur multiplicatif $*$ pour signifier deux programmes exécutés en parallèle. On voit aisément comment $\wedge$ ne conviendrait pas à notre problème : $NC \wedge NC$ signifierait que nos deux programmes se partagent un espace mémoire, ce qui n'est pas le cas pour notre parallélisation forte. A contrario, si $m \vDash NC * NC$, alors il existe $m_1, m_2$ tels que $m = m_1 \bullet m_2$ et $m_1 \vDash NC$ et $m_2 \vDash NC$, ce qui est une représentation acceptable de deux programmes exécutant la section non-critique du programme en parallèle dans deux sections différentes de la mémoire.

Considérons maintenant un processus $p_1$ utilisant $m_1$ et imaginons qu'il possède le token d'accès $t$, c'est à dire qu'il existe $m'_1$ tel que $m_1 \bullet t \sim_{p_1} m'_1$. Si $In$ est valide, alors on a en particulier $m_1 \vDash In$ et par application de $In$, comme on avait $m_1 \vDash NC$, on obtient $m'_1 \vDash C$. À présent, $p_1$ est en train d'exécuter la section critique avec $m'_1$. La question maintenant est de savoir si un autre processus $p_2$ peut accéder à la section critique avec l'espace mémoire $m_2$.

La propriété *Guard* devrait empêcher cela. En effet, si *Guard* est valide, on a en particulier $m \vDash Guard$ pour tous $p', p''$. Or, nous avons établi que $m_1 \bullet t \sim_{p_1} m'_1$. On a aussi $m = m_1 \bullet m2$ et donc, par composition à droite, $m \bullet t \sim_{p_1} m'_1 \bullet m_2$. En appliquant alors $m \vDash Guard$ avec $p' = p_1$ et $p'' = p_2$, on infère alors qu'il n'existe aucun $m'$ tel que $mbullett \sim_{p_2} m'$.

D'un autre côté, si $p_2$ accède à la section critique avec $m_2$, on aurait alors $m'_2$ tel que $m_2 \bullet t \sim_{p_2} m'_2$. On aurait alors $m \bullet t \sim_{p_2} m'_2 \bullet m_1$ ce qui contredit notre affirmation précédente. Par conséquent, $p_2$ ne peut pas exécuter la section critique.

Toutefois, lorsque nous sommes dans cette situation, on a $m'_1 \vDash C$ et on peut utiliser *Out* pour faire sortir $p_1$ de la section critique. Comme $m'_1 \vDash Out$, on obtient $m'_1 \vDash \neg\widetilde{\mathbf{M}}^t_{p_1}\top$ et $m'_1 \vDash \widetilde{\mathbf{M}}^e_{p_1}NC$. La première affirmation nous dit qu'il n'y a aucun $m'$ tel que $m'_1 \sim_{p_1} m' \bullet t$. Or, dans nos prémisses nous avions montré que $m'_1 \sim_{p_1} m_1 \bullet t$. ces deux affirmations sont contradictoire. Aussi, si l'on veut utiliser cette formule, on doit supprimer la relation $m'_1 \sim_{p_1} m_1 \bullet t$. Cela assure que $t$ n'est plus possédé par $p_1$. La deuxième partie que nous obtenons de *Out*, $m'_1 \vDash \widetilde{\mathbf{M}}^e_{p_1}NC$, nous donne un nouveau état de mémoire $m''_1$ tel que $m'_1 \sim_{p_1} m''_1$ et $m''_1 \vDash NC$, donc $p_1$ est de retour dans l'état non critique. Notons aussi que la suppression de $m'_1 \sim_{p_1} m_1 \bullet t$ empêche d'appliquer à nouveau la garde, et rien n'empêche $p_2$ d'entrer en section critique cette fois.

On voit ainsi comment ERL est particulièrement indiqué pour modéliser de tels accès conjoints ou concurrents, ses modalités permettant de caractériser les séparations strictes entre différentes entités (ici, les espaces mémoire) tandis que les opérateurs épistémiques distinguent les capacités et autorisations de chacun des agents modélisés.

Toutefois, on remarquera dans la fin de cet exemple que notre logique manque d'opérateurs pour exprimer certaines actions. En effet, on peut uniquement avec une telle logique formuler des affirmations sur les ressources et agents. On ne peut ni créer ni effacer de ressources ou de connaissance des agents sur les ressources. Ainsi, dans l'exemple décrit plus haut, le fait de supprimer le token $t$ de la mémoire d'un processus ne peut pas vraiment être spécifié. Il nous faut dire que le token $t$ ne doit pas être dans cette mémoire et ainsi choisir nous même un nouveau modèle qui convient. Pour décrire plus précisément de tels systèmes, on pourrait utiliser des systèmes formels d'avantage basés sur des actions, par exemple la logique de Hoare [61] qui a d'ors et déjà été utilisée comme base pour la logique de séparation [87]. On peut envisager un nouveau formalisme dans l'esprit de la logique de séparation ayant ERL comme cœur qui permettrait ainsi de raisonner non seulement sur l'état des systèmes à accès contrôlés, mais aussi sur leur transformation de manière plus systématique.

### 4.4.4 Modélisation par couches

Nous avons montré que ERL était un outil assez puissant pour modéliser de nombreux problèmes de contrôle d'accès et exprimer plusieurs propriétés de leurs comportement. Toutefois, l'approche que nous avons eue pour le moment a été plutôt ad-hoc, avec une approche spécifique pour chacune des situations modélisée. Nous voulons dans ce paragraphe adresser le fait que des procédés plus généraux de modélisation peuvent également utiliser ERL pour générer de manière plus efficace et plus systématique des modélisations de système complexe.

En particulier, on souhaite s'attarder sur le principe de *modélisation par couche*. On a déjà un peu mis en avant ce procédé dans notre premier exemple (la barrière de Schneier) en soulignant le problème du décalage entre la politique de construction de l'architecture d'un système et son implémentation concrète. Ces deux niveaux de modélisation peuvent être vues comme deux *couches* d'un même système.

Ce type de modélisation a déjà été développée grâce à des logiques proches de BI (mais plus faible) dans les structures appelées graphes à couches ou layered graphs [28,29,41]. De manière plus général, l'idée de modéliser un système par ses attributs les plus simple, puis de raffiner progressivement le modèle par l'ajout de spécifications de plus en plus précises (parcourant ainsi plusieurs niveaux de conceptualisation théorique et d'implémentation concrète) est un principe

bien connu en ingénierie et déjà décrit par plusieurs cadres formels. On citera notamment le langage B dont un des principes fondateurs est de permettre ce raffinement par couche [1].

Une telle façon de faire est particulièrement intéressante pour modéliser des problèmes de sécurité ou de contrôle d'accès. Les couches supérieures décrivent les principes généraux de sécurité tandis que les couches inférieures décrivent des cas particulier ou des implémentations pratiques. Par exemple, on aurait pu modéliser notre problème de barrière par deux couches : une théorique (la barrière filtre les entrées) et une physique qui hérite des propriétés théoriques mais en ajoute les défauts d'implémentation (on peut contourner la barrière).

Regardons à présent comment de tels principes peuvent être appliqués avec HRL. L'idée, comme nous l'avons dit, est de décrire un système par plusieurs niveaux de précisions, les couches. On commencera par modéliser les propriétés et les structures les plus générales du système, puis on raffinera ce modèle avec des couches plus précises, chacune ajoutant plus de détails à la précédente. Habituellement, une couche hérite des propriétés générales des couches précédentes.

L'idée principale dans l'usage de ces principes avec ERL est de diviser notre ensemble d'agents en plusieurs sous-ensembles qui représenterons chacun une couche différente du système :

$$A = A_0 \uplus A_1 \uplus \ldots \uplus A_n$$

On peut ensuite (si cela est pertinent) utiliser des schémas d'axiomes pour faire hériter des propriétés d'une couche à l'autre.

$$\forall i < n, \text{ si } \forall a \in A_i \; P[a] \text{ alors } \forall b \in A_{i+1} \; P[a/b]$$

Illustrons cette méthode par un exemple simple. On considère la modélisation d'une autoroute. La situation générale lorsqu'on prend l'autoroute est qu'une voiture rentre en prenant un ticket et paie le ticket en sortant de l'autoroute. On considère le modèle suivant :

1. $a$ est un agent arbitraire
2. $t$ est une ressource représentant le ticket
3. $O$, $M$ et $D$ sont trois propositions représentant respectivement le fait d'être en dehors de l'autoroute, sur l'autoroute, et endetté d'une certaine somme (correspondant au ticket).

On propose les deux formules suivantes pour modéliser l'entrée et la sortie de $a$ de l'autoroute :

$$In(a) \equiv O \rightarrow \widetilde{\mathbf{M}}_a^t M$$

$$Out(a) \equiv \widetilde{\mathbf{M}}_a^t M \rightarrow (D \wedge O)$$

On peut aussi modéliser la propriété de sécurité qui s'assure qu'aucun utilisateur sur l'autoroute ne le soit sans ticket :

$$Sec(a) \equiv M \rightarrow \widetilde{\mathbf{M}}_a^t M$$

À présent, si l'on veut raffiner ce modèle, par exemple en ajoutant l'utilisation des dispositifs de télé-péage (qui permet à un abonné de rentrer sur l'autoroute sans demander de ticket). Pour cela, on ajoute une ressource $d$ représentant le dispositif de télé-péage. On a alors :

$$In'(a) \equiv (O \wedge \mathbf{M}_a^d \top) \rightarrow M$$

$$Out'(a) \equiv (M \wedge \mathbf{M}_a^d \top) \rightarrow O$$

Notons qu'il est inutile de modéliser le paiement, car le télé-péage est géré séparément par un abonnement. On doit également mettre à jour notre politique de sécurité correctement :

$$Sec'(a) \equiv M \rightarrow (\widetilde{\mathbf{M}}_a^t M \vee \mathbf{M}_a^d \top)$$

Cette nouvelle version spécifie qu'un véhicule sur l'autoroute a soit un ticket, soit un dispositif de télé-péage. Finalement on peut généraliser ces propriétés avec des couches. On décompose notre ensemble d'agents $A$ en $A = A_0 \uplus A_1 \uplus A'$. $A_0$ est la couche supérieure décrivant toutes les voitures de l'autoroute, $A_1$ est le raffinement qui rajoute le télé-péage. $A'$ est spécifié ici pour exprimer qu'il peut exister d'autres agents encore inutilisés. On a alors :

$$\forall a \in A_0, \quad In(a) \wedge Out(a) \wedge Sec(a)$$

$$\forall a \in A_1, \quad In(a) \wedge Out(a) \wedge In'(a) \wedge Out'(a) \wedge Sec'(a)$$

On voit alors que les propriétés *In* et *Out* sont héritées (car une voiture, même avec un dispositif de télé-péage, peut encore prendre un ticket), mais *Sec* ne l'est pas puisque *Sec'* est plus générale et on a trivialement $Sec' \rightarrow Sec$.

On voit ainsi comment la séparation des agents en plusieurs sous-ensembles peut être très intéressante pour modéliser des systèmes par couches. Les agents cessent d'être de simples modèles ad-hoc de certains acteurs de notre système et sont utilisés plus comme des variables décrivant des types d'acteurs, dans différentes couches du systèmes. Cette manière de procéder peut être utilisée pour d'autres modélisations avec HRL et reflète en outre des principes connus et répandus de l'ingénierie.

## 4.5 Un calcul des tableaux pour ERL

Nous allons maintenant présenter un calcul des tableaux pour ERL. Ce calcul est inspiré du calcul pour ESL [37]. Les fondements de ce calcul ayant déjà été présentés dans les chapitres précédents (pour BBI et LTBI), on se focalisera ici sur les différences apportées par ERL(on abrégera ou on omettra notamment les preuves similaires à celles présentées dans les Chapitres 3 et 5).

En particulier, ce calcul des tableaux diffère de celui de ESL par l'introduction de labels correspondant aux ressources de *Res*, présentes dans la syntaxe.

### 4.5.1 Contraintes et labels

Une des particularité de ERL vis-à-vis des autres logiques pour lesquelles nous avons conçu des calculs des tableaux est la présence de certains symboles de ressources à l'intérieur de la syntaxe. Dans les autres calculs des tableaux, des labels de ressource sont introduits au fur et à mesure de la preuve, représentant les ressources forçant les formules.

Dans le cas particulier des ressources apparaissant dans les formules, ont doit s'assurer qu'un unique label représente chacune de ces ressources, pour que si une ressource est traitée à deux endroits différents du tableau, elle reçoive le même label à chaque fois. C'est le but de l'ensemble $\Lambda_r$ défini ici. Le lien avec les ressources de la syntaxe sera formalisé plus loin. Le reste des labels est défini de la manière habituelle.

**Définition 4.7** (Labels)**.** *On considère un ensemble fini de constantes $\Lambda_r$ tel que $|\Lambda_r| = |Res| - 1$. On l'étend en un ensemble de constantes $\gamma_r$ tel que $\Lambda_r \subset \gamma_r$, on a donc $\gamma_r = \Lambda_r \cup \{c_1, c_2, \ldots\}$.*

*Un* label de ressources *est un mot construit sur $\gamma_r$ où l'ordre des lettres n'est pas pris en compte (c'est à dire un multi-ensemble de constantes). On note $\varepsilon$ le mot vide. On note $L_r$ l'ensemble des labels de ressources.*

*On dit qu'un label $x$ est un* sous-label *de $y$ s'il existe $z$ tel que $y = xz$. L'ensemble des sous-labels de $x$ est noté $\mathcal{E}(x)$.*

Il nous faut maintenant définir les contraintes qui décrieront les relations entre les labels. Comme les ressources peuvent être liées soient par la relation d'équivalence $\sim$, soit par les relations $\sim_a$, nous introduisons deux classes de contraintes correspondant à ces deux groupes.

**Définition 4.8** (Contraintes)**.** *Une* contrainte de ressources *est une expression de la forme $x \simeq y$, où $x$ et $y$ sont des labels de ressources.*

*Une* contrainte d'agents *est une expression de la forme $x \eqsim_u y$, où $x$ et $y$ sont des labels de ressources et $u \in A$ est un agent.*

*Un* ensemble de contraintes *est un ensemble $\mathcal{C}$ qui contient des contraintes d'agents et des contraintes de ressources.*

*Soit $\mathcal{C}$ un ensemble de contraintes. Le* domaine *de $\mathcal{C}$, noté $\mathcal{D}_r(\mathcal{C})$ est l'ensemble de tous les sous-labels de ressource apparaissant dans $\mathcal{C}$. On a donc :*

$$\mathcal{D}_r(\mathcal{C}) = \left[\bigcup_{x \simeq y \in \mathcal{C}} (\mathcal{E}(x) \cup \mathcal{E}(y))\right] \cup \left[\bigcup_{x \eqsim_u y \in \mathcal{C}} (\mathcal{E}(x) \cup \mathcal{E}(y)).\right]$$

*L'*alphabet *de $\mathcal{C}$, noté $\mathcal{A}_r(\mathcal{C})$ est l'ensemble de toutes les constantes de ressources apparaissant dans $\mathcal{C}$, c'est à dire :*

$$\mathcal{A}_r(\mathcal{C}) = \gamma_r \cap \mathcal{D}_r(\mathcal{C})$$

Formalisons maintenant le lien entre les labels de $\Lambda_r$ et les ressources de *Res*. Comme les deux ensembles ont le même cardinal (à un élément près, pour gérer le cas de $e$ qui est représenté par l'élément vide), on peut dresser une bijection entre les deux.

**Définition 4.9** (Fonction lambda)**.** *On définit une fonction $\lambda : Res \mapsto L_r$ telle que :*

1. *$\lambda(e) = \varepsilon$ ;*
2. *pour tout $r \in Res \backslash \{e\}$, $\lambda(r) \in \Lambda_r$ ;*
3. *$\lambda$ est injective (si $\lambda(r) \simeq \lambda(r')$, alors $r \sim r'$).*

Notons que $\lambda$ est alors trivialement une bijection entre *Res* et $\Lambda_r \cup \{\varepsilon\}$.

On définit à présent la clôture des ensembles de contraintes qui permet de capturer les propriétés de relation d'équivalence de $\simeq$ et $\eqsim$.

**Définition 4.10** (Clôture de contraintes)**.** *Soit $\mathcal{C}$ un ensemble de contraintes. La clôture de $\mathcal{C}$, notée $\overline{\mathcal{C}}$, est le plus petit ensemble clos par les règles de la Figure 4.7 et tel que $\mathcal{C} \subseteq \overline{\mathcal{C}}$.*

Règles pour les contraintes de ressources :

$$\frac{}{\varepsilon \simeq \varepsilon}\ \langle \varepsilon \rangle \qquad \frac{x \simeq y}{y \simeq x}\ \langle s_r \rangle \qquad \frac{xy \simeq xy}{x \simeq x}\ \langle d_r \rangle \qquad \frac{x \simeq y \qquad y \simeq z}{x \simeq z}\ \langle t_r \rangle$$

$$\frac{x \simeq y \qquad yk \simeq yk}{xk \simeq yk}\ \langle c_r \rangle \qquad \frac{x =\!\!=_u y}{x \simeq x}\ \langle k_r \rangle$$

Règles pour les contraintes d'agent :

$$\frac{x \simeq x}{x =\!\!=_u x}\ \langle r_a \rangle \qquad \frac{x =\!\!=_u y}{y =\!\!=_u x}\ \langle s_a \rangle \qquad \frac{x =\!\!=_u y \qquad y =\!\!=_u z}{x =\!\!=_u z}\ \langle t_a \rangle \qquad \frac{x =\!\!=_u y \qquad x \simeq k}{k =\!\!=_u y}\ \langle k_a \rangle$$

FIGURE 4.7 – Règles de clôture des contraintes (pour tout agent $u \in A$)

Six de ces règles produisent des contraintes de ressource ($\langle \varepsilon \rangle$, $\langle s_r \rangle$, $\langle d_r \rangle$, $\langle t_r \rangle$, $\langle c_r \rangle$, et $\langle k_r \rangle$) tandis que les quatre autres produisent des contraintes d'agents ($\langle r_a \rangle$, $\langle s_a \rangle$, $\langle t_a \rangle$, et $\langle k_a \rangle$). La règle $\langle r_a \rangle$ est un peu particulière puisqu'elle produit une contrainte d'agents à partir d'une contrainte de ressources. Ainsi l'agent $u$ produit par cette règle doit appartenir à $A$.

La clôture permettra ainsi d'obtenir certaines nouvelles contraintes à partir des contraintes existantes, en dressant un arbre de preuve utilisant les règles de la Figure 4.7. Certains éléments de preuves étant assez récurrent, on introduit une série de règles secondaires pouvant être déduites de ces règles.

**Proposition 4.3.** *Les règles suivantes peuvent être déduites des règles de la Figure 4.7*

$$\frac{xk \simeq y}{x \simeq x}\ \langle p_l \rangle \qquad \frac{x \simeq yk}{y \simeq y}\ \langle p_r \rangle \qquad \frac{xk =\!\!=_u y}{x \simeq x}\ \langle q_l \rangle \qquad \frac{x =\!\!=_u yk}{y \simeq y}\ \langle q_r \rangle$$

$$\frac{x =\!\!=_u y \qquad x \simeq x' \qquad y \simeq y'}{x' =\!\!=_u y'}\ \langle w_a \rangle$$

*Démonstration.*
On dresse pour chaque règle l'arbre de preuve correspondant en utilisant les règles de la Figure 4.7 et les règles prouvées précédemment :

1.

$$\dfrac{xk \simeq y \qquad \dfrac{xk \simeq y}{y \simeq xk}\ \langle s_r \rangle}{\dfrac{xk \simeq xk}{x \simeq x}\ \langle d_r \rangle}\ \langle t_r \rangle$$

2.

$$\dfrac{\dfrac{x \simeq yk}{yk \simeq x}\ \langle s_r \rangle}{y \simeq y}\ \langle p_l \rangle$$

3.

$$\dfrac{\dfrac{xk \eqsim_u y}{xk \simeq xk}\ \langle k_r\rangle}{x \simeq x}\ \langle d_r\rangle$$

4.

$$\dfrac{\dfrac{x \eqsim_u yk}{yk \eqsim_u x}\ \langle s_a\rangle}{y \simeq y}\ \langle q_l\rangle$$

5.

$$\dfrac{\dfrac{\dfrac{x \eqsim_u y \qquad x \simeq x'}{\dfrac{x' \eqsim_u y}{y \eqsim_u x'}\ \langle s_a\rangle}\ \langle k_a\rangle \qquad y \simeq y'}{\dfrac{y' \eqsim_u x'}{x' \eqsim_u y'}\ \langle s_a\rangle}\ \langle k_a\rangle$$

□

Nous pouvons maintenant énoncer quelques propriétés intéressantes des clôtures de contraintes. Les premières sont les suivantes.

**Corollaire 4.1.** *Soit $\mathcal{C}$ un ensemble de contraintes et $u \in A$ un agent.*

1. *$x \in \mathcal{D}_r(\overline{\mathcal{C}})$ ssi $x \simeq x \in \overline{\mathcal{C}}$ ssi $x \eqsim_u x \in \overline{\mathcal{C}}$.*
2. *Si $xy \in \mathcal{D}_r(\overline{\mathcal{C}})$, $x' \simeq x \in \overline{\mathcal{C}}$, et $y' \simeq y \in \overline{\mathcal{C}}$, alors $xy \simeq x'y' \in \overline{\mathcal{C}}$.*

*Démonstration.* 1. Si $x \simeq x \in \overline{\mathcal{C}}$, alors $x \in \mathcal{D}_r(\overline{\mathcal{C}})$ par définition. De même, si $x \eqsim_u x \in \overline{\mathcal{C}}$, alors $x \in \mathcal{D}_r(\overline{\mathcal{C}})$. Montrons maintenant le sens inverse.

On suppose que $x \in \mathcal{D}_r(\overline{\mathcal{C}})$. Il existe alors $y, k$ (possiblement $k = \varepsilon$) tel que l'un des 4 cas suivant se produise :

- $xk \simeq y \in \overline{\mathcal{C}}$
  Alors par la règle $\langle p_l\rangle$ on a $x \simeq x \in \overline{\mathcal{C}}$.
- $y \simeq xk \in \overline{\mathcal{C}}$
  Alors par la règle $\langle p_r\rangle$ on a $x \simeq x \in \overline{\mathcal{C}}$
- $xk \eqsim_u y \in \overline{\mathcal{C}}$
  Alors par la règle $\langle q_l\rangle$ on a $x \simeq x \in \overline{\mathcal{C}}$
- $y \eqsim_u xk \in \overline{\mathcal{C}}$
  Alors par la règle $\langle q_r\rangle$ on a $x \simeq x \in \overline{\mathcal{C}}$

Dans tous les cas, on a alors $x \simeq x \in \overline{\mathcal{C}}$. En outre, si $x \simeq x \in \overline{\mathcal{C}}$, alors par la règle $\langle r_a\rangle$ on a $x \eqsim_u x \in \overline{\mathcal{C}}$.

2. On suppose $xy \in \mathcal{D}_r(\overline{\mathcal{C}})$, $x' \simeq x \in \overline{\mathcal{C}}$ et $y' \simeq y \in \overline{\mathcal{C}}$. Par 1. on obtient que $xy \simeq xy \in \overline{\mathcal{C}}$. Par deux applications de $\langle c_r \rangle$ on a alors $x'y \simeq xy \in \overline{\mathcal{C}}$ puis $x'y' \simeq xy \in \overline{\mathcal{C}}$. Enfin, par $\langle s_r \rangle$, on obtient $xy \simeq x'y' \in \overline{\mathcal{C}}$.

□

On formalise également le fait que la clôture de contrainte n'introduit pas de nouveau symbole de label.

**Proposition 4.4.** *Soit $\mathcal{C}$ un ensemble de contraintes. On a $\mathcal{A}_r(\mathcal{C}) = \mathcal{A}_r(\overline{\mathcal{C}})$.*

*Démonstration.*
La proposition est triviale vu que aucune règle de la Figure 4.7 n'introduit de nouvelle constante de ressource. Ainsi, si $x \in \mathcal{A}_r(\overline{\mathcal{C}})$, alors nécessairement $x \in \mathcal{A}_r(\mathcal{C})$. Donc $\mathcal{A}_r(\overline{\mathcal{C}}) \subseteq \mathcal{A}_r(\mathcal{C})$. En outre, comme $\mathcal{C} \subseteq \overline{\mathcal{C}}$, on a aussi évidemment $\mathcal{A}_r(\mathcal{C}) \subseteq \mathcal{A}_r(\overline{\mathcal{C}})$. □

Un autre résultat qui paraît trivial mais qu'il est important d'énoncer est la conservation de l'inclusion par le passage à la clôture.

**Proposition 4.5.** *Soit $\mathcal{C}$, $\mathcal{C}'$ deux ensembles de contraintes tels que $\mathcal{C} \subseteq \mathcal{C}'$.*

1. *Si $x \simeq y \in \overline{\mathcal{C}}$, alors $x \simeq y \in \overline{\mathcal{C}'}$*
2. *Si $x \eqsim_u y \in \overline{\mathcal{C}}$, alors $x \eqsim_u y \in \overline{\mathcal{C}'}$*

*Démonstration.*
On présente ici le résultat 1., la preuve pour 2. est similaire. On prouve le résultat par récurrence sur la taille de la preuve permettant d'obtenir $x \simeq y$.

- Cas de base
  Si la preuve est de taille 0, alors on a $x \simeq y \in \mathcal{C}$. Comme $\mathcal{C} \subseteq \mathcal{C}'$, on a alors $x \simeq y \in \mathcal{C}'$, et donc $x \simeq y \in \overline{\mathcal{C}'}$.
- Hérédité
  On suppose que la propriété est vraie pour toutes les contraintes de ressources obtenues par des arbres de taille $n$ (HR). On considère une contrainte $x \simeq y$ obtenue par une arbre de taille $n+1$. On discute selon la règle ayant produit $x \simeq y$ :
  - $\langle s_r \rangle$
    Alors $x \simeq y$ a été obtenu par $y \simeq x$ et $y \simeq x$ est obtenu par un arbre de taille $n$ au plus. En outre, on a $y \simeq x \in \overline{\mathcal{C}}$. Alors par (HR), $y \simeq x \in \overline{\mathcal{C}'}$. Alors par clôture par la règle $\langle s_r \rangle$, on a aussi $x \simeq y \in \overline{\mathcal{C}'}$, et la propriété tient pour $x \simeq y$.
  - $\langle t_r \rangle$
    Alors $x \simeq y$ a été obtenu par $x \simeq z$ et $z \simeq y$, et $x \simeq z$ et $z \simeq y$ sont obtenus par des arbre de taille $n$ au plus. En outre, on a $x \simeq z \in \overline{\mathcal{C}}$ et $z \simeq y \in \overline{\mathcal{C}}$. Alors par (HR), $x \simeq z \in \overline{\mathcal{C}'}$ et $z \simeq y \in \overline{\mathcal{C}'}$. Alors par clôture par la règle $\langle t_r \rangle$, on a aussi $x \simeq y \in \overline{\mathcal{C}'}$, et la propriété tient pour $x \simeq y$.
  - Les autres cas sont similaires.

  Finalement la proposition tient pour toutes les contraintes obtenues par des arbres de taille $n+1$ et, par récurrence, pour toutes les contraintes de ressource.

□

Enfin, on traite de la possibilité d'extraire de toute clôture un sous-ensemble fini qui contienne une contrainte en particulier. Ce lemme sera particulièrement important pour traiter le cas des branches infinies dans la preuve de complétude.

**Lemme 4.2.** *Soit $\mathcal{C}$ un ensemble de contraintes (possiblement infini).*

1. *Si $x \simeq y \in \overline{\mathcal{C}}$, alors il existe un ensemble fini $\mathcal{C}_f$ tel que $\mathcal{C}_f \subseteq \mathcal{C}$ et $x \simeq y \in \overline{\mathcal{C}_f}$.*
2. *Si $x \frown_u y \in \overline{\mathcal{C}}$, alors il existe un ensemble fini $\mathcal{C}_f$ tel que $\mathcal{C}_f \subseteq \mathcal{C}$ et $x \frown_u y \in \overline{\mathcal{C}_f}$.*

*Démonstration.*
On présente ici le résultat 1., la preuve pour 2. est similaire. On prouve le résultat par récurrence sur la taille de la preuve permettant d'obtenir $x \simeq y$.

- Cas de base
  Si la preuve est de taille 0, alors on a $x \simeq y \in \mathcal{C}$. On choisit alors $\mathcal{C}_f = \{x \simeq y\}$. $\mathcal{C}_f$ est fini, et comme $x \simeq y \in \mathcal{C}$, $\mathcal{C}_f \subseteq \mathcal{C}$. Enfin, comme $x \simeq y \in \mathcal{C}_f$, on a $x \simeq y \in \overline{\mathcal{C}_f}$.
- Hérédité
  On suppose que la propriété est vraie pour toutes les contraintes de ressources obtenues par des arbres de taille $n$ (HR). On considère une contrainte $x \simeq y$ obtenue par une arbre de taille $n+1$. On discute selon la règle ayant produit $x \simeq y$ :
  - $\langle s_r \rangle$
    Alors $x \simeq y$ a été obtenu par $y \simeq x$ et $y \simeq x$ est obtenu par un arbre de taille $n$ au plus. Alors par (HR), il existe un ensemble fini $\mathcal{C}_f$ tel que $\mathcal{C}_f \subseteq \mathcal{C}$ et $y \simeq x \in \overline{\mathcal{C}_f}$. Alors par clôture par la règle $\langle s_r \rangle$, on a aussi $x \simeq y \in \overline{\mathcal{C}_f}$, et la propriété tient pour $x \simeq y$.
  - $\langle t_r \rangle$
    Alors $x \simeq y$ a été obtenu par $x \simeq z$ et $z \simeq y$. Alors $x \simeq z$ et $z \simeq y$ sont obtenus par des arbre de taille $n$ au plus. Alors par (HR), il existe un ensemble fini $\mathcal{C}_1$ tel que $\mathcal{C}_1 \subseteq \mathcal{C}$ et $x \simeq z \in \overline{\mathcal{C}_1}$. De même, il existe un ensemble fini $\mathcal{C}_2$ tel que $\mathcal{C}_2 \subseteq \mathcal{C}$ et $z \simeq y \in \overline{\mathcal{C}_2}$. On prend alors $\mathcal{C}_f = \mathcal{C}_1 \cup \mathcal{C}_2$. Par construction, $\mathcal{C}_f$ est fini et $\mathcal{C}_f \subseteq \mathcal{C}$. En outre, comme $\mathcal{C}_1 \subseteq \mathcal{C}_f$ et que $x \simeq z \in \overline{\mathcal{C}_1}$, par la Proposition 4.5 on a $x \simeq z \in \overline{\mathcal{C}_f}$. On montre de même que $z \simeq y \in \overline{\mathcal{C}_f}$. Alors par clôture par la règle $\langle t_r \rangle$, on a aussi $x \simeq y \in \overline{\mathcal{C}_f}$, et la propriété tient pour $x \simeq y$.
  - Les autres cas sont similaires.

  Finalement la proposition tient pour toutes les contraintes obtenues par des arbres de taille $n+1$ et, par récurrence, pour toutes les contraintes de ressource.

□

Avant de définir le reste de nos tableaux, nous souhaitons introduire une variante des clôtures de contraintes qui nous permettra de traiter la variante ERL$^*$ que nous avons présentée dans la Définition 4.6. Pour rappel, la différence entre ERL$^*$ et ERL est l'ajout de la compatibilité de $\sim_a$ et $\bullet$ pour tout agent $a \in A$. Cela est traduit par l'ajout d'une règle à la clôture des contraintes.

**Définition 4.11** (Clôture des contraintes pour ERL$^*$)**.** *La clôture des contraintes pour le calcul des tableaux de la logique* ERL$^*$ *est définie comme pour ERL (cf Définition 4.10) mais avec l'ajout de la règle suivante aux règles de la Figure 4.7 :*

$$\frac{x \rightleftharpoons_u y \qquad yk \simeq yk}{xk \rightleftharpoons_u yk} \ \langle c_a \rangle$$

**Lemme 4.3.** *Les résultats du Corollaire 2, de la Proposition 4.4, de la Proposition 4.5 et du Lemme 4.2 sont toujours valables dans le cadre du calcul pour* ERL$^*$.

*Démonstration.*
Les preuves sont identiques à celles des résultats sus-mentionnés, il faut simplement ajouter le cas de la règle $\langle c_a \rangle$ à tous les raisonnements par récurrence, mais cela n'amène pas de difficulté particulière. □

### 4.5.2 Un calcul avec labels pour ERL

Nous pouvons maintenant construire nos tableaux avec les labels et contraintes que nous venons de définir. En premier lieu, nous attachons à chaque formule un label par la définition de formule étiquetée.

**Définition 4.12** (Formule étiquetée, ECSS)**.** *Une* formule étiquetée *est un triplet* $(S, \phi, x)$ *tel que* $S \in \{\mathbb{T}, \mathbb{F}\}$, $\phi \in \mathcal{L}$ *est une formule de ERL et* $x \in L_r$ *est un label de ressource. On note une telle formule* $\mathbb{S}\phi : x$.

*Un* ensemble d'énoncés contraints *(noté ECSS pour Epistemic Constrained Set of Statements) est une paire* $\langle \mathcal{F}, \mathcal{C} \rangle$*, où* $\mathcal{F}$ *est un ensemble de formules étiquetées et* $\mathcal{C}$ *est un ensemble de contraintes, et qui satisfait la propriété* $P_{ECSS}$ *définie par :*

$$\textit{si } \mathbb{S}\phi : x \in \mathcal{F}\textit{, alors } x \simeq x \in \overline{\mathcal{C}} \ \textit{(}P_{ECSS}\textit{).}$$

*Un ECSS* $\langle \mathcal{F}, \mathcal{C} \rangle$ *est* fini *si* $\mathcal{F}$ *et* $\mathcal{C}$ *sont finis.*
*La relation* $\preccurlyeq$ *est définie par* $\langle \mathcal{F}, \mathcal{C} \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}' \rangle$ *si* $\mathcal{F} \subseteq \mathcal{F}'$ *et* $\mathcal{C} \subseteq \mathcal{C}'$.
*On écrit* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C} \rangle$ *lorsque* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle \preccurlyeq \langle \mathcal{F}, \mathcal{C} \rangle$ *et que* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle$ *est fini.*

Ainsi, les ECSS permettent de décrire au cours de la preuve les formules considérées ainsi que les contraintes sur leurs labels. La proposition suivante nous permet en outre d'extraire d'un ECSS infini une branche finie contenant certaines contraintes.

**Proposition 4.6.** *Pour tout* $\langle \mathcal{F}_f, \mathcal{C} \rangle$*, où* $\mathcal{F}_f$ *est fini, il existe* $\mathcal{C}_f \subseteq \mathcal{C}$ *tel que* $\mathcal{C}_f$ *est fini et* $\langle \mathcal{F}_f, \mathcal{C}_f \rangle$ *est un ECSS.*

*Démonstration.*
Soit $\mathbb{S}\phi : x \in \mathcal{F}_f$. Comme $\langle \mathcal{F}_f, \mathcal{C} \rangle$, on a $x \simeq x \in \overline{\mathcal{C}}$. Alors par le Lemme 4.2, il existe un ensemble de contrainte fini $\mathcal{C}(x)$ tel que $x \simeq x \in \overline{\mathcal{C}(x)}$. On pose alors $\mathcal{C}_f = \cup_{\mathbb{S}\phi : x \in \mathcal{F}_f} \mathcal{C}(x)$. Comme chaque $\mathcal{C}(x)$ est fini et que $\mathcal{F}_f$ est fini aussi, alors $\mathcal{C}_f$ est fini.

Vérifions maintenant que $\langle \mathcal{F}_f, \mathcal{C}_f \rangle$ est bien un ECSS. Soit $\mathbb{S}\phi : x \in \mathcal{F}_f$. Alors comme on a dit plutôt, on a $x \simeq x \in \overline{\mathcal{C}(x)}$. Or, $\mathcal{C}(x) \subseteq \mathcal{C}_f$. Donc par la Proposition 4.5, $x \simeq x \in \overline{\mathcal{C}_f}$ et on vérifie bien la propriété $P_{ECSS}$ □

$$\frac{\mathbb{T}\mathbf{I} : x \in \mathcal{F}}{\langle \emptyset, \{x \simeq \varepsilon\}\rangle}\ \langle \mathbb{T}\mathbf{I}\rangle$$

$$\frac{\mathbb{T}\neg\phi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset\rangle}\ \langle \mathbb{T}\neg\rangle \qquad \frac{\mathbb{F}\neg\phi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x\}, \emptyset\rangle}\ \langle \mathbb{F}\neg\rangle$$

$$\frac{\mathbb{T}\phi \wedge \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x, \mathbb{T}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{T}\wedge\rangle \qquad \frac{\mathbb{F}\phi \wedge \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset\rangle \mid \langle \{\mathbb{F}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{F}\wedge\rangle$$

$$\frac{\mathbb{T}\phi \vee \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x\}, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{T}\vee\rangle \qquad \frac{\mathbb{F}\phi \vee \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x, \mathbb{F}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{F}\vee\rangle$$

$$\frac{\mathbb{T}\phi \to \psi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : x\}, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{T}\to\rangle \qquad \frac{\mathbb{F}\phi \to \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : x, \mathbb{F}\psi : x\}, \emptyset\rangle}\ \langle \mathbb{F}\to\rangle$$

$$\frac{\mathbb{T}\phi * \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i, \mathbb{T}\psi : c_j\}, \{x \simeq c_i c_j\}\rangle}\ \langle \mathbb{T}*\rangle \qquad \frac{\mathbb{F}\phi * \psi : x \in \mathcal{F} \text{ et } x \simeq yz \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\}, \emptyset\rangle \mid \langle \{\mathbb{F}\psi : z\}, \emptyset\rangle}\ \langle \mathbb{F}*\rangle$$

$$\frac{\mathbb{T}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F} \text{ et } xy \simeq xy \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\}, \emptyset\rangle \mid \langle \{\mathbb{T}\psi : xy\}, \emptyset\rangle}\ \langle \mathbb{T}\mathrel{-\!\!*}\rangle \qquad \frac{\mathbb{F}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i, \mathbb{F}\psi : xc_i\}, \{xc_i \simeq xc_i\}\rangle}\ \langle \mathbb{F}\mathrel{-\!\!*}\rangle$$

$$\frac{\mathbb{T}\mathbf{L}^r_u\phi : x \in \mathcal{F} \text{ et } x\lambda(r) =_u y \in \overline{\mathcal{C}}}{\langle \{\mathbb{T}\phi : y\}, \emptyset\rangle}\ \langle \mathbb{T}\mathbf{L}\rangle \qquad \frac{\mathbb{F}\mathbf{L}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : c_i\}, \{x\lambda(r) =_u c_i\}\rangle}\ \langle \mathbb{F}\mathbf{L}\rangle$$

$$\frac{\mathbb{T}\mathbf{M}^r_u\phi : x \in \mathcal{F} \text{ et } x =_u y\lambda(r) \in \overline{\mathcal{C}}}{\langle \{\mathbb{T}\phi : y\lambda(r)\}, \emptyset\rangle}\ \langle \mathbb{T}\mathbf{M}\rangle \qquad \frac{\mathbb{F}\mathbf{M}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : c_i\lambda(r)\}, \{x =_u c_i\lambda(r)\}\rangle}\ \langle \mathbb{F}\mathbf{M}\rangle$$

$$\frac{\mathbb{T}\mathbf{N}^r_u\phi : x \in \mathcal{F} \text{ et } x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}}{\langle \{\mathbb{T}\phi : y\lambda(r)\}, \emptyset\rangle}\ \langle \mathbb{T}\mathbf{N}\rangle \qquad \frac{\mathbb{F}\mathbf{N}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{F}\phi : c_i\lambda(r)\}, \{x\lambda(r) =_u c_i\lambda(r)\}\rangle}\ \langle \mathbb{F}\mathbf{N}\rangle$$

$$\frac{\mathbb{T}\widetilde{\mathbf{L}}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i\}, \{x\lambda(r) =_u c_i\}\rangle}\ \langle \mathbb{T}\widetilde{\mathbf{L}}\rangle \qquad \frac{\mathbb{F}\widetilde{\mathbf{L}}^r_u\phi : x \in \mathcal{F} \text{ et } x\lambda(r) =_u y \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\}, \emptyset\rangle}\ \langle \mathbb{F}\widetilde{\mathbf{L}}\rangle$$

$$\frac{\mathbb{T}\widetilde{\mathbf{M}}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i\lambda(r)\}, \{x =_u c_i\lambda(r)\}\rangle}\ \langle \mathbb{T}\widetilde{\mathbf{M}}\rangle \qquad \frac{\mathbb{F}\widetilde{\mathbf{M}}^r_u\phi : x \in \mathcal{F} \text{ et } x =_u y\lambda(r) \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\lambda(r)\}, \emptyset\rangle}\ \langle \mathbb{F}\widetilde{\mathbf{M}}\rangle$$

$$\frac{\mathbb{T}\widetilde{\mathbf{N}}^r_u\phi : x \in \mathcal{F}}{\langle \{\mathbb{T}\phi : c_i\lambda(r)\}, \{x\lambda(r) =_u c_i\lambda(r)\}\rangle}\ \langle \mathbb{T}\widetilde{\mathbf{N}}\rangle \qquad \frac{\mathbb{F}\widetilde{\mathbf{N}}^r_u\phi : x \in \mathcal{F} \text{ et } x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}}{\langle \{\mathbb{F}\phi : y\lambda(r)\}, \emptyset\rangle}\ \langle \mathbb{F}\widetilde{\mathbf{N}}\rangle$$

Note : $c_i$ et $c_j$ sont des nouvelles constantes de label, avec $c_i, c_j \notin \Lambda_r$.

FIGURE 4.8 – Règles pour le calcul des tableaux pour ERL

Ces objets étant définis, on peut établir les règles de notre calcul des tableaux. Ces règles sont énumérées dans la Figure 4.8.

Dans cette figure, la note '$c_i$ et $c_j$ sont de nouvelles constantes de label' signifie que $c_i \neq c_j \in \gamma_r \setminus (\mathcal{A}_r(\mathcal{C}) \cup \Lambda_r)$.

L'utilisation formelle des règles, et la construction d'un tableau sont alors dictées par la définition suivante :

**Définition 4.13** (Tableau pour ERL)**.** *Soit $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ un ECSS fini. Un* tableau *pour $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ est une liste de ECSS, appelés* branches*, construite de manière inductive de la manière suivante :*

1. *La liste à une branche $[\langle \mathcal{F}_0, \mathcal{C}_0 \rangle]$ est un tableau pour $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ ;*
2. *Si la liste $\mathcal{T}_m \oplus [\langle \mathcal{F}, \mathcal{C} \rangle] \oplus \mathcal{T}_n$ est un tableau pour $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$ et*

$$\frac{cond(\langle \mathcal{F}, \mathcal{C} \rangle)}{\langle \mathcal{F}_1, \mathcal{C}_1 \rangle \mid \ldots \mid \langle \mathcal{F}_k, \mathcal{C}_k \rangle}$$

*est une instance d'une règle de la Figure 4.8 pour laquelle la condition $cond(\langle \mathcal{F}, \mathcal{C} \rangle)$ est remplie, alors la liste $\mathcal{T}_m \oplus [\langle \mathcal{F} \cup \mathcal{F}_1, \mathcal{C} \cup \mathcal{C}_1 \rangle; \ldots; \langle \mathcal{F} \cup \mathcal{F}_k, \mathcal{C} \cup \mathcal{C}_k \rangle] \oplus \mathcal{T}_n$ est un tableau pour $\langle \mathcal{F}_0, \mathcal{C}_0 \rangle$.*

*Un tableau pour la formule $\phi$ est un tableau pour $\langle \{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\} \rangle$.*

On remarquera que dans cette définition, la branche de départ pour une formule $\phi$ est bien un ECSS (elle vérifie la propriété $P_{ECSS}$ de la Définition 4.12). En outre, par le Corollaire 2, on montre facilement que toute branche produite par une application d'une règle de la Figure 4.8 vérifie également $P_{ECSS}$.

Parmi les règles de la Figure 4.8 on peut mettre en avant deux catégories :

- Les règles $\langle \mathbb{T}\mathbf{I} \rangle$, $\langle \mathbb{T}* \rangle$, $\langle \mathbb{F}{-\!\!*} \rangle$, $\langle \mathbb{F}\mathbf{L} \rangle$, $\langle \mathbb{F}\mathbf{M} \rangle$, $\langle \mathbb{F}\mathbf{N} \rangle$, $\langle \mathbb{T}\widetilde{\mathbf{L}} \rangle$, $\langle \mathbb{T}\widetilde{\mathbf{M}} \rangle$, et $\langle \mathbb{T}\widetilde{\mathbf{N}} \rangle$ introduisent de nouveaux labels ($c_i$ et $c_j$) et de nouvelles contraintes, à l'exception de $\langle \mathbb{T}\mathbf{I} \rangle$ qui n'introduit qu'une contrainte.
- Les règles $\langle \mathbb{F}* \rangle$, $\langle \mathbb{T}{-\!\!*} \rangle$, $\langle \mathbb{T}\mathbf{L} \rangle$, $\langle \mathbb{T}\mathbf{M} \rangle$, $\langle \mathbb{T}\mathbf{N} \rangle$, $\langle \mathbb{F}\widetilde{\mathbf{L}} \rangle$, $\langle \mathbb{F}\widetilde{\mathbf{M}} \rangle$, et $\langle \mathbb{F}\widetilde{\mathbf{N}} \rangle$ nécessitent le choix d'une contrainte pour être appliquée.

De manière assez logique, nous devons donc respecter un *ordre d'application* des règles. Les règles de la première catégorie (qui produisent labels et contraintes) doivent être autant que faire ce peut appliquées avant celles de la deuxième (qui en consomment). Les autres règles peuvent être appliquées indifféremment, mais on préférera toujours appliquer la deuxième catégorie le plus tard possible.

Bien que cet ordre ne soit pas toujours essentiel, son non-respect peut amener à empêcher la clôture d'un tableau.

Définissons maintenant ce qu'est un tableau clos.

**Définition 4.14** (Conditions de clôture)**.** *Un ECSS $\langle \mathcal{F}, \mathcal{C} \rangle$ est* clos *si l'une des conditions suivantes tient, pour n'importe quelle formule $\phi \in \mathcal{L}$, et n'importe quels labels de ressource $x, y$ :*

1. *$\mathbb{T}\phi : x \in \mathcal{F}$, $\mathbb{F}\phi : y \in \mathcal{F}$ et $x \simeq y \in \overline{\mathcal{C}}$ ;*
2. *$\mathbb{F}\mathbf{I} : x \in \mathcal{F}$ et $x \simeq \varepsilon \in \overline{\mathcal{C}}$ ;*
3. *$\mathbb{F}\top : x \in \mathcal{F}$ ;*

4. $\mathbb{T}\bot : x \in \mathcal{F}$.

*Un ECSS est* ouvert *s'il n'est pas clos. Un tableau est* clos *si toutes ses branches sont closes. Une* preuve *pour une formule* $\phi$ *est un tableau clos pour* $\phi$.

Enfin, on peut utiliser cet calcul pour raisonner sur ERL$^*$en utilisant la clôture de contrainte de la Définition 4.11.

**Définition 4.15** (Tableaux pour ERL$^*$)**.** *Un tableau pour la logique* ERL$^*$ *est défini de la même manière que les tableaux pour ERL (Définition 4.13) mais avec les clôtures de contraintes pour* ERL$^*$ *(Définition 4.11). Les règles et les conditions de clôture sont inchangées.*

### 4.5.3 Un exemple de ERL-tableau

Pour illustrer notre calcul des tableaux, on se propose d'étudier un exemple. Nous allons revisiter l'une des formules proposées dans le Lemme 4.1 et en dresser une preuve avec un tableau. En l'occurrence, la formule que nous traitons est la numéro 2. de ce lemme, pour rappel :

$$\mathbf{M}_a^s\phi \rightarrow \mathbf{M}_a^r(\mathbf{M}_a^s\phi)$$

Comme indiqué dans la Définition 4.13, pour établir une preuve de notre formule, nous devons commencer par le ECSS suivant :

$$\langle \{\mathbb{F}\mathbf{M}_a^s\phi \rightarrow \mathbf{M}_a^r(\mathbf{M}_a^s\phi) : c_1\}, \{c_1 \simeq c_1\}\rangle$$

Pour plus de clarté, on préférera représenter les ensembles de formules ($\mathcal{F}$) et l'ensemble des contraintes ($\mathcal{C}$) de chaque ECSS séparément. En outre, on représentera les ECSS sous forme d'arbres, ce qui rend le développement du tableau plus pratique :

$$\begin{array}{cc} [\mathcal{F}] & [\mathcal{C}] \\ \surd_1 \mathbb{F}\mathbf{M}_a^s\phi \rightarrow \mathbf{M}_a^r(\mathbf{M}_a^s\phi) : c_1 & c_1 \simeq c_1 \end{array}$$

On doit décomposer la formule étiquetée $\mathbb{F}\mathbf{M}_a^s\phi \rightarrow \mathbf{M}_a^r(\mathbf{M}_a^s\phi) : c_1$, il faut donc utiliser la règle $\langle \mathbb{F} \rightarrow \rangle$. On produit deux nouvelles formules étiquetées, $\mathbb{T}\mathbf{M}_a^s\phi : c_1$ et $\mathbb{F}\mathbf{M}_a^r(\mathbf{M}_a^s\phi) : c_1$.

On continue ainsi à appliquer les règles de la Figure 4.8 jusqu'à clore notre tableau. Le tableau final est exposé dans la Figure 4.9. Par simplicité, et vu que les ensembles de ressources et de labels sont différents, on omettra les notation de type $\lambda(r)$ et on notera simplement $r$.

On a noté les étapes successives de construction du tableau par le symbole $\surd$. En voici le détail :

1. On applique $\langle \mathbb{F} \rightarrow \rangle$ comme indiqué ci-dessus.
2. On a créé deux formules contenant l'opérateur $\mathbf{M}$, mais d'après notre ordre de priorité, on doit d'abord appliquer $\langle \mathbb{F}\mathbf{M} \rangle$ (car elle génère un nouveau label). On génère une nouvelle formule avec un nouveau label $c_2$, ainsi que la contrainte $c_1 \eqsim_a c_2 r$.
3. On se retrouve dans le même cas que précédemment, et on doit à nouveau appliquer $\langle \mathbb{F}\mathbf{M} \rangle$ et on génère une nouvelle formule avec un nouveau label $c_3$, ainsi que la contrainte $c_2 r \eqsim_a c_3 s$.

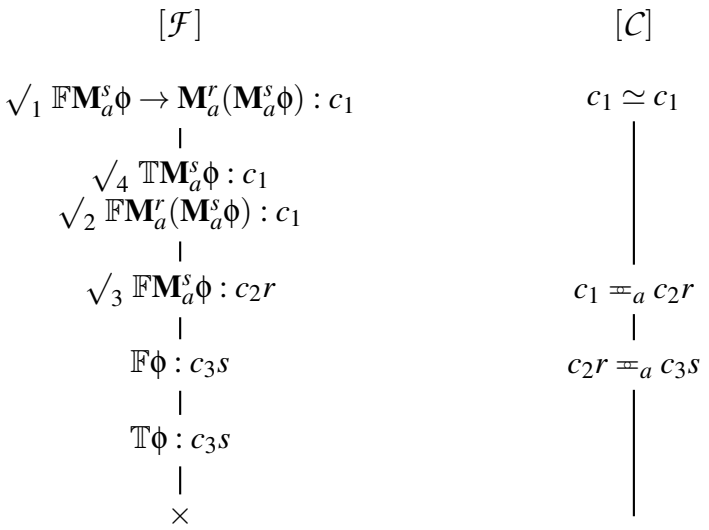



FIGURE 4.9 – Tableau pour $\mathbf{M}^s_a\phi \rightarrow \mathbf{M}^r_a(\mathbf{M}^s_a\phi)$

4. On peut enfin appliquer $\langle \mathbb{T}\mathbf{M} \rangle$. Il nous faut un label $z$ tel que $c_1 \eqcirc_a zs \in \overline{\mathcal{C}}$. Par clôture par la règle $\langle t_a \rangle$ avec l'agent $a$, on peut générer la contrainte $c_1 \eqcirc_a c_3 s$, et ainsi appliquer la règle avec $z = c_3$ ce qui produit la formule $\mathbb{T}\phi : c_3 s$. Comme on a produit précédemment $\mathbb{F}\phi : c_3 s$, on peut clore notre branche, et donc le tableau.

## 4.6 Propriétés du calcul des ERL-tableaux

Nous allons maintenant énoncer quelques résultats concernant notre calcul des tableaux. Bien évidemment, nous souhaitons que le calcul soit correct et complet vis-à-vis de notre sémantique. En outre, nous proposerons une méthode d'extraction de contre-modèles qui sera également un outil pour la preuve de complétude.

Nous serons assez concis dans les preuves que nous présentons ici (dans la mesure où elles sont similaires à celles de ESL [37] dont elle s'inspire ainsi que des calculs détaillés des Chapitres 3 et 5). On se contentera de donner le schéma des preuves et de détailler uniquement les points étant sensiblement différents dans ce formalisme.

On commence par prouver la correction du calcul.

### 4.6.1 Correction du calcul des ERL-tableaux

La correction se prouve par la notion de réalisabilité d'une branche, que nous commençons par définir.

**Définition 4.16** (Réalisation)**.** *Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un ECSS. Une* réalisation *de $\langle \mathcal{F}, \mathcal{C} \rangle$ est une paire $(\mathcal{K}, \lfloor . \rfloor)$ où $\mathcal{K} = (\mathcal{R}, \{\sim_a\}_{a \in A}, V)$ est un modèle et $\lfloor . \rfloor : \mathcal{D}_r(\mathcal{C}) \rightarrow R$ est définie par :*

1. *Pour tout $r \in Res$, on a $\lfloor \lambda(r) \rfloor = r$*
2. $\lfloor \varepsilon \rfloor = e$

3. $\lfloor . \rfloor$ *est une fonction totale (pour tout* $x \in \mathcal{D}_r(\mathcal{C})$, $\lfloor x \rfloor$ *est défini)*
4. *Si* $xy \in \mathcal{D}_r(\mathcal{C})$, *alors* $\lfloor x \rfloor \bullet \lfloor y \rfloor \downarrow$ *et* $\lfloor x \rfloor \bullet \lfloor y \rfloor = \lfloor xy \rfloor$
5. *Si* $\mathbb{T}\phi : x \in \mathcal{F}$ *alors* $\lfloor x \rfloor \models_{\mathcal{K}} \phi$
6. *Si* $\mathbb{F}\phi : x \in \mathcal{F}$ *alors* $\lfloor x \rfloor \not\models_{\mathcal{K}} \phi$
7. *Si* $x \simeq y \in \mathcal{C}$ *alors* $\lfloor x \rfloor = \lfloor y \rfloor$
8. *Si* $x =\!\!\!=_u y \in \mathcal{C}$ *alors* $\lfloor x \rfloor \sim_u \lfloor y \rfloor$

*On dit qu'un ECSS est* réalisable *s'il en existe une réalisation. On dit qu'un tableau est* réalisable *si au moins une de ses branches est réalisable.*

La notion de réalisation nous permet ainsi de dresser un modèle d'un tableau. Nous allons montrer trois résultats qui nous permettront de montrer la correction : l'extension de la réalisabilité aux clôtures de contraintes, la préservation de la réalisabilité par les règles du calcul et la non-réalisabilité des branches closes.

**Proposition 4.7.** *Soit* $\langle \mathcal{F}, \mathcal{C} \rangle$ *un ECSS et* $\mathcal{R} = (\mathcal{M}, \lfloor . \rfloor)$ *une réalisation de* $\langle \mathcal{F}, \mathcal{C} \rangle$. $\mathcal{R}$ *est aussi une réalisation de* $\langle \mathcal{F}, \overline{\mathcal{C}} \rangle$, *et donc*

1. *Pour tout* $x \in \mathcal{D}_r(\overline{\mathcal{C}})$, $\lfloor x \rfloor$ *est défini*
2. *Si* $x \simeq y \in \overline{\mathcal{C}}$ *alors* $\lfloor x \rfloor = \lfloor y \rfloor$
3. *Si* $x =\!\!\!=_u y \in \overline{\mathcal{C}}$ *alors* $\lfloor x \rfloor \sim_u \lfloor y \rfloor$

*Démonstration.*
On montre les points 1 à 8 de la Définition 4.16 pour $\langle \mathcal{F}, \overline{\mathcal{C}} \rangle$. Les points 5 et 6 ne concernent que les formules, donc sont trivialement vrais pour $\langle \mathcal{F}, \overline{\mathcal{C}} \rangle$. Les points 1 et 2 sont aussi triviaux, vu que les éléments de *Res* et $\varepsilon$ sont indépendants de $\mathcal{C}$. Enfin, on montre les points 3, 4, 7 et 8 par récurrence sur la taille de l'arbre permettant d'obtenir une contraintes de $\mathcal{D}_r(\overline{\mathcal{C}})$. □

**Lemme 4.4.** *Les règles du calcul pour ERL (règles de la Figure 4.8) préservent la réalisabilité.*

*Démonstration.*
On part d'une branche réalisable et on montre par cas que chaque règle amène au moins une branche réalisable. □

**Lemme 4.5.** *Les branches closes ne sont pas réalisables.*

*Démonstration.*
Par l'absurde. On suppose qu'une branche close est réalisable, puis on montre pour chacun des cas de clôture de la Définition 4.14 qu'il amène à une contradiction. □

Enfin, on peut montrer la correction de notre calcul :

**Théorème 4.2** (Correction)**.** *S'il existe une preuve d'une formule* $\phi$, *alors* $\phi$ *est valide.*

*Démonstration.*
On suppose qu'il existe une preuve de $\phi$. Alors il existe un tableau clos $\mathcal{T}_\phi$ pour le ECSS $\mathfrak{C} = \langle\{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\}\rangle$.

On suppose alors que $\phi$ n'est pas valide. Alors il y a un contre-modèle $\mathcal{K} = (\mathcal{R}, \{\sim_a\}_{a\in A}, V)$ et une ressource $r \in R$ tels que $r \not\Vdash_{\mathcal{K}} \phi$. Soit $\mathfrak{R} = (\mathcal{K}, \lfloor . \rfloor)$ tel que $\lfloor c_1 \rfloor = r$. Comme $\mathfrak{R}$ est une réalisation de $\mathfrak{C}$, par le Lemme 4.4, $\mathcal{T}_\phi$ est réalisable. De plus, par le lemme 4.5, $\mathcal{T}_\phi$ ne peut être clos, ce qui est absurde puisque $\mathcal{T}_\phi$ est une preuve. Donc $\phi$ est valide. □

On peut énoncer le même résultat pour ERL$^*$ :

**Théorème 4.3** (Correction du calcul pour ERL$^*$)**.** *S'il existe une preuve d'une formule $\phi$ de* ERL$^*$*, alors $\phi$ est valide.*

*Démonstration.*
La preuve est identique à celle du Théorème 4.2, les lemmes 4.4 et 4.5 ainsi que la Proposition 4.7 doivent être redémontrées pour les clôtures de ERL$^*$. Il suffit de traiter le cas de la règle supplémentaire énoncée dans la Définition 4.11 dans chaque récurrence sur la taille d'un arbre de déduction d'une clôture de contrainte. Cela ne soulève pas de difficulté supplémentaire. □

### 4.6.2 Extraction de contres-modèles

Nous présentons maintenant une méthode pour extraire des contre-modèles à partir d'un tableau pour une formule qui n'est pas clos. Dans un premier temps, nous devons nous assurer que le tableau est réellement non clos et complètement développé (toutes les extensions par des règles du calcul sont épuisées). Pour cela nous introduisons la notion de ECSS de Hintikka.

**Définition 4.17** (ECSS de Hintikka)**.** *Un ECSS $\langle \mathcal{F}, \mathcal{C}\rangle$ est un* ECSS de Hintikka *si pour toutes formules $\phi, \psi \in \mathcal{L}$, toute ressource $r \in Res$, tous labels $x, y, z \in L_r$ et tout agent $u \in A$ :*

1. $\mathbb{T}\phi : x \notin \mathcal{F}$ *ou* $\mathbb{F}\phi : y \notin \mathcal{F}$ *ou* $x \simeq y \notin \overline{\mathcal{C}}$,
2. $\mathbb{F}\mathrm{I} : x \notin \mathcal{F}$ *ou* $x \simeq \varepsilon \notin \overline{\mathcal{C}}$,
3. $\mathbb{F}\top : x \notin \mathcal{F}$,
4. $\mathbb{T}\bot : x \notin \mathcal{F}$,
5. *Si* $\mathbb{T}\mathrm{I} : x \in \mathcal{F}$, *alors* $x \simeq \varepsilon \in \overline{\mathcal{C}}$,
6. *Si* $\mathbb{T}\neg\phi : x \in \mathcal{F}$, *alors* $\mathbb{F}\phi : x \in \mathcal{F}$,
7. *Si* $\mathbb{F}\neg\phi : x \in \mathcal{F}$, *alors* $\mathbb{T}\phi : x \in \mathcal{F}$,
8. *Si* $\mathbb{T}\phi \wedge \psi : x \in \mathcal{F}$, *alors* $\mathbb{T}\phi : x \in \mathcal{F}$ *et* $\mathbb{T}\psi : x \in \mathcal{F}$,
9. *Si* $\mathbb{F}\phi \wedge \psi : x \in \mathcal{F}$, *alors* $\mathbb{F}\phi : x \in \mathcal{F}$ *ou* $\mathbb{F}\psi : x \in \mathcal{F}$,
10. *Si* $\mathbb{T}\phi \vee \psi : x \in \mathcal{F}$, *alors* $\mathbb{T}\phi : x \in \mathcal{F}$ *ou* $\mathbb{T}\psi : x \in \mathcal{F}$,
11. *Si* $\mathbb{F}\phi \vee \psi : x \in \mathcal{F}$, *alors* $\mathbb{F}\phi : x \in \mathcal{F}$ *et* $\mathbb{F}\psi : x \in \mathcal{F}$,
12. *Si* $\mathbb{T}\phi \to \psi : x \in \mathcal{F}$, *alors* $\mathbb{F}\phi : x \in \mathcal{F}$ *ou* $\mathbb{T}\psi : x \in \mathcal{F}$,
13. *Si* $\mathbb{F}\phi \to \psi : x \in \mathcal{F}$, *alors* $\mathbb{T}\phi : x \in \mathcal{F}$ *et* $\mathbb{F}\psi : x \in \mathcal{F}$,
14. *Si* $\mathbb{T}\phi * \psi : x \in \mathcal{F}$, *alors* $\exists y, z \in L_r$, $x \simeq yz \in \overline{\mathcal{C}}$ *et* $\mathbb{T}\phi : y \in \mathcal{F}$ *et* $\mathbb{T}\psi : z \in \mathcal{F}$,

15. *Si* $\mathbb{F}\phi * \psi : x \in \mathcal{F}$, *alors* $\forall y,z \in L_r$, $x \simeq yz \in \overline{\mathcal{C}}$ *implique* $\mathbb{F}\phi : y \in \mathcal{F}$ *ou* $\mathbb{F}\psi : z \in \mathcal{F}$,
16. *Si* $\mathbb{T}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $xy \in \mathcal{D}_r$ *implique* $\mathbb{F}\phi : y \in \mathcal{F}$ *ou* $\mathbb{T}\psi : xy \in \mathcal{F}$,
17. *Si* $\mathbb{F}\phi \mathrel{-\!\!*} \psi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $xy \in \mathcal{D}_r$ *et* $\mathbb{T}\phi : y \in \mathcal{F}$ *et* $\mathbb{F}\psi : xy \in \mathcal{F}$,
18. *Si* $\mathbb{T}\mathbf{L}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x\lambda(r) =_u y \in \overline{\mathcal{C}}$ *implique* $\mathbb{T}\phi : y \in \mathcal{F}$,
19. *Si* $\mathbb{F}\mathbf{L}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x\lambda(r) =_u y \in \overline{\mathcal{C}}$ *et* $\mathbb{F}\phi : y \in \mathcal{F}$,
20. *Si* $\mathbb{T}\mathbf{M}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x =_u y\lambda(r) \in \overline{\mathcal{C}}$ *implique* $\mathbb{T}\phi : y\lambda(r) \in \mathcal{F}$,
21. *Si* $\mathbb{F}\mathbf{M}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x =_u y\lambda(r) \in \overline{\mathcal{C}}$ *et* $\mathbb{F}\phi : y\lambda(r) \in \mathcal{F}$,
22. *Si* $\mathbb{T}\mathbf{N}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}$ *implique* $\mathbb{T}\phi : y\lambda(r) \in \mathcal{F}$,
23. *Si* $\mathbb{F}\mathbf{N}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}$ *et* $\mathbb{F}\phi : y\lambda(r) \in \mathcal{F}$,
24. *Si* $\mathbb{T}\widetilde{\mathbf{L}}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x\lambda(r) =_u y \in \overline{\mathcal{C}}$ *et* $\mathbb{T}\phi : y \in \mathcal{F}$,
25. *Si* $\mathbb{F}\widetilde{\mathbf{L}}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x\lambda(r) =_u y \in \overline{\mathcal{C}}$ *implique* $\mathbb{F}\phi : y \in \mathcal{F}$,
26. *Si* $\mathbb{T}\widetilde{\mathbf{M}}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x =_u y\lambda(r) \in \overline{\mathcal{C}}$ *et* $\mathbb{T}\phi : y\lambda(r) \in \mathcal{F}$,
27. *Si* $\mathbb{F}\widetilde{\mathbf{M}}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x =_u y\lambda(r) \in \overline{\mathcal{C}}$ *implique* $\mathbb{F}\phi : y\lambda(r) \in \mathcal{F}$,
28. *Si* $\mathbb{T}\widetilde{\mathbf{N}}^r_u\phi : x \in \mathcal{F}$, *alors* $\exists y \in L_r$, $x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}$ *et* $\mathbb{T}\phi : y\lambda(r) \in \mathcal{F}$,
29. *Si* $\mathbb{F}\widetilde{\mathbf{N}}^r_u\phi : x \in \mathcal{F}$, *alors* $\forall y \in L_r$,, $x\lambda(r) =_u y\lambda(r) \in \overline{\mathcal{C}}$ *implique* $\mathbb{F}\phi : y\lambda(r) \in \mathcal{F}$,

Les conditions 1 à 4 assurent qu'un ECSS de Hintikka n'est pas clos, les conditions 5 à 29 assurent qu'il est saturé (c'est à dire que toute application d'une règle de la Figure 4.8 ne change pas le ECSS).

Avant de décrire l'extraction de contre-modèle à proprement parler, nous devons pouvoir associer à nos labels des ressources de notre modèle. Une première étape, qui est standard, est d'extraire les classes d'équivalence des labels par rapport à la relation $\simeq$ (ce qui permet d'assurer que des labels équivalents seront représentés par une même ressource).

Toutefois, contrairement aux autres calculs déjà développés, il nous faut aller plus loin. En effet, nous avons besoin d'assurer que les labels représentant des ressources de la syntaxe *Res* soient bien reliées à ces ressources. Aussi, nous ne pouvons pas nous contenter d'utiliser les classes d'équivalences des labels comme ressource (ce qui est habituellement fait), il nous faut choisir un représentant par classe et, le cas échéant, choisir la ressource de *Res* concernée si nécessaire.

**Définition 4.18** (Classe d'équivalence, représentant)**.** *Soit $\mathcal{C}$ un ensemble de contraintes et $x \in \mathcal{D}_r(\overline{\mathcal{C}})$ un label.*

*La* classe d'équivalence *de x, notée $[x]$, est l'ensemble suivant :*

$$[x] = \{y \in L_r \mid x \simeq y \in \overline{\mathcal{C}}\}$$

.

*Le représentant de x, noté $\rho_x$ est défini par :*

- *S'il existe $r \in Res$ tel que $\lambda(r) \in [x]$, alors $\rho_x = r$ ;*
- *Sinon, on choisit un $y \in [x]$ arbitrairement et $\rho_x = y$.*

*On note $Rep(\mathcal{D}_r(\overline{C}))$ l'ensemble de tous les représentants de $\mathcal{D}_r(\overline{C})$, c'est à dire $Rep(\mathcal{D}_r(\overline{C})) = \{\rho_x \mid x \in \mathcal{D}_r(\overline{C})\}$.*

Notons qu'on a les résultats triviaux suivants :

- Pour tout $x$ on a $x \in [x]$ ;
- Pour tout $r \in Res$ on a $\rho_{\lambda(r)} = r$ ;
- Pour tout $x$ tel qu'il n'existe aucun $r \in Res$ tel que $\lambda(r) \in [x]$, on peut choisir $\rho_x = x$.

On a en outre le résultat suivant :

**Lemme 4.6.** *Pour tout ensemble de contrainte $\mathcal{C}$, on a $e \in Rep(\mathcal{D}_r(\overline{C}))$ et $\rho_\varepsilon = e$.*

*Démonstration.*
Par la règle $\langle\varepsilon\rangle$ de la Figure 4.7, on a $\varepsilon \in \mathcal{D}_r(\overline{C})$. En outre, on a $\varepsilon = \lambda(e)$ par la Définition 4.9. Donc, comme $\varepsilon \in [\varepsilon]$, on a $\lambda(e) \in [\varepsilon]$ et donc $\rho_\varepsilon = e$. Par suite, on a par construction que $e \in Rep(\mathcal{D}_r(\overline{C}))$. □

Nous pouvons maintenant définir la fonction oméga qui extrait d'une branche saturée un contre-modèle :

**Définition 4.19** (Fonction $\Omega$)**.** *Soit $\langle \mathcal{F}, \mathcal{C}\rangle$ un ECSS de Hintikka. La fonction $\Omega$ associe à $\langle \mathcal{F}, \mathcal{C}\rangle$ un triplet $\Omega(\langle \mathcal{F}, \mathcal{C}\rangle) = (\mathcal{R}, \{\sim_a\}_{a\in A}, [\![\cdot]\!], \vDash_{\mathcal{K}})$, où $\mathcal{R} = (R, \bullet, \sim)$, tel que :*

- *$R = Rep(\mathcal{D}_r(\overline{C})) \cup Res$*
- *si $\alpha \notin Rep(\mathcal{D}_r(\overline{C}))$ ou $\beta \notin Rep(\mathcal{D}_r(\overline{C}))$, $\alpha \bullet \beta = \uparrow$.*
- *sinon, $\alpha = \rho_x$ et $\beta = \rho_y$, et on a $\rho_x \bullet \rho_y = \begin{cases} \uparrow & si\ xy \notin \mathcal{D}_r(\overline{C}) \\ \rho_{xy} & sinon \end{cases}$*
- *$\sim$ est une relation d'équivalence sur R indéfinie sinon pour la réflexivité*
- *Pour tout $a \in A$, $\alpha \sim_a \beta$ ssi $\alpha = \rho_x$ et $\beta = \rho_y$ et $x =_a y \in \overline{C}$*
- *$\alpha \in [\![p]\!]$ si $\alpha = \rho_x$ et $\exists y \in L_r$ tel que $y \simeq x \in \overline{C}$ et $\mathbb{T}p : y \in \mathcal{F}$*
- *$\vDash_{\mathcal{K}}$ est défini inductivement à partir des éléments précédents de la même manière que dans la Définition 4.4.*

Il nous faut maintenant vérifier que la structure produite par $\Omega$ est bien un modèle.

**Lemme 4.7.** *Soit $\langle \mathcal{F}, \mathcal{C}\rangle$ un ECSS de Hintikka. $\Omega(\langle \mathcal{F}, \mathcal{C}\rangle)$ est un modèle.*

*Démonstration.*
On vérifie simplement tous les points de la Définition 4.4. La preuve ne présente pas de difficulté particulière, les propriétés des labels et des contraintes traduisant les propriétés nécessaires pour les ressources. □

Nous voulons maintenant prouver que le modèle par $\Omega$ est bien un contre-modèle dans le cas d'un tableau clos pour une formule.

**Lemme 4.8.** *Soit $\langle \mathcal{F}, \mathcal{C}\rangle$ un ECSS de Hintikka et $\mathcal{K} = \Omega(\langle \mathcal{F}, \mathcal{C}\rangle) = (\mathcal{R}, \{\sim_a\}_{a\in A}, [\![\cdot]\!], \vDash_{\mathcal{K}})$, avec $\mathcal{R} = (R, \bullet, \sim)$. Pour toute formule $\phi \in \mathcal{L}$, tout agent $a \in A$ et tout label $x \in \mathcal{D}_r(\overline{C})$, on a :*

1. *Si* $\mathbb{F}\phi : x \in \mathcal{F}$ *alors* $\rho_x \nVdash_{\mathcal{K}} \phi$ *;*
2. *Si* $\mathbb{T}\phi : x \in \mathcal{F}$ *alors* $\rho_x \vDash_{\mathcal{K}} \phi$

*Démonstration.*
Par récurrence sur la taille de $\phi$. La saturation assurée par la Définition 4.17 permet de reporter la propriété sur les sous-formules et d'utiliser ainsi l'hypothèse de récurrence. □

Ce qui entraîne simplement le résultat suivant :

**Lemme 4.9.** *Soit* $\langle \mathcal{F}, \mathcal{C} \rangle$ *un ECSS de Hintikka tel que* $\mathbb{F}\phi : x \in \mathcal{F}$*. La formule* $\phi$ *n'est pas valide et* $\Omega(\langle \mathcal{F}, \mathcal{C} \rangle)$ *est un contre-modèle de* $\phi$*.*

*Démonstration.*
Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un ECSS de Hintikka tel que $\mathbb{F}\phi : x \in \mathcal{F}$. Soit $\mathcal{K} = \Omega(\langle \mathcal{F}, \mathcal{C} \rangle)$. Par le Lemme 4.7, $\mathcal{K}$ est un modèle. Comme $\langle \mathcal{F}, \mathcal{C} \rangle$ est un ECSS, alors par $P_{ECSS}$ et le Corollaire 2, $x \in \mathcal{D}_r(\overline{\mathcal{C}})$. Donc, par le lemme 4.8, on a $\rho_x \nVdash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est un contre-modèle de la formule $\phi$ et par suite $\phi$ n'est pas valide. □

Ainsi, on a montré que la construction de $\Omega(\langle \mathcal{F}, \mathcal{C} \rangle)$ pour un ECSS de Hintikka $\langle \mathcal{F}, \mathcal{C} \rangle$ permet d'extraire un contre-modèle d'une formule. Notons que cette méthode est identique pour les tableaux ERL$^*$ (voir Définition 4.15).

### 4.6.3 Un exemple d'extraction de contre-modèles

Nous allons à présent illustrer notre méthode d'extraction de contre-modèle par un exemple.

On considère la formule $\mathbf{L}^s_a\phi \rightarrow \mathbf{L}^r_a(\mathbf{L}^s_a\phi)$. On a $A = \{a\}$ et $Res = \{e, r\}$. On dresse le tableau pour $\phi$. Le tableau obtenu est représenté dans la Figure 4.10.

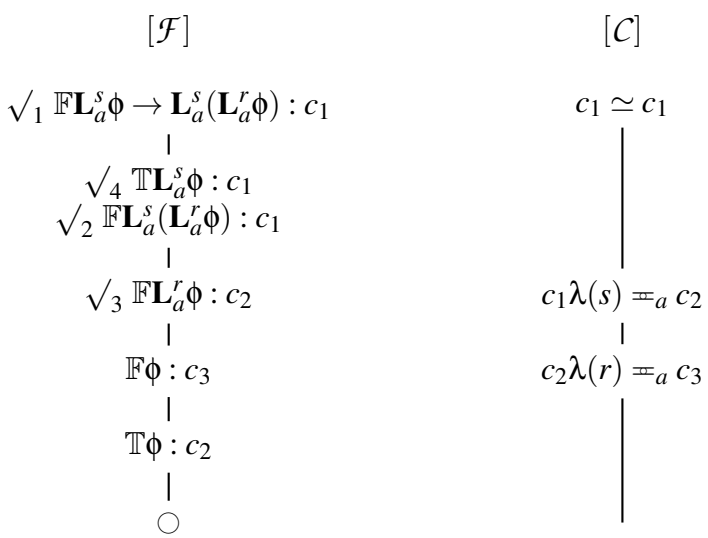



FIGURE 4.10 – Tableau pour $\mathbf{L}^s_a\phi \rightarrow \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$

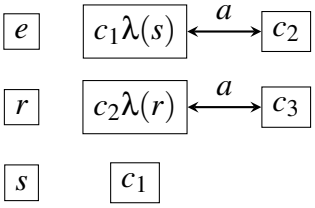
On peut voir comment, à l'étape 4 on ne peut trouver que $c_2$ comme label convenable pour $c_1 s \eqdot_a x$ et ainsi le tableau n'es pas clos. L'unique branche du tableau est donc ouverte, et c'est une branche de Hintikka. On peut donc extraire un contre-modèle avec la Définition 4.19.

On construit donc le modèle $\Omega(\langle\{\mathbb{F}\mathbf{L}^s_a\phi \to \mathbf{L}^s_a(\mathbf{L}^r_a\phi) : c_1\}, \{\rangle c_1 \simeq c_1\}) = (\mathcal{R}, \{\sim_a\}_{a\in A}, [\![\cdot]\!], \vDash_{\mathcal{K}})$, où $\mathcal{R} = (R, \bullet, \sim)$.

- Pour déterminer $R$ nous devons d'abord déterminer la classe d'équivalence de chacun des labels. Comme nous n'avons aucune relation de type $x \simeq y$ (sinon les relations réflexives) dans la clôture, on a $[x] = \{x\}$ pour tout $x \in \mathcal{D}_r(\overline{C})$, et par suite $\rho_x = x$ pour tout $x$ à l'exception de $rho_{\lambda(s)} = s$ et $rho_{\lambda(r)} = r$. Finalement, on a $R = \{e, c_1, c_2, c_3, s, r, c_1\lambda(s), c_2\lambda(r)\}$.
- La composition de ressources est donnée par le tableau suivant :

| $\bullet$ | $e$ | $r$ | $s$ | $c_1$ | $c_2$ | $c_3$ | $c_1\lambda(s)$ | $c_2\lambda(r)$ |
|---|---|---|---|---|---|---|---|---|
| $e$ | $e$ | $r$ | $s$ | $c_1$ | $c_2$ | $c_3$ | $c_1\lambda(s)$ | $c_2\lambda(r)$ |
| $r$ | $r$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $c_2\lambda(r)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $s$ | $s$ | $\uparrow$ | $\uparrow$ | $c_1\lambda(s)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $c_1$ | $c_1$ | $\uparrow$ | $c_1\lambda(s)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $c_2$ | $c_2$ | $c_2\lambda(r)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $c_3$ | $c_3$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $c_1\lambda(s)$ | $c_1\lambda(s)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |
| $c_2\lambda(r)$ | $c_2\lambda(r)$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ | $\uparrow$ |

- Les relations d'équivalence ($\sim$ et $\sim_a$) sont données par le schéma suivant (la réflexivité n'est pas représentée) :




- L'interprétation est uniquement définie pour $\phi$ et on a $[\![\phi]\!] = \{c_2\}$.

On peut maintenant vérifier que nous avons bien obtenu un contre-modèle de $\mathbf{L}^s_a\phi \to \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$.

1. Comme $c_2 \in [\![\phi]\!]$, on a $c_2 \vDash \phi$.
2. Comme $\{x \in R | c_1 \bullet s \sim_a x\} = \{c_2\}$, on a par (1), $c_1 \vDash \mathbf{L}^s_a\phi$.
3. Comme $c_3 \notin [\![\phi]\!]$, on a $c_3 \nvDash \phi$.
4. Comme $c_2 \bullet r = c_2\lambda(r)$ et que $c_2\lambda(r) \sim_a c_3$, par (3) on a $c_2 \nvDash \mathbf{L}^r_a\phi$.
5. Comme $c_1 \bullet s = c_1\lambda(s)$ et $c_1\lambda(s) \sim_a c_2$, par (4) on a $c_1 \nvDash \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$.
6. Par (2) et (5) on conclue que $c_1 \nvDash \mathbf{L}^s_a\phi \to \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$.

Comme on a trouvé une ressource $c_1$ de notre modèle telle que $c_1 \nvDash \mathbf{L}^s_a\phi \to \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$, notre formule $\mathbf{L}^s_a\phi \to \mathbf{L}^s_a(\mathbf{L}^r_a\phi)$ est bien non valide et notre modèle en est un contre-modèle.

### 4.6.4 Complétude du calcul des ERL-tableaux

La preuve de complétude consiste en la construction d'une branche de Hintikka à partir d'une branche non close d'un tableau en utilisant une *stratégie équitable* et un *oracle*.

**Définition 4.20** (Stratégie équitable)**.** *Une* stratégie équitable *est une suite de formule étiquetées et de contraintes d'agents* $(S_i)_{i\in\mathbb{N}}$ *avec* $S_i \in (\{\mathbb{T},\mathbb{F}\}\times\mathcal{L}\times L_r)\cup(L_r\times A\times L_r)$ *pour tout* $i\in\mathbb{N}$ *tel que chacune des formules étiquetées et chacune des contraintes d'agents apparaisse un nombre infini de fois dans la suite, c'est à dire que* $\{i\in\mathbb{N}\mid S_i\equiv\mathbb{S}F:x\}$ *et* $\{i\in\mathbb{N}\mid S_i\equiv x=\!\!\!=_u y\}$ *sont infinis pour toute* $\mathbb{S}F:x\in\{\mathbb{T},\mathbb{F}\}\times\mathcal{L}\times L_r$ *et toute* $x=\!\!\!=_u y\in L_r\times A\times L_r$.

**Proposition 4.8.** *Il existe une stratégie équitable.*

*Démonstration.*
La preuve repose sur la cardinalité des divers ensembles impliqués dans la définition d'une stratégie équitable et le fait qu'ils soient tous dénombrables. □

La stratégie équitable va nous permettre de construire une branche saturée en sélectionnant des formules arbitraires et en les ajoutant à notre branche. La répétition de chacune des formules permet de certifier la saturation. Toutefois, nous devons également sélectionner les formules nous permettant de construire une branche ouverte. C'est l'oracle qui va nous permettre cette opération. Un oracle est simplement un ensemble de formules vérifiant certaines propriétés que nous allons tout d'abord énoncer :

**Définition 4.21** (ECSS clos, fini, saturé)**.** *Soit* $\mathcal{P}$ *un ensemble de ECSS.*

1. $\mathcal{P}$ *est* $\preccurlyeq$-clos *si on a* $\langle\mathcal{F},\mathcal{C}\rangle\in\mathcal{P}$ *chaque fois que* $\langle\mathcal{F},\mathcal{C}\rangle\preccurlyeq\langle\mathcal{F}',\mathcal{C}'\rangle$ *et* $\langle\mathcal{F}',\mathcal{C}'\rangle\in\mathcal{P}$.
2. $\mathcal{P}$ *est de* caractère fini *si on a* $\langle\mathcal{F},\mathcal{C}\rangle\in\mathcal{P}$ *chaque fois que* $\langle\mathcal{F}_f,\mathcal{C}_f\rangle\in\mathcal{P}$, *pour tout* $\langle\mathcal{F}_f,\mathcal{C}_f\rangle\preccurlyeq_f\langle\mathcal{F},\mathcal{C}\rangle$.
3. $\mathcal{P}$ *est* saturé *si pour tout* $\langle\mathcal{F},\mathcal{C}\rangle\in\mathcal{P}$ *et pour toute instance*

$$\frac{cond(\mathcal{F},\mathcal{C})}{\langle\mathcal{F}_1,\mathcal{C}_1\rangle \mid \ldots \mid \langle\mathcal{F}_k,\mathcal{C}_k\rangle}$$

*d'une règle de la Figure 4.8, si* $cond(\mathcal{F},\mathcal{C})$ *est vérifiée, alors* $\langle\mathcal{F}\cup\mathcal{F}_i,\mathcal{C}\cup\mathcal{C}_i\rangle\in\mathcal{P}$ *pour au moins un* $i\in\{1,\ldots,k\}$.

Ce qui nous conduit à la définition de l'oracle :

**Définition 4.22** (Oracle)**.** *Un* oracle *est un ensemble de ECSS non clos qui est* $\preccurlyeq$*-clos, de caractère fini et saturé.*

Et on a le résultat suivant qui est central pour notre preuve de complétude.

**Lemme 4.10.** *Il existe un oracle qui contient tous les ECSS pour lesquels il n'existe aucun tableau clos.*

Pour prouver la complétude du calcul, on considère ensuite une formule $\phi$ pour laquelle il n'existe pas de preuve et on montre qu'il en existe un contre-modèle. On note $\mathcal{T}_0$ le tableau initial pour $\phi$, ce qui veut dire que :

1. $\mathcal{T}_0 = [\langle \{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\}\rangle]$
2. $\mathcal{T}_0$ ne peut pas être clos (puisqu'il n'y a pas de preuve).

On doit ensuite construire une branche de Hintikka à partir de ce tableau initial. Par le Lemme 4.10, il existe un oracle $\mathcal{P}$ qui contient tous les ECSS pour lesquels il n'existe aucun tableau clos. Par la Proposition 4.8, il existe une stratégie équitable $\mathcal{S} = \{\mathcal{S}_i\}_{i\in\mathbb{N}}$. Comme $\mathcal{T}_0$ ne peut être clos, son unique branche appartient à l'oracle $\mathcal{P}$, c'est à dire que $\langle \{\mathbb{F}\varphi : c_1\}, \{c_1 \simeq c_1\}\rangle \in \mathcal{P}$.

On construit alors une suite $\langle \mathcal{F}_i, \mathcal{C}_i\rangle_{i\geqslant 0}$ comme suit :

- $\langle \mathcal{F}_0, \mathcal{C}_0\rangle = \langle \{\mathbb{F}\varphi : c_1\}, \{c_1 \simeq c_1\}\rangle$
- Si $\mathcal{S}_i$ est une formule étiquetée de la forme $\mathbb{S}F : x$ :
  - Si $\langle \mathcal{F}_i \cup \{\mathbb{S}F : x\}, \mathcal{C}_i\rangle \notin \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{i+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_i\rangle$
  - Si $\langle \mathcal{F}_i \cup \{\mathbb{S}F : x\}, \mathcal{C}_i\rangle \in \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{i+1}\rangle = \langle \mathcal{F}_i \cup \{\mathbb{S}F : x\} \cup F_e, \mathcal{C}_i \cup \mathcal{C}_e\rangle$ où $F_e$ et $\mathcal{C}_e$ sont donnés par :

| $\mathbb{S}_i$ | $F_i$ | $F_e$ | $\mathcal{C}_e$ |
|---|---|---|---|
| $\mathbb{T}$ | $\mathbf{I}$ | $\emptyset$ | $\{x \simeq \mathfrak{e}\}$ |
| $\mathbb{T}$ | $\phi * \psi$ | $\{\mathbb{T}\phi : \mathfrak{a}, \mathbb{T}\psi : \mathfrak{b}\}$ | $\{x \simeq \mathfrak{a}\mathfrak{b}\}$ |
| $\mathbb{F}$ | $\phi \mathrel{-\!\!*} \psi$ | $\{\mathbb{T}\phi : \mathfrak{a}, \mathbb{F}\psi : x\mathfrak{a}\}$ | $\{x\mathfrak{a} \simeq x\mathfrak{a}\}$ |
| $\mathbb{F}$ | $\mathbf{L}^r_u\phi$ | $\{\mathbb{F}\phi : \mathfrak{a}\}$ | $\{x\lambda(r) =\!\!=_u \mathfrak{a}\}$ |
| $\mathbb{F}$ | $\mathbf{M}^r_u\phi$ | $\{\mathbb{F}\phi : \mathfrak{a}\lambda(r)\}$ | $\{x =\!\!=_u \mathfrak{a}\lambda(r)\}$ |
| $\mathbb{F}$ | $\mathbf{N}^r_u\phi$ | $\{\mathbb{F}\phi : \mathfrak{a}\lambda(r)\}$ | $\{x\lambda(r) =\!\!=_u \mathfrak{a}\lambda(r)\}$ |
| $\mathbb{T}$ | $\widetilde{\mathbf{L}}^r_u\phi$ | $\{\mathbb{T}\phi : \mathfrak{a}\}$ | $\{x\lambda(r) =\!\!=_u \mathfrak{a}\}$ |
| $\mathbb{T}$ | $\widetilde{\mathbf{M}}^r_u\phi$ | $\{\mathbb{T}\phi : \mathfrak{a}\lambda(r)\}$ | $\{x =\!\!=_u \mathfrak{a}\lambda(r)\}$ |
| $\mathbb{T}$ | $\widetilde{\mathbf{N}}^r_u\phi$ | $\{\mathbb{T}\phi : \mathfrak{a}\lambda(r)\}$ | $\{x\lambda(r) =\!\!=_u \mathfrak{a}\lambda(r)\}$ |
| Sinon | | $\emptyset$ | $\emptyset$ |

avec $\mathfrak{a} = c_{2i+2}$ et $\mathfrak{b} = c_{2i+3}$.

- Si $\mathcal{S}_i$ est une contrainte d'agent de la forme $x\lambda(r) =\!\!=_u y$ :
  - Si $\gamma_r \cap (\mathcal{E}(x) \cup \mathcal{E}(y)) \not\subseteq \{c_1, ..., c_{2i+1}\}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{i+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_i\rangle$
  - Si $\langle \mathcal{F}_i, \mathcal{C}_i \cup \{x\lambda(r) =\!\!=_u y\}\rangle \notin \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{i+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_i\rangle$
  - Si $\langle \mathcal{F}_i, \mathcal{C}_i \cup \{x\lambda(r) =\!\!=_u y\}\rangle \in \mathcal{P}$ alors $\langle \mathcal{F}_{i+1}, \mathcal{C}_{i+1}\rangle = \langle \mathcal{F}_i, \mathcal{C}_i \cup \{x\lambda(r) =\!\!=_u y\}\rangle$

Avec ce procédé, on construit une branche qui va tendre vers un ECSS de Hintikka. On va montrer ce résultat.

**Proposition 4.9.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathbb{F}\phi : c_1 \in \mathcal{F}_i$ *et* $c_1 \simeq c_1 \in \mathcal{C}_i$
2. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$ *et* $\mathcal{C}_i \subseteq \mathcal{C}_{i+1}$

3. $\langle \mathcal{F}_i, \mathcal{C}_i \rangle_{i \geqslant 0} \in \mathcal{P}$
4. $\mathcal{A}_r(\mathcal{C}_i) \subseteq \{c_1, c_2, \ldots, c_{2i+1}\}$

*Démonstration.*
1. et 2. sont triviaux vu que $\mathcal{C}_i$ est une suite croissante pour l'inclusion. On prouve 3. et 4. par récurrence sur $i$ et par cas selon le type de $\mathcal{S}_i$.

□

Le ECSS limite $\langle \mathcal{F}_\infty, \mathcal{C}_\infty \rangle$ de la suite $\langle \mathcal{F}_i, \mathcal{C}_i \rangle_{i \geqslant 0}$ est défini par

$$\mathcal{F}_\infty = \bigcup_{i \geqslant 0} \mathcal{F}_i \quad \text{et} \quad \mathcal{C}_\infty = \bigcup_{i \geqslant 0} \mathcal{C}_i$$

**Proposition 4.10.** *On a les propriétés suivantes :*

1. $\langle \mathcal{F}_\infty, \mathcal{C}_\infty \rangle \in \mathcal{P}$
2. *Pour toute formule étiquetée* $\mathbb{S}\phi : x$, *si* $\langle \mathcal{F}_\infty \cup \{\mathbb{S}\phi : x\}, \mathcal{C}_\infty \rangle \in \mathcal{P}$ *alors* $\mathbb{S}\phi : x \in \mathcal{F}_\infty$
3. *Pour toute contrainte d'agent* $x\lambda(r) =_u y$, *si* $\langle \mathcal{F}_\infty, \mathcal{C}_\infty \cup \{x\lambda(r) =_u y\} \rangle \in \mathcal{P}$ *alors* $x\lambda(r) =_u y \in \mathcal{C}_\infty$

Ces résultats indiquent que le ECSS limite fait partie de l'oracle et qu'il est en outre saturé (tout ajout de règle ou de contrainte d'agent le laisse inchangé).

**Lemme 4.11.** *Le ECSS limite est un ECSS de Hintikka.*

*Démonstration.*
Par la Proposition 4.10, $\langle \mathcal{F}_\infty, \mathcal{C}_\infty \rangle \in \mathcal{P}$. On vérifie alors que toutes les propriétés de la Définition 4.17 sont vérifiée. □

On a donc réussi à construire à partir de notre formule $\phi$ une branche de Hintikka, ce qui assure par notre méthode d'extraction de contre-modèle que la formule n'est pas valide. D'où le théorème de complétude :

**Théorème 4.4** (Complétude)**.** *Soit* $\phi$ *une formule. Si* $\phi$ *est valide, alors il existe une preuve de* $\phi$.

*Démonstration.*
Supposons qu'il n'y ait pas de preuve de $\phi$. On montre que $\phi$ n'est pas valide. La méthode que nous avons présentée ci-dessus nous permet de construire un ECSS limite $\langle \mathcal{F}_\infty, \mathcal{C}_\infty \rangle$ à partir de $\phi$. Par le Lemme 4.11, ce ECSS est un ECSS de Hintikka. Par le point 1 de la Proposition 4.9, $\mathbb{F}\phi : c_1 \in \mathcal{F}_i$ pour tout $i \geqslant 0$, donc $\mathbb{F}\phi : c_1 \in \mathcal{F}_\infty$ par construction du ECSS limite. Enfin, par le Lemme 4.8, $\phi$ n'est pas valide. C'est absurde, donc il existe une preuve de $\phi$. □

Notre calcul est donc bien complète. Le résultat est aussi valable pour ERL$^*$ :

**Théorème 4.5** (Complétude du calcul pour ERL$^*$)**.** *Soit* $\phi$ *une formule de* ERL$^*$. *Si* $\phi$ *est valide, alors il existe une preuve de* $\phi$.

*Démonstration.*
La preuve est identique à celle du Théorème 4.4, les lemmes et propositions doivent être redémontrées pour les clôtures de ERL$^*$. Il suffit de traiter le cas de la règle supplémentaire énoncée dans la Définition 4.11 dans chaque récurrence sur la taille d'un arbre de déduction d'une clôture de contrainte. Cela ne soulève pas de difficulté supplémentaire. □

# Conclusion

Dans la lignée de l'extension épistémique ESL, qui ajoutait à BBI des modalités épistémiques modélisant la connaissance d'agents, nous avons proposé une nouvelle logique ERL qui est une extension épistémique de BBI avec des opérateurs prenant en compte l'accès à des ressources à la fois par des agents particulier et sous condition de l'accès à une ressource particulière.

Nous avons également proposé une variante ERL$^*$ qui rend compatible la composition des ressources et l'accès des agents à celles-ci, ce qui met à jour d'intéressantes interactions entre nos nouveaux opérateurs.

Nous avons montré comment ERL est une extension conservative à la fois de BBI et de la logique épistémique, ce qui en fait un outil potentiel pour raisonner sur l'un et sur l'autre (notamment par le calcul des tableaux).

Nous avons montré comment ERL est un outil particulièrement adapté pour modéliser des problèmes de contrôle d'accès, avec des accès conditionnés par la possession de ressources particulières (symbolisant les clés, cartes et codes des systèmes concrets), y compris dans des situations complexes telles que des ressources à accès conjoint ou concurrent. Nous avons aussi exposé comment la structure d'ERL permet d'adopter la technique de modélisation par couches.

Nous avons enfin présenté un calcul des tableaux correct et complet pour ERL, avec une méthode d'extraction de contre-modèles.

Notons que la modélisation avec ERL pourrait être améliorée par son extension, par exemple par un système d'annonces comme pour ESL [33], ou encore l'internalisation de ressources, qui est le sujet de notre prochain chapitre.

# Chapitre 5

# Logiques BBI hybrides

Nous avons exploré jusqu'à maintenant deux extensions distinctes des logiques de séparation BI et BBI : une extension de BI par des opérateurs modaux temporels d'une part, une extension de BBI par des opérateurs modaux épistémiques d'autre part. On se propose maintenant non pas d'étendre les capacités d'expressivité d'une logique existante par de nouveaux opérateurs, mais plutôt de revisiter la sémantique originelle de BBI pour en accroître l'expressivité.

Nous nous sommes inspirés de la Logique Hybride [14–16,89], extension de la logique modale, pour incorporer à la syntaxe de BBI des symboles permettant de spécifier la ressource validant une formule. Cet ajout a deux intérêts majeurs.

Premièrement, il étend l'expressivité de notre logique en autorisant la caractérisation d'une formule par une ressource directement au niveau de la syntaxe. Cette capacité facilite la modélisation de systèmes et permet notamment d'exprimer que deux formules distinctes évoluent dans le même contexte.

Deuxièmement, notre ajout permet d'expliciter dans le langage logique lui-même l'axiomatique qui caractérise la composition des ressources. On peut ainsi définir une logique de séparation minimaliste puis ajouter des schémas d'axiomes pour retrouver la sémantique de BBI ou bien encore explorer d'autres variantes de BBI en omettant certains axiomes ou en ajoutant d'avantage. On peut par exemple considérer le cas d'une logique où les ressources ne peuvent pas commuter, comme cela a déjà été fait pour les "layered graphs" [41].

Enfin, nous proposons un calcul des tableaux pour cette nouvelle formulation. Ce calcul, bien que s'inspirant de celui pour BBI, est complètement nouveau dans le sens où il n'utilise pas de labels mais emploie de la même manière les nominaux de la syntaxe, qui caractérisent chacun une ressource. On obtient donc un calcul similaire à celui de BBI mais sans labels. En outre, en traduisant les axiomes en règles de tableaux, on obtient alors un calcul général qui permet de raisonner sur des variantes de BBI sans redéfinir de cadre sémantique, simplement en interdisant ou en autorisant certaines règles.

## 5.1 Vers une logique BI hybride

Nous avons évoqué dans le Chapitre 3 les logiques modales comme la logique S4 qui permettent, entre autre, de raisonner sur le temps.

Toutefois, les logiques modales ne sont pas aussi expressives que, par exemple, la logique du premier ordre [15]. La raison en est la nature des quantificateurs utilisés comme modalités. En effet, les opérateurs modaux que nous avons notés $\Diamond$ et $\Box$ sont des quantificateurs respectivement existentiel et universel. Ils permettent donc d'exprimer des propositions sur un monde possible, ou sur tous les mondes possibles, mais pas sur un monde particulier.

Par exemple, dans un cadre temporel, on peut exprimer qu'il fera beau un jour ($\Diamond Beau$) ou qu'il fera beau tous les jours ($\Box Beau$), mais pas qu'il fera beau un jour en particulier (disons mercredi prochain). Même si certaines logiques temporelles permettent plus de précision avec l'opérateur $\bigcirc$ désignant un état suivant (et qui permettrait par exemple d'exprimer "il fera beau demain" [65]), on est loin de pouvoir exprimer des propriétés sur des instants particuliers.

C'est donc dans cette optique d'expressivité ciblée qu'ont été développées les logiques hybrides. Prior en proposa le concept dans des travaux des années soixante [78,79]. Une formalisation plus complète a été proposée par Blackburn [14–16,89]. L'idée en est simple. Pour chaque monde modal possible, on introduit une variable propositionnelle qui n'est validée que par ce monde. Par exemple, pour tout temps $t$ on introduit $P(t)$ qui n'est valide qu'en $t$. On peut alors, par exemple, exprimer $P(mercredi) \wedge Beau$ qui nous indiquera qu'il fait beau mercredi.

On se rapproche ainsi d'une logique du premier ordre, sans toutefois en avoir l'expressivité complète. Les logiques hybrides sont donc un juste milieu entre les logiques modales et les logiques du premier ordre, avec une lisibilité proche des premières, mais un pouvoir d'expression plus proche des secondes [5].

Concrètement, cela est effectué en rajoutant un ensemble de symboles appelés *nominaux*, habituellement $i, j, k, \ldots$ qui jouent un rôle similaire aux propositions atomiques, mais qui ne sont valides que dans un monde monde particulier (c'est ainsi que $i$ peut par exemple modéliser notre "mercredi" de l'exemple précédent, on utilisera alors la formule $i \wedge Beau$ pour dire qu'il fait beau mercredi). Un autre opérateur usuel dans les logiques hybrides est un opérateur de saut qui force une formule à être valide dans un monde (noté par un nominal). Une notation courante pour cet opérateur est le symbole @. Reprenant notre exemple précédent, on pourra de manière plus simple noter $@_i(Beau)$ pour dire qu'il fait beau mercredi.

Cette extension, qui semble purement syntaxique, a un véritable impact sémantique (comme nous venons de l'expliquer) mais aussi calculatoire. Plusieurs calculs ont été proposés pour les logiques hybrides, notamment un calcul des séquents [89] et plusieurs calculs des tableaux [15,17]. Il est intéressant de noter que le calcul des séquents utilise une forme de labels pour marquer les mondes modaux désignés par les nominaux hybrides. Quant aux tableaux, deux formes sont proposées, avec ou sans labels [17] : en effet, si l'on peut utiliser des labels afin de marquer les mondes modaux comme traditionnellement, on peut également utiliser dans ce but les nominaux fournis par la dimension hybride. Ce développement des logiques hybrides (on en trouve également une variante intuitionniste [18]) montre bien comment l'utilisation de ce type d'internalisation apporte un véritable intérêt, à la fois en terme de modélisation (on étend l'expressivité de la logique) et en terme calculatoire (puisque la logique hybride a une complexité moindre que le premier ordre qui pourrait la remplacer [5]).

L'hybridation de logiques modales a déjà introduit dans les logiques sous-structurelles. La *Logique Linéaire Hybride* ou HyLL [38], ajoute des opérateurs hybrides modaux à LL de la même manière que les modalités S4 étendaient BI dans DBI. La *Logique BBI Hybride* ou HyBBI [20] propose un mécanisme plus intéressant, puisqu'elle utilise les mécanismes des logiques hybrides et les incorporent à BBI, utilisant les nominaux pour décrire des ressources à

la place des mondes modaux. Ainsi, $@_i(\phi)$ indique que la formule $\phi$ est validée par la ressource désignée par le nominal $i$.

Le premier avantage de l'internalisation est celui que nous avons déjà mentionné pour les premières logiques hybrides, celui d'étendre le pouvoir d'expression. En effet, BBI souffre du même problème que les logiques modales. Lorsqu'on énonce par exemple que $r \vDash A * B$, on dit qu'il existe $r_1, r_2$ tels que $r = r_1 \bullet r_2$ et $r_1 \vDash A$ et $r_2 \vDash B$. Mais comme $r_1$ et $r_2$ sont introduits existentiellement, on ne peut rien énoncer sur ces deux ressources.

Si l'on veut énoncer que notre espace de travail se sépare en deux espaces, puis énoncer des propriétés différentes de ces deux espaces, la décomposition de $*$ ne nous permet pas de capturer l'un des deux sous-espaces en particulier. Par exemple, si on énonce $r \vDash (A * B) \wedge (C * D)$, on ne peut garantir que $A$ et $C$ sont dépendants de la même ressource, car la décomposition de $A * B$ d'une part et $C * D$ d'autre part peut générer deux paires de sous-ressources différentes. Bien sûr, on peut résoudre ce problème en regroupant dans une même sous-formule toutes les propositions traitant d'une même ressource, par exemple $(A \wedge C) * (B \wedge D)$. Toutefois, cet exercice peut être lourd et contre-intuitif dans le cadre d'une modélisation.

Il semble donc pertinent d'ajouter des opérateurs permettant de cerner des ressources précises et de les réutiliser autre part. Par exemple, considérons la formule suivante :

$$i * j \wedge @_i(A) \wedge @_j(B) \wedge @_i(C) \wedge @_i(D)$$

Cette formule traduit l'idée que nous voulions exprimer dans l'illustration précédente, à savoir que notre ressource courante se décompose en deux ressources correspondant ici aux nominaux $i$ et $j$, avec la première qui valide $A$ et $C$ tandis que la deuxième valide $B$ et $D$. On a ainsi pu représenter facilement les propriétés de nos ressources sans avoir besoin de les grouper en amont par ressource. En outre, cette formulation de nos propriétés est dans une mise en forme très proche, par exemple, d'un langage de programmation où chaque espace mémoire est affecté à une variable : les nominaux exprimés font directement apparaître le nom des ressources utilisées (codant par exemple des espaces mémoires) et l'opérateur @ exprime la propriété de chaque ressource.

Outre cette extension d'expressivité, l'internalisation de ressources nous ouvre la porte à une autre innovation intéressante. En effet, le fait d'avoir des ressources (ou leurs représentants directs) dans la syntaxe nous autorise à énoncer, par des formules du langage, des propriétés sur les ressources.

Cette opportunité permet de changer la manière de formaliser les logiques de ressources. On peut en effet formuler une propriété du cadre formel d'une logique (par exemple, le fait que la composition entre ressources soit commutative) par un schéma d'axiome. L'idée est alors d'appauvrir au maximum le formalisme sémantique des logiques de ressources, et de remplacer chacune des propriétés par un axiome.

Si ce changement parait, à première vue, purement cosmétique, il revêt en réalité un intérêt plus profond. Le fait d'avoir un contrôle des propriétés de notre logique par des axiomes nous permet de facilement accéder à des variantes de la logique BBI simplement en supprimant ou rajoutant des axiomes. Par exemple, si l'on veut travailler sur une variante de BBI qui n'est pas commutative, il suffit de ne pas considérer l'axiome apportant la commutativité, sans avoir besoin de redéfinir une nouvelle logique avec toutes les structures nécessaires.

La Logique BBI Hybride HyBBI [20] réalise une telle hybridation de BBI et en propose

un calcul à la Hilbert. On définit ici une logique hybride similaire mais avec quatre nouvelles perspectives :

1. La sémantique que nous utilisons est légèrement différente puisque nous utilisons *directement* les ressources dans la syntaxe, au lieu de relier nominaux et ressources par une interprétation.
2. Nous commençons par définir une logique où la composition entre ressources ne possède *aucune propriété*, pour ensuite reconstruire BBI par des axiomes. Cette particularité permet aussi à notre logique d'explorer des variantes de BBI comme explicité ci-dessus.
3. Nous proposons pour cette logique sans propriétés un calcul des tableaux mais *sans labels* qui peut être étendu par des règles supplémentaires pour travailler pour toute extension axiomatique (telles que présentées dans le point précédent).
4. Enfin, nous voulons explorer d'avantage les capacités de modélisation de ces logiques hybrides.

La logique de base que nous définissons ainsi est dénommée HRL (pour Hybrid Resource Logic). D'un point de vue formel, la mise en place de cette logique est assez simple. La syntaxe est identique à celle de BI, avec l'ajout des deux nouveaux opérateurs mentionnés hérités des logiques hybride : les nominaux et l'opérateur $@$. Nous ajoutons aussi un opérateur complémentaire à $-\!\!*$ pour compenser le fait que HRL ne comprend pas de commutativité des ressources (il faut donc composer à droite et à gauche dans la sémantique de $-\!\!*$). Cet opérateur complémentaire sera noté $*\!\!-$.

D'un point de vue sémantique, nous utilisons des structures similaires à celle de BBI, en prenant bien soin de ne donner aucune propriété à la composition $\bullet$. Pour simplifier les choses d'avantage, nous imposerons de choisir les symboles de nominaux comme ressources, au lieu de définir un isomorphisme entre nominaux et ressources qui alourdirait grandement la structure [7].

Ensuite, nous formalisons une restriction de HRL par trois axiomes exprimant la commutativité, l'associativité et la présence d'un élément neutre pour la composition entre ressources. La version de HRL munie de ces axiomes est appelée HBBI (pour Hybrid Boolean BI). Nous montrons qu'elle est équivalente sémantiquement à BBI.

Nous proposons aussi un calcul des tableaux pour HRL qui a la particularité de ne pas utiliser de labels (puisque les nominaux peuvent remplir un rôle similaire). Un point intéressant est que ce calcul pour HRL peut être étendu facilement en un calcul pour HBBI en rajoutant des règles qui correspondent aux axiomes de HBBI. Ce calcul étendu peut donc être utilisé pour raisonner sur la logique BBI. Cette pratique (rajouter une règle correspondant à un axiome) peut être généralisée, le même calcul peut alors être utilisé pour raisonner sur plusieurs variantes de HRL (par exemple, une version non commutative de BBI).

Enfin, quelques mots sur le choix de la sémantique classique par rapport à l'intuitionisme. Nous avons fait le choix de nous centrer sur BBI pour deux raisons : d'abord, comme nous l'avons illustré dans le Chapitre 2, BBI est plus aisée à manipuler pour modéliser des systèmes, il semble donc logique de l'employer dans cette logique, puisque la facilitation de la modélisation est l'un de nos objectifs. En outre, la présence d'une équivalence à la place d'une relation d'ordre permet de générer des relations très intéressantes entre nominaux avec des formules très simples (voir par exemple le Lemme 5.1).

7. Cette contrainte n'en est pas vraiment une étant donné que l'ensemble de nominaux peut être redéfini à loisir.

Toutefois, on peut à l'avenir imaginer des travaux sur une version intuitionniste de HRL, fondée sur BI plutôt que sur BBI. Il faudrait simplement adapter la sémantique des opérateurs hybrides. $r \vDash i$ signifierait alors $r \sqsubseteq i$ plutôt que $r \sim i$. Il serait possible de distinguer les versions intuitionnistes et classiques des opérateurs en utilisant des symboles différents. Une perspective possible serait alors l'expression des logiques de ressources intuitionnistes et classiques dans un même formalisme. BI et BBI seraient alors deux instanciations particulières d'une logique générique (de la même manière que BBI est, sous la forme de HBBI, une instanciation de HRL).

## 5.2 La logique HRL

### 5.2.1 Syntaxe de la logique HRL

Nous allons maintenant définir la syntaxe de notre nouvelle logique. Sans surprise, cette syntaxe sera très proche de celle de BI. La différence réside dans l'ajout de deux nouveaux opérateurs et d'une nouvelle classe de symboles atomiques. Ces symboles, que nous appelons *nominaux* sont ceux qui nous serviront à forcer une ressource à valider une formule. L'opérateur $@()$ aura une fonctionnalité similaire mais nous permettra des facilités d'écriture.

Enfin, nous sommes contraints d'introduire un nouvel opérateur $*\!\!-$. C'est une conséquence du fait que notre sémantique de HRL ne comprendra pas la commutativité de la composition entre ressources, il nous faut considérer deux implications multiplicatives différentes, l'une où la ressource forçant le prémisse est composée à droite de celle forçant l'implication, l'autre où elle est composée à gauche.

On propose donc une logique de ressources minimaliste, dont la sémantique sera classique, et qui intègre des labels internalisés, les nominaux. Nous dénommons cette logique HRL pour Hybrid Resource Logic (Logique de Ressources Hybride).

**Définition 5.1** (Syntaxe de HRL)**.** *Soit Prop un ensemble de symboles propositionnels et soit Nom un ensemble non vide et dénombrable de symbole nominaux. Le langage $\mathcal{L}$ de la logique HRL est donnée par la grammaire suivante pour tout $p \in Prop$ et pour tout $i \in Nom$ :*

$$X ::= p \mid \top \mid \bot \mid \neg X \mid X \wedge X \mid X \vee X \mid X \to X \mid \mathrm{I} \mid X * X \mid X \mathrel{-\!\!*} X \mid X \mathrel{*\!\!-} X \mid i \mid @_i(X)$$

Dans l'intégralité de ce chapitre, $\mathcal{L}$ désignera le langage de HRL sauf indication contraire. Notez qu'on inclut à nouveau la négation dans la syntaxe originale, comme pour le chapitre précédent.

L'usage d'un nominal $i$ va forcer la formule courante à être valide pour une ressource précise représentée par $i$. Dans le même esprit, $@_i(X)$ force la formule dans le champ de l'opérateur à être valide à la ressource associée à $i$.

Formalisons maintenant la sémantique de la logique HRL.

### 5.2.2 Sémantique de la logique HRL

Comme nous l'avons déjà annoncé, nous voulons que notre logique porte le minimum d'information dans sa structure, afin de pouvoir reconstruire par des axiomes des logiques plus fortes ou plus faibles que BBI.

En outre, on souhaite que les nominaux jouent le rôle de marqueur pour les ressources. Aussi, plutôt que de définir un ensemble de ressources comme dans les approches précédentes, nous allons utiliser les nominaux en leur ajoutant une structure.

**Définition 5.2** (Structure de ressources faible)**.** *Une* structure de ressources faible *associée à Nom est un triplet* $\mathcal{R} = (\bullet, e, \sim)$ *tel que :*

- *e est un élément de Nom.*
- $\bullet : Nom \times Nom \rightharpoonup Nom$ *est une fonction partielle sur Nom.*
- $\sim$ *est une relation d'équivalence sur Nom compatible (à droite et à gauche) avec* $\bullet$*, c'est à dire que si* $r \bullet s \downarrow$ *(resp.* $s \bullet r \downarrow$*) et que* $r \sim r'$*, alors* $r' \bullet s \downarrow$ *(resp.* $s \bullet r' \downarrow$*) et* $r \bullet s \sim r' \bullet s$ *(resp.* $s \bullet r \sim s \bullet r'$*).*

On appelle *e* la ressource unité et $\bullet$ la composition de ressources. Toutefois, contrairement à précédemment, aucune propriété n'est donnée sur ces deux éléments.

**Définition 5.3** (Interprétation)**.** *Soit* $\mathcal{R} = (\bullet, e, \sqsubseteq)$ *une structure de ressources faible pour Nom. Une* interprétation *de Prop pour* $\mathcal{R}$ *est une fonction* $[\![\cdot]\!] : Prop \to \mathbb{P}(Nom)$ *qui est monotone sur Prop, c'est à dire que pour tout* $p \in Prop$*, pour tout* $r, r' \in Nom$*, si* $r \sim r'$ *et* $r \in [\![p]\!]$ *alors* $r' \in [\![p]\!]$*.*

On peut maintenant définir des modèles pour donner une sémantique à notre logique

**Définition 5.4** (Modèle)**.** *Un* modèle *de HRL est un triplet* $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *où* $\mathcal{R} = (\bullet, e, \sim)$ *est une structure de ressources faible pour Nom,* $[\![\cdot]\!]$ *est une interprétation de Prop pour* $\mathcal{R}$ *et* $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times Nom$ *une relation de forcing, définie inductivement par les relations suivantes pour tout* $p \in Prop, i \in Nom, \phi, \psi \in \mathcal{L}$ *:*

- $r \vDash_{\mathcal{K}} p$ *ssi* $r \in [\![p]\!]$*.*
- $r \vDash_{\mathcal{K}} \mathrm{I}$ *ssi* $e \sim r$*.*
- $r \vDash_{\mathcal{K}} \top$ *toujours.*
- $r \vDash_{\mathcal{K}} \bot$ *jamais.*
- $r \vDash_{\mathcal{K}} \neg\phi$ *ssi on a pas* $r \vDash_{\mathcal{K}} .\phi$*.*
- $r \vDash_{\mathcal{K}} \phi \wedge \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *et* $r \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} \phi \vee \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *ou* $r \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} \phi \to \psi$ *ssi* $r \vDash_{\mathcal{K}} \phi$ *implique* $r \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} \phi * \psi$ *ssi il existe* $r', r'' \in Nom$ *tels que* $r' \bullet r'' \downarrow$ *et* $r' \bullet r'' \sim r$ *et* $r' \vDash_{\mathcal{K}} \phi$ *et* $r'' \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ *ssi pour tout* $r' \in Nom$ *tel que* $r \bullet r' \downarrow$ *et* $r' \vDash_{\mathcal{K}} \phi$*, on a* $r \bullet r' \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} \phi \mathrel{*\!\!-} \psi$ *ssi pour tout* $r' \in Nom$ *tel que* $r' \bullet r \downarrow$ *et* $r' \vDash_{\mathcal{K}} \phi$*, on a* $r' \bullet r \vDash_{\mathcal{K}} \psi$*.*
- $r \vDash_{\mathcal{K}} i$ *ssi* $r \sim i$*.*
- $r \vDash_{\mathcal{K}} @_i(\phi)$ *ssi* $i \vDash_{\mathcal{K}} \phi$*.*

Une remarque intéressante est que, selon cette définition $e$ et I sont équivalents ($r \vDash_{\mathcal{K}} e$ ssi $r \vDash_{\mathcal{K}} \mathrm{I}$). Cela est logique puisque le rôle de I est justement de marquer une formule validée par $e$. Si les deux écritures sont équivalentes, on préfèrera utiliser I pour éviter des confusions, en particulier si l'on travaille dans le cadre de la version hybride de BBI (voir plus loin).

On définit comme pour les logiques précédentes la notion de validité, de conséquence logique et d'équivalence, et on donne le théorème qui permet de prouver une conséquence.

**Définition 5.5** (Validité)**.** *On dit qu'une formule $\phi$ de $\mathcal{L}$ est valide dans HRL et on note $\vDash_{HRL} \phi$ lorsque $r \vDash_{\mathcal{K}} \phi$ pour toute ressource $r$ dans tout modèle $\mathcal{K}$.*

*On dit en outre que $\psi$ est une conséquence logique de $\phi$ dans HRL et on note $\phi \vDash_{HRL} \psi$ lorsque $r \vDash_{\mathcal{K}} \phi$ implique $r \vDash_{\mathcal{K}} \psi$ pour toute ressource $r$ de tout modèle $\mathcal{K}$.*

*On dit que deux formules $\psi$ et $\phi$ sont équivalentes dans HRL et on note $\phi \equiv_{HRL} \psi$ lorsqu'on a $\phi \vDash_{HRL} \psi$ et $\psi \vDash_{HRL} \phi$.*

Lorsque le contexte n'est pas ambigü, on omettra les indices et on notera $\vDash \phi$, $\phi \vDash \psi$ ou $\phi \equiv \psi$.

**Théorème 5.1** (Conséquence logique)**.** *Soit deux formules $\phi$ et $\psi$. On a $\phi \vDash \psi$ ssi $\vDash \phi \to \psi$.*

La preuve de ce théorème est identique à celle du théorème similaire du Chapitre 4 (Théorème 4.1).

Un des intérêts de HRL est d'axiomatiser les propriétés des ressources de nos modèles. Pour cela, on montre les résultats suivants :

**Lemme 5.1.** *Soit $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ un modèle, avec $\mathcal{R} = (\bullet, e, \sim)$, soient $r, i, j, k$ des nominaux. On a :*

| | | |
|---|---|---|
| $r \vDash_{\mathcal{K}} i * j$ | *ssi* | $r \sim i \bullet j$. |
| $r \vDash_{\mathcal{K}} @_i(j)$ | *ssi* | $i \sim j$. |
| $r \vDash_{\mathcal{K}} @_i(\neg j)$ | *ssi* | $i \not\sim j$ *(c'est à dire qu'on a pas* $i \sim j$*).* |
| $r \vDash_{\mathcal{K}} @_i(I)$ | *ssi* | $i \sim e$. |
| $r \vDash_{\mathcal{K}} @_i(j \wedge k)$ | *ssi* | $i \sim j \sim k$. |
| $r \vDash_{\mathcal{K}} @_i(j \vee k)$ | *ssi* | $i \sim j$ *ou* $i \sim k$. |
| $r \vDash_{\mathcal{K}} @_i(j \to k)$ | *ssi* | *si* $i \sim j$ *alors* $i \sim k$. |
| $r \vDash_{\mathcal{K}} @_i(j * k)$ | *ssi* | $i \sim j \bullet k$. |
| $r \vDash_{\mathcal{K}} @_i(j \mathrel{-\!\!*} k)$ | *ssi* | *si* $i \bullet j \downarrow$, *alors* $i \bullet j \sim k$. |
| $r \vDash_{\mathcal{K}} @_i(j \mathrel{*\!\!-} k)$ | *ssi* | *si* $j \bullet i \downarrow$, *alors* $j \bullet i \sim k$. |

*Démonstration.*
On utilise la définition de $\vDash_{\mathcal{K}}$ dans la Définition 5.4 :

- $r \vDash_{\mathcal{K}} j * k$ ssi il existe $r', r'' \in Nom$ tels que $r' \bullet r'' \downarrow$ et $r \sim r' \bullet r''$ et $r' \vDash_{\mathcal{K}} j$ et $r'' \vDash_{\mathcal{K}} k$ ssi il existe $r', r'' \in Nom$ tels que $r' \bullet r'' \downarrow$ et $r \sim r' \bullet r''$ et $r' \sim j$ et $r'' \sim k$ ssi $r \sim j \bullet k$ (par compatibilité de $\sim$ et $\bullet$).
- $r \vDash_{\mathcal{K}} @_i(j)$ ssi $i \vDash_{\mathcal{K}} j$ ssi $i \sim j$.
- $r \vDash_{\mathcal{K}} @_i(\neg j)$ ssi $i \vDash_{\mathcal{K}} \neg j$ ssi $i \nvDash_{\mathcal{K}} j$ ssi $i \not\sim j$.
- $r \vDash_{\mathcal{K}} @_i(I)$ ssi $i \vDash_{\mathcal{K}} \mathrm{I}$ ssi $i \sim e$.

- $r \vDash_{\mathcal{K}} @_i(j \wedge k)$ ssi $i \vDash_{\mathcal{K}} j \wedge k$ ssi $i \vDash_{\mathcal{K}} j$ et $i \vDash_{\mathcal{K}} k$ ssi $i \sim j$ et $i \sim k$ ssi $i \sim j \sim k$.
- $r \vDash_{\mathcal{K}} @_i(j \vee k)$ ssi $i \vDash_{\mathcal{K}} j \vee k$ ssi $i \vDash_{\mathcal{K}} j$ ou $i \vDash_{\mathcal{K}} k$ ssi $i \sim j$ ou $i \sim k$.
- $r \vDash_{\mathcal{K}} @_i(j \to k)$ ssi $i \vDash_{\mathcal{K}} j \to k$ ssi si $i \vDash_{\mathcal{K}} j$ alors $i \vDash_{\mathcal{K}} k$ ssi si $i \sim j$ alors $i \sim k$.
- $r \vDash_{\mathcal{K}} @_i(j * k)$ ssi $i \vDash_{\mathcal{K}} j * k$ ssi $i \sim j \bullet k$ par le premier point de ce lemme.
- $r \vDash_{\mathcal{K}} @_i(j \mathrel{-\!\!*} k)$ ssi $i \vDash_{\mathcal{K}} j \mathrel{-\!\!*} k$ ssi pour tout $r' \in Nom$ tel que $i \bullet r' \downarrow$ et $r' \vDash_{\mathcal{K}} j$ on a $i \bullet r' \vDash_{\mathcal{K}} k$ ssi si $i \bullet j \downarrow$, alors $i \bullet j \vDash_{\mathcal{K}} k$ ssi si $i \bullet j \downarrow$, alors $i \bullet j \sim k$.
- $r \vDash_{\mathcal{K}} @_i(j \mathrel{*\!\!-} k)$ ssi $i \vDash_{\mathcal{K}} j \mathrel{*\!\!-} k$ ssi pour tout $r' \in Nom$ tel que $r' \bullet i \downarrow$ et $r' \vDash_{\mathcal{K}} j$ on a $r' \bullet i \vDash_{\mathcal{K}} k$ ssi si $j \bullet i \downarrow$, alors $j \bullet i \vDash_{\mathcal{K}} k$ ssi si $j \bullet i \downarrow$, alors $j \bullet i \sim k$.

□

Enfin, nous allons énoncer deux lemmes utiles pour le calcul des tableaux qui sera développée plus loin. Ces lemme autorisent de remplacer un nominal par un nominal équivalent ou par la conjonction multiplicative de deux autres qu'il valide.

Jusqu'à la fin de cette thèse, pour deux formule $\phi$ et $\psi$, on notera $\phi[\psi]$ pour affirmer que $\psi$ est une sous-formule de $\phi$. Pour une autre formule $\pi$, et si on a $\phi[\psi]$, on notera $\phi[\psi/\pi]$ pour désigner la formule où toutes les occurrences de $\psi$ dans $\phi$ sont remplacées par $\pi$.

**Lemme 5.2.** *Soit $\phi$ une formule de HRL. Soit $\mathcal{K}$ un modèle. Soient $i, j \in Nom$ des nominaux tels que $i \sim j$. Si $\phi[i]$, alors pour tout $r \in Nom$, $r \vDash_{\mathcal{K}} \phi[i]$ ssi $r \vDash_{\mathcal{K}} \phi[i/j]$.*

*Démonstration.*
On montre le résultat par récurrence sur la taille de $\phi$ (Définition 2.2).

- Cas de base
  Si $\phi$ est de taille 1, alors $\phi = i$ et $\phi[i/j] = j$. On a $r \vDash_{\mathcal{K}} i$ ssi $r \sim i$ ssi $r \sim j$ (par transitivité de $\sim$) ssi $r \vDash_{\mathcal{K}} j$. Le lemme est donc vrai pour toute formule de taille 1.
- Hérédité
  On suppose que la propriété est vraie pour toute formule de taille plus petite que $n$ (HR) et on démontre pour une formule de taille $n+1$ selon la nature de $\phi$. On suppose que $\phi[i]$.
  - $\phi = \neg\phi'$
    Dans ce cas là on a $\phi'[i]$ et $\phi[i/j] = \neg\phi'[i/j]$. On a donc $r \vDash_{\mathcal{K}} \phi$ ssi $r \vDash_{\mathcal{K}} \neg\phi'$ ssi $r \nvDash_{\mathcal{K}} \phi'$ ssi $r \nvDash_{\mathcal{K}} \phi'[i/j]$ par (HR), ssi $r \vDash_{\mathcal{K}} \neg\phi'[i/j]$ ssi $r \vDash_{\mathcal{K}} \phi[i/j]$.
  - $\phi = @_i(\phi')$
    On a alors $\phi[i/j] = @_j(\phi'[i/j])$. Par suite, $r \vDash_{\mathcal{K}} \phi$ ssi $r \vDash_{\mathcal{K}} @_i(\phi')$ ssi $i \vDash_{\mathcal{K}} \phi'$ ssi $i \vDash_{\mathcal{K}} \phi[i/j]$ par (HR), ssi $j \vDash_{\mathcal{K}} [i/j]$ par monotonie ssi $r \vDash_{\mathcal{K}} @_j(\phi'[i/j])$ ssi $r \vDash_{\mathcal{K}} \phi[i/j]$
  - $\phi = @_k(\phi')$ avec $k \neq i$
    On a alors $\phi[i/j] = @_k(\phi'[i/j])$. Par suite, $r \vDash_{\mathcal{K}} \phi$ ssi $r \vDash_{\mathcal{K}} @_k(\phi')$ ssi $k \vDash_{\mathcal{K}} \phi'$ ssi $k \vDash_{\mathcal{K}} \phi[i/j]$ par (HR), ssi $r \vDash_{\mathcal{K}} @_k(\phi'[i/j])$ ssi $r \vDash_{\mathcal{K}} \phi[i/j]$
  - $\phi = \psi \wedge \psi'$
    Dans ce cas là on a $\psi[i]$ ou $\psi'[i]$ (possiblement les deux) et $\phi[i/j] = \psi[i/j] \wedge \psi'[i/j]$ (possiblement l'une des deux sous-formules est inchangée). Comme $\psi$ et $\psi'$ sont de taille inférieure à $n$, on peut leur appliquer (HR). On a alors $r \vDash_{\mathcal{K}} \phi$ ssi $r \vDash_{\mathcal{K}} \psi \wedge \psi'$ ssi $r \vDash_{\mathcal{K}} \psi$ et $r \vDash_{\mathcal{K}} \psi'$ ssi $r \vDash_{\mathcal{K}} \psi[i/j]$ et $r \vDash_{\mathcal{K}} \psi'[i/j]$ par (HR) ssi $r \vDash_{\mathcal{K}} \psi[i/j] \wedge \psi'[i/j]$ ssi $r \vDash_{\mathcal{K}} \phi[i/j]$.

- Les autres cas se traitent de manière similaire.

□

**Lemme 5.3.** *Soit $\phi$ une formule de HRL. Soit $\mathcal{K}$ un modèle. Soient $i, j, k \in Nom$ des nominaux tels que $i \sim j \bullet k$. On a :*

1. *Si $\phi[j*k]$, alors pour tout $r \in Nom$, $r \vDash_{\mathcal{K}} \phi[j*k]$ ssi $r \vDash_{\mathcal{K}} \phi[j*k/i]$.*
2. *Si $\phi[i]$, alors pour tout $r \in Nom$, $r \vDash_{\mathcal{K}} \phi[i]$, ssi $r \vDash_{\mathcal{K}} \phi[i/j*k]$, où les occurrences de i dans l'opérateur $@_i$ ne sont pas prises en compte.*

*Démonstration.*
On montre les deux résultats par récurrence sur la taille de $\phi$.

1. - Cas de base
   Si $\phi$ est de taille 1, le résultat ne peut s'appliquer car on ne peut avoir $\phi[j*k]$.
   Si $\phi$ est de taille 2, alors $\phi = j*k$. On a $r \vDash_{\mathcal{K}} j*k$ ssi $r \sim j \bullet k$ (d'après le Lemme 5.1) ssi $i \sim r$ (par transitivité de $\sim$) ssi $r \vDash_{\mathcal{K}} i$. Le lemme est donc vrai pour toute formule de taille 2.
   - Hérédité
   On suppose que la propriété est vraie pour toute formule de taille plus petite que $n$ et on démontre pour une formule de taille $n+1$ selon la nature de $\phi$. Le contenu de la preuve est similaire à la preuve du Lemme 5.2.
2. - Cas de base
   Si $\phi$ est de taille 1, alors $\phi = i$. On a $r \vDash_{\mathcal{K}} i$ ssi $r \sim i$ ssi $r \sim j \bullet k$ (par transitivité de $\sim$) ssi $r \vDash_{\mathcal{K}} j*k$ (d'après le Lemme 5.1). Le lemme est donc vrai pour toute formule de taille 1.
   - Hérédité
   On suppose que la propriété est vraie pour toute formule de taille plus petite que $n$ et on démontre pour une formule de taille $n+1$ selon la nature de $\phi$. Le contenu de la preuve est similaire à la preuve du Lemme 5.2..

□

Nous allons maintenant voir comment l'ajout d'axiomes nous permet d'obtenir une logique dont la sémantique est équivalente à BBI.

## 5.3 La logique BBI hybride (HBBI)

Comme nous l'avons exposé, l'un des objectifs de l'internalisation est de déplacer les spécificités structurelles de la composition des ressources depuis le cadre sémantique (structure de monoïde dans les modèles) vers le cadre syntaxique (série d'axiomes).

Nous proposons dans cette section une série d'axiomes qui permettent d'obtenir une restriction de HRL dont la sémantique est équivalente à celle de BBI.

On définit les formules suivantes :

$$(BI)_n \equiv @_i(i * \mathrm{I})$$
$$(BI)_c \equiv j * k \to k * j$$
$$(BI)_a \equiv j * (k * l) \to (j * k) * l$$

On considère l'ensemble $(BI)_{AX} = \{(BI)_n, (BI)_c, (BI)_a\}$

Ces formules, utilisées comme schémas d'axiomes, définissent une nouvelle logique que l'on dénote HBBI.

**Définition 5.6** (HBBI)**.** *La logique HBBI (pour Hybrid BBI) est définie comme la logique HRL mais admet l'ensemble de propositions $(BI)_{AX}$ comme axiomes, c'est à dire que pour tout $i, j, k, l \in Nom$, les formules de $(BI)_{AX}$ sont valides.*

La validité, la conséquence et l'équivalence logique dans HBBI sont définies comme dans la Définition 5.5 et notée $\vDash_{HBBI}$ et $\equiv_{HBBI}$ lorsque le contexte est ambigu.

On souhaite maintenant montrer que HBBI et BBI sont sémantiquement équivalentes, pour les opérateurs qu'elles partagent. On montre cela en prouvant que les axiomes de HBBI apportent la structure nécessaire à la sémantique de BBI.

**Lemme 5.4.** *Soit $\mathcal{R} = (\bullet, e, \sim)$ une structure de ressources faible associée à Nom de HBBI (selon la Définition 5.2). Alors $(Nom, e, \bullet, \sim)$ est un monoïde de ressources partiel (selon la Définition 2.14).*

*Démonstration.*
On a déjà par la Définition 5.2 que $\sim$ est une relation d'équivalence sur *Nom* et que $\sim$ est compatible avec $\bullet$. Il reste à montrer que $(Nom, e, \bullet)$ est un monoïde partiel commutatif. Par définition, *Nom* est un ensemble, $e \in Nom$ et $\bullet$ est une fonction partielle sur *Nom*. On doit montrer que $\bullet$ est associative, commutative et admet $e$ comme élément neutre.

1. Soit $r \in Nom$. Comme on est dans HBBI, l'axiome $(BI)_n$ est valide pour $r$. Soit $\mathcal{K}$ un modèle de HBBI et $r' \in Nom$. On a $r' \vDash_{\mathcal{K}} (BI)_n$ pour $r$, donc $r' \vDash_{\mathcal{K}} @_r(r * \mathrm{I})$, c'est à dire $r \vDash_{\mathcal{K}} r * \mathrm{I}$. Il existe donc $r_1, r_2 \in Nom$ tels que $r_1 \bullet r_2 \downarrow$ et $r \sim r_1 \bullet r_2$ et $r_1 \vDash_{\mathcal{K}} r$ et $r_2 \vDash_{\mathcal{K}} \mathrm{I}$. On en déduit que $r_1 \sim r$ et $r_2 \sim e$ par définition de la relation de forcing. Donc on a $r \bullet e \downarrow$ et $r \sim r \bullet e$ (par compatibilité de $\sim$ et $\bullet$). Comme ce raisonnement est valable pour tout $r \in Nom$, on a que $e$ est élément neutre de $\bullet$ (pour le moment, $e$ est uniquement neutre à droite, mais on va montrer que $\bullet$ est en outre commutative).
2. Soit $r_1, r_2 \in Nom$ tels que $r_1 \bullet r_2 \downarrow$. Comme on est dans HBBI, l'axiome $(BI)_c$ est valide pour $r_1$ et $r_2$. Soit $\mathcal{K}$ un modèle de HBBI. On a en particulier $r_1 \bullet r_2 \vDash_{\mathcal{K}} (BI)_c$ pour $r_1$ et $r_2$, donc $r_1 \bullet r_2 \vDash_{\mathcal{K}} r_1 * r_2 \to r_2 * r_1$, c'est à dire que si $r_1 \bullet r_2 \vDash_{\mathcal{K}} r_1 * r_2$, alors $r_1 \bullet r_2 \vDash_{\mathcal{K}} r_2 * r_1$. Or on a $r_1 \bullet r_2 \vDash_{\mathcal{K}} r_1 * r_2$ par le premier point du Lemme 5.1. Donc on a $r_1 \bullet r_2 \vDash_{\mathcal{K}} r_2 * r_1$, c'est à dire que $r_2 \bullet r_1 \downarrow$ et $r_1 \bullet r_2 \sim r_2 \bullet r_1$ par le premier point du Lemme 5.1. Comme ce raisonnement est valable pour tout $r_1, r_2 \in Nom$, on a montré que $\bullet$ est commutative.
3. Soit $r_1, r_2, r_3 \in Nom$ tels que $r_1 \bullet (r_2 \bullet r_3) \downarrow$. Comme on est dans HBBI, l'axiome $(BI)_a$ est valide pour $r_1$, $r_2$ et $r_3$. Soit $\mathcal{K}$ un modèle de HBBI. On a en particulier $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}}$

$(BI)_c$ pour $r_1$, $r_2$ et $r_3$, donc $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}} r_1 * (r_2 * r_3) \to (r_1 * r_2) * r_3$, c'est à dire que si $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}} r_1 * (r_2 * r_3)$, alors $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}} (r_1 * r_2) * r_3$. Or on a $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}} r_1 * (r_2 * r_3)$ par le premier point du Lemme 5.1. Donc on a $r_1 \bullet (r_2 \bullet r_3) \vDash_{\mathcal{K}} (r_1 * r_2) * r_3$, c'est à dire qu'il existe $r, r' \in Nom$ tels que $r \bullet r' \downarrow$ et $r_1 \bullet (r_2 \bullet r_3) \sim r \bullet r'$ et $r \vDash_{\mathcal{K}} r_1 * r_2$ et $r' \vDash_{\mathcal{K}} r_3$. Par le premier point du Lemme 5.1, on en déduit que $r_1 \bullet r_2 \downarrow$ et $r \sim r_1 \bullet r_2$. Enfin on a $r' \sim r_3$ puis, par compatibilité de $\bullet$ et $\sim$, $(r_1 \bullet r_2) \bullet r_3 \downarrow$ et $r_1 \bullet (r_2 \bullet r_3) \sim (r_1 \bullet r_2) \bullet r_3$. Comme ce raisonnement est valable pour tout $r_1, r_2, r_3 \in Nom$, on a montré que $\bullet$ est associative.

□

On peut également, de manière bien plus directe, avoir la propriété inverse, exprimée par le lemme suivant.

**Lemme 5.5.** *Soit $\mathcal{R} = (Nom, e, \bullet, \sim)$ un monoïde de ressources partiel construit sur Nom (selon la Définition 2.14). Alors $(\bullet, e, \sim)$ est une structure faible associée à Nom (selon la Définition 5.2).*

*Démonstration.*
Selon la Définition 2.14, $e \in Nom$, $\bullet$ est une fonction partielle sur $Nom$ et $\sim$ est une relation d'équivalence sur $Nom$ compatible avec $\bullet$. On a donc directement le résultat. □

Enfin, en remarquant que $\mathcal{L}_{BBI} \subset \mathcal{L}$, on a l'équivalence sémantique souhaitée, à savoir :

**Théorème 5.2** (Équivalence sémantique de HBBI et BBI)**.** *Soit $\phi$ une formule de $\mathcal{L}_{BBI}$. Si tous les modèles de BBI sont construits sur Nom, on a $\vDash_{BBI} \phi$ ssi $\vDash_{HBBI} \phi$.*

*Démonstration.* On a montré dans les lemmes 5.5 et 5.4 que pour peu qu'ils soient construits sur $Nom$, tout monoïde partiel commutatif est une structure faible et réciproquement. Une interprétation selon la Définition 5.3 est définie identiquement qu'une interprétation de BBI (comme définie dans le Chapitre 2). Finalement, les relations de forcing pour HBBI et pour BBI sont définies à l'identique. Ainsi, tout modèle pour HBBI définit un modèle pour BBI et vice-versa. Le langage $\mathcal{L}_{BBI}$ étant aussi commun, on a donc l'équivalence sémantique. □

La condition que les modèles de BBI doivent être construits sur $Nom$ peut paraître forte, mais il ne faut pas oublier que $Nom$ peut être redéfini à loisir. On peut ainsi, si l'on veut rester formel, travailler dans BBI avec un monoïde $(R, e, \bullet, \sim)$ puis définir $Nom = R$ pour avoir le modèle équivalent dans HBBI.

On observe enfin que, vu que les axiomes de HBBI assurent la commutativité des nominaux pour $\bullet$, l'opérateur $*\!\!-$ devient superflu, puisqu'on a le résultat suivant :

**Lemme 5.6.** *Soit $\phi$ et $\psi$ deux formules de HBBI. On a $\phi *\!\!- \psi \equiv_{HBBI} \phi \mathrel{-\!\!*} \psi$.*

*Démonstration.*
Preuve directe à partir de la commutativité de $\bullet$ dans HBBI. □

## 5.4 Expressivité de HRL

Nous allons maintenant montrer ce que HRL permet d'exprimer. Comme nous l'avons expliqué plus haut, le développement de HRL a principalement été motivé par deux choses : d'un côté l'extension du pouvoir d'expression par l'ajout de nominaux, de l'autre la modularité de la logique par l'axiomatisation des règles structurelles des ressources. Nous allons illustrer tour à tour ces deux aspects.

### 5.4.1 Nominaux et expressivité

Nous souhaitons tout d'abord montrer un exemple concret de ce qu'apportent les nominaux et l'opérateur @ à l'expressivité de BBI. On reprend l'exemple que nous avions développé en premier dans le Chapitre 2. Pour rappel, il s'agit de considérer un ensemble de ressources élémentaires
$Res = \{e, p_1, p_2, c_{50}, c_{20}, c_{10}\}$ modélisant des pièces d'euros et de centimes d'euros, qu'on étend par la composition $\bullet$ en un ensemble $R$. On donne aussi un ensemble de propositions $Prop = \{Obj_{(0.30)}, Obj_{(1.70)}, Obj_{(2)}\}$ où chacun des $Obj_{(x)}$ représente la capacité d'acheter un objet à $x$ euros, et on convient intuitivement que $r \vDash \phi$ signifie que la somme d'argent représentée par $r$ permet d'effectuer exactement l'opération décrite par $\phi$. Par exemple on peut énoncer la chose suivante :

$$p_2 \vDash Obj_{(2)} \wedge (Obj_{(0.30)} * Obj_{(1.70)})$$

qui signifie qu'avec la somme représentée par une pièce de 2 euros, on peut à la fois acheter un objet à 2 euros ou bien diviser la somme en deux parties, chacune permettant d'acheter un objet à 30 centimes et un objet à 1,70 euros.

Tout d'abord, on souhaite conserver les propriétés de BI dans le cadre de cet exemple (puisque les ensembles de pièces de monnaies vérifient la commutativité, l'associativité et possèdent un élément neutre), on va donc considérer HBBI pour modéliser cette situation.

Pour modéliser notre situation avec HBBI, nous devons pouvoir utiliser les ressources de notre exemple dans nos modèles, et donc les intégrer aux nominaux. On choisit donc d'imposer $R \subseteq Nom$. Nous avons démontré avec le Théorème 5.2 que HBBI est sémantiquement équivalente à BBI. Ainsi, avec notre exemple, puisque nous avons pris soin d'insérer $R$ dans $Nom$, et pour peu qu'on construise des modèles avec $e$ comme élément neutre, on peut exprimer toute formule que l'on exprimait déjà avec BBI.

Mais HBBI n'est pas qu'une manière différente de formuler ce que nous capturions déjà avec BBI, nous disposons également de deux nouveaux moyens d'expressions : les nominaux et l'opérateur @. Comme nous avons inclus $R$ dans $Nom$, nous pouvons tout d'abord utiliser les symboles de ressource dans nos formules.

Par exemple, la formule $p_2 \wedge Obj(0.30)$ n'est pas satisfiable (on ne peut trouver une ressource à la fois équivalente à une pièce de 2 euros et à un objet coûtant 30 centimes). En revanche, la formule $(p_1 * p_1) \wedge Obj_{(2)}$ est validée, par exemple par $p_2$. Ceci nous indique qu'une pièce de 2 euros peut à la fois être décomposée en deux pièces de 1 euro et être utilisée pour acheter un objet à 2 euros.

L'introduction de nominaux nous permet ainsi d'exprimer à la fois des propriétés sur les ressources (comme le permettait BBI) et sur la structure des ressources dans le même temps. Ainsi dans notre exemple, on peut énoncer des formules portant à la fois sur les sommes d'argent et sur les objets à acheter, ce que ne permettait pas l'usage seul de BBI.

L'opérateur @ n'apporte pas d'expressivité supplémentaire si l'on continue à se contenter des ressources de $R$. En effet, une formule de type $@_x(\phi)$ n'est valide que si $x$ force $\phi$. Cela devient donc une sous-formule indépendante du contexte. Par exemple, la formule $@_{p_2}(Obj_{(2)})$ est valide alors que $@_{p_2}(Obj_{(0.30)})$ ne l'est pas, indépendamment de tout contexte. Ces deux formules sont donc équivalentes respectivement à $\top$ et $\bot$.

En revanche, @ et les nominaux peuvent être utilisés conjointement de manière beaucoup plus intéressante avec d'autre nominaux, utilisés comme des variables de ressources. Pour l'illustrer, on se propose de modéliser le rendu de monnaie.

En effet, comme nous l'avons dit, $r \vDash \phi$ signifie, dans cet exemple, que la somme représentée par $r$ permet d'effectuer *exactement* l'action $\phi$. Nous avons vu dans le Chapitre 2 comment l'usage de BI peut au contraire modéliser qu'une somme permet d'effectuer *au moins* une action. Nous pouvons recréer ceci avec BBI : par exemple, la formule $Obj_{(0.30)} * \top$ est validée par une ressource qui permet d'acheter un objet à 30 centimes, mais éventuellement est plus grande. En effet, $\top$ étant validée par n'importe quelle ressource, on a bien une formule qui est validée par n'importe quelle ressource valant au moins 30 centimes.

Toutefois, ni cette astuce, ni l'usage de BI ne permettent de quantifier ou de gérer la différence entre la ressource forçant la formule et la ressource effective nécessaire pour acheter l'objet. Dit autrement, on sait qu'on rend la monnaie, mais on ne sait pas combien.

L'utilisation de nominaux nous permet de reformuler les choses. Considérons un nominal $x \notin R$. Nous utilisons ce $x$ comme variable. On peut alors écrire, au lieu de $Obj_{(0.30)} * \top$, la formule $Obj_{(0.30)} * x$. On conserve alors le sens initial, à savoir que si $r \vDash Obj_{(0.30)} * x$, alors $val(r) \geq 0.30$ mais on a également que $val(x) = val(r) - 0.30$, et $x$ représente donc une ressource qui a exactement la valeur du rendu de la monnaie dans la transaction.

Il est alors possible d'utiliser cette valeur, par $x$, dans une autre partie de formule. Par exemple, $(Obj_{(0.30)} * x) \wedge @_x(Obj_{(1.70)})$ est satisfaite par la ressource $p_2$. En effet, une pièce de 2 euro permet d'acquérir un objet à 30 centimes, et la monnaie restante correspond exactement à la valeur d'un objet à 1,70 euros. Si l'on veut que la somme restante permette d'acquérir au moins cet objet, on peut réitérer le processus en énonçant $(Obj_{(0.30)} * x) \wedge @_x(Obj_{(1.70)} * \top)$, ou encore pour être plus expressif, on peut utiliser un nouveau nominal pour récupérer à nouveau la somme restante. Ainsi on peut finalement énoncer la propriété suivante :

$$p_2 \bullet p_1 \vDash (Obj_{(0.30)} * x) \wedge @_x(Obj_{(1.70)} * y)$$

Cette formule (qui est valide) assure qu'une pièce de 2 euros et une pièce de 1 euro permettent d'acquérir un objet à 30 centimes, que la somme restante suffit encore à acheter un objet à 1,70 euros, et que la somme restante est portée par une ressource notée $y$ (on a donc $val(y) = 1$).

On voit ici comment l'usage de nominaux et de @ permet d'écrire de manière intuitive des propriétés portant sur des ressources et des propositions à la fois, rendant ainsi les logiques de séparation beaucoup plus facile à utiliser pour la modélisation.

### 5.4.2 Extensions et restrictions pour HBBI

Un objectif de HRL était de modéliser les propriétés propres à BBI par des axiomes afin de pouvoir aisément étendre ou restreindre ces propriétés. Il est ainsi possible, avec le langage de HRL, d'exprimer non seulement la sémantique de BBI, mais aussi celle de nombreuses variantes.

Nous avons déjà montré, par le Théorème 5.2, comment l'ajout des axiomes $(BI)_c$, $(BI)_a$ et $(BI)_n$ dans HRL permettait d'obtenir exactement le pouvoir d'expression de la logique BBI. Ces trois schémas axiomes assurent respectivement la commutativité de $\bullet$ dans tout modèle, son associativité et son acceptance de $e$ comme élément neutre. Il est aisé de voir comment l'omission d'un de ces trois axiomes génère une logique où la propriété correspondante manque.

De telles variations de BI sont déjà utilisées. L'exemple le plus marquant est l'omission de la commutativité des ressources. Les logiques de séparation non commutatives sont un outil de modélisation courant. On peut citer l'utilisation d'une telle logique pour modéliser des "layered graphs", notamment dans [28,29,41].

Avec l'ajout des axiomes $(BI)_a$ et $(BI)_n$ à HRL , les opérateurs $*$ , $-\!\!*$ et $*\!\!-$ seraient exactement équivalents aux opérateurs non commutatifs des layered graphs, $\blacktriangleleft$, $\blacktriangleright$ et $\multimap$ tels que décrits dans [28].

La restriction des axiomes de HBBI permet ainsi d'explorer de nombreuses variantes de BBI avec le même formalisme. En outre, comme nous l'expliciterons plus loin, il est aisé de transformer ces axiomes en règles pour notre calcul des tableaux, ce qui rend ce calcul utilisable pour toutes les variantes de BBI concernée (il suffit de s'interdire ou non certaines règles selon la logique examinée). Notre calcul des tableaux peut alors être utilisée pour explorer les différentes variantes de BBI, repérant les endroits où une règle propre à une variante en particulier est utilisée dans une preuve. Nous développerons ce point dans la Section 5.7.1.

Nous pouvons également ajouter d'avantages d'axiomes pour modéliser de nouvelles propriétés des ressources.

Considérons par exemple l'existence possible de ressources inversibles. Pour rappel, une ressource $r$ est inversible s'il existe une ressource $r'$ telle que $r \bullet r' \sim e$. Une telle propriété peut aisément être exprimée dans HRL (alors que la chose aurait été complexe avec BI).

Pour tout nominal $i$, on considère la formule suivante :

$$Inv(i) = (i * \top) \wedge \mathrm{I}$$

On suppose que cette formule est valide. Soit $\mathcal{K}$ un modèle et $r \in Nom$. Comme $Inv(i)$ est valide, on a $r \vDash_{\mathcal{K}} Inv(i)$, c'est à dire que $r \vDash_{\mathcal{K}} (i * \top) \wedge \mathrm{I}$. On a donc $r \vDash_{\mathcal{K}} i * \top$ et $r \vDash_{\mathcal{K}} \mathrm{I}$. Donc il existe $r', r''$ tels que $r'' \sim i$ et $r \sim r'' \bullet r'$ et en outre on a $r \sim e$, donc finalement par composition on a qu'il existe $r'$ tel que $i \bullet r' \sim e$.

On peut donc affirmer que la formule $Inv(i)$ signifie bien que $i$ est inversible. Si elle est érigée en axiome, elle force donc toute ressource à être inversible. On peut même aller plus loin en transformant légèrement $Inv(i)$. Si l'on établit la formule $(i * x) \wedge \mathrm{I}$, on a non seulement que $i$ est inversible mais on fait aussi émerger le nominal $x$ qui est l'inverse de $i$ et peut être réutilisé dans une autre formule.

Enfin, on peut transformer l'existence d'un inverse pour tout nominal en une règle pour notre calcul des tableaux et raisonner ainsi sur une logique où les ressources sont inversibles. Nous développerons ce point dans la Section 5.7.1.

## 5.5 Un calcul sans labels pour HRL

Nous allons maintenant présenter un calcul des tableaux pour HRL qui, bien évidemment, pourra être adapté pour le cas particulier de HBBI. Ce calcul s'inspire du calcul pour BBI [66] que nous avons déjà présenté dans le Chapitre 2. Toutefois, une différence majeure est que, contrairement au calcul sus-nommé, celui pour HRL n'utilise pas de labels et de contraintes sur les labels.

En effet, comme nous disposons de symboles internalisés qui nous permettent de représenter des ressources à l'intérieur de la syntaxe, nous pouvons mettre en forme les contraintes de labels par des formules de notre syntaxe, en utilisant les nominaux comme labels.

L'approche restera la même, avec un label et une étiquette de vérité associés à chaque formule, mais au lieu d'une présentation de la forme $\mathbb{T}\phi : x$ où $x$ est un élément appartenant à une structure externe, nous utiliserons l'opérateur $@()$ et un nominal $x$ dans une écriture $\mathbb{T}\ @_x(\phi)$. Les contraintes seront elles aussi des formules au lieu d'appartenir à un ensemble séparé.

Cette inclusion des labels dans la syntaxe permet une grande simplification formelle du calcul en éliminant les structures auxiliaires. Toutefois, cela introduit également plus de complexité dans la résolution, la quantité de règles applicables grandissant dans le même temps. On obtient ainsi un calcul qui semble plus systématique, mais aussi parfois moins lisible.

### 5.5.1 Formules et tableaux

Commençons par définir formellement les structures simplifiées utiles pour notre calcul.

**Définition 5.7** (Formule étiquetée, ensemble d'énoncés hybrides)**.** *Une* formule étiquetée *est un couple* $(\mathbb{S}, \Phi)$ *avec* $\mathbb{S} \in \{\mathbb{T}, \mathbb{F}\}$ *et* $\Phi$ *une formule de HRL de la forme* $\Phi = @_x(\phi)$ *où* $x \in Nom$ *et* $\phi \in \mathcal{L}$*. On note* $\mathbb{S}\ @_x(\phi)$ *une formule étiquetée* $(\mathbb{S}, @_x(\phi))$.

*On appelle* SHS *(pour Set of Hybrid Statements ou ensemble d'énoncés hybrides) et on note* $\mathcal{F}$ *un ensemble de formules étiquetées. L'*alphabet *d'un ensemble d'énoncés, noté* $\mathcal{A}(\mathcal{F})$ *est l'ensemble des nominaux apparaissant dans cet ensemble :*

$$\begin{aligned}\mathcal{A}(\mathcal{F}) &= \{x \in Nom / \mathbb{S}\ @_x(\phi) \in \mathcal{F}\} \\ &\cup \{x \in Nom / \mathbb{S}\ @_y(\phi[x]) \in \mathcal{F}\} \\ &\cup \{x \in Nom / \mathbb{S}\ @_y(\phi[@_x(\psi)]) \in \mathcal{F}\}\end{aligned}$$

On notera $\mathcal{F}_f \subseteq_f \mathcal{F}$ si $\mathcal{F}_f \subseteq \mathcal{F}$ et $\mathcal{F}_f$ est fini.

Notons bien que dans cette définition, le nominal $x$ d'une formule étiquetée $\mathbb{S}\ @_x(\phi)$ est un nominal non composé. Cette contrainte forcera l'addition de certaines règles pour raisonner sur des formules valides pour une ressource composée. On pourrait toute fois concevoir un autre calcul où les formules porte des nominaux composés, notamment pour simplifier l'établissement des tableaux.

La Figure 5.1 présente les règles de notre calcul des tableaux. Dans toutes ces règles, les $c_i$ et $c_j$ sont des nominaux neufs (c'est à dire que $c_i, c_j \notin \mathcal{A}(\mathcal{F})$ avant d'appliquer la règle en question). Une exception concerne la règle $\langle i+\rangle$ : dans cette règle, si un nominal $c_i$ a déjà été introduit par une autre occurrence de cette même règle, avec les même nominaux $y$ et $z$, alors on peut réutiliser le même nominal $c_i$ (c'est même nécessaire pour certaines preuves).

Dans chacune des règles $\langle @ \rangle$, $\langle i_t \rangle$, $\langle i+ \rangle$, $\langle i- \rangle$ et $\langle i_p \rangle$, le symbole $\mathbb{S}$ désigne $\mathbb{T}$ ou $\mathbb{F}$ et doit être le même dans les prémisses et les conclusions.

Pour expliciter la manière d'utiliser ces règles, nous allons donner la définition formelle d'un tableau.

**Définition 5.8** (HRL-Tableau)**.** *Soit $\mathcal{F}_0$ un SHS fini. Un HRL-tableau pour $\mathcal{F}_0$ est une liste de SHS, appelés branches, construit de manière inductive en utilisant les règles suivantes :*

- *La liste à une branche $[\![\mathcal{F}_0]\!]$ est un tableau pour $[\![\mathcal{F}_0]\!]$ ;*
- *Si la liste $\mathcal{T}_m \oplus [\![\mathcal{F}]\!] \oplus \mathcal{T}_n$ est un HRL-tableau pour $\mathcal{F}_0$ et*

$$\frac{H_1, \ldots, H_l}{C_1^1, \ldots, C_{p1}^1 \mid \ldots \mid C_1^k, \ldots, C_{pk}^k}$$

*est une règle de la Figure 5.1 tel que $H_i \in \mathcal{F}$ pour tout i, alors la liste $\mathcal{T}_m \oplus [\![\mathcal{F} \cup \{C_1^1, \ldots, C_{p1}^1\}; \ldots; \mathcal{F} \cup \{C_1^k, \ldots, C_{pk}^k\}]\!] \oplus \mathcal{T}_n$ est un HRL-tableau pour $\mathcal{F}_0$.*

*Un HRL-tableau pour une formule $\phi$ est un HRL-tableau pour le SHS $\{\mathbb{F}\ @_{c_1}(\phi)\}$ où $c_1$ est un nominal n'apparaissant pas dans $\phi$.*

De même que dans les calculs des tableaux que nous avons présentées dans les chapitres précédentes, on observe deux catégories de règles en particulier :

1. Les règles $\langle \mathbb{T}* \rangle$, $\langle \mathbb{F} -\!\!* \rangle$, $\langle \mathbb{F} *\!\!- \rangle$ et $\langle i+ \rangle$ introduisent de nouveaux nominaux dans les formules.
2. Les règles $\langle \mathbb{F}* \rangle$, $\langle \mathbb{T} -\!\!* \rangle$, $\langle \mathbb{T} *\!\!- \rangle$, $\langle i_s \rangle$, $\langle i_t \rangle$, $\langle i- \rangle$ et $\langle i_p \rangle$ nécessitent des nominaux dans leur prémisses pour être appliquées.

Il convient d'appliquer autant que possible les règles de la catégorie 1. avant celles de la catégorie 2. pour générer le plus de nominaux avant de les consommer.

On peut remarquer que des règles étiquetées $\mathbb{T}$ jouent le même rôle que les contraintes de labels (par exemple dans la règle $\langle \mathbb{T}* \rangle$), avec le raccourci que $@_x(y)$ est valide ssi $x \sim y$ (comme exprimé par le Lemme 5.1).

Les règles $\langle i_r \rangle$, $\langle i_s \rangle$ et $\langle i_t \rangle$ expriment respectivement la réflexivité, la symétrie et la transitivité de la relation d'équivalence $\sim$. La règle $\langle i_t \rangle$ est un peu plus générale puisqu'elle permet de remplacer n'importe quel nominal dans une formule par un autre nominal qui est équivalent. Le cas particulier où $\phi = z$ traduit la transitivité.

La règle $\langle i_p \rangle$ permet de substituer à nominal un autre nominal équivalent.

Enfin, les règles $\langle i+ \rangle$ et $\langle i- \rangle$ permettent d'introduire et de supprimer des nominaux composés en les remplaçant par des simples. Cette manipulation évite d'utiliser des nominaux composés dans l'opérateur $@\,()$ des formules étiquetées.

On remarque que la Figure 5.1 n'inclut pas de règles exprimant les propriétés de $\bullet$ (comme la commutativité et la transitivité). C'est tout à fait normal puisque HRL n'inclut pas de telles propriétés. Nous verrons plus loin comment l'ajout de nouvelles règles permettent de travailler avec des extensions de HRL (comme HBBI).

Nous allons maintenant énoncer les règles de clôture de nos tableaux.

8. Dans la règle $\langle i+ \rangle$, on peut prendre un $c_i$ déjà utilisé s'il a été introduit pour le même couple de nominaux $(y, z)$

$$\frac{}{\mathbb{T}\ @_e(\mathbf{I})}\ \langle \mathbb{T}\mathbf{I}\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi\wedge\psi)}{\mathbb{T}\ @_x(\phi),\ \mathbb{T}\ @_x(\psi)}\ \langle \mathbb{T}\wedge\rangle \qquad \frac{\mathbb{F}\ @_x(\phi\wedge\psi)}{\mathbb{F}\ @_x(\phi)\mid \mathbb{F}\ @_x(\psi)}\ \langle \mathbb{F}\wedge\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi\vee\psi)}{\mathbb{T}\ @_x(\phi)\mid \mathbb{T}\ @_x(\psi)}\ \langle \mathbb{T}\vee\rangle \qquad \frac{\mathbb{F}\ @_x(\phi\vee\psi)}{\mathbb{F}\ @_x(\phi),\ \mathbb{F}\ @_x(\psi)}\ \langle \mathbb{F}\vee\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi\to\psi)}{\mathbb{F}\ @_x(\phi)\mid \mathbb{T}\ @_x(\psi)}\ \langle \mathbb{T}\to\rangle \qquad \frac{\mathbb{F}\ @_x(\phi\to\psi)}{\mathbb{T}\ @_x(\phi),\ \mathbb{F}\ @_x(\psi)}\ \langle \mathbb{F}\to\rangle$$

$$\frac{\mathbb{T}\ @_x(\neg\phi)}{\mathbb{F}\ @_x(\phi)}\ \langle \mathbb{T}\neg\rangle \qquad \frac{\mathbb{F}\ @_x(\neg\phi)}{\mathbb{T}\ @_x(\phi)}\ \langle \mathbb{F}\neg\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi * \psi)}{\mathbb{T}\ @_{c_i}(\phi),\ \mathbb{T}\ @_{c_j}(\psi),\ \mathbb{T}\ @_x(c_i * c_j)}\ \langle \mathbb{T}*\rangle \qquad \frac{\mathbb{F}\ @_x(\phi * \psi), \mathbb{T}\ @_x(y * z)}{\mathbb{F}\ @_y(\phi)\mid \mathbb{F}\ @_z(\psi)}\ \langle \mathbb{F}*\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi \mathrel{-\!\!*} \psi), \mathbb{T}\ @_z(x * y)}{\mathbb{F}\ @_y(\phi)\mid \mathbb{T}\ @_z(\psi)}\ \langle \mathbb{T}\mathrel{-\!\!*}\rangle \qquad \frac{\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi)}{\mathbb{T}\ @_{c_i}(\phi),\ \mathbb{F}\ @_{c_j}(\psi),\ \mathbb{T}\ @_{c_j}(x * c_i)}\ \langle \mathbb{F}\mathrel{-\!\!*}\rangle$$

$$\frac{\mathbb{T}\ @_x(\phi *\!\!- \psi), \mathbb{T}\ @_z(y * x)}{\mathbb{F}\ @_y(\phi)\mid \mathbb{T}\ @_z(\psi)}\ \langle \mathbb{T}*\!\!-\rangle \qquad \frac{\mathbb{F}\ @_x(\phi *\!\!- \psi)}{\mathbb{T}\ @_{c_i}(\phi),\ \mathbb{F}\ @_{c_j}(\psi),\ \mathbb{T}\ @_{c_j}(c_i * x)}\ \langle \mathbb{F}*\!\!-\rangle$$

$$\frac{\mathbb{S}\ @_x(@_y(\phi))}{\mathbb{S}\ @_y(\phi)}\ \langle @\rangle \qquad \frac{}{\mathbb{T}\ @_x(x)}\ \langle i_r\rangle$$

$$\frac{\mathbb{T}\ @_x(y)}{\mathbb{T}\ @_y(x)}\ \langle i_s\rangle \qquad \frac{\mathbb{S}\ @_x(\phi), \mathbb{T}\ @_y(x)}{\mathbb{S}\ @_y(\phi)}\ \langle i_t\rangle$$

$$\frac{\mathbb{S}\ @_x(\phi[y * z])}{\mathbb{S}\ @_x(\phi[y * z/c_i]),\ \mathbb{T}\ @_{c_i}(y * z)}\ \langle i{+}\rangle \qquad \frac{\mathbb{S}\ @_x(\phi[y]), \mathbb{T}\ @_y(z * t)}{\mathbb{S}\ @_x(\phi[y/z * t])}\ \langle i{-}\rangle$$

$$\frac{\mathbb{S}\ @_x(\phi[y]), \mathbb{T}\ @_y(z)}{\mathbb{S}\ @_x(\phi[y/z])}\ \langle i_p\rangle$$

$x, y, z, t$ sont des nominaux et $c_i, c_j$ des nouveaux nominaux [8].

FIGURE 5.1 – Règle des tableaux pour HRL

**Définition 5.9** (Clôture)**.** *Un SHS $\mathcal{F}$ est* clos *si l'une des conditions suivantes est vérifiée (pour $\phi \in \mathcal{L}$ et $x \in Nom$) :*

1. $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$
2. $\mathbb{T}\ @_x(\bot) \in \mathcal{F}$
3. $\mathbb{F}\ @_x(\top) \in \mathcal{F}$

*Un SHS est* ouvert *s'il n'est pas clos. Un HRL-tableau est* clos *si toutes ses branches sont closes. Une* HRL-preuve *pour une formule $\phi$ est un HRL-tableau clos pour $\phi$.*

### 5.5.2 Un exemple de HRL-preuve

Nous présentons ici un exemple de preuve pour HRL. On se propose de dresser le tableau pour la formule suivante :

$$@_i(A) \wedge (i * B) \rightarrow A * B$$

Comme indiqué dans la Définition 5.8, nous devons développer le SHS suivant :

$$\{\mathbb{F}\ @_{c_1}(@_i(A) \wedge (i * B) \rightarrow A * B)\}$$

Le nominal $c_1$ est un nominal neuf, qui n'apparaît pas dans la formule. De même que pour les calculs des tableaux des chapitres précédents, nous représenterons l'évolution de notre tableau par un graphe plutôt que par une liste de formules. On commence donc avec le nœud suivant :

$$\mathbb{F}\ @_{c_1}(@_i(A) \wedge (i * B) \rightarrow A * B)$$

Nous n'avons qu'une seule formule dans notre SHS, étiquetée avec $\mathbb{F}$ et dont l'opérateur le plus externe est $\rightarrow$. On peut donc lui appliquer la règle $\langle \mathbb{F} \rightarrow \rangle$, ce qui produit deux nouvelles formules étiquetées avec $c_1$. On marque d'un $\surd$ la formule développée : On continue ainsi à

$$\begin{array}{c} \surd_1\ \mathbb{F}\ @_{c_1}(@_i(A) \wedge (i * B) \rightarrow A * B) \\ | \\ \mathbb{T}\ @_{c_1}(@_i(A) \wedge (i * B)) \\ \mathbb{F}\ @_{c_1}(A * B) \end{array}$$

développer les formules en prenant bien soin de respecter l'ordre de priorité (les règles générant des nominaux avant celles en consommant).

Le tableau complètement développé est exposé dans la Figure 5.2. En voici les étapes :

1. On développe l'implication et on génère deux sous-formules étiquetées.
2. On applique la règle $\langle \mathbb{T} \wedge \rangle$, ce qui produit deux nouvelles formules, toutes étiquetées par $c_1$.
3. On applique la règle $\langle @ \rangle$ qui génère la formule $\mathbb{T}\ @_i(A)$.

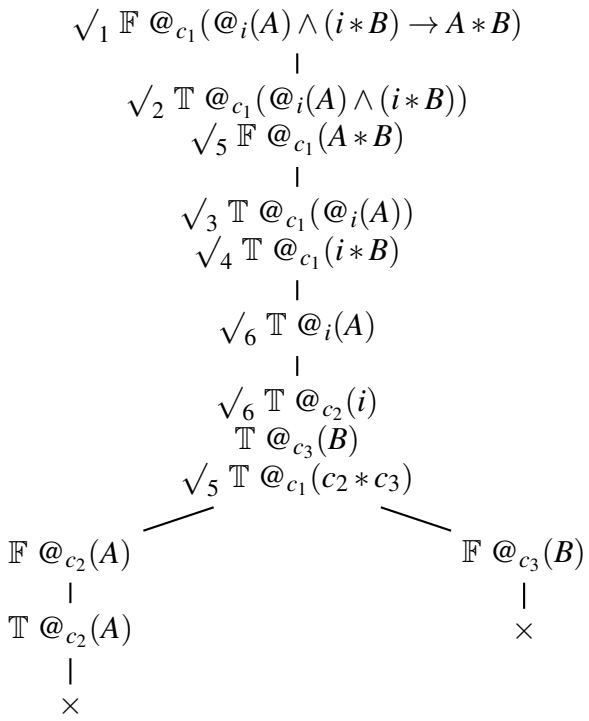



FIGURE 5.2 – HRL-tableau pour $@_i(A) \wedge (i * B) \rightarrow A * B$

4. Les formules restant à développer correspondent respectivement aux règles $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}* \rangle$. Notre ordre de priorité nous impose de commencer par $\langle \mathbb{T}* \rangle$. On génère donc deux nouveaux nominaux $c_2$ et $c_3$ ainsi que trois formules.
5. On peut enfin développer la règle $\langle \mathbb{F}* \rangle$ en utilisant la formule étiquetée $\mathbb{T}\ @_{c_1}(c_2 * c_3)$. On génère deux nouvelles branches, l'une portant $\mathbb{F}\ @_{c_2}(A)$, l'autre $\mathbb{F}\ @_{c_3}(B)$. La branche de droite ainsi créée est d'ors et déjà close (avec la formule $\mathbb{T}\ @_{c_3}(B)$).
6. L'intuition indique un rapprochement de $\mathbb{F}\ @_{c_2}(A)$ et de $\mathbb{T}\ @_i(A)$ dans la branche de gauche. On doit pour cela relier $i$ et $c_2$, ce qui est possible grâce à la formule $\mathbb{T}\ @_{c_2}(i)$ générée précédemment. La règle $\langle i_t \rangle$ convient à nos besoins, ce qui produit $\mathbb{T}\ @_{c_2}(A)$ et produit la clôture de la branche avec $\mathbb{F}\ @_{c_2}(A)$.

Toutes les branches de notre tableau sont donc closes et on a obtenu une preuve de notre formule $@_i(A) \wedge @_j(B) \wedge (i * j) \rightarrow A * B$.

## 5.6 Propriétés du calcul des HRL-tableaux

Maintenant que nous avons défini convenablement un calcul des tableaux pour raisonner sur HRL, il nous faut en étudier les propriétés. Nous allons d'abord montrer que ce calcul est correct, puis donner une technique d'extraction de contre-modèles pour des tableaux non clos, enfin utiliser cette technique pour prouver la complétude du calcul avec un schéma de preuve similaire à celui utilisé dans les chapitres précédents.

Ici, les preuves seront simplifiées par l'absence de labels et d'ensemble de contraintes : nominaux et formules ont le même statut et peuvent donc être traités similairement dans les preuves. En contrepartie, le plus grand nombre de règles étoffe les preuves par cas.

### 5.6.1 Correction du calcul des HRL-tableaux

On prouve la correction en introduisant pour les tableaux de HRL la notion de branche réalisable.

**Définition 5.10** (Réalisabilité)**.** *Soit $\mathcal{F}$ un SHS. On dit que $\mathcal{F}$ est* réalisable *s'il existe un modèle $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ tel que :*

- *si $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$, alors $x \vDash_{\mathcal{K}} \phi$*
- *si $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$, alors $x \nvDash_{\mathcal{K}} \phi$*

*On dit qu'un tableau est réalisable si au moins une de ses branches est réalisable.*

La notion de réalisabilité nous assure un lien entre les tableaux et les formules. En outre, cette notion est préservée par l'extension d'un SHS par nos règles de tableaux, comme explicité par ce lemme :

**Lemme 5.7.** *Les règles du calcul des tableaux pour HRL (Figure 5.1) préservent la réalisabilité.*

*Démonstration.*
On considère un tableau réalisable et on montre que son extension par une règle reste réalisable, en discutant selon la règle d'extension utilisée.
La preuve complète est en Annexe B. □

Une fois assuré que la réalisabilité est conservée par extension des tableaux, il nous faut statuer sur les tableaux clos.

**Lemme 5.8.** *Les branches closes ne sont pas réalisables.*

*Démonstration.*
Soit $\mathcal{F}$ une branche close. On suppose que cette branche est réalisable par un modèle $\mathcal{K}$. Il y a trois cas :

- $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. On a alors $x \vDash_{\mathcal{K}} \phi$ et $x \nvDash_{\mathcal{K}} \phi$, ce qui est absurde.
- $\mathbb{T}\ @_x(\bot) \in \mathcal{F}$. On a alors $x \vDash_{\mathcal{K}} \bot$, ce qui est absurde.
- $\mathbb{F}\ @_x(\top) \in \mathcal{F}$. On a alors $x \nvDash_{\mathcal{K}} \top$, ce qui est absurde.

Tous les cas étant absurdes, on en déduit que $\mathcal{F}$ n'est pas réalisable.

□

Enfin, on peut conclure sur la correction de notre calcul.

**Théorème 5.3** (Correction)**.** *S'il existe une preuve pour une formule $\phi$ de HRL,alors elle est valide.*

*Démonstration.*
Supposons qu'il existe une preuve pour $\phi$. Alors il y a un tableau clos $\mathcal{T}_\phi$ pour $\{\mathbb{F}\ @_{c_1}(\phi)\}$. Supposons que $\phi$ n'est pas valide. Alors il existe un contre-modèle $\mathcal{K}$ et un nominal $r \in Nom$ tel que $r \nVdash_{\mathcal{K}} \phi$. Comme $c_1$ n'apparait pas dans $\phi$, $\sim$ n'est pas défini pour $c_1$ dans $\mathcal{K}$. On peut donc définir un modèle $\mathcal{K}'$ identique à $\mathcal{K}$ mais où $c_1 \sim r$. On a alors $c_1 \nVdash_{\mathcal{K}'} \phi$ par monotonie. Alors $\mathcal{K}'$ est une réalisation de $\{\mathbb{F}\ @_{c_1}(\phi)\}$. Par le Lemme 5.7, $\mathcal{T}_\phi$ est réalisable. Alors par le Lemme 5.8, $\mathcal{T}_\phi$ ne peut être clos, ce qui est absurde car toute preuve est close. Donc $\phi$ doit être valide. □

Notre calcul des tableaux pour HRL est donc correct.

### 5.6.2 Extraction de contre-modèles

Nous allons maintenant présenter la manière d'extraire un contre-modèle d'un tableau non clos. Pour cela, nous allons d'abord caractériser une branche ouverte et saturée avec la notion de SHS de Hintikka, similaire à celle présentée dans les chapitres précédents. L'objectif est toujours de caractériser une branche non close et saturée, mais nous devons ici prendre en compte la saturation par les règles de la Figure 5.1 et les conditions de clôture de la Définition 5.9.

**Définition 5.11** (SHS de Hintikka)**.** *Un SHS $\mathcal{F}$ est appelé* SHS de Hintikka *si pour toutes formules $\phi, \psi \in \mathcal{L}$ et pour tous nominaux $x, y, z, t \in \mathcal{A}(\mathcal{F})$ on a :*

1. $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}$ *ou* $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}$
2. $\mathbb{T}\ @_x(\bot) \notin \mathcal{F}$
3. $\mathbb{F}\ @_x(\top) \notin \mathcal{F}$
4. $\mathbb{T}\ @_e(\mathrm{I}) \in \mathcal{F}$
5. *Si* $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$
6. *Si* $\mathbb{F}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ *ou* $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$
7. *Si* $\mathbb{T}\ @_x(\phi \vee \psi) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *ou* $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$
8. *Si* $\mathbb{F}\ @_x(\phi \vee \psi) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$
9. *Si* $\mathbb{T}\ @_x(\phi \to \psi) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ *ou* $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$
10. *Si* $\mathbb{F}\ @_x(\phi \to \psi) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$
11. *Si* $\mathbb{T}\ @_x(\neg\phi) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$
12. *Si* $\mathbb{F}\ @_x(\neg\phi) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$
13. *Si* $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, *alors* $\exists y, z \in \mathcal{A}(\mathcal{F})$ *tel que* $\mathbb{T}\ @_y(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(\psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$
14. *Si* $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_y(\phi) \in \mathcal{F}$ *ou* $\mathbb{F}\ @_z(\psi) \in \mathcal{F}$
15. *Si* $\mathbb{T}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(x * y) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_y(\phi) \in \mathcal{F}$ *ou* $\mathbb{T}\ @_z(\psi) \in \mathcal{F}$
16. *Si* $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$, *alors* $\exists y, z \in \mathcal{A}(\mathcal{F})$ *tels que* $\mathbb{T}\ @_y(\phi) \in \mathcal{F}$ *et* $\mathbb{F}\ @_z(\psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(x * y) \in \mathcal{F}$
17. *Si* $\mathbb{T}\ @_x(\phi \mathrel{*\!\!-} \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(y * x) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_y(\phi) \in \mathcal{F}$ *ou* $\mathbb{T}\ @_z(\psi) \in \mathcal{F}$

18. *Si* $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$, *alors* $\exists y,z \in \mathcal{A}(\mathcal{F})$ *tels que* $\mathbb{T}\ @_y(\phi) \in \mathcal{F}$ *et* $\mathbb{F}\ @_z(\psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(y * x) \in \mathcal{F}$
19. *Si* $\mathbb{T}\ @_x(@_y(\phi)) \in \mathcal{F}$ *alors* $\mathbb{T}\ @_y(\phi) \in \mathcal{F}$
20. *Si* $\mathbb{F}\ @_x(@_y(\phi)) \in \mathcal{F}$ *alors* $\mathbb{F}\ @_y(\phi) \in \mathcal{F}$
21. $\mathbb{T}\ @_x(x) \in \mathcal{F}$
22. *Si* $\mathbb{T}\ @_x(y) \in \mathcal{F}$ *alors* $\mathbb{T}\ @_y(x) \in \mathcal{F}$
23. *Si* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_y(\phi) \in \mathcal{F}$.
24. *Si* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_y(\phi) \in \mathcal{F}$.
25. *Si* $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\exists t \in \mathcal{A}(\mathcal{F})$ *tel que* $\mathbb{T}\ @_x(\phi[y * z/t]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_t(y * z)$ *pour toute substitution* $[y * z/t]$.
26. *Si* $\mathbb{F}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\exists t \in \mathcal{A}(\mathcal{F})$ *tel que* $\mathbb{F}\ @_x(\phi[y * z/t]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_t(y * z)$ *pour toute substitution* $[y * z/t]$.
27. *Si* $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z * t) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi[y/z * t]) \in \mathcal{F}$.
28. *Si* $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z * t) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi[y/z * t]) \in \mathcal{F}$.
29. *Si* $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(\phi[y/z]) \in \mathcal{F}$.
30. *Si* $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z) \in \mathcal{F}$, *alors* $\mathbb{F}\ @_x(\phi[y/z]) \in \mathcal{F}$.

Dans cette définition, les points 1 à 3 assurent qu'un SHS de Hintikka n'est pas clos tandis que les points 4 à 30 assurent qu'on ne peut plus appliquer de règles sur ce SHS (la branche est saturée).

Notons que les points 19 à 24, correspondant aux propriétés de la relation d'équivalence, ainsi que les points 25 à 30, correspondant aux substitutions d'un nominal par un nominal ou un couple de nominal, sont toutes assez fastidieuses à obtenir de manière extensive. On peut s'abstenir d'appliquer ces points puisque la définition de la fonction $\Omega$ que nous donnons ci-après assure que notre contre-modèle possède les propriétés appropriées. Nous expliciterons ce point plus loin.

**Définition 5.12** (Fonction Oméga)**.** *La fonction* $\Omega$ *est une fonction qui associe un SHS* $\mathcal{F}$ *à un triplet* $\Omega(\mathcal{F}) = \mathcal{K}$ *avec* $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ *et* $\mathcal{R} = (\bullet, e, \sim)$, *tel que :*

- *e est un élément de Nom.*
- $\sim$ *est une relation d'équivalence sur Nom définie par :*
  - *pour tout* $x, y \in \mathcal{A}(\mathcal{F})$, $x \sim y$ *ssi* $\mathbb{T}\ @_x(y) \in \mathcal{F}$ ;
  - *s'il existe* $x \in \mathcal{A}(\mathcal{F})$ *tel que* $\mathbb{T}\ @_x(\mathrm{I}) \in \mathcal{F}$, *alors* $x \sim e$.
- $\bullet$ *est une fonction partielle sur Nom définie par :*
  - *s'il existe* $x \in \mathcal{A}(\mathcal{F})$ *tel que* $\mathbb{T}\ @_x(r * r') \in \mathcal{F}$, *alors* $r \bullet r' \downarrow$ *et* $x \sim r \bullet r'$ ;
  - *sinon,* $r \bullet r' \uparrow$.
- $[\![\cdot]\!]$ *est une fonction de Prop dans* $\mathbb{P}(Nom)$ *telle que pour tout* $p \in Prop$, $[\![p]\!] = \{x \in \mathcal{A}(\mathcal{F}) / \mathbb{T}\ @_x(p) \in \mathcal{F}\}$.
- $\vDash_{\mathcal{K}} \subseteq \mathcal{L} \times Nom$ *est une relation définie inductivement de la même manière que la relation de forcing de la Définition 5.4.*

La fonction Oméga, ainsi définie, construit en réalité un modèle pourvu qu'elle soit appliquée sur une branche saturée non close. C'est ce que montre le lemme suivant :

**Lemme 5.9.** *Soit $\mathcal{F}$ un SHS de Hintikka. $\Omega(\mathcal{F})$ est un modèle.*

*Démonstration.* On montre tout d'abord que $\mathcal{R}$ est une structure de ressources faible (Définition 5.2).

- $e$ est bien un élément de $Nom$ ;
- $\bullet$ est bien une fonction partielle sur $Nom$ ;
- $\sim$ est correctement définie sur $\mathcal{F}$. En effet, les points 21, 22 et 23 (avec $\phi = z$) montrent que la définition de $\sim$ sur $\mathcal{F}$ respecte la réflexivité, symétrie et transitivité de $\sim$.
- Montrons que $\sim$ est compatible à droite avec $\bullet$. Soient $r, r', s \in Nom$ tels que $r \bullet s \downarrow$ et $r \sim r'$. Puisque $r \bullet s \downarrow$, $r, s \in \mathcal{A}(\mathcal{F})$ et il existe $x \in \mathcal{A}(\mathcal{F})$ tel que $r \bullet s \sim x$ et $\mathbb{T}\ @_x(r * s) \in \mathcal{F}$. Puisque $r \sim r'$, $r' \in \mathcal{A}(\mathcal{F})$ et $\mathbb{T}\ @_r(r') \in \mathcal{F}$. Alors, par le point 29 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_x(r' * s) \in \mathcal{F}$. Alors $r' \bullet s \downarrow$ et $r' \bullet s \sim x \sim r \bullet s$. Donc $\sim$ est compatible à droite avec $\bullet$.
- On montre de même que $\sim$ est aussi compatible à gauche avec $\bullet$.

Ainsi $\mathcal{R}$ est bien une structure de ressources faible. $[\![\cdot]\!]$ est bien une fonction de $Prop$ dans $\mathbb{P}(Nom)$. Montrons qu'elle est monotone. Soit $p \in Prop$. Soient $r, r' \in Nom$ tels que $r \sim r'$ et $r \in [\![p]\!]$. Alors on a $\mathbb{T}\ @_r(p) \in \mathcal{F}$. Comme $r \sim r'$, on a $\mathbb{T}\ @_r(r') \in \mathcal{F}$. Alors par le point 28, on a $\mathbb{T}\ @_{r'}(p) \in \mathcal{F}$, et donc $r' \in [\![p]\!]$. Par conséquent, $[\![\cdot]\!]$ est monotone, et par suite c'est une interprétation.

Finalement, comme $\vDash_{\mathcal{K}}$ est définie comme pour un modèle, on a bien que $\mathcal{K}$ est un modèle. □

On parvient donc à construire par $\Omega$ un modèle de HRL. Toutefois, il faut également que ce modèle soit un contre-modèle de $\phi$. Pour atteindre ce but, on énonce le lemme suivant.

**Lemme 5.10.** *Soit $\mathcal{F}$ un SHS de Hintikka et $\mathcal{K} = \Omega(\mathcal{F})$. Pour toute formule $\phi \in \mathcal{L}$ et pour tout $x \in \mathcal{A}(\mathcal{F})$ on a :*

- *Si $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$, alors $x \vDash_{\mathcal{K}} \phi$*
- *Si $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$, alors $x \nvDash_{\mathcal{K}} \phi$*

*Démonstration.*
La preuve de ce lemme est donnée en Annexe B. □

Enfin, on peut montrer que notre fonction $\Omega$ extrait bien des contre-modèles dans les branches ouvertes saturées :

**Lemme 5.11.** *Soit $\mathcal{F}$ un SHS de Hintikka tel que $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. Soit $\mathcal{K} = \Omega(\mathcal{F})$. La formule $\phi$ n'est pas valide et $\mathcal{K}$ est un contre-modèle de $\phi$.*

*Démonstration.*
Par le Lemme 5.9, $\mathcal{K}$ est un modèle. Par le Lemme 5.10, on a $x \nvDash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est un contre-modèle pour $\phi$. □

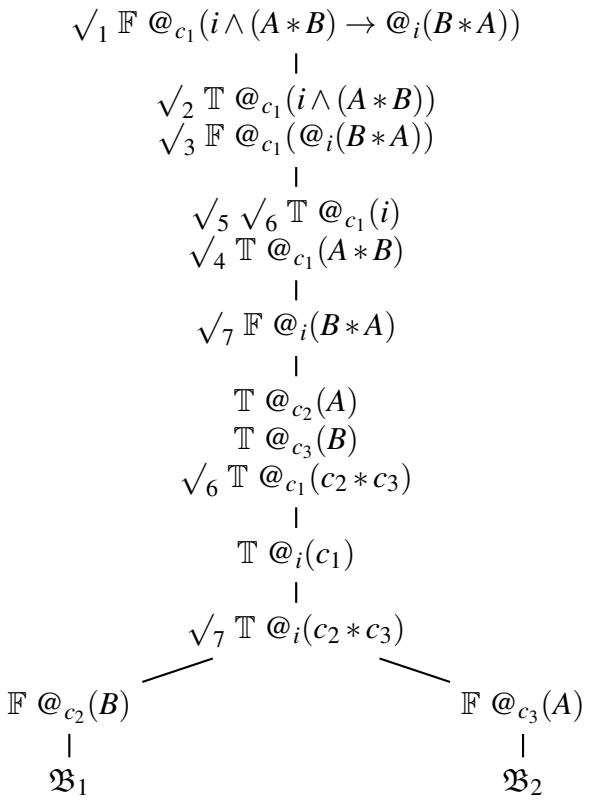



FIGURE 5.3 – HRL-tableau pour $i \wedge (A * B) \to @_i(B * A)$

### 5.6.3 Un exemple d'extraction de contre-modèles

Nous allons maintenant illustrer comment nous pouvons concrètement extraire un contre-modèle d'un tableau non clos. On se propose de dresser le tableau de la formule suivante (où $A, B \in Prop$ et $i \in Nom$) :

$$i \wedge (A * B) \to @_i(B * A)$$

La Figure 5.3 présente le tableau pour cette formule.

Comme on peut le constater, le développement du tableau mène à deux branches non closes $\mathfrak{B}_1$ et $\mathfrak{B}_2$. Informellement, on comprend que l'absence de commutativité de $\bullet$ dans HRL empêche de produire les formules $\mathbb{F}\ @_{c_2}(A)$ et $\mathbb{F}\ @_{c_3}(B)$ qu'on pourrait avoir. Notez que les deux branches ne diffèrent que par une formule étiquetée $\mathbb{F}$ et que de telles formules ne rentrent pas en compte dans la construction d'un contre-modèle (comme indiqué dans la Définition 5.12). Ainsi, l'extraction nous donne un résultat identique pour les deux branches.

Considérons la branche $\mathfrak{B}_1$. Il est important de noter que cette branche n'est pas tout à fait une branche de Hintikka car elle n'est pas complètement saturée. En effet, les règles $\langle i_r \rangle$, $\langle i_t \rangle$ et $\langle i_p \rangle$ n'ont pas été appliquées autant que faire ce peu. Cela dit, de telles manipulations ne font que remplacer des nominaux par d'autres qu'on sait déjà équivalents (en l'occurrence, $i$ par

$c_1$ ou $c_1$ par $i$). Il n'est donc pas nécessaire d'appliquer ces règles jusqu'au bout pour obtenir un contre-modèle satisfaisant, comme nous allons le montrer. De manière générale, on peut souvent omettre ce type de règles lorsqu'on recherche un contre-modèle.

La Définition 5.12 nous donne un modèle $\Omega(\mathfrak{B}_1) = \mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ avec $\mathcal{R} = (\bullet, e, \sim)$ défini par :

1. $e$ est un élément arbitraire de $Nom$ (nous n'avons pas de contrainte ici puisque I n'apparait pas dans notre tableau)
2. $\sim$ est une relation d'équivalence sur $Nom$ et on a en outre $i \sim c_1$ puisque $\mathbb{T}\ @_{c_1}(i) \in \mathfrak{B}_1$
3. $\bullet$ est définie par :
   - $c_2 \bullet c_3 \downarrow$ car $\mathbb{T}\ @_{c_1}(c_2 * c_3) \in \mathfrak{B}_1$, et de plus $c_1 \sim c_2 \bullet c_3$. On remarque qu'on obtient alors par transitivité que $i \sim c_2 \bullet c_3$ ce qui est cohérent puisque $\mathbb{T}\ @_i(c_2 * c_3) \in \mathfrak{B}_1$.
   - pour tout autre $r, r' \in Nom$, $r \bullet r' \uparrow$
4. $[\![\cdot]\!]$ est définie par :
   - $[\![A]\!] = c_2$
   - $[\![B]\!] = c_3$
5. $\vDash_{\mathcal{K}}$ est définie comme habituellement

On notera que $c_3 \bullet c_2$ n'est pas défini puisque $\bullet$ n'est pas commutative (c'est d'ailleurs ce qui rend le tableau non clos). Vérifions maintenant que $\mathcal{K}$ est bien un contre-modèle de $i \wedge (A * B) \rightarrow @_i(B * A)$.

1. Comme $\sim$ est une relation d'équivalence, on a en particulier $i \sim i$ et donc $i \vDash_{\mathcal{K}} i$.
2. Comme $c_2 \in [\![A]\!]$, on a $c_2 \vDash_{\mathcal{K}} A$.
3. Comme $c_3 \in [\![B]\!]$, on a $c_3 \vDash_{\mathcal{K}} B$.
4. De 2 et 3, et comme on a $i \sim c_2 \bullet c_3$ par transitivité, on conclut que $i \vDash_{\mathcal{K}} A * B$.
5. De 1 et 4, on déduit que $i \vDash_{\mathcal{K}} i \wedge (A * B)$.
6. Comme $c_2 \notin V(B)$, on a $c_2 \nvDash_{\mathcal{K}} B$.
7. Comme $c_3 \notin V(A)$, on a $c_3 \nvDash_{\mathcal{K}} A$.
8. Soit $x, y \in Nom$. On a deux cas :
   - si $x = c_2$ et $y = c_3$, alors $x \bullet y \downarrow$, $i \sim x \bullet y$ mais on a $x \nvDash_{\mathcal{K}} B$ et $y \nvDash_{\mathcal{K}} A$ d'après 6 et 7 ;
   - dans tous les autres cas, $x \bullet y \uparrow$.

   Donc finalement, pour tout $x, y \in Nom$, ou bien $x \bullet y \uparrow$, ou bien $i \sim x \bullet y$ et $x \nvDash_{\mathcal{K}} B$ et $y \nvDash_{\mathcal{K}} A$. Donc finalement, $i \nvDash_{\mathcal{K}} B * A$.
9. De 8, on a que pour tout $r \in Nom$, on a $r \nvDash_{\mathcal{K}} @_i(B * A)$. En particulier, on a $i \nvDash_{\mathcal{K}} @_i(B * A)$.
10. De 5 et 9 on tire que $i \nvDash_{\mathcal{K}} i \wedge (A * B) \rightarrow @_i(B * A)$, et donc $\mathcal{K}$ est un contre-modèle de cette formule.

Nous avons donc une méthode pour extraire des contre-modèles. Nous allons à présent utiliser ce procédé pour montrer la complétude de notre calcul des tableaux.

### 5.6.4 Complétude du calcul des HRL-tableaux

Nous pouvons, enfin, présenter notre preuve de complétude pour notre calcul. Cette preuve reprend la technique utilisée pour BI et ses variantes, que nous avons déjà employée dans les chapitres précédentes.

La preuve comportera toutefois quelques différences notables. La première consiste à étendre légèrement la notion de *stratégie équitable*. En effet, pour appliquer la règle $\langle i+\rangle$, nous devons sélectionner non seulement une formule du SHS, mais aussi le couple $(y,z)$ sur lequel l'appliquer.

Pour refléter ce fait dans notre stratégie équitable, nous numérotons toutes les apparitions de $y*z$ dans une formule et nous adjoignons à chaque formule sélectionner un entier dans cet intervalle.

**Définition 5.13** (Ensemble de paires)**.** *Soit $\phi \in \mathcal{L}$ une formule de HRL. L'*ensemble des paires *de $\phi$, noté $\mathcal{E}(\phi)$ est défini par :*

$$\mathcal{E}(\phi) = \{x*y | \phi[x*y]\}$$

*On appelle* formule indexée *tout couple $(\mathbb{S}\ @_x(\phi), i)$ où $\mathbb{S}\ @_x(\phi)$ est une formule étiquetée et $0 \leq i \leq n$ avec $n = |\mathcal{E}(\phi)|$.*

Notons que, si l'indice d'une formule indexée est 0, aucune paire de nominaux n'est désignée par elle dans la formule. Cela permet d'appliquer une autre règle que $\langle i+\rangle$ sur une formule. Définissons maintenant notre notion de stratégie équitable.

**Définition 5.14** (Stratégie équitable)**.** *Une* stratégie équitable *est une suite de formules indexées $(S_i)_{i\in\mathbb{N}}$ telle que chaque formule étiquetée apparait un nombre infini de fois dans la suite, c'est à dire que $\{i \in \mathbb{N} | S_i = (\mathbb{S}\ @_x(\phi), j)\}$ est infini pour tout $\mathbb{S} \in \{\mathbb{T}, \mathbb{F}\}$, $x \in Nom$, $F \in \mathcal{L}$, $j \in [0, |\mathcal{E}(\phi)|]$.*

**Proposition 5.1.** *Il existe une stratégie équitable.*

*Démonstration.*
Soit $X = \{\mathbb{T}, \mathbb{F}\} \times Nom \times \mathcal{L} \times \mathbb{N}$. Une formule indexée est choisie dans $X$. Comme *Prop* et *Nom* sont dénombrables, $\mathcal{L}$ est aussi dénombrable et $X$ est donc dénombrable. Donc $\mathbb{N} \times X$ est dénombrable et il existe une fonction surjective $\phi : \mathbb{N} \to \mathbb{N} \times X$. Soit $p : \mathbb{N} \times X \to X$ défini par $p(i,x) = x$ et $u = p \circ \phi$. On montre que $u$ est une stratégie équitable en montrant que pour tout $x \in X$, $u^{-1}(x)$ est infini. Soit $x \in X$. $u^{-1}(\{x\}) = \phi^{-1}(p^{-1}(\{x\}))$. Mais $p^{-1}(\{x\}) = \{(i,x) | i \in \mathbb{N}\}$ donc $p^{-1}(\{x\})$ est infini. Comme $\phi$ est surjective, $\phi^{-1}(p^{-1}(\{x\}))$ est aussi infini. □

Nous allons maintenant définir l'autre objet essentiel à notre preuve, l'oracle. Pour cela, nous définissons d'abord les propriétés d'un oracle.

**Définition 5.15** (Branches closes, finies et saturées)**.** *Soit $\mathcal{P}$ un ensemble de SHS.*

1. *$\mathcal{P}$ est* $\subseteq$-clos *si $\mathcal{F} \in \mathcal{P}$ chaque fois qu'on a $\mathcal{F} \subseteq \mathcal{F}'$ et $\mathcal{F}' \in \mathcal{P}$.*
2. *$\mathcal{P}$ est de* caractère fini *si $\mathcal{F} \in \mathcal{P}$ chaque fois qu'on a $\mathcal{F}_f \in \mathcal{P}$ pour tout $\mathcal{F}_f \subseteq_f \mathcal{F}$.*
3. *$\mathcal{P}$ est* saturé *si pour tout $\mathcal{F} \in \mathcal{P}$ et pour toute instance*

$$\frac{H_1,\ldots,H_l}{C_1^1,\ldots,C_{p1}^1 \mid \ldots \mid C_1^k,\ldots,C_{pk}^k}$$

*de règle de la Figure 5.1, si $H_i \in \mathcal{F}$ pour tout $i \in \{1,\ldots,l\}$, alors $\mathcal{F} \cup \{C_1^j,\ldots,C_{p_j}^j\} \in \mathcal{P}$ pour au moins un $j \in \{1,\ldots,k\}$.*

**Définition 5.16** (Oracle)**.** *Un* oracle *est un ensemble de SHS non clos qui est $\subseteq$-clos, de caractère fini et saturé.*

**Lemme 5.12.** *Il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe aucun tableau clos.*

*Démonstration.*
La preuve de ce lemme est donnée en Annexe B. Le schéma de preuve est identique à celui des autres logiques (voir les chapitres précédents) mais sont simplifiées par le fait que ce calcul ne possède pas de contraintes de ressources et de clôture de contraintes. □

Pour montrer la complétude, on considère une formule $\phi$ pour laquelle il n'existe pas de preuve et on montre qu'il en existe un contre-modèle.

Par commodité, on suppose qu'on peut déterminer un ensemble $Nom_e$ de *nominaux efficaces*, défini comme l'ensemble des nominaux apparaissant dans des formules de HRL. Concrètement, un tel ensemble peut être déterminé pour tout ensemble de formules qu'on considère. On peut alors définir un ensemble de nominaux $(c_i)_{i\in\mathbb{N}}$ tel que $\forall i, c_i \notin Nom_e$.

On note $\mathcal{T}_0$ le tableau initial pour $\phi$. On a :

1. $\mathcal{T}_0 = [\{\mathbb{F}\ @_{c_1}(\phi)\}]$
2. $\mathcal{T}_0$ ne peut pas être clos

On présente maintenant une méthode pour obtenir un SHS de Hintikka. Par le Lemme 5.12 il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe pas de tableau clos. On note $\mathcal{P}$ cet oracle. Par la Proposition 5.1 il existe une stratégie équitable. On la note $\mathcal{S}$ et on note $\mathcal{S}_i = (F_i, n_i)$ la $i^{\text{ème}}$ formule de $\mathcal{S}$. Comme $\mathcal{T}_0$ ne peut être clos, son unique branche appartient à l'oracle, c'est à dire $\{\mathbb{F}\ @_{c_1}(\phi)\} \in \mathcal{P}$.

On construit maintenant la suite $\mathcal{F}_{i\geq 0}$comme ceci :

- $\mathcal{F}_0 = \{\mathbb{F}\ @_{c_1}(\phi)\}$
- Si $\mathcal{F}_i \cup \{F_i\} \notin \mathcal{P}$ alors $\mathcal{F}_{i+1} = \mathcal{F}_i$
- Si $\mathcal{F}_i \cup \{F_i\} \in \mathcal{P}$ alors $\mathcal{F}_{i+1} = \mathcal{F}_i \cup \{F_i\} \cup \mathcal{F}_e$ où $\mathcal{F}_e$ est défini par :

| $n_i$ | $F_i$ | $\mathcal{F}_e$ |
|---|---|---|
| 0 | $\mathbb{T}\ @_x(\phi * \psi)$ | $\{\mathbb{T}\ @_{\mathfrak{a}}(\phi), \mathbb{T}\ @_{\mathfrak{b}}(\psi), \mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b})\}$ |
| 0 | $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi)$ | $\{\mathbb{T}\ @_{\mathfrak{a}}(\phi), \mathbb{F}\ @_{\mathfrak{b}}(\psi), \mathbb{T}\ @_{\mathfrak{b}}(x * \mathfrak{a})\}$ |
| 0 | $\mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi)$ | $\{\mathbb{T}\ @_{\mathfrak{a}}(\phi), \mathbb{F}\ @_{\mathfrak{b}}(\psi), \mathbb{T}\ @_{\mathfrak{b}}(\mathfrak{a} * x)\}$ |
| $0 < n \leq \lvert\mathcal{E}(F_i)\rvert$ | $\mathbb{T}\ @_x(\phi[y * z])$ | $\{\mathbb{T}\ @_x(\phi[y * z/\mathfrak{a}]), \mathbb{T}\ @_{\mathfrak{a}}(y * z)\}$ |
| $0 < n \leq \lvert\mathcal{E}(F_i)\rvert$ | $\mathbb{F}\ @_x(\phi[y * z])$ | $\{\mathbb{F}\ @_x(\phi[y * z/\mathfrak{a}]), \mathbb{T}\ @_{\mathfrak{a}}(y * z)\}$ |
| Dans les autres cas | | $\emptyset$ |

avec $\mathfrak{a} = c_{2i+2}$ et $\mathfrak{b} = c_{2i+3}$ et $y * z$ est le $n^{\text{ème}}$ élément de $\mathcal{E}(F_i)$

Cette construction de SHS nous permet de contrôler l'introduction de nouveaux labels, tout en nous assurant que les règles générant les dits labels sont saturées.

**Proposition 5.2.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$
2. $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_i$
3. $(\mathcal{F}_i)_{i\geq 0} \in \mathcal{P}$
4. $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$

*Démonstration.*
La preuve de cette proposition est donnée en Annexe B. □

On considère maintenant le SHS limite $\mathcal{F}_\infty$ défini par :

$$\mathcal{F}_\infty = \bigcup_{i\geq 0} \mathcal{F}_i$$

**Proposition 5.3.** *Les propriétés suivantes sont vraies :*

1. $\mathcal{F}_\infty \in \mathcal{P}$
2. *Pour toute formule étiquetée* $\mathbb{S}\ @_x(\phi)$, *si* $\mathcal{F}_\infty \cup \{\mathbb{S}\ @_x(\phi)\} \in \mathcal{P}$, *alors* $\mathbb{S}\ @_x(\phi) \in \mathcal{F}_\infty$

*Démonstration.*

1. Soit $\mathcal{F}_f$ tel que $\mathcal{F}_f \subseteq_f \mathcal{F}_\infty$. Comme $\mathcal{F}_f$ est fini, et comme la suite $(\mathcal{F}_i)_{i\geq 0}$ est croissante (d'après le point 1. de la Proposition 5.2), il existe $j \in \mathbb{N}$ tel que $\mathcal{F}_f \in \mathcal{F}_j$. Par le point 3. de la Proposition 5.2, on a $\mathcal{F}_j \in \mathcal{P}$. Comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_f \in \mathcal{P}$. On a ainsi $\mathcal{F}_f \in \mathcal{P}$ pour tout $\mathcal{F}_f \subseteq_f \mathcal{F}_\infty$, ce qui veut dire que $\mathcal{F}_\infty \in \mathcal{P}$ puisque $\mathcal{P}$ est de caractère fini.
2. Soit $\mathbb{S}\ @_x(\phi)$ tel que $\mathcal{F}_\infty \cup \{\mathbb{S}\ @_x(\phi)\} \in \mathcal{P}$. Comme $\mathcal{S}$ est une stratégie équitable, il existe $k$ tel que $F_k = \mathbb{S}\ @_x(\phi)$. On a alors $\mathcal{F}_k \subseteq \mathcal{F}_\infty$ et donc $\mathcal{F}_k \cup \{\mathbb{S}\ @_x(\phi)\} \subseteq \mathcal{F}_\infty \cup \{\mathbb{S}\ @_x(\phi)\}$. Comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_k \cup \{\mathbb{S}\ @_x(\phi)\} \in \mathcal{P}$. Donc par construction, on a $\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{F_k\} \cup \mathcal{F}_e$, c'est à dire que $F_k \in \mathcal{F}_{k+1}$. Alors, par le point 1. de la Proposition 5.2, on a $F_k \in \mathcal{F}_\infty$ ce qui est le résultat attendu.

□

**Lemme 5.13.** *Le SHS limite est un SHS de Hintikka.*

*Démonstration.*
On vérifie toutes les conditions de la Définition 5.11 :

1. On suppose que $\mathbb{T}\ @_x(\phi) \in \mathcal{F}_\infty$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}_\infty$. Alors $\mathcal{F}_\infty$ est fermé, et donc $\mathcal{F}_\infty \notin \mathcal{P}$ par définition de l'oracle. C'est absurde d'après le point 1 de la Proposition 5.3. Donc $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}_\infty$ et $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}_\infty$.
4. On suppose que $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}_\infty$. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle \mathbb{T}\wedge \rangle$, on a $\mathcal{F}_\infty \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$. Comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_\infty \cup \{\mathbb{T}\ @_x(\phi)\} \in \mathcal{P}$ et $\mathcal{F}_\infty \cup \{\mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{T}\ @_x(\phi) \in \mathcal{F}_\infty$ et $\mathbb{T}\ @_x(\psi) \in \mathcal{F}_\infty$.

5. On suppose que $\mathbb{F}\ @_x(\phi \wedge \psi) \in \mathcal{F}_\infty$. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle \mathbb{F}\wedge \rangle$, on a $\mathcal{F}_\infty \cup \{\mathbb{F}\ @_x(\phi)\} \in \mathcal{P}$ ou $\mathcal{F}_\infty \cup \{\mathbb{F}\ @_x(\psi)\} \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{F}\ @_x(\phi) \in \mathcal{F}_\infty$ ou $\mathbb{F}\ @_x(\psi) \in \mathcal{F}_\infty$.
12. On suppose $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}_\infty$. Il existe $j$ tel que $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}_j$. Comme $\mathcal{S}$ est une stratégie équitable, il existe $k > j$ tel que $F_k = \mathbb{T}\ @_x(\phi * \psi)$. Par le point 1. de la Proposition 5.2, $\mathcal{F}_j \subseteq \mathcal{F}_k$, donc $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}_k$. On a alors $\mathcal{F}_k \cup \{\mathbb{T}\ @_x(\phi * \psi)\} = \mathcal{F}_k$. Par le point 3. de la Proposition 5.2, $\mathcal{F}_k \in \mathcal{P}$, donc $\mathcal{F}_k \cup \{\mathbb{T}\ @_x(\phi * \psi)\} \in \mathcal{P}$. On a alors $\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{F_k\} \cup \{\{\mathbb{T}\ @_\mathfrak{a}(\phi), \mathbb{T}\ @_\mathfrak{b}(\psi), \mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b})\}$ avec $\mathfrak{a} = c_{2i+2}$ et $\mathfrak{b} = c_{2i+3}$. On a donc $\mathbb{T}\ @_\mathfrak{a}(\phi) \in \mathcal{F}_{k+1}$, $\mathbb{T}\ @_\mathfrak{b}(\psi) \in \mathcal{F}_{k+1}$ et $\mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b}) \in \mathcal{F}_{k+1}$. Par construction de $\mathcal{F}_\infty$, on a $\mathcal{F}_{k+1} \subseteq \mathcal{F}_\infty$ et donc $\mathbb{T}\ @_\mathfrak{a}(\phi) \in \mathcal{F}_\infty$, $\mathbb{T}\ @_\mathfrak{b}(\psi) \in \mathcal{F}_\infty$ et $\mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b}) \in \mathcal{F}_\infty$. En outre, par le point 4. de la Proposition 5.2, on a $\mathfrak{a} \in \mathcal{A}(\mathcal{F}_\infty)$ et $\mathfrak{b} \in \mathcal{A}(\mathcal{F}_\infty)$. Ainsi on a bien vérifié la condition 13. de la Définition 5.11 avec $y = \mathfrak{a}$ et $z = \mathfrak{b}$.
13. On suppose que $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}_\infty$ et $\mathbb{T}\ @_x(y * z) \in \mathcal{F}_\infty$. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle \mathbb{F}* \rangle$, on a $\mathcal{F}_\infty \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$ ou $\mathcal{F}_\infty \cup \{\mathbb{F}\ @_z(\psi)\} \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{F}\ @_y(\phi) \in \mathcal{F}_\infty$ ou $\mathbb{F}\ @_z(\psi) \in \mathcal{F}_\infty$.
20. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle i_r \rangle$, $\mathcal{F}_\infty \cup \mathbb{T}\ @_x(x) \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{T}\ @_x(x) \in \mathcal{F}_\infty$.
24. On suppose que $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}_\infty$. Il existe $j$ tel que $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}_j$. Comme $\phi[y*z]$, on a $y*z \in \mathcal{E}(\phi)$. Il existe donc $n$ tel que $y*z$ est le $n^{\text{ième}}$ élément de $\mathcal{E}(\phi)$. Comme $\mathcal{S}$ est une stratégie équitable, il existe $k > j$ tel que $\mathcal{S}_k = (\mathbb{T}\ @_x(\phi), n)$. Par le point 3. de la Proposition 5.2, $\mathcal{F}_k \in \mathcal{P}$, donc $\mathcal{F}_k \cup \{\mathbb{T}\ @_x(\phi)\} \in \mathcal{P}$. On a alors $\mathcal{F}_{k+1} = \mathcal{F}_k \cup \{\mathbb{T}\ @_x(\phi)\} \cup \{\{\mathbb{T}\ @_\mathfrak{x}(\phi[y*z/\mathfrak{a}]), \mathbb{T}\ @_\mathfrak{a}(y*z)\}$ avec $\mathfrak{a} = c_{2i+2}$. On a donc $\mathbb{T}\ @_x(\phi[y*z/\mathfrak{a}]) \in \mathcal{F}_{k+1}$ et $\mathbb{T}\ @_\mathfrak{a}(y*z) \in \mathcal{F}_{k+1}$. Par construction de $\mathcal{F}_\infty$, on a $\mathcal{F}_{k+1} \subseteq \mathcal{F}_\infty$ et donc $\mathbb{T}\ @_x(\phi[y*z/\mathfrak{a}]) \in \mathcal{F}_\infty$ et $\mathbb{T}\ @_\mathfrak{a}(y*z) \in \mathcal{F}_\infty$. En outre, par le point 4. de la Proposition 5.2, on a $\mathfrak{a} \in \mathcal{A}(\mathcal{F}_\infty)$. Enfin, ce raisonnement est valable pour tout $y*z$ dans $\phi$. Ainsi on a bien vérifié la condition 25. de la Définition 5.11 avec $t = \mathfrak{a}$.

Les autres cas sont similaires. □

**Théorème 5.4** (Complétude)**.** *Soit $\phi$ une formule de HRL. Si $\phi$ est valide, alors il existe une preuve pour $\phi$.*

*Démonstration.*
On suppose que $\phi$ est valide. On suppose qu'il n'y a pas de preuve pour $\phi$. La méthode que nous avons présentée permet de construire un SHS limite $\mathcal{F}_\infty$. Par le Lemme 5.13, $\mathcal{F}_\infty$ est un SHS de Hintikka. Par le point 2. de la Proposition 5.2, $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_i$ pour tout $i$, c'est à dire que $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_\infty$ par construction. Alors, d'après le Lemme 5.11, $\phi$ n'est pas valide. C'est absurde. Donc il existe une preuve pour $\phi$. □

## 5.7 Un calcul des tableaux pour HBBI

Comme nous l'avons fait remarquer, les règles de la Figure 5.1 ne prennent pas en compte les additions axiomatiques de HBBI. Nous allons montrer comment nous pouvons intégrer des

variantes de HRL, comme HBBI, à notre calcul des tableaux.

### 5.7.1 Trois manières d'étendre le calcul des HRL-tableaux

Pour toute restriction de HRL par une axiomatique, nous pouvons étendre le calcul des tableaux que nous venons d'énoncer. Pour cela, nous rajoutons aux règles de la Figure 5.1 des règles additionnelles correspondants aux axiomes rajoutés à HRL. Nous présentons ici trois manières de le faire, et nous proposons comme exemple les règles pour HBBI :

#### Première approche : axiomes comme formules de départ

Les axiomes sont des formules valides, ils peuvent donc être intégrées comme des formules étiquetées par $\mathbb{T}$ dans les hypothèses de départ d'un tableau (puisque ce calcul est correct).

Au lieu du SHS de départ habituel $\{\mathbb{F}\ @_{c_1}(\phi)\}$, on doit ainsi inclure toutes les instances de tous les schémas d'axiomes de la variante de HRL qu'on considère.

Par exemple, les axiomes de HBBI étant les suivants :

$$
\begin{aligned}
(BI)_n &\equiv @_i(i * \mathrm{I}) \\
(BI)_c &\equiv j * k \to k * j \\
(BI)_a &\equiv j * (k * l) \to (j * k) * l
\end{aligned}
$$

Le SHS de départ de tout tableau pour HBBI serait donc le suivant :

$$
\begin{aligned}
\mathcal{F} \;=\; & \{\mathbb{F}\ @_{c_1}(\phi)\} \\
\cup\; & \bigcup_{i \in Nom} \{\mathbb{T}\ @_{c_1}(@_i(i * \mathrm{I}))\} \\
\cup\; & \bigcup_{j,k \in Nom} \{\mathbb{T}\ @_{c_1}(j * k \to k * j)\} \\
\cup\; & \bigcup_{j,k,l \in Nom} \{\mathbb{T}\ @_{c_1}(j * (k * l) \to (j * k) * l)\}
\end{aligned}
$$

#### Deuxième approche : axiomes comme règles axiomatiques

Les axiomes peuvent aussi être écrits sous forme de règles axiomatiques (sans prémisses). Cette méthode présente l'avantage de maintenir le SHS initial $\{\mathbb{F}\ @_{c_1}(\phi)\}$ et est une traduction plus directe de l'axiomatique : chaque schéma d'axiome correspond directement à une règle.

Pour le cas de HBBI, les axiomes suivants :

$$
\begin{aligned}
(BI)_n &\equiv @_i(i * \mathrm{I}) \\
(BI)_c &\equiv j * k \to k * j \\
(BI)_a &\equiv j * (k * l) \to (j * k) * l
\end{aligned}
$$

se traduisent en les trois règles ci-dessous :

$$\frac{}{\mathbb{T}\ @_x(@_y(y*\mathrm{I}))}\ \langle BI_n - ax\rangle \qquad \frac{}{\mathbb{T}\ @_x(y*z\to z*y)}\ \langle BI_c - ax\rangle$$

$$\frac{}{\mathbb{T}\ @_x(y*(z*t)\to(y*z)*t)}\ \langle BI_a - ax\rangle$$

On voit facilement comment cette méthode est équivalente à la première : chaque formule qu'on aurait trouvée dans le SHS initial de la première méthode peut être déduite par l'une des règles axiomatiques de la deuxième.

**Troisième approche : simplification des règles axiomatiques**

On remarque dans l'exemple particulier de HBBI que nous avons présenté dans la méthode précédente que les règles axiomatiques introduites peuvent être changées par des applications d'autres règles de la Figure 5.1. Par exemple, on peut développer un axiome de la forme $A \to B$ par la règle $\langle \mathbb{T} \to \rangle$ et un axiome de la forme $@_x(@_y(A))$ par la règle $\langle @ \rangle$.

Ainsi, dans l'exemple particulier de HBBI, on peut effectuer les changement suivants aux règles introduites :

$$\frac{}{\mathbb{T}\ @_x(@_y(y*\mathrm{I}))}\ \langle BI_n - ax\rangle \longrightarrow \frac{}{\mathbb{T}\ @_x(x*\mathrm{I})}\ \langle BI_n\rangle$$

$$\frac{}{\mathbb{T}\ @_x(y*z\to z*y)}\ \langle BI_c - ax\rangle \longrightarrow \frac{\mathbb{T}\ @_x(y*z)}{\mathbb{T}\ @_x(z*y)}\ \langle BI_c\rangle$$

$$\frac{}{\mathbb{T}\ @_x(y*(z*t)\to(y*z)*t)}\ \langle BI_a - ax\rangle \longrightarrow \frac{\mathbb{T}\ @_x(y*(z*t))}{\mathbb{T}\ @_x((y*z)*t)}\ \langle BI_a\rangle$$

De manière évidente, cette approche est équivalente à la deuxième, et donc également à la première.

Ces trois approches d'extension du calcul des HRL-tableaux peuvent être utilisées indifféremment pour raisonner sur toute restriction axiomatique de HRL. Il conviendra bien entendu de prouver que la correction et la complétude du calcul sont maintenus par ces extensions (ce que nous allons faire dans le cas de HBBI).

Parmi les trois approches, on préférera choisir la dernière dès que c'est possible, puisque les règles y sont moins nombreuses et plus concises.

Notons que cette méthode d'extension du calcul peut être répétée pour de nouveaux axiomes. Par exemple, si nous considérons l'axiome introduit dans la Section 5.4.2 pour caractériser les ressources inversibles :

$$Inv(i) = (i*\top)\wedge \mathrm{I}$$

On peut alors transformer cet axiome en une règle avec nos méthodes, ce qui produit la règle présentée par la Figure 5.4.

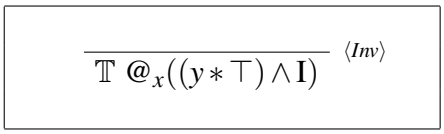


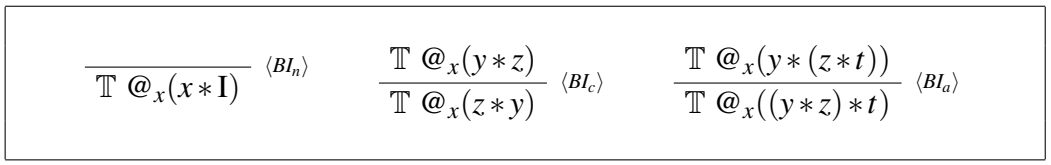

FIGURE 5.4 – Règle additionnelle pour l'inversibilité des ressources

Nous allons maintenant formaliser plus précisément l'extension du calcul pour le cas de HBBI.

### 5.7.2 Règles supplémentaires pour HBBI

Nous définissons un calcul des tableaux pour HBBI par la définition suivante :

**Définition 5.17** (HBBI-tableau)**.** *Soit $\phi$ une formule de HBBI. Un HBBI-tableau pour $\phi$ est un HRL-tableau pour $\phi$ où les 3 règles suivantes sont ajoutées aux règles de la Figure 5.1 :*




*Une HBBI-preuve pour $\phi$ est un HBBI-tableau clos pour $\phi$.*

Étant donné le Théorème 5.2, et vu la correction et la complétude du calcul pour HBBI que nous allons exposer plus loin, il est évident qu'une HBBI-preuve est aussi un moyen de valider une formule de BBI, ou d'en extraire un contre-modèle.

### 5.7.3 Un exemple de HBBI-tableaux

Nous allons à présent regarder un exemple de tableau avec une formule de HBBI. Comme on l'avait exposé dans la Définition 5.17, le calcul est le même mais trois règles supplémentaires peuvent être appliquées, correspondant aux trois axiomes définissant HBBI.

Comme HBBI et BBI sont sémantiquement équivalentes (ainsi que démontré dans le Théorème 5.2), on se propose de développer un exemple de formule valide dans BBI.

La formule en question, qui met notamment en exergue l'associativité de $\bullet$, est la suivante :

$$((A * (B \mathrel{-\!\!*} C)) * D) \rightarrow (A * ((B \mathrel{-\!\!*} C) * D))$$

Comme dans l'exemple précédent, on commence avec un simple SHS que nous allons développer.

$$\{\mathbb{F}\ @_{c_1}(((A * (B \mathrel{-\!\!*} C)) * D) \rightarrow (A * ((B \mathrel{-\!\!*} C) * D)))\}$$

Nous écrivons à nouveau les SHS sous forme de graphe. Encore une fois, l'opérateur externe est une implication ce qui démarre notre tableau de manière similaire.

On continue à développer le tableau en appliquant les règles et en les marquant successivement de $\surd$. Le tableau complet est donné en Figure 5.5. Les étapes en sont les suivantes :

$$\begin{array}{c}\mathbb{F}\ @_{c_1}(((A*(B\mathrel{-\!\!*}C))*D)\to(A*((B\mathrel{-\!\!*}C)*D)))\\ |\\ \mathbb{T}\ @_{c_1}((A*(B\mathrel{-\!\!*}C))*D)\\ \mathbb{F}\ @_{c_1}(A*((B\mathrel{-\!\!*}C)*D))\end{array}$$

1. On applique $\langle\mathbb{F}\to\rangle$ qui produit deux nouvelles formules.
2. On a le choix d'appliquer deux règles, $\langle\mathbb{T}*\rangle$ ou $\langle\mathbb{F}*\rangle$. Selon nos priorités, nous devons appliquer $\langle\mathbb{T}*\rangle$ d'abord, ce qui produit trois nouvelles formules.
3. Encore une fois on peut appliquer $\langle\mathbb{T}*\rangle$ ou $\langle\mathbb{F}*\rangle$ et on doit commencer par $\langle\mathbb{T}*\rangle$.
4. À ce stade, on souhaiterait appliquer la règle $\langle\mathbb{F}*\rangle$, mais il faut le faire intelligemment. On voit bien que le nominal $c_3$ doit être associé à $D$, $c_4$ à $A$ et $c_5$ à $B\mathrel{-\!\!*}C$. Pour cela, nous devons recombiner les nominaux, d'abord en fusionnant $\mathbb{T}\ @_{c_1}(c_2*c_3)$ et $\mathbb{T}\ @_{c_2}(c_4*c_5)$ en utilisant la règle $\langle i-\rangle$.
5. On souhaite à présent changer le parenthésage, ce qui est possible car dans HBBI on peut appliquer la règle $\langle BI_a\rangle$. Toutefois, le schéma de la règle n'est pas dans le bon sens. En effet, le prémisse de la règle est $\mathbb{T}\ @_x(y*(z*t))$ et on a la formule $\mathbb{T}\ @_{c_1}((c_4*c_5)*c_3)$ dans notre tableau. On doit donc appliquer plusieurs instances successives de $\langle BI_a\rangle$ et $\langle BI_c\rangle$ pour obtenir le schéma désiré. Cette étape est omise du tableau pour le moment et sera détaillée et discutée plus loin.
6. Enfin, nous introduisons un nominal intermédiaire par la règle $\langle i+\rangle$.
7. On peut maintenant utiliser $\langle\mathbb{F}*\rangle$, qui produit deux branches, l'une d'entre elles étant d'ors et déjà close.
8. Dans l'autre branche, on peut utiliser à nouveau $\langle\mathbb{F}*\rangle$ et créer deux nouvelles branches, toutes deux closes.

Finalement toutes les branches sont closes et on a obtenu une HBBI-preuve de notre formule $((A*(B\mathrel{-\!\!*}C))*D)\to(A*((B\mathrel{-\!\!*}C)*D))$ ce qui en confirme la validité dans HBBI par le théorème de correction que nous allons énoncer ci-après et également, par le Théorème 5.2, sa validité dans BBI.

Revenons maintenant sur l'étape de l'associativité. Cette étape devient complexe puisqu'on souhaite utiliser l'associativité sur $(c_4*c_5)*c_3$ pour produire $c_4*(c_5*c_3)$, mais la règle $\langle BI_a\rangle$ s'applique avec le schéma $x*(y*z)$. Ainsi, un raisonnement formel nous force à jongler avec les règles $\langle BI_a\rangle$ et $\langle BI_c\rangle$ comme explicité dans la Figure 5.6.

Ce détail met en avant comment l'apparition des propriétés de $\bullet$ dans les règles rend fastidieuse des manipulations qui étaient simples et souvent implicites dans les calculs avec labels et contraintes, de par la nature même des labels et le concept de clôture d'ensembles de contraintes (voir par exemple 2.17 et 2.18). Les opérations décrites explicitement ici, dans la Figure 5.6, sont effectuées implicitement lors du calcul de la clôture des contraintes d'états dans les calculs avec labels. Nous examinons plus en détail dans le Chapitre 6 les liens entre le calcul des HBBI-tableaux et le calcul des BBI-tableaux avec labels.

On peut toutefois remédier à ce problème. On peut écrire de nouvelles règles qui généralisent des manipulations telles que celles que nous avons faites dans la Figure 5.6. Par exemple, le raisonnement de cette figure, généralisé, justifie l'introduction de la règle suivante :

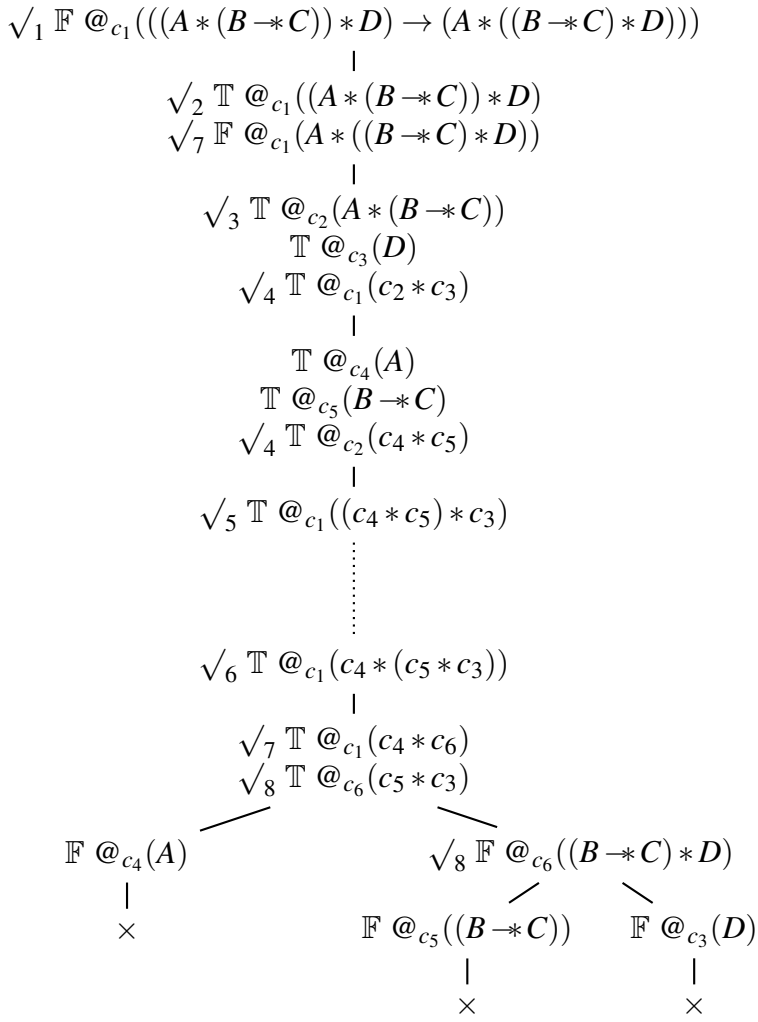



FIGURE 5.5 – HBBI-tableau pour $((A * (B \mathrel{-\!\!*} C)) * D) \to (A * ((B \mathrel{-\!\!*} C) * D))$

$$
\begin{array}{c}
\mathbb{T}\ @_{c_1}((c_4 * c_5) * c_3) \\
\Big| \quad \langle BI_c \rangle \\
\mathbb{T}\ @_{c_1}(c_3 * (c_4 * c_5)) \\
\Big| \quad \langle BI_a \rangle \\
\mathbb{T}\ @_{c_1}((c_3 * c_4) * c_5) \\
\Big| \quad \langle BI_c \rangle \\
\mathbb{T}\ @_{c_1}(c_5 * (c_3 * c_4)) \\
\Big| \quad \langle BI_a \rangle \\
\mathbb{T}\ @_{c_1}((c_5 * c_3) * c_4) \\
\Big| \quad \langle BI_c \rangle \\
\mathbb{T}\ @_{c_1}(c_4 * (c_5 * c_3))
\end{array}
$$

FIGURE 5.6 – Développement particulier de l'associativité dans le tableau de la Figure 5.5

$$
\frac{\mathbb{T}\ @_x((y * z) * t)}{\mathbb{T}\ @_x(y * (z * t))}\ \langle BI_a^* \rangle
$$

Toutefois, il ne faut pas oublier que l'introduction des propriétés de $\bullet$ par de nouvelles règles, si elle alourdit parfois les preuves, est aussi ce qui garantit la flexibilité du calcul et sa capacité à être étendu pour de nouvelles axiomatiques. On peut par exemple s'interdire certaines règles pour examiner l'impact de certains axiomes sur les variantes de HRL.

### 5.7.4 Propriétés du calcul des HBBI-tableaux

Nous présentons ici les preuves que l'extension du calcul des HRL-tableaux en le calcul des HBBI-tableaux préserve la correction et la complétude de ce calcul.

#### I) Correction du calcul des HBBI-tableaux

Commençons par montrer que le calcul pour HBBI est correct. La notion de réalisibilité est la même pour les HBBI-tableaux, avec la notion que tout modèle qui réalise une branche doit être un modèle de HBBI, donc valider les axiomes de la Définition 5.6. Il reste ensuite à s'assurer que les nouvelles règles de la Définition 5.17 conservent également la réalisabilité.

**Lemme 5.14.** *Les règles du calcul des tableaux pour HBBI (Définition 5.17) préservent la réalisabilité.*

*Démonstration.*
Soit $\mathcal{T}$ un tableau réalisable. Par définition, il contient une branche réalisable $\mathcal{F}$. Soit $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ le modèle de HBBI qui réalise $\mathcal{F}$.

On étend $\mathcal{T}$ par une des règles de la Figure 5.1. Si la branche étendue n'est pas $\mathcal{F}$, alors $\mathcal{F}$ reste une branche du nouveau tableau et ce dernier reste réalisable. Dans le cas contraire, c'est $\mathcal{F}$ qui est étendue et on discute suivant la règle appliquée :

- $\langle BI_n \rangle$
  Comme $\mathcal{K}$ est un modèle de HBBI, l'axiome $(BI)_n$ est en particulier valide pour tout nominal dans ce modèle. Soit $r \in Nom$. On a $r \vDash_{\mathcal{K}} (BI)_n$ pour $x$, c'est à dire $r \vDash_{\mathcal{K}} @_x(x * \mathrm{I})$. Par suite on a $x \vDash_{\mathcal{K}} x * \mathrm{I}$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}$.
- $\langle BI_c \rangle$
  Alors on a $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} y * z$. Comme $\mathcal{K}$ est un modèle de HBBI, $x \vDash_{\mathcal{K}} (BI)_c$ pour $y$ et $z$, c'est à dire que $x \vDash_{\mathcal{K}} y * z \rightarrow z * y$. Par suite on a $x \vDash_{\mathcal{K}} z * y$ et $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\}$.
- $\langle BI_a \rangle$
  Alors on a $\mathbb{T}\ @_x(y * (z * t)) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} y * (z * t)$. Comme $\mathcal{K}$ est un modèle de HBBI, $x \vDash_{\mathcal{K}} (BI)_a$ pour $y$, $z$ et $t$, c'est à dire que $x \vDash_{\mathcal{K}} y * (z * t) \rightarrow (y * z) * t$. Par suite on a $x \vDash_{\mathcal{K}} (y * z) * t$ et $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x((y * z) * t)\}$.
- Pour toutes les règles de la Figure 5.1, le Lemme 5.7 assure le résultat.

□

Cela nous mène directement au théorème suivant.

**Théorème 5.5** (Correction des HBBI-tableaux)**.** *S'il existe une HBBI-preuve de* $\phi$*, alors* $\phi$ *est valide pour HBBI.*

*Démonstration.*
La preuve est identique à celle du Théorème 5.3 mais en utilisant le Lemme 5.14 à la place du Lemme 5.7. □

Enfin, cela nous permet d'utiliser notre calcul pour raisonner sur BBI.

**Corollaire 5.1.** *Soit* $\phi$ *une formule de* $\mathcal{L}_{BBI}$*. S'il existe une HBBI-preuve de* $\phi$*, alors* $\phi$ *est valide dans BBI.*

*Démonstration.*
Supposons qu'il existe une HBBI-preuve de $\phi$. Par le Théorème 5.5, $\phi$ est HBBI-valide. Par le Théorème 5.2, $\phi$ est BBI-valide. □

Ainsi notre calcul est correct dans sa variante pour HBBI, ce qui en fait un calcul pour BBI. Notons au passage que la correction de futures extensions ne devrait pas poser de problème vu ce travail, dans la mesure où les règles introduites dans l'axiomatique sont correctement traduites dans le calcul des tableaux.

**II) Complétude du calcul des HBBI-tableaux**

Regardons maintenant la complétude du calcul pour HBBI. Pour cela, on reprend toutes les étapes de la preuve de complétude pour HRL en rajoutant les règles propres à HBBI dans les définitions et les preuves.

**Définition 5.18** (SHS de Hintikka pour HBBI - extension de 5.11)**.** *Un SHS de Hintikka pour HBBI est donné par la Définition 5.11 avec l'ajout des 3 conditions suivantes :*

*31.* $\mathbb{T}\ @_x(x*\mathrm{I}) \in \mathcal{F}$.

*32. Si* $\mathbb{T}\ @_x(y*z) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x(z*y) \in \mathcal{F}$.

*33. Si* $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}$, *alors* $\mathbb{T}\ @_x((y*z)*t) \in \mathcal{F}$.

La sémantique de HBBI étant la même que celle de HRL, la définition de $\Omega$ ne change pas, ni le Lemme 5.9. Les lemmes 5.10 et 5.11 sont aussi inchangés.

On modifie la suite des définitions et propositions comme suit :

**Définition 5.19** (Branches closes, finies et saturées pour HBBI - extension de 5.15)**.** *Soit $\mathcal{P}$ un ensemble de SHS.*

1. *$\mathcal{P}$ est* $\subseteq$-clos *si $\mathcal{F} \in \mathcal{P}$ chaque fois qu'on a $\mathcal{F} \subseteq \mathcal{F}'$ et $\mathcal{F}' \in \mathcal{P}$.*
2. *$\mathcal{P}$ est de* caractère fini *si $\mathcal{F} \in \mathcal{P}$ chaque fois qu'on a $\mathcal{F}_f \in \mathcal{P}$ pour tout $\mathcal{F}_f \subseteq_f \mathcal{F}$.*
3. *$\mathcal{P}$ est* saturé *si pour tout $\mathcal{F} \in \mathcal{P}$ et pour toute instance*

$$\frac{H_1, \ldots, H_l}{C^1_1, \ldots, C^1_{p1} \mid \ldots \mid C^k_1, \ldots, C^k_{pk}}$$

*de règle de la Figure 5.1* ou de règle de la Définition 5.17*, si $H_i \in \mathcal{F}$ pour tout $i \in \{1, \ldots, l\}$, alors $\mathcal{F} \cup \{C^j_1, \ldots, C^j_{p_j}\} \in \mathcal{P}$ pour au moins un $j \in \{1, \ldots, k\}$.*

La définition d'un oracle pour un HBBI-tableau est alors :

**Définition 5.20** (Oracle pour les HBBI tableaux)**.** *Un* oracle *est un ensemble de SHS non clos qui est $\subseteq$-clos, de caractère fini et saturé selon la Définition 5.19.*

**Lemme 5.15.** *Il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe aucun HBBI-tableau clos.*

*Démonstration.*
La preuve de ce lemme est une variante de celle du Lemme 5.12. Elle est donnée en Annexe B. □

Pour montrer la complétude, on considère une formule $\phi$ pour laquelle il n'existe pas de preuve et on montre qu'il en existe un contre-modèle.

On note $\mathcal{T}_0$ le tableau initial pour $\phi$. On a :

1. $\mathcal{T}_0 = [\{\mathbb{F}\ @_{c_1}(\phi)\}]$

2. $\mathcal{T}_0$ ne peut pas être clos

Par le Lemme 5.15 il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe pas de tableau clos. On note $\mathcal{P}$ cet oracle. Par la Proposition 5.1 il existe une stratégie équitable. On la note $\mathcal{S}$ et on note $\mathcal{S}_i = (F_i, n_i)$ la $i^{\text{ème}}$ formule de $\mathcal{S}$. Comme $\mathcal{T}_0$ ne peut être clos, son unique branche appartient à l'oracle, c'est à dire $\{\mathbb{F}\ @_{c_1}(\phi)\} \in \mathcal{P}$.

On construit maintenant la suite $\mathcal{F}_{i\geq 0}$ de la même manière que dans la section 5.6.4 à partir de l'oracle $\mathcal{P}$.

**Proposition 5.4.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$
2. $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_i$
3. $(\mathcal{F}_i)_{i\geq 0} \in \mathcal{P}$
4. $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$

*Démonstration.*
La preuve est identique à celle de la Proposition 5.2. □

On considère maintenant le SHS limite $\mathcal{F}_\infty$ défini par :

$$\mathcal{F}_\infty = \bigcup_{i\geq 0} \mathcal{F}_i$$

**Proposition 5.5.** *Les propriétés suivantes sont vraies :*

1. $\mathcal{F}_\infty \in \mathcal{P}$
2. *Pour toute formule étiquetée $\mathbb{S}\ @_x(\phi)$, si $\mathcal{F}_\infty \cup \{\mathbb{S}\ @_x(\phi)\} \in \mathcal{P}$, alors $\mathbb{S}\ @_x(\phi) \in \mathcal{F}_\infty$*

*Démonstration.*
La preuve est identique à celle de la Proposition 5.3. □

**Lemme 5.16.** *Le SHS limite est un SHS de Hintikka.*

*Démonstration.*
On vérifie toutes les conditions de la Définition 5.18 :

31. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle BI_n \rangle$, $\mathcal{F}_\infty \cup \mathbb{T}\ @_x(x * \mathrm{I}) \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{T}\ @_x(x * \mathrm{I}) \in \mathcal{F}_\infty$.
32. On suppose que $\mathbb{T}\ @_x(y * z) \in \mathcal{F}_\infty$. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle BI_c \rangle$, on a $\mathcal{F}_\infty \cup \{\mathbb{T}\ @_x(z * y)\} \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{T}\ @_x(z * y) \in \mathcal{F}_\infty$.
33. On suppose que $\mathbb{T}\ @_x(y * (z * t)) \in \mathcal{F}_\infty$. D'après le point 1 de la Proposition 5.3, $\mathcal{F}_\infty \in \mathcal{P}$ et par construction $\mathcal{P}$ est saturé, donc par saturation par la règle $\langle BI_c \rangle$, on a $\mathcal{F}_\infty \cup \{\mathbb{T}\ @_x((y * z) * t)\} \in \mathcal{P}$. Par le point 2. de la Proposition 5.3, on a donc $\mathbb{T}\ @_x((y * z) * t) \in \mathcal{F}_\infty$.

Les autres cas sont ceux du Lemme 5.13. □

Enfin on a :

**Théorème 5.6** (Complétude des HBBI-tableaux)**.** *Soit* $\phi$ *une formule de HBBI. Si* $\phi$ *est valide, alors il existe une HBBI-preuve pour* $\phi$*.*

*Démonstration.*
On suppose que $\phi$ est valide. On suppose qu'il n'y a pas de preuve pour $\phi$. La méthode que nous avons présentée permet de construire un SHS limite $\mathcal{F}_\infty$. Par le Lemme 5.16, $\mathcal{F}_\infty$ est un SHS de Hintikka. Par le point 2. de la Proposition 5.4, $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_i$ pour tout $i$, c'est à dire que $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_\infty$ par construction. Alors, d'après le Lemme 5.11, $\phi$ n'est pas valide. C'est absurde. Donc il existe une preuve pour $\phi$. □

On a en outre le résultat suivant pour les formules de BBI :

**Corollaire 5.2.** *Soit* $\phi$ *une formule de* $\mathcal{L}_{BBI}$*. Si* $\phi$ *est valide dans BBI, alors il existe une HBBI-preuve pour* $\phi$*.*

*Démonstration.* Comme $\phi$ est valide dans BBI, par le Théorème 5.2, $\phi$ est valide dans HBBI. Alors par le Théorème 5.6, $\phi$ a une HBBI-preuve. □

Nous avons donc montré la correction et la complétude du calcul des HBBI-tableaux.

Nous avons expliqué dans la section 5.7.1 comment nous pouvons étendre le calcul des HRL-tableaux pour toute axiomatique supplémentaire. La question se pose donc de prouver la correction et la complétude systématiquement pour chacun de ces nouveaux calculs.

L'étude présente de la correction et complétude pour les HBBI-tableaux semble indiquer que la correction d'une nouvelle variante de HRL ne serait pas difficile à prouver. En effet il s'agit essentiellement de vérifier que les nouvelles règles préservent toujours la réalisabilité des branches.

Quant à la complétude, la preuve est aussi assez similaire au cas général si les règles sont bien énoncées (c'est à dire qu'elle traduisent effectivement les axiomes). Toutefois, on note que dans le cas étudié (celui de HBBI) les règles axiomatiques n'introduisent pas de nouveaux nominaux, ce qui permet d'utiliser telle quelle la construction de SHS limite énoncé pour HRL dans la section 5.6.4. Si l'une des nouvelles règles introduit de nouveaux nominaux, il faudra sûrement modifier la méthode de construction de la branche limite et donc modifier le schéma de preuve utilisé pour HBBI.

# Conclusion

En continuant les travaux sur la logique BBI hybride de [20], nous avons donc introduit une nouvelle forme de logique, HRL, qui reprend les bases (classiques) des logiques de séparation, mais sans imposer aucune propriétés sur la composition entre ressources. En revanche, HRL autorise l'internalisation des ressources, utilisant les symboles de ressources comme des symboles propositionnels.

Cette manière de construire les logiques de séparation autorise deux choses : tout d'abord, elle étend l'expressivité de ces logiques en autorisant de construire des formules exprimant des propriétés à la fois sur les ressources et leurs propriétés.

En outre, l'introduction de ressources dans la syntaxe permet de construire des axiomes qui apportent des propriétés aux ressources. Par exemple, la logique HBBI que nous avons proposé est une extension de HRL équivalente à BBI, mais l'ajout ou l'omission d'axiomes peut mener à d'autres logiques , comme une version non commutative de BBI ou une variante avec l'introduction de ressources inversibles.

Enfin, HRL est munie d'un calcul des tableaux, correct et complet, qui peut être étendu pour ces variantes axiomatiques, ce qui en fait un outil générique pour de nombreuses variantes de BBI. On montre dans le chapitre suivant (Chapitre 6) que ce calcul est équivalent au calcul pour BBI que nous avons présenté dans le Chapitre 2.

Il serait intéressant de développer une version intuitionniste de HRL, en remplaçant la sémantique des nominaux par une notion de pré-ordre au lieu d'une équivalence. Cela permettrait de modéliser BI et ses variantes comme nous l'avons fait ici pour BBI. En outre, il doit être possible d'exprimer les nominaux à la fois avec leur sémantique classique et leur sémantique intuitionniste, ce qui permettrait de concevoir une logique complètement générale, permettant de retrouver les sémantiques de BI et de BBI par différents axiomes.

Enfin, la combinaison de HRL avec d'autres extensions (modales, temporelles, ...) pourrait permettre une expressivité renouvelée des logiques de séparation. Nous développons ce point plus en avant dans le chapitre de conclusion (Chapitre 7).

# Chapitre 6

# BBI-tableaux et HBBI-tableaux

Dans ce chapitre, nous souhaitons étudier les relations entre le calcul des tableaux pour HBBI que nous avons introduit dans le Chapitre 5 et le calcul des tableaux pour BBI tel que présenté dans le Chapitre 2.

Nous avons déjà noté que notre calcul des tableaux, bien qu'inspiré des calculs des tableaux à labels de BI et BBI [49,66], présente la particularité de ne pas utiliser à proprement parler de labels externes, utilisant à la place les opérateurs de HRL et les nominaux. Toutefois, on ne peut s'empêcher de noter les grandes similarités entre le calcul des HBBI-tableaux présenté dans ce chapitre, et le calcul traditionnel des BBI-tableaux, comme nous l'avons rappelé dans le Chapitre 2.

Cette dichotomie et pourtant proximité entre les deux calculs, avec et sans labels, fait également écho aux discussions quant à la pertinence de l'utilisation des labels dans des calculs de tableaux en général. L'utilisation de labels dans un calcul des tableaux apporte une certaine lisibilité (étant donné que les labels représentent directement les structures de Kripke de la sémantique) et il a été montré dans certains contextes qu'ils améliorent également la complexité des calculs [56]. Toutefois, les calculs sans labels, comme le calcul des séquents, sont souvent considérés comme plus purs que ceux qui en utilisent, étant donné que les labels sont des structures supplémentaires non liés à la logique elle-même.

Notons en outre que pour ce qui est des logiques hybrides, les deux systèmes coexistent. En effet, si le calcul des tableaux d'origine pour la logique hybride est similaire au nôtre (il utilise l'opérateur @ à la place de labels [15]), il existe également des calculs des tableaux avec préfixes, une méthode extrêmement similaire à l'usage de labels [17].

Il paraît ainsi intéressant, d'étudier la différence entre notre calcul des HBBI-tableaux et la calcul des BBI-tableaux. Les deux étant équivalents, puisqu'ils sont tous deux corrects et complets vis-à-vis de la sémantique de BBI, on peut légitimement se demander quelles sont les différences. Pour cela, on se propose d'étudier la traduction un HBBI-tableau en un BBI-tableau et vice-versa.

## 6.1 Des HBBI-tableaux vers les BBI-tableaux

La forme des HBBI-tableaux et des BBI-tableaux étant proches, le principe de la traduction de l'un vers l'autre est assez simple : il s'agit de transformer les formules de HBBI marquant des

contraintes entre ressources en contraintes de ressources de la méthode de BBI ou vice-versa. La traduction des autres formules est directe.

En premier lieu, nous devons classifier les HBBI-tableaux pour distinguer les formules correspondant à des formules de BBI et celles correspondant à des contraintes de ressources.

### 6.1.1 Classification des HBBI-tableaux

Si HBBI permet de retrouver l'expressivité de BBI, il faut remarquer que la syntaxe de HBBI contient également des nominaux et les opérateurs @ et $*\!\!-$, hérités de HRL. Il faut donc caractériser les tableaux qui raisonnent uniquement sur les formules communes à BBI et HBBI.

**Définition 6.1** (Formules et tableaux purs)**.** *Une formule de HBBI est dite* pure *si elle ne contient pas de nominaux, ni d'opérateur @, ni d'opérateur $*\!\!-$.*

*Un HBBI-tableau est* pur *s'il est construit à partir d'une formule pure et si les règle $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}-\!\!* \rangle$ ne sont jamais appliquées sur des formules qui ne contiennent que des nominaux.*

Une formule pure est donc aussi une formule de BBI. Rappelons que dans HBBI, $*\!\!-$ est inutile (voir le Lemme 5.6). Notons qu'un tableau pur ne contient pas nécessairement que des formules pures. En effet, l'application des règles $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}-\!\!* \rangle$ permettent de générer des formules non pures.

La seconde contrainte, forçant l'application des règles $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}-\!\!* \rangle$ sur des formules qui contiennent au moins un symbole propositionnel, ne change rien au tableau. En effet, appliquées sur des formules qui ne contiennent que des nominaux n'apporte aucune information puisqu'elles ne font que remplacer un ou plusieurs nominaux par d'autres nominaux équivalents. C'est ainsi une manière de minimaliser la taille du tableau pour une formule (même si d'autres optimisations sont certainement possibles).

Dans la suite, l'opérateur atomique I sera indifféremment considéré comme une proposition et comme un nominal. En effet, I et le nominal $e$ sont complètement interchangeables dans HBBI puisqu'ils ont la même sémantique. Ainsi, une formule composée exclusivement de nominaux et de I sera considérée comme entièrement nominale, alors qu'une formule pure ne contenant que I comme nominal sera toujours pure.

Pour simplifier notre propos, enfin, on classifiera les règles du calcul de HBBI en trois catégories distinctes.

**Définition 6.2** (Classification des règles)**.** *Les règles $\langle \mathbb{T}*\!\!- \rangle$, $\langle \mathbb{F}*\!\!- \rangle$ et $\langle @ \rangle$ sont appelées* règles impropres*. Les règles $\langle \mathbb{T}\mathrm{I} \rangle$, $\langle i_r \rangle$, $\langle i_s \rangle$, $\langle i_t \rangle$, $\langle i+ \rangle$, $\langle i- \rangle$, $\langle i_p \rangle$ ainsi que les règles propres à HBBI c'est à dire $\langle BI_n \rangle$, $\langle BI_c \rangle$ et $\langle BI_a \rangle$ sont appelées* règles à labels*. Enfin les autres règles sont appelées* règles pures*.*

Les règles impropres correspondent aux manipulations des formules n'apparaissant pas dans notre étude (c'est à dire ni pures, ni produites par la décomposition de formules pures).

Les règles pures sont les règles de manipulation des formules de BBI et correspondent donc justement aux règles portant le même nom du calcul des tableaux de BBI.

Les règles à labels manipulent des nominaux et correspondent donc aux règles sur les contraintes du calcul des tableaux de BBI, tels que donnés dans la définition des clôtures de contraintes pour ce calcul.

On veut maintenant plus précisément caractériser le fait que les formules non pures apparaissant dans un HBBI-tableau sont en réalité composées de nominaux et font écho aux contraintes de labels du calcul de BBI. En outre, on veut montrer que chaque formule de ce type est obtenue par l'enchaînement de règles à labels de la même manière que les contraintes sont produites par clôtures de leur ensemble.

Pour un HBBI-tableau $\mathcal{T}$ on note $\mathcal{R}(\mathcal{T}) = \{R_1, R_2, \dots R_n\}$ l'ensemble (possiblement infini) des règles appliquées à un SHS singleton pour obtenir $\mathcal{T}$. L'ordre des règles dans $\mathcal{R}(\mathcal{T})$ est celui dans lequel les règles ont été appliquées pour créer $\mathcal{T}$.

**Lemme 6.1.** *Soit $\mathcal{T}$ un HBBI-tableau pur. Soit $F = \mathbb{S}\ @_x(\phi)$ une formule de $\mathcal{T}$ telle que $\phi$ n'est pas pure. On a alors :*

1. *$\phi$ est composée uniquement de nominaux $\{e, c_1, \dots, c_n\}$ et d'occurrences des opérateurs $*$ et* I, *$\mathbb{S} = \mathbb{T}$.*
2. *Il existe un ensemble de formules étiquetées $\{F_0, \dots, F_k\}$ et un ensemble de règles $\mathcal{R}_F \subseteq \mathcal{R}(\mathcal{T})\}$ tels que :*
   - *les règles de $\mathcal{R}_F$ sont toutes des règles à labels ;*
   - *les formules $\{F_0, \dots, F_k\}$ sont toutes non pures ;*
   - *les formules $\{F_0, \dots, F_k\}$ sont toutes obtenues par l'application des règles $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F} -\!\!* \rangle$ à des formules pures ;*
   - *il existe un arbre de preuve $\mathcal{T}_F \subseteq \mathcal{T}$, composé des règles de $\mathcal{R}_F$, ayant pour axiomes les formules $\{F_0, \dots, F_k\}$ et générant à la fin la formule $F$.*

*Démonstration.*
On démontre les deux résultats par récurrence sur la taille de $\mathcal{T}$.

- Cas de base
  Soit $\mathcal{T}$ un HBBI-tableau pur de taille 1. Comme $\mathcal{T}$ est pur, c'est un tableau pour une formule pure. Comme il est de taille 1, il est donc composé d'une unique formule étiquetée dont la formule est pure. Alors $\mathcal{T}$ ne contient pas de formules non pures et aucun $F$ ne convient. Le lemme est donc vrai pour tous les tableaux purs de taille 1.
- Hérédité
  On suppose que le lemme est vrai pour tous les tableaux purs de taille $n$ (HR), on va montrer qu'il est vrai pour tout les tableaux purs de taille $n+1$.

  Soit $\mathcal{T}$ un tableau pur de taille $n+1$. Il a été obtenu par l'application d'une règle $R$ sur un tableau pur $\mathcal{T}'$ de taille $n$.

  Soit $F = \mathbb{S}\ @_x(\phi)$ une formule apparaissant dans $\mathcal{T}$ telle que $\phi$ n'est pas pure. Si $F$ n'a pas été introduite par $R$, alors $F \in \mathcal{T}'$ et on a 1. et 2. par (HR).

  Supposons maintenant que $F$ est introduite par $R$. $R$ a été appliquée à partir d'une ou plusieurs formules de $\mathcal{T}'$. Par (HR), ces formules sont soit pures, soit composées uniquement de nominaux et de l'opérateur $*$. On discute selon la nature de $R$ :
  - Si $R = \langle @ \rangle$, alors $R$ a été appliquée sur une formule non pure contenant un $@$, ce cas n'est donc pas possible.
  - Si $R = \langle \mathbb{T} -\!\!* \rangle$ ou $R = \langle \mathbb{F} -\!\!* \rangle$, alors $R$ a été a été appliquée sur une formule non pure contenant $-\!\!*$, ce cas n'est donc pas possible.

- Si $R = \langle \mathbb{T}\mathrm{I} \rangle$, $F = \mathbb{T}\ @_e(\mathrm{I})$. Alors le point 1. est vrai. En outre, le point 2. est vrai pour $\mathcal{R}_F = \{R\}$ et un ensemble de formules vides.
- Si $R = \langle \mathbb{T}* \rangle$, soit $\psi = A * B$ la formule sur laquelle la règle a été appliquée. Par (HR), $\psi$ est soit pure, soit composée exclusivement de nominaux et d'opérateurs $*$. Comme $\mathcal{T}$ est pur, le second cas n'est pas possible. Donc $\psi$ et pure, et par suite $A$ et $B$ le sont. Ainsi, les formules étiquetées $\mathbb{T}\ @_{c_i}(A)$ et $\mathbb{T}\ @_{c_j}(B)$ sont pures. On a donc nécessairement $F = \mathbb{T}\ @_x(c_i * c_j)$ où $c_i$ et $c_j$ sont de nouveaux labels. Alors le point 1. est vrai. En outre, le point 2. est vrai pour un ensemble vide de règles $\mathcal{R}_F = \emptyset$ et l'ensemble de formules $\{F\}$.
- Si $R = \langle \mathbb{F}{-\!\!*} \rangle$, soit $\psi = A -\!\!* B$ la formule sur laquelle la règle a été appliquée. Par (HR), $\psi$ est pure, et par suite $A$ et $B$ le sont. Ainsi, les formules étiquetées $\mathbb{T}\ @_{c_i}(A)$ et $\mathbb{F}\ @_{c_j}(B)$ sont pures. On a donc nécessairement $F = \mathbb{T}\ @_{c_j}(x * c_i)$ où $c_i$ et $c_j$ sont de nouveaux labels. Alors le point 1. est vrai. En outre, le point 2. est vrai pour un ensemble vide de règles $\mathcal{R}_F = \emptyset$ et l'ensemble de formules $\{F\}$.
- Si $R$ est une autre des règles pures, par le point 1. de (HR), la formule sur laquelle la règle est appliquée est pure, donc les deux formules produites sont également pures, donc $F$ ne peut pas avoir été produite par ces règles.
- Si $R = \langle i_r \rangle$, alors le point 1. est vérifié, et le point 2. tient pour un ensemble de formules vides et $\mathcal{R}_F = \{R\}$.
- Si $R = \langle i_s \rangle$, le point 1. est vérifié. Soit $F'$ la prémisse de $R$. Par (HR), il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F'$. Alors $\{F_0, \ldots F_k\}$ et $\mathcal{R}_{F'} \cup \{R\}$ vérifient le point 2. pour $\mathcal{T}$.
- Si $R = \langle i_t \rangle$, on a deux cas :
  a) Si $R$ est appliquée sur des formules étiquetées $F' = \mathbb{T}\ @_x(\psi)$ et $F'' = \mathbb{T}\ @_y(x)$, $F'$ peut être non pure, mais par (HR) elle est de la forme requise par le point 1. et la conclusion est donc aussi de la forme requise. Le point 1. est vérifié. Par (HR), il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F'$. Il existe aussi un ensemble de formules $\{F'_0, \ldots F'_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F''}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F''$. Alors $\{F_0, \ldots F_k\} \cup \{F'_0, \ldots F'_k\}$ et $\mathcal{R}_{F'} \cup \mathcal{R}_{F''} \cup \{R\}$ vérifient le point 2 .pour $\mathcal{T}$.
  b) Dans les autres cas, le prémisse de gauche doit être pure, et la formule produite aussi par conséquent. Donc $F$ ne peut être produite par cette règle.
- Si $R = \langle i{+} \rangle$, d'après le point 1. de (HR), la règle doit être appliquée sur une formule étiquetée $F' = \mathbb{T}\ @_x(\psi)$ et ne contenant que des nominaux et des opérateurs $*$. Alors les deux formules produites vérifient 1. En outre, par (HR), il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F'$. Alors $\{F_0, \ldots F_k\}$ et $\mathcal{R}_{F'} \cup \{R\}$ vérifient le point 2. pour $\mathcal{T}$, quelque soit la conclusion choisie pour $F$.

- Si $R = \langle i- \rangle$, par (HR) $R$ est appliquée sur des formules étiquetées $F' = \mathbb{T}\ @_x(\psi[y])$ et $F'' = \mathbb{T}\ @_y(z * t)$. Alors le point 1. est vérifié. En outre, par l'hypothèse de récurrence, il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F'$. Il existe aussi un ensemble de formules $\{F'_0, \ldots F'_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F''}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F''$. Alors $\{F_0, \ldots F_k\} \cup \{F'_0, \ldots F'_k\}$ et $\mathcal{R}_{F'} \cup \mathcal{R}_{F''} \cup \{R\}$ vérifient le point 2. pour $\mathcal{T}$.
- Si $R = \langle i_p \rangle$, par (HR) $R$ est appliquée sur des formules étiquetées $F' = \mathbb{T}\ @_x(\psi[y])$ et $F'' = \mathbb{T}\ @_y(z)$. Alors le point 1. est vérifié. En outre, par (HR), il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F'$. Il existe aussi un ensemble de formules $\{F'_0, \ldots F'_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F''}$ vérifiant le point 2. et permettant de construire un tableau amenant à $F''$. Alors $\{F_0, \ldots F_k\} \cup \{F'_0, \ldots F'_k\}$ et $\mathcal{R}_{F'} \cup \mathcal{R}_{F''} \cup \{R\}$ vérifient le point 2. pour $\mathcal{T}$.
- Si $R = \langle BI_n \rangle$, alors la nouvelle formule introduite, $\mathbb{T}\ @_x(x * \mathrm{I})$, est composée uniquement de nominaux puisque I est à la fois un nominal et une proposition (comme indiqué au début de cette section). Alors le point 1. et vérifié et le point 2. tient pour un ensemble de formules vides et $\mathcal{R}_F = \{R\}$.
- Si $R = \langle BI_c \rangle$ ou $R = \langle BI_a \rangle$, le point 1. est vrai étant donné la forme de ces deux règles. En outre, par (HR), il existe un ensemble de formules $\{F_0, \ldots F_k\}$ et un ensemble de règles à labels $\mathcal{R}_{F'}$ vérifiant le point 2. et permettant de construire un tableau amenant à la prémisse $F'$ de $R$. Alors $\{F_0, \ldots F_k\}$ et $\mathcal{R}_{F'} \cup \{R\}$ vérifient le point 2. pour $\mathcal{T}$.

□

### 6.1.2 Labellisation

On a donc restreint les formules de nos HBBI-tableaux purs aux formules pures et aux ensembles de nominaux joints par des $*$. On montre maintenant leur correspondance respective avec les formules et les contraintes du calcul de BBI.

Formalisons la transformation d'une formule étiquetée de HBBI en une formule ou une contrainte de BBI :

**Définition 6.3** (Labellisation d'une HBBI-formule)**.** *Soit $F = \mathbb{S}\ @_x(\phi)$ une formule étiquetée de HBBI. La labellisation $\lambda(F)$ de $F$ est définie par :*

- *Si $\phi$ est pure, $\lambda(F) = \mathbb{S}\phi : x$*
- *Si $\mathbb{S} = \mathbb{T}$ et $\phi$ est composée uniquement de nominaux, d'opérateurs* I *et $*$, alors $\lambda(F) = x \simeq l(\phi)$ où $l(\phi)$ est défini récursivement par :*
  - *$l(\mathrm{I}) = l(e) = \varepsilon$, le mot vide*
  - *$l(i) = i$ pour tout autre nominal*
  - *$l(A * B) = l(A)l(B)$*
- *Dans tous les autres cas, $\lambda(F)$ est indéfinie.*

On peut maintenant généraliser ce processus pour transformer des SHS de HBBI en CSS de BBI. On sera dans cette procédure attentif aux nominaux introduits par la règle $\langle i+\rangle$. En effet, ces nominaux sont simplement des outils pour substituer à un nominal un couple de nominaux qui le composent, permettant ainsi de conserver des nominaux simples dans la partie gauche des formules. Il n'y a pas d'équivalent dans BBI, les labels composés y étant utilisés indifféremment des simples. On doit donc substituer à tout nominal de cette forme le couple de nominal qu'il représente, et répéter éventuellement la procédure dans le cas de compositions complexes.

**Définition 6.4** (Labellisation d'un HBBI-tableau)**.** *Soit $\mathcal{T}$ un HBBI-tableau pur. Soit $\mathcal{F} = \{F_1, \ldots, F_n\}$ une branche de $\mathcal{T}$. La* labellisation *de $\mathcal{F}$, notée $\lambda(\mathcal{F})$ est un couple d'ensembles $\lambda(\mathcal{F}) = \langle \mathcal{F}_\mathcal{F}, \mathcal{C}_f \rangle$ construite de la manière suivante :*

- *Si $F_i = \mathbb{S}\ @_x(\phi)$ est une formule pure, alors $\lambda(F_i) \in \mathcal{F}_\mathcal{F}$ et, dans le cas du CSS initial, $x \simeq x \in \mathcal{C}_\mathcal{F}$ ;*
- *Si $F_i$ est une formule contenant uniquement des nominaux et qu'elle a été introduite par une application des règles $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F}{-\!\!*} \rangle$, alors $\lambda(F_i) \in \mathcal{C}_\mathcal{F}$ ;*
- *Dans ces deux cas, si $F_i$ contient un nominal $c_i$ n'étant pas introduit par une règle $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F}{-\!\!*} \rangle$, alors il existe une occurrence de la règle $\langle i+\rangle$ qui a introduit $c_i$, et on remplace syntaxiquement tout occurrence de $c_i$ par l'expression $yz$ où $y$ et $z$ sont les labels apparaissant dans la règle $\langle i+\rangle$ correspondante. On répète le processus jusqu'à que tous les nominaux de $c_i$ proviennent de $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F}{-\!\!*} \rangle$.*
- *Dans tous les autres cas, $F_i$ n'influe pas sur $\lambda(\mathcal{F})$.*

*La* labellisation *$\lambda(\mathcal{T})$ de $\mathcal{T}$ est la liste de tous les $\lambda(\mathcal{F})$ pour tous les $\mathcal{F} \in \mathcal{T}$.*

On a donc obtenu ce qui semble être un BBI-tableau à partir d'un HBBI-tableau. Il nous faut toutefois démontrer que c'est bien le cas. Notez que les formules non pures introduites par des règles à labels, pourtant cruciales dans les tableaux de HBBI, sont ignorées dans cette traduction. Ce n'est en effet pas nécessaire de les traduire puisque la clôture de contraintes permet en réalité de reconstituer leur image. On va tout d'abord montrer ce résultat.

**Lemme 6.2.** *Soit $\mathcal{T}$ un HBBI-tableau pur. Soit $\mathcal{F}$ une branche de $\mathcal{T}$ et $F$ une formule non pure de $\mathcal{F}$. Alors on a $\lambda(F) \in \overline{\mathcal{C}_\mathcal{F}}$, à substitution près des nominaux comme décrit dans la Définition 6.4, où $\overline{\mathcal{C}_\mathcal{F}}$ est la clôture de l'ensemble $\mathcal{C}_\mathcal{F}$ par les règles de clôture de contraintes de labels du calcul des tableaux de BBI.*

*Démonstration.*
Comme $F$ est non pure, d'après le Lemme 6.1, $F$ est composée exclusivement de nominaux et d'opérateurs $*$. En outre, il existe un arbre de preuve $\mathcal{T}_F$ composé de règles à labels et de formules non pures issues de $\langle \mathbb{T}* \rangle$ et $\langle \mathbb{F}{-\!\!*} \rangle$.

On raisonne par récurrence sur la taille de l'arbre $\mathcal{T}_F$.

- Cas de base
  Si $\mathcal{T}_F$ est de taille 0, alors c'est que $F$ est elle-même obtenue par l'application d'une règle $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F}{-\!\!*} \rangle$. Alors par la Définition 6.4, $\lambda(F) \in \mathcal{C}_\mathcal{F}$. Or par définition de la clôture, $\mathcal{C}_\mathcal{F} \subseteq \overline{\mathcal{C}_\mathcal{F}}$, donc la propriété est vérifiée.

- Hérédité
  On suppose que la propriété est vraie pour toutes formules $F$ ayant un arbre $\mathcal{T}_F$ de taille inférieure ou égale à $n$ (HR). On montre qu'elle est vraie pour les formules ayant des arbres de taille $n+1$.

  Soit $F$ une formule non pure de $\mathcal{F}$ ayant un arbre $\mathcal{T}_F$ de taille $n+1$. Il existe alors une règle à labels $R$ telle que $F$ est obtenue à partir de $R$ et d'un arbre $\mathcal{T}'_F$ de taille $n$. On discute selon la nature de $R$.

  - Si $R = \langle \mathbb{T}\mathrm{I} \rangle$. Alors $F = \mathbb{T}\ @_e(I)$. On a donc $\lambda(F) = \varepsilon \simeq \varepsilon$, et $\varepsilon \simeq \varepsilon \in \overline{\mathcal{C}_\mathcal{F}}$ par application de la règle $\langle \varepsilon \rangle$.
  - Si $R = \langle i_r \rangle$. Alors $F = \mathbb{T}\ @_x(x)$ et $x$ apparaît déjà dans $\mathcal{T}_F$. On a donc $\lambda(F) = x \simeq x$ et $x \simeq x \in \overline{\mathcal{C}_\mathcal{F}}$ par application d'une des règles $\langle d_r \rangle$ ou $\langle p_l \rangle$ ou $\langle p_r \rangle$.
  - Si $R = \langle i_s \rangle$. Alors $F = \mathbb{T}\ @_y(x)$ et $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(y)$. Par (HR), on a alors $x \simeq y \in \overline{\mathcal{C}_F}$. Alors par application de la règle $\langle s_r \rangle$ on a $y \simeq x \in \overline{\mathcal{C}_F}$, c'est à dire $\lambda(F) \in \overline{\mathcal{C}_F}$.
  - Si $R = \langle i_t \rangle$. Alors $F = \mathbb{T}\ @_y(\phi)$ (où $\phi$ est à nouveau une formule composée de nominaux et de $*$) et $\mathcal{T}'_F$ a produit les formules $\mathbb{T}\ @_y(x)$ et $\mathbb{T}\ @_x(\phi)$. Par (HR), on a alors $y \simeq x \in \overline{\mathcal{C}_F}$ et $x \simeq l(\phi) \in \overline{\mathcal{C}_F}$. Alors par application de la règle $\langle t_r \rangle$ on a $y \simeq l(\phi) \in \overline{\mathcal{C}_F}$ c'est à dire $\lambda(F) \in \overline{\mathcal{C}_F}$.
  - Si $R = \langle i+ \rangle$, alors $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(\phi[y*z])$ où $\phi$ est à nouveau une formule composée de nominaux et de $*$. On a deux cas :
    - $F = \mathbb{T}\ @_x(\phi[y*z/c_i])$. Alors $c_i$ est produit par une règle $\langle i+ \rangle$ et est donc remplacé par substitution dans $\lambda(F)$ par $yz$. Or par $\lambda$, $y*z$ est justement transformé en $yz$. Ainsi $\lambda(\mathbb{T}\ @_x(\phi[y*z])) = \lambda(\mathbb{T}\ @_x(\phi[y*z/c_i]))$ à cause de cette substitution. Or, par hypothèse de récurrence on a $\lambda(\mathbb{T}\ @_x(\phi[y*z])) \in \overline{\mathcal{C}_F}$, donc $\lambda(F) \in \overline{\mathcal{C}_F}$.
    - $F = \mathbb{T}\ @_{c_i}(y*z)$. Alors $\lambda(F) = c_i \simeq yz = yz \simeq yz$ par substitution. Or $yz \simeq yz \in \overline{\mathcal{C}_\mathcal{F}}$ par application de la règle $\langle d_r \rangle$, donc $\lambda(F) \in \overline{\mathcal{C}_F}$.
  - Si $R = \langle i- \rangle$, alors $F = \mathbb{T}\ @_x(\phi[y/z*t])$ et $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(\phi[y])$ et $\mathbb{T}\ @_y(z*t)$. Alors par (HR) on a $x \simeq l(\phi)[y] \in \overline{\mathcal{C}_\mathcal{F}}$ et $y \simeq zt \in \overline{\mathcal{C}_\mathcal{F}}$. Alors par applications successives des règles $\langle d_r \rangle$, $\langle c_r \rangle$ et $\langle t_r \rangle$, on peut obtenir $x \simeq l(\phi)[y/zt] \in \overline{\mathcal{C}_\mathcal{F}}$, c'est à dire $\lambda(\phi) \in \overline{\mathcal{C}_\mathcal{F}}$.
  - Si $R = \langle i_p \rangle$, alors $F = \mathbb{T}\ @_x(\phi[y/z])$ et $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(\phi[y])$ et $\mathbb{T}\ @_y(z)$. Alors par (HR) on a $x \simeq l(\phi)[y] \in \overline{\mathcal{C}_\mathcal{F}}$ et $y \simeq z \in \overline{\mathcal{C}_\mathcal{F}}$. Alors par applications successives des règles $\langle d_r \rangle$, $\langle c_r \rangle$ et $\langle t_r \rangle$, on peut obtenir $x \simeq l(\phi)[y/z] \in \overline{\mathcal{C}_\mathcal{F}}$, c'est à dire $\lambda(\phi) \in \overline{\mathcal{C}_\mathcal{F}}$.
  - Si $R = \langle BI_n \rangle$, alors $F = \mathbb{T}\ @_x(x*\mathrm{I})$. On a alors $\lambda(F) = x \simeq x\varepsilon = x \simeq x$ car $\varepsilon$ est le mot vide. On se ramène alors au cas $\langle i_r \rangle$.
  - Si $R = \langle BI_c \rangle$, alors $F = \mathbb{T}\ @_x(z*y)$ et $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(y*z)$. Par (HR) on a $x \simeq yz \in \overline{\mathcal{C}_\mathcal{F}}$. Comme les éléments de $\mathcal{C}_\mathcal{F}$ sont des mots sur les labels (c'est à dire des multi-ensembles non ordonnés), on a donc $yz = zy$ et donc aussi $x \simeq zy \in \overline{\mathcal{C}_\mathcal{F}}$, c'est à dire $\lambda(\phi) \in \overline{\mathcal{C}_\mathcal{F}}$.

- Si $R = \langle BI_c \rangle$, alors $F = \mathbb{T}\ @_x((y*z)*t)$ et $\mathcal{T}'_F$ a produit $\mathbb{T}\ @_x(y*(z*t))$. Par (HR) on a $x \simeq y(zt) \in \overline{\mathcal{C}_\mathcal{F}}$. Comme les éléments de $\mathcal{C}_\mathcal{F}$ sont des mots sur les labels (c'est à dire des multi-ensembles non ordonnés), on a donc $y(zt) = (yz)t$ et donc aussi $x \simeq (yz)t \in \overline{\mathcal{C}_\mathcal{F}}$, c'est à dire $\lambda(\phi) \in \overline{\mathcal{C}_\mathcal{F}}$.

Finalement la propriété est héréditaire, donc vraie quelque soit la taille de $\mathcal{T}_F$.

□

Avec ce résultat on peut maintenant montrer que notre transformation produit bien un BBI-tableau et que la clôture du tableau est transférée par cette transformation.

**Théorème 6.1.** *[Propriétés des tableaux labellisés] Soit $\mathcal{T}$ un HBBI-tableau pur.*

1. *$\lambda(\mathcal{T})$ est un BBI-tableau.*
2. *si $\mathcal{T}$ est clos, $\lambda(\mathcal{T})$ l'est aussi.*
3. *si $\lambda(\mathcal{T})$ est clos, $\mathcal{T}$ peut être étendu (par les règles du calcul pour HBBI) en un tableau clos.*

*Démonstration.*

La preuve de ce théorème est donnée dans l'Annexe C.

□

On a ainsi vu que par une simple transformation, presque directe, on pouvait retrouver le BBI-tableau correspondant à un HBBI-tableau.

### 6.1.3 Un exemple de labellisation (d'un HBBI-tableau vers un BBI-tableau)

En guise d'exemple de traduction, on se propose de traduire en BBI-tableau l'exemple de HBBI que nous avions traité dans la section 5.5. Si nous regardons la Définition 6.4, il s'agit simplement de traduire par la transformation de la Définition 6.3 les formules pures et les formules non-pures introduites par certaines règles.

Regardons comment cela se traduit sur notre exemple. La Figure 6.1 présente le HBBI-tableau et la traduction des formules étiquetées du tableau.

La formule 1 étant la racine de notre tableau, elle génère non seulement une formule étiquetée, mais aussi la contrainte $c_1 \simeq c_1$. Les autres formules pures (2,3,4,5,7,8,14,15,16 et 17) sont traduites directement. Les formules non pures sont traduites seulement si elles sont issues des règles $\langle \mathbb{T}* \rangle$ ou $\langle \mathbb{F}-\!\!* \rangle$. Cela entraîne la traduction des formules 6 et 9. Les autres formules sont ignorées, y compris la partie (non développée ici) entre les formules 10 et 11.

Un cas particulier est la traduction de la formule 15. En effet, le nominal de cette formule est $c_6$ qui n'est pas un label apparaissant dans les formules pures. Il a donc été introduit par une règle $\langle i+ \rangle$. On retrouve la formule qui définit la valeur de $c_6$. Il s'agit de la formule 13, qui nous indique que $c_6$ peut être remplacé par $c_5c_3$. Ainsi la traduction exclut les nominaux surnuméraires du calcul de HBBI.

Finalement, il ne reste qu'à reconstituer le BBI-tableau en séparant les formules et les contraintes, tout en conservant la structure. Le résultat est donné dans la Figure 6.2.

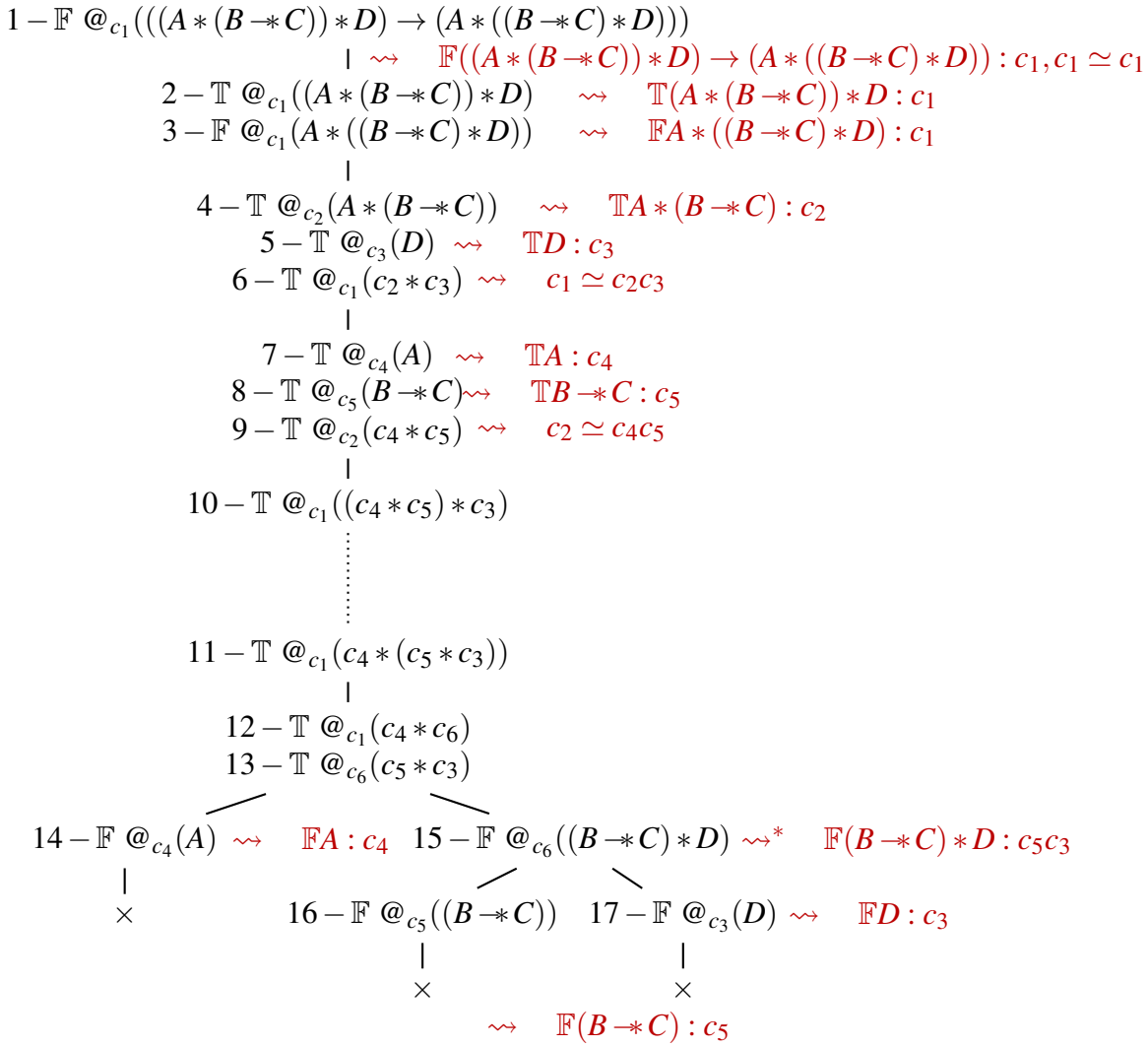



FIGURE 6.1 – Exemple de traduction de HBBI-tableau

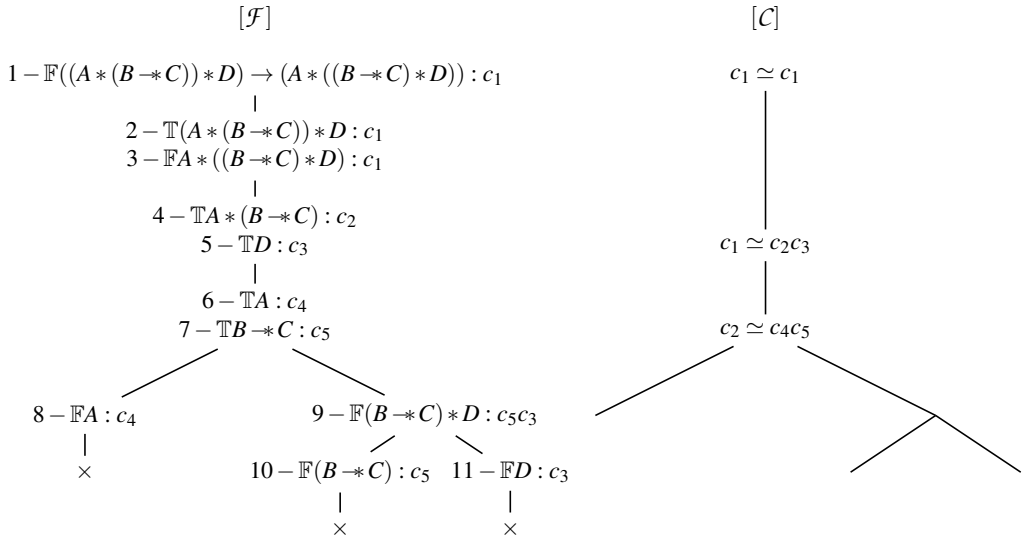



FIGURE 6.2 – Tableau traduit

On peut vérifier que c'est bien un tableau pour la formule de départ. La formule 1 correspond bien à un CSS de départ pour la formule. Sa décomposition par la règle $\langle\mathbb{F}\to\rangle$ produit bien 2 et 3. La décomposition de 2 par $\langle\mathbb{T}*\rangle$ produit 4, 5 et la contrainte correspondante $c_1\simeq c_2c_3$. La décomposition de 4 par $\langle\mathbb{T}*\rangle$ produit 6, 7 et la contrainte correspondante $c_2\simeq c_4c_5$. On doit ensuite décomposer 3 avec $\langle\mathbb{F}*\rangle$. On doit chercher $x$ et $y$ tels que $c_1\simeq xy$. La forme du tableau final indique $x=c_4$ et $y=c_3c_5$, mais ce résultat est-il possible ? Il suffit de construire un arbre de preuve pour $c_1\simeq c_4(c_3c_5)$ dans la clôture des contraintes. On a $c_1\simeq c_2c_3\in\mathcal{C}$. De cela on peut tirer $c_2c_3\simeq c_2c_3$, puis en combinant avec $c_2\simeq c_4c_5$, on obtient $c_2c_3\simeq(c_4c_5)c_3$. Par permutation et association des labels, on a alors $c_2c_3\simeq c_4(c_3c_5)$, puis par combinaison avec $c_1\simeq c_2c_3$ on obtient la contrainte désirée, $c_1\simeq c_4(c_3c_5)$. On peut donc appliquer $\langle\mathbb{F}*\rangle$ pour produire 8 et 9. Dans la branche de gauche, on a alors clôture du tableau en produisant $c_4\simeq c_4$. À droite, on réutilise $\langle\mathbb{F}*\rangle$ en utilisant la contrainte $c_5c_3\simeq c_5c_3$, facile à produire, et on a deux clôture en ajoutant $c_5\simeq c_5$ et $c_3\simeq c_3$.

On a donc bien réussi à créer un BBI-tableau à partir du tableau de HBBI, et on a conservation du caractère clos du tableau.

## 6.2 Des HBBI-tableaux vers les BBI-tableaux

Former un HBBI-tableau à partir d'un BBI-tableau est plus complexe que dans l'autre sens. En effet, une contrainte forte des HBBI-tableaux est la présence de nominaux non composés à gauche de chaque formule. En traduisant les labels en nominaux, il faudra donc prendre bien soin d'examiner les cas particuliers des labels composés. Cela complexifiera grandement nos preuves.

### 6.2.1 Délabellisation

**Définition 6.5** (Formules délabellisées)**.** *Soit $\mathbb{S}\phi : x$ une formule étiquetée et $x \simeq y$ une contrainte de labels du calcul des tableaux de BBI. Soit $N \subseteq Nom$ un ensemble de symboles nominaux n'apparaissant pas dans les labels du calcul des tableaux de BBI. On définit une fonction $\delta$ associant à chaque formule ou contrainte du calcul de BBI une formule étiquetée du calcul pour HBBI :*

1. *La fonction d est définie par :*
   - $d(\varepsilon) = \mathrm{I}$ ;
   - $d(x) = x$ *pour tout label x qui n'est pas composé ;*
   - $d(xy) = d(x) * d(y)$.
2. *Si x est un label non composé, on a* $\delta(\mathbb{S}\phi : x) = \{\mathbb{S}\ @_x(\phi)\}$ ;
3. *Si x est un label composé, on prend un symbole* $c \in N$ *et on a* $\delta(\mathbb{S}\phi : x) = \{\mathbb{S}\ @_c(\phi), \mathbb{T}\ @_c(d(x))\}$ ;
4. *Si x est un label non composé, on a* $\delta(x \simeq y) = \{\mathbb{T}\ @_x(d(y))\}$ ;
5. *Si x est un label composé, on prend un symbole* $c \in N$ *et on a* $\delta(x \simeq y) = \{\mathbb{T}\ @_c(d(y)), \mathbb{T}\ @_c(d(x))\}$.

*Dans tous ces cas, si $x = \varepsilon$, x est remplacé par e dans la formule délabellisée.*

De même que pour la règle $\langle i+\rangle$ du calcul pour HBBI, on peut (et il faut dans certains cas) utiliser deux fois le même $c \in N$ s'il est utilisé deux fois pour le même $d(x)$ (ou $d(y)$).

Avec cette manière de traduire formules et contraintes, on peut donc construire l'image d'une formule étiquetée ou d'une contrainte du calcul de BBI en une formule de HBBI. Toutefois, avant de traduire des branches entières, il reste un problème à résoudre. En effet, le calcul des tableaux pour BBI fait grand usage de la clôture de ses ensembles de contraintes pour générer de nouvelles contraintes, utilisées dans certaines règles ($\langle \mathbb{F}* \rangle$ ou $\langle \mathbb{T}{-\!\!*} \rangle$). Comme nous n'avons pas de dispositifs comparables dans HBBI, il nous faut traduire la clôture des contraintes en formules et en règles de HBBI.

**Lemme 6.3.** *Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un CSS du calcul des tableaux de BBI. Soit $c \in \overline{\mathcal{C}}$ une contrainte produite par clôture de l'ensemble de contraintes $\mathcal{C}$. Il existe un ensemble $\mathcal{R}(c)$ de formules étiquetées du calcul des tableaux pour HBBI contenant $\delta(c)$ tel que pour tout $r \in \mathcal{R}(c)$ on ait soit :*

- *il existe $c' \in \mathcal{C}$ tel que $r \in \delta(c')$,*
- *ou il existe une règle à labels qui produit r à partir d'autres éléments de $\mathcal{R}(c)$.*

*Démonstration.*

La preuve de ce lemme est donnée dans l'Annexe C. □

Notez que le résultat final du lemme est l'existence d'une suite de formules convenant, mais qu'il ne donne pas de construction exacte de cette suite. La preuve fournit une correspondance

directe des règles de clôture des contraintes avec les listes de formules de HBBI. Cette correspondance peut être utilisée littéralement, mais il est fort probable que l'intuition fournisse une alternative plus concise.

On peut maintenant construire une image satisfaisante des tableaux de BBI.

**Définition 6.6** (Tableaux délabellisées)**.** *Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un CSS du calcul des tableaux de BBI.*

*On a :*

$$\delta(\langle \mathcal{F}, \mathcal{C} \rangle) = (\bigcup_{F \in \mathcal{F}} \delta(F)) \cup (\bigcup_{c \in \mathcal{C}} \delta(c)) \cup (\bigcup_{c \in Req} \mathcal{R}(c))$$

*où Req est l'ensemble des contraintes apparaissant dans les prémisses de règles $\langle \mathbb{F}* \rangle$ ou $\langle \mathbb{T}{-\!\!*} \rangle$ utilisées pour construire $\langle \mathcal{F}, \mathcal{C} \rangle$.*

*La délabellisation d'un BBI-tableau $\mathcal{T}$, noté $\delta(\mathcal{T})$ est la liste des $\delta(\langle \mathcal{F}, \mathcal{C} \rangle)$ pour tous les $\langle \mathcal{F}, \mathcal{C} \rangle$ de $\mathcal{T}$.*

Comme précédemment, on va montrer que cette procédure produit bel et bien des tableaux du calcul pour HBBI et que la clôture est préservée par cette transition.

**Théorème 6.2.** *[Propriétés des tableaux délabellisés] Soit $\mathcal{T}$ un BBI-tableau.*

1. *$\delta(\mathcal{T})$ est un HBBI-tableau pur.*
2. *si $\mathcal{T}$ est clos, $\delta(\mathcal{T})$ peut être étendu (par les règles du calcul pour HBBI) en un tableau clos.*
3. *si $\delta(\mathcal{T})$ est clos, alors $\mathcal{T}$ est clos.*

*Démonstration.*

La preuve de ce théorème est donnée dans l'Annexe C. □

### 6.2.2 Un exemple de délabellisation (d'un BBI-tableau vers un HBBI-tableau)

On se propose en guise d'exemple de traduire un BBI- tableau pour une formule simple. On considère la formule $(A * (A \mathrel{-\!\!*} B)) \to B$. On dresse le BBI-tableau pour cette formule. Le tableau obtenu est représenté dans la Figure 6.3.

Dans ce tableau, on développe d'abord l'opérateur $\to$ (1) puis l'opérateur $*$ (2) ce qui produit la nouvelle contrainte $c_1 \simeq c_2 c_3$. Enfin, en (3), on doit sélectionner un label $x$ tel que $c_3 x \simeq c_3 x \in \overline{\mathcal{C}}$. Le label $c_2$ convient. On obtient donc deux nouvelles branches, toutes deux closes : dans celle de gauche, on a $\mathbb{F}A : c_2 \in \mathcal{F}$ et $\mathbb{T}A : c_2 \in \mathcal{F}$ ; dans celle de droite, on a $\mathbb{T}B : c_3 c_2 \in \mathcal{F}$, $\mathbb{F}B : c_1 \in \mathcal{F}$ et $c_1 \simeq c_3 c_2 \in \overline{\mathcal{C}}$.

Comme indiqué dans la Définition 6.2, nous devons établir :

- l'image $\delta(F)$ de chaque formule de $\mathcal{F}$ ;
- l'image $\delta(c)$ de chaque contrainte de $\mathcal{C}$
- pour toute contrainte $c$ utilisée pour une règle $\langle \mathbb{F}* \rangle$ ou $\langle \mathbb{T}{-\!\!*} \rangle$, l'ensemble $\mathcal{R}(c)$. Ici il n'existe qu'une telle contrainte, $c_3 c_2 \simeq c_3 c_2$.

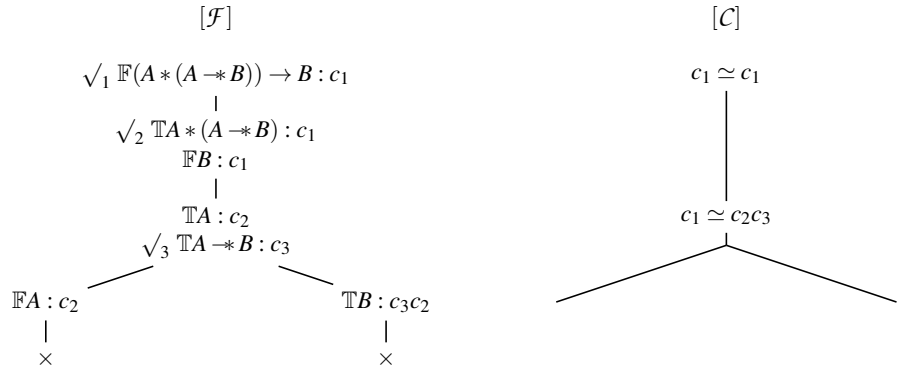



FIGURE 6.3 – BBI-Tableau pour $(A*(A\mathrel{-\!\!*}B))\to B$

Construisons donc les ensembles nécessaires :

**Formules**

$\delta(\mathbb{F}(A*(A\mathrel{-\!\!*}B))\to B : c_1) = \{\mathbb{F}\ @_{c_1}((A*(A\mathrel{-\!\!*}B))\to B)\}$

$\delta(\mathbb{T}A*(A\mathrel{-\!\!*}B) : c_1) = \{\mathbb{T}\ @_{c_1}(A*(A\mathrel{-\!\!*}B))\}$

$\delta(\mathbb{F}B : c_1) = \{\mathbb{F}\ @_{c_1}(B)\}$

$\delta(\mathbb{T}A : c_2) = \{\mathbb{T}\ @_{c_2}(A)\}$

$\delta(\mathbb{T}A\mathrel{-\!\!*}B : c_3) = \{\mathbb{T}\ @_{c_3}(A\mathrel{-\!\!*}B)\}$

$\delta(\mathbb{F}A : c_2) = \{\mathbb{F}\ @_{c_2}(A)\}$

Enfin, pour déterminer $\delta(\mathbb{T}B : c_3c_2)$, comme le label est composé, il nous faut introduire un nouveau nominal qui n'apparaît pas dans le tableau. On le note $c_4$ :

$\delta(\mathbb{T}B : c_3c_2) = \{\mathbb{T}\ @_{c_4}(B), \mathbb{T}\ @_{c_4}(c_3*c_2)\}$

**Contraintes**

$\delta(c_1 \simeq c_1) = \{\mathbb{T}\ @_{c_1}(c_1)\}$

$\delta(c_1 \simeq c_2c_3) = \{\mathbb{T}\ @_{c_1}(c_2*c_3)\}$

**Clôture de contraintes**

Le Lemme 6.3 nous indique qu'il existe une série de formules étiquetées pour HBBI telles que chacune soit l'image d'une contrainte ou produite par une règle à labels, et qui contient la formule recherchée, à savoir $\delta(c_3c_2 \simeq c_3c_2)$.

Commençons par déterminer $\delta(c_3c_2 \simeq c_3c_2)$. Il nous faut encore un nouveau nominal $c_5$ , et on a :

$\delta(c_3c_2 \simeq c_3c_2) = \{\mathbb{T}\ @_{c_5}(c_3*c_2)\}$

Toutefois, comme on a déjà utiliser $c_4$ pour former $\mathbb{T}\ @_{c_4}(c_3*c_2)$, on a intérêt à utiliser plutôt $c_4$ à la place de $c_5$. On préférera donc :

$\delta(c_3c_2 \simeq c_3c_2) = \{\mathbb{T}\ @_{c_4}(c_3*c_2)\}$

Pour déterminer la série de formules amenant à $\delta(c_3c_2 \simeq c_3c_2)$, on peut considérer l'arbre de preuve amenant à $c_3c_2 \simeq c_3c_2$ et traduire chacune des règles, comme dans la preuve du Lemme 6.3. Cela dit, il est plus simple de chercher simplement un arbre par nous-même. L'arbre suivant convient :

$$\dfrac{\dfrac{\mathbb{T}\ @_{c_1}(c_2*c_3)}{\mathbb{T}\ @_{c_1}(c_3*c_2)}\ \langle BI_c\rangle}{\mathbb{T}\ @_{c_1}(c_4), \mathbb{T}\ @_{c_4}(c_3*c_2)}\ \langle i{+}\rangle$$

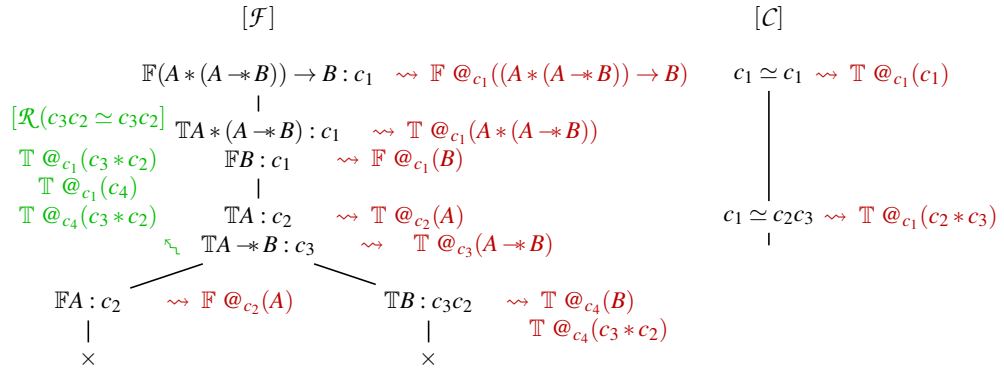



FIGURE 6.4 – BBI-Tableau pour $(A * (A -\!\!* B)) \to B$ avec les délabellisations

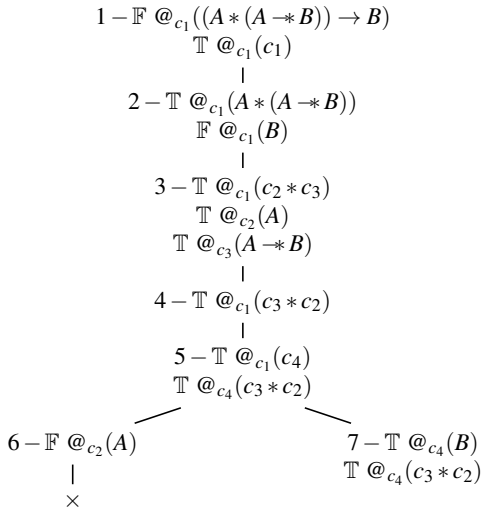



FIGURE 6.5 – BBI-Tableau traduit

Dans la dernière règle $\langle i+\rangle$, on peut se permettre d'utiliser $c_4$ puisque $c_4$ a été introduit par la même formule qui est produite par la règle, $\mathbb{T}\ @_{c_4}(c_3 * c_2)$. L'arbre de preuve est convenable puisque son axiome, $\mathbb{T}\ @_{c_1}(c_2 * c_3)$, correspond à $\delta(c_1 \simeq c_2c_3)$.

Finalement, on a réussi à atteindre $\mathbb{T}\ @_{c_4}(c_3 * c_2)$, et on a :

$$\mathcal{R}(c_3c_2 \simeq c_3c_2) = \{\mathbb{T}\ @_{c_1}(c_2 * c_3), \mathbb{T}\ @_{c_1}(c_3 * c_2), \mathbb{T}\ @_{c_1}(c_4), \mathbb{T}\ @_{c_4}(c_3 * c_2)\}$$

On a alors tous les éléments pour dresser notre tableau délabellisé (Figure 6.4).
Ce qui produit le HBBI-tableau suivant (Figure 6.5).
On peut vérifier que le résultat est bien un HBBI-tableau :

1. La première formule étiquetée $\mathbb{F}\ @_{c_1}((A * (A -\!\!* B)) \to B)$. Correspond au SHS de départ d'une preuve pour la formule $(A * (A -\!\!* B)) \to B$. La seconde formule $\mathbb{T}\ @_{c_1}(c_1)$ peut être produite par la règle $\langle i_r\rangle$. Toutefois, cette formule n'es pas utilisée dans le reste du tableau et peut donc être retirée du tableau.

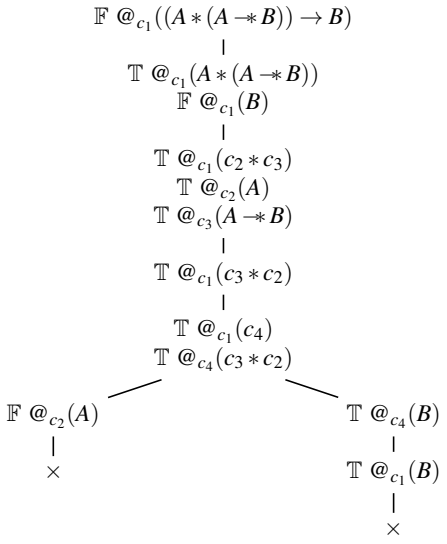



FIGURE 6.6 – BBI-Tableau traduit, simplifié et clos

2. Ces formules sont le produit de la décomposition de $\mathbb{F}\ @_{c_1}((A * (A \mathrel{-\!\!*} B)) \to B)$ (1) par la règle $\langle \mathbb{F} \to \rangle$.
3. Ces formules sont le produit de la décomposition de la $\mathbb{T}\ @_{c_1}(A * (A \mathrel{-\!\!*} B))$ (2) par la règle $\langle \mathbb{F} * \rangle$.
4. Cette formule est le produit de la décomposition de $\mathbb{T}\ @_{c_1}(c_2 * c_3)$ (3) par la règle $\langle i_s \rangle$.
5. Ces formules sont le produit de la décomposition de $\mathbb{T}\ @_{c_1}(c_3 * c_2)$ (4) par la règle $\langle i+ \rangle$.
6. En utilisant $\mathbb{T}\ @_{c_3}(A \mathrel{-\!\!*} B)$ (3) avec $\mathbb{T}\ @_{c_4}(c_3 * c_2)$ (5), on obtient cette première branche par $\langle \mathbb{T} \mathrel{-\!\!*} \rangle$. Cette branche est close puisque on y trouve aussi $\mathbb{T}\ @_{c_2}(A)$.
7. En utilisant $\mathbb{T}\ @_{c_3}(A \mathrel{-\!\!*} B)$ (3) avec $\mathbb{T}\ @_{c_4}(c_3 * c_2)$ (5), on obtient cette deuxième branche par $\langle \mathbb{T} \mathrel{-\!\!*} \rangle$. La formule $\mathbb{T}\ @_{c_4}(c_3 * c_2)$, produite par la délabellisation est redondante (avec 5) et peut donc être supprimée. La branche n'est pas close mais peut l'être avec une application de $\langle i_t \rangle$ sur $\mathbb{T}\ @_{c_1}(c_4)$ et $\mathbb{T}\ @_{c_4}(B)$.

Ainsi, on obtient bien un tableau et on peut l'étendre en un tableau clos comme représenté dans la Figure 6.6 (où nous avons aussi supprimé les formules excédentaires).

Pour conclure, on a mis à jour une traduction entre les BBI-tableaux et les HBBI-tableaux purs, dans les deux sens, et en conservant le caractère clos des tableaux. Cela nous montre comment l'utilisation de labels peut être remplacée par l'utilisation d'un système de nominaux internes comme il en existe dans HRL.

Toutefois, l'exemple que nous avons traité ci-dessous met aussi un lumière les avantages des deux formes de preuves. On voit comment le tableau original (HBBI) est plus complexe, puisqu'il comporte de nombreuses formules absentes du tableau BBI. Toutefois, l'apparition de ces formules supplémentaires est dictée par l'application plus systématique des règles de manipulation des nominaux de HBBI, alors que le calcul de BBI s'appuie sur la clôture de

contraintes qui, si elle est plus directe, s'appuie sur l'intuition. Ainsi, on peut généralement dire que le calcul de HBBI est plus complexe mais plus systématique que celle de BBI.

Enfin, notons que ces considérations ne prennent pas en compte le fait que HBBI présente également l'avantage supplémentaire de pouvoir s'étendre au delà de BBI en utilisant HRL.

# Conclusion

Nous avons décrit une transformation qui permet à partir d'un HBBI-tableau pour une formule de BBI d'obtenir un BBI-tableau pour la même formule. En outre, le caractère clos du tableau est conservé par cette transformation. On a également montré la transformation inverse (d'un BBI-tableau vers un HBBI-tableau).

Ces transformations nous montrent comment les labels du calcul pour BBI et les nominaux supplémentaires du calcul pour HBBI sont deux présentations différentes d'un même processus et sont équivalents. Ainsi, l'usage des HBBI-tableaux permet de s'affranchir des labels.

Toutefois, on a également montré comment le calcul des HBBI-tableaux était moins aisé à utiliser de par le grand nombre de règles à appliquer, étant donné que toutes les manipulations sont explicites, y compris celles qui étaient implicites dans le calcul pour BBI. Le calcul pour HBBI est donc plus systématique, mais moins facile à utiliser.

# Chapitre 7

# Conclusion et perspectives

Nous avons présenté dans cette thèse des extensions et variantes des logiques BI et BBI que nous avons commencé par présenter et comparé. Comme nous l'avons dit, la capacité de ces logiques à exprimer à la fois la consommation et la production de ressources, mais aussi la séparation et le partage des propriétés par des ressources, en font un excellent noyau pour décrire des systèmes et des programmes. Ces capacités sont notamment très appréciables pour utiliser BI et BBI comme noyau pour des outils de modélisation de systèmes et de programmes, comme cela a été fait pour la logique de séparation [21,87]. Il est ainsi légitime de chercher à étudier et étendre ces logiques pour étendre leur capacités d'expression.

Nous avons d'abord proposé une logique LTBI, extension de BI par des opérateurs modaux temporels, issus de la logique temporelle linéaire LTL. Nous avons ensuite défini une logique ERL qui étend BBI par des opérateurs épistémiques exprimant l'accès à une propriété par un agent moyennant une ressource. Enfin, nous avons proposé une logique HRL qui internalise les ressources dans la syntaxe de BBI ce qui permet à la fois d'en étendre l'expressivité et d'en moduler facilement la sémantique à l'aide d'axiomes.

Si on aborde la question de l'expressivité des logiques, les trois logiques proposées ont chacune la particularité d'étendre le pouvoir d'expressivité de BI et BBI.

La logique LTBI permet d'ajouter un dynamisme aux logiques de ressources. Ce point avait déjà été réalisé par d'autres extensions de BI dans le passé [34,35], mais l'utilisation de LTL comme noyau pour ce qui est de la dimension temporelle offre de nombreuses perspectives intéressantes, notamment avec l'usage de l'opérateur $\circ$ qui permet de désigner l'état suivant. Cela permet à la fois de modéliser l'évolution de systèmes de consommation dans le temps (ce que permettait déjà les versions modales plus générales comme [34]) mais aussi de décrire des transitions de systèmes pouvant être décomposés en sous-systèmes distincts. Enfin, l'utilisation des ressources comme marqueurs de futurs possibles ouvre la capacité d'utiliser LTBI comme une logique temporelle branchante, au pouvoir d'expression différent des logiques temporelles branchantes traditionnelles comme CTL.

La logique ERL offre une variété d'expression en termes de sécurité, plus particulièrement de contrôle d'accès. Nous avons notamment montré comment elle pouvait être utilisée pour décrire un espace à accès restreint, y compris les éventuelles failles d'exécution de ce dernier. Nous avons aussi proposé un exemple d'accès conjoint à une ressource et d'accès concurrent à une

ressource. Nous avons montré en outre comment les sous-ensembles d'agents d'ERL peuvent être un outil pour la modélisation d'un système par raffinement de couches conceptuelles.

Concernant les logiques HRL et HBBI, nous avons montré comment l'ajout de nominaux représentant les ressources au cœur de la syntaxe permet d'exprimer de nouvelles propriétés sur les ressources, ou en tout cas fournit une manière plus lisible de les exprimer. Nous avons également expliqué comment la modularité de HRL par l'ajout et le retrait d'axiomes permet d'obtenir aisément des variantes différentes de BBI dont l'expressivité varie (par exemple, une version où les ressources ne peuvent pas commuter, ou bien une version où les ressources sont inversibles).

À partir de ces travaux, on peut envisager certaines perspectives quant à l'expressivité. Par exemple, intégrer des nominaux dans LTBI permettrait d'exprimer un véritable dynamisme de ressource : une formule comme $i \wedge \Diamond j$ exprimerait par exemple que $i$ se transforme en $j$ à un certain point de l'avenir. Dans le même esprit, les nominaux dans ERL permettraient de mieux utiliser les ressources dans nos modélisations. Ainsi, on pourrait avoir des formules $\mathbf{L}_a^s t$ qui exprime que l'agent $a$, avec la ressource $s$, peut avoir accès à la ressource $t$. On voit ici comment les nominaux peuvent permettre de manipuler les ressources plus finement. On pourrait aussi étendre les propriétés d'ERL par des dimensions temporelles (avoir accès à une propriété un jour, ou toujours y avoir accès). Enfin, avoir une logique qui intègre nos différents types de modalités permettrait d'étendre grandement notre pouvoir d'expression, par exemple pour modéliser plus efficacement des propriétés de sécurité.

La conception de calculs des tableaux pour ces logiques a été une question importante à l'appui de la définition de nos logiques. Ces calculs sont inspiré des travaux sur BI et BBI [49,66] qui ont la particularité d'utiliser les concepts de labels et de contraintes pour décrire les structures des ressources. Le challenge pour chacune de nos logiques était d'étendre ou de modifier les labels et les contraintes pour traduire correctement nos sémantiques tout en conservant les bonnes propriétés du calcul des tableaux original (correction et complétude). Pour chaque calcul, nous avons présenté des exemples, une preuve de correction et de complétude et une méthode d'extraction de contre-modèles.

Les tableaux pour LTBI possèdent un système de double labels inspiré des travaux de [34]. La particularité de ce calcul provient du fait que les labels temporels doivent respecter la structure des entiers naturels. La linéarité des labels temporels est assurée par des règles temporelles, mais la possible densité de ces labels reste un problème non résolu, même si des pistes prometteuses sont avancées (voir la section 3.4.3). LTBI possède en outre un autre calcul, un calcul des c-tableaux, inspiré des calculs pour LTL, et qui utilise des structures de graphe pour se passer de labels temporels. Nous en avons montré la correction. Établir une correspondance formelle entre les tableaux et les c-tableaux de LTBI, en exhibant une méthode de conversion de l'un en l'autre est un objectif majeur de nos perspectives.

Les tableaux pour ERL font appel à des contraintes étiquetées par des ressources, ce qui force à la prudence lors de l'établissement des liens entre labels et ressources (notamment pour la preuve de correction et l'établissement de contre-modèles).

Les tableaux pour HRL sont fortement inspirés de ceux pour BBI mais en changent l'approche et la mise en forme, puisque les nominaux propres à ERL sont utilisés en place des labels de ressources. Cela rend les tableaux plus complexes, car la totalité des opérations sur les

ressources y est représenté, là où les tableaux traditionnels utilisaient la clôture des ensembles de contraintes. D'un autre côté, l'absence de contraintes et de labels rend les preuves de correction et de complétude beaucoup plus direct, et l'inclusion de toutes les manipulations sur les ressources comme des règles de même nature rend le calcul plus facile à systématiser. Enfin, tout comme HRL peut être étendue par divers axiomes menant à différentes variantes comme HBBI (équivalente sémantiquement à BBI), le calcul des tableaux pour HRL peut être facilement étendu par des règles qui traduisent ces nouveaux axiomes. On a montré la correction et la complétude de la variante pour HBBI du calcul. On a aussi montré l'équivalence des calculs pour BBI et HBBI en proposant une méthode de traduction des tableaux d'une méthode vers l'autre.

En général, le travail sur ces calculs a mis en lumière la puissance de ce type de méthode formelle et son adaptabilité, tout en soulignant les difficultés que présentent l'introduction de nouveaux types de labels ou de contraintes. En particulier, la technique de preuve de complétude introduite pour BBI [66], à base d'oracle et de stratégie équitable, a fait ses preuves comme outil malléable pour montrer la complétude de calculs des tableaux divers, même si son adaptation réclame un véritable travail de fond.

Outre l'intérêt de chaque calcul vis-à-vis de la logique associée, nous avons présenté dans les Chapitres 3,5 et 6 deux pistes de réflexion sur le calcul des tableaux en général, et sur l'utilisation de labels en particulier.

Dans les Chapitres 5 et 6, nous avons montré comment l'internalisation de ressources pouvait permettre une alternative équivalente aux labels et contraintes de labels. Nous avons aussi vu comment les labels offrent une lisibilité humaine plus grande, mais nécessitent plus d'intuition lors de l'établissement des tableaux. Cela montre comment les labels ne sont pas forcément essentiels mais contribuent à la compréhension des preuves par tableaux.

Dans le chapitre 3, nous avons proposé un calcul des tableaux inspiré de celui pour LTL qui utilise des structures de graphe pour que la structure des états dans le tableau soit exprimée non pas par des labels mais par les arêtes du tableau elles-mêmes. Bien que la traduction de c-tableaux en tableaux (et vice-versa) ne soit pas encore établie, on peut à nouveau exhiber cet exemple comme une possibilité de supprimer les labels (ici labels d'état) dans un calcul de tableaux. Encore une fois, la différence de facilité d'écriture est sensible, puisque les c-tableaux sont typiquement plus complexes que leurs équivalents avec labels, et leur clôture est plus ardue à vérifier. Toutefois, on remarque à nouveau que ce calcul sans label est plus systématique et fait moins appel à l'intuition.

Enfin, l'exemple des c-tableaux pose la question de la possibilité d'utiliser des structures similaires, par exemple des graphes particuliers, pour un calcul de BI ou BBI : l'idée serait de faire porter la composition et la séparation des ressources par certaines arêtes des tableaux plutôt que par les labels. Cette piste de recherche est confortée par les travaux existants sur les graphes de ressource dans les tableaux à labels pour BI [48].

Notre recherche amène de nombreux possibles futurs travaux, à la fois pour compléter des points inachevés de cette thèse et pour explorer des thématiques prometteuses.

Tout d'abord, chacune des logiques que nous avons proposées s'appuie ou bien sur BI ou

bien sur BBI : LTBI est construite sur BI et possède donc des opérateurs additifs intuitionnistes, tandis que ERL, HRL et HBBI s'appuient sur BBI et possèdent donc des opérateurs additifs classiques. Il faudrait présenter pour chacune une alternative correspondante, c'est à dire une version classique de LTBI, une version intuitionniste de ERL et une version intuitionniste de HRL et HBBI.

Pour ce qui est de la logique LTBI, la priorité est d'amener une preuve de la conjecture 3.1 pour solidifier le calcul des tableaux présenté. Le travaux sur les c-tableaux est aussi incomplet : il faut présenter une preuve de complétude et étudier les liens entre tableaux et c-tableaux, si possible par une traduction de l'un vers l'autre. L'utilisation de LTBI comme logique temporelle branchante est également une direction de recherche à explorer, notamment quant à son utilisation pour modéliser des processus parallèles. Enfin, il faudra s'intéresser à la possibilité de concevoir un calcul des tableaux sans labels pour les logiques BI ou BBI en s'inspirant des c-tableaux.

Pour la logique ERL, l'ajout d'annonce à la manière de PAL [76] permettrait une extension intéressante d'expressivité. L'étude approfondie de l'utilisation d'ERL comme logique pour décrire des protocoles de communication est aussi un travail à mener.

Pour la logique HRL, de nombreuses pistes peuvent être explorées, avec la possibilité de moduler HRL par l'ajout et la suppression d'axiomes. Un exemple est l'étude des ressources inversibles que nous avons esquissé. En outre, la possibilité d'utiliser le calcul des tableaux de HRL pour étudier ces différentes variantes est un atout.

Enfin, un axe d'étude que nous avons mis de côté pour chacune de nos trois logiques est l'étude de leur éventuelle décidabilité ainsi que de leur complexité. Les travaux déjà menés à ce sujet pour BI et BBI [49,69] laissent présager que ces preuves sont loin d'être triviales pour les extensions que nous avons présentées.

Un dernier axe important est celui de l'implémentation des différents calculs des tableaux sur machine. Cette dernière pourrait mettre en avant la modularité de ce type de méthode formelle : en effet, nos différents calculs sont très proches dans leur esprit (avec l'appui commun sur les labels et les contraintes), et l'implémentation solide de l'un des calculs devrait pouvoir mener à une adaptation rapide pour un autre de ces calculs. Concernant le calcul pour LTBI, la dualité avec le calcul des c-tableaux offre en outre un lien vers la modélisation systématique par des automates de Büchi. Enfin, concernant le calcul pour HRL, la possibilité d'étendre ou de restreindre le calcul par l'ajout ou la suppression de règles axiomatiques est un argument supplémentaire pour en faire une implémentation efficace et modulaire.

# Bibliographie


[1] J. R. Abrial. *The B-book : assigning programs to meanings*. Cambridge University Press, New York and NY and USA, 1996.

[2] G. Anderson, M. Collinson, and D. J. Pym. Trust Domains : An Algebraic, Logical, and Utility-Theoretic Approach. In *Trust and Trustworthy Computing, TRUST 2013, London, UK. Proceedings*, number 7904 in LNCS, pages 232–249, 2013.

[3] G. Anderson and D. Pym. Trust domains in system models : algebra, logic, utility, and combinators. *Journal of Logic and Computation*, 2015.

[4] G. Anderson and D. Pym. A calculus and logic of bunched resources and processes. *Theoretical Computer Science*, 614 :63–96, 2016.

[5] C. Areces, P. Blackburn, and M. Marx. A Road-Map on Complexity for Hybrid Logics. In J. Flum and M. Rodríguez-Artalejo, editors, *EACSL, Madrid, Spain, Proceedings*, volume 1683 of *LNCS*, pages 307–321. Springer, 1999.

[6] A. Avron. Gentzen-type systems, resolution and tableaux. *Journal of Automated Reasoning*, 10(2) :265–281, 1993.

[7] P. Balbiani and J. Boudou. Decidability of Iteration-free PDL with Parallel Composition. In *Workshop on Automated Deduction : Decidability, Complexity, Tractability, ADDCT'14, Vienna, Austria*, 2014.

[8] P. Balbiani and D. Galmiche. About intuitionistic public announcement logic. In *11th conference on Advances in Modal logic (AiML 2016)*, pages pp. 97–116, Budapest, Hungary, Aug. 2016.

[9] A. Baltag, B. Coecke, and M. Sadrzadeh. Epistemic actions as resources. *Journal of Logic and Computation*, 17(3) :555–585, 2007.

[10] M. R. Benevides, R. de Freitas, and P. Viana. Propositional Dynamic Logic with Storing, Recovering and Parallel Composition. *Logical and Semantic Frameworks, with Applications Workshop (LSFA 2010), ENTCS*, 269 :95 – 107, 2011.

[11] J. Benthem. *Handbook of Logic in Artificial Intelligence and Logic Programming IV : Epistemic and Temporal Reasoning*, chapter Temporal Logic, pages 241–350. Oxford University Press, 1995.

[12] P. N. Benton, G. M. Bierman, V. de Paiva, and M. Hyland. A Term Calculus for Intuitionistic Linear Logic. In M. Bezem and J. F. Groote, editors, *Typed Lambda Calculi and Applications, TLCA '93, Utrecht, The Netherlands, Proceedings*, volume 664 of *LNCS*, pages 75–90. Springer, 1993.

[13] N. Biri and D. Galmiche. A Separation Logic for Resource Distribution : Extended Abstract. In *Foundations of Software Technology and Theoretical Computer Science, FST TCS 2003, Mumbai, India, Proceedings*, volume 2914 of *LNCS*, pages 23–37, 2003.

[14] P. Blackburn. Internalizing labelled deduction. *Journal of Logic and Computation*, 10(1) :137–168, 2000.

[15] P. Blackburn. Representation, reasoning, and relational structures : A hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3), 2000.

[16] P. Blackburn. Arthur Prior and Hybrid Logic. *Synthese*, 150(3) :329–372, 2006.

[17] T. Bolander and T. Braüner. Tableau-based Decision Procedures for Hybrid Logic. *JLC*, 16(6) :737–763, 2006.

[18] T. Braüner. Intuitionistic hybrid logic : Introduction and survey. *Information and Computation*, 209(12) :1437 – 1446, 2011. Intuitionistic Modal Logic and Applications (IMLA 2008).

[19] J. Brotherston. A Unified Display Proof Theory for Bunched Logic. *Electronic Notes in Theoretical Computer Science*, 265 :197 – 211, 2010. Proceedings of the 26th Conference on the Mathematical Foundations of Programming Semantics (MFPS 2010).

[20] J. Brotherston and J. Villard. Parametric Completeness for Separation Theories. *SIGPLAN Not.*, 49(1) :453–464, Jan. 2014.

[21] C. Calcagno, D. Distefano, J. Dubreil, D. Gabi, P. Hooimeijer, M. Luca, P. O'Hearn, I. Papakonstantinou, J. Purbrick, and D. Rodriguez. Moving Fast with Software Verification. In K. Havelund, G. Holzmann, and R. Joshi, editors, *NASA Formal Methods*, pages 3–11, Cham, 2015. Springer International Publishing.

[22] G. Cantor. Beiträge zur Begründung der transfiniten Mengenlehre. *Mathematische Annalen*, 46(4) :481–512, Nov. 1895.

[23] L. Cardelli and A. Gordon. Anytime, Anywhere : Modal Logics for Mobile Ambients. pages 365–377. ACM Press, 2000.

[24] I. Cervesato. Typed Multiset Rewriting Specifications of Security Protocols. *Electr. Notes Theor. Comput. Sci.*, 40 :8–51, 2000.

[25] E. M. Clarke and E. A. Emerson. Design and Synthesis of Synchronization Skeletons Using Branching-Time Temporal Logic. In *Logic of Programs*, pages 52–71, 1981.

[26] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic Verification of Finite-state Concurrent Systems Using Temporal Logic Specifications. *ACM Trans. Program. Lang. Syst.*, 8(2) :244–263, Apr. 1986.

[27] E. M. Clarke, Jr., O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, Cambridge, MA, USA, 1999.

[28] M. Collinson, K. McDonald, and D. Pym. A substructural logic for layered graphs. *Journal of Logic and Computation*, 24(4) :953–988, 2014.

[29] M. Collinson, K. McDonald, and D. Pym. Layered graph logic as an assertion language for access control policy models. *Journal of Logic and Computation*, 27(1) :41–80, 2017.

[30] M. Collinson, B. Monahan, and D. Pym. *A discipline of mathematical systems modelling*. College Publications, 2012.

[31] M. Collinson and D. Pym. Algebra and logic for resource-based systems modelling. *Mathematical Structures in Computer Science*, 19(5) :959–1027, 2009.

[32] J.-R. Courtault. *Logiques de ressources dynamiques : modèles, propriétés et preuves*. PhD thesis, Université de Lorraine, 2015.

[33] J.-R. Courtault, H. v. Ditmarsch, and D. Galmiche. A Public Announcement Separation Logic. *Mathematical Structures in Computer Science (MSCS)*, 2017 (accepted, unpublished).

[34] J.-R. Courtault and D. Galmiche. A Modal BI Logic for Dynamic Resource Properties. In *International Symposium LFCS, San Diego, CA, USA, Proceedings*, volume 7734 of *LNCS*, pages 134–148. Springer, 2013.

[35] J.-R. Courtault and D. Galmiche. A modal separation logic for resource dynamics. *Journal of Logic and Computation*, page exv031, 2015.

[36] J.-R. Courtault, D. Galmiche, and D. Pym. A logic of separating modalities. *Theoretical Computer Science*, 637 :30–58, 2016.

[37] J.-R. Courtault, H. Van Ditmarsch, and D. Galmiche. An epistemic separation logic. In *International Workshop on Logic, Language, Information, and Computation*, pages 156–173. Springer, 2015.

[38] J. Despeyroux and K. Chaudhuri. A Hybrid Linear Logic for Constrained Transition Systems. In LIPIcs, editor, *TYPES'2013*, volume 26, pages 150–168, Toulouse, France, Apr. 2013.

[39] E. W. Dijkstra. Over de sequentialiteit van procesbeschrijvingen. circulated privately, undated, 1962 or 1963.

[40] H. v. Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Springer Publishing Company, Incorporated, 1st edition, 2007.

[41] S. Docherty and D. Pym. Intuitionistic layered graph logic. In *International Joint Conference on Automated Reasoning*, pages 469–486. Springer, 2016.

[42] J. M. Dunn. *Relevance Logic and Entailment*, pages 117–224. Springer Netherlands, Dordrecht, 1986.

[43] M. J. Fischer and R. E. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, 18(2) :194 – 211, 1979.

[44] M. Fitting. *First-order Logic and Automated Theorem Proving (2Nd Ed.)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1996.

[45] D. Galmiche, P. Kimmel, and D. Pym. A Substructural Epistemic Resource Logic. In *Indian Conference on Logic and Its Applications*, number 10119 in LNCS, pages 106–122. Springer, 2017.

[46] D. Galmiche and D. Méry. Semantic Labelled Tableaux for Propositional BI. *J. Log. Comput.*, 13(5) :707–753, 2003.

[47] D. Galmiche and D. Méry. Characterizing Provability in BI's Pointer Logic through Resource Graphs. In *Logic for Programming, Artificial Intelligence, and Reasoning, LPAR 2005, Montego Bay, Jamaica, Proceedings*, number 3835 in LNCS, pages 459–473, 2005.

[48] D. Galmiche and D. Méry. Resource Graphs and Countermodels in Resource Logics. *ENTCS*, 125(3) :117–135, 2005.

[49] D. Galmiche, D. Méry, and D. Pym. The semantics of BI and resource tableaux. *Mathematical Structures in Computer Science*, 15(6) :1033–1088, 2005.

[50] D. Garg, L. Bauer, K. D. Bowers, F. Pfenning, and M. K. Reiter. A Linear Logic of Authorization and Knowledge. In D. Gollmann, J. Meier, and A. Sabelfeld, editors, *Computer Security – ESORICS 2006*, pages 297–312. Springer Berlin Heidelberg, 2006.

[51] P. Gastin and D. Oddoux. Fast LTL to Büchi Automata Translation. In G. Berry, H. Comon, and A. Finkel, editors, *Computer Aided Verification*, pages 53–65, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[52] R. Gerth, D. Peled, M. Y. Vardi, and P. Wolper. *Simple On-the-fly Automatic Verification of Linear Temporal Logic*, pages 3–18. Springer US, Boston, MA, 1996.

[53] J.-Y. Girard. Linear Logic. *Theoretical Computer Science*, 50 :1–102, 1987.

[54] R. Girle. *Modal Logics and Philosophy*. Acumen, 2009.

[55] R. Goldblatt. *Logics of Time and Computation*. Center for the Study of Language and Information, Stanford, CA, USA, 1987.

[56] G. Governatori. Labelled Modal Tableaux. In C. Areces and R. Goldblatt, editors, *Advances in Modal Logic, Volume 7*, pages 87–110. CSLI Publications, 2008.

[57] J. Y. Halpern and R. Pucella. *Modeling Adversaries in a Logic for Security Protocol Analysis*, pages 115–132. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[58] D. Harel, J. Tiuryn, and D. Kozen. *Dynamic Logic*. MIT Press, Cambridge, MA, USA, 2000.

[59] M. Hennessy and R. Milner. Algebraic Laws for Nondeterminism and Concurrency. *J. ACM*, 32(1) :137–161, Jan. 1985.

[60] A. Herzig. A Simple Separation Logic. In L. Libkin, U. Kohlenbach, and R. de Queiroz, editors, *Logic, Language, Information, and Computation*, pages 168–178, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[61] C. A. R. Hoare. An Axiomatic Basis for Computer Programming. *Commun. ACM*, 12(10) :576–580, 1969.

[62] Z. Hou, A. Tiu, and R. Goré. A Labelled Sequent Calculus for BBI : Proof Theory and Proof Search. In *Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX 2013, Nancy, France, Proceedings*, number 8123 in LNCS, pages 172–187, 2013.

[63] S. S. Ishtiaq and P. W. O'Hearn. BI as an Assertion Language for Mutable Data Structures. In C. Hankin and D. Schmidt, editors, *Symposium on Principles of Programming Languages, POPL 2001, London, UK,*, pages 14–26. ACM, 2001.

[64] N. Kamide. Temporal BI : Proof system, semantics and translations. *Theoretical Computer Science*, 492 :40–69, 2013.

[65] F. Kröger and S. Merz. *Temporal Logic and State Systems*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2008.

[66] D. Larchey-Wendling. The formal strong completeness of partial monoidal Boolean BI. *J. Log. Comput.*, 26(2) :605–640, 2016.

[67] D. Larchey-Wendling and D. Galmiche. Exploring the relation between Intuitionistic BI and Boolean BI : an unexpected embedding. *Mathematical Structures in Computer Science*, 19(3) :435–500, 2009.

[68] D. Larchey-Wendling and D. Galmiche. The Undecidability of Boolean BI through Phase Semantics. In *Symposium on Logic in Computer Science, LICS 2010, Edinburgh, United Kingdom*, IEEE, pages 140–149, 2010.

[69] D. Larchey-Wendling and D. Galmiche. Nondeterministic Phase Semantics and the Undecidability of Boolean BI. *ACM Trans. Comput. Log.*, 14(1) :6 :1–6 :41, 2013.

[70] C. Lewis and C. Langford. *Symbolic logic*. Century philosophy series. The Century co., 1932.

[71] P. Maier. Intuitionistic LTL and a New Characterization of Safety and Liveness. In J. Marcinkowski and A. Tarlecki, editors, *Computer Science Logic*, pages 295–309, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[72] D. Méry. *Preuves et sémantiques dans des logiques de ressources*. PhD thesis, Université Nancy 1, 2004.

[73] P. Naumov and J. Tao. Budget-Constrained Knowledge in Multiagent Systems. In G. Weiss, P. Yolum, R. H. Bordini, and E. Elkind, editors, *Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey,*, pages 219–226. ACM, 2015.

[74] P. W. O'Hearn and D. J. Pym. The logic of bunched implications. *Bulletin of Symbolic Logic*, 5(2) :215–244, 1999.

[75] P. W. O'Hearn, J. C. Reynolds, and H. Yang. Local Reasoning about Programs that Alter Data Structures. In *Computer Science Logic, CSL 2001, Paris, France, Proceedings*, pages 1–19, 2001.

[76] J. A. Plaza. Logics of Public Communications. In *Proc. of the $4^{th}$ ISMIS*, pages 201–216. Oak Ridge National Laboratory, 1989.

[77] A. Pnueli. The Temporal Logic of Programs. In *Foundations of Computer Science, FOCS 1977, Providence, Rhode Island, USA,*, IEEE, pages 46–57, 1977.

[78] A. N. Prior. *Past, Present and Future*. Oxford University Press, 1967.

[79] A. N. Prior. *Papers on Time and Tense*. Oxford University Press, 2003.

[80] D. Pym. *The Semantics and Proof Theory of the Logic of Bunched Implications*. Applied Logic Series. Springer Netherlands, 2002.

[81] D. J. Pym, P. W. O'Hearn, and H. Yang. Possible worlds and resources : the semantics of BI. *Theoretical Computer Science*, 315(1) :257–305, 2004.

[82] D. J. Pym and C. M. N. Tofts. A Calculus and logic of resources and processes. *Formal Asp. Comput.*, 18(4) :495–517, 2006.

[83] D. J. Pym and C. M. N. Tofts. Systems Modelling via Resources and Processes : Philosophy, Calculus, Semantics, and Logic. *ENTCS*, 172 :545–587, 2007.

[84] J. P. Queille and J. Sifakis. Specification and verification of concurrent systems in CESAR. In M. Dezani-Ciancaglini and U. Montanari, editors, *International Symposium on Programming*, pages 337–351, Berlin, Heidelberg, 1982. Springer Berlin Heidelberg.

[85] S. Read. *Relevant Logic : A Philosophical Examination of Inference*. Blackwell, 1988.

[86] G. Restall. *An Introduction to Substructural Logics*. Routledge, 2000.

[87] J. C. Reynolds. Separation logic : a logic for shared mutable data structures. In *Proc. 17th Annual IEEE Symp. Logic in Computer Science*, pages 55–74, 2002.

[88] B. Schneier. The weakest link (`https://www.schneier.com/blog/archives/2005/02/the_weakest_lin.html`)., 2005.

[89] J. Seligman. Internalization : The case of hybrid logics. *Journal of logic and computation*, 11(5) :671–689, 2001.

[90] D. B. Skillicorn and D. Talia. Models and Languages for Parallel Computation. *ACM Comput. Surv.*, 30(2) :123–169, June 1998.

[91] B. Toninho and L. Caires. A Spatial-Epistemic Logic for Reasoning about Security Protocols. In K. Chatzikokolakis and V. Cortier, editors, *Security Issues in Concurrency, SecCo 2010, Paris, France,*, volume 51 of *EPTCS*, pages 1–15, 2010.

[92] H. van Ditmarsch, J. Y. Halpern, W. W. van der Hoek, and B. P. Kooi, editors. *Handbook of epistemic logic*. College Publications, London, 2015.

[93] M. Y. Vardi. From Church and Prior to PSL. In O. Grumberg and H. Veith, editors, *25 Years of Model Checking - History, Achievements, Perspectives*, volume 5000 of *Lecture Notes in Computer Science*, pages 150–171. Springer, 2008.

# Annexe A

# Preuves du Chapitre 3

## A.1 Preuve du Lemme 3.8

**Lemme 3.8.** *Les règles d'extensions (Figure 3.4) préservent la réalisabilité.*

*Démonstration.*
Soit $\mathcal{T}$ un LTBI-tableau réalisable. Il contient un CTSS réalisable $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$. On montre que l'application d'une des règles de LTBI à $\mathcal{T}$ produit un tableau qui est réalisable. Soit $(\mathcal{K}, \lfloor . \rfloor, \lceil . \rceil)$ une réalisation de $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ avec le modèle linéaire $\mathcal{K} = (\mathcal{M}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ tel que $\mathcal{M} = (R, \bullet, e, \pi, \sqsubseteq, S)$ est un monoïde linéaire et $[\![\cdot]\!]$ est une interprétation linéaire. Soit $\mathcal{T}'$ le LTBI-tableau obtenu en appliquant une règle du calcul des tableaux pour LTBI sur une formule ou une contrainte de $\mathcal{T}$. On a trois cas :

1. La règle est appliquée sur une formule $\mathbb{S}\phi : (x,u)$ de $\mathcal{T}$. Si $\mathbb{S}\phi : (x,u) \notin \mathcal{F}$ alors $\mathcal{T}'$ est réalisable puisque l'application d'une règle ne modifie que le CTSS sur laquelle est appliquée, donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ n'est pas modifié et reste réalisable. Pour le cas où $\mathbb{S}\phi : (x,u) \in \mathcal{F}$, on prouve par cas selon la nature de $\phi$ :
   - $\mathbb{T}p : (x,u)$ $(p \in Prop)$
     Aucune règle ne peut être appliquée sur cette formule.
   - $\mathbb{F}p : (x,u)$ $(p \in Prop)$
     Aucune règle ne peut être appliquée sur cette formule.
   - $\mathbb{T}\mathrm{I} : (x,u)$
     Comme $(\mathcal{K}, \lfloor . \rfloor, \lceil . \rceil)$ est une réalisation de $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \mathrm{I}$. Alors, par définition $e \sqsubseteq \lfloor x \rfloor$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s \rangle$.
   - $\mathbb{F}\mathrm{I} : (x,u)$
     Aucune règle ne peut être appliquée sur cette formule.
   - $\mathbb{T}\top : (x,u)$
     Aucune règle ne peut être appliquée sur cette formule.
   - $\mathbb{F}\top : (x,u)$
     Aucune règle ne peut être appliquée sur cette formule.
   - $\mathbb{T}\bot : (x,u)$
     Aucune règle ne peut être appliquée sur cette formule.

- $\mathbb{F}\bot : (x,u)$
  Aucune règle ne peut être appliquée sur cette formule.

- $\mathbb{T}\phi \wedge \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi \wedge \psi$. Donc $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ et $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{F}\phi \wedge \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi \wedge \psi$. Donc $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ ou $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. C'est une réalisation d'au moins une des nouvelles branches $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$ et $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{T}\phi \vee \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi \vee \psi$. Donc $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ ou $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. C'est une réalisation d'au moins une des nouvelles branches $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$ et $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{F}\phi \vee \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi \vee \psi$. Donc $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ et $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u), \mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{T}\phi \to \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi \to \psi$. Alors par définition, pour tout $r \in R$, si $\lfloor x \rfloor \sqsubseteq r$ et $(r, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ alors $(r, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. En outre, par la condition de la règle, $x \leqslant y \in \overline{\mathcal{C}_r}$. Par le Lemme 3.6, $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$. Donc $(\lfloor y \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ ou $\lfloor y \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. C'est une réalisation d'au moins une des nouvelles branches $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$ et $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{F}\phi \to \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi \to \psi$. Alors, par définition, il existe $r \in R$ tel que $\lfloor x \rfloor \sqsubseteq r$, $(r, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ et $(r, \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. Comme $c_i$ est une nouvelle constante de ressources ($c_i \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$) et que la clôture de $\mathcal{C}_r$ n'introduit pas de nouvelles constantes de ressources, $\lfloor c_i \rfloor$ n'est pas défini. On peut alors étendre la réalisation par $\lfloor c_i \rfloor = r$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_i,u), \mathbb{F}\psi : (c_i,u)\}, \mathcal{C}_r \cup \{x \leqslant c_i\}, \mathcal{C}_s \rangle$.

- $\mathbb{T}\phi * \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi * \psi$. Alors, par définition, $\exists r', r'' \in R$ tels que $r' \bullet r'' \sqsubseteq \lfloor x \rfloor$ et $(r', \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ et $(r'', \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. Comme $c_i$ et $c_j$ sont des nouvelles constantes de ressources, $\lfloor c_i \rfloor$ et $\lfloor c_j \rfloor$ ne sont pas définis. On peut alors étendre la réalisation par $\lfloor c_i \rfloor = r'$ et $\lfloor c_j \rfloor = r''$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_i,u), \mathbb{T}\psi : (c_j,u)\}, \mathcal{C}_r\{c_i c_j \leqslant x\}, \mathcal{C}_s \rangle$.

- $\mathbb{F}\phi * \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi * \psi$. Alors, par définition, $\forall r', r'' \in R$ tels que $r' \bullet r'' \sqsubseteq \lfloor x \rfloor$, $(r', \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ ou $(r'', \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. Par la condition de la règle, $yz \leqslant x \in \overline{\mathcal{C}_r}$. Par le Lemme 3.6, $\lfloor y \rfloor \bullet \lfloor z \rfloor \sqsubseteq \lfloor x \rfloor$. Donc $(\lfloor y \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ ou $(\lfloor z \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. C'est une réalisation d'au moins une des nouvelles branches $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

- $\mathbb{T}\phi \mathrel{-\!\!*} \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$. Alors, par définition, $\forall r \in R$ tel que $(r, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$, $(\lfloor x \rfloor \bullet r, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. Par la condition de la règle, $xy \leqslant xy \in \overline{\mathcal{C}_r}$. Donc $\lfloor y \rfloor$ et $\lfloor xy \rfloor$ sont définis et $\lfloor xy \rfloor = \lfloor x \rfloor \bullet \lfloor y \rfloor$. Il y a deux cas :
  - $(\lfloor y \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi$ :
    Dans ce cas on a une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.
  - $(\lfloor y \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ :
    Alors $(\lfloor x \rfloor \bullet \lfloor y \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \psi$. C'est une réalisation de l'autre nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (xy,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.

  En conclusion, on a une réalisation d'au moins une des deux nouvelles branches.
- $\mathbb{F}\phi \mathrel{-\!\!*} \psi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$. Alors, par définition, $\exists r' \in R$ tel que $(r, \lceil u \rceil) \vDash_{\mathcal{K}} \phi$ et $(\lfloor x \rfloor \bullet r, \lceil u \rceil) \nvDash_{\mathcal{K}} \psi$. Comme $c_i$ est une nouvelle constante de label de ressources, $\lfloor c_i \rfloor$ est indéfini. On peut donc étendre notre réalisation en posant $\lfloor c_i \rfloor = r$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. En outre, par réflexivité de $\sqsubseteq$, $\lfloor x \rfloor \bullet \lfloor c_i \rfloor \sqsubseteq \lfloor x \rfloor \bullet \lfloor c_i \rfloor$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_i,u), \mathbb{F}\psi : (xc_i,u)\}, \mathcal{C}_r \cup \{xc_i \leqslant xc_i\}, \mathcal{C}_s \rangle$.
- $\mathbb{T}\Diamond\phi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \Diamond\phi$. Alors par définition, $\exists i \in S$ tel que $\lceil u \rceil \leq i$ et $(\lfloor x \rfloor, i) \vDash_{\mathcal{K}} \phi$. Comme $l_i$ est une nouvelle constante de label d'états, $\lceil l_i \rceil$ n'est pas défini. On peut donc étendre la réalisation par $\lceil l_i \rceil = i$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l_i)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l_i\} \rangle$.
- $\mathbb{F}\Diamond\phi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \Diamond\phi$. Alors, par définition, $\forall i \in S$ tel que $\lceil u \rceil \leq i$, $(\lfloor x \rfloor, i) \nvDash_{\mathcal{K}} \phi$. Par la condition de la règle, $u \lhd v \in \overline{\mathcal{C}_s}$. Par le Lemme 3.6, $\lceil u \rceil \leq \lceil v \rceil$. Donc $(\lfloor x \rfloor, \lceil v \rceil) \nvDash_{\mathcal{K}} \phi$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,v)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.
- $\mathbb{T}\Box\phi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \Box\phi$. Alors, par définition, $\forall i \in S$ tel que $\lceil u \rceil \leq i$, $(\lfloor x \rfloor, i) \vDash_{\mathcal{K}} \phi$. Par la condition de la règle, $u \lhd v \in \overline{\mathcal{C}_s}$. Par le Lemme 3.6, $\lceil u \rceil \leq \lceil v \rceil$. Donc $(\lfloor x \rfloor, \lceil v \rceil) \vDash_{\mathcal{K}} \phi$. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,v)\}, \mathcal{C}_r, \mathcal{C}_s \rangle$.
- $\mathbb{F}\Box\phi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \nvDash_{\mathcal{K}} \Box\phi$. Alors, par définition, $\exists i \in S$ tel que $\lceil u \rceil \leq i$ et $(\lfloor x \rfloor, i) \nvDash_{\mathcal{K}} \phi$. Comme $l_i$ est une nouvelle constante de label d'états, $\lceil l_i \rceil$ n'est pas défini. On peut donc étendre la réalisation par $\lceil l_i \rceil = i$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle branche $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l_i)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l_i\} \rangle$.
- $\mathbb{T}\bigcirc\phi : (x,u)$
  Dans ce cas, $(\lfloor x \rfloor, \lceil u \rceil) \vDash_{\mathcal{K}} \bigcirc\phi$. Donc $(\lfloor x \rfloor, \lceil u \rceil + 1) \vDash_{\mathcal{K}} \phi$. Comme $l_i$ est une nouvelle constante de label d'états, $\lceil l_i \rceil$ n'est pas défini. On peut donc étendre la réalisation par $\lceil l_i \rceil = \lceil u \rceil + 1$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l_i)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l_i\} \rangle$.

- $\mathbb{F}\bigcirc\phi : (x,u)$
  Dans ce cas, $(\lfloor x\rfloor, \lceil u\rceil) \nVdash_{\mathcal{K}} \bigcirc\phi$. Donc $(\lfloor x\rfloor, \lceil u\rceil + 1) \nVdash_{\mathcal{K}} \phi$. Comme $l_i$ est une nouvelle constante de label d'états, $\lceil l_i\rceil$ n'est pas défini. On peut donc étendre la réalisation par $\lceil l_i\rceil = \lceil u\rceil + 1$, ce qui préserve les conditions de la Définition 3.18 et du Lemme 3.6. C'est une réalisation de la nouvelle $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l_i)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l_i\}\rangle$.

2. La règle est $\langle\vartriangleleft\rangle$, appliquée sur un couple de contraintes d'états $u \vartriangleleft v$ et $u \vartriangleleft w$. Si $\{u \vartriangleleft v, u \vartriangleleft w\} \notin \overline{\mathcal{C}_s}$ alors $\mathcal{T}'$ est réalisable puisque la règle ne modifie que le CTSS sur laquelle elle est appliquée, donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ n'est pas modifié et reste réalisable.
   Dans le cas contraire, comme $\lceil v\rceil$ et $\lceil w\rceil$ sont des nombres naturels, on doit avoir $\lceil v\rceil \leq \lceil w\rceil$ ou $\lceil w\rceil \leq \lceil v\rceil$. On a donc une réalisation d'une des nouvelles branches $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \vartriangleleft w\}\rangle$ et $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \vartriangleleft v\}\rangle$

3. La règle est $\langle\blacktriangleleft\rangle$, appliquée sur un couple de contraintes d'états $u \blacktriangleleft v$ et $u \vartriangleleft w$. Si $\{u \blacktriangleleft v, u \vartriangleleft w\} \notin \overline{\mathcal{C}_s}$ alors $\mathcal{T}'$ est réalisable puisque la règle ne modifie que le CTSS sur laquelle elle est appliquée, donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ n'est pas modifié et reste réalisable.
   Dans le cas contraire, par le Lemme 3.6, on doit avoir $\lceil v\rceil = \lceil u\rceil + 1$ et $\lceil u\rceil \leq \lceil w\rceil$. On a alors deux cas :
   - $\lceil u\rceil = \lceil w\rceil$, on a alors une réalisation de la nouvelle branche $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \simeq w\}\rangle$.
   - $\lceil u\rceil < \lceil w\rceil$, alors $\lceil u\rceil \neq \lceil w\rceil$ et comme $\lceil v\rceil = \lceil u\rceil + 1$ on a $\lceil v\rceil \leq \lceil w\rceil$ (puisque $S \subseteq \mathbb{N}$). On a alors une réalisation de la nouvelle branche $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \vartriangleleft w, u \not\simeq w\}\rangle$.

   En conclusion, on a une réalisation d'au moins une des deux nouvelles branches.

□

## A.2 Preuve du Lemme 3.15

**Lemme 3.15.** *Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ un CTSS de Hintikka. Soit $\Omega(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle) = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ où $\mathcal{R} = (R, \bullet, e, \pi, \sqsubseteq, S)$. On a le résultat suivant pour toute formule $\phi$, pour tout label de ressources r et pour tout label d'états u :*

*(a)* $(\pi, \omega([u])) \vDash_{\mathcal{K}} \phi$.

*(b) Si* $\mathbb{F}\phi : (r,u) \in \mathcal{F}$ *et r est consistant, alors* $(r, \omega([u])) \nVdash_{\mathcal{K}} \phi$ .

*(c) Si* $\mathbb{T}\phi : (r,u) \in \mathcal{F}$ *et r est consistant, alors* $(r, \omega([u])) \vDash_{\mathcal{K}} \phi$.

*Démonstration.*
Par le Lemme 3.14, $(\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ est un modèle linéaire de ressources. La propriété (a) est validée par le Lemme 3.2. On prouve les deux autres propriétés simultanément par récurrence sur la taille de $\phi$. Notons que si $r$ est consistant, alors $r \neq \pi$, car $\pi$ n'est pas un label.

- Cas de base
  - $\mathbb{F}p : (r,u) \in \mathcal{F}$ tel que $p \in Prop$ :
    On suppose que $(r, \omega([u])) \vDash_{\mathcal{K}} p$. Alors $(r, \omega([u])) \in [\![p]\!]$. Par la Définition 3.21, $\exists r' \in R$ tel que $r' \sqsubseteq r$ et $\mathbb{T}p : (r', u) \in \mathcal{F}$. Comme $r' \sqsubseteq r$ et $r \neq \pi$, $r' \leqslant r \in \overline{\mathcal{C}_{r\omega}}$. Comme $\overline{\mathcal{C}_{r\omega}} \subseteq \overline{\mathcal{C}_r}$, on a $r' \leqslant r \in \overline{\mathcal{C}_r}$. Donc $\mathbb{T}p : (r', u) \in \mathcal{F}$, $\mathbb{F}p : (r,u) \in \mathcal{F}$ et $r' \leqslant r \in \overline{\mathcal{C}_r}$. C'est absurde car $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS de Hintikka (condition 1 de la Définition 3.19). Donc $(r, \omega([u])) \nVdash_{\mathcal{K}} p$.

- $\mathbb{T}p : (r,u) \in \mathcal{F}$ tel que $p \in Prop$ :
  Par définition de $\Omega$, et comme $\sqsubseteq$ est réflexive, $(r, \omega([u])) \in [\![p]\!]$. Donc $(r, \omega([u])) \vDash_{\mathcal{K}} p$.
- $\mathbb{F}\top : (r,u) \in \mathcal{F}$ :
  Dans ce cas, $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ n'est pas un CTSS de Hintikka (condition 3 de la Définition 3.19). Ce cas est absurde.
- $\mathbb{T}\top : (r,u) \in \mathcal{F}$ :
  Par définition de la relation de forcing, on a toujours $(r, \omega([u])) \vDash_{\mathcal{K}} \top$.
- $\mathbb{F}\bot : (r,u) \in \mathcal{F}$ :
  On suppose que $(r, \omega([u])) \vDash_{\mathcal{K}} \bot$. Alors, par définition, $\pi \sqsubseteq r$. Si $r \neq \pi$ alors par définition de $\Omega$, $\pi \leqslant r \in \overline{\mathcal{C}_{r\omega}}$ ce qui est absurde car $\pi \notin \mathcal{D}_r(\overline{\mathcal{C}_{r\omega}})$. Donc $(r, \omega([u])) \nvDash_{\mathcal{K}} \bot$.
- $\mathbb{T}\bot : (r,u) \in \mathcal{F}$ :
  Alors on doit avoir que $r$ est inconsistant, ce qui n'est pas possible par hypothèse. Donc ce cas est absurde.
- $\mathbb{F}\mathrm{I} : (r,u) \in \mathcal{F}$ :
  Par la condition 2 de la Définition 3.19, $1_r \leqslant r \notin \overline{\mathcal{C}_r}$. Comme $\overline{\mathcal{C}_{r\omega}} \subseteq \overline{\mathcal{C}_r}$, $1_r \leqslant r \notin \overline{\mathcal{C}_r}$. Par définition de $\Omega$ et comme $r \neq \pi$ on a $e \not\sqsubseteq r$. Donc $(r, \omega([u])) \nvDash_{\mathcal{K}} \mathrm{I}$.
- $\mathbb{T}\mathrm{I} : (r,u) \in \mathcal{F}$ :
  Par la condition 6 de la Définition 3.19, $1_r \leqslant r \in \overline{\mathcal{C}_r}$. Comme $r$ est consistant, par la Proposition 3.4, $1_r$ est consistant. Donc $1_r \leqslant r_r \in \overline{\mathcal{C}_{r\omega}}$. Donc $e \sqsubseteq r$ et on conclut que $(r, \omega([u])) \vDash_{\mathcal{K}} \mathrm{I}$.

- Hérédité : on suppose que (b) et (c) sont vraies pour toute formule $\phi$ et $\psi$ de taille $n$ (HR) et on montre la propriété pour des formules de taille $n+1$.
  - $\mathbb{F}\phi \wedge \psi : (r,u) \in \mathcal{F}$ :
    Par la condition 8 de la Définition 3.19, $\mathbb{F}\phi : (r,u) \in \mathcal{F}$ ou $\mathbb{F}\psi : (r,u) \in \mathcal{F}$. Alors par (HR), $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi$ ou $(r, \omega([u])) \nvDash_{\mathcal{K}} \psi$. Par suite $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi \wedge \psi$.
  - $\mathbb{T}\phi \wedge \psi : (r,u) \in \mathcal{F}$ :
    Preuve similaire.
  - $\mathbb{F}\phi \vee \psi : (r,u) \in \mathcal{F}$ :
    Preuve similaire.
  - $\mathbb{T}\phi \vee \psi : (r,u) \in \mathcal{F}$ :
    Preuve similaire.
  - $\mathbb{F}\phi \to \psi : (r,u) \in \mathcal{F}$ :
    Par la condition 12 de la Définition 3.19, il existe $r' \in L_r$ tel que $r \leqslant r' \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\phi : (r',u) \in \mathcal{F}$ et $\mathbb{F}\psi : (r',u) \in \mathcal{F}$. Il y a deux cas :
    - $r'$ est consistant :
      Par $\Omega$, la Proposition 3.4 et (HR), on a alors $r' \in R$ tel que $r \sqsubseteq r'$, $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$ et $(r', \omega([u])) \nvDash_{\mathcal{K}} \psi$. Donc $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi \to \psi$.
    - $r'$ est inconsistant :
      Dans ce cas, $\mathbb{F}\psi : (r',u) \in \mathcal{F}$ et $r'$ est inconsistant. C'est absurde par la condition 4 de la Définition 3.19.

- $\mathbb{T}\phi \to \psi : (r,u) \in \mathcal{F}$ :
  Soit $r' \in R$ tel que $r \sqsubseteq r'$. On suppose que $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$. Il y a deux cas :
  - $r' = \pi$ :
    Alors $(r', \omega([u])) \vDash_{\mathcal{K}} \psi$ par le Lemme 3.2.
  - $r' \neq \pi$ :
    Alors $r'$ est consistant. Comme $r \sqsubseteq r'$, par définition de $\Omega$, $r \leqslant r' \in \overline{C_r}$. Donc par la condition 11 de la Définition 3.19 et par (HR), $(r', \omega([u])) \nvDash_{\mathcal{K}} \phi$ ou $(r', \omega([u])) \vDash_{\mathcal{K}} \psi$. Comme on a supposé que $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$, alors $(r', \omega([u])) \vDash_{\mathcal{K}} \psi$.

  Finalement $(r, \omega([u])) \vDash_{\mathcal{K}} \phi \to \psi$.
- $\mathbb{F}\phi * \psi : (r,u) \in \mathcal{F}$ :
  Soient $y$ et $z$ tels que $y \bullet z \sqsubseteq r$. Comme $r \neq \pi$, on a $y \bullet z \neq \pi$, c'est à dire $y \neq \pi$ et $z \neq \pi$. Par suite, $y$ et $z$ sont des labels consistants. Par définition de $\Omega$, $yz \leqslant r \in \overline{C_r}$. Par la condition 14 de la Définition 3.19, $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ ou $\mathbb{F}\psi : (z,u) \in \mathcal{F}$. Par (HR), $(y, \omega([u])) \nvDash_{\mathcal{K}} \phi$ ou $(z, \omega([u])) \nvDash_{\mathcal{K}} \psi$. On conclut donc que $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi * \psi$.
- $\mathbb{T}\phi * \psi : (r,u) \in \mathcal{F}$ :
  Par la condition 13 de la Définition 3.19, il existe $r_1, r_2 \in L_r$ tels que $r_1 \circ r_2 \leqslant r \in \overline{C_r}$ et $\mathbb{T}\phi : (r_1,u) \in \mathcal{F}$ et $\mathbb{T}\psi : (r_2,u) \in \mathcal{F}$. Comme $r$ est consistant, par la Proposition 3.4 $r_1 \circ r_2$ est consistant, donc $r_1$ et $r_2$ sont consistants. Par (HR), $(r_1, s) \vDash_{\mathcal{K}} \phi$ et $(r_2, \omega([u])) \vDash_{\mathcal{K}} \psi$. Donc $(r, \omega([u])) \vDash_{\mathcal{K}} \phi * \psi$.
- $\mathbb{F}\phi \mathrel{-\!\!*} \psi : (r,u) \in \mathcal{F}$ :
  Par la condition 14 de la Définition 3.19, il existe $r' \in L_r$ tel que $r \circ r' \in \mathcal{D}_r(\overline{C_r})$, $\mathbb{T}\phi : (r',u) \in \mathcal{F}$ et $\mathbb{F}\psi : (r \circ r', u) \in \mathcal{F}$. Il y a deux cas :
  - $r \circ r'$ est consistant :
    Par la Proposition 3.4, $r'$ est consistant. Alors, par (HR), $r' \in R$ et $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$ et $(r \bullet r', \omega([u])) \nvDash_{\mathcal{K}} \psi$. Alors $(r, \omega([u])) \nvDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$.
  - $r \circ r'$ est inconsistant :
    Comme $\mathbb{F}\psi : (r \circ r', u) \in \mathcal{F}$, c'est absurde par la condition 4 de la Définition 3.19.
- $\mathbb{T}\phi \mathrel{-\!\!*} \psi : (r,u) \in \mathcal{F}$ :
  Soit $r' \in R$ tel que $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$. Il y a deux cas :
  - $r \circ r'$ est inconsistant :
    Alors $r \bullet r' = \pi$ et donc par le Lemme 3.2, $(r \bullet r', \omega([u])) \vDash_{\mathcal{K}} \psi$.
  - $r \circ r'$ est consistant :
    Alors, $r \circ r' \in \mathcal{D}_r(\overline{C_{r\omega}})$ et comme $\overline{C_{r\omega}} \subseteq \overline{C_r}$, on a $r \circ r' \in \mathcal{D}_r(\overline{C_r})$. De plus, par la Proposition 3.4, $r'$ est consistant. Alors par la condition 15 de la Définition 3.19, $\mathbb{F}\phi : (r',u)$ ou $\mathbb{T}\psi : (r \circ r', u)$. Par (HR), $(r', \omega([u])) \nvDash_{\mathcal{K}} \phi$ ou $(r \bullet r', \omega([u])) \vDash_{\mathcal{K}} \psi$. Par hypothèse, comme $(r', \omega([u])) \vDash_{\mathcal{K}} \phi$, alors $(r \bullet r', \omega([u])) \vDash_{\mathcal{K}} \psi$.

  En conclusion, $(r, \omega([u])) \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$.
- $\mathbb{F}\Diamond\phi : (r,u) \in \mathcal{F}$ :
  Soit $j \in S$ tel que $\omega([u]) \leq j$. Par définition de $\Omega$, il existe $v \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $j =$

$\omega([v])$. On a donc $\omega([u]) \leq \omega([v])$. Par le point 4 du Lemme 3.13, on a alors $u \lhd v \in \overline{\mathcal{C}_s}$. Par la condition 18 de la Définition 3.19 on a alors, $\mathbb{F}\phi : (r,v) \in \mathcal{F}$. Donc, par (HR), $(r,j) \nVdash_{\mathcal{K}} \phi$. Finalement $(r,\omega([u])) \nVdash_{\mathcal{K}} \Diamond\phi$.

- $\mathbb{T}\Diamond\phi : (r,u) \in \mathcal{F}$ :
  Par la condition 17 de la Définition 3.19, il existe $v \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $u \lhd v \in \overline{\mathcal{C}_s}$ et $\mathbb{T}\phi : (r,v) \in \mathcal{F}$. Par $\Omega$, le Lemme 3.13 et (HR), $\omega([u]) \leq \omega([v])$ et $r,\omega([v]) \vDash_{\mathcal{K}} \phi$. Donc $(r,\omega([u])) \vDash_{\mathcal{K}} \Diamond\phi$.
- $\mathbb{F}\Box\phi : (r,u) \in \mathcal{F}$ :
  Preuve similaire.
- $\mathbb{T}\Box\phi : (r,u) \in \mathcal{F}$ :
  Preuve similaire.
- $\mathbb{F}\bigcirc\phi : (r,u) \in \mathcal{F}$ :
  Par la condition 22 de la Définition 3.19, il existe $v \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ et $\mathbb{F}\phi : (r,v) \in \mathcal{F}$. Par le point 5 du Lemme 3.13, on a alors $\omega([v]) = \omega([u]) + 1$.
  Par $\Omega$ et (HR), on a alors $(r,\omega([v])) \nVdash_{\mathcal{K}} \phi$. Ainsi $(r,\omega([u])) \nVdash_{\mathcal{K}} \bigcirc\phi$.
- $\mathbb{T}\bigcirc\phi : (r,u) \in \mathcal{F}$ :
  Par la condition 21 de la Définition 3.19, il existe $v \in \mathcal{A}_s(\mathcal{C}_s)$ tel que $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ et $\mathbb{T}\phi : (r,v) \in \mathcal{F}$. Par le point 5 du Lemme 3.13, on a alors $\omega([v]) = \omega([u]) + 1$.
  Par $\Omega$ et (HR), on a alors $(r,\omega([v])) \vDash_{\mathcal{K}} \phi$. Ainsi $(r,\omega([u])) \vDash_{\mathcal{K}} \bigcirc\phi$.

□

## A.3 Preuve du Lemme 3.17

On donne ici une preuve du Lemme 3.17 qui assure qu'il existe un oracle qui contient tous les CTSS finis pour lesquels il n'y a pas de LTBI-tableau clos. C'est une variation de la preuve similaire pour DBI dans [34].

On va considérer différentes propriétés d'ensembles de CTSS, et en montrant progressivement que l'ensemble qu'on considère les possède, on montrera que c'est un oracle.

**Définition A.1** (Propriété de consistance)**.** *Une* propriété de consistance *est un ensemble de CTSS $\mathcal{P}$ qui satisfait les conditions suivantes pour tout CTSS, $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}$, pour tout $\phi, \psi \in \mathcal{L}$, pour tout $x \in L_r$ et pour tout $u,v,w \in L_s$ :*

1. $\mathbb{T}\phi : (x,u) \notin \mathcal{F}$ *ou* $\mathbb{F}\phi : (y,u) \notin \mathcal{F}$ *ou* $x \leqslant y \notin \overline{\mathcal{C}_r}$
2. $\mathbb{F}\mathrm{I} : (x,u) \notin \mathcal{F}$ *ou* $1_r \leqslant x \notin \overline{\mathcal{C}_r}$
3. $\mathbb{F}\top : (x,u) \notin \mathcal{F}$
4. $\mathbb{F}\phi : (x,u) \notin \mathcal{F}$ *ou $x$ est consistant*
5. $u \simeq v \notin \overline{\mathcal{C}_s}$ *ou* $u \not\simeq v \notin \overline{\mathcal{C}_s}$
6. *Si* $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}$, *alors* $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s \rangle \in \mathcal{P}$
7. *Si* $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$, *alors* $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}$
8. *Si* $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$, *alors* $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}$

9. *Si* $\mathbb{T}\phi \vee \psi : (x,u) \in \mathcal{F}$, *alors* $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
10. *Si* $\mathbb{F}\phi \vee \psi : (x,u) \in \mathcal{F}$, *alors* $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u), \mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
11. *Si* $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$, *alors* $\forall y \in L_r$, $x \leqslant y \in \overline{\mathcal{C}_r} \Rightarrow \langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
12. *Si* $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$, *alors* $\exists c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}$
13. *Si* $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$, *alors* $\exists c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $c_1 \neq c_2$ *et* $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}$
14. *Si* $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}$, *alors* $\forall y,z \in L_r$, $y \circ z \leqslant x \in \overline{\mathcal{C}_r} \Rightarrow \langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (z,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
15. *Si* $\mathbb{T}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}$, *alors* $\forall y \in L_r$, $xy \leqslant xy \in \overline{\mathcal{C}_r} \Rightarrow \langle \mathcal{F} \cup \{\mathbb{F}\phi : (y,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (xy,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
16. *Si* $\mathbb{F}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}$, *alors* $\exists c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (xc,u)\}, \mathcal{C}_r \cup \{xc \leqslant xc\}, \mathcal{C}_s\rangle \in \mathcal{P}$
17. *Si* $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$, *alors* $\exists l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}$
18. *Si* $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$, *alors* $\forall v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s} \Rightarrow \langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,v)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
19. *Si* $\mathbb{T}\Box\phi : (x,u) \in \mathcal{F}$, *alors* $\forall v \in L_s$, $u \lhd v \in \overline{\mathcal{C}_s} \Rightarrow \langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,v)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$
20. *Si* $\mathbb{F}\Box\phi : (x,u) \in \mathcal{F}$, *alors* $\exists l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}$
21. *Si* $\mathbb{T}\bigcirc\phi : (x,u) \in \mathcal{F}$, *alors* $\exists l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l\}\rangle \in \mathcal{P}$
22. *Si* $\mathbb{F}\bigcirc\phi : (x,u) \in \mathcal{F}$, *alors* $\exists l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l\}\rangle \in \mathcal{P}$
23. *Si* $u \lhd v \in \overline{\mathcal{C}_s}$ *et* $u \lhd w \in \overline{\mathcal{C}_s}$, *alors* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \in \mathcal{P}$
24. *Si* $u \blacktriangleleft v \in \overline{\mathcal{C}_s}$ *et* $u \lhd w \in \overline{\mathcal{C}_s}$, *alors* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \simeq w\}\rangle \in \mathcal{P}$ *ou* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w, u \not\simeq w\}\rangle \in \mathcal{P}$

Les conditions 1, 2, 3, 4 et 5 correspondent aux conditions qui assurent qu'un CTSS n'est pas clos. Les autres conditions assurent que si on applique une règle de la Figure 3.4, le CTSS obtenu est toujours dans $\mathcal{P}$. On peut remarquer que tout CTSS obtenu vérifie toujours la propriété ($P_{CSS}$) de la Définition 3.11.

**Définition A.2** (Propriété de consistance alternée)**.** *Une* propriété de consistance alternée *est un ensemble de CTSS* $\mathcal{P}$ *qui vérifie les conditions de la Définition A.1, à l'exception des conditions 12, 13, 16, 17, 20, 21 et 22, remplacées respectivement par les conditions 12', 13', 16', 17',20', 21' et 22' décrites ci-dessous.*

12'. *Si* $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$, *alors* $\forall c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}$
13'. *Si* $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$, *alors* $\forall c_1 \neq c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}$
16'. *Si* $\mathbb{F}\phi \mathrel{-\!\!*} \psi : (x,u) \in \mathcal{F}$, *alors* $\forall c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (xc,u)\}, \mathcal{C}_r \cup \{xc \leqslant xc\}, \mathcal{C}_s\rangle \in \mathcal{P}$

*17'.* *Si* $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$, *alors* $\forall l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}$

*20'.* *Si* $\mathbb{F}\Box\phi : (x,u) \in \mathcal{F}$, *alors* $\forall l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}$

*21'* *Si* $\mathbb{T}\bigcirc\phi : (x,u) \in \mathcal{F}$, *alors* $\forall l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l\}\rangle \in \mathcal{P}$

*22'* *Si* $\mathbb{F}\bigcirc\phi : (x,u) \in \mathcal{F}$, *alors* $\forall l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \blacktriangleleft l\}\rangle \in \mathcal{P}$

Ces nouvelles notions sont en relation avec le lemme que nous voulons prouver par la proposition suivante.

**Proposition A.1.** *L'ensemble de tous les CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos est une propriété de consistance.*

*Démonstration.*
Soit $\mathcal{P}$ l'ensemble de tous les CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos. On montre que $\mathcal{P}$ est une propriété de consistance. Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$.

1. Si $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$ alors le CTSS $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est clos. Donc $[\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle]$ est un LTBI-tableau clos. C'est contradictoire, puisque $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ (il n'existe pas de LTBI-tableau clos pour ce CTSS).
2. Similaire au cas 1.
3. Similaire au cas 1.
4. Similaire au cas 1.
5. Similaire au cas 1.
6. Si $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}$. On suppose que $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle \notin \mathcal{P}$ (c'est à dire qu'il existe un tableau clos pour ce CTSS). Par la règle $\langle \mathbb{T}\mathrm{I}\rangle$, $[\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}$.
7. Si $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{T}\wedge\rangle$, $[\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$.
8. Si $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$ et $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{F}\wedge\rangle$, $[\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle;\ \langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$.
9. Similaire au cas 8.
10. Similaire au case 7.
11. Si $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$. Soit $c \in L_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_r}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$ et $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{T}\to\rangle$, $[\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle;\ \langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Alors $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$ ou $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$.

12. Si $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$. On choisit $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$ (notons que $\gamma_r$ est infini et $\mathcal{A}_r(\mathcal{C}_r)$ est fini car $\mathcal{C}_r$ est fini). On suppose que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{F} \to \rangle$, $[\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}$.
13. Similaire au cas 12.
14. Similaire au cas 11.
15. Similaire au cas 11.
16. Similaire au cas 12.
17. Si $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$. On choisit $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$ (notons que $L_s$ est infini et $\mathcal{A}_s(\mathcal{C}_s)$ est fini car $\mathcal{C}_s$ est fini). On suppose que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{T}\Diamond\rangle$, $[\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \cup \{u \lhd l\}]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l), \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}$.
18. Si $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$. Soit $l \in L_s$ tel que $u \lhd l \in \overline{\mathcal{C}_u}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}$. Par la règle $\langle \mathbb{F}\Diamond\rangle$, $[\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Alors $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un LTBI-tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$.
19. Similaire au cas 18.
20. Similaire au cas 17.
21. Similaire au cas 17.
22. Similaire au cas 17.
23. Si $u \lhd v \in \overline{\mathcal{C}_s}$ et $u \lhd w \in \overline{\mathcal{C}_s}$. On suppose que $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \notin \mathcal{P}$ et $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \notin \mathcal{P}$. Par la règle $\langle \lhd \rangle$, $[\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle; \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle]$ est un LTBI-tableau pour $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ a un tableau clos, ce qui est contradictoire. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \in \mathcal{P}$.
24. Similaire au cas 23.

On conclut que $\mathcal{P}$ est une propriété de consistance. □

Nous allons maintenant montrer que nous pouvons étendre cet ensemble en une propriété de consistance alternée, puis en un Oracle.

**Proposition A.2.** *Toute propriété de consistance peut être étendue en une propriété de consistance $\preccurlyeq$-close (voir Définition 3.23).*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance. Soit $\mathcal{P}^{\preccurlyeq}$ sa $\preccurlyeq$-clôture définie par :

$$\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{\preccurlyeq} \text{ iff } \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \text{ for some } \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \in \mathcal{P}$$

On a $\mathcal{P}^{\preccurlyeq}$ qui contient $\mathcal{P}$ et qui est $\preccurlyeq$-close. On montre maintenant que $\mathcal{P}^{\preccurlyeq}$ est une propriété de consistance.

Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{\preccurlyeq}$. Alors il existe $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \in \mathcal{P}$ tel que $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle$.

1. On suppose que $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$. Alors $\mathbb{T}\phi : (x,u) \in \mathcal{F}'$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}'$ et $x \leqslant y \in \overline{\mathcal{C}_r}'$ car $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$. Mais cela est contradictoire car $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$ et $\mathcal{P}$ satisfont la condition 1 de la Définition A.1.
2. Similaire au cas 1.
3. Similaire au cas 1.
4. On suppose que $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ et $x$ est inconsistant. Alors il existe deux labels de ressources $y$ et $z$ tels que $yz \leqslant x \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (y,u) \in \mathcal{F}$. Alors $\mathbb{F}\phi : (x,u) \in \mathcal{F}'$, $yz \leqslant x \in \overline{\mathcal{C}'_r}$ et $\mathbb{T}\bot : (y,u) \in \mathcal{F}'$ puisque $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$. Par conséquent, $x$ et inconsistant dans $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$. En outre, puisque $\mathbb{F}\phi : (x,u) \in \mathcal{F}'$, ceci est contradictoire puisque $\mathcal{P}$ satisfait la condition 4 de la Définition A.1.
5. Similaire au cas 1.
6. On suppose que $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$ on a $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}'$. Donc $\langle \mathcal{F}', \mathcal{C}'_r \cup \{1_r \leqslant x\}, \mathcal{C}'_s \rangle \in \mathcal{P}$. Comme $\mathcal{F} \subseteq \mathcal{F}'$, $\mathcal{C}_r \cup \{1_r \leqslant x\} \subseteq \mathcal{C}'_r \cup \{1_r \leqslant x\}$ et $\mathcal{C}_s \subseteq \mathcal{C}'_s$, on a $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
7. On suppose que $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}'$. Donc $\langle \mathcal{F}' \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$. Mais $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}' \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle$. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
8. On suppose que $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}'$. Donc $\langle \mathcal{F}' \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}' \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$. Mais $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}' \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle$ et $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}' \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle$. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
9. Similaire au cas 8.
10. Similaire au cas 7.
11. On suppose que $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$. Soit $c \in L_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_r}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}'$ et $x \leqslant c \in \overline{\mathcal{C}'_r}$. Donc $\langle \mathcal{F}' \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}' \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$ ou $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
12. On suppose que $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}'$. Donc il existe $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}'_r)$ tel que $\langle \mathcal{F}' \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}'_r \cup \{x \leqslant c\}, \mathcal{C}'_s \rangle \in \mathcal{P}$. Comme $\mathcal{C}_r \subseteq \mathcal{C}'_r$, on a $\mathcal{A}_r(\mathcal{C}_r) \subseteq \mathcal{A}_r(\mathcal{C}'_r)$. Donc $\gamma_r \setminus \mathcal{A}_r(\mathcal{C}'_r) \subseteq \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$. Ainsi $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$. En outre $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
13. On suppose que $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}'$.Donc il existe $c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}'_r)$ tel que $c_1 \neq c_2$, $\langle \mathcal{F}' \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}'_r \cup \{c_1 c_2 \leqslant x\}, \mathcal{C}'_s \rangle \in \mathcal{P}$. Comme $\mathcal{C}_r \subseteq \mathcal{C}'_r$, on a $\mathcal{A}_r(\mathcal{C}_r) \subseteq \mathcal{A}_r(\mathcal{C}'_r)$. Donc $\gamma_r \setminus \mathcal{A}_r(\mathcal{C}'_r) \subseteq \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$. Par conséquent $c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, $c_1 \neq c_2$. En outre $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1 c_2 \leqslant x\}, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
14. On suppose que $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}$. Soit $c_1, c_2 \in L_r$ tel que $c_1 c_2 \leqslant x \in \overline{\mathcal{C}_r}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}'$ et $c_1 c_2 \leqslant x \in \overline{\mathcal{C}'_r}$. Donc $\langle \mathcal{F}' \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}' \cup \{\mathbb{T}\psi : (c_2,u)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$ ou $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.

15. Similaire au cas 14.
16. Similaire au cas 13.
17. On suppose que $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}'$. Donc il existe $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}'_s)$ tel que $\langle \mathcal{F}' \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}'_r, \mathcal{C}'_s \cup \{x \lhd l\} \rangle \in \mathcal{P}$. Comme $\mathcal{C}_s \subseteq \mathcal{C}'_s$ , on a $\mathcal{A}_s(\mathcal{C}_s) \subseteq \mathcal{A}_s(\mathcal{C}'_s)$. Alors $L_s \setminus \mathcal{A}_s(\mathcal{C}'_s) \subseteq L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$. Par conséquent $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$. En outre $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{x \lhd l\} \rangle \in \mathcal{P}^{\preccurlyeq}$.
18. On suppose que $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$. Soit $l \in L_s$ tel que $u \lhd l \in \overline{\mathcal{C}_s}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}'$ et $u \lhd l \in \overline{\mathcal{C}'_s}$. Donc $\langle \mathcal{F}' \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^{\preccurlyeq}$.
19. Similaire au cas 18.
20. Similaire au cas 17.
21. Similaire au cas 17.
22. Similaire au cas 17.
23. On suppose que $u \lhd v \in \overline{\mathcal{C}_s}$ et $u \lhd w \in \overline{\mathcal{C}_s}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $u \lhd v \in \overline{\mathcal{C}'_s}$ et $u \lhd w \in \overline{\mathcal{C}'_s}$. Comme $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}$, on a $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \cup \{v \lhd w\} \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \cup \{w \lhd v\} \rangle \in \mathcal{P}$. Mais $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\} \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \cup \{v \lhd w\} \rangle$ et $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\} \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \cup \{w \lhd v\} \rangle$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\} \rangle \in \mathcal{P}^{\preccurlyeq}$ ou $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\} \rangle \in \mathcal{P}^{\preccurlyeq}$.
24. Similaire au cas 23.

En conclusion, $\mathcal{P}^{\preccurlyeq}$ est une propriété de consistance. □

Pour étendre encore en une propriété de consistance alternée, on doit pouvoir généraliser les cas de la Définition A.1 aux cas de la Définition A.2. Pour cela, on cherche à montrer que l'on peut remplacer un label par un autre.

**Définition A.3** (Substitution)**.** *Une* substitution *est une fonction* $\tau : \gamma_r \cup \{1_r\} \cup L_s \longrightarrow \gamma_r \cup \{1_r\} \cup L_s$*, telle que :*

- $\forall c_i \in \gamma_r \cdot \tau(c_i) \in \gamma_r$
- $\forall l_i \in L_s \cdot \tau(l_i) \in L_s$
- $\tau(1_r) = 1_r$

*On étend cette définition à tous les labels de ressources comme suit. Soit* $c_{i_1} \circ c_{i_2} \circ \ldots \circ c_{i_k} \in L_r$, $\tau(c_{i_1} \circ c_{i_2} \circ \ldots \circ c_{i_k}) = \tau(c_{i_1}) \circ \tau(c_{i_2}) \circ \ldots \circ \tau(c_{i_k})$.

*On étend cette définition aux formules et contraintes par :*

- *Pour toute formule étiquetée* $\mathbb{S}\phi : (x,u)$, $\tau(\mathbb{S}\phi : (x,u)) = \mathbb{S}\phi : (\tau(x), \tau(u))$
- *Pour toute contrainte de ressources* $x \leqslant y$, $\tau(x \leqslant y) = \tau(x) \leqslant \tau(y)$
- *Pour toute contrainte d'état* $u \lhd v$, $\tau(u \star v) = \tau(u) \star \tau(v)$

*On étend cette définition aux CTSS par :*

- *Pour tout ensemble de formules étiquetées,* $\mathcal{F}$, $\tau(\mathcal{F}) = \{\tau(\mathbb{S}\phi : (x,u) \mid \mathbb{S}\phi : (x,u) \in \mathcal{F}\}$
- *Pour tout ensemble de contraintes de ressources* $\mathcal{C}_r$, $\tau(\mathcal{C}_r) = \{\tau(x \leqslant y) \mid x \leqslant y \in \mathcal{C}_r\}$

- *Pour tout ensemble de contraintes d'état* $\mathcal{C}_s$, $\tau(\mathcal{C}_s) = \{\tau(u \star v) \mid u \star v \in \mathcal{C}_s\}$
- *Pour tout CTSS* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$, $\tau(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle) = \langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle$

Cette définition s'étend par clôture :

**Lemme A.1.** *Soit* $\tau$ *une substitution,* $\mathcal{C}_r$ *un ensemble de contraintes de ressources et* $\mathcal{C}_s$ *un ensemble de contraintes d'état. On a les propriétés suivantes :*

1. $\tau(\overline{\mathcal{C}_r}) \subseteq \overline{\tau(\mathcal{C}_r)}$
2. $\tau(\overline{\mathcal{C}_s}) \subseteq \overline{\tau(\mathcal{C}_s)}$

*Démonstration.*
On montre la propriété 1, la preuve de 2 est similaire.

Soit $\mathcal{C}_\sim$ défini par $x \leqslant y \in \mathcal{C}_\sim$ ssi $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Pour tout $x \leqslant y \in \overline{\mathcal{C}_\sim}$, on montre que $x \leqslant y \in \mathcal{C}_\sim$ par récurrence sur la taille $n$ de l'arbre de preuve de $x \leqslant y$.

Cas de base $n = 0$ :
Dans ce cas $x \leqslant y \in \overline{\mathcal{C}_\sim}$ car $x \leqslant y \in \mathcal{C}_\sim$. La propriété est donc vraie.

Hérédité.
On suppose que toute contrainte de ressource de $\overline{\mathcal{C}_\sim}$ ayant un arbre de preuve de taille $n$ appartient à $\mathcal{C}_\sim$ (HR). On montre le même résultat pour toute contrainte de $\overline{\mathcal{C}_\sim}$ ayant un arbre de preuve de taille $n+1$. On discute selon la règle ayant permis d'obtenir la contrainte.

- $\langle t_r \rangle$ :
  L'arbre de preuve est de la forme $\dfrac{\overline{\dfrac{\ldots}{x \leqslant y}} \quad \overline{\dfrac{\ldots}{y \leqslant z}}}{x \leqslant z}\ \langle t_r \rangle$ . Par (HR), on a $x \leqslant y \in \mathcal{C}_\sim$ et $y \leqslant z \in \mathcal{C}_\sim$. Par définition, $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$ et $\tau(y) \leqslant \tau(z) \in \overline{\tau(\mathcal{C}_r)}$. Par la règle $\langle t_r \rangle$, $\tau(x) \leqslant \tau(z) \in \overline{\tau(\mathcal{C}_r)}$. Donc $x \leqslant z \in \mathcal{C}_\sim$.
- $\langle c_r \rangle$ :
  L'arbre de preuve est de la forme $\dfrac{\overline{\dfrac{\ldots}{ky \leqslant ky}} \quad \overline{\dfrac{\ldots}{x \leqslant y}}}{kx \leqslant ky}\ \langle c_r \rangle$ . Par (HR), on a $ky \leqslant ky \in \mathcal{C}_\sim$ et $x \leqslant y \in \mathcal{C}_\sim$. Par définition, $\tau(ky) \leqslant \tau(ky) \in \overline{\tau(\mathcal{C}_r)}$ et $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Donc $\tau(k) \circ \tau(y) \leqslant \tau(k) \circ \tau(y) \in \overline{\tau(\mathcal{C}_r)}$ et $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Par la règle $\langle c_r \rangle$, $\tau(k) \circ \tau(x) \leqslant \tau(k) \circ \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Donc $\tau(kx) \leqslant \tau(ky) \in \overline{\tau(\mathcal{C}_r)}$. Par suite $kx \leqslant ky \in \mathcal{C}_\sim$.
- Les autres cas sont similaires.

Ainsi $\overline{\mathcal{C}_\sim} \subseteq \mathcal{C}_\sim$. Par définition, $\tau(\mathcal{C}_r) \subseteq \overline{\tau(\mathcal{C}_r)}$. Soit maintenant $x \leqslant y \in \mathcal{C}_r$. On a alors $\tau(x) \leqslant \tau(y) \in \tau(\mathcal{C}_r)$. Comme $\tau(\mathcal{C}_r) \subseteq \overline{\tau(\mathcal{C}_r)}$, on a $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Par définition de $\mathcal{C}_\sim$, $x \leqslant y \in \mathcal{C}_\sim$. Donc $\mathcal{C}_r \subseteq \mathcal{C}_\sim$ et aussi $\overline{\mathcal{C}_r} \subseteq \overline{\mathcal{C}_\sim}$. Comme $\overline{\mathcal{C}_\sim} \subseteq \mathcal{C}_\sim$, on a $\overline{\mathcal{C}_r} \subseteq \mathcal{C}_\sim$. Soit $x \leqslant y \in \tau(\overline{\mathcal{C}_r})$, il existe $m \leqslant n \in \overline{\mathcal{C}_r}$ tel que $x = \tau(m)$ et $y = \tau(n)$. Comme $\overline{\mathcal{C}_r} \subseteq \mathcal{C}_\sim$ , on a $m \leqslant n \in \mathcal{C}_\sim$. Donc $\tau(m) \leqslant \tau(n) \in \overline{\tau(\mathcal{C}_r)}$. Par conséquent $x \leqslant y \in \overline{\tau(\mathcal{C}_r)}$.

On a montré que si $x \leqslant y \in \tau(\overline{\mathcal{C}_r})$, alors $x \leqslant y \in \overline{\tau(\mathcal{C}_r)}$. On peut conclure que $\tau(\overline{\mathcal{C}_r}) \subseteq \overline{\tau(\mathcal{C}_r)}$. □

**Corollaire A.1.** *Soit* $\tau$ *une substitution,* $\mathcal{C}_r$ *un ensemble de contraintes de ressources,* $\mathcal{C}_s$ *un ensemble de contraintes d'état. Les propriétés suivantes sont vérifiées :*

1. *Si* $x \leqslant y \in \overline{\mathcal{C}_r}$, *alors* $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$

2. *Si* $u \star v \in \overline{\mathcal{C}_s}$, *alors* $\tau(u) \star \tau(v) \in \overline{\tau(\mathcal{C}_s)}$

*Démonstration.*
On montre la propriété 1, la preuve de 2 est similaire.
Soit $x \leqslant y \in \overline{\mathcal{C}_r}$. Par définition, $\tau(x) \leqslant \tau(y) \in \tau(\overline{\mathcal{C}_r})$. Donc par le Lemme A.1, $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. □

**Proposition A.3.** *Soit* $\tau$ *be a substitution. On a les propriétés suivantes :*

1. *Si* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ *est un CTSS, alors* $\tau(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle)$ *est un CTSS.*
2. *Si* $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf} \rangle$ *est un CTSS fini, alors* $\tau(\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf} \rangle)$ *est un CTSS fini.*
3. *Si* $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$ *alors* $\tau(\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle) \preccurlyeq \tau(\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle)$ .

*Démonstration.*
Soit $\tau$ une substitution.

1. On doit montrer que $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle$ satisfait la propriété $(P_{CSS})$ de la Définition 3.11. Soit $\mathbb{S}\phi : (x,u) \in \tau(\mathcal{F})$. Par définition, il existe $\mathbb{S}\phi : (x',u') \in \mathcal{F}$ tel que $x = \tau(x')$ et $u = \tau(u')$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$ est un CTSS, on a $x' \leqslant x' \in \overline{\mathcal{C}_r}$ et $u' \lhd u' \in \overline{\mathcal{C}_s}$. Par le Corollaire A.1, $\tau(x') \leqslant \tau(x') \in \overline{\tau(\mathcal{C}_r)}$ et $\tau(u') \lhd \tau(u') \in \overline{\tau(\mathcal{C}_s)}$. Donc $x \leqslant x \in \overline{\tau(\mathcal{C}_r)}$ et $u \lhd u \in \overline{\tau(\mathcal{C}_s)}$. Par conséquent $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle$ est un CTSS.
2. Par le point précédent $\langle \tau(\mathcal{F}_f), \tau(\mathcal{C}_{rf}), \tau(\mathcal{C}_{sf}) \rangle$ est un CTSS. Soit $f : \mathcal{F}_f \longrightarrow \tau(\mathcal{F}_f)$ défini par $f(\mathbb{S}\phi : (x,u)) = \mathbb{S}\phi : (\tau(x), \tau(u))$. $f$ est surjective. Comme $\mathcal{F}_f$ est fini et qu'il y a une fonction surjective $f : \mathcal{F}_f \longrightarrow \tau(\mathcal{F}_f)$ alors $\tau(\mathcal{F}_f)$ est fini. Les preuves pour $\tau(\mathcal{C}_{rf})$ et $\tau(\mathcal{C}_{sf})$ sont similaires. Donc $\langle \tau(\mathcal{F}_f), \tau(\mathcal{C}_{rf}), \tau(\mathcal{C}_{sf}) \rangle$ est un CTSS fini.
3. Soit $\mathbb{S}\phi : (x,u) \in \tau(\mathcal{F})$. Il existe $m \in L_r$ et $n \in L_s$ tels que $\tau(m) = x$, $\tau(n) = u$ et $\mathbb{S}\phi : (m,n) \in \mathcal{F}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \preccurlyeq \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle$, on a $\mathbb{S}\phi : (m,n) \in \mathcal{F}'$. Donc $\mathbb{S}\phi : (x,y) \in \tau(\mathcal{F}')$. Les preuves pour $\tau(\mathcal{C}_r)$ et $\tau(\mathcal{C}_s)$ sont similaires. On conclut que $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \preccurlyeq \langle \tau(\mathcal{F}'), \tau(\mathcal{C}'_r), \tau(\mathcal{C}'_s) \rangle$.

□

**Proposition A.4.** *Soit* $\tau$ *une substitution, les propriétés suivantes sont vérifiées :*

- $\tau(\mathcal{F}) \cup \{\mathbb{S}\phi : (\tau(x), \tau(u))\} = \tau(\mathcal{F} \cup \{\mathbb{S}\phi : (x,u)\})$
- $\tau(\mathcal{C}_r) \cup \{\tau(x) \leqslant \tau(y)\} = \tau(\mathcal{C}_r \cup \{x \leqslant y\})$
- $\tau(\mathcal{C}_s) \cup \{\tau(u) \star \tau(v)\} = \tau(\mathcal{C}_s \cup \{u \star v\})$

*Démonstration.*
La preuve est directe vu la Définition A.3. □

On peut alors enfin conclure quant à l'extension en une propriété de consistance alternée.

**Proposition A.5.** *Toute propriété de consistance* $\preccurlyeq$*-close peut être étendue en une propriété de consistance alternée* $\preccurlyeq$*-close.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance $\preccurlyeq$-close. Soit $\mathcal{P}^+$ défini par :

$\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$ iff $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$ for a substitution $\tau : \gamma_r \cup \{1\} \cup L_s \longrightarrow \gamma_r \cup \{1\} \cup L_s$

$\mathcal{P}^+$ contient $\mathcal{P}$. En considérant la substitution identité, on a $\mathcal{P} \subseteq \mathcal{P}^+$.

On montre que $\mathcal{P}^+$ est $\preccurlyeq$-close. Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$. Soit $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle \preccurlyeq \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$, il existe une substitution $\tau$ telle que $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par la Proposition A.3, $\langle \tau(\mathcal{F}'), \tau(\mathcal{C}'_r), \tau(\mathcal{C}'_s) \rangle \preccurlyeq \langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, on a $\langle \tau(\mathcal{F}'), \tau(\mathcal{C}'_r), \tau(\mathcal{C}'_s) \rangle \in \mathcal{P}$. Donc $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s \rangle \in \mathcal{P}^+$.

On montre maintenant que $\mathcal{P}^+$ est une propriété de consistance alternée. Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$. Par définition, il existe une substitution $\tau$ telle que $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$.

1. On suppose que $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$. Par définition et le Corollaire A.1, $\mathbb{T}\phi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$ et $\mathbb{F}\phi : (\tau(y), \tau(u)) \in \tau(\mathcal{F})$ et $\tau(x) \leqslant \tau(y) \in \overline{\tau(\mathcal{C}_r)}$. Ceci est contradictoire car $\mathcal{P}$ est une propriété de consistance.
2. Similaire au cas 1.
3. Similaire au cas 1.
4. On suppose que $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ et $x$ est inconsistant. Alors il existe deux labels de ressource $y$ et $z$ tels que $yz \leqslant x \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (y,u) \in \mathcal{F}$.Par définition et le Corollaire A.1, $\mathbb{F}\phi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$, $\tau(y) \circ \tau(z) \leqslant \tau(x) \in \overline{\tau(\mathcal{C}_r)}$ et $\mathbb{T}\bot : (\tau(y), \tau(u)) \in \tau(\mathcal{F})$. Donc $x$ est inconsistant dans $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle$. En outre, comme $\mathbb{F}\phi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$, ceci est contradictoire car $\mathcal{P}$ est une propriété de consistance.
5. Similaire au cas 1.
6. On suppose que $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}$. Alors $\mathbb{T}\mathrm{I} : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r) \cup \{1_r \leqslant \tau(x)\}, \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Comme $\tau(1_r) = 1_r$ then $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r) \cup \{\tau(1_r) \leqslant \tau(x)\}, \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r \cup \{1_r \leqslant x\}), \tau(S) \rangle \in \mathcal{P}$. Donc $\langle \mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s \rangle \in \mathcal{P}^+$.
7. On suppose que $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Alors $\mathbb{T}\phi \wedge \psi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, $\langle \tau(\mathcal{F}) \cup \{\mathbb{T}\phi : (\tau(x), \tau(u)), \mathbb{T}\psi : (\tau(x), \tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$.
8. On suppose que $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Alors $\mathbb{F}\phi \wedge \psi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\phi : (\tau(x), \tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\psi : (\tau(x), \tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$.
9. Similaire au cas 8.
10. Similaire au cas 7.
11. On suppose que $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$. Alors $\mathbb{T}\phi \to \psi : (\tau(x), \tau(u)) \in \tau(\mathcal{F})$. Soit $c \in L_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_r}$. Par le Corollaire A.1, $\tau(x) \leqslant \tau(c) \in \overline{\tau(\mathcal{C}_r)}$. En outre, comme $\mathcal{P}$ est une propriété de consistance, $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\phi : (\tau(c), \tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F}) \cup \{\mathbb{T}\psi : (\tau(c), \tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$ ou $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s \rangle \in \mathcal{P}^+$.

12'. On suppose que $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$. Soit $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$. Alors $\mathbb{F}\phi \to \psi : (\tau(x),\tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, il existe $c' \in \gamma_r \setminus \mathcal{A}_r(\tau(\mathcal{C}_r))$ tel que $\langle \tau(\mathcal{F}) \cup \{\mathbb{T}\phi : (c',\tau(u)), \mathbb{F}\psi : (c',\tau(u))\}, \tau(\mathcal{C}_r) \cup \{\tau(x) \leqslant c'\}, \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Soit $\tau' = \tau[c \mapsto c']$. Comme $c \notin \mathcal{A}_r(\mathcal{C}_r)$, on a $\tau(\mathcal{C}_r) = \tau'(\mathcal{C}_r)$. En outre, par la propriété $(P_{CSS})$, $c$ n'apparaît pas dans $\mathcal{F}$, donc $\tau(\mathcal{F}) = \tau'(\mathcal{F})$ et $\tau(x) = \tau'(x)$. Évidemment, on a aussi $\tau(\mathcal{C}_s) = \tau'(\mathcal{C}_s)$. Par conséquent, $\langle \tau'(\mathcal{F}) \cup \{\mathbb{T}\phi : (\tau'(c),\tau'(u)), \mathbb{F}\psi : (\tau'(c),\tau'(u))\}, \tau'(\mathcal{C}_r) \cup \{\tau'(x) \leqslant \tau'(c)\}, \tau'(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau'(\mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}), \tau'(\mathcal{C}_r \cup \{x \leqslant c\}), \tau'(\mathcal{C}_s)\rangle \in \mathcal{P}$. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}^+$.

13'. On suppose que $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$. Soit $c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$, tels que $c_1 \neq c_2$. Alors $\mathbb{T}\phi * \psi : (\tau(x),\tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, il existe $c'_1, c'_2 \in \gamma_r \setminus \mathcal{A}_r(\tau(\mathcal{C}_r))$ tels que $c'_1 \neq c'_2$, $\langle \tau(\mathcal{F}) \cup \{\mathbb{T}\phi : (c'_1,\tau(u)), \mathbb{T}\psi : (c'_2,\tau(u))\}, \tau(\mathcal{C}_r) \cup \{c'_1 c'_2 \leqslant \tau(x)\}, \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Soit $\tau' = \tau[c_1 \mapsto c'_1, c_2 \mapsto c'_2]$. Comme $c_1 \notin \mathcal{A}_r(\mathcal{C}_r)$ et $c_2 \notin \mathcal{A}_r(\mathcal{C}_r)$, on a $\tau(\mathcal{C}_r) = \tau'(\mathcal{C}_r)$. En outre, par la propriété $(P_{CSS})$, $c_1$ et $c_2$ n'apparaissent pas dans $\mathcal{F}$, donc $\tau(\mathcal{F}) = \tau'(\mathcal{F})$ et $\tau(x) = \tau'(x)$. Évidemment, on a aussi $\tau(\mathcal{C}_s) = \tau'(\mathcal{C}_s)$. Donc $\langle \tau'(\mathcal{F}) \cup \{\mathbb{T}\phi : (\tau'(c_1),\tau'(u)), \mathbb{T}\psi : (\tau'(c_2),\tau'(u))\}, \tau'(\mathcal{C}_r) \cup \{\tau'(c_1)\tau'(c_2) \leqslant \tau'(x)\}, \tau'(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau'(\mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}), \tau'(\mathcal{C}_r \cup \{c_1 c_2 \leqslant x\}), \tau'(\mathcal{C}_s)\rangle \in \mathcal{P}$. Donc $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1 c_2 \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}^+$.

14. On suppose que $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}$. On a alors $\mathbb{F}\phi * \psi : (\tau(x),\tau(u)) \in \tau(\mathcal{F})$. Soit $c_1, c_2 \in L_r$ tels que $c_1 c_2 \leqslant x \in \overline{\mathcal{C}_r}$. Par le Corollaire A.1, $\tau(c_1 c_2) \leqslant \tau(x) \in \overline{\tau(\mathcal{C}_r)}$. Par définition des substitutions, $\tau(c_1)\tau(c_2) \leqslant \tau(x) \in \overline{\tau(\mathcal{C}_r)}$. En outre, comme $\mathcal{P}$ est une propriété de consistance, on a $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\phi : (\tau(c_1),\tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\psi : (\tau(c_2),\tau(u))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\psi : (c_1,u)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par conséquent, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^+$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^+$.

15. Similaire au cas 14.

16'. Similaire au cas 13'.

17'. On suppose que $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$. Soit $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$. Alors $\mathbb{T}\Diamond\phi : (\tau(x),\tau(u)) \in \tau(\mathcal{F})$. Comme $\mathcal{P}$ est une propriété de consistance, alors il existe $l' \in L_s \setminus \mathcal{A}_s(\tau(\mathcal{C}_s))$ tel que $\langle \tau(\mathcal{F}) \cup \{\mathbb{T}\phi : (\tau(x),l')\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \cup \{\tau(u) \lhd l'\}\rangle \in \mathcal{P}$. Soit $\tau' = \tau[l \mapsto l']$. Comme $l \notin \mathcal{A}_s(\mathcal{C}_s)$, on a $\tau(\mathcal{C}_s) = \tau'(\mathcal{C}_s)$. En outre, par la propriété $(P_{CSS})$, $l$ n'apparaît pas dans $\mathcal{F}$, donc $\tau(\mathcal{F}) = \tau'(\mathcal{F})$ et $\tau(u) = \tau'(u)$. Comme on a $\tau(\mathcal{C}_r) = \tau'(\mathcal{C}_r)$, $\langle \tau'(\mathcal{F}) \cup \{\mathbb{T}\phi : (\tau'(x),\tau'(l))\}, \tau'(\mathcal{C}_r), \tau'(\mathcal{C}_s) \cup \{\tau'(u) \lhd \tau'(l)\}\rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau'(\mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}), \tau'(\mathcal{C}_r), \tau'(\mathcal{C}_s \cup \{u \lhd l\})\rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}^+$.

18. On suppose que $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$. Alors $\mathbb{F}\Diamond\phi : (\tau(x),\tau(u)) \in \tau(\mathcal{F})$. Soit $l \in L_s$ tel que $u \lhd l \in \overline{\mathcal{C}_s}$. Par le Corollaire A.1, $\tau(u) \lhd \tau(l) \in \overline{\tau(\mathcal{C}_s)}$. En outre, comme $\mathcal{P}$ est une propriété de consistance, on a $\langle \tau(\mathcal{F}) \cup \{\mathbb{F}\phi : (\tau(x),\tau(l))\}, \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par la Proposition A.4, $\langle \tau(\mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s)\rangle \in \mathcal{P}$. Par conséquent $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^+$.

19. Similaire au cas 18.

20'. Similaire au cas 17'.

21'. Similaire au cas 17'.

22'. Similaire au cas 17'.

23 On suppose $u \lhd v \in \overline{\mathcal{C}_s}$ et $u \lhd w \in \overline{\mathcal{C}_s}$. Alors, par le Corollaire A.1, on a $\tau(u) \lhd \tau(v) \in \overline{\tau(\mathcal{C}_s)}$ et $\tau(u) \lhd \tau(w) \in \overline{\tau(\mathcal{C}_s)}$. Comme $\mathcal{P}$ est une propriété de consistance, $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \cup \{\tau(v) \lhd \tau(w)\}\rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s) \cup \{\tau(w) \lhd \tau(v)\}\rangle \in \mathcal{P}$. Par la Proposition A.4, on a $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s \cup \{v \lhd w\})\rangle \in \mathcal{P}$ ou $\langle \tau(\mathcal{F}), \tau(\mathcal{C}_r), \tau(\mathcal{C}_s \cup \{w \lhd v\})\rangle \in \mathcal{P}$. Finalement, par définition de $\mathcal{P}^+$ on a $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \in \mathcal{P}^+$ ou $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \in \mathcal{P}^+$.

24 Similaire au cas 23.

□

**Proposition A.6.** *Toute propriété de consistance alternée $\preccurlyeq$-close peut être étendue en une propriété de consistance alternée $\preccurlyeq$-close de caractère finie.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance alternée $\preccurlyeq$-close.

Soit $\mathcal{P}^{fc}$ défini par :

$$\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc} \text{ iff } \langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P} \text{ for all } \langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$$

On montre que $\mathcal{P}^{fc}$ contient $\mathcal{P}$.
Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est $\preccurlyeq$-close, $\mathcal{P}$ est $\preccurlyeq_f$-close. Donc, pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$ par définition. Ainsi $\mathcal{P} \subseteq \mathcal{P}^{fc}$.

On montre que $\mathcal{P}^{fc}$ est $\preccurlyeq$-clos.
Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \preccurlyeq \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Montrons que $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle$. Alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$, par définition. Ainsi $\langle \mathcal{F}', \mathcal{C}'_r, \mathcal{C}'_s\rangle \in \mathcal{P}^{fc}$ par définition.

On montre que $\mathcal{P}^{fc}$ est de caractère fini.
Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ un CTSS. On suppose que pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}^{fc}$. Montrons que $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. On a alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}^{fc}$ et ce CTSS et fini. En outre, $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ donc $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Ainsi $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

On montre maintenant que $\mathcal{P}^{fc}$ est une propriété de consistance alternée.
Soit $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

1. On suppose que $\mathbb{T}\phi : (x,u) \in \mathcal{F}$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}$ et $x \leqslant y \in \overline{\mathcal{C}_r}$. Par $(P_{CSS})$, $u \lhd u \in \overline{\mathcal{C}_s}$. Par compacité (Lemme 3.4), il existe $\mathcal{C}_{rf} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{sf} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{rf}$ et $\mathcal{C}_{sf}$ soit finis, $x \leqslant y \in \overline{\mathcal{C}_{rf}}$ et $u \lhd u \in \overline{\mathcal{C}_{sf}}$. Alors $\langle \{\mathbb{T}\phi : (x,u), \mathbb{F}\phi : (y,u)\}, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. On remarque par le Corollaire 3.1 que ce CTSS satisfait $(P_{CSS})$. Par définition $\langle \{\mathbb{T}\phi : (x,u), \mathbb{F}\phi : (y,u)\}, \mathcal{C}_{rf}, \mathcal{C}_{sf}\}\rangle \in \mathcal{P}$. Mais cela est contradictoire car $\mathcal{P}$ est une propriété de consistance alternée.
2. Similaire au cas 1.
3. Similaire au cas 1.
4. On suppose que $\mathbb{F}\phi : (x,u) \in \mathcal{F}$ et $x$ est inconsistant. Par $(P_{CSS})$, $u \lhd u \in \overline{\mathcal{C}_s}$. En outre, il existe deux labels $y$ et $z$ tels que $yz \leqslant x \in \overline{\mathcal{C}_r}$ et $\mathbb{T}\bot : (y,u) \in \mathcal{F}$. Par compacité (Lemme 3.4), il existe $\mathcal{C}_{rf} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{sf} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{rf}$ et $\mathcal{C}_{sf}$ soient finis, $yz \leqslant x \in \overline{\mathcal{C}_{rf}}$ et $u \lhd u \in$

$\overline{\mathcal{C}_{sf}}$. Donc $\langle\{\mathbb{F}\phi : (x,u), \mathbb{T}\bot : (y,u)\}, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. On remarque par le Corollaire 3.1 que ce CTSS satisfait ($P_{CSS}$). Par définition $\langle\{\mathbb{F}\phi : (x,u), \mathbb{T}\bot : (y,u)\}, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$ et $x$ est inconsistant dans ce CTSS. Mais cela est contradictoire car $\mathcal{P}$ est une propriété de consistance alternée.

5. Similaire au cas 1.
6. On suppose que $\mathbb{TI} : (x,u) \in \mathcal{F}$. Comme $\langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS, $x \leqslant x \in \overline{\mathcal{C}_r}$ et $u \lhd u \in \overline{\mathcal{C}_s}$. Par compacité (Lemme 3.4), il existe $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r0}$ et $\mathcal{C}_{s0}$ sont finis, $x \leqslant x \in \overline{\mathcal{C}_{r0}}$ et $u \lhd u \in \overline{\mathcal{C}_{s0}}$. On montre que $\langle\mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle$. Comme $\langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS et $\mathcal{F}_f \subseteq \mathcal{F}$ alors $\langle\mathcal{F}_f, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS. Par la Proposition 3.3, il existe $\mathcal{C}_{r1} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s1} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r1}$ et $\mathcal{C}_{s1}$ sont finis et $\langle\mathcal{F}_f, \mathcal{C}_{r1}, \mathcal{C}_{s1}\rangle$ est un CTSS. Soit $\mathcal{F}'_f = \mathcal{F}_f \cup \{\mathbb{TI} : (x,u)\}$, $\mathcal{C}'_{rf} = \mathcal{C}_{rf} \setminus \{1 \leqslant x\} \cup \mathcal{C}_{r0} \cup \mathcal{C}_{r1}$ et $\mathcal{C}'_{sf} = \mathcal{C}_{sf} \cup \mathcal{C}_{s0} \cup \mathcal{C}_{s1}$. Alors $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS fini et $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Par définition, $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est une propriété de consistance alternée et comme $\mathbb{TI} : (x,u) \in \mathcal{F}'_f$, on a $\langle\mathcal{F}'_f, \mathcal{C}'_{rf} \cup \{1_r \leqslant x\}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos et que $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle\mathcal{F}'_f, \mathcal{C}'_{rf} \cup \{1_r \leqslant x\}, \mathcal{C}'_{sf}\rangle$, on a $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Comme cela est vrai pour tout $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle$, on a $\langle\mathcal{F}, \mathcal{C}_r \cup \{1_r \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.
7. On suppose que $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Comme $\langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS, $x \leqslant x \in \overline{\mathcal{C}_r}$ et $u \lhd u \in \overline{\mathcal{C}_s}$. Par compacité (Lemme 3.4), il existe $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tel que $\mathcal{C}_{r0}$ et $\mathcal{C}_{s0}$ sont finis, $x \leqslant x \in \overline{\mathcal{C}_{r0}}$ et $u \lhd u \in \overline{\mathcal{C}_{s0}}$. On montre que $\langle\mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle\mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Soit $\mathcal{F}'_f = \mathcal{F}_f \setminus \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\} \cup \{\mathbb{T}\phi \wedge \psi : (x,u)\}$, $\mathcal{C}'_{rf} = \mathcal{C}_{rf} \cup \mathcal{C}_{r0}$ et $\mathcal{C}'_{sf} = \mathcal{C}_{sf} \cup \mathcal{C}_{s0}$. Alors
$\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS fini et $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Par définition, $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est une propriété de consistance alternée, et comme $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}'_f$, on a $\langle\mathcal{F}'_f \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos et comme $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle\mathcal{F}'_f \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$, alors $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Comme cela est vrai pour tout $\langle\mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle\mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$, alors $\langle\mathcal{F} \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.
8. On suppose que $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}$. Par contradiction, on suppose que $\langle\mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$ et $\langle\mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$. Alors par définition de $\mathcal{P}^{fc}$, il existe $\langle\mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq_f \langle\mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ et $\langle\mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq_f \langle\mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ tel que $\langle\mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \notin \mathcal{P}$ et $\langle\mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \notin \mathcal{P}$. Par compacité, il existe un ensemble fini $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ tel que $x \leqslant x \in \overline{\mathcal{C}_{r0}}$ et un ensemble fini $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tel que $u \lhd u \in \overline{\mathcal{C}_{s0}}$. Soit $\mathcal{F}'_f = \mathcal{F}^A_f \setminus \{\mathbb{F}\phi : (x,u)\} \cup \mathcal{F}^B_f \setminus \{\mathbb{F}\psi : (x,u)\} \cup \{\mathbb{F}\phi \wedge \psi : (x,u)\}$. Soit $\mathcal{C}'_{rf} = \mathcal{C}^A_{rf} \cup \mathcal{C}^B_{rf} \cup \mathcal{C}_{r0}$. Soit $\mathcal{C}'_{sf} = \mathcal{C}^A_{sf} \cup \mathcal{C}^B_{sf} \cup \mathcal{C}_{s0}$. Alors $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS et $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle\mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Alors, par définition de $\mathcal{P}^{fc}$, $\langle\mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}'_f$ et comme $\mathcal{P}$ est une propriété de consistance alternée, on a $\langle\mathcal{F}'_f \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$ ou $\langle\mathcal{F}'_f \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. On remarque que $\langle\mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq \langle\mathcal{F}'_f \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ et on a $\langle\mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq \langle\mathcal{F}'_f \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, $\langle\mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \in \mathcal{P}$ ou $\langle\mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \in \mathcal{P}$. C'est absurde. Donc $\langle\mathcal{F} \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$ ou $\langle\mathcal{F} \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

9. Similaire au cas 8.
10. Similaire au cas 7.
11. On suppose que $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}$. Soit $c \in L_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_r}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$ et $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$. Alors, par définition de $\mathcal{P}^{fc}$, il existe $\langle \mathcal{F}_f^A, \mathcal{C}_{rf}^A, \mathcal{C}_{sf}^A\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ et $\langle \mathcal{F}_f^B, \mathcal{C}_{rf}^B, \mathcal{C}_{sf}^B\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ tel que $\langle \mathcal{F}_f^A, \mathcal{C}_{rf}^A, \mathcal{C}_{sf}^A\rangle \notin \mathcal{P}$ et $\langle \mathcal{F}_f^B, \mathcal{C}_{rf}^B, \mathcal{C}_{sf}^B\rangle \notin \mathcal{P}$. Par compacité, il existe un ensemble fini $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_{r0}}$ et un ensemble fini $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tel que $u \lhd u \in \overline{\mathcal{C}_{s0}}$. Soit $\mathcal{F}_f' = \mathcal{F}_f^A \setminus \{\mathbb{F}\phi : (c,u)\} \cup \mathcal{F}_f^B \setminus \{\mathbb{T}\psi : (c,u)\} \cup \{\mathbb{T}\phi \to \psi : (x,u)\}$. Soit $\mathcal{C}_{rf}' = \mathcal{C}_{rf}^A \cup \mathcal{C}_{rf}^B \cup \mathcal{C}_{r0}$. Soit $\mathcal{C}_{sf}' = \mathcal{C}_{sf}^A \cup \mathcal{C}_{sf}^B \cup \mathcal{C}_{s0}$. Alors $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle$ est un CTSS et $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Alors, par définition de $\mathcal{P}^{fc}$, $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \in \mathcal{P}$. Comme $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}_f'$, $x \leqslant c \in \overline{\mathcal{C}_{rf}'}$ et comme $\mathcal{P}$ est une propriété de consistance alternée $\langle \mathcal{F}_f' \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}_f' \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \in \mathcal{P}$. On remarque que $\langle \mathcal{F}_f^A, \mathcal{C}_{rf}^A, \mathcal{C}_{sf}^A\rangle \preccurlyeq \langle \mathcal{F}_f' \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle$ et $\langle \mathcal{F}_f^B, \mathcal{C}_{rf}^B, \mathcal{C}_{sf}^B\rangle \preccurlyeq \langle \mathcal{F}_f' \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, $\langle \mathcal{F}_f^A, \mathcal{C}_{rf}^A, \mathcal{C}_{sf}^A\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}_f^B, \mathcal{C}_{rf}^B, \mathcal{C}_{sf}^B\rangle \in \mathcal{P}$. C'est absurde. Par conséquent, $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$ ou $\langle \mathcal{F} \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

12'. On suppose que $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}$. Soit $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS, $x \leqslant x \in \overline{\mathcal{C}_r}$ et $u \lhd u \in \overline{\mathcal{C}_s}$. Par compacité, il existe $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r0}$ et $\mathcal{C}_{s0}$ sont finis, $x \leqslant x \in \overline{\mathcal{C}_{r0}}$ et $u \lhd u \in \overline{\mathcal{C}_{s0}}$. On montre que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle$. Soit $\mathcal{F}_f' = \mathcal{F}_f \setminus \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\} \cup \{\mathbb{F}\phi \to \psi : (x,u)\}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS et $\mathcal{F}_f' \subseteq \mathcal{F}$, donc $\langle \mathcal{F}_f', \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS. Par la Proposition 3.3, il existe $\mathcal{C}_{r1} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s1} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r1}$ et $\mathcal{C}_{s1}$ sont finis et $\langle \mathcal{F}_f', \mathcal{C}_{r1}, \mathcal{C}_{s1}\rangle$ est un CTSS. Soit $\mathcal{C}_{rf}' = \mathcal{C}_{rf} \setminus \{x \leqslant c\} \cup \mathcal{C}_{r0} \cup \mathcal{C}_{r1}$ et $\mathcal{C}_{sf}' = \mathcal{C}_{sf} \cup \mathcal{C}_{s0} \cup \mathcal{C}_{s1}$. Alors $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle$ est un CTSS fini et $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Par définition, $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \in \mathcal{P}$. Alors, comme $\mathcal{C}_{rf}' \subseteq \mathcal{C}_r$ on a $\mathcal{A}_r(\mathcal{C}_{rf}') \subseteq \mathcal{A}_r(\mathcal{C}_r)$. Comme $\mathcal{P}$ est une propriété de consistance alternée, que $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}_f'$ et que $c \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_{rf}')$, on a $\langle \mathcal{F}_f' \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_{rf}' \cup \{x \leqslant c\}, \mathcal{C}_{sf}'\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ et $\preccurlyeq$-clos et que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle \mathcal{F}_f' \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_{rf}' \cup \{x \leqslant c\}, \mathcal{C}_{sf}'\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Comme cela est vrai pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle$, on a $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c,u), \mathbb{F}\psi : (c,u)\}, \mathcal{C}_r \cup \{x \leqslant c\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

13'. On suppose que $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}$. Soit $c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_r)$ tels que $c_1 \neq c_2$.
On montre que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_1,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$. Soit $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle$. Soit $\mathcal{F}_f' = \mathcal{F}_f \setminus \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\} \cup \{\mathbb{T}\phi * \psi : (x,u)\}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS et $\mathcal{F}_f' \subseteq \mathcal{F}$, alors $\langle \mathcal{F}_f', \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS. Par la Proposition 3.3, il existe $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r0}$ et $\mathcal{C}_{s0}$ sont finis et $\langle \mathcal{F}_f', \mathcal{C}_{r0}, \mathcal{C}_{s0}\rangle$ est un CTSS. Soit $\mathcal{C}_{rf}' = \mathcal{C}_{rf} \setminus \{c_1c_2 \leqslant x\} \cup \mathcal{C}_{r0}$ et $\mathcal{C}_{sf}' = \mathcal{C}_{sf} \cup \mathcal{C}_{s0}$. Alors $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle$ est un CTSS fini et $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Par définition, $\langle \mathcal{F}_f', \mathcal{C}_{rf}', \mathcal{C}_{sf}'\rangle \in \mathcal{P}$. En outre, comme $\mathcal{C}_{rf}' \subseteq \mathcal{C}_r$, on a $\mathcal{A}_r(\mathcal{C}_{rf}') \subseteq \mathcal{A}_r(\mathcal{C}_r)$. Comme $\mathcal{P}$ est une propriété de consistance alternée, que $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}_f'$ et que $c_1, c_2 \in \gamma_r \setminus \mathcal{A}_r(\mathcal{C}_{rf}')$ et $c_1 \neq c_2$, on a $\langle \mathcal{F}_f' \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_{rf}' \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_{sf}'\rangle \in$

$\mathcal{P}$. Comme $\mathcal{P}$ et $\preccurlyeq$-clos et que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}'_{rf} \cup \{c_1c_2 \leqslant x\}, \mathcal{C}'_{sf}\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Comme cela est vrai pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle$, on a $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (c_1,u), \mathbb{T}\psi : (c_2,u)\}, \mathcal{C}_r \cup \{c_1c_2 \leqslant x\}, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

14. On suppose que $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}$. Soit $c_1, c_2 \in L_r$ tels que $c_1c_2 \leqslant x \in \overline{\mathcal{C}_r}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$ et $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (c_2,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$. Alors, par définition de $\mathcal{P}^{fc}$, il existe $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{F}\psi : (c_2,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ tels que $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \notin \mathcal{P}$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \notin \mathcal{P}$. Par compacité, il existe un ensemble fini $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ tel que $c_1c_2 \leqslant x \in \overline{\mathcal{C}_{r0}}$ et un ensemble fini $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tel que $u \lhd u \in \overline{\mathcal{C}_{s0}}$. Soit $\mathcal{F}'_f = \mathcal{F}^A_f \setminus \{\mathbb{F}\phi : (c_1,u)\} \cup \mathcal{F}^B_f \setminus \{\mathbb{F}\psi : (c_2,u)\} \cup \{\mathbb{F}\phi * \psi : (x,u)\}$. Soit $\mathcal{C}'_{rf} = \mathcal{C}^A_{rf} \cup \mathcal{C}^B_{rf} \cup \mathcal{C}_{r0}$. Soit $\mathcal{C}'_{sf} = \mathcal{C}^A_{sf} \cup \mathcal{C}^B_{sf} \cup \mathcal{C}_{s0}$. Alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS et $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Alors, par définition $\mathcal{P}^{fc}$, $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}'_f$, $c_1c_2 \leqslant x \in \overline{\mathcal{C}'_{rf}}$ et que $\mathcal{P}$ est une propriété de consistance alternée, on a $\langle \mathcal{F}'_f \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}'_f \cup \{\mathbb{F}\psi : (c_2,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. On remarque que $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f \cup \{\mathbb{F}\psi : (c_2,u)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, on a $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \in \mathcal{P}$. Ceci est absurde. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (c_1,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$ ou $\langle \mathcal{F} \cup \{\mathbb{F}\psi : (c_2,u)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

15. Similaire au cas 14.

16'. Similaire au cas 13'.

17'. On suppose que $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}$. Soit $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}_s)$.
On montre que $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{x \lhd l\}\rangle \in \mathcal{P}^{fc}$. Let $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle$. Soit $\mathcal{F}'_f = \mathcal{F}_f \setminus \{\mathbb{T}\phi : (x,l)\} \cup \{\mathbb{T}\Diamond\phi : (x,u)\}$. Comme $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS et que $\mathcal{F}'_f \subseteq \mathcal{F}$, $\langle \mathcal{F}'_f, \mathcal{C}_r, \mathcal{C}_s\rangle$ est un CTSS. Par la Proposition 3.3, il existe $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ et $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tels que $\mathcal{C}_{r0}$ et $\mathcal{C}_{s0}$ sont finis et $\langle \mathcal{F}'_f, \mathcal{C}_{r0}, \mathcal{C}_{s0}\rangle$ est un CTSS. Soit $\mathcal{C}'_{rf} = \mathcal{C}_{rf} \cup \mathcal{C}_{r0}$ et $\mathcal{C}'_{sf} = \mathcal{C}_{sf} \setminus \{u \lhd l\} \cup \mathcal{C}_{s0}$. Alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS fini et $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Par définition, $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. En outre, comme $\mathcal{C}'_{sf} \subseteq \mathcal{C}_s$ on a $\mathcal{A}_s(\mathcal{C}'_{sf}) \subseteq \mathcal{A}_s(\mathcal{C}_s)$. Comme $\mathcal{P}$ est une propriété de consistance alternée, que $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}'_f$ et que $l \in L_s \setminus \mathcal{A}_s(\mathcal{C}'_{sf})$, on a $\langle \mathcal{F}'_f \cup \{\mathbb{T}\phi : (x,l)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{u \lhd l\}\rangle \in \mathcal{P}$. Comme $\mathcal{P}$ et $\preccurlyeq$-clos et que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f \cup \{\mathbb{T}\phi : (x,l), \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{u \lhd l\}\rangle$, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. Comme cela est vrai pour tout $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l), \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle$, on a $\langle \mathcal{F} \cup \{\mathbb{T}\phi : (x,l), \mathcal{C}_r, \mathcal{C}_s \cup \{u \lhd l\}\rangle \in \mathcal{P}^{fc}$.

18. On suppose que $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}$. Soit $l \in L_s$ tel que $u \lhd l \in \overline{\mathcal{C}_s}$. On suppose que $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \notin \mathcal{P}^{fc}$. Alors, par définition de $\mathcal{P}^{fc}$, il existe $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq_f \langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle$ tel que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \notin \mathcal{P}$. Par compacité, il existe un ensemble fini $\mathcal{C}_{r0} \subseteq \mathcal{C}_r$ tel que $x \leqslant x \in \overline{\mathcal{C}_{r0}}$ et un ensemble fini $\mathcal{C}_{s0} \subseteq \mathcal{C}_s$ tel que $u \lhd l \in \overline{\mathcal{C}_{s0}}$. Soit $\mathcal{F}'_f = \mathcal{F}_f \setminus \{\mathbb{F}\phi : (x,l)\} \cup \{\mathbb{F}\Diamond\phi : (x,u)\}$. Soit $\mathcal{C}'_{rf} = \mathcal{C}_{rf} \cup \mathcal{C}_{r0}$. Soit $\mathcal{C}'_{sf} = \mathcal{C}_{sf} \cup \mathcal{C}_{s0}$. Alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS et $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc, par définition de $\mathcal{P}^{fc}$, $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}'_f$, $u \lhd l \in \overline{\mathcal{C}'_{sf}}$ et comme $\mathcal{P}$ est une propriété de consistance alternée, on a $\langle \mathcal{F}'_f \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. On remarque que $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f \cup$

$\{\mathbb{F}\psi : (x,l)\}, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, on a $\langle \mathcal{F}_f, \mathcal{C}_{rf}, \mathcal{C}_{sf}\rangle \in \mathcal{P}$. C'est absurde. Donc $\langle \mathcal{F} \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_r, \mathcal{C}_s\rangle \in \mathcal{P}^{fc}$.

19. Similaire au cas 18.

20'. Similaire au cas 17.

21'. Similaire au cas 17.

22'. Similaire au cas 17.

23 On suppose que $u \lhd v \in \overline{\mathcal{C}_s}$ et $u \lhd w \in \overline{\mathcal{C}_s}$. Par l'absurde, on suppose que $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \notin \mathcal{P}^{fc}$ et $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \notin \mathcal{P}^{fc}$. Alors par définition de $\mathcal{P}^{fc}$, il existe $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle$ tel que $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \notin \mathcal{P}$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \notin \mathcal{P}$. Par compacité, il existe deux ensembles finis $\mathcal{C}_{s1}$ et $\mathcal{C}_{s2}$ tels que $u \lhd v \in \overline{\mathcal{C}_{s1}}$ et $u \lhd w \in \overline{\mathcal{C}_{s2}}$. Soit $\mathcal{F}'_f = \mathcal{F}^A_f \cup \mathcal{F}^B_f$, $\mathcal{C}'_{rf} = \mathcal{C}^A_{rf} \cup \mathcal{C}^B_{rf}$ et $\mathcal{C}'_{sf} = \mathcal{C}^A_{sf} \setminus \{v \lhd w\} \cup \mathcal{C}^B_{sf} \setminus \{w \lhd v\} \cup \mathcal{C}_{s1} \cup \mathcal{C}_{s2}$. Alors $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle$ est un CTSS et $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \preccurlyeq_f \langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s\rangle$. Donc, par définition, $\mathcal{P}^{fc}$, $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf}\rangle \in \mathcal{P}$. Comme $u \lhd v \in \overline{\mathcal{C}'_{sf}}$ et $u \lhd w \in \overline{\mathcal{C}'_{sf}}$, et comme $\mathcal{P}$ est une propriété de consistance alternée, on a $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{v \lhd w\}\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{w \lhd v\}\rangle \in \mathcal{P}$. On remarque que $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{v \lhd w\}\rangle$ et $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \preccurlyeq \langle \mathcal{F}'_f, \mathcal{C}'_{rf}, \mathcal{C}'_{sf} \cup \{w \lhd v\}\rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, on a $\langle \mathcal{F}^A_f, \mathcal{C}^A_{rf}, \mathcal{C}^A_{sf}\rangle \in \mathcal{P}$ ou $\langle \mathcal{F}^B_f, \mathcal{C}^B_{rf}, \mathcal{C}^B_{sf}\rangle \in \mathcal{P}$. C'est absurde. Donc $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{v \lhd w\}\rangle \in \mathcal{P}^{fc}$ ou $\langle \mathcal{F}, \mathcal{C}_r, \mathcal{C}_s \cup \{w \lhd v\}\rangle \in \mathcal{P}^{fc}$.

24 Similaire au cas 23.

□

**Lemme 3.17.** *Il existe un oracle qui contient tous les CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos.*

*Démonstration.*
On considère l'ensemble des CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos. Par la Proposition A.1, cet ensemble est une propriété de consistance. Par la Proposition A.2, on peut l'étendre en une propriété de consistance $\preccurlyeq$-close. Par la Proposition A.5, on peut l'étendre en une propriété de consistance alternée $\preccurlyeq$-close. Finalement, par la Proposition A.6, on peut l'étendre en une propriété de consistance alternée $\preccurlyeq$-close de caractère fini. Par les conditions 6 à 24 de la Définition A.2, l'ensemble construit est saturé. Par les conditions 1 à 5 de la même définition, c'est un ensemble de CTSS non clos. Donc cet ensemble est un oracle. Comme l'ensemble construit est une extension de l'ensemble des CTSS finis pour lesquels il n'existe pas de LTBI-tableau clos, il existe bien un oracle qui contienne cet ensemble. □

## A.4 Preuve de la Proposition 3.6

**Proposition 3.6.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathbb{F}\varphi : (1_r, l_1) \in \mathcal{F}_i$, $1_r \leqslant 1_r \in \mathcal{C}_{ri}$ *et* $l_1 \lhd l_1 \in \mathcal{C}_{si}$
2. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$, $\mathcal{C}_{ri} \subseteq \mathcal{C}_{ri+1}$ *et* $\mathcal{C}_{si} \subseteq \mathcal{C}_{si+1}$

3. $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle_{0 \leqslant i} \in \mathcal{P}$
4. $\mathcal{A}_r(\mathcal{C}_{ri}) \subseteq \{1_r, c_1, c_2, ..., c_{2i}\}$
5. $\mathcal{A}_s(\mathcal{C}_{si}) \subseteq \{l_1, l_2, ..., l_{i+1}\}$

*Démonstration.*

1. La propriété est vraie pour $i = 0$. Comme $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1} \rangle$ est obtenu par extension (par l'union $\cup$) de $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle$, cette propriété est vraie pour tout $i \geqslant 0$.
2. Cette propriété est vraie par les mêmes arguments que le point 1.

3, 4, 5. On prouve les propriétés 3, 4 et 5 simultanément par récurrence sur $i$.
Le cas de base ($i = 0$) est vrai, car $\langle \mathcal{F}_0, \mathcal{C}_{r0}, \mathcal{C}_{s0} \rangle = \langle \{\mathbb{F}\varphi : (1_r, l_1)\}, \{1_r \leqslant 1_r\}, \{l_1 \lhd l_1\} \rangle \in \mathcal{P}$ par hypothèse. En outre les points 4 et 5 sont trivialement vérifiés.
Comme hypothèse de récurrence (IH) on suppose que les points 3, 4 et 5 sont vrais pour $i = n$. On montre qu'elles restent vraies pour $i = n+1$ :

- Si $S_i$ est une formule étiquetée de la forme $\mathbb{S}F : (x, u)$ :
  - Si $\langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{rn}, \mathcal{C}_{sn} \rangle \notin \mathcal{P}$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n, \mathcal{C}_{rn}, \mathcal{C}_{sn} \rangle$. Donc les propriétés sont vraies dans ce cas par (IH).
  - Si $\langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{rn}, \mathcal{C}_{sn} \rangle \in \mathcal{P}$ alors comme c'est un CTSS, $x \leqslant x \in \overline{\mathcal{C}_{rn}}$ et $u \lhd u \in \overline{\mathcal{C}_{sn}}$ par $(P_{css})$. Donc $\gamma_r \cap \mathcal{E}(x) \subseteq \mathcal{A}_r(\mathcal{C}_{rn})$ **(1)**.
    - Si $\mathbb{S} = \mathbb{F}$ et $F = \phi \to \psi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{T}\phi : (c_{2n+1}, u), \mathbb{F}\psi : (c_{2n+1}, u)\}, \mathcal{C}_{rn} \cup \{x \leqslant c_{2n+1}\}, \mathcal{C}_{sn} \rangle$. Par (HI), $c_{2n+1} \notin \mathcal{A}_r(\mathcal{C}_{rn})$, donc c'est un nouveau label. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{F} \to \rangle$ et en utilisant le label $c_{2n+1}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn}) \cup \{c_{2n+1}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn})$. Donc, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
    - Si $\mathbb{S} = \mathbb{T}$ et $F = \phi * \psi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{T}\phi : (c_{2n+1}, u),\ \mathbb{T}\psi : (c_{2n+2}, u)\}, \mathcal{C}_{rn} \cup \{c_{2n+1}c_{2n+2} \leqslant x\}, \mathcal{C}_{sn} \rangle$. Par (HR), $c_{2n+1} \notin \mathcal{A}_r(\mathcal{C}_{rn})$ et $c_{2n+2} \notin \mathcal{A}_r(\mathcal{C}_{rn})$, donc ce sont de nouvelles constantes de ressources. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{T}* \rangle$ et en utilisant les labels $c_{2n+1}$ et $c_{2n+2}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn}) \cup \{c_{2n+1}, c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn})$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
    - Si $\mathbb{S} = \mathbb{F}$ et $F = \phi \mathrel{-\!\!*} \psi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{T}\phi : (c_{2n+1}, u), \mathbb{F}\psi : (xc_{2n+1}, u)\}, \mathcal{C}_{rn} \cup \{xc_{2n+1} \leqslant xc_{2n+1}\}, \mathcal{C}_{sn} \rangle$. Si (HR), $c_{2n+1} \notin \mathcal{A}_r(\mathcal{C}_{rn})$, donc c'est une nouvelle constante de ressources. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{F} \mathrel{-\!\!*} \rangle$ et en utilisant le label $c_{2n+1}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn}) \cup \{c_{2n+1}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn})$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
    - Si $\mathbb{S} = \mathbb{T}$ et $F = \mathrm{I}$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{rn} \cup \{1_r \leqslant x\}, \mathcal{C}_{sn} \rangle$. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{T}I \rangle$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn})$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn})$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.

- Si $\mathbb{S} = \mathbb{T}$ et $F = \Diamond\phi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{T}\phi : (x, l_{n+2})\}, \mathcal{C}_{rn},\ \mathcal{C}_{sn} \cup \{u \lhd l_{n+2}\} \rangle$. Par (HR), $l_{n+2} \notin \mathcal{A}_s(\mathcal{C}_{sn})$, donc c'est une nouvelle constante d'états. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{T}\Diamond \rangle$ et en utilisant le label $l_{n+2}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn})$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn}) \cup \{l_{n+2}\}$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
- Si $\mathbb{S} = \mathbb{F}$ et $F = \Box\phi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{F}\phi : (x, l_{n+2})\}, \mathcal{C}_{rn},\ \mathcal{C}_{sn} \cup \{u \lhd l_{n+2}\} \rangle$. Par (HR), $l_{n+2} \notin \mathcal{A}_s(\mathcal{C}_{sn})$, donc c'est une nouvelle constante d'états. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{F}\Box \rangle$ et en utilisant le label $l_{n+2}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn})$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn}) \cup \{l_{n+2}\}$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1 - r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
- Si $\mathbb{S} = \mathbb{T}$ et $F = \bigcirc\phi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{T}\phi : (x, l_{n+2})\}, \mathcal{C}_{rn},\ \mathcal{C}_{sn} \cup \{u \blacktriangleleft l_{n+2}\} \rangle$. Par (HR), $l_{n+2} \notin \mathcal{A}_s(\mathcal{C}_{sn})$, donc c'est une nouvelle constante d'états. Par saturation de $\mathcal{P}$ Par la règle $\langle \mathbb{T}\bigcirc \rangle$ et utilisant le label label $l_{n+2}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. PAr **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn})$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn}) \cup \{l_{n+2}\}$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
- Si $\mathbb{S} = \mathbb{F}$ et $F = \bigcirc\phi$ alors $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle = \langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\} \cup \{\mathbb{F}\phi : (x, l_{n+2})\}, \mathcal{C}_{rn},\ \mathcal{C}_{sn} \cup \{u \blacktriangleleft l_{n+2}\} \rangle$. Par (HR), $l_{n+2} \notin \mathcal{A}_s(\mathcal{C}_{sn})$, donc c'est une nouvelle constante d'états. Par saturation de $\mathcal{P}$ par la règle $\langle \mathbb{F}\bigcirc \rangle$ et en utilisant le label $l_{n+2}$, $\langle \mathcal{F}_{n+1}, \mathcal{C}_{rn+1}, \mathcal{C}_{sn+1} \rangle \in \mathcal{P}$. Par **(1)**, $\mathcal{A}_r(\mathcal{C}_{rn+1}) = \mathcal{A}_r(\mathcal{C}_{rn})$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) = \mathcal{A}_s(\mathcal{C}_{sn}) \cup \{l_{n+2}\}$. Alors, par (HR), $\mathcal{A}_r(\mathcal{C}_{rn+1}) \subseteq \{1_r, c_1, c_2, ..., c_{2n+2}\}$ et $\mathcal{A}_s(\mathcal{C}_{sn+1}) \subseteq \{l_1, l_2, ..., l_{n+2}\}$.
- Dans les autres cas, $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1} \rangle = \langle \mathcal{F}_i \cup \{\mathbb{S}_i F_i : (x_i, u_i)\}, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle$. Alors $\langle \mathcal{F}_n \cup \{\mathbb{S}F : (x,u)\}, \mathcal{C}_{rn}, \mathcal{C}_{sn} \rangle \in \mathcal{P}$, donc la propriété 3 est vraie. Les propriétés 4 et 5 sont vraies par (HR).

- Si $S_i$ est une contrainte d'états de la forme $u \star v$ :
  - Si $\{u, v\} \not\subseteq \{l_1, \ldots, l_{i+1}\}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1} \rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle$, alors les propriétés 3, 4 et 5 sont vraie par (HR).
  - Si $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\} \rangle \notin \mathcal{P}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1} \rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle$ , donc les propriétés 3, 4 et 5 sont vraie par (HR)..
  - Si $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\} \rangle \in \mathcal{P}$, alors on a $\langle \mathcal{F}_{i+1}, \mathcal{C}_{ri+1}, \mathcal{C}_{si+1} \rangle = \langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \cup \{u \star v\} \rangle$ et la propriété 3 est vraie. Par (HR), la propriété 4 est aussi vraie. Finalement, comme on est dans le cas où $\{u, v\} \subseteq \{l_1, \ldots, l_{i+1}\}$, par (HR), la propriété 5 est aussi vraie.

□

## A.5 Preuve du Lemme 3.18

**Lemme 3.18.** *Le CTSS limite est un CTSS de Hintikka.*

*Démonstration.*
Par la propriété 1 de la Proposition 3.7, $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$. On vérifie que toutes les conditions de la Définition 3.19 :

1. On suppose que $\mathbb{T}\phi : (x,u) \in \mathcal{F}_\infty$ et $\mathbb{F}\phi : (y,u) \in \mathcal{F}_\infty$ et $x \leqslant y \in \overline{\mathcal{C}_{r\infty}}$. Alors $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle$ est clos. Alors, par définition d'un oracle, $\langle \mathcal{F}_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \notin \mathcal{P}$. C'est absurde, donc la condition 1 est vérifiée.
2. Similaire à la condition 1.
3. Similaire à la condition 1.
4. Similaire à la condition 1.
5. Similaire à la condition 1.
6. On suppose que $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}_\infty$. Alors il existe $j \in \mathbb{N}$ tel que $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}_j$. En outre, il existe $k \geqslant j$ tel que le $k^{\text{ième}}$ élément de notre stratégie équitable est $\mathbb{T}\mathrm{I} : (x,u)$. Comme la suite $\langle \mathcal{F}_i, \mathcal{C}_{ri}, \mathcal{C}_{si} \rangle_{0 \leqslant i}$ est croissante (propriété 2 de la Proposition 3.6), on a $\mathbb{T}\mathrm{I} : (x,u) \in \mathcal{F}_k$. Par la propriété 3 de la Proposition 3.6, $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \rangle \in \mathcal{P}$. Donc $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1} \rangle = \langle \mathcal{F}_k, \mathcal{C}_{rk} \cup \{1_r \leqslant x\}, \mathcal{C}_{sk} \rangle$. Donc $1_r \leqslant x \in \overline{\mathcal{C}_{r\infty}}$.
7. On suppose que $\mathbb{T}\phi \wedge \psi : (x,u) \in \mathcal{F}_\infty$. Comme $\mathcal{P}$ est saturé, on a $\langle \mathcal{F}_\infty \cup \{\mathbb{T}\phi : (x,u), \mathbb{T}\psi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ par la règle $\langle \mathbb{T}\wedge \rangle$. Comme $\mathcal{P}$ est $\preccurlyeq$-clos, on a $\langle \mathcal{F}_\infty \cup \{\mathbb{T}\phi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ et $\langle \mathcal{F}_\infty \cup \{\mathbb{T}\psi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$. Par la propriété 2 de la Proposition 3.7, $\mathbb{T}\phi : (x,u) \in \mathcal{F}_\infty$ et $\mathbb{T}\psi : (x,u) \in \mathcal{F}_\infty$.
8. On suppose que $\mathbb{F}\phi \wedge \psi : (x,u) \in \mathcal{F}_\infty$. Comme $\mathcal{P}$ est saturé, $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\phi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\psi : (x,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ par la règle $\langle \mathbb{F}\wedge \rangle$. Par la propriété 2 de la Proposition 3.7, $\mathbb{F}\phi : (x,u) \in \mathcal{F}_\infty$ ou $\mathbb{F}\psi : (x,u) \in \mathcal{F}_\infty$.
9. Similaire à la condition 8.
10. Similaire à la condition 7.
11. On suppose que $\mathbb{T}\phi \to \psi : (x,u) \in \mathcal{F}_\infty$. Soit $c \in L_r$ tel que $x \leqslant c \in \overline{\mathcal{C}_{r\infty}}$. Comme $\mathcal{P}$ est saturé, $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}_\infty \cup \{\mathbb{T}\psi : (c,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ par la règle $\langle \mathbb{T} \to \rangle$. Par la propriété 2 de la Proposition 3.7, $\mathbb{F}\phi : (c,u) \in \mathcal{F}_\infty$ ou $\mathbb{T}\psi : (c,u) \in \mathcal{F}_\infty$.
12. On suppose que $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}_\infty$. Par un argument similaire à celui de la condition 6, il existe $k \in \mathbb{N}$ tel que :
    - le $k^{\text{ième}}$ élément de notre stratégie équitable est $\mathbb{F}\phi \to \psi : (x,u)$
    - $\mathbb{F}\phi \to \psi : (x,u) \in \mathcal{F}_k$
    - $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \rangle \in \mathcal{P}$.

    On a donc $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1} \rangle = \langle \mathcal{F}_k \cup \{\mathbb{T}\phi : (a,u), \mathbb{F}\psi : (a,u)\}, \mathcal{C}_{rk} \cup \{x \leqslant a\}, \mathcal{C}_{sk} \rangle$, où $a = c_{2k+1}$. Donc $x \leqslant a \in \overline{\mathcal{C}_{r\infty}}$, $\mathbb{T}\phi : (a,u) \in \mathcal{F}_\infty$ et $\mathbb{F}\psi : (a,u) \in \mathcal{F}_\infty$.
13. On suppose que $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}_\infty$. Par un argument similaire à celui de la condition 6, il existe $k \in \mathbb{N}$ tel que :
    - le $k^{\text{ième}}$ élément de notre stratégie équitable est $\mathbb{T}\phi * \psi : (x,u)$
    - $\mathbb{T}\phi * \psi : (x,u) \in \mathcal{F}_k$
    - $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \rangle \in \mathcal{P}$.

Donc $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1} \rangle = \langle \mathcal{F}_k \cup \{\mathbb{T}\phi : (a,u), \mathbb{T}\psi : (b,u)\}, \mathcal{C}_{rk} \cup \{ab \leqslant x\}, \mathcal{C}_{sk} \rangle$, où $a = c_{2k+1}$ et $b = c_{2k+2}$. Donc $ab \leqslant x \in \overline{\mathcal{C}_{r\infty}}$, $\mathbb{T}\phi : (a,u) \in \mathcal{F}_\infty$ et $\mathbb{T}\psi : (a,u) \in \mathcal{F}_\infty$.

14. On suppose que $\mathbb{F}\phi * \psi : (x,u) \in \mathcal{F}_\infty$. Soit $c,d \in L_r$ tel que $cd \leqslant x \in \overline{\mathcal{C}_{r\infty}}$. Comme $\mathcal{P}$ est saturé, on a $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\phi : (c,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ ou $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\psi : (d,u)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ par la règle $\langle \mathbb{F}* \rangle$. Par la propriété 2 de la Proposition 3.7, $\mathbb{F}\phi : (c,u) \in \mathcal{F}_\infty$ ou $\psi : (d,u) \in \mathcal{F}_\infty$.
15. Similaire à la condition 11.
16. Similaire à la condition 12.
17. On suppose que $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}_\infty$. Par un argument similaire à celui de la condition 6, il existe $k \in \mathbb{N}$ tel que :
    - le $k^{\text{ième}}$ élément de notre stratégie équitable est $\mathbb{T}\Diamond\phi : (x,u)$
    - $\mathbb{T}\Diamond\phi : (x,u) \in \mathcal{F}_k$
    - $\langle \mathcal{F}_k, \mathcal{C}_{rk}, \mathcal{C}_{sk} \rangle \in \mathcal{P}$.

    Alors $\langle \mathcal{F}_{k+1}, \mathcal{C}_{rk+1}, \mathcal{C}_{sk+1} \rangle = \langle \mathcal{F}_k \cup \{\mathbb{T}\phi : (x,c)\}, \mathcal{C}_{rk}, \mathcal{C}_{sk} \cup \{u \lhd c\} \rangle$, où $c = l_{k+2}$. Alors $u \lhd c \in \overline{\mathcal{C}_{s\infty}}$ et $\mathbb{T}\phi : (x,c) \in \mathcal{F}_\infty$.
18. On suppose que $\mathbb{F}\Diamond\phi : (x,u) \in \mathcal{F}_\infty$. Soit $l \in L_s$ tel que $u \lhd l \in \overline{\mathcal{C}_{s\infty}}$. Comme $\mathcal{P}$ est saturé, on a $\langle \mathcal{F}_\infty \cup \{\mathbb{F}\phi : (x,l)\}, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \rangle \in \mathcal{P}$ par la règle $\langle \mathbb{F}\Diamond \rangle$. Par la propriété 2 de la Proposition 3.7, $\mathbb{F}\phi : (x,l) \in \mathcal{F}_\infty$.
19. Similaire à la condition 18.
20. Similaire à la condition 17.
21. Similaire à la condition 17.
22. Similaire à la condition 17.
23. On suppose que $u \lhd v \in \overline{\mathcal{C}_{s\infty}}$ et $u \lhd w \in \overline{\mathcal{C}_{s\infty}}$. Comme $\mathcal{P}$ est saturé, alors $\langle F_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{v \lhd w\} \rangle \in \mathcal{P}$ ou $\langle F_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{w \lhd v\} \rangle \in \mathcal{P}$ par la règle $\langle \lhd \rangle$. Alors, par la propriété 3 de la Proposition 3.7, $v \lhd w \in \mathcal{C}_{s\infty}$ ou $w \lhd v \in \mathcal{C}_{s\infty}$.
24. On suppose que $u \blacktriangleleft v \in \overline{\mathcal{C}_{s\infty}}$ et $u \lhd w \in \overline{\mathcal{C}_{s\infty}}$. Comme $\mathcal{P}$ est saturé, alors $\langle F_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{u \simeq w\} \rangle \in \mathcal{P}$ ou $\langle F_\infty, \mathcal{C}_{r\infty}, \mathcal{C}_{s\infty} \cup \{v \lhd w, u \not\simeq w\} \rangle \in \mathcal{P}$ par la règle $\langle \blacktriangleleft \rangle$. Alors, par la propriété 3 de la Proposition 3.7, $u \simeq w \in \mathcal{C}_{s\infty}$ ou $v \lhd w \in \mathcal{C}_{s\infty}$ et $u \not\simeq w \in \mathcal{C}_{s\infty}$.

□

## A.6 Preuve du Lemme 3.21

**Lemme 3.21.** *Soit $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ un c-tableau. Soit $C \in \mathcal{V}$ un TPN réalisable. Parmi tous les TPN $C'$ tels que $(C,C') \in \mathcal{E}$ (c'est à dire les fils de C), au moins un est réalisable et le modèle de sa réalisation étend celui de C.*

*Démonstration.*
S'il existe $C'$ tel que $(C,C') \in \mathcal{E}$, alors $C$ a été étendu par une règle de la figure 3.12. On montre que quelque soit la règle, au moins un des TPN obtenus est réalisable. On note $C = \langle \mathcal{F}^+, \mathcal{F}^-, \mathcal{C}_r \rangle$. On note $\mathcal{R} = (\mathcal{K}, \lfloor . \rfloor, s)$ la réalisation de $C$.

- $\langle \mathrm{I}^- \rangle$ ou $\langle \bot \rangle$ ou $\langle \top \rangle$ ou $\langle \blacksquare \rangle$
  Dans ces cas là, d'après le Lemme 3.20, $C$ n'est pas réalisable. Ces cas ne sont donc pas possibles.

- $\langle \mathrm{I}^+ \rangle$
  Dans ce cas $\mathrm{I} : x \in \mathcal{F}^+$, et donc par définition $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \mathrm{I}$. Alors par la Définition 3.4, on a $e \sqsubseteq \lfloor x \rfloor$. Donc $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+ \setminus \{\mathrm{I} : x\}, \mathcal{F}^-, \mathcal{C}_r \cup \{1_r \leqslant x\} \rangle$, qui est l'unique fils de $C$.

- $\langle \wedge^+ \rangle$
  Dans ce cas $A \wedge B : x \in \mathcal{F}^+$. Alors par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \wedge B$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A$ et $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} B$. Donc $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+ \cup \{A : x, B : x\} \setminus \{A \wedge B : x\}, \mathcal{F}^-, \mathcal{C}_r \rangle$, qui est l'unique fils de $C$.

- $\langle \wedge^- \rangle$
  Dans ce cas $A \wedge B : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} A \wedge B$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} A$ ou $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} B$. Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{A : x\} \setminus \{A \wedge B : x\}, \mathcal{C}_r \rangle$ et $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{B : x\} \setminus \{A \wedge B : x\}, \mathcal{C}_r \rangle$, qui sont les deux seuls fils de $C$.

- $\langle \vee^+ \rangle$
  Dans ce cas $A \vee B : x \in \mathcal{F}^+$. Alors par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \vee B$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A$ ou $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} B$. Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+ \cup \{A : x\} \setminus \{A \vee B : x\}, \mathcal{F}^-, \mathcal{C}_r \rangle$ et $\langle \mathcal{F}^+ \cup \{B : x\} \setminus \{A \vee B : x\}, \mathcal{F}^-, \mathcal{C}_r \rangle$, qui sont les deux seuls fils de $C$.

- $\langle \vee^- \rangle$
  Dans ce cas $A \vee B : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} A \vee B$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} A$ et $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} B$. Donc $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{A : x, B : x\} \setminus \{A \vee B : x\}, \mathcal{C}_r \rangle$, qui est l'unique fils de $C$.

- $\langle \rightarrow^+ \rangle$
  Dans ce cas $A \rightarrow B : x \in \mathcal{F}^+$ et $x \leqslant y \in \overline{\mathcal{C}_r}$. Alors par définition, et par le Lemme 3.21, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \rightarrow B$ et $\lfloor x \rfloor \sqsubseteq \lfloor y \rfloor$. Donc par la Définition 3.4, on a que si $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} A$, alors $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} B$, c'est à dire que $(\lfloor y \rfloor, s) \nvDash_{\mathcal{K}} A$ ou $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} B$ Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+ \setminus \{A \rightarrow B : x\}, \mathcal{F}^- \cup \{A : x\}, \mathcal{C}_r \rangle$ et $\langle \mathcal{F}^+ \cup \{B : x\} \setminus \{A \rightarrow B : x\}, \mathcal{F}^-, \mathcal{C}_r \rangle$, qui sont les deux seuls fils de $C$.

- $\langle \rightarrow^- \rangle$
  Dans ce cas $A \rightarrow B : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nvDash_{\mathcal{K}} A \rightarrow B$. Donc par la Définition 3.4, il existe $r \in R$ tel que $x \sqsubseteq r$ et $(r, s) \vDash_{\mathcal{K}} A$ et $(r, s) \nvDash_{\mathcal{K}} B$. Comme $c_i$ est un nouveau label de ressources, $\lfloor c_i \rfloor$ n'est pas défini. On peut donc étendre $\lfloor . \rfloor$ en posant $\lfloor c_i \rfloor = r$. Alors cette extension de $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+ \cup \{A : c_i\}, \mathcal{F}^- \cup \{B : c_i\} \setminus \{A \rightarrow B : x\}, \mathcal{C}_r \cup \{x \leqslant c_i\} \rangle$, qui est l'unique fils de $C$.

- $\langle *^+ \rangle$
  Dans ce cas $A * B : x \in \mathcal{F}^+$. Alors par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A * B$. Donc par la Définition 3.4, il existe $r_1, r_2 \in R$ tel que $r_1 \bullet r_2 \sqsubseteq \lfloor x \rfloor$ et $(r_1, s) \vDash_{\mathcal{K}} A$ et $(r_2, s) \vDash_{\mathcal{K}} B$. Comme $c_i$ et $c_j$ sont de nouveaux labels de ressources, $\lfloor c_i \rfloor$ et $\lfloor c_j \rfloor$ ne sont pas définis. On peut donc étendre $\lfloor . \rfloor$ en posant $\lfloor c_i \rfloor = r_1$ et $\lfloor c_j \rfloor = r_2$. Alors cette extension de $\mathcal{R}$ est

une réalisation du nouveau TPN $\langle \mathcal{F}^+ \cup \{A : c_i, B : c_j\} \setminus \{A * B : x\}, \mathcal{F}^-, \mathcal{C}_r \cup \{c_i c_j \leqslant x\}\rangle$, qui est l'unique fils de $C$.

- $\langle \wedge^- \rangle$
Dans ce cas $A * B : x \in \mathcal{F}^-$ et $yz \leqslant x \in \overline{\mathcal{C}_r}$. Alors par définition, et par le Lemme 3.21, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A * B$ et $\lfloor y \rfloor \bullet \lfloor z \rfloor \sqsubseteq \lfloor x \rfloor$. Donc par la Définition 3.4, on a $(\lfloor y \rfloor, s) \nVdash_{\mathcal{K}} A$ ou $(\lfloor z \rfloor, s) \nVdash_{\mathcal{K}} B$. Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{A : y\} \setminus \{A * B : x\}, \mathcal{C}_r\rangle$ et $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{B : z\} \setminus \{A * B : x\}, \mathcal{C}_r\rangle$, qui sont les deux seuls fils de $C$.

- $\langle \mathrel{-\!\!*}^+ \rangle$
Dans ce cas $A \mathrel{-\!\!*} B : x \in \mathcal{F}^+$ et $xy \leqslant xy \in \overline{\mathcal{C}_r}$. Alors par définition, et par le Lemme 3.21, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \mathrel{-\!\!*} B$ et $\lfloor x \rfloor \bullet \lfloor y \rfloor \sqsubseteq \lfloor x \rfloor \bullet \lfloor y \rfloor$. Donc par la Définition 3.4, on a que si $(\lfloor y \rfloor, s) \vDash_{\mathcal{K}} A$, alors $(\lfloor x \rfloor \bullet \lfloor y \rfloor, s) \vDash_{\mathcal{K}} B$, c'est à dire que $(\lfloor y \rfloor, s) \nVdash_{\mathcal{K}} A$ ou $(\lfloor x \rfloor \bullet \lfloor y \rfloor, s) \vDash_{\mathcal{K}} B$ Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+ \setminus \{A \to B : x\}, \mathcal{F}^- \cup \{A : y\}, \mathcal{C}_r\rangle$ et $\langle \mathcal{F}^+ \cup \{B : xy\} \setminus \{A \to B : x\}, \mathcal{F}^-, \mathcal{C}_r\rangle$, qui sont les deux seuls fils de $C$.

- $\langle \mathrel{-\!\!*}^- \rangle$
Dans ce cas $A \mathrel{-\!\!*} B : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A \mathrel{-\!\!*} B$. Donc par la Définition 3.4, il existe $r \in R$ tel que $(r, s) \vDash_{\mathcal{K}} A$ et $(\lfloor x \rfloor \bullet r, s) \nVdash_{\mathcal{K}} B$. Comme $c_i$ est un nouveau label de ressources, $\lfloor c_i \rfloor$ n'est pas défini. On peut donc étendre $\lfloor . \rfloor$ en posant $\lfloor c_i \rfloor = r$. En outre, comme $\sqsubseteq$ est réflexive on a $\lfloor x \rfloor \bullet r \sqsubseteq \lfloor x \rfloor \bullet r$. Alors cette extension de $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+ \cup \{A : c_i\}, \mathcal{F}^- \cup \{B : xc_i\} \setminus \{A \mathrel{-\!\!*} B : x\}, \mathcal{C}_r \cup \{xc_i \leqslant xc_i\}\rangle$, qui est l'unique fils de $C$.

- $\langle \Box^+ \rangle$
Dans ce cas $\Box A : x \in \mathcal{F}^+$. Alors par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \Box A$. Alors par le Lemme 3.3, on a aussi $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \wedge \bigcirc\Box A$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A$ et $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \bigcirc\Box A$. Donc $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+ \cup \{A : x, \bigcirc\Box A : x\} \setminus \{\Box A : x\}, \mathcal{F}^-, \mathcal{C}_r\rangle$, qui est l'unique fils de $C$.

- $\langle \Box^- \rangle$
Dans ce cas $\Box A : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \Box A$. Alors par le Lemme 3.3, on a aussi $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A \wedge \bigcirc\Box A$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A$ ou $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \bigcirc\Box A$. Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{A : x\} \setminus \{\Box A : x\}, \mathcal{C}_r\rangle$ et $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{\bigcirc\Box A : x\} \setminus \{\Box A : x\}, \mathcal{C}_r\rangle$, qui sont les deux seuls fils de $C$.

- $\langle \Diamond^+ \rangle$
Dans ce cas $\Diamond A : x \in \mathcal{F}^+$. Alors par définition, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \Diamond A$. Alors par le Lemme 3.3, on a aussi $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A \vee \bigcirc\Diamond A$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} A$ ou $(\lfloor x \rfloor, s) \vDash_{\mathcal{K}} \bigcirc\Diamond A$. Donc $\mathcal{R}$ est une réalisation d'un des nouveaux TPN $\langle \mathcal{F}^+ \cup \{A : x\} \setminus \{\Diamond A : x\}, \mathcal{F}^-, \mathcal{C}_r\rangle$ et $\langle \mathcal{F}^+ \cup \{\bigcirc\Diamond A : x\} \setminus \{\Diamond A : x\}, \mathcal{F}^-, \mathcal{C}_r\rangle$, qui sont les deux seuls fils de $C$.

- $\langle \Diamond^- \rangle$
Dans ce cas $\Diamond A : x \in \mathcal{F}^-$. Alors par définition, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \Diamond A$. Alors par le Lemme 3.3, on a aussi $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A \vee \bigcirc\Diamond A$. Donc par la Définition 3.4, on a $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} A$ et $(\lfloor x \rfloor, s) \nVdash_{\mathcal{K}} \bigcirc\Diamond A$. Donc $\mathcal{R}$ est une réalisation du nouveau TPN $\langle \mathcal{F}^+, \mathcal{F}^- \cup \{A : x, \bigcirc\Diamond A : x\} \setminus \{\Diamond A : x\}, \mathcal{C}_r\rangle$, qui est l'unique fils de $C$.

- $\langle \bigcirc \rangle$
  Dans ce cas $\mathcal{F}^+ = \bigcup\{\bigcirc A_i : x_i\} \cup \bigcup\{p_j : x_j\}$ et $\mathcal{F}^- = \bigcup\{\bigcirc B_i : x_i\} \cup \bigcup\{q_j : x_j\}$. Alors on a en particulier que pour tout $i$, $(\lfloor x_i \rfloor, s) \vDash_{\mathcal{K}} \bigcirc A_i$ et pour tout $j$, $(\lfloor x_j \rfloor, s) \nvDash_{\mathcal{K}} \bigcirc B_i$. On peut alors étendre $\mathcal{K}$ en $\mathcal{K}'$ en posant $S' = S \cup \{s+1\}$. On pose alors $\mathcal{R}' = (\mathcal{K}', \lfloor . \rfloor, s+1)$. Dans ce modèle on a alors pour tout $i$, $(\lfloor x_i \rfloor, s+1) \vDash_{\mathcal{K}} A_i$ et pour tout $j$, $(\lfloor x_j \rfloor, s+1) \nvDash_{\mathcal{K}} B_i$ par la Définition 3.4. Alors $\mathcal{R}'$ est une réalisation du nouveau TPN $\langle \bigcup\{A_i : x_i\}, \bigcup\{B_i : x_i\}, C_r \rangle$, qui est l'unique fils de $C$.

□

# Annexe B

# Preuves du Chapitre 5

## B.1 Preuve du Lemme 5.7

**Lemme 5.7.** *Les règles du calcul des tableaux pour HRL (Figure 5.1) préservent la réalisabilité.*

*Démonstration.*
Soit $\mathcal{T}$ un tableau réalisable. Par définition, il contient une branche réalisable $\mathcal{F}$. Soit $\mathcal{K} = (\mathcal{R}, [\![\cdot]\!], \vDash_{\mathcal{K}})$ le modèle qui réalise $\mathcal{F}$.

On étend $\mathcal{T}$ par une des règles de la Figure 5.1. Si la branche étendue n'est pas $\mathcal{F}$, alors $\mathcal{F}$ reste une branche du nouveau tableau et ce dernier reste réalisable. Dans le cas contraire, c'est $\mathcal{F}$ qui est étendue et on discute suivant la règle appliquée :

- $\langle \mathbb{T}\mathrm{I} \rangle$
  Par définition on a $e \vDash_{\mathcal{K}} \mathrm{I}$ car $e \sim e$, donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_e(\mathrm{I})\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{T}\wedge \rangle$
  Alors on a $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi \wedge \psi$, ce qui veut dire que $x \vDash_{\mathcal{K}} \phi$ et $x \vDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F}\wedge \rangle$
  Alors on a $\mathbb{F}\ @_x(\phi \wedge \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi \wedge \psi$, ce qui veut dire que ou bien $x \nvDash_{\mathcal{K}} \phi$, ou bien $x \nvDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation d'au moins une des deux nouvelles branches $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\}$ et $\mathcal{F} \cup \{\mathbb{F}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{T}\vee \rangle$
  Alors on a $\mathbb{T}\ @_x(\phi \vee \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi \vee \psi$, ce qui veut dire que ou bien $x \vDash_{\mathcal{K}} \phi$, ou bien $x \vDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation d'au moins une des deux nouvelles branches $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi)\}$ et $\mathcal{F} \cup \{\mathbb{T}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F}\vee \rangle$
  Alors on a $\mathbb{F}\ @_x(\phi \vee \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi \vee \psi$, ce qui veut dire que $x \nvDash_{\mathcal{K}} \phi$ et $x \nvDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi), \mathbb{F}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.

- $\langle \mathbb{T} \rightarrow \rangle$
  Alors on a $\mathbb{T}\ @_x(\phi \rightarrow \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi \rightarrow \psi$, ce qui veut dire que ou bien $x \nvDash_{\mathcal{K}} \phi$, ou bien $x \vDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation d'au moins une des deux nouvelles branches $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\}$ et $\mathcal{F} \cup \{\mathbb{T}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F} \rightarrow \rangle$
  Alors on a $\mathbb{F}\ @_x(\phi \rightarrow \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi \rightarrow \psi$, ce qui veut dire que $x \vDash_{\mathcal{K}} \phi$ et $x \nvDash_{\mathcal{K}} \psi$. Alors $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{F}\ @_x(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{T} \neg \rangle$
  Alors on a $\mathbb{T}\ @_x(\neg\phi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \neg\phi$, ce qui veut dire que $x \nvDash_{\mathcal{K}} \phi$. Alors $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F} \neg \rangle$
  Alors on a $\mathbb{F}\ @_x(\neg\phi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \neg\phi$, ce qui veut dire que $x \vDash_{\mathcal{K}} \phi$. Alors $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{T} * \rangle$
  Alors on a $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi * \psi$ ce qui veut dire qu'il existe $r_1, r_2 \in Nom$ tels que $r_1 \bullet r_2 \downarrow$, $x \sim r_1 \bullet r_2$, $r_1 \vDash_{\mathcal{K}} \phi$ et $r_2 \vDash_{\mathcal{K}} \psi$. Comme $c_i, c_j \notin \mathcal{A}(\mathcal{F})$, $\sim$ n'est pas défini pour ces nominaux dans $\mathcal{K}$[9]. On définit donc le modèle $\mathcal{K}'$ identique à $\mathcal{K}$ mais où $c_i \sim r_1$ et $c_j \sim r_2$. Alors on a par compatibilité que $c_i \bullet c_j \downarrow$ et que $x \sim c_i \bullet c_j$, c'est à dire que $x \vDash_{\mathcal{K}'} c_i * c_j$ par le premier point du Lemme 5.1. En outre, par monotonie, on a que $c_i \vDash_{\mathcal{K}'} \phi$ et $c_j \vDash_{\mathcal{K}'} \psi$. Finalement, $\mathcal{K}'$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F} * \rangle$
  Alors on a $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}$ et $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi * \psi$ et $x \vDash_{\mathcal{K}} y * z$. Ce dernier point nous indique par le premier point du Lemme 5.1 que $y \bullet z \downarrow$ et $x \sim y \bullet z$. Or, comme $x \nvDash_{\mathcal{K}} \phi * \psi$, pour tout $r, r' \in Nom$ on a ou bien $r \bullet r' \uparrow$, ou bien $x \not\sim r \bullet r'$ ou bien $r \nvDash_{\mathcal{K}} \phi$, ou bien $r' \nvDash_{\mathcal{K}} \psi$. Comme $y \bullet z \downarrow$ et $x \sim y \bullet z$, on a ou bien $y \nvDash_{\mathcal{K}} \phi$, ou bien $z \nvDash_{\mathcal{K}} \psi$. Finalement, $\mathcal{K}$ est une réalisation d'au moins une des deux nouvelles branches $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}$ et $\mathcal{F} \cup \{\mathbb{F}\ @_z(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{T} \mathrel{-\!\!*} \rangle$
  Alors on a $\mathbb{T}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$ et $\mathbb{T}\ @_z(x * y) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$ et $z \vDash_{\mathcal{K}} x * y$. Ce dernier point nous indique par le premier point du Lemme 5.1 que $x \bullet y \downarrow$ et $z \sim x \bullet y$. Or, comme $x \vDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$, pour tout $r \in Nom$ tel que $x \bullet r \downarrow$ et $r \vDash_{\mathcal{K}} \phi$, on a $x \bullet r \vDash_{\mathcal{K}} \psi$ . Comme $x \bullet y \downarrow$, on a donc pour $r = y$ que si $y \vDash_{\mathcal{K}} \phi$, alors $x \bullet y \vDash_{\mathcal{K}} \psi$. On a donc ou bien $y \nvDash_{\mathcal{K}} \phi$, ou bien $x \bullet y \vDash_{\mathcal{K}} \psi$. Dans ce deuxième cas, par monotonie et comme $z \sim x \bullet y$ on a aussi $z \vDash_{\mathcal{K}} \psi$. Finalement, $\mathcal{K}$ est une réalisation d'au moins une des deux nouvelles branches $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}$ et $\mathcal{F} \cup \{\mathbb{T}\ @_z(\psi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle \mathbb{F} \mathrel{-\!\!*} \rangle$
  Alors on a $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi \mathrel{-\!\!*} \psi$. Il existe donc

9. Dans le cas où *sim* est défini, l'absence de $c_i$ et $c_j$ dans $\mathcal{F}$ assure que le modèle identique à $\mathcal{K}$ sauf concernant $c_i$ et $c_j$ est aussi une réalisation de $\mathcal{F}$ et on peut travailler avec celui-ci sans perte de validité de notre raisonnement.

$r \in Nom$ tel que $x \bullet r \downarrow$ et $r \vDash_{\mathcal{K}} \phi$ et $x \bullet r \nvDash_{\mathcal{K}} \psi$. Comme $c_i \notin \mathcal{A}(\mathcal{F})$, on peut définir un modèle $\mathcal{K}'$ identique à $\mathcal{K}$ et tel que $r \sim c_i$. On a alors par compatibilité que $x \bullet c_i \downarrow$ et par monotonie $c_i \vDash_{\mathcal{K}'} \phi$. Comme $c_j \notin \mathcal{A}(\mathcal{F})$, on peut aussi définir dans $\mathcal{K}'$ que $c_j \sim x \bullet c_i$ et on a par monotonie que $c_j \nvDash_{\mathcal{K}'} \psi$. Enfin, $c_j \sim x \bullet c_i$ donc $c_j \vDash_{\mathcal{K}'} x * c_i$ par le premier point du Lemme 5.1. Finalement, $\mathcal{K}'$ est une réalisation de la nouvelle branche $\mathcal{A}(\mathcal{F}) \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{F}\ @_{c_j}(\psi), \mathbb{T}\ @_{c_j}(x * c_i)\}$ et $\mathcal{T}$ reste réalisable.

- $\langle \mathbb{T}* \leftarrow \rangle$
  Preuve identique à $\langle \mathbb{T} \rightarrow * \rangle$ à commutation près entre $x$ et $y$.
- $\langle \mathbb{F}* \leftarrow \rangle$
  Preuve identique à $\langle \mathbb{F} \rightarrow * \rangle$ à commutation près entre $x$ et $c_i$.
- $\langle @ \rangle$
  Il y a deux cas :
  - Si $\mathbb{T}\ @_x(@_y(\phi)) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} @_y(\phi)$ et donc $y \vDash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi)\}$ et $\mathcal{T}$ reste réalisable.
  - Si $\mathbb{F}\ @_x(@_y(\phi)) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} @_y(\phi)$ et donc $y \nvDash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle i_r \rangle$
  Cette règle stipule qu'on peut générer la formule $\mathbb{T}\ @_x(x)$ dans $\mathcal{F}$. En effet, comme $\sim$ est réflexive, quelque soit le nominal $x$ on a $x \sim x$ et donc $x \vDash_{\mathcal{K}} x$. Ainsi $\mathcal{K}$ est une réalisation de $\mathcal{F} \cup \{\mathbb{T}\ @_x(x)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle i_s \rangle$
  On a $\mathbb{T}\ @_x(y) \in \mathcal{F}$. Comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} y$, donc $x \sim y$. Comme $\sim$ est symétrique, on a donc $y \sim x$ et donc $y \vDash_{\mathcal{K}} x$. Ainsi $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_y(x)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle i_t \rangle$
  Il y a deux cas :
  - Si $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{T}\ @_y(s) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi$ et $y \vDash_{\mathcal{K}} x$. De ce dernier point on tire $y \sim x$ et donc par monotonie $y \vDash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi)\}$ et $\mathcal{T}$ reste réalisable.
  - Si $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{T}\ @_y(x) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \nvDash_{\mathcal{K}} \phi$ et $x \vDash_{\mathcal{K}} y$. De ce dernier point on tire $y \sim x$ et donc par monotonie $y \nvDash_{\mathcal{K}} \phi$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}$ et $\mathcal{T}$ reste réalisable.
- $\langle i+ \rangle$
  Il y a deux cas :
  - Si $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi$. Comme $c_i \notin \mathcal{A}(\mathcal{F})$, on peut définir un modèle $\mathcal{K}'$ identique à $\mathcal{K}$ et tel que $c_i \sim y \bullet z$. On a alors par le premier point du Lemme 5.1 que $c_i \vDash_{\mathcal{K}'} y * z$ et par le point 1 du Lemme 5.3 on a que $x \vDash_{\mathcal{K}'} \phi[y * z / c_i]$. Donc $\mathcal{K}'$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y * z / c_i]), \mathbb{T}\ @_{c_i}(y * z)\}$ et $\mathcal{T}$ reste réalisable.

- Si $\mathbb{F}\ @_x(\phi[y*z]) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \nVdash_{\mathcal{K}} \phi$. Comme $c_i \notin \mathcal{A}(\mathcal{F})$, on peut définir un modèle $\mathcal{K}'$ identique à $\mathcal{K}$ et tel que $c_i \sim y \bullet z$. On a alors par le premier point du Lemme 5.1 que $c_i \vDash_{\mathcal{K}'} y*z$ et par le point 1 du Lemme 5.3 on a que $x \nVdash_{\mathcal{K}'} \phi[y*z/c_i]$. Donc $\mathcal{K}'$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y*z/c_i]), \mathbb{T}\ @_{c_i}(y*z)\}$ et $\mathcal{T}$ reste réalisable.

- $\langle i- \rangle$
  Il y a deux cas :
  - Si $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ et $\mathbb{T}\ @_y(z*t) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi$ et $y \vDash_{\mathcal{K}} z*t$. Par le premier point du Lemme 5.1, on a alors $y \sim z \bullet t$ et par le deuxième point du Lemme 5.3 on a par suite $x \vDash_{\mathcal{K}} \phi[y/z*t]$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y/z*t])\}$ et $\mathcal{T}$ reste réalisable.
  - Si $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ et $\mathbb{T}\ @_y(z*t) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \nVdash_{\mathcal{K}} \phi$ et $y \vDash_{\mathcal{K}} z*t$. Par le premier point du Lemme 5.1, on a alors $y \sim z \bullet t$ et par le deuxième point du Lemme 5.3 on a par suite $x \nVdash_{\mathcal{K}} \phi[y/z*t]$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y/z*t])\}$ et $\mathcal{T}$ reste réalisable.

- $\langle i_p \rangle$
  Il y a deux cas :
  - Si $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ et $\mathbb{T}\ @_y(z) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \vDash_{\mathcal{K}} \phi$ et $y \vDash_{\mathcal{K}} z$. On a alors $y \sim z$ et par le Lemme 5.2 on a par suite $x \vDash_{\mathcal{K}} \phi[y/z]$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y/z])\}$ et $\mathcal{T}$ reste réalisable.
  - Si $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ et $\mathbb{T}\ @_y(z) \in \mathcal{F}$, comme $\mathcal{F}$ est réalisable, on a $x \nVdash_{\mathcal{K}} \phi$ et $y \vDash_{\mathcal{K}} z$. On a alors $y \sim z$ et par le Lemme 5.2 on a par suite $x \nVdash_{\mathcal{K}} \phi[y/z]$. Donc $\mathcal{K}$ est une réalisation de la nouvelle branche $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y/z])\}$ et $\mathcal{T}$ reste réalisable.

□

## B.2 Preuve du Lemme 5.10

**Lemme 5.10.** *Soit $\mathcal{F}$ un SHS de Hintikka et $\mathcal{K} = \Omega(\mathcal{F})$. Pour toute formule $\phi \in \mathcal{L}$ et pour tout $x \in \mathcal{A}(\mathcal{F})$ on a :*

- *Si $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$, alors $x \vDash_{\mathcal{K}} \phi$*
- *Si $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$, alors $x \nVdash_{\mathcal{K}} \phi$*

*Démonstration.*
On démontre par récurrence sur la taille de $\phi$.

- Cas de base On montre que le lemme est vrai pour les formules de taille 1.
  - Si $\mathbb{T}\ @_x(p) \in \mathcal{F}$ avec $p \in Prop$.
    Alors par la Définition 5.12, on a $x \in [\![p]\!]$ et donc $x \vDash_{\mathcal{K}} p$.
  - Si $\mathbb{F}\ @_x(p) \in \mathcal{F}$ avec $p \in Prop$.
    Alors d'après le point 1 de la Définition 5.11, on a $\mathbb{T}\ @_x(p) \notin \mathcal{F}$ et par la Définition 5.12, on a $x \notin [\![p]\!]$ et donc $x \nVdash_{\mathcal{K}} p$.

- Si $\mathbb{T}\ @_x(i) \in \mathcal{F}$ avec $i \in Nom$.
  Alors par la Définition 5.12, on a $x \sim i$ et donc $x \vDash_{\mathcal{K}} i$.
- Si $\mathbb{F}\ @_x(i) \in \mathcal{F}$ avec $i \in Nom$.
  Alors d'après le point 1 de la Définition 5.11, on a $\mathbb{T}\ @_x(i) \notin \mathcal{F}$ et par la Définition 5.12, on a $x \not\sim i$ et donc $x \nvDash_{\mathcal{K}} i$.
- Si $\mathbb{T}\ @_x(\top) \in \mathcal{F}$.
  Alors $x \vDash_{\mathcal{K}} \top$ puisque c'est toujours le cas.
- Si $\mathbb{F}\ @_x(\top) \in \mathcal{F}$.
  D'après le point 3 de la Définition 5.11, ce cas n'existe pas.
- Si $\mathbb{T}\ @_x(\bot) \in \mathcal{F}$.
  D'après le point 2 de la Définition 5.11, ce cas n'existe pas.
- Si $\mathbb{F}\ @_x(\bot) \in \mathcal{F}$.
  Alors $x \nvDash_{\mathcal{K}} \bot$ puisque c'est toujours le cas.
- Si $\mathbb{T}\ @_x(\mathrm{I}) \in \mathcal{F}$.
  D'après la Définition 5.12, on a alors $x \sim e$ et donc $x \vDash_{\mathcal{K}} \mathrm{I}$.
- Si $\mathbb{F}\ @_x(\mathrm{I}) \in \mathcal{F}$.
  Alors d'après le point 1 de la Définition 5.11, on a $\mathbb{T}\ @_x(\mathrm{I}) \notin \mathcal{F}$ et par la Définition 5.12, on a alors $x \not\sim e$ et donc $x \nvDash_{\mathcal{K}} \mathrm{I}$.

Finalement, le lemme est vrai pour toutes les formules de taille 1.

- Hérédité On suppose que le lemme est vrai pour les formules de taille $n$ (HR) et on montre qu'il est toujours vrai pour une formule $\phi$ de taille $n+1$. On discute selon l'opérateur de $\phi$.
  - Si $\mathbb{T}\ @_x(\neg\phi') \in \mathcal{F}$.
    Alors par le point 11 de la Définition 5.11, on doit avoir $\mathbb{F}\ @_x(\phi') \in \mathcal{F}$. Alors comme $\phi'$ est de taille $n$, par (HR) on a $x \nvDash_{\mathcal{K}} \phi'$ et donc $x \vDash_{\mathcal{K}} \neg\phi'$.
  - Si $\mathbb{F}\ @_x(\neg\phi') \in \mathcal{F}$.
    Alors par le point 12 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_x(\phi') \in \mathcal{F}$. Alors comme $\phi'$ est de taille $n$, par (HR) on a $x \vDash_{\mathcal{K}} \phi'$ et donc $x \nvDash_{\mathcal{K}} \neg\phi'$.
  - Si $\mathbb{T}\ @_x(@_y(\phi')) \in \mathcal{F}$ avec $y \in Nom$.
    Alors par le point 19 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_y(\phi') \in \mathcal{F}$. Par (HR), on a alors $y \vDash_{\mathcal{K}} \phi'$ et par suite $x \vDash_{\mathcal{K}} @_y(\phi')$.
  - Si $\mathbb{F}\ @_x(@_y(\phi')) \in \mathcal{F}$ avec $y \in Nom$.
    Alors par le point 20 de la Définition 5.11, on doit avoir $\mathbb{F}\ @_y(\phi') \in \mathcal{F}$. Par (HR), on a alors $y \nvDash_{\mathcal{K}} \phi'$ et par suite $x \nvDash_{\mathcal{K}} @_y(\phi')$.
  - Si $\mathbb{T}\ @_x(\psi \wedge \psi') \in \mathcal{F}$.
    Alors par le point 5 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$ et $\mathbb{T}\ @_x(\psi') \in \mathcal{F}$. Par (HR), on a alors $x \vDash_{\mathcal{K}} \psi$ et $x \vDash_{\mathcal{K}} \psi'$, donc $x \vDash_{\mathcal{K}} \psi \wedge \psi'$.
  - Si $\mathbb{F}\ @_x(\psi \wedge \psi') \in \mathcal{F}$.
    Alors par le point 6 de la Définition 5.11, on doit avoir $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$ ou $\mathbb{F}\ @_x(\psi') \in \mathcal{F}$. Par (HR), on a alors $x \nvDash_{\mathcal{K}} \psi$ ou $x \nvDash_{\mathcal{K}} \psi'$, donc $x \nvDash_{\mathcal{K}} \psi \wedge \psi'$.

- Si $\mathbb{T}\ @_x(\psi \vee \psi') \in \mathcal{F}$.
  Alors par le point 7 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$ ou $\mathbb{T}\ @_x(\psi') \in \mathcal{F}$. Par (HR), on a alors $x \vDash_{\mathcal{K}} \psi$ ou $x \vDash_{\mathcal{K}} \psi'$, donc $x \vDash_{\mathcal{K}} \psi \vee \psi'$.
- Si $\mathbb{F}\ @_x(\psi \vee \psi') \in \mathcal{F}$.
  Alors par le point 8 de la Définition 5.11, on doit avoir $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\psi') \in \mathcal{F}$. Par (HR), on a alors $x \nvDash_{\mathcal{K}} \psi$ et $x \nvDash_{\mathcal{K}} \psi'$, donc $x \nvDash_{\mathcal{K}} \psi \vee \psi'$.
- Si $\mathbb{T}\ @_x(\psi \to \psi') \in \mathcal{F}$.
  Alors par le point 9 de la Définition 5.11, on doit avoir $\mathbb{F}\ @_x(\psi) \in \mathcal{F}$ ou $\mathbb{T}\ @_x(\psi') \in \mathcal{F}$. Par (HR), on a alors $x \nvDash_{\mathcal{K}} \psi$ ou $x \vDash_{\mathcal{K}} \psi'$, donc $x \vDash_{\mathcal{K}} \psi \to \psi'$.
- Si $\mathbb{F}\ @_x(\psi \to \psi') \in \mathcal{F}$.
  Alors par le point 10 de la Définition 5.11, on doit avoir $\mathbb{T}\ @_x(\psi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\psi') \in \mathcal{F}$. Par(HR), on a alors $x \vDash_{\mathcal{K}} \psi$ et $x \nvDash_{\mathcal{K}} \psi'$, donc $x \nvDash_{\mathcal{K}} \psi \to \psi'$.
- Si $\mathbb{T}\ @_x(\psi * \psi') \in \mathcal{F}$.
  Alors par le point 11 de la Définition 5.11, il existe $y, z \in \mathcal{A}(\mathcal{F})$ tels que $\mathbb{T}\ @_y(\psi) \in \mathcal{F}$, $\mathbb{T}\ @_z(\psi') \in \mathcal{F}$ et $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$. Par (HR) on a alors $y \vDash_{\mathcal{K}} \psi$ et $z \vDash_{\mathcal{K}} \psi'$. En outre, par la Définition 5.12, on a $y \bullet z \downarrow$ et $x \sim y \bullet z$. Donc $x \vDash_{\mathcal{K}} \psi * \psi'$.
- Si $\mathbb{F}\ @_x(\psi * \psi') \in \mathcal{F}$.
  Soient $y, z \in Nom$ tels que $x \sim y \bullet z$. Alors d'après la Définition 5.12, on doit avoir $\mathbb{T}\ @_x(y * z)$. Par le point 14 de la Définition 5.11, on a alors $\mathbb{F}\ @_y(\psi) \in \mathcal{F}$ ou $\mathbb{F}\ @_z(\psi') \in \mathcal{F}$. Par (HR), $y \nvDash_{\mathcal{K}} \psi$ ou $z \nvDash_{\mathcal{K}} \psi'$. Comme cela est vrai pour tout $y, z \in Nom$ tels que $x \sim y \bullet z$, on a $x \nvDash_{\mathcal{K}} \psi * \psi'$.
- Si $\mathbb{T}\ @_x(\psi \mathrel{-\!\!*} \psi') \in \mathcal{F}$.
  Soient $y \in Nom$ tels que $x \bullet y \downarrow$. Alors d'après la Définition 5.12, il existe $z \in Nom$ tel que $\mathbb{T}\ @_z(x * y)$ et $z \sim x \bullet y$. Par le point 15 de la Définition 5.11, on a alors $\mathbb{F}\ @_y(\psi) \in \mathcal{F}$ ou $\mathbb{T}\ @_z(\psi') \in \mathcal{F}$. Par (HR), $y \nvDash_{\mathcal{K}} \psi$ ou $z \vDash_{\mathcal{K}} \psi'$, c'est à dire par monotonie $x \bullet y \vDash_{\mathcal{K}} \psi'$. Comme cela est vrai pour tout $y \in Nom$ tels que $x \bullet y \downarrow$, on a $x \vDash_{\mathcal{K}} \psi \mathrel{-\!\!*} \psi'$.
- Si $\mathbb{F}\ @_x(\psi \mathrel{-\!\!*} \psi') \in \mathcal{F}$.
  Alors par le point 16 de la Définition 5.11, il existe $y, z \in \mathcal{A}(\mathcal{F})$ tels que $\mathbb{T}\ @_y(\psi) \in \mathcal{F}$, $\mathbb{F}\ @_z(\psi') \in \mathcal{F}$ et $\mathbb{T}\ @_z(x * y) \in \mathcal{F}$. Par (HR) on a alors $y \vDash_{\mathcal{K}} \psi$ et $z \nvDash_{\mathcal{K}} \psi'$. En outre, par la Définition 5.12, on a $x \bullet y \downarrow$ et $z \sim x \bullet y$. Par monotonie on a donc $x \bullet y \nvDash_{\mathcal{K}} \psi'$. Donc $x \nvDash_{\mathcal{K}} \psi \mathrel{-\!\!*} \psi'$.
- Si $\mathbb{T}\ @_x(\psi \mathrel{-\!\!*} \psi') \in \mathcal{F}$ ou $\mathbb{F}\ @_x(\psi \mathrel{-\!\!*} \psi') \in \mathcal{F}$, la preuve est similaire aux deux points précédents mais avec les points 17 et 18 de la Définition 5.11.

□

## B.3 Preuve du Lemme 5.12

Cette preuve est une variation de la preuve équivalente pour BBI [66].

Pour prouver qu'il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe aucun tableau clos, nous considérons cet ensemble et nous l'étendons progressivement jusqu'à obtenir toutes les propriétés d'un oracle.

**Définition B.1** (Propriété de consistance)**.** *Une* propriété de consistance *est un ensemble $\mathcal{P}$ de SHS qui satisfait les conditions suivante pour tout $\mathcal{F} \in \mathcal{P}$, tout $\phi, \psi \in \mathcal{L}$, tout $x, y, z \in Nom$ :*

1. $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}$ *ou* $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}$
2. $\mathbb{T}\ @_x(\bot) \notin \mathcal{F}$
3. $\mathbb{F}\ @_x(\top) \notin \mathcal{F}$
4. $\mathcal{F} \cup \{\mathbb{T}\ @_e(\mathrm{I})\} \in \mathcal{P}$
5. *Si* $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$
6. *Si* $\mathbb{F}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\psi)\} \in \mathcal{P}$
7. *Si* $\mathbb{T}\ @_x(\phi \vee \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$
8. *Si* $\mathbb{F}\ @_x(\phi \vee \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi), \mathbb{F}\ @_x(\psi)\} \in \mathcal{P}$
9. *Si* $\mathbb{T}\ @_x(\phi \to \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$
10. *Si* $\mathbb{F}\ @_x(\phi \to \psi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{F}\ @_x(\psi)\} \in \mathcal{P}$
11. *Si* $\mathbb{T}\ @_x(\neg\phi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\} \in \mathcal{P}$
12. *Si* $\mathbb{F}\ @_x(\neg\phi) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi)\} \in \mathcal{P}$
13. *Si* $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, *alors* $\exists c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$ *et* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\} \in \mathcal{P}$
14. *Si* $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{F}\ @_z(\psi)\} \in \mathcal{P}$
15. *Si* $\mathbb{T}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(x * y) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{T}\ @_z(\psi)\} \in \mathcal{P}$
16. *Si* $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$, *alors* $\exists c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$ *et* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{F}\ @_{c_j}(\psi)\ \mathbb{T}\ @_{c_j}(x * c_i)\} \in \mathcal{P}$
17. *Si* $\mathbb{T}\ @_x(\phi \mathrel{*\!\!-} \psi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_z(y * x) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$ *ou* $\mathcal{F} \cup \{\mathbb{T}\ @_z(\psi)\} \in \mathcal{P}$
18. *Si* $\mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi) \in \mathcal{F}$, *alors* $\exists c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$ *et* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{F}\ @_{c_j}(\psi), \mathbb{T}\ @_{c_j}(c_i * x)\} \in \mathcal{P}$
19. *Si* $\mathbb{T}\ @_x(@_y(\phi)) \in \mathcal{F}$ *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi)\} \in \mathcal{P}$
20. *Si* $\mathbb{F}\ @_x(@_y(\phi)) \in \mathcal{F}$ *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$
21. $\mathcal{F} \cup \{\mathbb{T}\ @_x(x)\} \in \mathcal{P}$
22. *Si* $\mathbb{T}\ @_x(y) \in \mathcal{F}$ *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_y(x)\} \in \mathcal{P}$
23. *Si* $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi)\} \in \mathcal{P}$.
24. *Si* $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$ *et* $\mathbb{T}\ @_x(y) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\} \in \mathcal{P}$.
25. *Si* $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\exists c_i \in Nom \backslash \mathcal{A}(\mathcal{F})$ *tel que* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y * z/c_i]), \mathbb{T}\ @_{c_i}(y * z)\} \in \mathcal{P}$ *pour toute substitution* $[y * z/c_i]$.
26. *Si* $\mathbb{F}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\exists c_i \in Nom \backslash \mathcal{A}(\mathcal{F})$ *tel que* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y * z/c_i]), \mathbb{T}\ @_{c_i}(y * z)\} \in \mathcal{P}$ *pour toute substitution* $[y * z/c_i]$.
27. *Si* $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z * t) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y/z * t])\} \in \mathcal{P}$.
28. *Si* $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z * t) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y/z * t])\} \in \mathcal{P}$.

29. *Si* $\mathbb{T}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y/z])\} \in \mathcal{P}$.
30. *Si* $\mathbb{F}\ @_x(\phi[y]) \in \mathcal{F}$ *et* $\mathbb{T}\ @_y(z) \in \mathcal{F}$, *alors* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y/z])\} \in \mathcal{P}$.

Les conditions 1 à 3 assurent que le SHS n'est pas clos. Les autres conditions assurent qu'en appliquant les règles de la Figure 5.1 l'un des nouveaux SHS appartient à $\mathcal{P}$.

**Définition B.2** (Propriété de consistance alternée)**.** *Une* propriété de consistance alternée *est un ensemble $\mathcal{P}$ de SHS qui vérifie les conditions d'une propriété de consistance, à l'exception des conditions 13, 16, 18, 25 et 26 qui sont remplacée par les suivantes :*

13'. *Si* $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, *alors* $\forall c_i, c_j \in Nom\backslash\mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$, *on a* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\} \in \mathcal{P}$

16'. *Si* $\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi) \in \mathcal{F}$, *alors* $\forall c_i, c_j \in Nom\backslash\mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$, *on a* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{F}\ @_{c_j}(\psi)\ \mathbb{T}\ @_{c_j}(x * c_i)\} \in \mathcal{P}$

18'. *Si* $\mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi) \in \mathcal{F}$, *alors* $\forall c_i, c_j \in Nom\backslash\mathcal{A}(\mathcal{F})$ *tel que* $c_i \neq c_j$, *on a* $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{F}\ @_{c_j}(\psi), \mathbb{T}\ @_{c_j}(c_i * x)\} \in \mathcal{P}$

25'. *Si* $\mathbb{T}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\forall c_i \in Nom\backslash\mathcal{A}(\mathcal{F})$ *on a* $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi[y * z/c_i]), \mathbb{T}\ @_{c_i}(y * z)\} \in \mathcal{P}$ *pour toute substitution* $[y * z/c_i]$.

26'. *Si* $\mathbb{F}\ @_x(\phi[y * z]) \in \mathcal{F}$ *alors* $\forall c_i \in Nom\backslash\mathcal{A}(\mathcal{F})$, *on a* $\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi[y * z/c_i]), \mathbb{T}\ @_{c_i}(y * z)\} \in \mathcal{P}$ *pour toute substitution* $[y * z/c_i]$.

**Proposition B.1.** *L'ensemble de tous les SHS finis pour lesquels il n'existe pas de tableau clos est une propriété de consistance.*

*Démonstration.*
Soit $\mathcal{P}$ l'ensemble de tous les SHS finis pour lesquels il n'existe pas de tableau clos. On montre que $\mathcal{P}$ est une propriété de consistance. Soit $\mathcal{F} \in \mathcal{P}$, soit $\phi, \psi \in \mathcal{L}$, soient $x, y, z \in Nom$.

1.-3. Comme $\mathcal{F}$ ne peut pas être clos, ces trois propositions sont vraies.

5. Si $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, on suppose que $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \notin \mathcal{P}$. Alors, par extension de $\mathcal{F}$ par la règle $\langle\mathbb{T}\wedge\rangle$, $[\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}]$ est un tableau pour $\mathcal{F}$. Comme par hypothèse, $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \notin \mathcal{P}$, on peut obtenir un tableau clos à partir de $[\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}]$, donc on peut obtenir un tableau clos à partir de $\mathcal{F}$, donc $\mathcal{F} \notin \mathcal{P}$. C'est absurde, donc $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$.

13. Si $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$. On choisit $c_i, c_j \in Nom\backslash\mathcal{A}(\mathcal{F})$. On suppose que $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \notin \mathcal{P}$. D'après la règle $\langle\mathbb{T}*\rangle$, comme $c_i$ et $c_j$ sont nouveaux par rapport à $\mathcal{F}$, $[\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\}]$ est un tableau pour $\mathcal{F}$. Comme on a supposé $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \notin \mathcal{P}$, il existe un tableau clos pour $\mathcal{F}$, $[\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\}]$, donc il existe un tableau clos pour $\mathcal{F}$ et $\mathcal{F} \notin \mathcal{P}$. C'est absurde, donc $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \in \mathcal{P}$.

Les autres cas sont similaires. □

**Proposition B.2.** *Toute propriété de consistance peut être étendue en une propriété de consistance $\subseteq$-close.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance. On définit la $\subseteq$-clôture de $\mathcal{P}$, notée $\mathcal{P}^{\subseteq}$, par :

$$\mathcal{F} \in \mathcal{P}^{\subseteq} \text{ ssi } \mathcal{F} \subseteq \mathcal{F}' \text{ pour un certain } \mathcal{F}' \in \mathcal{P}$$

On a $\mathcal{P} \subseteq \mathcal{P}^{\subseteq}$ car $\subseteq$ est réflexive. $\mathcal{P}^{\subseteq}$ est $\subseteq$-clos car $\subseteq$ est transitive. On montre que $\mathcal{P}^{\subseteq}$ est une propriété de consistance.

Soit $\mathcal{F} \in \mathcal{P}^{\subseteq}$. Alors il existe $\mathcal{F}' \in \mathcal{P}$ tel que $\mathcal{F} \subseteq \mathcal{F}'$. Montrons que les points de la Définition B.1 sont vérifiés :

1. Supposons que $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. Alors, comme $\mathcal{F} \subseteq \mathcal{F}'$, $\mathbb{T}\ @_x(\phi) \in \mathcal{F}'$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}'$. Mais c'est impossible car $\mathcal{P}$ vérifie la condition 1 de la définition. Donc $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}$ ou $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}$.
5. Si $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, alors comme $\mathcal{F} \subseteq \mathcal{F}'$, on a $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}'$. Alors comme $\mathcal{P}$ vérifie la condition 5 de la définition, on a $\mathcal{F}' \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$. Comme $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \subseteq \mathcal{F}' \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$, on a alors $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}^{\subseteq}$.
13. Si $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, alors comme $\mathcal{F} \subseteq \mathcal{F}'$ on a $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}'$. Alors comme $\mathcal{P}$ vérifie la condition 13 de la définition, on a $\exists c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})'$ tel que $c_i \neq c_j$ et $\mathcal{F}' \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \in \mathcal{P}$. Comme $\mathcal{F} \subseteq \mathcal{F}'$, on a $\mathcal{A}(\mathcal{F}) \subseteq \mathcal{A}(\mathcal{F})'$. Donc $Nom \backslash \mathcal{A}(\mathcal{F})' \subseteq Nom \backslash \mathcal{A}(\mathcal{F})$, et ainsi $c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$. De plus, comme $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \subseteq \mathcal{F}' \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\}$ on a $\mathcal{F} \cup \{\mathbb{T}\ @_y(\phi), \mathbb{T}\ @_z(\psi), \mathbb{T}\ @_x(y * z)\} \in \mathcal{P}^{\subseteq}$.

Les autres cas sont similaires. □

**Définition B.3** (Substitution)**.**

1. *On appelle* substitution *toute fonction* $\sigma : Nom \to Nom$ *telle que* $\sigma(e) = e$.
2. *On étend la notion de substitution aux formules qui ne sont pas des nominaux par la définition récursive suivante :*
   - *Pour tout* $p \in Prop$, $\sigma(p) = p$
   - $\sigma(\top) = \top$
   - $\sigma(\bot) = \bot$
   - $\sigma(\mathrm{I}) = \mathrm{I}$
   - $\sigma(\neg\phi) = \neg\sigma(\phi)$
   - $\sigma(@_i(\phi)) = @_{\sigma(i)}(\sigma(\phi))$
   - $\sigma(\phi \wedge \psi) = \sigma(\phi) \wedge \sigma(\psi)$
   - $\sigma(\phi \vee \psi) = \sigma(\phi) \vee \sigma(\psi)$
   - $\sigma(\phi \to \psi) = \sigma(\phi) \to \sigma(\psi)$
   - $\sigma(\phi * \psi) = \sigma(\phi) * \sigma(\psi)$
   - $\sigma(\phi \mathrel{-\!\!*} \psi) = \sigma(\phi) \mathrel{-\!\!*} \sigma(\psi)$
   - $\sigma(\phi \mathrel{*\!\!-} \psi) = \sigma(\phi) \mathrel{*\!\!-} \sigma(\psi)$

3. *On étend la notion de substitution aux formules étiquetées par :*

$$\sigma(\mathbb{S}\ @_x(\phi)) = \mathbb{S}\ @_{\sigma(x)}(\sigma(\phi))$$

4. *On étend la notion de substitution à toute branche $\mathcal{F}$ par :*

$$\sigma(\mathcal{F}) = \{\sigma(\mathbb{S}\ @_x(\phi)) | \mathbb{S}\ @_x(\phi) \in \mathcal{F}\}$$

**Proposition B.3.** *Pour toute substitution $\sigma$, on a :*

$$\sigma(\mathcal{F}) \cup \{\sigma(\mathbb{S}\ @_x(\phi))\} = \sigma(\mathcal{F} \cup \{\mathbb{S}\ @_x(\phi)\})$$

La preuve est directe.

**Proposition B.4.** *Pour toute substitution $\sigma$, on a :*

1. *Si $\mathcal{F}_f$ est un SHS fini, alors $\sigma(\mathcal{F})$ est un SHS fini.*
2. *Si $\mathcal{F} \subseteq \mathcal{F}'$, alors $\sigma(\mathcal{F}) \subseteq \sigma(\mathcal{F}')$.*

*Démonstration.*

1. La preuve est directe étant donné que $|\sigma(\mathcal{F})| = |\mathcal{F}|$.
2. Soit $\mathcal{F}$ et $\mathcal{F}'$ deux SHS tels que $\mathcal{F} \subseteq \mathcal{F}'$. Soit $F \in \sigma(\mathcal{F})$. Par définition de $\sigma$, $F$ est de la forme $\sigma(\mathbb{S}\ @_x(\phi))$ avec $\mathbb{S}\ @_x(\phi) \in \mathcal{F}$. Comme $\mathcal{F} \subseteq \mathcal{F}'$, on a $\mathbb{S}\ @_x(\phi) \in \mathcal{F}'$, donc $\sigma(\mathbb{S}\ @_x(\phi)) \in \sigma(\mathcal{F}')$, c'est à dire que $F \in \sigma(\mathcal{F}')$.

□

**Proposition B.5.** *Toute propriété de consistance $\subseteq$-close peut être étendue en une propriété de consistance $\subseteq$-close alternée.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance $\subseteq$-close. Soit $\mathcal{P}^+$ l'ensemble défini par :

$$\mathcal{F} \in \mathcal{P}^+ \text{ ssi } \sigma(\mathcal{F}) \in \mathcal{P} \text{ pour une substitution } \sigma$$

On remarque tout d'abord que $\mathcal{P} \subseteq \mathcal{P}^+$ (en considérant la substitution identité). On montre que $\mathcal{P}^+$ est $\subseteq$-clos. Soit $\mathcal{F} \in \mathcal{P}^+$ et $\mathcal{F}'$ tel que $\mathcal{F}' \subseteq \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^+$, il existe une substitution $\sigma$ telle que $\sigma(\mathcal{F}) \in \mathcal{P}$. Par la Proposition B.4, on a $\sigma(\mathcal{F}') \subseteq \sigma(\mathcal{F})$. Comme $\mathcal{P}$ est $\subseteq$-clos, on a $\sigma(\mathcal{F}') \in \mathcal{P}$. Par conséquent, $\mathcal{F}' \in \mathcal{P}^+$, et donc $\mathcal{P}^+$ est $\subseteq$-clos. On montre que $\mathcal{P}^+$ est un propriété de consistance alternée. Soit $\mathcal{F} \in \mathcal{P}^+$. Il existe une substitution $\sigma$ telle que $\sigma(\mathcal{F}) \in \mathcal{P}$.

1. Supposons que $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. Alors $\sigma(\mathbb{T}\ @_x(\phi)) \in \sigma(\mathcal{F})$ et $\sigma(\mathbb{F}\ @_x(\phi)) \in \sigma(\mathcal{F})$, c'est à dire que $\mathbb{T}\ @_{\sigma(x)}(\sigma(\phi)) \in \sigma(\mathcal{F})$ et $\mathbb{F}\ @_{\sigma(x)}(\sigma(\phi)) \in \sigma(\mathcal{F})$. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$, ceci est absurde, car $\mathcal{P}$ est une propriété de consistance. Donc $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}$ ou $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}$.

5. Si $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, alors $\sigma(\mathbb{T}\ @_x(\phi \wedge \psi)) \in \sigma(\mathcal{F})$, c'est à dire $\mathbb{T}\ @_{\sigma(x)}(\sigma(\phi) \wedge \sigma(\psi)) \in \sigma(\mathcal{F})$. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$ et que $\mathcal{P}$ est une propriété de consistance, par le point 5 de la Définition B.1, on a $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma(x)}(\sigma(\phi)), \mathbb{T}\ @_{\sigma(x)}(\sigma(\psi))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \{\sigma(\mathbb{T}\ @_x(\phi)), \sigma(\mathbb{T}\ @_x(\psi))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \sigma(\{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}) \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}) \in \mathcal{P}$ (d'après la Proposition B.2). Donc on a $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}^+$.

13'. Si $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, alors $\sigma(\mathbb{T}\ @_x(\phi * \psi)) \in \sigma(\mathcal{F})$, c'est à dire que $\mathbb{T}\ @_{\sigma(x)}(\sigma(\phi) * \sigma(\psi)) \in \sigma(\mathcal{F})$. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$ et que $\mathcal{P}$ est une propriété de consistance, par le point 13 de la Définition B.1, $\exists c_i, c_j \in Nom \backslash \mathcal{A}(\sigma(\mathcal{F}))$ tel que $c_i \neq c_j$ et $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{c_i}(\sigma(\phi)), \mathbb{T}\ @_{c_j}(\sigma(\psi)), \mathbb{T}\ @_{\sigma(x)}(c_i * c_j)\} \in \mathcal{P}$. Soit $c'_i, c'_j \in Nom \backslash \mathcal{A}(\mathcal{F})$. Soit $\sigma'$ la substitution définie par $\sigma' = \sigma[c'_i \mapsto c_i, c'_j \mapsto c_j]$. Comme $c'_i \notin \mathcal{A}(\mathcal{F})$ et $c'_j \notin \mathcal{A}(\mathcal{F})$, on a $\sigma(\mathcal{F}) = \sigma'(\mathcal{F})$, $\sigma(\phi) = \sigma'(\phi)$, $\sigma(\psi) = \sigma'(\psi)$ et $\sigma(x) = \sigma'(x)$. En outre, par définition $c_i = \sigma'(c'_i)$ et $c_j = \sigma'(c'_j)$. On a donc $\sigma'(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma'(c'_i)}(\sigma'(\phi)), \mathbb{T}\ @_{\sigma'(c'_j)}(\sigma'(\psi)), \mathbb{T}\ @_{\sigma'(x)}(\sigma'(c'_i) * \sigma'(c'_j))\} \in \mathcal{P}$, c'est à dire que $\sigma'(\mathcal{F} \cup \{\mathbb{T}\ @_{c'_i}(\phi), \mathbb{T}\ @_{c'_j}(\psi), \mathbb{T}\ @_x(c'_i * c'_j)\}) \in \mathcal{P}$, donc $\mathcal{F} \cup \{\mathbb{T}\ @_{c'_i}(\phi), \mathbb{T}\ @_{c'_j}(\psi), \mathbb{T}\ @_x(c'_i * c'_j)\} \in \mathcal{P}^+$. Comme cela est vrai pour tout $c'_i, c'_j \in Nom \backslash \mathcal{A}(\mathcal{F})$, $\mathcal{F}$ vérifie bien la condition 12' de la Définition B.2.

Les autres cas sont similaires. □

**Proposition B.6.** *Toute propriété de consistance $\subseteq$-close alternée peut être étendue en une propriété de consistance $\subseteq$-close alternée de caractère fini.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance $\subseteq$-close alternée. On définit $\mathcal{P}^{fc}$ par :

$$\mathcal{F} \in \mathcal{P}^{fc} \text{ ssi } \mathcal{F}_f \in \mathcal{P} \text{ pour tout } \mathcal{F}_f \subseteq_f \mathcal{F}$$

On montre que $\mathcal{P} \subseteq \mathcal{P}^{fc}$.

Soit $\mathcal{F} \in \mathcal{P}$. Soit $\mathcal{F}_f$ tel que $\mathcal{F}_f \subseteq_f \mathcal{F}$. Alors, comme $\mathcal{P}$ est $\subseteq$-close, on a $\mathcal{F}_f \in \mathcal{P}$, donc $\mathcal{F} \in \mathcal{P}^{fc}$.

On montre que $\mathcal{P}^{fc}$ est $\subseteq$-close.

Soit $\mathcal{F} \in \mathcal{P}^{fc}$. Soit $\mathcal{F}'$ tel que $\mathcal{F}' \subseteq \mathcal{F}$. Soit $\mathcal{F}'_f$ tel que $\mathcal{F}'_f \subseteq \mathcal{F}'$. Alors $\mathcal{F}'_f \subseteq_f \mathcal{F}$ par transitivité. Donc par définition de $\mathcal{P}^{fc}$, on a $\mathcal{F}'_f \in \mathcal{P}$, donc par définition $\mathcal{F}' \in \mathcal{P}^{fc}$.

On montre que $\mathcal{P}^{fc}$ est de caractère fini.

Soit $\mathcal{F}$ un SHS. On suppose que pour tout $\mathcal{F}_f \subseteq_f \mathcal{F}$, $\mathcal{F}_f \in \mathcal{P}^{fc}$. On montre que $\mathcal{F} \in \mathcal{P}^{fc}$. Par définition, pour tout $\mathcal{F}'_f \subseteq_f \mathcal{F}_f$, on a $\mathcal{F}'_f \in \mathcal{P}$. En particulier, $\mathcal{F}_f \subseteq_f \mathcal{F}_f$, donc $\mathcal{F}_f \in \mathcal{P}$. Donc par définition, $\mathcal{F} \in \mathcal{P}^{fc}$.

On montre que $\mathcal{P}^{fc}$ est une propriété de consistance alternée. Soit $\mathcal{F} \in \mathcal{P}^{fc}$.

1. Supposons que $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. On considère l'ensemble $\{\mathbb{T}\ @_x(\phi), \mathbb{F}\ @_x(\phi)\}$. Cet ensemble est fini, et on a évidemment $\{\mathbb{T}\ @_x(\phi), \mathbb{F}\ @_x(\phi)\} \subseteq_f \mathcal{F}$. Donc par définition de $\mathcal{P}^{fc}$, on a $\{\mathbb{T}\ @_x(\phi), \mathbb{F}\ @_x(\phi)\} \in \mathcal{P}$. Ceci est absurde car $\mathcal{P}$ est une propriété de consistance alternée. Donc $\mathbb{T}\ @_x(\phi) \notin \mathcal{F}$ ou $\mathbb{F}\ @_x(\phi) \notin \mathcal{F}$.

5. Si $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, alors on considère l'ensemble $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$. Soit $\mathcal{F}_f$ un ensemble fini tel que $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$. Soit $\mathcal{F}'_f$ l'ensemble défini par $\mathcal{F}'_f = (\mathcal{F}_f \backslash \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}) \cup \{\mathbb{T}\ @_x(\phi \wedge \psi)\}$. Par construction, $\mathcal{F}'_f$ est fini.

De plus, comme $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}$, on a $\mathcal{F}'_f \subseteq_f \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^{fc}$, on a alors que $\mathcal{F}'_f \in \mathcal{P}$. Comme $\mathbb{T}\ @_x(\phi \wedge \psi) \in \mathcal{F}'_f$ et que $\mathcal{P}$ est une propriété de consistance alternée, on a $\mathcal{F}'_f \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}$. Or $\mathcal{F}_f \subseteq \mathcal{F}'_f \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$. Donc, comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_f \in \mathcal{P}$. Comme cela est vrai pour tout $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}$, on en déduit que $\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\} \in \mathcal{P}^{fc}$.

13'. On suppose $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$. Soit $c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$. On considère l'ensemble $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}$. Soit $\mathcal{F}_f$ un ensemble fini tel que $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}$. Soit $\mathcal{F}'_f$ l'ensemble défini par $\mathcal{F}'_f = (\mathcal{F}_f \backslash \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}) \cup \{\mathbb{T}\ @_x(\phi * \psi)\}$. Par construction, $\mathcal{F}'_f$ est fini. De plus, comme $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$, on a $\mathcal{F}'_f \subseteq_f \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^{fc}$, on a alors que $\mathcal{F}'_f \in \mathcal{P}$. Comme $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}'_f$ et que $\mathcal{P}$ est une propriété de consistance alternée, on a $\mathcal{F}'_f \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\} \in \mathcal{P}$. Or $\mathcal{F}_f \subseteq \mathcal{F}'_f \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}$. Donc, comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_f \in \mathcal{P}$. Comme cela est vrai pour tout $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}$, on en déduit que $\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\} \in \mathcal{P}^{fc}$. Comme cela est vrai pour tout $c_i, c_j \in Nom \backslash \mathcal{A}(\mathcal{F})$, $\mathcal{F}$ vérifie bien la condition 13' de la Définition B.2.

Les autres cas sont similaires. □

**Lemme 5.12.** *Il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe aucun tableau clos.*

*Démonstration.*
On considère l'ensemble $\mathcal{P}$ de tous les SHS finis pour lesquels il n'existe aucun tableau clos. Par la Proposition B.1, $\mathcal{P}$ est une propriété de consistance. Par la Proposition B.2, on peut l'étendre en une propriété de consistance $\subseteq$-close. Par la Proposition B.5, on peut l'étendre en une propriété de consistance alternée $\subseteq$-close. Par la Proposition B.6, on peut l'étendre en une propriété de consistance alternée $\subseteq$-close de caractère fini. Par les conditions 4 à 30 de la Définition B.2, cette extension est saturée. Par les conditions 1 à 3 de la Définition B.2, cette extension ne contient que des SHS non clos. Cet ensemble est donc un oracle. Comme cet oracle est une extension de $\mathcal{P}$, il existe bien un oracle que contient $\mathcal{P}$. □

## B.4 Preuve de la Proposition 5.2

**Proposition 5.2.** *Pour tout $i \in \mathbb{N}$, les propriétés suivantes sont vérifiées :*

1. $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$
2. $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_i$
3. $(\mathcal{F}_i)_{i \geq 0} \in \mathcal{P}$
4. $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$

*Démonstration.*

1. $\mathcal{F}_{i+1}$ est ou bien $\mathcal{F}_i$ ou bien une extension de $\mathcal{F}_i$ par $\cup$, d'où ce résultat.

2. On a $\mathbb{F}\ @_{c_1}(\phi) \in \mathcal{F}_0$ par construction. Comme $\mathcal{F}_i \subseteq \mathcal{F}_{i+1}$ pour tout $i$, une simple récurrence donne le résultat.

3.,4. On démontre les deux résultats par récurrence.

- Cas de base
  On a déjà noté que $\{\mathbb{F}\ @_{c_1}(\phi)\} \in \mathcal{P}$, d'où le point 3. Le point 4. est évident puisque $\mathcal{A}(\{\mathbb{F}\ @_{c_1}(\phi)\})\backslash Nom_e = \{c_1\}$.
- Hérédité
  On considère les deux propriétés vraies pour un certain $i$ (HR), on démontre qu'elles restent vraies pour $i+1$.
  - Si $\mathcal{F}_i \cup \{F_i\} \notin \mathcal{P}$ alors $\mathcal{F}_{i+1} = \mathcal{F}_i$ et les deux propriétés sont vraies par (HR).
  - Dans le cas contraire, on a 6 cas :
    - Si $n_i = 0$ et $F_i = \mathbb{T}\ @_x(\phi * \psi)$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}, c_{2i+3}\}$. Or, par (HR), on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas. En outre, on a $F_i \cup \{\mathbb{T}\ @_x(\phi * \psi)\} \in \mathcal{P}$, donc par saturation par la règle $\langle \mathbb{T}* \rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$ et $\mathfrak{b} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{T}\ @_x(\phi * \psi)\} \cup \{\mathbb{T}\ @_\mathfrak{a}(\phi), \mathbb{T}\ @_\mathfrak{b}(\psi), \mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b})\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.
    - Si $n_i = 0$ et $F_i = \mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi)$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}, c_{2i+3}\}$. Or, par (HR), on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas. En outre, on a $F_i \cup \{\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi)\} \in \mathcal{P}$, donc par saturation par la règle $\langle \mathbb{F}\mathrel{-\!\!*} \rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$ et $\mathfrak{b} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{F}\ @_x(\phi \mathrel{-\!\!*} \psi)\} \cup \{\mathbb{T}\ @_\mathfrak{a}(\phi), \mathbb{F}\ @_\mathfrak{b}(\psi), \mathbb{T}\ @_\mathfrak{b}(x * \mathfrak{a})\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.
    - Si $n_i = 0$ et $F_i = \mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi)$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}, c_{2i+3}\}$. Or, par (HR), on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas. En outre, on a $F_i \cup \{\mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi)\} \in \mathcal{P}$, donc par saturation par la règle $\langle \mathbb{F}\mathrel{*\!\!-} \rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$ et $\mathfrak{b} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{F}\ @_x(\phi \mathrel{*\!\!-} \psi)\} \cup \{\mathbb{T}\ @_\mathfrak{a}(\phi), \mathbb{F}\ @_\mathfrak{b}(\psi), \mathbb{T}\ @_\mathfrak{b}(\mathfrak{a} * x)\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.
    - Si $n_i \neq 0$ et $F_i = \mathbb{T}\ @_x(\phi * \psi)$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}, c_{2i+3}\}$. Or, par (HR), on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas. En outre, on a $F_i \cup \{\mathbb{T}\ @_x(\phi * \psi)\} \in \mathcal{P}$, donc par saturation par la règle $\langle \mathbb{T}* \rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$ et $\mathfrak{b} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{T}\ @_x(\phi * \psi)\} \cup \{\mathbb{T}\ @_\mathfrak{a}(\phi), \mathbb{T}\ @_\mathfrak{b}(\psi), \mathbb{T}\ @_x(\mathfrak{a} * \mathfrak{b})\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.
    - Si $n_i = 0$ et $F_i = \mathbb{T}\ @_x(\phi[y * z])$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}\}$. Or, par HR, on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas.

En outre, on a $F_i \cup \{\mathbb{T}\ @_x(\phi[y*z])\} \in \mathcal{P}$, donc par saturation par la règle $\langle i+\rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{T}\ @_x(\phi[y*z])\} \cup \{\mathbb{T}\ @_x(\phi[y*z/\mathfrak{a}]), \mathbb{T}\ @_\mathfrak{a}(y*z)\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.

- Si $n_i = 0$ et $F_i = \mathbb{F}\ @_x(\phi[y*z])$, alors $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i) \cup \{c_{2i+2}\}$. Or, par (HR), on a $\mathcal{A}(\mathcal{F}_i) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+1}\}$ et donc $\mathcal{A}(\mathcal{F}_{i+1}) \subseteq Nom_e \cup \{c_1, \ldots, c_{2i+3}\}$, ce qui démontre le point 4. dans ce cas.
  En outre, on a $F_i \cup \{\mathbb{F}\ @_x(\phi[y*z])\} \in \mathcal{P}$, donc par saturation par la règle $\langle i+\rangle$, et comme par (HR) on a $\mathfrak{a} \notin \mathcal{A}(\mathcal{F}_i)$, on doit avoir $F_i \cup \{\mathbb{F}\ @_x(\phi[y*z])\} \cup \{\mathbb{F}\ @_x(\phi[y*z/\mathfrak{a}]), \mathbb{T}\ @_\mathfrak{a}(y*z)\} \in \mathcal{P}$, c'est à dire que $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.
- Dans les cas où $\mathcal{F}_e = \emptyset$, $\mathcal{A}(\mathcal{F}_{i+1}) = \mathcal{A}(\mathcal{F}_i)$ et le point 4. est évident. En outre, par hypothèse, $\mathcal{F}_i \cup \{F_i\} \in \mathcal{P}$ par hypothèse, c'est à dire $\mathcal{F}_{i+1} \in \mathcal{P}$, d'où la preuve du point 3.

□

## B.5 Preuve du Lemme 5.15

On donne ici la preuve du Lemme 5.15. Il s'agit d'étendre chacune des définitions et propositions de la Section B.3.

On étend d'abord la notion de propriété de consistance.

**Définition B.4** (Propriété de consistance pour HBBI - extension de la Définition B.1)**.** *Une* propriété de consistance *pour HBBI est un ensemble $\mathcal{P}$ de SHS qui satisfait toutes les conditions de la Définition B.1 ainsi que les points additionnels suivants :*

*31.* $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \in \mathcal{P}$.

*32. Si* $\mathbb{T}\ @_x(y*z) \in \mathcal{F}$*, alors* $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \in \mathcal{P}$.

*33. Si* $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}$*, alors* $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \in \mathcal{P}$.

La notion de propriété de consistance alternée est étendue par les mêmes conditions.

**Définition B.5** (Propriété de consistance alternée pour HBBI - extension de la Définition B.2)**.** *Une* propriété de consistance alternée *pour HBBI est un ensemble $\mathcal{P}$ de SHS qui vérifie les conditions d'une propriété de consistance alternée (Définition B.2) ainsi que les trois conditions additionnelles de la Définition B.4.*

On a ensuite les Propositions suivantes, identiques à leurs versions de la Section B.3 :

**Proposition B.7.** *L'ensemble de tous les SHS finis pour lesquels il n'existe pas de tableau clos est une propriété de consistance.*

*Démonstration.*
Soit $\mathcal{P}$ l'ensemble de tous les SHS finis pour lesquels il n'existe pas de tableau clos. On montre que $\mathcal{P}$ est une propriété de consistance. Les conditions 1. à 30. sont prouvées comme dans la preuve de la Proposition B.1. Les points 31. à 33. sont prouvés ci-dessous :

31. On suppose que $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \notin \mathcal{P}$. Alors, par extension de $\mathcal{F}$ par la règle $\langle BI_n \rangle$, $[\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I})]$ est un tableau pour $\mathcal{F}$. Comme par hypothèse $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \notin \mathcal{P}$, on peut obtenir un tableau clos à partir de $[\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I})]$, donc on peut obtenir un tableau clos à partir de $\mathcal{F}$ et donc $\mathcal{F} \notin \mathcal{P}$. C'est absurde, donc $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \in \mathcal{P}$.
32. Si $\mathbb{T}\ @_x(y*z) \in \mathcal{F}$, on suppose que $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \notin \mathcal{P}$. Alors, par extension de $\mathcal{F}$ par la règle $\langle BI_c \rangle$, $[\mathcal{F} \cup \mathbb{T}\ @_x(z*y)]$ est un tableau pour $\mathcal{F}$. Comme par hypothèse $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \notin \mathcal{P}$, on peut obtenir un tableau clos à partir de $[\mathcal{F} \cup \mathbb{T}\ @_x(z*y)]$, donc on peut obtenir un tableau clos à partir de $\mathcal{F}$ et donc $\mathcal{F} \notin \mathcal{P}$. C'est absurde, donc $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \in \mathcal{P}$.
33. Si $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}$, on suppose que $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \notin \mathcal{P}$. Alors, par extension de $\mathcal{F}$ par la règle $\langle BI_a \rangle$, $[\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t)]$ est un tableau pour $\mathcal{F}$. Comme par hypothèse $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \notin \mathcal{P}$, on peut obtenir un tableau clos à partir de $[\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t)]$, donc on peut obtenir un tableau clos à partir de $\mathcal{F}$ et donc $\mathcal{F} \notin \mathcal{P}$. C'est absurde, donc $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \in \mathcal{P}$.

□

**Proposition B.8.** *Toute propriété de consistance peut être étendue en une propriété de consistance $\subseteq$-close.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance. On définit la $\subseteq$-clôture de $\mathcal{P}$, notée $\mathcal{P}^{\subseteq}$ comme dans la preuve de la Proposition B.2. On démontre comme dans cette preuve que $\mathcal{P}^{\subseteq}$ est $\subseteq$-close. On montre que $\mathcal{P}^{\subseteq}$ est une propriété de consistance. Les points 1. à 30. sont identiques à la preuve de la Proposition B.2. Les points 31. à 33. sont démontrés ci-dessous :

31. Comme $\mathcal{P}$ vérifie la condition 31 de la définition, on a $\mathcal{F}' \cup \mathbb{T}\ @_x(x*\mathrm{I}) \in \mathcal{P}$. Comme $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \subseteq \mathcal{F}' \cup \mathbb{T}\ @_x(x*\mathrm{I})$, on a alors $\mathcal{F} \cup \mathbb{T}\ @_x(x*\mathrm{I}) \in \mathcal{P}^{\subseteq}$.
32. Si $\mathbb{T}\ @_x(y*z) \in \mathcal{F}$, alors comme $\mathcal{F} \subseteq \mathcal{F}'$ on a $\mathbb{T}\ @_x(y*z) \in \mathcal{F}'$. Donc comme $\mathcal{P}$ vérifie la condition 32 de la définition, on a $\mathcal{F}' \cup \mathbb{T}\ @_x(z*y) \in \mathcal{P}$. Comme $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \subseteq \mathcal{F}' \cup \mathbb{T}\ @_x(z*y)$, on a alors $\mathcal{F} \cup \mathbb{T}\ @_x(z*y) \in \mathcal{P}^{\subseteq}$.
33. Si $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}$, alors comme $\mathcal{F} \subseteq \mathcal{F}'$ on a $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}'$. Donc comme $\mathcal{P}$ vérifie la condition 33 de la définition, on a $\mathcal{F}' \cup \mathbb{T}\ @_x((y*z)*t) \in \mathcal{P}$. Comme $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \subseteq \mathcal{F}' \cup \mathbb{T}\ @_x((y*z)*t)$, on a alors $\mathcal{F} \cup \mathbb{T}\ @_x((y*z)*t) \in \mathcal{P}^{\subseteq}$.

□

On conserve ensuite de manière inchangée la définition d'une substitution (Définition B.3). Par conséquent, les Propositions B.3 et B.4 sont toujours valables, puisqu'elle concernent les substitutions.

On peut alors montrer les proposition suivantes :

**Proposition B.9.** *Toute propriété de consistance $\subseteq$-close peut être étendue en une propriété de consistance $\subseteq$-close alternée.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance $\subseteq$-close. On définit l'ensemble $\mathcal{P}^{+}$ comme dans la preuve de la Proposition B.5 et, comme dans cette preuve, on montre qu'il est $\subseteq$-clos et qu'il contient $\mathcal{P}$. On montre ensuite que $\mathcal{P}^{+}$ est une propriété de consistance alternée. Les points 1. à 30.

se montre comme dans la preuve de la Proposition B.5. Les points 31. à 33. sont montrés ci-dessous :

31. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$ et que $\mathcal{P}$ est une propriété de consistance, par le point 31 de la Définition B.1, on a $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma(x)}(\sigma(x) * \mathrm{I})\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma(x)}(\sigma(x) * \sigma(\mathrm{I}))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \{\sigma(\mathbb{T}\ @_x(x * \mathrm{I}))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}) \in \mathcal{P}$ (d'après la Proposition B.2). Donc on a $\mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\} \in \mathcal{P}^+$.
32. Si $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$, alors $\sigma(\mathbb{T}\ @_x(y * z)) \in \sigma(\mathcal{F})$, c'est à dire $\mathbb{T}\ @_{\sigma(x)}(\sigma(y) * \sigma(z)) \in \sigma(\mathcal{F})$. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$ et que $\mathcal{P}$ est une propriété de consistance, par le point 32 de la Définition B.1, on a $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma(x)}(\sigma(z) * \sigma(y))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \{\sigma(\mathbb{T}\ @_x(z * y))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\}) \in \mathcal{P}$ (d'après la Proposition B.2). Donc on a $\mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\} \in \mathcal{P}^+$.
33. Si $\mathbb{T}\ @_x(y * (z * t)) \in \mathcal{F}$, alors $\sigma(\mathbb{T}\ @_x(y * (z * t))) \in \sigma(\mathcal{F})$, c'est à dire $\mathbb{T}\ @_{\sigma(x)}(\sigma(y) * (\sigma(z) * \sigma(t))) \in \sigma(\mathcal{F})$. Comme $\sigma(\mathcal{F}) \in \mathcal{P}$ et que $\mathcal{P}$ est une propriété de consistance, par le point 33 de la Définition B.1, on a $\sigma(\mathcal{F}) \cup \{\mathbb{T}\ @_{\sigma(x)}((\sigma(y) * \sigma(z)) * \sigma(t))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F}) \cup \{\sigma(\mathbb{T}\ @_x((y * z) * t))\} \in \mathcal{P}$, c'est à dire que $\sigma(\mathcal{F} \cup \{\mathbb{T}\ @_x((y * z) * t)\}) \in \mathcal{P}$ (d'après la Proposition B.2). Donc on a $\mathcal{F} \cup \{\mathbb{T}\ @_x((y * z) * t)\} \in \mathcal{P}^+$.

□

**Proposition B.10.** *Toute propriété de consistance $\subseteq$-close alternée peut être étendue en une propriété de consistance $\subseteq$-close alternée de caractère fini.*

*Démonstration.*
Soit $\mathcal{P}$ une propriété de consistance $\subseteq$-close alternée. On définit $\mathcal{P}^{fc}$ comme dans la preuve de la Proposition B.6 et, comme dans cette preuve, on montre qu'il est $\subseteq$-clos, de caractère fini et qu'il contient $\mathcal{P}$. On montre ensuite que $\mathcal{P}^{fc}$ est une propriété de consistance alternée. Les points 1. à 30. se montre comme dans la preuve de la Proposition B.6. Les points 31. à 33. sont montrés ci-dessous :

31. On considère l'ensemble $\mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}$. Soit $\mathcal{F}_f$ un ensemble fini tel que $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}$. Soit $\mathcal{F}'_f$ l'ensemble défini par $\mathcal{F}'_f = (\mathcal{F}_f \backslash \{\mathbb{T}\ @_x(x * \mathrm{I})\})$. Par construction, $\mathcal{F}'_f$ est fini. De plus on a $\mathcal{F}'_f \subseteq_f \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^{fc}$, on a alors que $\mathcal{F}'_f \in \mathcal{P}$. Comme $\mathcal{P}$ est une propriété de consistance alternée, on a $\mathcal{F}'_f \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\} \in \mathcal{P}$. Or $\mathcal{F}_f = \mathcal{F}'_f \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}$, donc $\mathcal{F}_f \in \mathcal{P}$. Comme cela est vrai pour tout $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\}$, on en déduit que $\mathcal{F} \cup \{\mathbb{T}\ @_x(x * \mathrm{I})\} \in \mathcal{P}^{fc}$.
32. Si $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$, alors on considère l'ensemble $\mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\}$. Soit $\mathcal{F}_f$ un ensemble fini tel que $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\}$. Soit $\mathcal{F}'_f$ l'ensemble défini par $\mathcal{F}'_f = (\mathcal{F}_f \backslash \{\mathbb{T}\ @_x(z * y)\}) \cup \{\mathbb{T}\ @_x(y * z)\}$. Par construction, $\mathcal{F}'_f$ est fini. De plus, comme $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$, on a $\mathcal{F}'_f \subseteq_f \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^{fc}$, on a alors que $\mathcal{F}'_f \in \mathcal{P}$. Comme $\mathbb{T}\ @_x(y * z) \in \mathcal{F}'_f$ et que $\mathcal{P}$ est une propriété de consistance alternée, on a $\mathcal{F}'_f \cup \{\mathbb{T}\ @_x(z * y)\} \in \mathcal{P}$. Or $\mathcal{F}_f \subseteq \mathcal{F}'_f \cup \{\mathbb{T}\ @_x(z * y)\}$. Donc, comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_f \in \mathcal{P}$. Comme cela est vrai pour tout $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\}$, on en déduit que $\mathcal{F} \cup \{\mathbb{T}\ @_x(z * y)\} \in \mathcal{P}^{fc}$.
33. Si $\mathbb{T}\ @_x(y * (z * t)) \in \mathcal{F}$, alors on considère l'ensemble $\mathcal{F} \cup \{\mathbb{T}\ @_x((y * z) * t)\}$. Soit $\mathcal{F}_f$ un ensemble fini tel que $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x((y * z) * t)\}$. Soit $\mathcal{F}'_f$ l'ensemble défini par $\mathcal{F}'_f = (\mathcal{F}_f \backslash \{\mathbb{T}\ @_x((y * z) * t)\}) \cup \{\mathbb{T}\ @_x(y * (z * t))\}$. Par construction, $\mathcal{F}'_f$ est fini. De plus,

comme $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}$, on a $\mathcal{F}'_f \subseteq_f \mathcal{F}$. Comme $\mathcal{F} \in \mathcal{P}^{fc}$, on a alors que $\mathcal{F}'_f \in \mathcal{P}$. Comme $\mathbb{T}\ @_x(y*(z*t)) \in \mathcal{F}'_f$ et que $\mathcal{P}$ est une propriété de consistance alternée, on a $\mathcal{F}'_f \cup \{\mathbb{T}\ @_x((y*z)*t)\} \in \mathcal{P}$. Or $\mathcal{F}_f \subseteq \mathcal{F}'_f \cup \{\mathbb{T}\ @_x((y*z)*t)\}$. Donc, comme $\mathcal{P}$ est $\subseteq$-clos, on a $\mathcal{F}_f \in \mathcal{P}$. Comme cela est vrai pour tout $\mathcal{F}_f \subseteq_f \mathcal{F} \cup \{\mathbb{T}\ @_x((y*z)*t)\}$, on en déduit que $\mathcal{F} \cup \{\mathbb{T}\ @_x((y*z)*t)\} \in \mathcal{P}^{fc}$.

□

Finalement, toutes les propositions sont vérifiées avec l'ajout de ces nouveaux points et on peut étendre la preuve du Lemme :

**Lemme 5.15.** *Il existe un oracle qui contient tous les SHS finis pour lesquels il n'existe aucun HBBI-tableau clos.*

*Démonstration.*
On considère l'ensemble $\mathcal{P}$ de tous les SHS finis pour lesquels il n'existe aucun tableau clos. Par la Proposition B.7, $\mathcal{P}$ est une propriété de consistance. Par la Proposition B.8, on peut l'étendre en une propriété de consistance $\subseteq$-close. Par la Proposition B.9, on peut l'étendre en une propriété de consistance alternée $\subseteq$-close. Par la Proposition B.10, on peut l'étendre en une propriété de consistance alternée $\subseteq$-close de caractère fini. Par les conditions 4 à 30 de la Définition B.5, cette extension est saturée. Par les conditions 1 à 3 de la Définition B.5, cette extension ne contient que des SHS non clos. Cet ensemble est donc un oracle. Comme cet oracle est une extension de $\mathcal{P}$, il existe bien un oracle que contient $\mathcal{P}$. □

# Annexe C

# Preuves du Chapitre 6

## C.1 Preuve du Théorème 6.1

**Théorème 6.1.** *[Propriétés des tableaux labellisés] Soit $\mathcal{T}$ un HBBI-tableau pur.*

1. *$\lambda(\mathcal{T})$ est un BBI-tableau.*
2. *si $\mathcal{T}$ est clos, $\lambda(\mathcal{T})$ l'est aussi.*
3. *si $\lambda(\mathcal{T})$ est clos, $\mathcal{T}$ peut être étendu (par les règles du calcul pour HBBI) en un tableau clos.*

*Démonstration.*

1. On démontre ce résultat pour tous les tableaux purs par récurrence sur la taille de $\mathcal{T}$
   - Cas de base

     Si $\mathcal{T}$ est de taille 1, alors $\mathcal{T} = [\![\{\mathbb{F}\ @_{c_1}(\phi)\}]\!]$. Par définition on a alors $\lambda(\mathcal{T}) = [\![\langle\{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\}\rangle]\!]$. C'est un BBI-tableau à un seul CSS pour la formule $\phi$. Donc la propriété est vraie pour tous les HBBI-tableaux purs de taille 1.
   - Hérédité

     On suppose la propriété est vraie pour tous les HBBI-tableaux purs de taille $n$ (HR). On considère un tableau $\mathcal{T}$ de taille $n+1$. $\mathcal{T}$ a alors été obtenu par l'application d'une règle $R$ à un tableau $\mathcal{T}'$ de taille $n$. On discute selon le type de $R$. Dans toute la suite, on note $\langle\rangle_{BBI}$ les règles du calcul des tableaux de BBI (pour distinguer les deux calculs, les noms des règles étant identiques).
     - Comme $\mathcal{T}$ est pur, $R$ ne peut être ni $\langle @\rangle$, ni $\langle \mathbb{T}*\!\!-\rangle$, ni $\langle \mathbb{F}*\!\!-\rangle$.
     - Si $R = \langle \mathbb{T}\wedge\rangle$, alors il existe une formule $\mathbb{T}\ @_x(\phi\wedge\psi)$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\mathcal{F}]\!] \oplus \mathcal{T}_b$ et $\mathbb{T}\ @_x(\phi\wedge\psi) \in \mathcal{F}$. Alors, par construction, on a $\lambda(\mathcal{T}') = \lambda(\mathcal{T}_a) \oplus [\![\lambda(\mathcal{F})]\!] \oplus \lambda(\mathcal{T}_b)$ et $\lambda(\mathcal{F}) = \langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F}\rangle$ et $\mathbb{T}\phi\wedge\psi : x \in \mathcal{F}_\mathcal{F}$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\mathcal{F} \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}]\!] \oplus \mathcal{T}_b$, et donc $\lambda(\mathcal{T}) = \lambda(\mathcal{T}_a) \oplus [\![\langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{T}\phi : x, \mathbb{T}\psi : x\}, \mathcal{C}_\mathcal{F}\rangle]\!] \oplus \lambda(\mathcal{T}_b)$. Comme $\lambda(\mathcal{T}')$ est un tableau par (HR), $\lambda(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{T}\wedge\rangle_{BBI}$ à $\lambda(\mathcal{T}')$.

- Si $R = \langle \mathbb{F}\wedge \rangle$, alors il existe une formule $\mathbb{F}\ @_x(\phi \wedge \psi)$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\mathcal{F}]\!] \oplus \mathcal{F}_b$ et $\mathbb{F}\ @_x(\phi \wedge \psi) \in \mathcal{F}$. Alors, par construction, on a $\lambda(\mathcal{T}') = \lambda(\mathcal{T}_a) \oplus [\![\lambda(\mathcal{F})]\!] \oplus \lambda(\mathcal{T}_b)$ et $\lambda(\mathcal{F}) = \langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ et $\mathbb{F}\phi \wedge \psi : x \in \mathcal{F}_\mathcal{F}$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\mathcal{F} \cup \{\mathbb{F}\ @_x(\phi)\}; \mathcal{F} \cup \{\mathbb{F}\ @_x(\psi)\}]\!] \oplus \mathcal{T}_b$, et donc $\lambda(\mathcal{T}) = \lambda(\mathcal{T}_a) \oplus [\![\langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{F}\phi : x\}, \mathcal{C}_\mathcal{F}\rangle; \langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{F}\psi : x\}, \mathcal{C}_\mathcal{F}\rangle]\!] \oplus \lambda(\mathcal{T}_b)$. Comme $\lambda(\mathcal{T}')$ est un tableau par (HR), $\lambda(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{F}\wedge \rangle_{BBI}$ à $\lambda(\mathcal{T}')$.
- Si $R = \langle \mathbb{T}* \rangle$, alors il existe une formule $\mathbb{T}\ @_x(\phi * \psi)$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\mathcal{F}]\!] \oplus \mathcal{T}_b$ et $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}$. Alors, par construction, on a $\lambda(\mathcal{T}') = \lambda(\mathcal{T}_a) \oplus [\![\lambda(\mathcal{F})]\!] \oplus \lambda(\mathcal{T}_b)$ et $\lambda(\mathcal{F}) = \langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ et $\mathbb{T}\phi * \psi : x \in \mathcal{F}_\mathcal{F}$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\mathcal{F} \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}]\!] \oplus \mathcal{T}_b$, et donc $\lambda(\mathcal{T}) = \lambda(\mathcal{T}_a) \oplus [\![\langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{T}\phi : c_i, \mathbb{T}\psi : c_j\}, \mathcal{C}_\mathcal{F} \cup \{x \simeq c_i c_j\}\rangle]\!] \oplus \lambda(\mathcal{T}_b)$. Comme $\lambda(\mathcal{T}')$ est un tableau par (HR), $\lambda(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{T}* \rangle_{BBI}$ à $\lambda(\mathcal{T}')$.
- Si $R = \langle \mathbb{F}* \rangle$, alors il existe deux formules $\mathbb{F}\ @_x(\phi * \psi)$ et $\mathbb{T}\ @_x(y * z)$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\mathcal{F}]\!] \oplus \mathcal{T}_b$ et $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}$ et $\mathbb{T}\ @_x(y * z) \in \mathcal{F}$. Alors, par construction, on a $\lambda(\mathcal{T}') = \lambda(\mathcal{T}_a) \oplus [\![\lambda(\mathcal{F})]\!] \oplus \lambda(\mathcal{T}_b)$ et $\lambda(\mathcal{F}) = \langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ et $\mathbb{F}\phi * \psi : x \in \mathcal{F}_\mathcal{F}$. En outre, par le Lemme 6.2, on a, à substitution près, $\lambda(\mathbb{T}\ @_x(y * z)) \in \overline{\mathcal{C}_\mathcal{F}}$, c'est à dire $x \simeq yz \in \overline{\mathcal{C}_\mathcal{F}}$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}; \mathcal{F} \cup \{\mathbb{F}\ @_y(\phi)\}]\!] \oplus \mathcal{T}_b$, et donc $\lambda(\mathcal{T}) = \lambda(\mathcal{T}_a) \oplus [\![\langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{F}\phi : y\}, \mathcal{C}_\mathcal{F}\rangle; \langle \mathcal{F}_\mathcal{F} \cup \{\mathbb{F}\psi : z\}, \mathcal{C}_\mathcal{F}\rangle]\!] \oplus \lambda(\mathcal{T}_b)$. Comme $\lambda(\mathcal{T}')$ est un tableau par (HR), $\lambda(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{F}* \rangle_{BBI}$ à $\lambda(\mathcal{T}')$.
- Si $R$ est une règle à label, elle n'introduit pas de formule pure et donc $\lambda(\mathcal{T}) = \lambda(\mathcal{T}')$, ce qui prouve la propriété par récurrence.
- Les autres cas sont similaires.

Finalement la propriété est vraie dans tous les cas, ainsi $\lambda(\mathcal{T})$ est toujours un tableau.

2. On suppose que $\mathcal{T}$ est clos. Soit $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ un CSS de $\lambda(\mathcal{T})$. Il existe alors un SHS de $\mathcal{T}$, $\mathcal{F}$, dont $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est l'image. Par hypothèse, $\mathcal{F}$ est clos. On a alors 3 cas :
   - $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}$. On a alors $\mathbb{T}\phi : x \in \mathcal{F}_\mathcal{F}$ et $\mathbb{F}\phi : x \in \mathcal{F}_\mathcal{F}$. en outre, par $P_{CSS}$, on a $x \simeq x \in \mathcal{C}_\mathcal{F}$. Donc $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est clos.
   - $\mathbb{T}\ @_x(\bot) \in \mathcal{F}$. On a alors $\mathbb{T}\bot : x \in \mathcal{F}_\mathcal{F}$, donc $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est clos.
   - $\mathbb{F}\ @_x(\top) \in \mathcal{F}$. On a alors $\mathbb{F}\top : x \in \mathcal{F}_\mathcal{F}$, donc $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est clos.

   Dans tous les cas, $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est clos. Comme ceci est vrai pour tout CSS, $\lambda(\mathcal{T})$ est clos.
3. On suppose que $\lambda(\mathcal{T})$ est clos. Soit $\mathcal{F}$ un SHS de $\mathcal{T}$. Alors $\langle \mathcal{F}_\mathcal{F}, \mathcal{C}_\mathcal{F} \rangle$ est clos par hypothèse. On a 4 cas :
   - $\mathbb{T}\phi : x \in \mathcal{F}_\mathcal{F}$, $\mathbb{F}\phi : y \in \mathcal{F}_\mathcal{F}$ et $x \simeq y \in \overline{\mathcal{C}_\mathcal{F}}$. Alors $\mathbb{T}\ @_x(\phi) \in \mathcal{F}$ et $\mathbb{F}\ @_y(\psi) \in \mathcal{F}$. En outre, par un raisonnement similaire à celui du Lemme 6.2, on peut construire à partir de formules non pures de $\mathcal{F}$ et en utilisant des règles à labels la formule $\mathbb{T}\ @_y(x)$. Par la règle $\langle i_t \rangle$ on produit alors $\mathbb{T}\ @_y(\phi)$ et on a que $\mathcal{F}$ est clos.

- $\mathbb{F}$I : $x \in \mathcal{F}_\mathcal{F}$ et $x \simeq \varepsilon \in \overline{\mathcal{C}_\mathcal{F}}$. Alors $\mathbb{F}\ @_x(\mathrm{I}) \in \mathcal{F}$. En outre, par un raisonnement similaire à celui du Lemme 6.2, on peut construire à partir de formules non pures de $\mathcal{F}$ et en utilisant des règles à labels la formule $\mathbb{T}\ @_x(\mathrm{I})$. Alors $\mathcal{F}$ est clos.
- $\mathbb{F}\top$ : $x \in \mathcal{F}_\mathcal{F}$. Alors $\mathbb{F}\ @_x(\top) \in \mathcal{F}$ et $\mathcal{F}$ est clos.
- $\mathbb{T}\bot$ : $x \in \mathcal{F}_\mathcal{F}$. Alors $\mathbb{T}\ @_x(\bot) \in \mathcal{F}$ et $\mathcal{F}$ est clos.

Dans tous les cas, $\mathcal{F}$ est étendu en une branche close. Comme ceci est vrai pour tout SHS, $\mathcal{T}$ peut être étendu en un tableau clos.

□

## C.2 Preuve du Lemme 6.3

**Lemme 6.3.** *Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un CSS du calcul des tableaux de BBI. Soit $c \in \overline{\mathcal{C}}$ une contrainte produite par clôture de l'ensemble de contraintes $\mathcal{C}$. Il existe un ensemble $\mathcal{R}(c)$ de formules étiquetées du calcul des tableaux pour HBBI contenant $\delta(c)$ tel que pour tout $r \in \mathcal{R}(c)$ on ait soit :*

- *il existe $c' \in \mathcal{C}$ tel que $r \in \delta(c')$,*
- *ou il existe une règle à labels qui produit r à partir d'autres éléments de $\mathcal{R}(c)$.*

*Démonstration.*
On montre cela par récurrence sur la taille $n$ de la preuve pour obtenir $c$.

- Cas de base
  Si la preuve est de taille 0, on a deux cas :
  - $c \in \mathcal{C}$ et on peut choisir $\mathcal{R}(c) = \delta(c)$, donc la propriété est trivialement vraie.
  - $c$ est obtenue par la règle sans prémisses $\langle \varepsilon \rangle$. Alors $c = \varepsilon \simeq \varepsilon$ et $\delta(c) = \{\mathbb{T}\ @_e(I)\}$. L'ensemble $\mathcal{R}(c) = \delta(c)$ suffit, puisque la règle $\langle \mathbb{T}\mathrm{I} \rangle$ produit justement cette formule.
- Hérédité
  On suppose que la propriété est vraie pour toutes les contraintes $c$ ayant un arbre de preuve de taille $n$ (HR), on démontre pour les arbres de taille $n+1$. Si $c$ est obtenue par un arbre de taille $n+1$, alors elle est obtenue par l'application d'une règle $R$ sur la conclusion d'un arbre $T$ de taille $n$. On discute selon la nature de $R$ :
  - $R = \langle s_r \rangle$. Alors $c = y \simeq x$ et $T$ a produit $c' = x \simeq y$.
    - Si $x$ et $y$ sont simples, $\delta(c) = \{\mathbb{T}\ @_y(x)\}$ et $\delta(c') = \{\mathbb{T}\ @_x(y)\}$. Alors la règle $\langle i_s \rangle$ permet d'obtenir $\delta(c)$ à partir de $\delta(c')$, et donc $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$ convient.
    - Si $x$ est composé et $y$ est simple, $\delta(c) = \{\mathbb{T}\ @_y(d(x))\}$ et $\delta(c') = \{\mathbb{T}\ @_z(y), \mathbb{T}\ @_z(d(x))\}$. Alors les règles $\langle i_t \rangle$ et $\langle i_s \rangle$ permettent d'obtenir $\delta(c)$ à partir de $\delta(c')$, et donc $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup \{\mathbb{T}\ @_y(z)\}$ convient.

- Si $x$ est simple et $y$ est composée, $\delta(c) = \{\mathbb{T}\ @_z(x), \mathbb{T}\ @_z(d(y))\}$ et $\delta(c') = \{\mathbb{T}\ @_x(d(y))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_z(d(y))$ et $\mathbb{T}\ @_x(z)$, en utilisant un ensemble $F$ de formules intermédiaires. On produit alors $\delta(c)$ par une application de la règle $\langle i_s\rangle$, et $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_x(z)\}$ convient.
- Si $x$ est composée et $y$ est composée, $\delta(c) = \{\mathbb{T}\ @_z(d(x)), \mathbb{T}\ @_z(d(y))\}$ et $\delta(c') = \{\mathbb{T}\ @_z(d(y)), \mathbb{T}\ @_z(d(x))\}$. Alors $\mathcal{R}(c) = \mathcal{R}(c')$ convient.

- $R = \langle d_r\rangle$. Alors $c = x \simeq x$ et $T$ a produit $c' = xy \simeq xy$.
  - Si $x$ est simple, $\delta(c) = \{\mathbb{T}\ @_x(x)\}$. Alors la règle $\langle i_r\rangle$ permet d'obtenir $\delta(c)$, et donc $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$ convient.
  - Si $x$ est composé, $\delta(c) = \{\mathbb{T}\ @_z(d(x)), \mathbb{T}\ @_z(d(x))\} = \{\mathbb{T}\ @_z(d(x))\}$ et $\delta(c') = \{\mathbb{T}\ @_{z'}(d(x) * d(y))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_z(d(x))$ et $\mathbb{T}\ @_{z'}(z * d(y))$, en utilisant un ensemble $F$ de formules intermédiaires. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_{z'}(z * y)\}$ convient.

- $R = \langle t_r\rangle$. Alors $c = x \simeq z$ et $T$ a produit $c' = x \simeq y$ et $c'' = y \simeq z$.
  - Si $x$ et $y$ sont simples, $\delta(c) = \{\mathbb{T}\ @_x(d(z))\}$ et $\delta(c') = \{\mathbb{T}\ @_x(y)\}$ et $\delta(c'') = \{\mathbb{T}\ @_y(d(z))\}$. Alors la règle $\langle i_p\rangle$ permet d'obtenir $\delta(c)$, et donc $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$ convient.
  - Si $x$ est composé et $y$ est simple, $\delta(c) = \{\mathbb{T}\ @_t(d(z)), \mathbb{T}\ @_t(d(x))\}$ et $\delta(c') = \{\mathbb{T}\ @_{t'}(y), \mathbb{T}\ @_{t'}(d(x))\}$ et $\delta(c'') = \{\mathbb{T}\ @_y(d(z))\}$. Alors la règle $\langle i_p\rangle$ permet d'obtenir $\delta(c)$ avec $t = t'$, et donc $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$ convient.
  - Si $x$ est simple et $y$ est composé, $\delta(c) = \{\mathbb{T}\ @_x(d(z))\}$ et $\delta(c') = \{\mathbb{T}\ @_x(d(y))\}$ et $\delta(c'') = \{\mathbb{T}\ @_t(d(z)), \mathbb{T}\ @_t(d(y))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_x(t')$ et $\mathbb{T}\ @_{t'}(d(y))$ (par $\mathbb{T}\ @_x(d(y))$) et aussi les formules $\mathbb{T}\ @_t(t')$ et $\mathbb{T}\ @_{t'}(d(y))$ (par $\mathbb{T}\ @_t(d(y))$) , en utilisant un ensemble $F$ de formules intermédiaires. Puis on produit $\mathbb{T}\ @_{t'}(t)$ par $\langle i_s\rangle$, $\mathbb{T}\ @_x(t)$ par $\langle i_p\rangle$ et enfin $\mathbb{T}\ @_x(d(z))$ par $\langle i_p\rangle$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_x(t'), \mathbb{T}\ @_{t'}(d(y)), \mathbb{T}\ @_t(t'), \mathbb{T}\ @_{t'}(t), \mathbb{T}\ @_x(t)\}$ convient.
  - Si $x$ et $y$ sont composés, $\delta(c) = \{\mathbb{T}\ @_t(d(z)), \mathbb{T}\ @_t(d(x))\}$ et $\delta(c') = \{\mathbb{T}\ @_{t'}(d(y)), \mathbb{T}\ @_{t'}(d(x))\}$ et $\delta(c'') = \{\mathbb{T}\ @_{t''}(d(z)), \mathbb{T}\ @_{t''}(d(y))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_{t'}(u)$ et $\mathbb{T}\ @_u(d(y))$ (par $\mathbb{T}\ @_{t'}(d(y))$) et aussi les formules $\mathbb{T}\ @_{t''}(u)$ et $\mathbb{T}\ @_u(d(y))$ (par $\mathbb{T}\ @_{t''}(d(y))$) , en utilisant un ensemble $F$ de formules intermédiaires. Ensuite, avec deux applications des règles $\langle i_t\rangle$ et $\langle i_s\rangle$, on produit $\delta(c)$ avec $t = u$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_{t'}(u), \mathbb{T}\ @_u(t'), \mathbb{T}\ @_u(d(y)), \mathbb{T}\ @_{t''}(u), \mathbb{T}\ @_u(t''), \mathbb{T}\ @_u(d(y))\}$ convient.

- $R = \langle c_r\rangle$. Alors $c = xk \simeq yk$ et $T$ a produit $c' = x \simeq y$ et $c'' = yk \simeq yk$.
  - Si $x$ et $y$ sont simples, $\delta(c) = \{\mathbb{T}\ @_z(x * d(k)), \mathbb{T}\ @_z(y * d(k))\}$ et $\delta(c') = \{\mathbb{T}\ @_x(y)\}$ et $\delta(c'') = \{\mathbb{T}\ @_t(y * d(k))\}$. Alors la règle $\langle i_s\rangle$ produit $\mathbb{T}\ @_y(x)$ et la règle $\langle i_p\rangle$ produit $\mathbb{T}\ @_t(x * d(k))$, ce qui permet d'obtenir $\delta(c)$ avec $z = t$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup \{\mathbb{T}\ @_y(x)\}$ convient.

- Si $x$ est composé et $y$ est simple, $\delta(c) = \{\mathbb{T}\ @_z(d(x)*d(k)), \mathbb{T}\ @_z(y*d(k))\}$ et $\delta(c') = \{\mathbb{T}\ @_u(y), \mathbb{T}\ @_u(d(x))\}$ et $\delta(c'') = \{\mathbb{T}\ @_t(y*d(k))\}$. Alors la règle $\langle i_s\rangle$ produit $\mathbb{T}\ @_y(u)$ et la règle $\langle i_p\rangle$ produit $\mathbb{T}\ @_y(d(x))$ puis une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ produit $\mathbb{T}\ @_t(d(x)*d(k))$ en utilisant un ensemble $F$ de formules intermédiaires, ce qui permet d'obtenir $\delta(c)$ avec $z = t$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_y(u), \mathbb{T}\ @_y(d(x))\}$ convient.
- Si $x$ est simple et $y$ est composé, $\delta(c) = \{\mathbb{T}\ @_z(x*d(k)), \mathbb{T}\ @_z(d(y)*d(k))\}$ et $\delta(c') = \{\mathbb{T}\ @_x(d(y))\}$ et $\delta(c'') = \{\mathbb{T}\ @_t(d(y)*d(k))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_x(u)$ et $\mathbb{T}\ @_u(d(y))$ (par $\mathbb{T}\ @_x(d(y))$) et aussi les formules $\mathbb{T}\ @_t(u*d(k))$ et $\mathbb{T}\ @_u(d(y))$ (par $\mathbb{T}\ @_t(d(y)*d(k))$), en utilisant un ensemble $F$ de formules intermédiaires. Ensuite, par la règle $\langle i_s\rangle$ on produit $\mathbb{T}\ @_u(x)$, puis par $\langle i_p\rangle$ $\mathbb{T}\ @_t(x*d(k))$, ce qui permet d'obtenir $\delta(c)$ avec $z = t$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_x(u), \mathbb{T}\ @_t(u*d(k)), \mathbb{T}\ @_u(d(y)), \mathbb{T}\ @_u(x)\}$ convient.
- Si $x$ et $y$ sont composés, $\delta(c) = \{\mathbb{T}\ @_z(d(x)*d(k)), \mathbb{T}\ @_z(d(y)*d(k))\}$ et $\delta(c') = \{\mathbb{T}\ @_t(d(y)), \mathbb{T}\ @_t(d(x))\}$ et $\delta(c'') = \{\mathbb{T}\ @_u(d(y)*d(k))\}$. Alors on peut produire par une série d'application des règles $\langle i+\rangle$ et $\langle i-\rangle$ les formules $\mathbb{T}\ @_t(v)$ et $\mathbb{T}\ @_v(d(y))$ (par $\mathbb{T}\ @_t(d(y))$) et aussi les formules $\mathbb{T}\ @_u(v*d(k))$ et $\mathbb{T}\ @_v(d(y))$ (par $\mathbb{T}\ @_u(d(y)*d(k))$), en utilisant un ensemble $F$ de formules intermédiaires. Ensuite, par la règle $\langle i_s\rangle$ on produit $\mathbb{T}\ @_v(t)$, puis par $\langle i_p\rangle$ $\mathbb{T}\ @_u(t*d(k))$ et enfin $\mathbb{T}\ @_u(d(x)*d(k))$ , ce qui permet d'obtenir $\delta(c)$ avec $z = u$. Alors $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c) \cup F \cup \{\mathbb{T}\ @_t(v), \mathbb{T}\ @_u(v*d(k)), \mathbb{T}\ @_v(d(y)), \mathbb{T}\ @_v(t), \mathbb{T}\ @_u(t*d(k))\}$ convient.

- Si $R$ est une utilisation de l'associativité sur les nominaux. On peut se contenter de traduire l'associativité à droite des égalités (quitte à appliquer la règle $\langle s_r\rangle$) et uniquement sur des labels simples (quitte à appliquer à nouveau associativité et commutativité pour associer des groupes de labels) et sans contextes (quitte à les rajouter par la règle $\langle c_r\rangle$). On peut aussi se contenter de la transitivité à droite des compositions (quitte à utiliser à nouveau associativité et commutativité). $R$ serait donc la règle suivante (qui est implicite dans le véritable calcul des tableaux) :

$$\frac{x \simeq y(zt)}{x \simeq (yz)t}\ ass$$

  On a donc $\delta(c) = \{\mathbb{T}\ @_x((y*z)*t)\}$ et $\delta(c') = \{\mathbb{T}\ @_x(y*(z*t))\}$. On peut alors produire $\delta(c)$ à partir de $\delta(c')$ par la règle $\langle BI_a\rangle$ et on a $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$.

- Si $R$ est une utilisation de la commutativité sur les nominaux. On peut se contenter de traduire la commutativité à droite des égalités (quitte à appliquer la règle $\langle s_r\rangle$) et uniquement sur des labels simples (quitte à appliquer à nouveau associativité et commutativité pour commuter des groupes de labels) et sans contextes (quitte à les rajouter par la règle $\langle c_r\rangle$). $R$ serait donc la règle suivante (qui est implicite dans le véritable calcul des tableaux) :

$$\frac{x \simeq yz}{x \simeq zy}\ com$$

On a donc $\delta(c) = \{\mathbb{T}\ @_x(y * z)\}$ et $\delta(c') = \{\mathbb{T}\ @_x(z * y)\}$. On peut alors produire $\delta(c)$ à partir de $\delta(c')$ par la règle $\langle BI_c \rangle$ et on a $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$.

- Si $R$ est une utilisation de l'élément neutre $\varepsilon$ sur les nominaux. On peut se contenter de traduire la neutralité à droite des égalités (quitte à appliquer la règle $\langle s_r \rangle$) et uniquement sur des labels simples (quitte à appliquer à nouveau associativité et commutativité pour simplifier l'élément neutre dans des groupes de labels) et sans contextes (quitte à les rajouter par la règle $\langle c_r \rangle$). $R$ serait donc la règle suivante (qui est implicite dans le véritable calcul des tableaux) :

$$\frac{}{x \simeq x\varepsilon}\ ^{neu}$$

On a donc $\delta(c) = \{\mathbb{T}\ @_x(x * \mathrm{I})\}$. On peut alors produire $\delta(c)$ par la règle $\langle BI_n \rangle$ et on a $\mathcal{R}(c) = \mathcal{R}(c') \cup \delta(c)$.

Finalement, dans tous les cas, on parvient à trouver un ensemble $\mathcal{R}(c)$ qui convient, pour toutes les contraintes $c$ ayant un arbre de taille $n+1$. Alors par récurrence, $\mathcal{R}(c)$ existe pour toute contrainte $c \in \overline{C}$.

□

## C.3 Preuve du Théorème 6.2

**Théorème 6.2.** *[Propriétés des tableaux délabellisés] Soit $\mathcal{T}$ un BBI-tableau.*

1. *$\delta(\mathcal{T})$ est un HBBI-tableau pur.*
2. *si $\mathcal{T}$ est clos, $\delta(\mathcal{T})$ peut être étendu (par les règles du calcul pour HBBI) en un tableau clos.*
3. *si $\delta(\mathcal{T})$ est clos, alors $\mathcal{T}$ est clos.*

*Démonstration.*

1. On démontre ce résultat pour tous les tableaux par récurrence sur la taille $n$ de $\mathcal{T}$
   - Cas de base
     Si $\mathcal{T}$ est de taille 1, alors $\mathcal{T} = [\![\langle\{\mathbb{F}\phi : c_1\}, \{c_1 \simeq c_1\}\rangle]\!]$. Par définition on a alors $\lambda(\mathcal{T}) = [\![\{\mathbb{F}\ @_{c_1}(\phi)\}]\!]$. C'est un HBBI-tableau pur à un seul SHS pour la formule $\phi$. Donc la propriété est vraie pour tous les BBI-tableaux de taille 1.
   - Hérédité On suppose la propriété est vraie pour tous les BBI-tableaux de taille $n$ (HR). On considère un tableau $\mathcal{T}$ de taille $n+1$. $\mathcal{T}$ a alors été obtenu par l'application d'une règle $R$ à un tableau $\mathcal{T}'$ de taille $n$. On discute selon le type de $R$. Dans toute la suite, on note $\langle\rangle_{BBI}$ les règles du calcul des tableaux de BBI (pour distinguer les deux calculs, les noms des règles étant identiques).
     - Si $R = \langle \mathbb{T}\mathrm{I} \rangle_{BBI}$, alors il existe une formule $\mathbb{T}\mathrm{I} : x$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C} \rangle]\!] \oplus \mathcal{T}_b$ et $\mathbb{T}\mathrm{I} : x \in \mathcal{F}$. Alors, par construction, on a $\delta(\mathcal{T}') = \delta(\mathcal{T}_a) \oplus [\![\delta(\langle \mathcal{F}, \mathcal{C} \rangle)]\!] \oplus \delta(\mathcal{T}_b)$ et $\delta(\langle \mathcal{F}, \mathcal{C} \rangle) = \mathcal{F}'$ et $\mathbb{T}\ @_x(\mathrm{I}) \in \mathcal{F}'$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C} \cup \{x \simeq \varepsilon\} \rangle]\!] \oplus \mathcal{T}_b$, et donc $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{T}\ @_x(\mathrm{I})\}]\!] \oplus \delta(\mathcal{T}_b)$.On a ainsi $\delta(\mathcal{T}) = \delta(\mathcal{T}')$ et donc la propriété tient dans ce cas par (HR).

- Si $R = \langle \mathbb{T}\wedge\rangle_{BBI}$, alors il existe une formule $\mathbb{T}\phi\wedge\psi : x$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$ et $\mathbb{T}\phi\wedge\psi : x \in \mathcal{F}$. Alors, par construction, on a $\delta(\mathcal{T}') = \delta(\mathcal{T}_a) \oplus [\![\delta(\langle \mathcal{F}, \mathcal{C}\rangle)]\!] \oplus \delta(\mathcal{T}_b)$ et $\delta(\langle \mathcal{F}, \mathcal{C}\rangle) = \mathcal{F}'$ et $\mathbb{T}\ @_x(\phi\wedge\psi) \in \mathcal{F}'$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\langle \mathcal{F} \cup \{\mathbb{T}\phi : x, \mathbb{T}\psi : x\}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$, et donc $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{T}\ @_x(\phi), \mathbb{T}\ @_x(\psi)\}]\!] \oplus \delta(\mathcal{T}_b)$. Comme $\delta(\mathcal{T}')$ est un tableau par (HR), $\delta(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{T}\wedge\rangle$ à $\delta(\mathcal{T}')$.

  Dans le cas où $x$ est composé, on a $\mathbb{T}\ @_c(\phi\wedge\psi) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(d(x)) \in \mathcal{F}'$. On peut alors choisir à nouveau $c$ pour construire $\delta(\mathcal{T})$ à partir de $x$ et on a $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{T}\ @_c(\phi), \mathbb{T}\ @_c(\psi), \mathbb{T}\ @_c(d(x))\}]\!] \oplus \delta(\mathcal{T}_b)$ ce qui correspond à nouveau à l'application de la règle $\langle \mathbb{T}\wedge\rangle$.

- Si $R = \langle \mathbb{F}\wedge\rangle_{BBI}$, alors il existe une formule $\mathbb{F}\phi\wedge\psi : x$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$ et $\mathbb{F}\phi\wedge\psi : x \in \mathcal{F}$. Alors, par construction, on a $\delta(\mathcal{T}') = \delta(\mathcal{T}_a) \oplus [\![\delta(\langle \mathcal{F}, \mathcal{C}\rangle)]\!] \oplus \delta(\mathcal{T}_b)$ et $\delta(\langle \mathcal{F}, \mathcal{C}\rangle) = \mathcal{F}'$ et $\mathbb{F}\ @_x(\phi\wedge\psi) \in \mathcal{F}'$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\langle \mathcal{F} \cup \{\mathbb{F}\phi : x\}, \mathcal{C}\rangle; \langle \mathcal{F} \cup \{\mathbb{F}\psi : x\}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$, et donc $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{F}\ @_x(\phi)\}; \mathcal{F}' \cup \{\mathbb{F}\ @_x(\psi)\}]\!] \oplus \delta(\mathcal{T}_b)$. Comme $\delta(\mathcal{T}')$ est un tableau par (HR), $\delta(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{F}\wedge\rangle$ à $\delta(\mathcal{T}')$.

  Dans le cas où $x$ est composé, on a $\mathbb{F}\ @_c(\phi\wedge\psi) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(d(x)) \in \mathcal{F}'$. On peut alors choisir à nouveau $c$ pour construire $\delta(\mathcal{T})$ à partir de $x$ et on a $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{F}\ @_c(\phi), \mathbb{T}\ @_c(d(x))\}; \mathcal{F}' \cup \{\mathbb{F}\ @_c(\psi), \mathbb{T}\ @_c(d(x))\}]\!] \oplus \delta(\mathcal{T}_b)$ ce qui correspond à nouveau à l'application de la règle $\langle \mathbb{F}\wedge\rangle$.

- Si $R = \langle \mathbb{T}*\rangle_{BBI}$, alors il existe une formule $\mathbb{T}\phi * \psi : x$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$ et $\mathbb{T}\phi * \psi : x \in \mathcal{F}$. Alors, par construction, on a $\delta(\mathcal{T}') = \delta(\mathcal{T}_a) \oplus [\![\delta(\langle \mathcal{F}, \mathcal{C}\rangle)]\!] \oplus \delta(\mathcal{T}_b)$ et $\delta(\langle \mathcal{F}, \mathcal{C}\rangle) = \mathcal{F}'$ et $\mathbb{T}\ @_x(\phi * \psi) \in \mathcal{F}'$. On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\langle \mathcal{F} \cup \{\mathbb{T}\phi : c_i, \mathbb{T}\psi : c_j\}, \mathcal{C} \cup \{x \simeq c_i c_j\}\rangle]\!] \oplus \mathcal{T}_b$, et donc $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_x(c_i * c_j)\}]\!] \oplus \delta(\mathcal{T}_b)$. Comme $\delta(\mathcal{T}')$ est un tableau par (HR), $\delta(\mathcal{T})$ l'est aussi par application de la règle $\langle \mathbb{T}*\rangle$ à $\delta(\mathcal{T}')$.

  Dans le cas où $x$ est composé, on a $\mathbb{T}\ @_c(\phi * \psi) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(d(x)) \in \mathcal{F}'$. On peut alors choisir à nouveau $c$ pour construire $\delta(\mathcal{T})$ à partir de $x$ et on a $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{T}\ @_{c_i}(\phi), \mathbb{T}\ @_{c_j}(\psi), \mathbb{T}\ @_c(c_i * c_j), \mathbb{T}\ @_c(d(x))\}]\!] \oplus \delta(\mathcal{T}_b)$ ce qui correspond à nouveau à l'application de la règle $\langle \mathbb{T}*\rangle$.

- Si $R = \langle \mathbb{F}*\rangle_{BBI}$, alors il existe une formule $\mathbb{F}\phi * \psi : x$ dans $\mathcal{T}'$ sur laquelle la règle est appliquée. C'est à dire que $\mathcal{T}' = \mathcal{T}_a \oplus [\![\langle \mathcal{F}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$ et $\mathbb{F}\phi * \psi : x \in \mathcal{F}$ et $x \simeq yz \in \overline{\mathcal{C}}$. Alors, par construction, on a $\delta(\mathcal{T}') = \delta(\mathcal{T}_a) \oplus [\![\delta(\langle \mathcal{F}, \mathcal{C}\rangle)]\!] \oplus \delta(\mathcal{T}_b)$ et $\delta(\langle \mathcal{F}, \mathcal{C}\rangle) = \mathcal{F}'$ et $\mathbb{F}\ @_x(\phi * \psi) \in \mathcal{F}'$. En outre, comme $x \simeq yz \in \mathit{Req}$ et que $\delta(x \simeq yz) \in \mathcal{R}(x \simeq yz)$, on a par construction que $\delta(x \simeq yz) \in \mathcal{F}'$, c'est à dire que $\mathbb{T}\ @_x(d(y) * d(z)) \in \mathcal{F}'$, c'est à dire $\mathbb{T}\ @_x(y * z) \in \mathcal{F}'$ si $y$ et $z$ sont non composés.

  On a alors $\mathcal{T} = \mathcal{T}_a \oplus [\![\langle \mathcal{F} \cup \{\mathbb{F}\phi : y\}, \mathcal{C}\rangle; \langle \mathcal{F} \cup \{\mathbb{F}\psi : z\}, \mathcal{C}\rangle]\!] \oplus \mathcal{T}_b$, et donc $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{F}\ @_y(\phi)\} \cup \mathcal{R}(x \simeq yz); \mathcal{F}' \cup \{\mathbb{F}\ @_z(\psi)\} \cup \mathcal{R}(x \simeq yz)]\!] \oplus \delta(\mathcal{T}_b)$. Comme $\delta(\mathcal{T}')$ est un tableau par (HR), $\delta(\mathcal{T})$ l'est aussi par application de la

règle $\langle \mathbb{F}* \rangle$ à $\delta(\mathcal{T}')$, sous réserve que l'ajout de $\mathcal{R}(x \simeq yz)$ en fasse toujours un tableau (ce que nous démontrerons plus loin).

Dans le cas où $x$ est composé, on a $\mathbb{F}\ @_c(\phi * \psi) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(d(x)) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(y*z) \in \mathcal{F}'$, ce qui ne change pas le raisonnement.

Dans le cas où $y$ est composé, on a $\mathbb{T}\ @_x(c*z) \in \mathcal{F}'$ et $\mathbb{T}\ @_c(d(y)) \in \mathcal{F}'$, et on a $\delta(\mathcal{T}) = \delta(\mathcal{T}_a) \oplus [\![\mathcal{F}' \cup \{\mathbb{F}\ @_c(\phi)\} \cup \mathcal{R}(x \simeq yz); \mathcal{F}' \cup \{\mathbb{F}\ @_z(\psi)\} \cup \mathcal{R}(x \simeq yz)]\!] \oplus \delta(\mathcal{T}_b)$ avec toujours $\mathbb{T}\ @_c(d(y)) \in \mathcal{F}'$ , ce qui correspond à nouveau à l'application de $\langle \mathbb{F}* \rangle$. Même raisonnement si $z$ est composé.

En outre, pour toute formule $r \in \mathcal{R}(x \simeq yz)$, on a :

- soit il existe $c' \in \mathcal{C}$ tel que $r \in \delta(c')$ et donc, par construction on a déjà $r \in \delta(\mathcal{T}')$
- soit $r$ peut être déduit d'une autre formule de $\mathcal{R}(x \simeq yz)$ par une règle du calcul des tableaux de HBBI et donc le résultat est encore un tableau.

Finalement l'ajout de $r \in \mathcal{R}(x \simeq yz)$ maintient la structure de tableau de $\delta(\mathcal{T})$ tout en assurant que $\delta(x \simeq yz) \in \delta(\mathcal{T})$.

- Les autres cas sont similaires.

Finalement, pour tout tableau $\mathcal{T}$ de taille $n+1$, $\delta(\mathcal{T})$ est un HBBI-tableau pur, et par récurrence ce résultat est vrai pour tout tableau.

2. On suppose que $\mathcal{T}$ est clos. Soit $\mathcal{F}'$ un SHS de $\delta(\mathcal{T})$. Il existe alors un CSS de $\mathcal{T}$, $\langle \mathcal{F}, \mathcal{C} \rangle$ dont $\mathcal{F}'$ est l'image. Par hypothèse, $\langle \mathcal{F}, \mathcal{C} \rangle$ est clos. On a alors 4 cas :
   - $\mathbb{T}\phi : x \in \mathcal{F}, \mathbb{F}\phi : y \in \mathcal{F}$ et $x \simeq y \in \overline{\mathcal{C}}$. En outre, par le Lemme 6.3, on peut construire $\mathcal{R}(x \simeq y)$ et l'ajouter à $\delta(\mathcal{T})$ l'étendre en un tableau qui contienne $\delta(x \simeq y)$.
     - Si $x$ et $y$ sont simples, alors $\mathbb{T}\ @_x(\phi) \in \mathcal{F}'$, $\mathbb{F}\ @_y(\phi) \in \mathcal{F}'$ et $\mathbb{T}\ @_x(y) \in \mathcal{F}'$. On peut alors produire $\mathbb{T}\ @_y(\phi) \in \mathcal{F}'$ par la règle $\langle i_p \rangle$ et on a clôture de $\mathcal{F}'$.
     - Si $x$ est simple et $y$ est composé, alors $\mathbb{T}\ @_x(\phi) \in \mathcal{F}'$, $\mathbb{F}\ @_z(\phi) \in \mathcal{F}'$, $\mathbb{T}\ @_z(d(y)) \in \mathcal{F}'$ et $\mathbb{T}\ @_x(d(y)) \in \mathcal{F}'$. Alors on peut produire par une série d'application des règles $\langle i+ \rangle$ et $\langle i- \rangle$ les formules $\mathbb{T}\ @_z(t)$ et $\mathbb{T}\ @_t(d(y))$ (par $\mathbb{T}\ @_z(d(y))$) ainsi que $\mathbb{T}\ @_x(t)$ et $\mathbb{T}\ @_t(d(y))$ (par $\mathbb{T}\ @_x(d(y))$). On produit alors $\mathbb{T}\ @_t(z)$ et $\mathbb{T}\ @_t(x)$ par $\langle i_s \rangle$, et enfin $\mathbb{T}\ @_t(\phi)$, $\mathbb{F}\ @_t(\phi)$ par $\langle i_p \rangle$ et on a clôture de $\mathcal{F}'$.
     - Si $x$ est composé et $y$ est simple, alors $\mathbb{T}\ @_z(\phi) \in \mathcal{F}'$, $\mathbb{T}\ @_z(d(x)) \in \mathcal{F}'$ , $\mathbb{F}\ @_y(\phi) \in \mathcal{F}'$, $\mathbb{T}\ @_t(y) \in \mathcal{F}'$ et $\mathbb{T}\ @_t(d(x)) \in \mathcal{F}'$. Alors on peut produire par une série d'application des règles $\langle i+ \rangle$ et $\langle i- \rangle$ les formules $\mathbb{T}\ @_z(u)$ et $\mathbb{T}\ @_u(d(x))$ (par $\mathbb{T}\ @_z(d(x))$) ainsi que $\mathbb{T}\ @_t(u)$ et $\mathbb{T}\ @_u(d(x))$ (par $\mathbb{T}\ @_t(d(x))$). On produit alors $\mathbb{T}\ @_y(u)$ par $\langle i_t \rangle$ et $\langle i_s \rangle$, puis $\mathbb{T}\ @_u(\phi)$ et $\mathbb{F}\ @_u(\phi)$ par $\langle i_t \rangle$ et $\langle i_s \rangle$ et on a clôture de $\mathcal{F}'$.
     - Si $x$ et $y$ sont composés, alors $\mathbb{T}\ @_z(\phi) \in \mathcal{F}'$, $\mathbb{F}\ @_t(\phi) \in \mathcal{F}'$, $\mathbb{T}\ @_z(d(x)) \in \mathcal{F}'$, $\mathbb{T}\ @_t(d(y)) \in \mathcal{F}'$, $\mathbb{T}\ @_u(d(x)) \in \mathcal{F}'$ et $\mathbb{T}\ @_u(d(y)) \in \mathcal{F}'$. Alors on peut produire par une série d'application des règles $\langle i+ \rangle$ et $\langle i- \rangle$ les formules $\mathbb{T}\ @_z(v)$ et $\mathbb{T}\ @_v(d(x))$ (par $\mathbb{T}\ @_z(d(x))$) ainsi que $\mathbb{T}\ @_u(v)$ et $\mathbb{T}\ @_v(d(x))$ (par $\mathbb{T}\ @_u(d(x))$). On produit alors $\mathbb{T}\ @_v(u)$ par $\langle i_s \rangle$, puis $\mathbb{T}\ @_z(u)$ par $\langle i_p \rangle$,

puis $\mathbb{T}\ @_u(\phi)$ par $\langle i_t \rangle$ et $\langle i_s \rangle$. D'un autre côté, on peut produire par une série d'application des règles $\langle i+ \rangle$ et $\langle i- \rangle$ les formules $\mathbb{T}\ @_t(w)$ et $\mathbb{T}\ @_w(d(y))$ (par $\mathbb{T}\ @_t(d(y))$) ainsi que $\mathbb{T}\ @_u(w)$ et $\mathbb{T}\ @_w(d(y))$ (par $\mathbb{T}\ @_u(d(y))$). On produit alors $\mathbb{T}\ @_w(u)$ par $\langle i_s \rangle$, puis $\mathbb{T}\ @_t(u)$ par $\langle i_p \rangle$, puis $\mathbb{F}\ @_u(\phi)$ par $\langle i_t \rangle$ et $\langle i_s \rangle$ et on a clôture de $\mathcal{F}'$.

- $\mathbb{F}\mathrm{I} : x \in \mathcal{F}$ et $x \simeq \varepsilon \in \overline{\mathcal{C}}$. En outre, par le Lemme 6.3, on peut construire $\mathcal{R}(x \simeq \varepsilon)$ et l'ajouter à $\delta(\mathcal{T})$ l'étendre en un tableau qui contienne $\delta(x \simeq \varepsilon)$.
  - Si $x$ est simple, $\delta(x \simeq \varepsilon) = \{\mathbb{T}\ @_x(\mathrm{I})\}$ et $\mathbb{F}\ @_x(\mathrm{I}) \in \mathcal{F}'$, on a donc clôture.
  - Si $x$ est composé, $\delta(x \simeq \varepsilon) = \{\mathbb{T}\ @_y(\mathrm{I}), \mathbb{T}\ @_y(d(x))\}$, $\mathbb{F}\ @_z(\mathrm{I}) \in \mathcal{F}'$ et $\mathbb{T}\ @_z(d(x)) \in \mathcal{F}'$. Alors on peut produire par une série d'application des règles $\langle i+ \rangle$ et $\langle i- \rangle$ les formules $\mathbb{T}\ @_y(t)$ et $\mathbb{T}\ @_t(d(x))$ (par $\mathbb{T}\ @_y(d(x))$) ainsi que $\mathbb{T}\ @_z(t)$ et $\mathbb{T}\ @_t(d(x))$ (par $\mathbb{T}\ @_z(d(x))$). Alors deux applications de $\langle i_s \rangle$ et $\langle i_t \rangle$ produisent $\mathbb{T}\ @_t(\mathrm{I})$ et $\mathbb{F}\ @_t(\mathrm{I})$, ce qui clôt le SHS.
- $\mathbb{F}\top : x \in \mathcal{F}$. Alors $\mathbb{F}\ @_x(\top) \in \mathcal{F}'$ et $\mathcal{F}'$ est clos, ou bien $\mathbb{F}\ @_y(\top) \in \mathcal{F}'$ et $\mathbb{T}\ @_y(d(x)) \in \mathcal{F}'$ et $\mathcal{F}'$ est clos.
- $\mathbb{T}\bot : x \in \mathcal{F}$. Alors $\mathbb{T}\ @_x(\bot) \in \mathcal{F}'$ et $\mathcal{F}'$ est clos, ou bien $\mathbb{T}\ @_y(\bot) \in \mathcal{F}'$ et $\mathbb{T}\ @_y(d(x)) \in \mathcal{F}'$ et $\mathcal{F}'$ est clos.

Dans tous les cas, $\mathcal{F}'$ est étendu en une branche close. Comme ceci est vrai pour tout SHS, $\delta(\mathcal{T})$ peut être étendu en un tableau clos.

3. On suppose que $\delta(\mathcal{T})$ est clos. Soit $\langle \mathcal{F}, \mathcal{C} \rangle$ un CSS de $\mathcal{T}$. Il existe alors un SHS de $\mathcal{T}$, $\mathcal{F}'$ tel que $\delta(\mathcal{F}') = \langle \mathcal{F}, \mathcal{C} \rangle$. Par hypothèse, $\mathcal{F}'$ est clos. On a alors 3 cas :
   - $\mathbb{T}\ @_x(\phi) \in \mathcal{F}'$ et $\mathbb{F}\ @_x(\phi) \in \mathcal{F}'$. On a deux cas :
     - $x$ est un label simple de $\mathcal{T}$. Alors on doit avoir $\mathbb{T}\phi : x \in \mathcal{F}$ et $\mathbb{F}\phi : x \in \mathcal{F}$. En outre, comme $\langle \mathcal{F}, \mathcal{C} \rangle$ est un CSS, on a $x \simeq x \in \overline{\mathcal{C}}$, donc $\langle \mathcal{F}, \mathcal{C} \rangle$ est clos.
     - $x$ a été introduit par une règle $\langle i+ \rangle$. Comme $x$ est le même dans les deux règles, il doit exister un label composé $y$ tel que $\mathbb{T}\ @_x(d(y)) \in \mathcal{F}'$ et on a $\mathbb{T}\phi : y \in \mathcal{F}$ et $\mathbb{F}\phi : y \in \mathcal{F}$. Par le même raisonnement on a alors $\langle \mathcal{F}, \mathcal{C} \rangle$ clos.
   - $\mathbb{T}\ @_x(\bot) \in \mathcal{F}'$. On a alors $\mathbb{T}\bot : x \in \mathcal{F}$, ou alors il existe un label composé $y$ tel que $\mathbb{T}\ @_x(d(y)) \in \mathcal{F}'$ et on a $\mathbb{T}\bot : y \in \mathcal{F}$. Dans les deux cas, $\langle \mathcal{F}, \mathcal{C} \rangle$ est clos.
   - $\mathbb{F}\ @_x(\top) \in \mathcal{F}'$. On a alors $\mathbb{F}\top : x \in \mathcal{F}$, ou alors il existe un label composé $y$ tel que $\mathbb{T}\ @_x(d(y)) \in \mathcal{F}'$ et on a $\mathbb{F}\top : y \in \mathcal{F}$. Dans les deux cas, $\langle \mathcal{F}, \mathcal{C} \rangle$ est clos.

   Dans tous les cas, $\langle \mathcal{F}, \mathcal{C} \rangle$ est clos. Comme ceci est vrai pour tout CSS, $\mathcal{T}$ est clos.

□