



HAL
open science

Radiosité à base d'ondelettes sur des surfaces paramétriques

François Cuny

► **To cite this version:**

François Cuny. Radiosité à base d'ondelettes sur des surfaces paramétriques. Informatique [cs]. Institut National Polytechnique de Lorraine, 2000. Français. NNT : 2000INPL082N . tel-01750479

HAL Id: tel-01750479

<https://hal.univ-lorraine.fr/tel-01750479v1>

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>



Département de formation doctorale en informatique

Institut National
Polytechnique de Lorraine

École doctorale IAE + M

Radiosité à base d'ondelettes sur des surfaces paramétriques.

THÈSE

présentée et soutenue publiquement le vendredi 13 octobre 2000

pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine
(spécialité informatique)

par

François CUNY

Composition du jury

Rapporteurs :

Pr. Claude KIRCHNER
Pr. François SILLION
Pr. Pascal GUITTON

INRIA Lorraine
INRIA Rhône-Alpes
Université de Bordeaux

Examineur :

Pr. Jean-Laurent MALLET

INPL

Directeur de thèse :

Pr. Jean-Claude PAUL

INRIA Lorraine

Pr. ALONSO

INRIA Lorraine



D 136 024203 5

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



Mis en page avec la classe thloria.

136024 2035

i

[1] 2000 CUNY, F.

À Stéph

Remerciements

Merci à toutes les personnes qui ont rendu mon travail possible.

En premier lieu, je tiens à remercier tout particulièrement Jean-Claude PAUL pour la dynamique qu'il a insufflée à ce travail, pour son soutien de tous les jours, et pour l'intérêt constant qu'il a porté à ma thèse. Je voudrais aussi te remercier, Jean-Claude, pour l'excellente relation que nous avons pu avoir et qui m'a permis de mener à bien ce travail dans les meilleures conditions. Merci pour ta confiance.

Je remercie aussi les membres du jury, Claude KIRCHNER, François SILLION, Pascal GUITTON et Jean-Laurent MALLET de m'avoir fait l'honneur d'être les rapporteurs et examinateurs de ma thèse. Un merci tout spécial aux rapporteurs pour leur diligence entre la réception du manuscrit et la date de retour des rapports.

Je tiens spécialement à remercier The Black Lord OSNOLA (Laurent ALONSO pour les non initiés) pour les discussions fructueuses que nous avons eues, pour ses explications toujours brillantes, et pour son aide permanente.

Un grand merci à tous mes compagnons de labeur (et de foot), sans qui tout ce travail n'aurait pas été possible. J'ai nommé : toF le tacle WINKLER, Xav la bibine CAVIN, Hervé la tendinite BARTHÉLÉMY, Alm le transfuge FILBOIS et Sylvain le mut PETITJEAN.

Merci à tous les p'tits gars d'ISA pour leur soutien constant pendant et en dehors du travail et pour la bonne humeur.

Un merci tout particulier te reviens, Stéphan. Merci pour ton soutien durant ces trois années, merci pour ta relecture attentive du manuscrit et merci surtout pour ce que tu es pour moi.

Merci à mes parents pour leurs corrections orthographiques.

Enfin, merci à tous ceux qui m'ont permis d'en arriver là : la famille, les amis, les copains, les collègues... tous quoi.

Résumé

Dans cette thèse, nous introduisons une méthode générale pour effectuer le calcul de radiosit  dans des sc nes constitu es d'une grande vari t  de surfaces param triques. Contrairement aux approches pr c dentes qui n cessitaient une triangulation des surfaces en entr e, notre m thode b n ficie de la richesse de la repr sentation math matique de la g om trie. Au coeur de notre m thode se trouve le *Maillage Virtuel*, une abstraction de la g om trie initiale permettant d'appliquer les algorithmes classiques d'illumination   des surfaces complexes exactement comme si ces surfaces  taient de simples triangles. Le maillage virtuel est constitu  d'un ensemble de domaines normalis s sur lesquels les surfaces sont projet es gr ce   une fonction permettant de pr server leurs propri t s  nerg tiques. Les valeurs de radiosit  sont calcul es sur ces domaines normalis s avant d' tre re-projet es sur les surfaces originales.

Nous commen ons par d montrer l'efficacit  des fonctions ondelettes d'ordre  lev  quand elles sont utilis es au sein d'un algorithme coh rent pr vu pour g rer l'aspect fonctionnel de la fonction de radiosit . Ensuite, nous d crivons l'implantation dans cet algorithme du maillage virtuel pour les primitives planes   bords quelconques et les morceaux de surfaces quadriques. Nous donnons des exemples d'utilisation de notre m thode sur divers environnements de r alit  virtuelle. Gr ce   ces exemples nous montrons qu'en int grant de fa on exacte les surfaces courbes dans le processus de r solution, le maillage virtuel permet d'obtenir plus rapidement un rendu, de meilleure qualit  et de fa on plus naturelle qu'avec les m thodes pr c dentes.

Mots-cl s: Synth se d'images, illumination globale, radiance, radiosit , ondelettes, surfaces param triques, quadriques, maillage virtuel.

Abstract

In this thesis, we introduce a general-purpose method for computing radiosity on scenes made of a large variety of parametric surfaces (including arbitrary planar primitives and low-degree curved patches). By contrast with past approaches that require a tessellation of the input surfaces, our method takes advantage of the rich and compact mathematical representation of objects. At its core lies the *virtual mesh*, an abstraction of the input geometry that allows complex shapes to be illuminated as if they were simple primitives. The virtual mesh is a collection of normalized square domains to which the input surfaces are mapped while preserving their energetic properties. Radiosity values are then computed on these supports before being lifted back to the original surfaces.

To demonstrate the power of our method, we describe a high-order wavelet radiosity implementation that uses the virtual mesh. Before this description we show that, with an appropriated implementation, higher order wavelets are actually bringing an improvement over standard hierarchical radiosity and lower order wavelets. Then we describe the implementation of the virtual mesh on arbitrary planar primitives and quadric surface patches. Examples of objects and environments, designed for interactive applications or virtual reality, are presented. They prove that, by exactly integrating curved surfaces in the resolution process, the virtual mesh allows complex scenes to be rendered more quickly, more accurately and much more naturally than with previously known methods.

Keywords: Image synthesis, global illumination, radiance, radiosity, wavelets, parametric surfaces, quadrics, virtual mesh.

Sommaire

Introduction

Chapitre 1

Équation de radiance

1.1	Introduction	11
1.2	Modélisation physique	13
1.2.1	Aspects physiques	13
1.2.2	Expression du transport	14
1.2.3	Analyse de l'équation de radiance	16
1.2.4	Portée des hypothèses	19
1.3	Résolution mathématique	22
1.3.1	Analyse fonctionnelle	22
1.3.2	Approche variationnelle	25
1.3.3	Résolution du système linéaire	27
1.4	Aspects algorithmiques	31
1.4.1	Ensemble \mathcal{M}	31
1.4.2	Fonctions de base (ϕ_i)	33

Chapitre 2

Algorithme de radiosité à base d'ondelettes

2.1	Introduction	42
2.2	État de l'art	43
2.2.1	Oracle de subdivision	44
2.2.2	Calcul de la visibilité	48
2.2.3	Mémoire utilisée	51
2.3	Rendre efficace l'algorithme de radiosité à base d'ondelettes	51

2.3.1	Oracle de subdivision	52
2.3.2	Intégrer la visibilité	57
2.3.3	Algorithmes de radiosité sans liens d'interaction	60
2.4	Bases d'ondelettes	62
2.4.1	Ondelettes pour la méthode de GALERKIN	62
2.4.2	Bases pour la méthode de collocation de points	64
2.5	Comparaison de différentes bases	67
2.5.1	Protocole expérimental	68
2.5.2	Résultats	74
2.5.3	Discussion et comparaison de nos résultats avec ceux des études précédentes	80
2.6	Conclusion	80

Chapitre 3

Maillage virtuel

3.1	Introduction	84
3.2	État de l'art	90
3.2.1	Trois raisons pour utiliser des carrés normalisés	90
3.2.2	Recherches antérieures pour gérer les surfaces complexes	92
3.2.3	Nos contributions	94
3.3	Principe du maillage virtuel	95
3.3.1	Passage des primitives quelconques aux carrés normalisés	96
3.3.2	Fonction de radiosité sur le maillage virtuel	102
3.3.3	Déduire la radiosité sur les primitives originales	107
3.4	Maillage virtuel et radiosité à base d'ondelettes	108
3.4.1	Algorithme	108
3.4.2	Coefficients de <i>push-pull</i>	109
3.4.3	Primitives auto-émettrices	114
3.4.4	Gérer la hiérarchie	114
3.5	Applications	115
3.5.1	Primitives planes quelconques	116
3.5.2	Quadriques	116
3.6	Résultats expérimentaux	125
3.6.1	Résultats avec des surfaces planes à bords complexes	126
3.6.2	Résultats avec les surfaces paramétriques	129

3.7 Conclusion	135
--------------------------	-----

Chapitre 4

Expérimentations

4.1 Protocole expérimental	138
4.1.1 Caractéristiques matérielles et logicielles	139
4.1.2 Critères de performance	139
4.1.3 Scènes de test	141
4.2 Résultats	146
4.2.1 Modélisation plus naturelle	147
4.2.2 Simulations plus efficace	148
4.2.3 Moins de place en mémoire	148
4.2.4 Simulations plus rapides	149
4.2.5 Simulations très réalistes	150
4.3 Autres solutions	155
4.3.1 Algorithme parallèles	156
4.3.2 <i>Clustering</i>	156
4.3.3 <i>Face cluster</i>	157

Conclusion

Annexe A

Intégrale elliptique.

A.1 Intégrales elliptiques de première espèce	165
A.1.1 Définitions	165
A.1.2 AGM et transformations de LANDEN	166
A.1.3 Inverses des intégrales elliptiques de première espèce	168
A.1.4 Les fonctions Theta	170
A.1.5 Approximations des fonctions elliptiques	172
A.2 Intégrales elliptiques de seconde espèce	173
A.2.1 Définitions	173
A.2.2 Inverses des fonctions intégrales de seconde espèce	175
A.2.3 Approximations	176

Annexe B**Projections en 2D**

B.1 Ellipse dont les axes sont orthogonaux.	179
B.2 Ellipse dont les axes ne sont pas orthogonaux	180
B.3 Hyperbole dont les axes sont orthogonaux	181
B.4 Hyperbole dont les axes ne sont pas orthogonaux.	182
B.5 Parabole dont les axes sont orthogonaux	183
B.6 Parabole dont les axes ne sont pas orthogonaux.	184

Annexe C**Les quadriques**

C.1 Définition et notations	185
C.2 Classification générale	185
C.2.1 Cas où les valeurs propres sont toutes non-nulles	186
C.2.2 Cas où une des valeurs propres est nulle	187
C.3 Les quadriques et leur fonction de projection à Jacobien constant	189
C.3.1 Ellipsoïde	189
C.3.2 Hyperboloïde à une nappe	190
C.3.3 Hyperboloïde à deux nappes	191
C.3.4 Paraboloides elliptique	191
C.3.5 Paraboloides hyperbolique	192
C.3.6 Cône	193
C.3.7 Cylindres	193

Bibliographie

Table des figures

1.1	Efficacités lumineuses relatives pour la vision diurne et nocturne.	15
1.2	Schéma pour l'équation de radiance.	16
1.3	Visualisation de l'effet des opérateurs K et G (d'après J. Arvo).	25
1.4	Subdivisions hiérarchiques	35
1.5	Subdivisions et interactions hiérarchiques	36
1.6	Approximation avec une base de fonctions constantes.	36
1.7	Approximation avec une base de fonctions non constantes.	37
1.8	Construction d'une base linéaire en 2D	37
2.1	Calcul de l'erreur par majoration des coefficients	46
2.2	Évolution du maillage dans un coin en fonction de la précision.	48
2.3	Effet de l'utilisation des masques de visibilité.	50
2.4	Passage des points de contrôles sur un carré aux points de contrôle correspondants sur un cylindre.	53
2.5	Effet de la prise en compte de la validité de la fonction émettrice.	55
2.6	Effet de la mise à l'échelle d'un émetteur	56
2.7	Maillage due à la prise en compte de la visibilité dans l'oracle.	59
2.8	Effet de l'augmentation du nombre de points de quadrature (n) en cas d'erreur de visibilité.	60
2.9	$\mathcal{M}_{00}(u,v)$ — Polynôme de LEGENDRE constant.	63
2.10	Polynômes de LEGENDRE bi-linéaires.	63
2.11	Polynômes de LEGENDRE bi-quadratiques.	64
2.12	Polynômes de LEGENDRE bi-cubiques.	65
2.13	Base linéaire de polynômes de LAGRANGE.	66
2.14	Base de polynômes de LAGRANGE quadratique.	67
2.15	Scène du bloqueur	68
2.16	Scène du tube	69
2.17	Scène de la salle à manger (402 surfaces initiales)	69
2.18	Scène de la salle de cours (3153 surfaces initiales)	70
2.19	Résultat d'une simulation avec différentes valeurs de l'erreur globale	72
2.20	Erreur globale en fonction de l'erreur locale.	73
2.21	Comparaison visuelle de nos trois bases d'ondelettes	75
2.22	Effet de l'utilisation du Gouraud shading sur le résultat obtenu avec la base de HAAR	76
2.23	Temps de calcul (ordonnée en s) en fonction de l'erreur globale (abscisse).	77

2.24	Mémoire maximum utilisé (ordonnée en kB) en fonction de l'erreur globale (abscisse)	78
3.1	Salle de la géométrie	86
3.2	Supports normalisés pour une représentation fonctionnelle.	87
3.3	Illumination d'une scène contenant un tore troué.	88
3.4	Construction d'une base linéaire en 2D	91
3.5	Projection d'un morceau de tore avec une courbe dessinée dessus.	100
3.6	Exemples de parallélogrammes englobants.	102
3.7	Projection d'un parallélogramme sur un carré normalisé.	102
3.8	Étendre le fonction de radiosit� de fa�on continue.	103
3.9	Carte trap�zo�dale d'un arrangement de segments dans le plan.	105
3.10	Poids des points de quadrature peuvent �tre vus comme l'aire de leur zone d'influence	106
3.11	Surface auto-�mettrice.	114
3.12	Subdivisions hi�rarchiques de la maille virtuelle associ�e � la surface d�limit�e par la courbe de la figure 3.5(a).	114
3.13	Utilisation des surfaces planes complexes en architecture.	117
3.14	9 types de quadriques dans leur forme normalis�e	120
3.15	Exemple de projection � Jacobien constant.	122
3.16	Illumination d'une sc�ne compos�e de 22 surfaces dont 9 surfaces planes � bords complexes et 13 morceaux de cylindres.	124
3.17	Utilisation des surfaces planes � bords complexes.	127
3.18	Taux de convergence en fonction du temps de simulation pour la sc�ne de l'Op�ra.	129
3.19	Mod�le de la th�i�re.	130
3.20	R�sultats sur la sc�ne de la th�i�re.	132
3.21	Jeu d'�checs maill� (a) et mod�lis� � base de quadriques (b).	132
3.22	Temps de calcul (sec) pour les tests sur la th�i�re	134
3.23	Nombre de mailles finales pour les tests sur la th�i�re	134
4.1	Mod�le de l'Op�ra de la place Stanislas.	142
4.2	Tholos de Delphes.	143
4.3	La sc�ne du Reality Center de Cortailod.	144
4.4	L'�tage du Soda Hall de Berkeley (VRML).	145
4.5	Salle de la g�om�trie	146
4.6	Exemple de mobilier du Soda Hall remodelis� � base de quadriques.	148
4.7	Nombre de mailles initiales et finales dans 3 de nos sc�nes de test, avec et sans l'utilisation des polygones quelconques	149
4.8	Taux de convergence (�nergie non encore transmise divis�e par l'�nergie initiale) en fonction du temps de calcul (en s).	150
4.9	Nos simulations.	151
4.10	Simulations d'un �clairage naturel de la Tholos	152
4.11	Simulations du Reality Center de Cortailod	153
4.12	Simulation du 4 ^{�me} �tage du Soda Hall	154

4.13 Simulation dans le Soda Hall avec la salle de la géométrie.	155
14 Le maillage de discontinuité.	163

Liste des tableaux

3.1	Classification des surfaces quadriques.	119
3.2	Résultats numériques pour l'Opéra	129
3.3	Résultats numériques pour la théière	133
4.1	Effet de l'utilisation de polygones complexes sur le nombre de primitives initiales	147
4.2	Résultats numériques pour les calculs d'illumination de l'Opéra, et du 4 ^{ème} étage du Soda Hall.	149

Liste des algorithmes

1	Une itération de la méthode de JACOBI	29
2	Une itération de la méthode de GAUSS-SEIDEL	30
3	Une étape de la méthode de SOUTHWELL	30
4	Choix de la maille à subdiviser	56
5	Oracle de subdivision	57
6	Construction du parallélogramme englobant	101
7	Pseudo code de notre algorithme d'émission à partir d'une primitive com- plexe en utilisant la maille virtuelle	107
8	Algorithme de radiosité à base d'ondelettes avec maillage virtuel.	109

Introduction

« Toute la conduite de notre vie dépend de nos sens, entre lesquels celui de la vue étant le plus universel et le plus noble, il n'y a point de doute que les inventions qui servent à augmenter sa puissance ne soient des plus utiles qui puissent être. »

René DESCARTES, *Discours de la méthode*

Au cours des deux dernières décennies l'informatique graphique a connu un essor extraordinaire. La génération et l'utilisation d'images de synthèse sont sorties des laboratoires de recherche pour se retrouver dans des applications aussi diverses et variées que la télévision et le cinéma, l'imagerie médicale, la visualisation de données scientifiques, la conception assistée par ordinateur ou le prototypage virtuel.

L'un des défis de la synthèse d'images est de *créer à partir de la description numérique d'environnements 3D des images aussi réalistes que possible de ces environnements.*

Pour arriver à un tel résultat il faut pouvoir modéliser les données géométriques et photométriques servant à décrire les environnements dont on veut faire des images. Ensuite, il faut simuler la propagation de l'énergie lumineuse dans ces environnements en gérant correctement les données dont on dispose pour profiter au maximum de la qualité des informations qu'elles représentent. Ce processus est connu sous le nom d'*illumination globale.*

Jusqu'à présent l'objectif principal des simulations était d'obtenir des images très réalistes permettant d'apprécier des effets lumineux tels que les ombres portées, les inter-réflexions ou les réflexions spéculaires (effet de miroir). Mais aujourd'hui, les utilisateurs sont beaucoup plus exigeants. Tout d'abord, les environnements qu'ils modélisent sont de plus en plus complexes en terme de nombre de surfaces. Ensuite ces environnements intègrent des données issues de mesures physiques réelles de plus en plus précises. Cet ensemble représente une quantité toujours plus importante d'informations qui est, de ce fait, de plus en plus complexe à intégrer et à gérer lors des simulations. Enfin, la demande n'est plus de pouvoir générer des images figées mais d'obtenir un résultat indépendant du point de vue dans lequel on peut se déplacer de façon interactive et avec lequel on peut interagir pour modifier en temps réel les caractéristiques géométriques et photométriques.

Le nouvel enjeu est donc de *pouvoir augmenter la précision des modèles géométriques et physiques ainsi que celle des calculs de simulation tout en améliorant l'efficacité des méthodes utilisées pour pouvoir obtenir des résultats visuellement parfaits et physiquement corrects en un minimum de temps.*

C'est à cet enjeu que nous nous sommes intéressés au cours de nos recherches et ce sont les solutions que nous avons développées pour y arriver que nous exposons dans mémoire.

État de l'art

Quand on considère l'histoire de la synthèse d'images, on constate qu'une grande partie des efforts de recherche dans ce domaine a eu pour but d'augmenter le photo-réalisme des images générées, c'est-à-dire d'obtenir des images de plus en plus semblables à des photos.

Or, la seule solution efficace pour obtenir des images photo-réalistes est de simuler la propagation de l'énergie lumineuse entre les objets qui composent la scène que l'on veut visualiser. En effet, cette simulation permet d'obtenir des images intégrant tous les effets lumineux qui donnent aux objets leur couleur et leur aspect naturel. Pour faire cela deux grands modèles permettant de décrire les réflexions lumineuses ont été proposés : le premier s'appuie sur une représentation locale des réflexions de la lumière alors que le deuxième se base sur un modèle global.

Le modèle local (ou direct) ne considère que les interactions entre les objets et les sources de lumière. Dans ce modèle, les objets ne sont vus que sous l'éclairage des sources lumineuses directes et aucune réflexion entre les objets de la scène ne peut être prise en compte. Beaucoup de recherches ont été faites pour améliorer le modèle local notamment pour l'élimination des faces cachées [Fol 90, Sut 74] avec, par exemple, le Z-Buffer [Cat 75] et pour la simulation de la réflexion de la lumière [Fol 90, Gou 71] avec l'interpolation de PHONG [Pho 75].

Le modèle global prend en compte les réflexions successives de la lumière. Il s'appuie sur une représentation théorique de la façon dont la lumière se réfléchit sur un objet pour se propager jusqu'à un autre objet. Dans ce modèle, la valeur de l'énergie lumineuse qui se trouve en un point d'un objet est donc le résultat de l'éclairage dû aux sources lumineuses (illumination directe) et aux réflexions de la lumière sur les autres objets de la scène (illumination indirecte). Le comportement global de la lumière tel qu'on sait le modéliser aujourd'hui est retranscrit en termes mathématiques par l'équation de radiance [Kaj 86]. Une fois l'équation de radiance définie, il faut pouvoir lui trouver une solution. Aux méthodes *déterministes* s'opposent celles dites *stochastiques*, basées historiquement sur des phénomènes probabilistes. Ces dernières, qui se fondent sur des approches à base de lancé de rayons ou de particules, permettent depuis peu d'obtenir des résultats indépendants du point de vue. Notre intérêt porte sur les méthodes déterministes par éléments finis et nous n'avons pas fait de comparaison avec les méthodes stochastiques.

La méthode déterministe que nous avons étudiée est la *radiosité* [Gor 84, Coh 85, Nis 85]. Elle est basée sur la résolution de l'équation de radiosité qui s'obtient à partir de l'équation de radiance en considérant que toutes les surfaces sont idéalement diffuses. On obtient par ce biais des solutions indépendantes du point de vue et intégrant de façon naturelle les inter-réflexions. Ces solutions ont l'avantage d'être extrêmement précises et surtout de donner des valeurs d'illumination physiquement proches de la réalité. Les méthodes utilisées pour résoudre l'équation de radiosité sont fondées sur des approximations qui permettent de rendre le problème calculable et de le réduire à une

taille compatible avec les ressources matérielles existantes aujourd'hui. Dans les approches développées à ce jour, on distingue les méthodes par échantillonnage de points (Monte-Carlo [Shi 94, Gla 95, Sta 99]) et les méthodes par éléments finis (approches variationnelles [Hec 91]). Ces méthodes sont détaillées dans les sections suivantes.

Une autre approche est de résoudre directement l'équation de radiance. Des solutions ont été proposées par Larry AUPPERLE et Pat HANRAHAN dans [Aup 93] et par Per Henrik CHRISTENSEN *et al.* dans [Chr 94, Chr 96] et nous avons aussi essayé d'appliquer nos algorithmes à cette résolution. Malheureusement, la complexité de l'équation fait que ces solutions ne s'appliquent qu'à des scènes de très petite taille et ne sont pas utilisables pour des applications du monde réel. Afin d'intégrer les effets dus aux réflexions spéculaires dans des solutions de radiosité des approches « Multi-Pass » ont été proposées par John R. WALLACE *et al.* dans [Wal 87], puis par François SILLION et Claude PUECH dans [Sil 89] et par Shenchang Eric CHEN *et al.* dans [Che 91]. Ces solutions permettent de bénéficier de l'efficacité de la méthode de radiosité tout en conservant la précision intrinsèque de l'équation de radiance mais les calculs restent longs et complexes et n'ont pas donné lieu à des implantations¹ industrielles.

Motivation

Seules les méthodes de radiosité permettent de générer des images répondant à l'enjeu que nous nous sommes fixé : des images réalistes et indépendantes du point de vue.

Cependant, malgré l'hypothèse simplificatrice des environnements idéalement diffus, la résolution de l'algorithme de radiosité est encore très complexe. Elle demande énormément de ressources en temps de calcul et en mémoire.

Pour que l'on puisse un jour faire des simulations fournissant des images photo-réalistes d'environnements issus de la vie courante (environnements très grands et physiquement très complexes) à partir de simulations de radiosité, plusieurs points restent à améliorer :

- **Modélisation** : les hypothèses physiques restent encore trop restrictives. La description simpliste de la géométrie au moyen de triangles et de polygones ne permet pas toujours de prendre en compte toutes les données topologiques des environnements à simuler. Les caractéristiques photométriques sont difficiles à décrire précisément et la représentation classique à base de couleurs RGB et de distribution spatiale constante ne suffit généralement pas pour simuler les effets complexes de la lumière. Il faut donc inventer et utiliser de nouveaux formats de données pour pouvoir mieux représenter les caractéristiques géométriques et photométriques des environnements.
- **Mathématique et théorique** : les méthodes variationnelles de résolution s'appuient sur des approximations à différents niveaux : la discrétisation de l'espace de projection, les fonctionnelles linéaires, le calcul des coefficients du système linéaire, la méthode itérative de résolution. Toutes ces approximations introduisent des erreurs dans les résultats. Ces erreurs, qui s'accumulent, peuvent créer des artefacts importants dans les images. Pour diminuer l'impact de ces approximations sur la

1. Nous utilisons le verbe implanter pour traduire le mot anglais « implement » qui signifie transcrire un algorithme dans un programme informatique.

qualité du résultat il faut augmenter la précision mathématique en se rapprochant des solutions théoriques exactes.

- **Algorithmique** : le coût des algorithmes de calcul reste très élevé, même en acceptant un taux d'erreur modéré. Ce coût est tel qu'il est souvent impossible de bien prendre en compte les informations fournies par les données initiales et de simuler avec une grande précision la propagation de la lumière. Il est donc primordial d'améliorer l'efficacité des algorithmes pour pouvoir les pousser plus loin et atteindre une plus grande précision.

Radiosité dans l'équipe ISA

Depuis plusieurs années, les chercheurs de l'équipe ISA du LORIA développent un algorithme de radiosité basé sur une résolution par éléments finis de l'équation de radiosité [Cha 97, Mer 97, Win 98]. Les recherches menées au sein de l'équipe avaient comme objectif principal de trouver des solutions aux deux problèmes suivants :

- L'amélioration des structures informatiques et algorithmiques existantes et le développement de nouvelles structures pour pouvoir mieux gérer la géométrie et les caractéristiques photométriques des scènes à simuler.
- L'étude de nouvelles solutions pour réduire la complexité de l'algorithme afin d'améliorer l'efficacité aussi bien au niveau du temps de calcul que de la place mémoire utilisée.

Toutes ces recherches ont été faites dans le but précis de pouvoir générer de plus en plus efficacement des images de plus en plus réalistes. Elles ont conduit les chercheurs de l'équipe à considérer des solutions générales basées sur des théories mathématiques. Ils ont ainsi trouvé des cadres théoriques aux solutions empiriques, ce qui leur a permis d'aborder efficacement les cas les plus compliqués que l'on retrouve dans les simulations de scènes venues d'applications industrielles.

Les points particuliers qui ont été abordés et traités par les chercheurs de l'équipe ISA sont les suivants :

- **La prise en compte des caractéristiques physiques correctes** : pour arriver à fournir un résultat réaliste, il faut commencer par considérer des données physiques exactes pour décrire l'environnement. Pascal DEVILLE a travaillé durant sa thèse [Dev 96] sur la description des sources de lumière et des spectres de lumière pour définir les couleurs et les matériaux des surfaces qui composent les scènes. Son travail à abouti à l'implantation d'un format de données permettant de gérer :
 - la totalité des sources de lumières ponctuelles que l'on retrouve dans les modèles industriels au moyen de la représentation C/γ [Caz 94, Dev 95],
 - toutes les descriptions spectrales des couleurs pour les matériaux au moyen de bases spectrales connues : MEYER, RGB, HVS, etc., ou par défaut au moyen de bases constantes ou linéaires par morceaux [Pau 95].
- **Des optimisations d'algorithmes pour résoudre l'équation de radiosité au moyen de méthodes par éléments finis** : les algorithmes de résolution par éléments finis sont issus de l'analyse numérique. Ils donnent des outils très puissants et

très généraux pour résoudre des équations intégrales telles que l'équation de radiosité. Pourtant, leur utilisation en radiosité s'est longtemps limitée à une implantation très simpliste et très limitative notamment pour la gestion des ressources mémoire. Slimane MERZOUK et Christophe WINKLER ont fait des études très poussées de ces algorithmes durant leur thèse [Mer 97, Win 98]. Ils ont abouti à la définition et à l'implantation d'un algorithme extrêmement efficace basé sur la méthode de résolution de SOUTHWELL mais qui effectue les calculs d'interaction à la volée sans les conserver en mémoire.

- **Une parallélisation de l'algorithme de radiosité**: récemment Xavier CAVIN a introduit une parallélisation de l'algorithme de radiosité dans la plate-forme logicielle de l'équipe. L'algorithme parallèle qu'il a développé tire profit de l'architecture CC-NUMA (Cache Coherent, Non Uniform Memory Access) des machines graphiques haute performance de la société SGI. Grâce à sa gestion de la mémoire partagée, cet algorithme est très efficace même dans des environnements dynamiques et non-uniformes. Cette implantation permet de faire des simulations de radiosité de très grande qualité sur des scènes très complexes [Cav 98].
- **Utilisation des cartes graphiques hautes performances pour accélérer les calculs de visibilité**: Laurent ALONSO a travaillé sur l'amélioration des algorithmes permettant de calculer la fonction de visibilité. Son travail a abouti à l'élaboration d'un algorithme hybride utilisant à la fois une structure spatiale accélératrice très performante: le BSP (Binary Space Partition) de Marc DE BERG [dB 92, dB 94, dB 95] et les cartes graphiques hautes performances [Alo 00b]. Cet algorithme a permis de gagner un facteur d'échelle dans la taille des scènes que l'on peut simuler.

Tous ces travaux ont abouti à la plate-forme logicielle Candela spécialement dédiée à la résolution de l'équation de radiosité. Candela sert à la fois de plate-forme de recherche pour tester de nouveaux algorithmes et de plate-forme logicielle efficace pour effectuer des simulations de très grande qualité sur des modèles très complexes.

Le travail présenté dans cette thèse a été effectué dans le but de poursuivre ces recherches. Deux objectifs principaux ont été abordés :

- Avoir une meilleure connaissance des *bases fonctionnelles* classiques servant de support aux méthodes par éléments finis et en développer de nouvelles plus efficaces.
- Pouvoir intégrer des données géométriques encore plus complexes en bénéficiant de la richesse et de l'aspect compact de la *représentation mathématique* des objets géométriques.

Solutions proposées

L'équation de radiance fait le lien entre la théorie physique et la simulation informatique de la propagation de la lumière. Elle nous permet, grâce à la théorie corpusculaire de la lumière, d'avoir une représentation numériquement calculable par approximation. Si on sait résoudre cette équation on peut donc obtenir des résultats d'illumination en accord avec la réalité physique (pour autant qu'on sache la modéliser).

D'un autre côté, les mathématiques nous fournissent des outils puissants de représentation, aussi bien fonctionnelle que géométrique, pouvant nous servir à limiter les approximations faites lors de la résolution de l'équation de radiance. Elles sont donc indispensables pour résoudre de façon efficace et précise une telle équation.

Faire le lien entre ces deux représentations n'est pas chose aisée. Il faut en effet choisir parmi les innombrables outils développés par les mathématiciens, ceux qui s'appliquent efficacement au problème donné. Il faut ensuite appliquer correctement ces outils pour bénéficier pleinement de toute leur puissance sans pour autant utiliser des représentations trop complexes qui deviendraient difficiles à gérer dans des simulations numériques.

Beaucoup de recherches en radiosit  ont port  sur l' tude et l'utilisation de ces solutions math matiques notamment avec :

- l'*analyse fonctionnelle* pour d montrer l'existence et l'unicit  d'une solution   l' quation de radiance [Arv 95, Win 98],
- la *formalisation variationnelle* donnant un cadre th orique aux m thodes de r solution par  l ments finis [Mer 97],
- les *m thodes de r solution it ratives* [Coh 93, Sil 94b] pour rendre le probl me calculable avec les moyens dont nous disposons aujourd'hui,
- la *repr sentation hi rarchique* et le *clustering* [Han 91, Smi 94, Sil 95a] pour diminuer la complexit  de l'algorithme en regroupant les interactions de m me importance,
- les *fonctions d'ordre  lev * et les ondelettes [Zat 93, Tro 93, Sch 93, Gor 93] pour avoir une repr sentation fonctionnelle plus compacte de la fonction de radiosit ,
- le *complexe de visibilit * [Ort 96a, Ort 96b, Dur 99a, Dur 99b] pour conna tre les relations de visibilit  entre les points de la sc ne,

et bien d'autres encore.

Cependant, les algorithmes existants utilisent rarement ces solutions   leur maximum. Par exemple, l'algorithme de radiosit    base d'ondelettes le plus utilis  est l'algorithme classique de radiosit  hi rarchique propos  en 1991 par Pat HANRAHAN *et al.* [Han 91] : cet algorithme n'utilise pas les fonctions d'ordre  lev .

Le travail que nous pr sentons dans cette th se vise justement   revenir aux outils math matiques pour les utiliser   leur maximum. Nous avons appliqu  ce principe dans deux cas bien pr cis : les fonctions ondelettes et les repr sentations g om triques.

Fonctions ondelettes

Les bases de fonctions ondelettes sont des bases de fonctions hi rarchiques qui ont  t  introduites en radiosit  en 1993 par Peter SCHRODER [Sch 94a] et Steven GORTLER [Gor 93]. La th orie sous-jacente permet de faire le lien entre la repr sentation hi rarchique de la fonction de radiosit  et les bases de fonctions d'ordre  lev . Malheureusement, leur utilisation en radiosit  est rest e tr s th orique et les seuls r sultats exp rimentaux qui ont  t  publi s  taient plut t d favorables pour ces bases (voir l' tude [Wil 97b, Wil 97a] men e par Andrew J. WILLMOTT et Paul S. HECKBERT en 1997).

Forts de la puissance mathématique théorique des ondelettes, nous avons faits l'hypothèse que, grâce à une conception de l'algorithme de radiosit  adapt e   la repr sentation fonctionnelle, l'utilisation des fonctions ondelettes d'ordre  lev  devaient apporter une r elle am lioration en temps de calcul et en pr cision sur les r sultats exp rimentaux. Pour v rifier cela, nous avons commenc  par effectuer un grand nombre de tests dans des situations tr s vari es. Ces tests nous ont permis de mieux comprendre les besoins algorithmiques associ s   l'utilisation des fonctions d'ordre  lev . Nous avons ainsi pu d terminer de nouvelles bases d'ondelettes plus efficaces et mieux adapt es   la r solution de l' quation de radiosit . Nous avons implant  ces ondelettes dans notre plate-forme logicielle et nous avons modifi  cette derni re pour qu'elle profite mieux du pouvoir d'approximation des fonctions ondelettes d'ordre  lev . Ces modifications ont touch es plusieurs points clefs de l'algorithme, tels que l'oracle de subdivision, le calcul de la fonction de visibilit , le stockage des liens en m moire ou encore l'algorithme de r solution.

Cette  tude, montre qu'avec notre implantation les fonctions ondelettes d'ordre  lev  apportent un r el gain par rapport   l'approche hi rarchique classique et par rapport aux fonctions d'ordre moins  lev . De plus, il existe des zones de comp tences pour chaque base d'ondelettes. Dans ces zones, la base concern e est plus efficace   tous points de vue que les autres bases. Enfin, plus la zone de comp tence correspond   des simulations de grande pr cision, plus l'ordre des fonctions de la base concern e est  lev .

Repr sentations g om triques

Une hypoth se de base de l'algorithme de radiosit  original est que les primitives g om triques composant les sc nes   illuminer sont des surfaces planes simples [Gor 84]. Malheureusement, aucune des d couvertes et am liorations dans le domaine n'a r ellement permis de s'affranchir totalement de cette hypoth se. Par exemple, dans les algorithmes de radiosit  hi rarchique les mailles finales sont toujours des triangles ou des parall logrammes. Le calcul de radiosit  par la m thode de GALERKIN pr sent e par Harold ZATZ dans [Zat 93] fait exception   la r gle en proposant une solution totalement ind pendante de la g om trie initiale gr ce   des calculs d'interaction bas s sur des approximations par quadratures. Cependant, les implantations de cette m thode dans des algorithmes de radiosit  hi rarchique se sont toujours limit es   l'utilisation de primitives g om triques planes. Les algorithmes de radiosit    base d'ondelettes [Gor 93, Sch 94a] font partie de cette cat gorie et n'ont jamais donn  lieu   des implantations ind pendantes de la g om trie.

  l'heure actuelle, tous les algorithmes permettant de simuler l'illumination globale dans des environnements diffus sur des primitives g om triques courbes utilisent d'une fa on ou d'une autre une approximation g om trique pour g rer ces primitives. Ainsi, la fonction de radiosit  est  valu e au prix d'une certaine erreur et les donn es g om triques ne sont jamais utilis es de fa on exacte pour effectuer les calculs. Par cons quent, avec aucune des solutions actuelles, l'int gration de primitives courbes dans un algorithme de radiosit  n'est apparue comme une extension logique et naturelle de la radiosit  sur des surfaces polygonales.

Afin de revenir à la description originale des surfaces, nous nous sommes intéressés à la possibilité d'utiliser directement des surfaces paramétriques dans l'algorithme de radiosité. Nous avons ainsi développé une méthode très générale pour effectuer le calcul de radiosité dans des scènes constituées d'une grande variété de surfaces paramétriques (incluant les surfaces planes avec des bords quelconques et les surfaces paramétriques courbes de degré peu élevé). Contrairement aux approches précédentes qui nécessitaient une triangulation des surfaces en entrée, notre méthode bénéficie de la richesse et de l'aspect compact de la représentation mathématique des objets géométriques. Au coeur de notre méthode se trouve le *Maillage Virtuel*, une abstraction de la géométrie initiale permettant d'appliquer les algorithmes classiques d'illumination à des surfaces complexes exactement comme si ces surfaces étaient de simples carrés. Le maillage virtuel est constitué d'un ensemble de domaines normalisés sur lesquels les surfaces initiales sont projetées grâce à une fonction de projection qui préserve leurs propriétés énergétiques. Les valeurs de radiosité sont alors calculées sur ces domaines normalisés avant d'être re-projetées sur les surfaces originales. Ainsi, notre approche dissocie la géométrie à proprement parler des supports de fonctions qu'elle représente.

Le maillage virtuel permet d'augmenter la précision et l'efficacité de l'algorithme en profitant de représentations mathématiques compactes. En effet, on augmente la complexité des fonctions permettant de décrire la géométrie pour intégrer beaucoup plus d'informations avec beaucoup moins de données. On simplifie, ainsi, les modèles en terme de nombre de surfaces tout en augmentant la précision de la description géométrique. L'utilisation des surfaces paramétriques directement dans le calcul de la fonction de radiosité nous permet de revenir à la description originale des surfaces et ainsi d'accéder à toutes les données topologiques intéressantes sans faire aucune approximation.

Plan du mémoire

L'organisation de ce mémoire est la suivante. Le problème posé par l'illumination globale d'une scène doit être modélisé pour pouvoir simuler les échanges radiatifs. Nous présentons en premier lieu les aspects physiques, mathématiques et algorithmiques qui permettent respectivement d'établir l'équation de radiance dans le cas général et en particulier celle de radiosité, de proposer un formalisme pour la résolution de ces équations et finalement de détailler les implications algorithmiques qui en découlent. Ces trois étapes sont détaillées dans le chapitre 1 (p. 11).

Dans le chapitre 2 (p. 41) nous présentons notre implantation de l'algorithme de radiosité à base d'ondelettes. Nous introduisons les nouvelles bases d'ondelettes que nous avons développées et nous montrons qu'avec cette implantation les fonctions ondelettes d'ordre élevé apportent un réel gain par rapport à l'approche hiérarchique classique et par rapport aux fonctions d'ordre moins élevé.

Dans le chapitre 3 (p. 83) nous présentons l'algorithme de radiosité avec le maillage virtuel. Nous décrivons comment nous construisons et comment nous utilisons le maillage virtuel. Nous précisons les détails d'implantation du maillage virtuel pour deux exemples de surfaces : les surfaces planes à bords quelconques et les surfaces quadriques (i.e. les surfaces paramétriques de degré 2). Finalement nous donnons des résultats qualitatifs et

quantitatifs pour montrer que le maillage virtuel permet d'obtenir des gains importants en précision et en efficacité.

Enfin, dans le chapitre 4 (p. 137) nous montrons à quel point l'utilisation générique de ce deux formalismes mathématiques représente un réel progrès pour la simulation de l'illumination globale sur des scènes issues d'applications du monde réel. Nous montrons, notamment, que les fonctions ondelettes d'ordre élevé associées au maillage virtuel permettent de faire des simulations sur des scènes industrielles.

Nous finissons notre mémoire par une conclusion qui nous permettra de récapituler les différents formalismes mathématiques que nous avons introduits. Nous donnons aussi des voies de recherches futures pour utiliser plus efficacement ces formalismes ou pour étendre leur utilisation à d'autres algorithmes.

Chapitre 1

Équation de radiance

« ... j'appelle images d'abord les ombres, ensuite les reliefs que l'on voit dans les eaux, ou à la surface des corps opaques, polis et brillants, et toutes les représentations semblables; ... »

PLATON, *République*, VI

Sommaire

1.1	Introduction	11
1.2	Modélisation physique	13
1.2.1	Aspects physiques	13
1.2.2	Expression du transport	14
1.2.3	Analyse de l'équation de radiance	16
1.2.4	Portée des hypothèses	19
1.3	Résolution mathématique	22
1.3.1	Analyse fonctionnelle	22
1.3.2	Approche variationnelle	25
1.3.3	Résolution du système linéaire	27
1.4	Aspects algorithmiques	31
1.4.1	Ensemble \mathcal{M}	31
1.4.2	Fonctions de base (ϕ_i)	33

1.1 Introduction

Le domaine de l'informatique graphique auquel nous nous intéressons est celui de la synthèse d'images photo-réalistes. Son but est de générer des images d'environnements virtuels si réalistes qu'il est très difficile de dire si ce sont des images de synthèse ou des photos de la réalité. Pour arriver à un tel résultat, nous utilisons des algorithmes qui simulent les phénomènes lumineux à l'intérieur des environnements virtuels. Le réalisme découle alors de la qualité avec laquelle on arrive à rendre compte de ces phénomènes en se basant sur des modèles physiques de la réalité et de la précision des simulations.

Le processus de création d'images de synthèses peut être décomposé en trois étapes. Une étape de modélisation et de formalisation mathématique des données du problème. Une étape de résolution mathématique, c'est-à-dire, une étape durant laquelle on s'attache à décrire les processus mathématiques qui permettent de discrétiser le problème afin de le rendre calculable. Et enfin, une dernière étape de résolution algorithmique qui consiste à trouver les meilleurs algorithmes pour résoudre numériquement le problème.

La première étape suppose que l'on commence par étudier ce que l'on veut faire pour avoir une meilleure appréhension de l'environnement et pouvoir en faire une modélisation précise. Cette modélisation doit fournir une représentation mathématique des lois physiques qui régissent le comportement de la lumière. Ces lois ne sont pas uniques, elles dépendent du point de vue que l'on adopte et des hypothèses que l'on fait pour en simplifier la représentation. Le modèle que l'on crée dépend donc du but que l'on s'est fixé et il est important de savoir précisément ce que l'on souhaite obtenir pour ne pas prendre en compte des informations dont l'impact sur le résultat final est minime ou ne correspond pas à ce que l'on attend.

Nous avons choisi d'utiliser l'équation de radiance pour modéliser les phénomènes qui conduisent à la simulation de l'illumination globale. Cette équation décrit la propagation de la lumière dans le vide. Elle se place d'un point de vue macroscopique et ne prend pas en compte les phénomènes de diffraction, de phosphorescence et de fluorescence.

La deuxième étape a pour but de trouver une solution mathématique à l'équation issue de la modélisation physique. Bien que les hypothèses de modélisation soient relativement restrictives, l'équation de radiance n'a généralement pas de solution analytique utilisable. Il est alors nécessaire de la reformuler en utilisant un formalisme mathématique adéquat. Ce formalisme doit conduire à l'établissement de méthodes permettant de faire une approximation numérique satisfaisante de la solution.

La fonction solution fait partie d'un espace fonctionnel de dimension infinie. L'analyse fonctionnelle constitue une approche naturelle pour résoudre ce système d'équations. Elle formalise l'étude des propriétés algébriques et topologiques d'espaces abstraits, et en particulier des espaces fonctionnels de dimension infinie. Il en découle la reformulation de l'équation de radiance sous forme d'opérateurs, qui ne sont autres que des fonctions d'un espace fonctionnel dans un autre. Ces opérateurs permettent de poser le problème sous une forme différente, appelée *formulation variationnelle*. Nous avons choisi de résoudre le problème par la méthode de l'*approximation variationnelle*, encore nommée *méthode de GALERKIN* ou *méthode de Ritz*. Une autre solution aurait été de lui préférer la méthode de *collocation de points*. Pour aboutir à un système d'équations résoluble, il suffit ensuite de définir les fonctions de base utilisées pour faire une approximation de la fonction de radiance.

Le système linéaire ainsi formé est extrêmement grand et ses coefficients sont très compliqués à évaluer, de plus, il n'existe généralement pas de solution pour le résoudre de façon exacte. La dernière étape consiste donc à trouver et à implanter les algorithmes qui vont permettre de faire une approximation aussi précise que possible dans l'espace fonctionnel que l'on a choisi.

L'implantation de ces algorithmes implique une maîtrise complète de la représentation numérique du modèle. Il faut donc définir parfaitement le cadre des représentations géométrique, photométrique et fonctionnelle nécessaires aux calculs. Une fois que ces représentations sont connues et délimitées, on peut s'intéresser aux algorithmes de résolution à proprement parler ; c'est-à-dire aux méthodes numériques de calcul et d'approximation et aux processus itératifs de résolution. La description de ces points peut être trouvée dans la thèse de Christophe WINKLER [Win 98].

Le plan de ce chapitre suit cette décomposition en trois étapes. Nous commençons dans une première section 1.2 (p. 13) par définir le cadre physique dans lequel on se place pour modéliser les environnements et les propriétés de la lumière. Puis nous exposons dans une deuxième section 1.3 (p. 22) les aspects mathématiques relatifs à la résolution de l'équation de transport issue de la modélisation. Enfin, nous terminons dans une troisième section 1.4 (p. 31) par approfondir la présentation des choix algorithmiques qui se rapportent à notre travail de recherche. Nous précisons notamment les choix relatifs à l'implantation des données fonctionnelles et géométriques.

1.2 Modélisation physique

Le but de la modélisation est d'avoir une représentation mathématique compacte et efficace des phénomènes physiques qui interviennent dans le processus d'illumination globale.

Dans cette section, nous définissons les données physiques sur lesquelles nous nous basons pour modéliser la propagation de la lumière et pour faire nos simulations. Nous soulignons, en effet, les propriétés importantes pour les applications que nous voulons faire en synthèse d'images et nous précisons le modèle d'observation physique dans lequel nous nous plaçons. Ce cadre général nous permettra d'établir l'équation du transport de la lumière dans un environnement sans milieu participatif.

Nous étudions cette équation en passant en revue les différents termes qui la composent. Enfin, nous terminons cette étude physique en rappelant toutes les hypothèses que notre modélisation implique et en présentant les autres hypothèses que nous faisons dans nos simulations.

1.2.1 Aspects physiques

La simulation des phénomènes lumineux requiert une connaissance des propriétés de la lumière, étudiées par l'optique. Deux approches sont alors possibles : la *théorie particulaire* et la *théorie ondulatoire*. Ces deux points de vue sont duaux, puisque ce double aspect de la lumière n'est qu'un cas particulier d'une propriété générale de la matière : l'association d'une onde à toute particule.

Suivant le choix du modèle d'observation, qu'il soit microscopique ou macroscopique, l'étude optique est divisée en trois domaines. Tout d'abord, l'*optique géométrique* prend en compte les propriétés macroscopiques de la lumière lors de sa propagation dans un environnement. Cette approximation, basée sur le *rayon lumineux*, permet de rendre compte des

ombres ainsi que des lois de réflexion et de réfraction. Cette approche est particulièrement appropriée pour la synthèse d'images.

Pourtant, elle est incapable de modéliser certains effets lors de l'interaction de la lumière avec des objets de petite taille ayant comme ordre de grandeur la longueur d'onde. Par contre, l'*optique ondulatoire* permet de modéliser ces effets. Ils peuvent survenir lors de l'interaction de la lumière avec les micro-facettes des surfaces. La diffusion, c'est-à-dire l'influence des impuretés du milieu participatif sur le faisceau lumineux, fait aussi partie de ces effets. Il faudrait donc se placer dans le cadre de l'optique ondulatoire pour simuler les effets de diffusion, de diffraction ou d'interférence. Le dernier modèle d'observation fait appel à la *théorie quantique* pour expliquer le résultat de l'interaction de la lumière avec des atomes ou des molécules.

L'optique géométrique convient donc parfaitement pour décrire la propagation de la lumière entre les surfaces. Néanmoins, l'utilisation de l'optique ondulatoire s'avère nécessaire pour simuler certains effets lors de l'interaction de la lumière avec les surfaces.

Toute simulation de la propagation de la lumière s'appuie forcément sur l'expression quantitative de transferts d'énergie lumineuse. Puis, une fois une solution définie, il faut prendre en compte la sensibilité physiologique de l'observateur pour la représenter.

La *radiométrie* est la science des mesures de l'énergie électromagnétique. Ainsi, des valeurs radiométriques sont utilisées pour la résolution de l'équation de radiance, puisque cette dernière se base sur le principe de la conservation de l'énergie. Ces grandeurs, comme le *flux énergétique*, peuvent être mesurées à l'aide de radiomètres.

D'un autre côté, la *photométrie* est la science des mesures de l'énergie lumineuse. Celle-ci traduit la perception physiologique d'un spectre électromagnétique par l'homme. En effet, l'œil est un organe sélectif qui perçoit mieux certaines longueurs d'onde que d'autres. Ainsi, une solution radiométrique doit être transformée en grandeur photométrique lors de la visualisation.

La relation entre une grandeur radiométrique f_r et photométrique f_p est donnée par un facteur dépendant de la longueur d'onde :

$$f_p = \int_{\lambda} f_r(\lambda) \mathcal{V}(\lambda) d\lambda.$$

La représentation de ce facteur en fonction de la longueur d'onde forme la courbe relative d'*efficacité lumineuse*. Pour répondre à la différence physiologique entre les perceptions diurne et nocturne, deux courbes $\mathcal{V}(\lambda)$ et $\mathcal{V}'(\lambda)$ ont été normalisées (voir figure 1.1). Ainsi, pour une vision nocturne, l'œil est plus sensible au bleu qu'au rouge. La courbe d'efficacité lumineuse adaptée doit donc être prise en compte lors de la visualisation du résultat. Notons que ces courbes n'existent pas seulement pour caractériser la réponse de l'œil humain à la lumière, mais également celle des pellicules photographiques. Il faut donc faire intervenir une courbe adéquate pour comparer le résultat à une photographie.

1.2.2 Expression du transport

Le but de la simulation des phénomènes lumineux est de modéliser la propagation de la lumière, ou plus précisément de l'énergie lumineuse, dans un certain environnement.

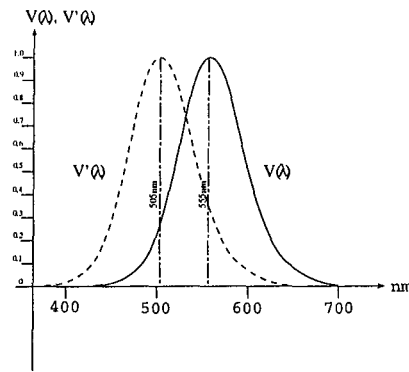


FIG. 1.1 – Efficacités lumineuses relatives pour la vision diurne et nocturne.

Ce processus de transfert d'énergie entre dans le cadre plus général des phénomènes de transport. Ceux-ci, pour un système fermé, s'expriment sous la forme d'une égalité qui traduit la conservation de l'énergie :

$$\text{entrée} + \text{création} = \text{sortie} + \text{accumulation}. \quad (1.1)$$

Parmi les trois différents processus généraux de transport d'énergie, nous nous intéressons uniquement à la *radiation*. En effet, elle représente une énergie transportée par des champs électromagnétiques, or l'énergie lumineuse est électromagnétique par essence. Les deux processus de transports que sont la *conduction* et la *convection* n'interviennent pas ici. De plus, nous n'étudions pas la dépendance temporelle de l'équation (1.1), pour deux raisons : la vitesse de la lumière est telle que l'éclairage d'une scène de taille humaine est instantanée, sans période transitoire visible ; en outre, nous ne prendrons pas en compte des matériaux différant la ré-émission de la lumière comme le font les objets phosphorescents. Cela écarte le terme d'accumulation de l'équation (1.1).

Nous nous plaçons dans le formalisme de l'optique géométrique pour décrire les transferts à grande échelle, et de l'optique ondulatoire pour modéliser les réflexions. Ainsi, les effets de diffusion et donc d'atténuation ne sont pas pris en compte, ce qui revient à étudier la propagation de la lumière dans le vide. Les réfractions ne sont pas abordées dans ce mémoire, mais peuvent être facilement extrapolées des réflexions. Comme l'ont fait James ARVO *et al.* dans [Arv 94], nous négligeons certains phénomènes comme les transferts d'énergie entre différentes bandes spectrales, ainsi que les effets de polarisation. Ces phénomènes interviennent essentiellement lors de la réflexion de la lumière sur des surfaces.

Dans le cadre de ces hypothèses, les processus physiques intervenant lors de la simulation des phénomènes lumineux sont régis par l'équation de radiance :

$$\underbrace{f(\lambda, x, \omega^\dagger)}_{\text{sortie}} = \underbrace{g(\lambda, x, \omega^\dagger)}_{\text{création}} + \int_{\Omega} \underbrace{f(\lambda, y, \omega^\dagger)}_{\text{entrée}} k(\lambda, x, \omega^\dagger \rightarrow \omega^\dagger) d\mu. \quad (1.2)$$

La résolution analytique de cette équation de FREDHOLM du second type reste un problème ouvert. Il faut donc faire appel à des méthodes d'approximation pour évaluer une solution de cette équation.

Mais commençons par l'analyser, pour faire ressortir les concepts physiques sous-jacents aux différents termes de cette équation (1.2).

1.2.3 Analyse de l'équation de radiance

La plupart des termes qui apparaissent dans l'équation de radiance (1.2) ont une portée *locale*. Ils n'ont pas besoin de la connaissance de l'ensemble de la scène pour être évalués, bien qu'ils influencent sur le résultat. Dans le cas contraire, on dit que la portée est *globale*.

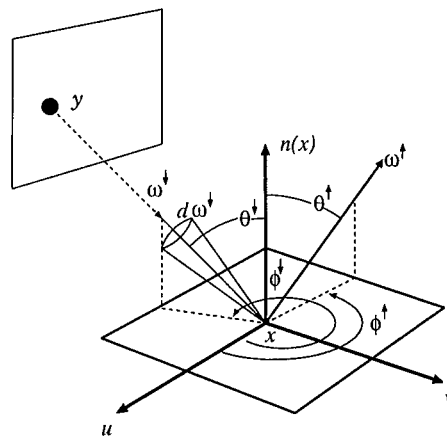


FIG. 1.2 – Schéma pour l'équation de radiance.

1.2.3.1 Notations générales

- λ , longueur d'onde restreinte à l'intervalle Λ ($\Lambda \subset \mathbb{R}^+$)², traduit la dépendance spectrale. Par exemple, les surfaces ont des propriétés physiques telles qu'elles réagissent différemment selon la longueur d'onde ;
- $x \equiv (x_1, x_2, x_3)$ caractérise la dépendance spatiale. Les points sont situés sur des surfaces de la scène, et peuvent s'exprimer de façon paramétrique pour chacune des surfaces $S_i : x = \psi_i(u, v)$;
- $\omega \equiv (\theta, \psi)$, ω^\downarrow et ω^\uparrow traduisent la dépendance directionnelle générale, incidente ou réfléchie ;
- Ω et $d\mu$ représentent respectivement l'hémisphère des directions incidentes centrée sur le point x , et la mesure associée définie par : $d\mu = \cos \theta^\downarrow d\omega^\downarrow$;
- \mathcal{S} est la sphère unité ;

2. Pratiquement, nous nous limitons au spectre visible [380 nm, 780 nm]. Cependant, ce n'est pas une obligation : un autre intervalle de longueurs d'onde peut être considéré, pourvu que les hypothèses de l'optique géométrique restent valides dans ce domaine.

- \mathcal{M} désigne la géométrie de l'environnement. Il est généralement décrit comme la réunion de surfaces, qui peuvent être théoriquement de nature quelconque : des surfaces planes, des quadriques, etc.

1.2.3.2 Fonction f

Dans l'équation (1.2), la fonction f désigne l'inconnue à définir, la fonction *radiance* aussi nommée *champ radiance* ou *distribution de radiance*. Elle caractérise, en un point donné, l'énergie lumineuse se propageant dans une direction donnée. Le but de la synthèse d'images, et donc du système informatique de simulation, est de faire une approximation de cette fonction de façon la plus précise possible. C'est une caractéristique locale de la scène, même si la résolution demande de prendre en compte la globalité de la scène.

$$f : \begin{cases} \Lambda \times \mathcal{M} \times \mathcal{S} & \longrightarrow \mathbb{R}^+, \\ (\lambda, x, \omega) & \longmapsto f(\lambda, x, \omega). \end{cases}$$

La fonction de radiance est définie d'un espace de dimension 5 vers un espace de dimension 1. Le choix des fonctions de base, et notamment du paramétrage de l'ensemble de définition de ces fonctions, est un problème primordial pour résoudre cette équation.

Le champ radiance représente une densité de flux par stéradian et par nanomètre, c'est-à-dire une quantité d'énergie rayonnée par unité de surface dans une direction et pour une longueur d'onde donnée. Son équation dimensionnelle est :

$$[f] = W/(m^2 \cdot sr \cdot nm).$$

1.2.3.3 Fonction g

La fonction g représente le terme initial de l'équation. Due en pratique aux sources lumineuses, elle représente l'énergie créée qui sert à illuminer la scène. L'éclairage direct, et bien sûr les inter-réflexions qui en résultent, dépendent totalement de ce terme. La précision avec laquelle il est évalué influe grandement sur le résultat f , donc il faut s'appliquer à bien modéliser ce terme.

Comme la fonction radiance f , la fonction locale g , parfois dénommée *fonction d'auto-émission*, est une densité de flux par stéradian et par nanomètre, d'équation dimensionnelle :

$$[g] = W/(m^2 \cdot sr \cdot nm).$$

1.2.3.4 Fonction k

La fonction k caractérise les propriétés physiques des matériaux associés aux surfaces. Idéalement, cette fonction devrait rendre compte de nombreux phénomènes. Malheureusement, la prise en compte de tous ces effets simultanément rendrait la résolution du problème bien trop complexe. De plus, ils n'interviennent que dans des cas très précis, qui sont en général assez rares.

Ainsi, nous négligeons les phénomènes de fluorescence, qui interviennent lorsque de l'énergie lumineuse d'une bande spectrale est ré-émise partiellement dans une autre bande

spectrale. Cet effet arrive avec des matériaux fluorescents, ou encore avec la *lumière noire*. Cela simplifie grandement l'expression de la fonction k , qui dépendrait d'un paramètre de plus.

Nous avons vu que les phénomènes de phosphorescence ne sont pas considérés. De plus, nous ne prendrons pas en compte les effets de polarisation de la lumière qui peuvent apparaître lors de l'interaction de la lumière avec un matériau.

Comme nous nous sommes restreints aux réflexions, cette fonction est appelée la *fonction de distribution bidirectionnelle de réflectance (brdf)*. C'est une propriété purement locale, indépendante de la globalité de la scène. De plus, dans le cas de surfaces planes, la sphère unité \mathcal{S} est restreinte à l'hémisphère Ω .

$$k : \begin{cases} \Lambda \times \mathcal{M} \times \mathcal{S}^2 & \longrightarrow \mathbb{R}^+, \\ (\lambda, x, \omega^\downarrow, \omega^\uparrow) & \longmapsto k(\lambda, x, \omega^\downarrow, \omega^\uparrow). \end{cases}$$

Elle représente, pour une longueur d'onde λ fixée, le rapport entre l'énergie reçue au point x dans un angle solide infinitésimal autour de la direction ω^\downarrow et l'énergie correspondante réfléchie dans la direction ω^\uparrow . L'équation dimensionnelle de la fonction k est :

$$[k] = sr^{-1}.$$

Le domaine de définition de cette fonction de réflexion est de dimension 7. De plus, cette fonction k rend compte d'interactions au niveau microscopique entre la lumière et des caractéristiques du matériau de la taille de la longueur d'onde. Ces phénomènes sont pris en compte par l'optique ondulatoire. Ainsi, de par sa nature, la fonction k est complexe et il n'existe pour ainsi dire pas d'expression générale capable de modéliser précisément n'importe quel matériau. On a donc recours à des modèles qui fournissent des approximations plus ou moins bonnes de la réalité physique.

1.2.3.5 Fonction implicite v

Dans l'équation (1.2), le terme y représente le point de la scène le plus proche de x , visible depuis x dans la direction $-\omega^\downarrow$. Pour formaliser cette notion, il faut introduire une fonction implicite y qui correspond à l'opération intuitive de tracer un rayon originaire de x dans la direction $-\omega^\downarrow$. Reprenons la définition de James ARVO [Arv 95] :

$$y = y(x, \omega^\downarrow) = x - v(x, \omega^\downarrow) \omega^\downarrow,$$

où la fonction v , appelée *fonction de visibilité*, est définie par :

$$v(x, \omega) = \inf_t \{t > 0; x - t\omega \in \mathcal{M}\}.$$

Elle retourne la plus petite distance du point x au point de \mathcal{M} le plus proche dans la direction $-\omega$. Si un tel point n'existe pas, on pose alors $v(x, \omega) = \infty$.

Cette fonction de visibilité a un caractère global, c'est-à-dire qu'elle reflète la géométrie entière de la scène. Ce terme est particulièrement important à deux niveaux. D'une part, les problèmes de visibilité sont les causes principales des discontinuités de la fonction de radiance. Il convient donc de calculer cette fonction implicite avec une grande précision.

D'autre part, l'évaluation de ce terme est d'une grande complexité, et le coût des calculs pour résoudre l'équation de radiance est en grande partie dû aux calculs de visibilité.

Toute la difficulté consiste donc à trouver un bon compromis entre la précision des calculs souhaitée et le temps nécessaire pour obtenir la solution.

1.2.4 Portée des hypothèses

Certaines hypothèses plus ou moins fortes ont été faites, et limitent le domaine de validité de l'équation de radiance (1.2).

L'hypothèse de radiosité n'est pas nécessaire pour établir le modèle des phénomènes lumineux. Néanmoins, elle est très souvent posée, pour faciliter la résolution de l'illumination globale.

1.2.4.1 Hypothèse de radiosité

Durant les premières années de recherche en synthèse d'images, les algorithmes existants ainsi que le matériel informatique ne permettaient pas de s'attaquer efficacement à un problème d'une telle envergure. Pour réduire la complexité de l'équation et ainsi la rendre résoluble en des temps acceptables, cette hypothèse forte de radiosité a été avancée.

Elle est basée sur le fait que la lumière se réfléchit de manière isotrope sur toutes les surfaces de la scène : cela revient à les considérer comme idéalement diffuses. Dans ce cas, un modèle *lambertien* est utilisé pour la distribution de la fonction bidirectionnelle de réflectance. L'utilisation des autres modèles qui prennent en compte la directionnalité est contradictoire avec l'hypothèse de radiosité. Néanmoins, certaines approches dérivées des méthodes de radiosité (la radiosité *directionnelle* introduite par David IMMEL *et al.* [Imm 86], les méthodes *multi-passes* proposées par John R. WALLACE *et al.* [Wal 87], ou encore l'utilisation d'harmoniques sphériques faite par François SILLION *et al.* [Sil 91]) permettent de prendre en compte ces modèles. Certains de ces modèles sont issus de l'optique géométrique [Coo 82], d'autres se basent sur l'optique ondulatoire [He 91, Coo 94], ou encore sur des mesures expérimentales [War 92].

Il résulte de l'hypothèse de radiosité que toutes les fonctions intervenant dans l'équation de radiance sont indépendantes de la variable directionnelle. Autrement dit :

$$\begin{aligned} \forall \omega, \omega' \in \mathcal{S}^2, \quad f(\lambda, x, \omega) &\equiv \tilde{f}(\lambda, x), \\ g(\lambda, x, \omega) &\equiv \tilde{g}(\lambda, x), \\ k(\lambda, x, \omega, \omega') &\equiv \rho(\lambda, x). \end{aligned}$$

Dans ce cas, \tilde{f} représente une densité de flux, c'est-à-dire une énergie par unité de surface et par unité de longueur d'onde. Cette fonction est alors nommée *radiosité spectrale* ou simplement *radiosité*. L'équation de radiosité, dérivée de l'équation de radiance, s'écrit sous la forme suivante :

$$\tilde{f}(\lambda, x) = \tilde{g}(\lambda, x) + \int_{\mathcal{M}} \tilde{f}(\lambda, y) \rho(\lambda, x) \frac{(\vec{n}(x) \cdot \vec{xy}) (\vec{n}(y) \cdot \vec{yx})}{\pi \|\vec{xy}\|^4} dS_y. \quad (1.3)$$

Deux ouvrages synthétisent les nombreuses recherches qui ont été effectuées sur la radiativité [Coh 93, Sil 94b]. Afin de simplifier les notations, il est d'usage de ne pas conserver le terme λ de la longueur d'onde dans l'expression de l'équation de radiativité. En effet, si on exclu les phénomènes de fluorescence, l'équation 1.3 peut être évaluée indépendamment pour chaque longueur d'onde λ . On réécrit alors généralement l'équation de la façon suivante³ :

$$\begin{aligned} B(x) &= E(x) + \rho(x) \int_S B(y) K(x,y) dy \\ K(x,y) &= G(x,y) V(x,y) \\ G(x,y) &= \frac{\cos \theta_x \cos \theta_y}{\pi \|x - y\|^2} \end{aligned} \quad (1.4)$$

où $V(x,y)$ est la visibilité entre x et y et vaut 1 si x voit y et 0 sinon. S correspond à l'ensemble des points 2D de \mathcal{M} . De cette façon pour la longueur d'onde λ considérée :

$$\begin{aligned} \tilde{f}(\lambda, x) &\equiv B(x), \\ \tilde{g}(\lambda, x) &\equiv E(x), \\ \rho(\lambda, x) &\equiv \rho(x). \end{aligned}$$

1.2.4.2 Milieu participatif

La prise en compte du milieu participatif peut se faire à deux niveaux différents. Une fois de plus, ceux-ci dépendent du modèle d'observation retenu pour modéliser les phénomènes.

Dans le contexte de l'optique géométrique, seule est considérée l'atténuation de l'énergie lumineuse le long d'un rayon. Dans ce cas, si s est le paramètre de la fonction de radiance f le long d'un rayon de direction ω , on obtient l'équation de transfert dans le milieu ambiant :

$$f(\lambda, x + s\omega, \omega) = f(\lambda, x, \omega) e^{-t_\lambda(s)}. \quad (1.5)$$

où la *distance optique* t_λ est donnée par $t_\lambda(s) = \int_0^s \tau_\lambda(s') ds'$, et le terme τ_λ représente le *coefficient d'extinction* au point considéré.

L'équation de radiance (1.2) peut alors être réécrite sous la forme :

$$f(\lambda, x, \omega^\uparrow) = g(\lambda, x, \omega^\uparrow) + \int_\Omega f(\lambda, y, \omega^\downarrow) e^{-t_\lambda(v(y, \omega^\downarrow))} k(\lambda, x, \omega^\downarrow \rightarrow \omega^\uparrow) d\mu.$$

La différence par rapport à l'expression classique réside dans l'apparition du terme d'atténuation dans l'équation. La résolution de ce problème se différencie de celle du problème dans le vide par une simple différence de l'évaluation du noyau de l'intégrale. Pratiquement, il faut bien sûr prévoir une structure adaptée, qui puisse stocker la valeur du coefficient

3. Ces notations correspondent à celle utilisée par Michael F. COHEN et John R. WALLACE dans [Coh 93] et seront réutilisées dans la suite de ce mémoire.

d'extinction en tous points de la scène. De plus, il ne faut pas non plus oublier de prendre en compte l'atténuation lors de la visualisation du résultat obtenu.

La prise en compte du milieu participatif devient bien plus complexe dans le cadre de l'optique ondulatoire. En effet, dans ce cas, il faut non seulement considérer le phénomène d'atténuation, mais aussi celui de diffusion⁴. Les particules du milieu ambiant absorbent alors une partie de l'énergie lumineuse pour la ré-émettre sous forme lumineuse, se comportant ainsi comme des sources secondaires.

L'équation de transfert (1.5) dans le milieu ambiant est alors passablement compliquée :

$$\frac{df_{\lambda}(\omega^{\uparrow})}{ds} = -\tau_{t,\lambda} f_{\lambda}(\omega^{\uparrow}) + \tau_{a,\lambda} f_{e,\lambda}(\omega^{\uparrow}) + \tau_{s,\lambda} \int_S f_{\lambda}(\omega^{\downarrow}) \Psi_{\lambda}(\omega^{\downarrow}, \omega^{\uparrow}) d\omega.$$

où l'on retrouve les coefficients d'extinction $\tau_{t,\lambda}$, d'absorption $\tau_{a,\lambda}$ et de diffusion $\tau_{s,\lambda}$. La fonction $f_{e,\lambda}$ représente l'auto-émission du milieu ambiant, alors que $\Psi_{\lambda}(\omega^{\downarrow}, \omega^{\uparrow})$ est le *facteur de polarisation* caractérisant le milieu, qui représente la fraction d'énergie lumineuse arrivant par la direction ω^{\downarrow} qui est diffusée dans la direction ω^{\uparrow} .

À notre connaissance, aucune publication n'étudie formellement ce problème. Néanmoins, de nombreuses approches tentent de le résoudre de manière empirique, en décomposant le volume contenant le milieu ambiant en petites cellules, pour le prendre en compte. Cette approche a été introduite par Holly E. RUSHMEIER et Kennet E. TORRANCE [Rus 87].

Dans le cadre de l'optique géométrique, la prise en compte du milieu ambiant ne pose pas trop de problèmes. Par contre, la complexité de cette approche d'un point de vue ondulatoire est toute autre.

1.2.4.3 Fluorescence

Nous avons considéré qu'aucune fuite d'énergie entre différentes bandes spectrales n'a lieu. Cela correspond à ne pas prendre en compte les phénomènes de fluorescence.

Pourtant, ces effets ne modifient pas considérablement l'équation de radiance. La seule différence est que la distribution de la fonction bidirectionnelle de réflectance a comme paramètres deux longueurs d'onde : $k(\lambda^{\downarrow} \rightarrow \lambda^{\uparrow}, x, \omega^{\downarrow} \rightarrow \omega^{\uparrow})$. Il faut donc ajouter au terme réfléchi une intégrale par rapport aux longueurs d'ondes pour obtenir l'équation de radiance appropriée :

$$f(\lambda, x, \omega^{\uparrow}) = g(\lambda, x, \omega^{\uparrow}) + \int_{\Omega} \int_{\lambda} f(\lambda', y, \omega^{\downarrow}) k(\lambda' \rightarrow \lambda, x, \omega^{\downarrow} \rightarrow \omega^{\uparrow}) d\lambda' d\mu.$$

La complexité du problème n'est que très peu modifiée. Le seul changement réside à nouveau dans le calcul du noyau de l'intégrale, qui comporte une simple intégrale sur le domaine des longueurs d'onde en plus. De plus, aucun changement n'intervient pour les calculs de visibilité, qui sont les plus pénalisants.

La difficulté, toute relative, vient de la modélisation de la fonction bidirectionnelle de réflectance.

4. En fait, le phénomène d'atténuation est une conséquence de celui de diffusion.

1.2.4.4 Autres hypothèses

Les dernières hypothèses faites sont de deux natures. La première est de nature temporelle, et concerne les phénomènes de phosphorescence. La seconde considère la structure de l'onde électromagnétique qu'est la lumière, et concerne la polarisation.

La phosphorescence s'accompagne généralement du phénomène de fluorescence. Une partie de l'énergie lumineuse reçue par une surface est absorbée, pour être ré-émise ultérieurement. Dans une certaine mesure, ce phénomène peut être modélisé en considérant que cette surface se comporte en tant que source de lumière. Il convient alors de déterminer l'intensité et le spectre de cette émission, qui dépendent des conditions d'éclairage préalable de cette surface.

L'effet de la polarisation de la lumière est souvent dû à la réflexion de la lumière sur certaines surfaces, notamment sur les vitres. La modélisation de ce phénomène demande de considérer la lumière non pas comme un flux d'énergie, mais comme une onde électromagnétique. Cela remet en cause toute la modélisation physique actuelle du problème.

Il se pose donc la question de savoir si l'influence de ces effets justifie la charge de travail nécessaire à les résoudre. En ce qui nous concerne, nous considérons ces effets comme négligeables par rapport aux applications qui nous intéressent.

1.3 Résolution mathématique

Une fois l'équation de radiance définie, il faut pouvoir lui trouver une solution. Aux méthodes *déterministes* s'opposent celles dites *stochastiques*, basées historiquement sur des phénomènes probabilistes. Ces dernières, basées sur des méthodes de lancer de rayons ou de particules, permettent depuis peu d'obtenir aussi des résultats indépendants du point de vue. Notre intérêt porte sur les méthodes déterministes par éléments finis.

Ces approches consistent à analyser les propriétés topologiques et métriques de l'espace de fonctions \mathcal{F} , auquel appartient la solution théorique f . La métrique fournit les bases pour l'intégration, alors que la topologie formalise les notions de proximité et de convergence qui sous-tendent l'analyse. Cette étude va permettre de définir des contraintes en terme de régularité, et de démontrer certaines caractéristiques fondamentales du problème, comme l'existence et l'unicité de la solution. De plus, elle permet de dégager des méthodes numériques aboutissant à l'approximation de la solution, ainsi qu'à une majoration de l'erreur.

1.3.1 Analyse fonctionnelle

1.3.1.1 Métrique et intégration de Lebesgue

L'équation (1.2) comporte une intégrale qui, comme toute intégrale, est basée sur une notion de mesure. La théorie de la mesure permet de formaliser cette notion.

Notons $\sigma(A)$ la mesure de l'aire du sous-ensemble A de la sphère \mathcal{S} . Elle permet de définir une autre mesure, plus appropriée au problème présent :

$$\mu(A) \equiv \int_A |\vec{\omega} \cdot \vec{n}(x)| d\sigma(\vec{\omega}),$$

où \vec{n} est la normale à la surface au point x . Ainsi, μ dépend implicitement du point x .

Cette nouvelle mesure introduit le terme cosinus dans l'équation, par l'intermédiaire du produit scalaire. Cela s'accorde avec la notion de densité de flux de la fonction de radiance. Cette mesure va servir tout au long de ce mémoire à définir l'intégration par rapport à la variable directionnelle.

L'intégration de LEBESGUE, qui s'appuie sur la notion de mesure, convient particulièrement à l'analyse fonctionnelle ; en effet, elle met en valeur la notion de transformation plutôt que l'aspect numérique de l'intégration. Bien que la notion d'intégrale de RIEMANN soit plus simple, elle est équivalente, lorsqu'elle existe, à celle de LEBESGUE. De plus, l'approche de LEBESGUE est bien plus robuste lors de passages aux limites. C'est pourquoi nous préférons la notion d'intégrale de LEBESGUE à celle de RIEMANN.

1.3.1.2 Espaces fonctionnels

Nous avons vu que la solution de l'équation de radiance est un élément de l'espace des fonctions à valeurs réelles définies sur $\Lambda \times \mathcal{M} \times \mathcal{S}$. Il convient donc de choisir la topologie⁵ sur laquelle baser l'analyse : nous avons opté pour la *topologie métrique*, qui dépend de la norme définie sur l'espace considéré.

Les normes L^p , exprimées par rapport à des intégrales de LEBESGUE, permettent de définir les espaces du même nom, notés L^p . Ces espaces dépendent des mesures μ et S sur lesquelles se basent les normes :

$$\|f\|_p = \left[\int_{\Lambda} \int_{\mathcal{S}} \int_{\Omega} |f(\lambda, x, \omega)|^p d\mu(\omega) dS(x) d\lambda \right]^{1/p},$$

où p est un nombre réel supérieur à 1 ou ∞ .

Une première propriété intéressante de ces espaces de fonctions est le fait qu'ils soient complets, c'est-à-dire que toute suite de CAUCHY est convergente. Ce sont donc des *espaces de Banach*, comme tout espace vectoriel complet.

Nous définissons donc l'espace fonctionnel auquel appartient f par :

$$\mathcal{F} = \{f : \Lambda \times \mathcal{M} \times \mathcal{S} \rightarrow \mathbb{R}^+, \text{ tel que } : f \in L^1, f \in L^2 \text{ et } f \in L^\infty\}.$$

En effet, la solution f que nous recherchons représente un phénomène physique, une densité de puissance lumineuse angulaire. Or, dans la réalité, la puissance lumineuse ne peut pas être infinie. De plus, il est raisonnable de supposer que la fonction de radiance n'est pas infinie sur un ensemble non négligeable. Donc, comme le domaine de définition est borné,

5. On peut définir différentes topologies sur un espace.

il en découle que la puissance p de la norme de f est intégrable. Ainsi, f appartient bien à l'espace \mathcal{F} .

Notons que la norme $\|\cdot\|_1$ de f représente la puissance (au sens physique du terme) de la fonction de radiance. La norme $\|\cdot\|_\infty$ quand à elle, correspond à la valeur maximale de la fonction définie sur un ensemble non négligeable. Pour finir, la norme $\|\cdot\|_2$ a des propriétés qui font que l'espace L^2 est un *espace de HILBERT*⁶.

En effet, la norme $\|\cdot\|_2$ est définie par rapport au produit scalaire suivant :

$$\langle f, g \rangle = \int_{\lambda} \int_S \int_{\Omega} f(\lambda, x, \omega) g(\lambda, x, \omega) d\mu dS d\lambda.$$

Il en découle naturellement la notion d'orthogonalité. C'est indispensable pour toutes les méthodes qui s'appuient sur des projections orthogonales, comme par exemple les méthodes de GALERKIN.

1.3.1.3 Opérateurs linéaires

Comme avec tout espace, on peut définir des fonctions qui transforment un élément de \mathcal{F} en un autre élément de \mathcal{F} . Dans le cas d'espaces fonctionnels, ces fonctions sont nommées *fonctionnelles* ou encore *opérateurs*. Parmi les opérateurs de \mathcal{F} vers \mathcal{F} , il y a notamment les opérateurs linéaires, ainsi que ceux de projection.

Les opérateurs linéaires \mathcal{H} vérifient la relation de linéarité classique suivante :

$$\mathcal{H}(\alpha f + \beta g) = \alpha \mathcal{H}(f) + \beta \mathcal{H}(g), \quad f, g \in \mathcal{F}, \alpha, \beta \in \mathbb{R}.$$

Une de leurs propriétés remarquables est que tout opérateur linéaire de norme finie est continu.

Les opérateurs dont le domaine d'arrivée est de dimension finie sont dits *dégénérés*. Certains opérateurs de projection sont dégénérés. Ce sont ceux qui projettent l'espace fonctionnel \mathcal{F} dans un sous-espace \mathcal{F}_n de dimension finie. Ils permettent de construire une approximation f_n de la solution, et de l'exprimer comme combinaison linéaire des éléments d'une base de \mathcal{F}_n .

Ce sont des opérateurs linéaires, K et G , qui permettent de récrire l'équation de radiance sous forme plus concise. Ils ont été introduits dans le domaine de la synthèse d'image par J. Kajiyama [Kaj 86], pour démontrer l'existence et l'unicité de la solution de l'équation de radiance. En utilisant la notation de [Arv 94], l'équation de radiance se récrit comme :

$$f = g + KGf, \tag{1.6}$$

où,

- $(Kh)(\lambda, x, \omega^\uparrow) \equiv \int_{\Omega} k(\lambda, x, \omega^\downarrow \rightarrow \omega^\uparrow) h(\lambda, x, \omega^\downarrow) d\mu(\omega^\downarrow)$,
- $(Gh)(\lambda, x, \omega^\downarrow) \equiv h(\lambda, y(x, \omega^\downarrow), \omega^\downarrow) = h(\lambda, x - v(x, \omega^\downarrow), \omega^\downarrow, \omega^\downarrow)$.

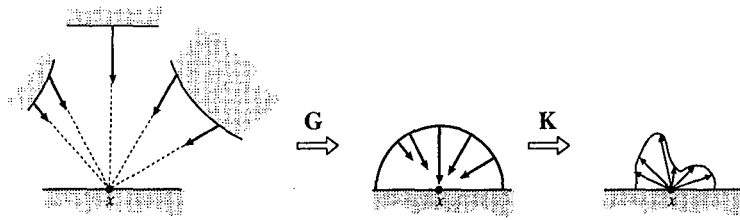


FIG. 1.3 – Visualisation de l'effet des opérateurs K et G (d'après J. Arvo).

Cette formulation (1.6) peut être rendue encore plus compacte, en introduisant l'opérateur linéaire M , défini par $M \equiv I - KG$ où I est l'opérateur identité. Ainsi, la fonction de radiance est exprimée directement en fonction de la fonction d'auto-émission :

$$Mf = g.$$

L'étude de l'opérateur M permet de démontrer l'existence de l'opérateur inverse M^{-1} , et de le développer en séries de Neumann convergentes⁷. Il en découle que les espaces L^p sont complets dans le cadre de l'illumination globale [Arv 95]. Ainsi, pour toute fonction d'auto-émission g de L^p , il existe une solution unique, la fonction de radiance, elle-même dans L^p .

1.3.2 Approche variationnelle

1.3.2.1 Formulation variationnelle

Parmi les différentes approches pour résoudre ce problème, nous avons opté pour une approche déterministe. Il faut donc reposer le problème sous une forme différente, plus facile à résoudre. Une approche consiste à exploiter la *formulation variationnelle* de l'équation de radiance, qui est une formulation *faible* mais équivalente du problème⁸.

En nous plaçant dans L^2 , espace hilbertien réel, soit la forme bilinéaire a définie par :

$$a : \begin{cases} \mathcal{F} \times \mathcal{F} & \longrightarrow \mathbb{R}, \\ (u, w) & \longrightarrow \langle Mu, w \rangle. \end{cases}$$

Cette fonction est continue sur \mathcal{F} , car c'est la forme bilinéaire associée à l'opérateur M , lui-même continu. De plus, elle satisfait la condition de coercivité comme l'a montré Slimane MERZOUK dans sa thèse [Mer 97].

Soit la forme linéaire continue l définie comme suit :

$$l : \begin{cases} \mathcal{F} & \longrightarrow \mathbb{R}, \\ w & \longrightarrow \langle g, w \rangle. \end{cases}$$

6. Un espace de HILBERT est un espace muni d'un produit scalaire, complet pour la norme issue de celui-ci.

7. Ce résultat n'est pas valable dans le cas de réflexions parfaites, c'est-à-dire sans aucune absorption, ce qui n'est pas le cas dans un monde réel.

8. Généralement, il n'y a pas équivalence entre une formulation *forte* et une formulation *faible* d'un problème. Mais l'existence d'une solution forte à la formulation forte implique l'existence d'une solution à la formulation faible, qui est la même que la solution forte.

Dans ces conditions, la formulation variationnelle du problème s'exprime comme suit :

$$\text{trouver } u \text{ dans } \mathcal{F} \text{ tel que : } \forall w \in \mathcal{F} \quad a(u,w) = l(w). \quad (1.7)$$

Les propriétés des deux formes a et l permettent de re-démontrer l'existence et l'unicité de la solution. Pourtant, cette formulation ne permet toujours pas de trouver la solution, qui est un élément de l'espace fonctionnel de dimension infinie \mathcal{F} .

1.3.2.2 Approximation variationnelle

Pour rendre le problème calculable, il faut chercher à le résoudre dans un espace de dimension finie. Pour formaliser cette approximation, une solution consiste à utiliser l'*approximation variationnelle*, encore appelée *méthode de GALERKIN*. Pour tout sous-espace \mathcal{F}_n de dimension n , on a :

$$\text{trouver } u_n \text{ dans } \mathcal{F}_n \text{ tel que : } \forall w_n \in \mathcal{F}_n \quad a_n(u_n, w_n) = l_n(w_n). \quad (1.8)$$

De même que pour la formulation variationnelle, l'existence et l'unicité de la solution est assurée. De plus, cette méthode garantit la convergence forte de la suite des solutions u_n vers la solution exacte u , lorsque n tend vers ∞ . Il s'agit en fait de la projection du problème dans un sous-espace \mathcal{F}_n relativement au produit scalaire.

Résoudre cette équation (1.8) est équivalent à résoudre un système d'équations linéaires. Pour l'établir il suffit d'exprimer u_n comme combinaison linéaire des éléments d'une base de \mathcal{F}_n .

1.3.2.3 Fonctions de base

En dimension finie, un espace vectoriel est défini par une de ses bases. Si (ϕ_i) est une base de l'espace \mathcal{F}_n , toute fonction f_n de cet espace est définie par ses coefficients (α_i) , tels que :

$$f_n = \sum_{i=0}^n \alpha_i \phi_i.$$

Le choix des fonctions de base est primordial. Il conditionne la précision de l'approximation par rapport à la solution réelle de l'équation de radiance. Par ailleurs, ce choix influence aussi la complexité des calculs pour résoudre le problème. Il s'agit donc de trouver le bon compromis entre la précision du résultat et la complexité des calculs.

Historiquement, les premiers calculs de radiosit  se sont bas s sur l'utilisation de fonctions de base constantes par morceaux [Gor 84]. Mais le choix des fonctions de base reste vaste : fonctions constantes, fonctions polynomiales d'ordre sup rieur, fonctions on-delettes...

1.3.2.4 Système linéaire

La construction du système linéaire correspondant au problème de l'illumination globale repose sur plusieurs propriétés : toute fonction de \mathcal{F}_n peut s'exprimer comme combinaison linéaire des fonctions d'une base \mathcal{B}_n , et les formes a et l sont respectivement bilinéaire et linéaire.

Ainsi, le problème peut être reposé sous la forme suivante :

$$\text{trouver } (\alpha_i) \in \mathbb{R}^n \text{ tels que : } \forall \phi_k \in \mathcal{B}_n, \sum_{i=0}^n \alpha_i a(\phi_i, \phi_k) = l(\phi_k).$$

Ceci peut être reformulé en adoptant l'écriture matricielle :

$$\begin{bmatrix} a(\phi_1, \phi_1) & a(\phi_1, \phi_2) & \cdots & a(\phi_1, \phi_n) \\ a(\phi_2, \phi_1) & a(\phi_2, \phi_2) & \cdots & a(\phi_2, \phi_n) \\ \vdots & \vdots & \ddots & \vdots \\ a(\phi_n, \phi_1) & a(\phi_n, \phi_2) & \cdots & a(\phi_n, \phi_n) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} L(\phi_1) \\ L(\phi_2) \\ \vdots \\ L(\phi_n) \end{bmatrix}. \quad (1.9)$$

Nous avons formalisé une approche du problème de l'illumination globale. Celle-ci est basée sur des choix de nature physique, puis sur des choix d'analyse, pour aboutir à un système linéaire. La résolution finale du problème posé dans sa nouvelle forme repose sur des techniques de calcul permettant d'évaluer les termes de la matrice, ainsi que sur des méthodes mathématiques de résolution de système.

Les algorithmes publiés dans la littérature sont généralement des cas particuliers de la formulation proposée dans l'équation (1.9). Ils dépendent du choix des fonctions de base ϕ_i , des méthodes de calcul des coefficients $a(\phi_i, \phi_j)$ et $l(\phi_i)$, et enfin de la méthode de résolution du système linéaire.

1.3.3 Résolution du système linéaire

La résolution de ce système linéaire repose sur deux facteurs : la méthode de calcul des coefficients de la matrice, mais aussi la technique de résolution du système.

Dans un premier temps, nous verrons comment faire une approximation des calculs de produits scalaires qui font intervenir des intégrales. Puis nous présentons diverses techniques d'analyse numérique permettant de résoudre ce système.

1.3.3.1 Calcul des coefficients $a(\phi_i, \phi_j)$ et $l(\phi_i)$

Les coefficients de la matrice peuvent, dans de très rares cas de configurations extrêmement simples, se déterminer de manière analytique. Pour pouvoir traiter la quasi totalité des cas, il faut se résoudre à utiliser des solutions numériques.

Différentes méthodes existent pour faire une approximation de la valeur de l'intégrale intervenant dans le calcul des coefficients de la matrice. Nous nous sommes arrêtés sur

des méthodes de quadrature qui permettent de faire une approximation de la valeur d'une intégrale par une somme pondérée :

$$H = \int_X h(x) dx \approx \tilde{H} = \sum_k \omega_k h(x_k),$$

où (x_k) sont les points de quadrature, et (ω_k) leurs poids associés. La précision des calculs dépend du nombre de points de quadrature utilisés, ainsi que de leur choix et de celui des poids associés.

Parmi ces méthodes, celle des quadratures de GAUSS permet d'évaluer de manière exacte des polynômes de degré $2n - 1$ avec n points. Ces quadratures sont donc bien appropriées pour des calculs sur des fonctions polynomiales, et permettent de fixer *a priori* le nombre de points de quadrature. Il est très important de minimiser ce nombre, pour éviter d'évaluer inutilement des termes très coûteux en temps de calcul.

Dans le cas de la synthèse d'images, cette intégration se fait en deux dimensions, sur le support des fonctions de base. Ces supports sont généralement des surfaces simples comme des triangles ou des parallélogrammes⁹. La double intégrale est alors évaluée comme deux intégrations successives en une seule dimension.

La restriction sur la forme des supports est dictée par un souci d'efficacité. En effet, les triangles et les parallélogrammes permettent l'utilisation de coefficients précalculés, pour les relations entre les différents niveaux de la hiérarchie de fonctions de base.

Le calcul des coefficients découle directement de la définition de la forme a et de son expression sous forme intégrale :

$$a(\phi_i, \phi_j) = \langle M\phi_i, \phi_j \rangle = \int_{S_1} \int_{S_2} M(x, y) \phi_i(x) \phi_j(y) dx dy$$

On utilise alors les quadratures de GAUSS pour discrétiser le problème et évaluer les coefficients aussi précisément que possible.

Ces calculs permettent d'établir la valeur des coefficients du système d'équations qu'il faut résoudre. Deux catégories de méthodes de résolution existent : les méthodes directes, et celles itératives.

Nous pouvons d'ores et déjà écarter les méthodes directes, car elles sont inadaptées à notre problème. En effet, ces méthodes nécessitent de calculer l'inverse de la matrice représentant le système linéaire. Comme cette matrice est de taille n^2 , la complexité de l'inversion est bien trop grande pour des applications du monde réel.

1.3.3.2 Méthodes de résolution itératives

Contrairement aux méthodes directes qui aboutissent à une solution exacte du système d'équations, les méthodes itératives se contentent de trouver une approximation à la solution. Elles sont basées sur la construction d'une suite de vecteurs $(x^{(k)})$ qui converge vers la solution de $Ax = b$.

9. Nous verrons dans le chapitre 3 (p. 83) comment nous abordons des supports plus complexes, en utilisant le maillage virtuel.

L'avantage de ces méthodes réside dans le fait qu'il n'y a pas forcément de problème de stockage. En effet, il est possible de calculer au vol les coefficients de la matrice nécessaires à une itération, sans nécessairement devoir les stocker.

Dans ces méthodes itératives, le passage d'un vecteur $x^{(k)}$ de la suite au suivant se fait en corrigeant successivement une ou plusieurs composantes de ce vecteur. Ces corrections s'effectuent en annulant une ou plusieurs composantes du vecteur résidu $r^{(k)} = b - Ax^{(k)}$.

Les itérations sont arrêtées lorsqu'un critère d'arrêt est satisfait. Usuellement, le critère est l'un des deux suivants :

$$\frac{\|r^{(k)}\|}{\|b\|} \leq \epsilon \quad \text{ou} \quad \|x^{(k)} - x^{(k+1)}\| \leq \epsilon \|x^{(k)}\|.$$

Il faut néanmoins se méfier de la deuxième condition, puisque celle-ci peut être vérifiée sans que $x^{(k)}$ soit proche du résultat.

Les normes les plus utilisées sont la norme du maximum ou la norme euclidienne.

Voici donc les principales méthodes itératives utilisées.

1.3.3.3 Méthode de JACOBI

L'expression des vecteurs de la suite est la suivante :

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j^{(k)} \right).$$

Cette méthode de JACOBI n'est définie que si $a_{ii} \neq 0$, c'est-à-dire si la diagonale de la matrice n'a pas de terme nul. C'est le cas en synthèse d'images, puisque a est positive définie, donc cette méthode ainsi que toutes celles dérivées sont applicables.

Algorithme 1 Une itération de la méthode de JACOBI

ENTRÉES: deux vecteurs x et y

- 1: **pour** tous les i **faire**
 - 2: $S \leftarrow b_i$
 - 3: **pour** tous les j différents de i **faire**
 - 4: $S \leftarrow S - a_{ij} x_j$
 - 5: **fin pour**
 - 6: $y_i \leftarrow S/a_{ii}$
 - 7: **fin pour**
 - 8: **pour** tous les i **faire**
 - 9: $x_i \leftarrow y_i$
 - 10: **fin pour**
-

La complexité de cette approche est en $O(n^2)$ par itération.

1.3.3.4 Méthode de GAUSS-SEIDEL

Par rapport à la méthode de JACOBI, celle de GAUSS-SEIDEL prend en compte les nouvelles valeurs des coefficients du vecteur $x^{(k+1)}$ pendant l'itération. Cela évite d'attendre la fin d'une itération pour tirer profit des calculs.

Dans ce cas, le calcul de la suite de vecteurs devient :

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right).$$

Algorithme 2 Une itération de la méthode de GAUSS-SEIDEL

ENTRÉES: un vecteur x

- 1: **pour** tous les i **faire**
 - 2: $S \leftarrow 0$
 - 3: **pour** tous les j différents de i **faire**
 - 4: $S \leftarrow S + a_{ij} x_j$
 - 5: **fin pour**
 - 6: $x_i \leftarrow b_i/a_{ii} - S/a_{ii}$
 - 7: **fin pour**
-

La complexité de cette méthode reste en $O(n^2)$, mais la convergence est plus rapide puisqu'on utilise plus rapidement des nouvelles valeurs de la suite de vecteurs.

1.3.3.5 Méthode de SOUTHWELL

Une dernière amélioration apportée à cette méthode réside dans le choix de l'ordre des coefficients de $x^{(k+1)}$ à calculer. L'indice choisi est celui pour lequel le résidu a le plus grand coefficient r_i .

Algorithme 3 Une étape de la méthode de SOUTHWELL

ENTRÉES: deux vecteurs $x \leftarrow 0$ et $r \leftarrow b$

- 1: choisir i tel que r_i est le plus grand
 - 2: $x_i \leftarrow x_i + r_i/a_{ii}$
 - 3: $S \leftarrow r_i$
 - 4: **pour** tous les j **faire**
 - 5: $r_j \leftarrow r_j - a_{ji}/a_{ii} S$
 - 6: **fin pour**
-

Le choix de l'indice i peut se faire en un temps $O(n)$, donc la complexité globale d'une itération reste elle aussi en $O(n^2)$.

1.4 Aspects algorithmiques

Les différentes notions que nous avons mises à jour dans les sections précédentes ont toutes un rôle bien défini dans le processus de résolution de l'équation de radiance. Ces rôles ont largement été étudiés individuellement par Slimane MERZOUK et par Christophe WINKLER dans leur thèses [Mer 97, Win 98]. Ils y décrivent précisément l'importance de chacune de ces notions et présentent les choix qui ont été faits pour leur implantation dans la plate-forme logicielle Candela. Les spécificités algorithmiques qui donnent à Candela la capacité de traiter des scènes extrêmement complexes aussi bien en terme de nombre de primitives qu'en terme de précision sont exposées dans le rapport de recherche [Cun 98b].

Dans cette section, nous nous concentrons sur certaines de ces notions. Nous détaillons l'origine et le rôle de celles qui nous ont intéressés dans notre recherche de précision et de fondements mathématiques pour la radiosit . Ainsi, nous approfondissons la pr sentation des choix algorithmiques relatifs aux fonctions de base ϕ_i et   l'ensemble g om trique \mathcal{M} . Par ailleurs l' tude des algorithmes relatifs aux fonctions f et g ,   la fonction de r flectance k ,   la d pendance spectrale λ ,   la visibilit  v et   la r solution du syst me lin aire ne sont pas abord es ici mais a d j   t  pr sent e par Christophe WINKLER dans sa th se [Win 98].

1.4.1 Ensemble \mathcal{M}

Cet ensemble, comme nous l'avons vu, repr sente la g om trie de la sc ne   illuminer. Diff rents types d'approche existent pour repr senter une sc ne : soit une description volumique des objets composant la sc ne, soit la mod lisation des surfaces limitant ces volumes. La seconde approche est certainement plus judicieuse pour notre application, puisque les interactions entre la lumi re et la g om trie se font au niveau des surfaces des objets.

Par la suite, nous nommons *primitive g om trique* toute repr sentation d'une surface, au sens g n ral du terme. Ainsi, un triangle, un polygone convexe, concave ou m me trou , ou encore une surface spline ou une NURBS sont autant de primitives g om triques. Les limitations pratiques quant   leur forme proviennent de consid rations de faisabilit  technique, et d pendent de l'utilisation de ces primitives.

1.4.1.1 D crire la g om trie

Le premier but des primitives est de d crire la g om trie de la sc ne, ce qui leur conf re un r le de persistance et de base de donn es. Il faut pouvoir  crire la description de la sc ne dans des fichiers pour pouvoir les r utiliser. Par ailleurs, la gestion des primitives   l'int rieur de la sc ne doit  tre pratique et facile. Il faut par exemple pouvoir d crire ces primitives dans des rep res locaux, tout en gardant la possibilit  de retrouver leurs caract ristiques dans le rep re absolu.

Pour mieux pr voir les formes que peuvent prendre ces donn es, il faut conna tre leur origine. Dans les applications du monde r el, la description de la sc ne provient de nombreux moyens d'acquisition.

L'origine la plus courante reste certainement les modeleurs du commerce, qui fournissent des données de formats divers. Certains fournissent directement des représentations surfaciques de la géométrie. Ce sont par exemples les modeleurs polygonaux, comme Arc+, ou les modeleurs à base de primitives plus complexes, comme Maya, 3DStudio-Max et Catia. D'autres ont une approche volumique, et procèdent par opérations booléennes sur des volumes ou des solides. Ce sont des modeleurs de type CSG (*Constructive Solid Geometry*), comme PDMS et SGDL, et leur résultat doit être transformé en surfaces pour pouvoir être utilisé. Les modeleurs peuvent être utilisés à partir de données mesurées, mais aussi de relevés architecturaux ou même archéologiques.

D'autres moyens moins courants proviennent de mesures faites par laser, qui génèrent de grandes quantités de triangles, ou encore de générateurs de fractales qui permettent de modéliser des plantes, des montagnes...

La représentation géométrique est primordiale dans la précision et l'efficacité de l'algorithme de radiosit . En effet, il est g n ralement impossible de repr senter exactement une sc ne r elle au moyen d'un mod le num rique. Les mod les sont, donc, des approximations g om triques de la r alit  ou des environnements virtuels imagin s par les concepteurs. Or, plus le mod le est capable d'int grer des surfaces complexes plus cette approximation est pr cise et proche du mod le original (r el ou virtuel). Par exemple, un mod le   base de polygones ne pourra jamais d crire exactement la courbure de la coque d'un bateau par contre cela pourra  tre fait pr cis ment si on mod lise   base de NURBS. En plus de cr er une erreur g om trique, ces approximations provoquent souvent la multiplication des surfaces initiales : repr senter une sph re avec des triangles peut requ rir plusieurs milliers de primitives si la pr cision demand e est importante alors que cela ne requiert qu'une seule primitive, quelle que soit la pr cision demand e, si on utilise des surfaces quadriques. On comprend bien, ici, que l'efficacit  de l'algorithme de radiosit , qui d pend fortement du nombre de primitives initiales, repose en grande partie sur la repr sentation g om trique et qu'une mauvaise repr sentation peut rendre une simulation impossible alors qu'elle serait « facile » et beaucoup plus pr cise avec une autre repr sentation mieux adapt e.

La diversit  des moyens d'acquisition et l' ventail des sc nes   simuler, nous ont pouss    faire le moins possible d'hypoth ses sur la forme et la nature des primitives g om triques. Pratiquement, des consid rations de faisabilit  et de calcul imposent des restrictions sur la nature des surfaces qu'il est possible de traiter effectivement. N anmoins, les travaux que nous pr sentons dans le chapitre 3 (p. 83) et que nous avons expos s dans [Alo 00b, Alo 00a, Cun 98a, Cun 99, Hol 00] donnent des m thodes efficaces pour traiter des surfaces de nature quelconque.

Les primitives g om triques interviennent   deux autres niveaux dans le probl me de l'illumination globale.

1.4.1.2 Support de la fonction de radiosit 

En premier lieu, les surfaces correspondent aux supports de la fonction inconnue f . Il faut donc pr voir une relation entre les primitives et le c t  fonctionnel. Cette relation a deux facettes : premi rement, il faut pouvoir stocker toutes les donn es caract risant

la fonction f sur la surface. Cela correspond à la détermination d'une base de fonctions, et au stockage des coefficients associés à chacune des fonctions de base. Deuxièmement, comme notre approche est hiérarchique (cet aspect sera développé dans ce qui suit), il faut associer aux surfaces des structures permettant de décrire des mailles qui servent de support aux fonctions de base. Ces mailles doivent aussi pouvoir être subdivisées pour affiner la précision de l'approximation de f .

Dans notre approche, il nous est apparu important de bien séparer la représentation géométrique de la notion de support fonctionnel. En effet, quand on considère une primitive géométrique en tant que support d'une fonction cela implique un grand nombre de restrictions comme nous le verrons dans le chapitre 3 (p. 83). Ces restrictions sont telles qu'elles ne permettent pas l'utilisation de n'importe quel type de primitives. En séparant la représentation géométrique des supports de la fonction de radiosit  on est alors capable d'enrichir la g om trie pour en avoir une repr sentation fid le mais plus compl te pouvant  tre utilis e comme support de la fonction. C'est cette deuxi me repr sentation abstraite de la g om trie que nous appelons maillage virtuel et qui nous permet de donner une approche tr s g n rique pour int grer proprement des primitives complexes dans n'importe quel algorithme de radiosit .

1.4.1.3 Calcul de visibilit 

  Pour finir, les primitives interviennent dans les calculs de visibilit . Ce sont elles qui, en fonction de leurs positions relatives, vont d terminer la visibilit  entre des points et ainsi cr er les ombres port es. Pour r soudre ce probl me, les primitives interviennent au niveau des calculs d'intersections avec des droites. De plus, une optimisation peut  tre faite   l'aide de volumes englobants qui d finissent une notion d'occupation de l'espace par les primitives.

1.4.2 Fonctions de base (ϕ_i)

L'utilisation de fonctions de base permet de faire une approximation de la fonction de radiance. Celle-ci est alors d finie par une s rie de poids α_i et s'exprime sous la forme d'une somme pond r e de ces fonctions. Pour la notation simplifi e, une approximation de la fonction de radiosit  B s' crit :

$$\tilde{B}(x) = \sum_i \alpha_i \phi_i(x) \quad (1.10)$$

Cette notion de fonctions de base est d'origine math matique. Elle est formellement introduite dans le processus de r solution lors de l'analyse fonctionnelle du probl me. Les fonctions de base d finissent l'espace fonctionnel dans lequel on cherche f (ou B pour la radiosit ). Ce choix va induire une aptitude plus ou moins grande   prendre en compte les variations de la fonction f . Ceci conditionne partiellement la pr cision avec laquelle la fonction f peut  tre calcul e. Il faut donc pouvoir choisir, pour l'algorithme de r solution, quel type de fonctions utiliser. Plusieurs bases de fonctions peuvent repr senter le m me espace fonctionnel mais chaque fonction a une unique repr sentation dans une base donn e.

Par ailleurs, le type de fonctions de base intervient sur les temps de calcul. En effet, les calculs de projection effectués pour déterminer les valeurs du noyau dépendent des fonctions de base utilisées. La place mémoire occupée par les coefficients qu'il faut stocker pour représenter f est elle aussi dépendante du type de base utilisée.

Comme nous l'avons vu lors de l'analyse de l'ensemble \mathcal{M} des primitives géométriques, chaque fonction est définie sur un support. Les fonctions de base doivent donc être liées à une gestion du maillage. Cela doit permettre de choisir entre différentes techniques de maillage, soit adaptatif, soit hiérarchique ou encore de discontinuité.

1.4.2.1 Radiosité hiérarchique

Une grande avancée relative aux fonctions de base a été faite avec l'introduction des fonctions hiérarchiques par Pat HANRAHAN *et al.* dans [Han 91]. L'idée est d'utiliser une base de fonction à plusieurs niveaux de précision :

$$\{ \{ \phi_1^1, \phi_2^1, \dots, \phi_{n_1}^1 \}, \{ \phi_1^2, \dots, \phi_{n_2}^2 \}, \dots, \{ \phi_1^N, \dots, \phi_{n_N}^N \} \}.$$

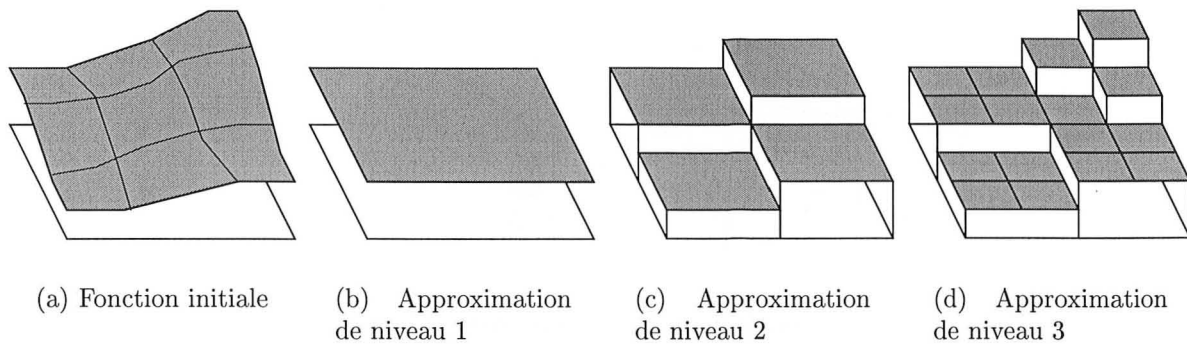
La fonction de radiance peut être exprimée dans chacune de ces base. Ainsi, pour chaque niveau de la hiérarchie on a une représentation de la fonction de radiosité que l'on peut utiliser sur tous les points de la scène.

$$\left. \begin{aligned} \tilde{B}^1(x) &\equiv \sum_{i=1}^{n_1} \alpha_i^1 \phi_i^1(x) \\ \tilde{B}^2(x) &\equiv \sum_{i=1}^{n_2} \alpha_i^2 \phi_i^2(x) \\ &\vdots \\ \tilde{B}^N(x) &\equiv \sum_{i=1}^{n_N} \alpha_i^N \phi_i^N(x) \end{aligned} \right\} \text{et } \forall i \in [1, N] : B(x) \sim \tilde{B}^i(x)$$

Dans l'algorithme de radiosité hiérarchique, toutes les fonctions de base sont constantes par morceaux et les différents niveaux de la hiérarchie correspondent donc aux différents niveaux de subdivisions des surfaces (voir figure 1.4). Plus la précision demandée est importante plus la profondeur des subdivisions est grande et plus les calculs sont longs; si le niveau 1 est celui de plus faible précision, on a toujours $n_1 < n_2 < \dots < n_N$ car le nombre de mailles augmentent avec les subdivisions, on sait aussi que $\tilde{B}^i(x)$ est une moins bonne approximation de $B(x)$ que $\tilde{B}^{i+1}(x)$ quel que soit $i \in [1, N-1]$.

Pour assurer la cohérence du modèle il y a une équivalence entre les différents niveaux : chaque fonction de base à un niveau donné peut être exprimée par une somme pondérée des fonctions de base du niveau inférieur.

$$\phi_j^i(x) = \sum_k \beta_{j,k}^{i+1} \phi_k^{i+1}(x)$$

FIG. 1.4 – *Subdivisions hiérarchiques*

où les poids β représentent le rapport des aires entre le niveau i et le niveau $i + 1$,

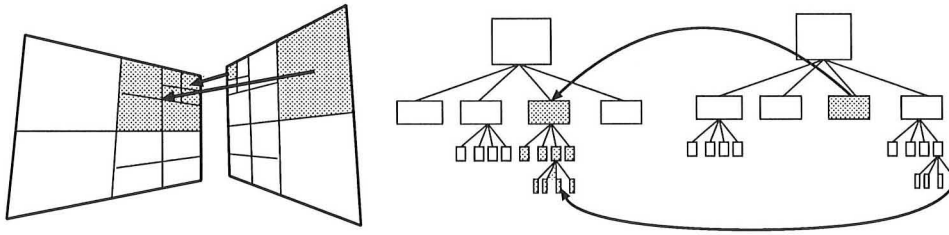
$$\beta = \frac{\text{aire}(\text{maille de niveau } i + 1)}{\text{aire}(\text{maille de niveau } i)}.$$

De même, on peut donner une approximation de la fonction à un niveau inférieur par une somme pondérée de fonctions de niveaux plus élevés. Cette équivalence entre les niveaux permet de maintenir une hiérarchie de représentations cohérente dans laquelle tous les niveaux renferment la même énergie et l'information des niveaux inférieurs contient toujours au moins l'information des niveaux supérieurs. Ainsi, les niveaux les plus bas permettent une représentation plus détaillée et plus précise de la fonction. Par contre, cette représentation fait intervenir plus de fonctions.

Pour utiliser efficacement la représentation hiérarchique de la fonction de radiance, il faut introduire une nouvelle notion, celle des *oracles*. Les oracles sont des fonctions qui permettent de choisir à quel niveau de précision chaque calcul d'interaction doit être fait. Cette notion est très importante car c'est elle qui, tout le long de l'algorithme, va régler la précision des calculs et par conséquent le temps et la mémoire nécessaire.

Grâce aux bases de fonctions hiérarchiques et aux oracles, on peut concentrer les calculs là où c'est vraiment nécessaire et faire un minimum de calculs là où la précision n'a pas besoin d'être très grande. Par exemple, l'interaction entre deux mailles très proches sera faite avec le maximum de précision alors qu'une interaction entre deux mailles éloignées pourra être faite avec un niveau de précision très faible et ne demander que très peu de calculs (voir figure 1.5).

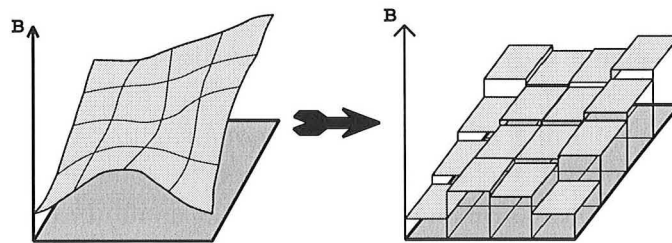
La complexité de l'algorithme classique est en $O(n^2)$ où n est le nombre de mailles finales. Avec les bases de fonctions hiérarchiques chaque surface initiale est à l'origine d'une représentation hiérarchique de la géométrie et la complexité de l'algorithme devient $O(k^2 + n)$ où k est le nombre de primitives initiales et n le nombre de mailles finales. Avec cette nouvelle complexité, il est possible de pousser beaucoup plus loin la précision des calculs et d'obtenir des solutions de très grande qualité en un minimum de temps. Le terme en k^2 reste néanmoins un point bloquant dans l'algorithme et de récents travaux ont permis de le faire disparaître en généralisant la représentation hiérarchique à toute la scène. Les méthodes de *clustering* proposées par Brian SMITS *et al.* dans [Smi 94] et

FIG. 1.5 – *Subdivisions et interactions hiérarchiques*

par François SILLION *et al.* dans [Sil 94a, Sil 95a] opèrent par regroupement de surfaces pour créer une hiérarchie englobant l'ensemble des primitives géométriques et ramener la complexité de l'algorithme à $O(k \log k + n)$.

1.4.2.2 Fonctions d'ordre élevé

Pour augmenter la précision des résultats, on peut soit augmenter le nombre de subdivisions et donc la quantité de calculs nécessaires pour évaluer chaque interaction, soit augmenter l'ordre des fonctions de base. Cette dernière solution consiste à utiliser des fonctions dont le pouvoir d'approximation est accru grâce aux variations internes de chaque fonction de base. On utilise ces variations internes pour mieux « coller » aux variations de la fonction de radiosit  et ainsi diminuer le nombre de fonctions de base nécessaires pour avoir une meilleure approximation.

FIG. 1.6 – *Approximation avec une base de fonctions constantes.*

À l'origine, les méthodes de radiosit  utilisaient toutes des fonctions de base constantes comme cela est présenté sur la figure 1.6. Ces fonctions demandent un découpage très fin pour faire une approximation de la radiosit  dont les variations sont nombreuses et généralement continues. L'utilisation de fonction d'ordre plus élevé, comme les fonctions linéaires présentées sur la figure 1.7 permet de « coller » plus rapidement et donc plus efficacement à la fonction de radiosit .

Les solutions proposées dans la littérature [Gor 93, Zat 93, Tro 93, Sch 94b] pour construire des bases de fonctions en 2D s'appuient sur l'utilisation d'un produit tensoriel de deux bases de fonctions en une dimension (la figure 1.8 montre la construction d'une base bi-linéaire). Cette construction utilise généralement des bases orthonormales de polynômes 1D telles que les bases de LEGENDRE ou de JACOBI. La figure 1.8 correspond aux deux premiers polynômes de LEGENDRE : $\phi_1(x) = 1$ et $\phi_2(x) = x$.

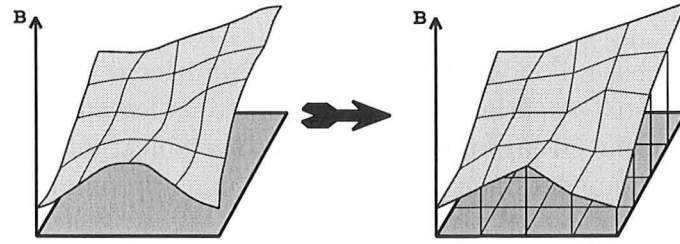


FIG. 1.7 – Approximation avec une base de fonctions non constantes.

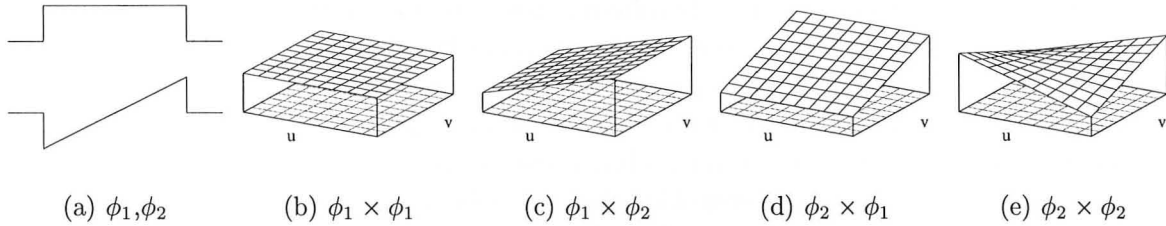


FIG. 1.8 – Construction d'une base linéaire en 2D

Grâce aux fonctions d'ordre élevé le nombre de fonctions nécessaires pour faire l'approximation de la fonction de radiosit  diminue fortement. Par contre, plus l'ordre des fonctions est  lev  plus les calculs d'interaction sont compliqu s. La m thode de GALERKIN introduite par Harold ZATZ dans [Zat 93] est une approche naturelle pour int grer les fonctions d'ordre  lev  dans l'algorithme de radiosit .

1.4.2.3 Ondelettes

Les ondelettes permettent de faire le lien entre les approches hi rarchiques et les fonctions d'ordre  lev . La th orie des ondelettes prend sa source dans les recherches en math matiques pures [Dau 92] et en traitement du signal [Mal 89]. Une tr s bonne introduction sur le sujet pourra  tre trouv e dans [Chu 92].

G. BEYKLIN *et al.* ont prouv  que, pour un grand nombre d'op rateurs lin aires, le syst me r sultant de la projection dans une base d'ondelettes pr sente beaucoup de coefficients proches de z ro [Bey 92]. Ils ont montr  que pour la classe particuli re d'op rateurs int grales appel s op rateurs de Calderon-Zygmund et pour tout $\varepsilon > 0$, il existe un $\delta(\varepsilon)$ tel que toutes les entr es de la matrice, sauf $O(n \log n)$, sont inf rieures   $\delta(\varepsilon)$ et peuvent  tre ignor es en causant une erreur inf rieure   ε dans le r sultat du calcul final.

Un op rateur int gral $K(f) = \int k(.,y)f(y)dy$ est dit de Calderon-Zygmund si le noyau $k : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ de l'int grale pr sente les propri t s suivantes :

$$\begin{aligned} \|k(x,y)\| &\leq \frac{1}{\|x-y\|^n} \\ \|\partial_x^M k\| + \|\partial_y^M k\| &\leq \frac{C_M}{\|x-y\|^{n+M}} \end{aligned} \quad \text{pour au moins un } M \text{ positif donn .}$$

En radiosit , $k(x,y) = \rho_\lambda(x) \frac{\cos \theta_x \cos \theta_y}{\pi r_{xy}^2} V(x,y)$ o  $r_{xy}^2 = \|x - y\|^2$. On montre alors ais ment qu'on a bien le r sultat escompt .

Ainsi, en radiosité, pour tout $\varepsilon > 0$, il existe $\delta(\varepsilon)$ tel que la projection de l'équation de radiosité dans une base d'ondelettes fournit une matrice d'interaction dont, au plus, $O(n \log n)$ coefficients sont supérieurs à $\delta(\varepsilon)$ et dont tous les autres coefficients peuvent être ignorés en causant une erreur inférieure à ε dans le résultat du calcul final. Ce résultat prouve l'efficacité théorique des ondelettes pour faire une approximation de la fonction de radiosité.

Ce sont Peter SCHRÖDER et Steven J. GORTLER qui ont introduit les ondelettes pour la radiosité dans [Sch 94a, Gor 93]. Les fonctions ondelettes permettent d'associer tous les avantages des méthodes citées précédemment : elles sont hiérarchiques, elles sont d'ordre quelconque et elles permettent de maintenir facilement la cohérence entre les niveaux de la hiérarchie grâce à des coefficients de passage qui ne dépendent pas du niveau de précision. Ces coefficients de passage, nommés coefficients de *push-pull*, permettent de faire passer les informations d'un niveau donné à un niveau voisin. Les coefficients de *push* permettent de transférer exactement la représentation fonctionnelle au niveau inférieur. Alors que les coefficients de *pull* permettent de transférer l'information au niveau supérieur en utilisant une approximation qui maintient la quantité d'énergie constante.

La construction d'une base d'ondelettes part de la base de fonction de niveau le plus faible et s'agrandit en utilisant les fonctions obtenues par translations et par homothéties de cette base. Tous les niveaux de précision peuvent ainsi être engendrés dynamiquement car ils correspondent toujours à la base de premier niveau avec une remise à l'échelle. On évite ainsi de stocker un trop grand nombre de représentations fonctionnelles et on peut donc utiliser les mêmes calculs pour échanger les informations entre deux niveaux successifs quel que soit l'endroit où on se trouve dans la hiérarchie.

La plus simple des bases d'ondelettes est la base de HAAR, elle correspond exactement au cas de la radiosité hiérarchique classique où toutes les fonctions sont constantes par morceaux.

Il existe un très grand nombre de bases d'ondelettes et nous montrons dans le chapitre 2 (p. 41) que ces fonctions sont parfaitement adaptées au calcul de radiosité et qu'elles apportent réellement un gain en temps et en place lors du processus de résolution. Nous montrons que les ondelettes d'ordre élevé sont les meilleures fonctions connues actuellement pour effectuer un calcul de grande précision sur des scènes de grande complexité. Nos résultats expérimentaux permettent notamment de confirmer les très bons résultats théoriques présentés par Peter SCHRÖDER et Steven J. GORTLER. Ces résultats théoriques n'avaient jamais été testés avec succès. Nos expérimentations sont aussi présentés dans [Cun 00].

1.4.2.4 Calcul des coefficients $a(\phi_i, \phi_j)$ et $l(\phi_i)$

Pour pouvoir résoudre l'équation de radiance, il faut être capable d'effectuer le calcul des coefficients $a(\phi_i, \phi_j)$ et $l(\phi_i)$.

Ce calcul doit pouvoir être fait pour tout type de fonctions de base. Il faut donc qu'il soit de nature assez générale pour ne pas dépendre de leur type.

Une façon usuelle de faire une approximation des intégrales intervenant dans ces coefficients est de les calculer par quadrature. Cette approche a l'avantage d'être complètement indépendante de la nature des fonctions de base. Cela permet d'introduire par la suite de nouvelles espèces de fonctions de base sans devoir changer les calculs. L'autre avantage est de pouvoir calculer ces coefficients entre des bases de types différents.

Chapitre 2

Algorithme de radiosité à base d'ondelettes

« Dans la conception axiomatique, la mathématique apparaît en somme comme un réservoir de formes abstraites, les structures mathématiques; et il se trouve — sans qu'on sache bien pourquoi — que certains aspects de la réalité expérimentale viennent se mouler en certaines de ces formes, comme par une sorte de préadaptation. Il n'est pas niable, bien entendu, que la plupart de ces formes avaient à l'origine un contenu intuitif bien déterminé; mais c'est précisément en les vidant volontairement de ce contenu qu'on a su leur donner toute l'efficacité qu'elles portaient en puissance, et qu'on les a rendues susceptibles de recevoir des interprétations nouvelles et de remplir pleinement leur rôle élaborateur. »

Nicolas BOURBAKI, *Éléments de mathématique*

Sommaire

2.1	Introduction	42
2.2	État de l'art	43
2.2.1	Oracle de subdivision	44
2.2.2	Calcul de la visibilité	48
2.2.3	Mémoire utilisée	51
2.3	Rendre efficace l'algorithme de radiosité à base d'ondelettes	51
2.3.1	Oracle de subdivision	52
2.3.2	Intégrer la visibilité	57
2.3.3	Algorithmes de radiosité sans liens d'interaction	60
2.4	Bases d'ondelettes	62
2.4.1	Ondelettes pour la méthode de GALERKIN	62
2.4.2	Bases pour la méthode de collocation de points	64
2.5	Comparaison de différentes bases	67
2.5.1	Protocole expérimental	68
2.5.2	Résultats	74
2.5.3	Discussion et comparaison de nos résultats avec ceux des études précédentes	80
2.6	Conclusion	80

Résumé

Depuis que les ondelettes ont été introduites dans les algorithmes de radiosité [Sch 94a, Gor 93, Sch 94b, Gor 95] il est étonnant de remarquer que très peu de recherches ont été faites pour étudier l'influence que peut avoir l'ordre des fonctions utilisées sur le temps de calcul et sur la qualité du résultat obtenu. Une étude [Wil 97b, Wil 97a] menée par Andrew J. WILLMOTT et Paul S. HECKBERT en 1997 a montré que calculer la fonction de radiosité en utilisant des fonctions ondelettes et plus spécialement en utilisant des ondelettes d'ordre élevé n'apportait pas de réelle amélioration à la méthode de radiosité hiérarchique. Ils ont aussi montré que l'utilisation des ondelettes engendrait un coût en place mémoire très important, les rendant totalement inefficaces lors des simulations sur des scènes de grande taille. Dans ce chapitre nous remettons en cause ce résultat en présentant une nouvelle façon d'implanter les fonctions ondelettes dans un algorithme de radiosité : notre implantation diffère des autres approches par plusieurs points clefs tels que l'oracle de subdivision, le calcul de la fonction de visibilité, le stockage des liens en mémoire ou encore l'algorithme de résolution. Nous montrons qu'avec cette implantation les fonctions ondelettes d'ordre élevé apportent un réel gain par rapport à l'approche hiérarchique classique et par rapport aux fonctions d'ordre moins élevé.

2.1 Introduction

Comme nous l'avons déjà vu au chapitre précédent, on calcule en général la fonction de radiosité en utilisant des méthodes d'approximation par éléments finis. Les plus efficaces de ces méthodes sont les approches hiérarchiques qui utilisent une représentation à plusieurs niveaux de détail de la fonction de radiosité pour réduire la complexité de l'algorithme. Dans leur version classique, ces approches utilisent des fonctions de base constantes mais elles ont récemment été enrichies par l'utilisation des fonctions ondelettes qui permettent de combiner l'aspect multi-résolution avec l'utilisation de fonctions d'ordre quelconque. La plus simple des bases d'ondelettes est la base de HAAR qui correspond exactement aux fonctions de bases constantes (voir la section 1.4.2.3 (p. 37)).

En théorie, plus l'ordre des fonctions utilisées est élevé, plus la représentation de la fonction de radiosité est compacte. En conséquence les fonctions d'ordre élevé utilisent moins de mémoire et procurent une représentation plus continue de la fonction de radiosité, ce qui assure un meilleur résultat visuel final. Les fonctions ondelettes d'ordre élevé devraient normalement être le meilleur choix pour faire des simulations de radiosité.

En pratique, la mémoire nécessaire pour stocker les valeurs d'interaction entre les objets de la scène croît aussi vite que la puissance quatrième de l'ordre des fonctions utilisées. Le nombre de ces valeurs est rapidement trop important pour que des fonctions ondelettes complexes puissent réellement être utilisées. De plus, la base de HAAR permet de faire beaucoup de simplifications grâce à son lien très prononcé avec la géométrie sur laquelle elle s'appuie (il y a équivalence entre l'aire du support d'une fonction de base constante et l'énergie qu'elle représente). Si ces optimisations ne sont pas retravaillées pour s'adapter aux ondelettes d'ordre plus élevé, elles peuvent totalement inhiber le pouvoir

d'approximation de ces dernières et les rendre complètement inefficaces. La seule étude expérimentale qui ait été menée sur la comparaison des ondelettes [Wil 97b, Wil 97a] a conclu que les problèmes pratiques accompagnant l'utilisation des ondelettes d'ordre élevé surpassaient largement le gain théorique que ces dernières étaient supposées apporter à un algorithme de radiosité hiérarchique.

Cependant, ces problèmes pratiques ne sont pas inhérents aux ondelettes d'ordre élevé, mais seulement à leur implantation dans la méthode de radiosité. Dans ce chapitre, nous présentons une nouvelle approche pour traiter les fonctions ondelettes d'ordre élevé. Cette nouvelle approche diffère des précédentes à différents niveaux importants de l'algorithme de radiosité. Les parties auxquelles nous avons porté notre attention vont du calcul de la visibilité à l'algorithme de résolution en passant par l'oracle de subdivision et le stockage des liens d'interaction. Nous avons développé et implanté notre approche en gardant toujours en tête que nous voulions travailler avec des fonctions d'ordre quelconque. Avec cette implantation nous avons prouvé que les avantages théoriques des ondelettes d'ordre élevé prenaient effectivement le pas sur les problèmes pratiques qui jusque-là les avaient rendues inutilisables et inutilisées. Grâce à cette approche les ondelettes d'ordre élevé peuvent fournir une meilleure approximation de la fonction de radiosité, même à proximité des limites d'ombres dues aux discontinuités de la fonction de visibilité, et ce avec une convergence plus rapide qu'avec des fonctions de base constantes. De plus, elles nécessitent finalement moins de place en mémoire que les autres fonctions.

Le plan du chapitre est le suivant : dans une première section 2.2 (p. 43) nous redonnons les solutions qui étaient utilisées jusqu'à présent pour résoudre les points clefs dont nous venons de parler. Dans la section 2.3 (p. 51) nous présentons notre approche en précisant les modifications que nous avons introduites au niveau de ces mêmes points clefs.

Le principal résultat que nous voulons présenter ici est l'étude expérimentale que nous avons menée sur la comparaison des différentes bases d'ondelettes en utilisant notre implantation de l'algorithme de radiosité. La section 2.5 (p. 67), nous permet de présenter cette expérimentation et d'en donner les résultats qualitatifs et quantitatifs. Nous montrons notamment que les fonctions ondelettes d'ordre élevé permettent de converger plus rapidement vers une solution plus précise et de meilleure qualité tout en utilisant moins de mémoire que les fonctions d'ordre moins élevé.

2.2 État de l'art

Comme nous l'avons vu dans la section 1.2.4 (p. 19) les algorithmes de radiosité résolvent l'équation d'illumination globale en se restreignant aux surfaces idéalement diffuses et en considérant que le milieu ambiant n'intervient pas dans les calculs.

Pour calculer la fonction de radiosité nous utilisons une approche par éléments finis. On projette la fonction inconnue $B(x)$ dans un espace fonctionnel de dimension finie :

$$\tilde{B}(x) = \sum_i \alpha_i \phi_i(x)$$

On cherche dans l'espace des fonctions ϕ_i la meilleure approximation de la fonction de radiosité. Pour augmenter la précision des résultats, il faut trouver la base de fonctions

optimale. L'utilisation des ondelettes (voir 1.4.2.3 (p. 37)) permet de bénéficier d'une représentation avec plusieurs niveaux de détail de la fonction de radiosité sur chaque objet de la scène. Cette représentation à plusieurs niveaux de détail permet de passer facilement d'une précision à l'autre en fonction des besoins du moment.

Il y a deux grandes classes d'« algorithmes itératifs » pour la résolution de l'équation de radiosité : les algorithmes de GAUSS-SEIDEL connus sous le nom de *gathering* et les algorithmes de SOUTHWELL ou *shooting*. Ces algorithmes sont décrits plus précisément dans la section 1.4 (p. 31). Dans la méthode de *gathering* chaque surface met à jour sa fonction de radiosité en utilisant l'énergie qu'elle reçoit des autres surfaces de la scène alors que pour le *shooting* chaque surface envoie de l'énergie dans la scène et les autres surfaces remettent immédiatement leur fonction de radiosité à jour. Dans les deux cas, le transport de l'énergie se fait par des liens d'interaction entre les fonctions ondelettes de chaque surface. En général, dans les algorithmes de radiosité à base d'ondelettes, ces liens sont calculés puis stockés pour sauvegarder les informations relatives aux interactions.

2.2.1 Oracle de subdivision

L'oracle de subdivision est au cœur des algorithmes de radiosité hiérarchiques car c'est lui qui détermine à quels niveaux de précision les fonctions ondelettes doivent être considérées pour calculer et stocker les liens d'interaction. En général, l'oracle a donc une grande influence à la fois sur la qualité de la solution et sur le temps passé à générer cette solution.

En théorie, la décision de subdiviser ou de ne pas subdiviser une interaction ne devrait être prise qu'avec une connaissance globale de la solution finale. L'oracle ne dispose que d'informations restreintes et très locales sur la solution : l'énergie qui doit être propagée depuis l'émetteur et la configuration géométrique de l'émetteur et du récepteur. Il doit cependant permettre de prendre une décision avec ces seules informations.

2.2.1.1 Coefficients d'interaction

Étant données deux mailles dans la scène, considérons leur interaction : la maille émettrice e , avec son approximation courante de la fonction de radiosité $\tilde{B}_e(y)$, envoie de l'énergie vers la maille réceptrice r . En utilisant une combinaison des équations 1.4 et 1.10 on peut exprimer la contribution de l'émetteur à la fonction de radiosité du récepteur par une équation intégrale :

$$B_{e \rightarrow r}(x) = \rho \sum_i \alpha_i \int_e \phi_i(y) K(x, y) dy \quad (2.1)$$

Pour évaluer l'interaction entre les deux mailles, on est donc amené à calculer des coefficients d'interaction C_{ij} pour traduire l'interaction entre les fonctions ϕ_i et ϕ_j respec-

tivement sur e et sur r . Ces coefficients sont définis de la façon suivante :

$$\begin{aligned}
 B_{e \rightarrow r}(x) &= \sum_j \beta_j \phi_j(x) \\
 \beta_j &= \int_r B_{e \rightarrow r}(x) \phi_j(x) \\
 \beta_j &= \rho \sum_i \alpha_i \int_r \int_e \phi_i(y) \phi_j(x) K(x,y) dy dx \\
 \beta_j &= \rho \sum_i \alpha_i C_{ij} \\
 C_{ij} &= \int_r \int_e \phi_i(y) \phi_j(x) K(x,y) dy dx
 \end{aligned}$$

L'évaluation du coefficient C_{ij} nécessite le calcul d'une intégrale complexe pour laquelle il n'existe généralement pas de solution analytique. Nous avons vu dans la section 1.1 (p. 11) qu'il y a deux grandes approches pour calculer l'interaction entre deux fonctions de base : l'approche par collocation de points qui consiste à minimiser l'erreur de propagation en un certain nombre de points x_j définis pour chacune des fonctions réceptrices et l'approche de GALERKIN qui consiste à minimiser l'erreur en moyenne sur tout le support de chaque fonction réceptrice. Avec les fonctions pour la collocation que nous utilisons dans notre implantation (voir la section 2.5 (p. 67)) les coefficients C_{ij} peuvent être remplacés par : $C_{ij} = \int_e \phi_i(y) K(x_j, y) dy$ ce qui ne laisse plus qu'une dimension à l'intégrale et diminue la complexité des calculs. Pour l'approche de GALERKIN on est obligé de conserver les deux dimensions à l'intégrale et de l'évaluer directement. La plupart du temps on effectue une approximation de l'intégrale en utilisant un calcul de quadrature.

2.2.1.2 Différents type d'oracles

La grande majorité des implantations de l'oracle de subdivision se base sur une estimation de l'erreur commise lors de l'approximation des coefficients C_{ij} . En fait on n'utilise pas directement cette erreur mais on la multiplie par l'énergie de l'émetteur afin de ne pas subdiviser des interactions qui ne transporterait que très peu d'énergie. Il y a différentes manières d'évaluer l'erreur commise lors du calcul des coefficients C_{ij} :

- **Erreur basée sur la valeur du coefficient** : la plus simple des approches est d'utiliser directement la valeur du coefficient C_{ij} et de dire que l'erreur est proportionnelle à cette valeur multipliée par l'énergie de l'émetteur. Cette approche est la première qui ait été proposée pour calculer la fonction oracle dans les algorithmes hiérarchiques. Elle a été décrite dès 1991 par Pat HANRAHAN *et al.* dans [Han 91].
- **Erreur basée sur un encadrement de la valeur des coefficients** : il est possible d'évaluer des bornes supérieure et inférieure à la valeur de l'interaction entre deux fonctions de base [Lis 94]. Bien que ces bornes ne procurent généralement que des valeurs très approchées et donc très pessimistes on peut utiliser la différence entre le maximum et le minimum pour calculer l'erreur commise sur le calcul des

coefficients d'interaction. Cette approche a été proposée par Marc STAMMINGER *et al.* dans [Sta 97a] et par Nicolas HOLZSCHUCH et François SILLION dans [Hol 98].

- **Erreur basée sur un calcul de l'interaction sur des points de test :** il est possible de vérifier la valeur du coefficient en calculant précisément l'interaction entre les fonctions de base en un certain nombre de points échantillons comme cela est exposé par S. GIBSON et R.J. HUBBOLD dans [Gib 96]. En la comparant à la valeur obtenue en utilisant le coefficient d'interaction. Cette implantation est celle exposée dans l'article de Steven J. GORTLER *et al.* [Gor 93].

Une variante de cette méthode a été proposée par Nicolas HOLZSCHUCH *et al.* dans [Hol 94]. Leur approche consiste à évaluer les coefficients à un niveau de subdivision supplémentaire puis à les comparer aux coefficients courants pour en déduire une erreur "a posteriori".

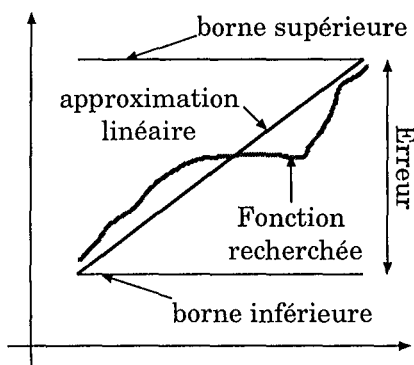


FIG. 2.1 – Calcul de l'erreur par majoration des coefficients

En étudiant les solutions exposées ci-dessus on remarque qu'un point commun à ces trois approches est qu'elles se basent toutes sur la valeur des coefficients C_{ij} pour évaluer la validité d'une interaction. Cette approche assure la convergence de l'algorithme vers une solution exacte mais elle est souvent trop exigeante. Elle entraîne généralement un maillage trop fin à des endroits où la fonction de radiosité que l'on vient de calculer procure, en fait, une très bonne approximation des variations de lumière. De plus, ce phénomène est d'autant plus pénalisant que les fonctions de base utilisées ont un fort pouvoir d'approximation. En effet, les coefficients C_{ij} sont généralement bornés par des valeurs constantes, c'est-à-dire qu'on encadre les fonctions interagissant sur chaque support par des fonctions constantes pour évaluer l'erreur commise lors du calcul. Or cet encadrement ne permet pas de repérer efficacement si les fonctions de base sont capables de donner une bonne approximation de la fonction recherchée grâce à leurs propres variations. La figure 2.1 montre l'approximation d'une fonction 2D par une fonction linéaire. La différence entre la borne supérieure et la borne inférieure de la fonction d'approximation définit l'erreur dont nous venons de parler. On peut constater sur la figure que cette erreur est beaucoup trop grande par rapport à l'erreur réellement commise. Cet exemple simple démontre à quel point un calcul peut rendre totalement inefficace les fonctions d'ordre élevé s'il ne prend pas correctement en compte leur pouvoir d'approximation. La représentation très compacte que l'on devrait obtenir lorsque l'on utilise des fonctions ondelettes d'ordre élevé pour représenter la fonction de radiosité devient donc une représentation très volumineuse et très peu efficace si on se base sur ce genre d'oracle. Pour profiter pleinement du pouvoir des ondelettes, il faut donc intégrer le calcul de la fonction de radiosité à l'intérieur de l'oracle de subdivision. Dans la section 2.3 (p. 51), nous présentons notre implantation de l'oracle de subdivision qui prend totalement en compte cette nouvelle considération.

2.2.1.3 Quelle grandeur physique veut-on contrôler ?

Nous venons de voir différentes façons de déterminer l'erreur commise lors de l'évaluation des coefficients d'interaction C_{ij} . Il faut maintenant décider sur quelle grandeur physique l'oracle de subdivision doit se baser pour décider s'il faut ou s'il ne faut pas subdiviser une interaction. Nous avons vu dans la section 1.3 (p. 22) que le but des algorithmes de radiosité est de minimiser la fonction résidu $r(x)$ avec :

$$r(x) = B(x) - \tilde{B}(x)$$

Chacun des trois différents oracles de subdivision présentés ci-dessus calcule une approximation ΔB de la valeur $\max(\|r(x)\|)$. Il est alors possible de choisir différentes valeurs de l'erreur que l'on veut tester selon le type de grandeur physique que l'on veut évaluer. Tous ces calculs d'erreur peuvent être interprétés comme le résultat d'un calcul de norme sur la fonction résidu. Mathématiquement la norme \mathcal{L}_p est définie de la façon suivante :

$$\mathcal{L}_p(f) = \left(\int_{\text{support}(f)} \|f(x)\|^p dx \right)^{1/p}$$

Trois mesures peuvent être intéressantes physiquement pour la fonction résidu :

- \mathcal{L}_∞ : ΔB peut être utilisée directement comme mesure de l'erreur. Cette mesure revient à considérer la norme \mathcal{L}_∞ de l'erreur qui évalue simplement la valeur maximum de $\|r\|$. Étant donné que l'unité de cette erreur est en $Watt/m^2$, cette erreur peut être vue comme une mesure de radiosité.
- \mathcal{L}_1 : la valeur $\Delta B \mathcal{A}$ (où \mathcal{A} est l'aire du récepteur) peut être vue comme une approximation de l'erreur pour la norme \mathcal{L}_1 . Avec une telle mesure de l'erreur on tolère une moins bonne approximation sur des surfaces de petite taille que sur des surfaces de grande taille. Une autre interprétation est de considérer qu'on ne contrôle plus la valeur de la radiosité transportée de l'émetteur vers le récepteur, mais qu'on contrôle maintenant la puissance énergétique transmise.
- \mathcal{L}_2 : pour utiliser la norme \mathcal{L}_2 il faut intégrer $r(x)^2$ sur la surface du récepteur. Si on considère que $|r(x)| \approx \Delta B$, l'intégrale de $r(x)^2$ équivaut grossièrement à $\Delta B^2 \mathcal{A}$. L'erreur ainsi mesurée est donc égale à $\Delta B^2 \sqrt{\mathcal{A}}$. $\sqrt{\mathcal{A}}$ est de l'ordre de grandeur de la distance maximum séparant deux points sur le récepteur. L'avantage de cette définition est donc que les surfaces allongées ont plus de chances d'être subdivisées que les surfaces dont la forme est proche d'un carré et dont l'aire est similaire. Or, il est connu qu'il est plus difficile de faire une approximation d'une fonction sur une surface allongée que sur un carré. Cette propriété peut donc être très utile pour améliorer l'efficacité de l'oracle.

2.2.1.4 Considération sur l'erreur en radiosité

Un point important doit être souligné et pris en compte lorsque l'on utilise la norme \mathcal{L}_∞ dans l'oracle de subdivision. En effet, dans le cas d'inter-réflexions entre deux surfaces qui forment un coin, l'évaluation du noyau¹⁰ devient délicate à cause de la décroissance en $1/r^2$

10. L'expression du noyau a été décrite dans la section 1.2 (p. 13).

où r représente la distance entre deux points des deux surfaces. Lorsque r tend vers 0 au niveau de l'intersection, l'approximation n'arrive plus à représenter fidèlement la fonction de radiosité. Dans cette configuration géométrique des mailles et avec l'oracle basé sur l'erreur en radiosité, une subdivision hiérarchique conserve les proportions géométriques. Il y a invariance d'échelle, et une erreur trop importante à un niveau le sera donc aussi à tous les niveaux inférieurs, de façon récursive.

La conséquence de ce problème est que dès que l'approximation n'est plus assez bonne dans un coin, les mailles seront subdivisées hiérarchiquement jusqu'à ce qu'un autre critère d'arrêt intervienne. Il faut donc introduire un nouveau critère d'arrêt dans ce genre d'oracle. Le plus naturel est d'utiliser une taille τ sous laquelle plus aucune subdivision n'est autorisée. Il en résulte une subdivision comme dans la figure 2.2, dont le nombre de mailles n'évolue pas régulièrement. La zone où la subdivision a atteint la taille minimum augmente régulièrement quand on demande à l'oracle d'augmenter sa précision en radiosité. On sait par contre que le seuil d'erreur en radiosité peut être dépassé dans cette zone.

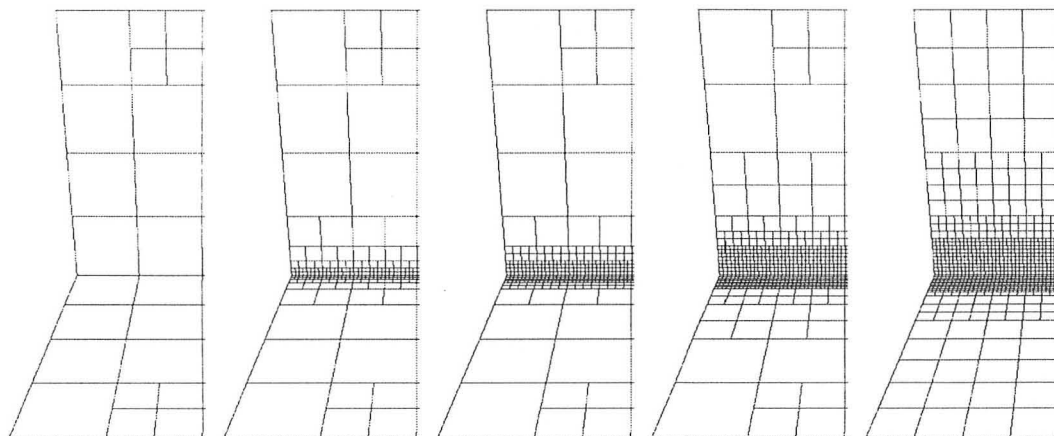


FIG. 2.2 – Évolution du maillage dans un coin en fonction de la précision.

2.2.2 Calcul de la visibilité

Paul HECKBERT a montré dans [Hec 92] que les discontinuités de la fonction de radiosité et de ses dérivées sont uniquement dues à des variations de la visibilité entre les objets de la scène. C'est pour cette raison qu'il faut porter une attention toute particulière à la façon dont on introduit les informations de visibilité dans l'algorithme de radiosité.

Comme nous l'avons vu dans la section précédente, on utilise des quadratures pour calculer les coefficients C_{ij} . Puisque les quadratures se basent sur un échantillonnage des supports des fonctions à intégrer, on est amené à évaluer un très grand nombre de fois la valeur du noyau $K(x,y)$. $K(x,y)$ décrivant le facteur d'interaction géométrique entre les points x et y , il faut connaître la valeur de la fonction de visibilité $V(x,y)$ entre ces deux points pour l'évaluer. Malheureusement, l'estimation de la visibilité est l'opération la plus coûteuse en temps de calcul dans l'algorithme de radiosité à base d'ondelettes.

La simple évaluation du noyau sans tenir compte de la visibilité est très nettement plus rapide que la même évaluation quand on y intègre le calcul de visibilité. Comme cela est expliqué par Nicolas HOLZSCHUCH *et al.* dans [Hol 94] le calcul de la visibilité peut prendre à lui seul plus de 70 % de la totalité du temps passé lors de la simulation. Plusieurs méthodes ont donc été développées pour fournir une estimation de la visibilité afin de réduire sensiblement le nombre de requêtes à effectuer. En général ces méthodes diminuent le nombre de requêtes en acceptant une certaine perte de fiabilité dans le résultat.

2.2.2.1 Approximation constante

La plus simple des accélérations qui ait été proposée consiste à considérer que la fonction de visibilité entre deux mailles a une valeur constante : $V(x,y) = c$. La constante c est égale à 1 quand on veut exprimer une visibilité totale entre les deux mailles, elle est égale à 0 si on veut considérer que les deux mailles ne se voient pas du tout et elle prend une valeur intermédiaire dans l'intervalle $]0,1[$ quand on cherche à exprimer une visibilité partielle entre les deux mailles. Cette méthode est celle qui a été utilisée par Pat HANRAHAN *et al.* dans leur algorithme de radiosité hiérarchique [Han 91] ainsi que par Steven J. GORLTER *et al.* pour leur implantation de l'algorithme de radiosité à base d'ondelettes [Gor 93]. Dans les deux cas la solution proposée pour estimer la valeur de la constante c est de lancer un certain nombre de rayons choisis de façon aléatoire entre les deux mailles, de calculer la visibilité sur chacun de ces rayons et d'en déduire une valeur moyenne de visibilité que l'on affecte à c .

Une amélioration possible de cette technique est de considérer les valeurs de la fonction de visibilité entre les sommets des mailles et d'en déduire une fonction de visibilité sur toute la surface des mailles par interpolation barycentrique. On n'a plus une valeur constante pour $V(x,y)$ mais une fonction linéaire qu'il faut ensuite intégrer dans le calcul du facteur de forme en la multipliant par la fonction purement géométrique. Cette technique n'a pas réellement été étudiée par le passé car elle ne représente un réel intérêt que si les fonctions de base utilisées sont au moins des fonctions linéaires pouvant rendre compte des variations de la fonction de visibilité. Or, la plupart des implantations actuelles de l'algorithme de radiosité utilisent des fonctions de base constantes ne pouvant pas intégrer des variations linéaires de la visibilité. Un tel calcul n'a donc pas d'intérêt dans les implantations classiques.

2.2.2.2 Masques de visibilité

L'utilisation de « masques de visibilité » a été proposée par Harold R. ZATZ dans [Zat 93] et suppose que lors du calcul de l'interaction entre deux mailles on sait créer une *bitmap* décrivant la visibilité sur la maille réceptrice. Les valeurs de la *bitmaps* varient dans l'intervalle $[0,1]$ et correspondent en chaque point de la maille réceptrice à la proportion de la maille émettrice visible depuis ce point. Si on sait créer une telle *bitmaps* on peut faire une approximation du calcul d'une interaction en sortant le terme de visibilité de l'intégrale et en utilisant à posteriori les valeurs stockées sur la *bitmaps* pour les multiplier par la fonction de radiosité évaluée sans le terme de visibilité. Un telle approche suppose que la fonction de radiosité a une valeur relativement constante sur la maille émettrice et que

les mailles interagissant sont quasiment parallèles. Sinon les erreurs dues au fait qu'on considère toujours la maille émettrice dans sa globalité pour évaluer la visibilité peuvent devenir très importantes comme le montre la figure 2.3. Sur cette figure, le masque de visibilité sur R est représenté par les couleurs grisée et noire (noire pour une valeur du masque nulle et grisée pour une valeur $\in]0,1[$). La zone D est noire car elle ne voit pas du tout E alors que la zone C est grisée car ses points voient tous au moins un point de la zone A . Supposons maintenant que seule la zone B soit émettrice sur E , en d'autres termes, la fonction de radiosité dans la zone A est nulle. Comme les points de E visibles depuis R sont tous situés dans la zone A , la fonction de radiosité transmise depuis E vers R devrait être nulle. Or si on utilise le masque de visibilité, on commence par évaluer la fonction transmise sans tenir compte de la visibilité. Cette fonction n'est pas nulle car la zone B contient de l'énergie à transmettre. En la multipliant par les valeurs du masque de visibilité on en déduit donc un transport énergétique non nul dans la zone C . Une telle erreur peut s'avérer très gênante car l'illumination d'une zone théoriquement obscure se détecte immédiatement et crée un artefact visuel qu'on ne peut pas faire disparaître facilement par la suite.

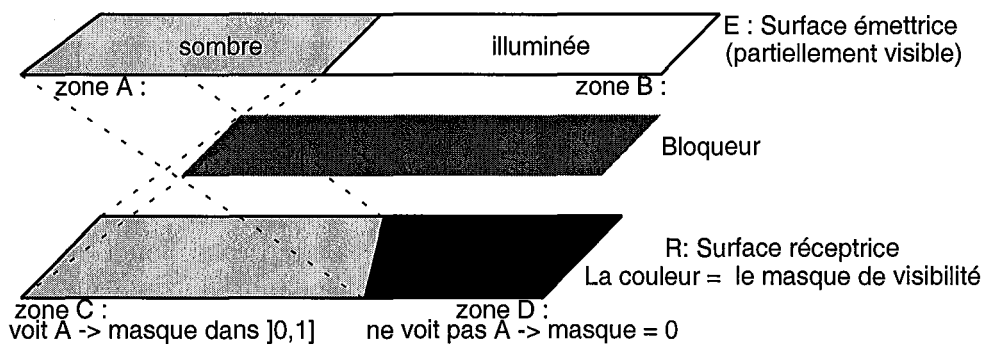


FIG. 2.3 – Effet de l'utilisation des masques de visibilité.

Dans la pratique, il est relativement difficile d'implanter les masques de visibilité. De plus, les problèmes décrits ci-dessus ne les rendent efficaces que pour des cas particuliers (source lumineuse surfacique située au plafond d'une pièce dont les principaux objets se trouvent sur le sol ...). Pour ces raisons, les masques de visibilité existent surtout au niveau théorique mais ne sont pas utilisés dans des implantations réelles.

2.2.2.3 Récapitulatif

Les méthodes que nous venons de présenter cherchent toutes à faire une approximation de la visibilité pour réduire le nombre de requêtes de visibilité par rapport au nombre total d'évaluation du noyau. Malheureusement ces approximations engendrent toujours de gros artefacts dans les solutions finales et génèrent des fonctions de radiosité assez peu précises particulièrement aux limites de visibilité comme l'ont montré Andrew WILLMOTT et Paul HECKBERT dans [Wil 97a]. Dans la section 2.3.2 (p. 57) nous présentons notre approche pour calculer la visibilité de façon rapide tout en gardant une réelle fiabilité dans le résultat.

2.2.3 Mémoire utilisée

L'évaluation des coefficients C_{ij} est assurément la partie la plus consommatrice en temps de calcul dans l'algorithme de radiosité. En effet, ce calcul suppose l'approximation d'une intégrale au moyen de quadratures faisant intervenir le terme de visibilité. Pour réduire l'impact de ce terme sur le temps de l'algorithme, la majorité des implantations stockent les coefficients C_{ij} en mémoire quand ils sont calculés pour la première fois et les réutilisent à chaque fois qu'ils sont nécessaires au cours du calcul. Ce stockage est fait sous la forme de liens d'interaction entre chaque couple de surfaces e et r .

Un problème important drainé par les implantations antérieures de l'algorithme de radiosité à base d'ondelettes est la quantité de mémoire nécessaire pour conserver tous ces liens tout au long du calcul. En effet, si on considère une base d'ondelettes d'ordre m , alors il y a m fonctions dans cette base. En dimension deux (tel que c'est le cas pour des fonctions surfaciques à appliquer dans des scènes de radiosité) il y a donc m^2 fonctions dans la base d'ondelettes. Comme il faut 1 coefficient C_{ij} pour chaque interaction entre deux fonctions de base, le nombre de coefficients C_{ij} nécessaires pour décrire totalement l'interaction entre les deux mailles e et r est égal au produit du nombre de fonctions de base sur e et du nombre de fonctions de base sur r . Par conséquent, si l'ordre des fonctions de base sur e et sur r est m , il faut calculer et conserver en mémoire m^4 coefficients C_{ij} pour stocker l'interaction entre ces deux mailles.

Il découle de cela que la mémoire nécessaire dans un algorithme de radiosité à base d'ondelettes est proportionnelle à la puissance quatrième du moment des ondelettes utilisées. Des ondelettes de moment 3 vont donc occuper une place de deux ordres de grandeur supérieure à des ondelettes de moment 1. Dans leur étude expérimentale [Wil 97a] Andrew WILLMOTT et Paul HECKBERT ont conclu que la place nécessaire en mémoire rendait effectivement inutilisables les fonctions ondelettes d'ordre élevé pour des simulations sur des scènes de taille importante et notamment sur toutes les scènes issues d'applications industrielles.

Des algorithmes expliquant qu'il était possible de s'affranchir totalement du stockage des liens d'interaction dans un algorithme de radiosité hiérarchique ont été présentés par Slimane MERZOUK dans [Mer 97] puis par Marc STAMMINGER *et al.* dans [Sta 98] et par Christophe WINKLER dans [Win 98]. On fait ainsi disparaître le principal problème des ondelettes d'ordre élevé : la place occupée en mémoire. Nous en profitons pour étudier les conséquences et l'impact que peut avoir le fait de ne plus conserver de liens en mémoire sur l'efficacité de l'algorithme.

2.3 Rendre efficace l'algorithme de radiosité à base d'ondelettes

Comme nous venons de le voir, les études expérimentales qui ont été menées sur les algorithmes de radiosité à base d'ondelettes ont toutes conclu que les fonctions d'ordre élevé sont toujours moins efficaces que les ondelettes de HAAR. Afin d'expliquer comment nous obtenons des résultats tout à fait différents avec notre approche, nous présentons maintenant notre manière de traiter les ondelettes. Pour cela nous détaillons tous les

points clefs de l'algorithme dont l'implantation diffère chez nous de l'implantation dans les algorithmes classiques : l'oracle de subdivision, le calcul de la fonction de visibilité et le fait de ne pas conserver les liens d'interaction en mémoire au cours de l'algorithme. Nous présentons aussi les conséquences que ces changements peuvent avoir tout au long de l'algorithme.

Nous avons implanté et testé tous les algorithmes que nous décrivons dans cette section sur la plate-forme logicielle Candela [Win 98] de l'équipe ISA au LORIA.

2.3.1 Oracle de subdivision

Contrairement aux approches présentées dans la section précédente, nous ne basons pas l'oracle de subdivision sur l'erreur commise en calculant les coefficients de propagation C_{ij} . Nous évaluons directement l'erreur commise sur la fonction de radiosité transmise. Notre implantation de l'oracle est relativement similaire à celle présentée par Philippe BEKAERT et Yves D. WILLEMS dans [Bek 96a, Bek 96b].

2.3.1.1 Points de contrôle

Pour évaluer l'erreur que l'on commet en calculant la fonction de radiosité, nous utilisons des points de contrôle sur la maille réceptrice. L'emplacement de ces points a été soigneusement choisi, de façon à fournir une information aussi pertinente que possible pour le calcul de radiosité. Ce choix a été fait suivant plusieurs critères :

- **La méthode d'approximation et les fonctions de base :** les points de contrôle ne sont pas les mêmes en GALERKIN et en collocation de points. En collocation de points, on choisit des points de contrôle différents de ceux de collocation. En effet, on sait déjà que l'approximation est optimale sur ces derniers. En GALERKIN les points de contrôle doivent être différents des points de quadrature pour la même raison.

Les valeurs aux coins de la maille réceptrice sont très représentatives de la qualité du résultat. Comme les points de quadrature dans la méthode de GALERKIN ne sont jamais confondus avec les coins de la maille réceptrice, on utilise toujours ces derniers comme points de contrôle. Quand cela est possible, on y ajoute le centre de la maille. Malheureusement ces points font généralement partie des points de collocation. Dans la méthode de collocation, il faut donc à chaque fois prendre en compte le type d'ondelettes utilisées pour choisir les points de contrôle. Les différentes ondelettes pour la collocation sont décrites dans la section 2.5 (p. 67). Pour chacune d'elles nous donnons les points de collocation et les points de contrôle correspondants.

- **La taille du récepteur :** pour prendre en compte la taille du récepteur nous avons introduit la possibilité d'ajouter une grille régulière de points de contrôle. Deux paramètres permettent de gérer la taille de cette grille : un nombre maximum de points suivant chaque direction et une longueur minimale entre deux points. On utilise toujours le critère générant le moins de points pour ne pas trop augmenter le temps de calcul.

- **la forme du récepteur** : des points de contrôle supplémentaires peuvent être ajoutés dans le cas de supports complexes. Pour des supports non plans les points de contrôle classiques peuvent s'avérer insuffisants pour tester toutes les directions vues par la surface. Sur la figure 2.4 on voit que des points qui sont pertinents sur un carré ne le sont plus du tout sur un cylindre. Dans le chapitre 3 (p. 83) nous décrivons comment nous gérons certaines surfaces complexes. Nous présentons alors comment nous avons fait pour ajouter automatiquement des points de contrôle aux endroits où cela s'avère nécessaire.

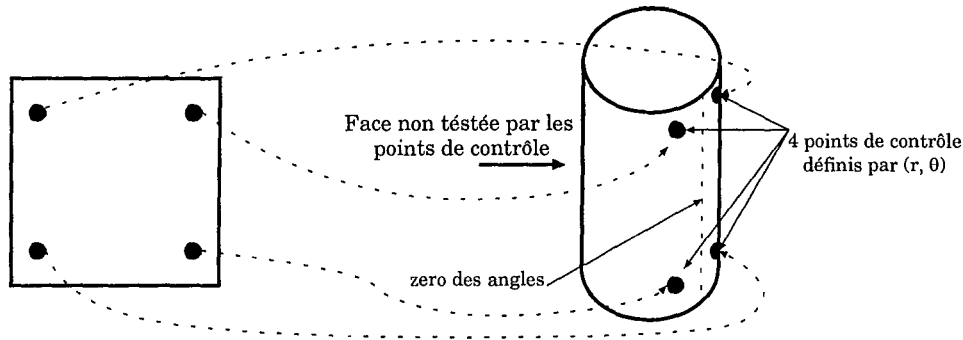


FIG. 2.4 – *Passage des points de contrôles sur un carré aux points de contrôle correspondants sur un cylindre.*

2.3.1.2 Calcul de l'erreur

Une fois la grille de points de contrôle choisie, on procède au calcul de l'erreur locale. Pour cela nous évaluons deux valeurs de la radiosité en chacun de ces points.

La première valeur est calculée en utilisant les coefficients $C_{i,j}$ (définis dans la section 2.2.1 (p. 44)) qui sont les coefficients d'interaction entre les fonctions ϕ_i et ϕ_j respectivement sur la maille émettrice e et sur la maille réceptrice r . Définissons maintenant les coefficients \tilde{C}_{ij} comme l'approximation numérique des coefficients $C_{i,j}$. Cette approximation est calculée au moyen de quadratures. L'approximation numérique de la radiosité émise depuis e vers le point x de r est égale à :

$$\tilde{B}_{e \rightarrow r}(x) = \rho \sum_j \left(\sum_i \tilde{C}_{i,j} \alpha_i \right) \phi_j(x)$$

Cette valeur de radiosité représente l'approximation de la radiosité que l'on transmettrait depuis la maille e vers le point x de la maille r si on effectuait l'interaction à ces niveaux de précision.

La deuxième valeur est calculée en utilisant un calcul de quadrature sur l'émetteur. Si la fonction représentant la radiosité à émettre depuis la maille e au niveau courant de précision est égale à $\tilde{B}_e(y) = \sum_i \alpha_i \phi_i(y)$, on peut calculer directement une valeur de la radiosité transmise depuis e vers un point x de la maille réceptrice r de la façon suivante :

$$\tilde{B}_{e \rightarrow x} = \sum_k w_k K(x, y_k) \tilde{B}_e(y_k)$$

o   les y_k sont les points de quadrature sur la surface   mettrice et les w_k sont les poids associ  s. La valeur $\tilde{B}_{e \rightarrow x}$ peut   tre consid  r  e comme la valeur « exacte » de la radiosit   transmise vers le point x depuis la surface e .

Pour effectuer le calcul de l'erreur nous utilisons ensuite les valeurs du r  sidu

$$\tilde{r}(x) = |\tilde{B}_{e \rightarrow r}(x) - \tilde{B}_{e \rightarrow x}|$$

sur chaque point de contr  le. Dans le logiciel Candela, nous avons la possibilit   de calculer les diff  rentes normes $\mathcal{L}_1, \mathcal{L}_2$ et \mathcal{L}_∞    partir de ces valeurs du r  sidu. Nos essais avec ces diff  rentes normes nous ont conduit aux m  mes conclusions que Marc STAMMINGER *et al.* dans [Sta 97b] : elles ont quasiment la m  me efficacit   mais un probl  me se pose avec la norme \mathcal{L}_∞ au niveau des intersections entre deux surfaces. Il est n  cessaire, pour cette norme, de d  finir une taille de maille minimale    ne pas d  passer lors des subdivisions.

2.3.1.3 Prise en compte du niveau de pr  cision courant pour la fonction   mettrice

Dans un algorithme hi  rarchique la fonction   mettrice existe    plusieurs niveaux de pr  cision. L'oracle de subdivision utilise le niveau de pr  cision courant pour faire son calcul d'erreur. Pour encore affiner la pr  cision de l'oracle, nous avons cherch      prendre en compte l'erreur due au fait qu'on n'utilise pas toujours la repr  sentation la plus pr  cise de la fonction qui   met lors du calcul d'une interaction.

Notre am  lioration consiste    pond  rer l'erreur   valu  e sur la grille de test au moyen de la diff  rence d'  nergie qui existe entre la repr  sentation de la fonction au niveau courant et celle au niveau le plus d  taill   :

$$d = \int_{\text{support}} |\tilde{B}_{\text{courant}}(x) - \tilde{B}_{\text{d  taill  }}(x)| dx$$

On peut   valuer cette diff  rence de fa  on exacte car nous avons une repr  sentation fonctionnelle de \tilde{B}    tous les niveaux de d  tail gr  ce aux bases d'ondelettes. Cette diff  rence d'  nergie divis  e par l'  nergie du niveau courant donne un nombre D qui tend vers 0 quand la fonction au niveau courant se rapproche de la fonction au niveau le plus d  taill  . Pour pond  rer l'erreur, on la multiplie par $1 + D$.

La figure 2.5 montre l'effet d'une telle pond  ration. La grande surface verticale du fond n'est   clair  e que par la r  flexion de la lumi  re sur la surface horizontale en bas de l'image. La lumi  re sur cette derni  re subit de fortes variations    cause de la surface rouge qui occulte en partie la source lumineuse. On voit sur la figure 2.5(a) que la fonction   mettrice a   t   moins bien prise en compte que dans la figure 2.5(b) o   nous avons utilis   notre am  lioration.

Les erreurs sur la figure 2.5(a) (et moins fortement sur la figure 2.5(b)) sont principalement dues au fait que les points de quadrature tombent de fa  on d  sordonn  e dans des zones sombres ou   clair  es de la surface du bas. Les coefficients C_{ij} ainsi calcul  s ne peuvent pas assurer la continuit   de la fonction sur le r  cepteur. La seule fa  on pour corriger efficacement ces erreurs de discr  tisation est d'affiner le calcul aux endroits o   cela s'av  re n  cessaire. Notre implantation est une tr  s bonne heuristique pour d  celer ces endroits.

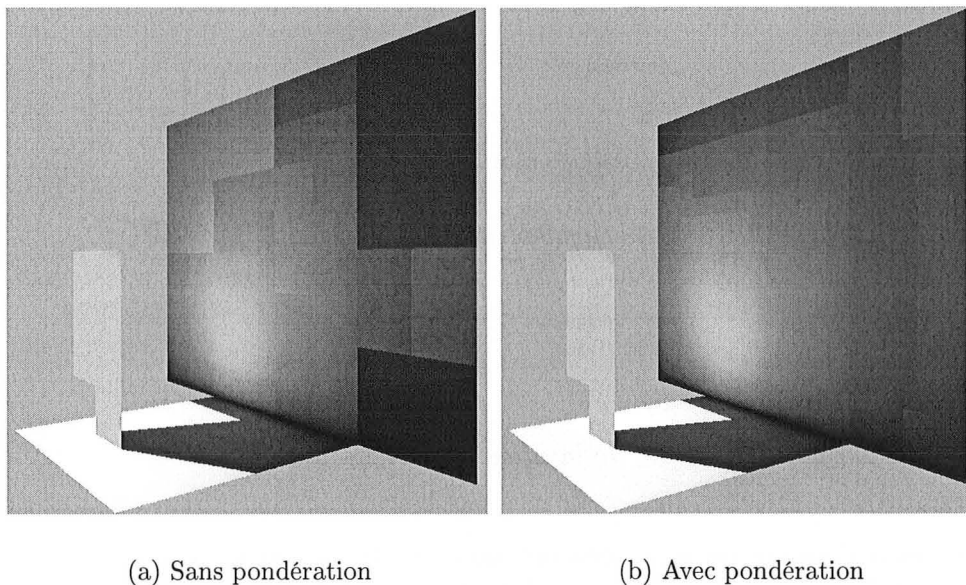


FIG. 2.5 – *Effet de la prise en compte de la validité de la fonction émettrice.*

Notre amélioration engendre plus de calculs puisqu'elle tend à augmenter la valeur de l'erreur que l'on teste. Sur l'exemple précédent, le nombre de propagations nécessaires pour le calcul avec pondération est de 30 % supérieur au nombre de propagations pour le calcul sans pondération. Le temps de calcul avec pondération est lui aussi de 30 % supérieur au temps de calcul sans pondération. Cependant d'après notre expérience, cette amélioration permet de concentrer les calculs là où il y en a le plus besoin. Nous avons constaté que le gain en qualité est plus important avec cette amélioration que quand on impose de façon générale un critère plus sévère à l'oracle de subdivision. Il est impossible de mesurer numériquement l'avantage d'une telle amélioration mais elle a donné des résultats de meilleure qualité en un temps plus rapide sur la plupart de nos tests.

2.3.1.4 Choix de la maille à subdiviser

Nous avons aussi travaillé sur le choix de la maille à subdiviser dans le cas où l'oracle décide de ne pas propager la radiosité.

Généralement, quand l'oracle conclut à la nécessité de subdiviser une interaction, on choisit la maille de plus grande aire pour la subdiviser en premier. Nous avons remarqué que ce processus pouvait pénaliser l'algorithme de radiosité et que le choix de la maille la plus grande n'était pas toujours le meilleur. En effet, prenons l'exemple très simple de la figure 2.6. Si on effectue un changement d'échelle sur la surface émettrice e il est naturel que l'illumination sur la surface réceptrice r reste la même. Si on utilise la méthode que l'on vient de décrire pour choisir la maille à subdiviser lors du calcul de l'interaction entre e et r , on n'obtiendra pas le même résultat dans le cas d'un émetteur de petite taille (figure 2.6(a)) et dans le cas d'un émetteur de grande taille (figure 2.6(b)). Dans ce dernier cas l'émetteur devra être beaucoup subdivisé avant que l'algorithme ne permette d'atteindre une subdivision adéquate sur le récepteur. En reprenant cet exemple avec

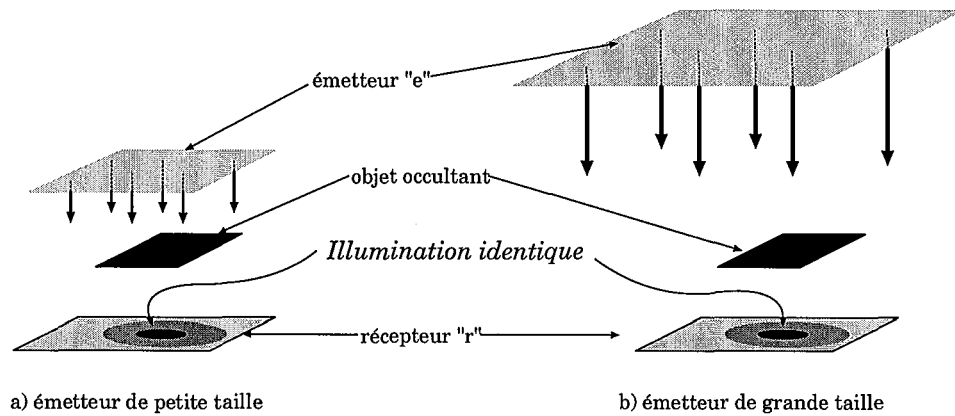


FIG. 2.6 – Effet de la mise à l'échelle d'un émetteur

d'autres mises à l'échelle on peut déduire qu'il existe une configuration optimale pour laquelle les subdivisions sur e et r sont les plus efficaces pour calculer l'interaction. Il ressort de cet exemple que ce n'est pas toujours la maille de plus grande aire qu'il faut subdiviser en premier.

Afin d'éviter ce genre de subdivisions systématiques nous avons introduit un poids aléatoire pour pondérer les aires des deux mailles qui interagissent. Le choix se fait en utilisant l'algorithme 4.

Algorithme 4 Choix de la maille à subdiviser

ENTRÉES: Maille e, r

SORTIES: Maille *résultat*

- 1: Réel $p =$ nombre au hasard $\in [0,1]$
 - 2: **si** $e.aire \times p > r.aire \times (1 - p)$ **alors**
 - 3: **retourne** e
 - 4: **sinon**
 - 5: **retourne** r
 - 6: **fin si**
-

Dans la pratique, cette amélioration a réduit le temps de calcul de 20 à 30 % sur toutes nos scènes de test.

2.3.1.5 Algorithme

L'algorithme qui correspond à notre implantation de l'oracle de subdivision est résumé dans ses grandes lignes sur l'algorithme 5. Le choix de la maille à subdiviser n'a pas été introduit dans cet algorithme pour simplifier la compréhension. En pratique, ce choix est généralement fait directement dans la fonction oracle. D'ailleurs nous l'avons introduit à cet endroit dans Candela en utilisant l'algorithme 4.

Algorithme 5 Oracle de subdivision**ENTRÉES:** Maille e, r **SORTIES:** Booléen *résultat*

- 1: Réel $\tilde{C}_{ij} = approximation(\int_r \int_e \phi_i(y)\phi_j(x)K(x,y)dydx)$
- 2: **pour tout** P_i points de contrôle **faire**
- 3: Réel $\tilde{B}_{e \rightarrow r}(P_i) = \rho \sum_j (\sum_i \tilde{C}_{i,j} \alpha_i) \phi_j(x)$
- 4: Réel $\tilde{B}_{e \rightarrow P_i} = \sum_k w_k K(P_i, y_k) \tilde{B}_e(y_k)$
- 5: Réel $\delta_i = |\tilde{B}_{e \rightarrow r}(P_i) - \tilde{B}_{e \rightarrow P_i}|$
- 6: **fin pour**
- 7: Réel $\mathcal{L}_n = \|\delta_i\|_n$
- 8: **si** $\mathcal{L}_n > \epsilon$ **alors**
- 9: **retourne** Faux
- 10: **sinon**
- 11: **retourne** Vrai
- 12: **fin si**

2.3.2 Intégrer la visibilité

2.3.2.1 Principe général

Notre façon de traiter la visibilité est fondamentalement différente des approches classiques antérieures. Nous calculons autant de requêtes de visibilité que nous en avons besoin sans chercher à faire une approximation globale de la fonction de visibilité. En fait, chaque fois que nous évaluons le noyau $K(x,y)$ entre deux points, nous calculons aussi la fonction de visibilité $V(x,y)$ entre ces deux points. Nous utilisons ensuite cette valeur de la fonction de visibilité directement dans le calcul du noyau telle qu'elle apparaît dans l'équation initiale 1.4. Ainsi, si x et y ne se voient pas, $V(x,y) = 0$ et on a aussi $K(x,y) = 0$.

En conséquence, la visibilité ne requiert pas de traitement spécial dans notre implantation. Les valeurs locales du noyau intègrent déjà toutes les informations de visibilité disponibles dans la scène.

Cette approche suppose que nous effectuons le calcul d'une requête de visibilité pour chaque évaluation du noyau $K(x,y)$. Le nombre de fois qu'il est nécessaire d'évaluer $K(x,y)$ est très important car ce noyau fournit une information locale entre deux points alors que l'algorithme de radiosité a besoin d'informations plus globales entre deux surfaces. L'utilisation de calculs de quadrature permet de passer efficacement de ces informations locales (ou « ponctuelles ») aux informations surfaciques nécessaires. Quand on utilise des fonctions ondelettes de moment m , on estime généralement qu'il faut m^2 points de quadrature pour passer convenablement des informations locales à l'information sur toute une maille. Dans le cas du calcul d'interaction entre deux mailles, le nombre de fois qu'il est nécessaire d'évaluer le noyau est donc de l'ordre de m^4 . Notre approche implique donc que l'on évalue en moyenne m^4 requêtes de visibilité pour chaque interaction entre deux mailles.

Sur des scènes de taille importante, une telle approche serait tout à fait impossible à envisager avec les algorithmes antérieurs de calcul de visibilité. Lancer des rayons à travers une scène de grande taille coûte très cher en temps de calcul même si on utilise des structures spatiales pour accélérer la traversée. Ainsi, envisager de lancer de l'ordre de m^4 rayons pour chaque interaction entre deux mailles de la scène est tout à fait impossible avec les capacités actuelles des ordinateurs.

Laurent ALONSO et Nicolas HOLZSCHUCH ont développé dans Candela un nouvel algorithme de visibilité qui permet de faire très rapidement énormément de requêtes entre deux surfaces [Alo 99]. Cet algorithme est une implantation hybride utilisant à la fois le matériel graphique et des heuristiques logicielles. Le matériel graphique (des cartes graphiques de hautes performances) est utilisé pour calculer plusieurs *bitmaps* qui permettent par interpolation de répondre à toutes les requêtes de visibilité extrêmement rapidement une fois qu'elles sont calculées. Les heuristiques logicielles permettent de vérifier la fiabilité du résultat renvoyé par les *bitmaps*. À chaque fois que ce résultat n'est pas certain, on revient à des méthodes de lancés de rayons pour connaître la visibilité. Notre algorithme de visibilité donne des résultats très fiables de façon extrêmement rapide. De plus, le goulot d'étranglement d'un tel algorithme n'est pas l'interrogation des *bitmaps* mais le calcul de ces dernières. Il est donc avantageux de les interroger un maximum une fois qu'elles sont créées.

Avec cette implantation du calcul de visibilité, le fait d'effectuer autant de requêtes de visibilité que de calculs du noyau est parfaitement envisageable. Nous avons pu tester notre algorithme sur des scènes de très grande taille (de l'ordre de 1 000 000 de polygones en entrée) sans que le calcul de visibilité soit un obstacle majeur.

Sur des scènes de petite taille (de l'ordre de quelques milliers de polygones en entrée) il est tout à fait possible d'utiliser un algorithme classique de visibilité pour répondre aux m^4 requêtes par interaction et notre algorithme s'applique donc sans problème.

2.3.2.2 Amélioration de l'efficacité pour l'oracle

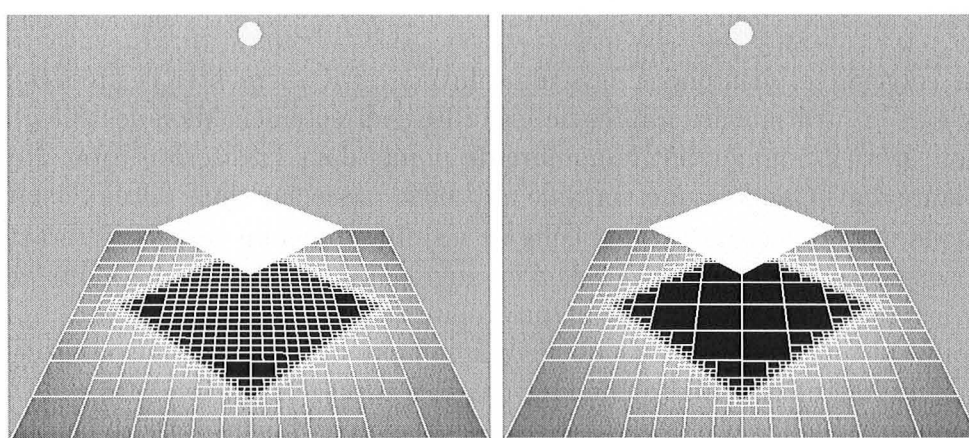
- **Un test sans visibilité:** lors du calcul de l'oracle, on doit évaluer les coefficients de transfert ainsi que des valeurs de transfert sur tous les points de test. Ces calculs représentent énormément d'évaluations du noyau et donc énormément d'évaluations de la visibilité. Malgré l'algorithme que nous venons de décrire pour accélérer les calculs de visibilité, il peut être intéressant de ne pas toujours prendre en compte la visibilité dans l'oracle de subdivision.

En effet, lors d'une interaction, la fonction de radiosité transmise peut présenter des variations qui ne sont pas seulement dues aux changements dans la fonction de visibilité. Ces variations viennent de la configuration géométrique entre les deux objets qui interagissent : les points les plus éloignés de la surface émettrice reçoivent moins d'énergie que les points les plus proches. Ces variations peuvent à elles seules engendrer la nécessité d'évaluer l'interaction à un niveau de subdivisions plus fin.

Pour permettre de ne pas calculer systématiquement la visibilité dans le calcul de l'oracle, nous pouvons dans notre implantation demander à l'oracle de toujours faire un premier test sans prendre en compte la visibilité. Si ce test conclut à la nécessité de subdiviser l'interaction, on effectue cette subdivision sans autre vérification. Par

contre, si le test ne conclut pas à la nécessité de subdiviser, on le refait en incluant la visibilité dans les calculs.

Bien que cet algorithme donne généralement de très bon résultats, il peut aussi pénaliser le calcul. En effet, considérons l'exemple de la figure 2.7(a) où une source de lumière ponctuelle puissante éclaire deux surfaces dont une cache une partie de l'autre. On voit sur la figure 2.7(a) que la surface du bas est très découpée dans une zone totalement non éclairée. Ce découpage est intervenu car l'oracle a conclu à la nécessité de subdiviser l'interaction avant de faire des tests de visibilité. Sur la figure 2.7(b), les tests de visibilité ont été effectués à chaque fois et l'oracle a ainsi permis de détecter la zone sombre où il n'a pas été nécessaire de subdiviser l'interaction.



(a) Sans visibilité

(b) Avec visibilité

FIG. 2.7 – *Maillage due à la prise en compte de la visibilité dans l'oracle.*

Il est impossible de donner une règle générale pour savoir s'il faut ou s'il ne faut pas utiliser cet algorithme pour accélérer les calculs. Cependant, notre expérience nous a montré que cet algorithme est toujours très efficace pour les interactions secondaires (après la propagation de l'énergie correspondant aux sources lumineuses). Pour le cas de l'illumination directe (propagation de l'énergie correspondant aux sources lumineuses), la seule solution pour savoir si on doit utiliser cette amélioration est de faire des tests ou de connaître a priori la configuration de la scène que l'on veut éclairer.

- **Augmentation du nombre de points de quadrature sur l'émetteur en cas de problèmes de visibilité.** Le calcul des coefficients \tilde{C}_{ij} est toujours le résultat de l'évaluation d'une intégrale non triviale. Pour faire une approximation numérique de ce genre d'intégrale, on est obligé de recourir à des techniques de discrétisation. Lorsque des problèmes de visibilité interfèrent dans le calcul des coefficients d'interaction, cette discrétisation peut générer des artefacts visuellement très gênants comme on peut le remarquer sur la figure 2.8(a). L'apparition de bandes lumineuses correspond au passage d'un point de la discrétisation d'une zone visible à une zone

invisible.

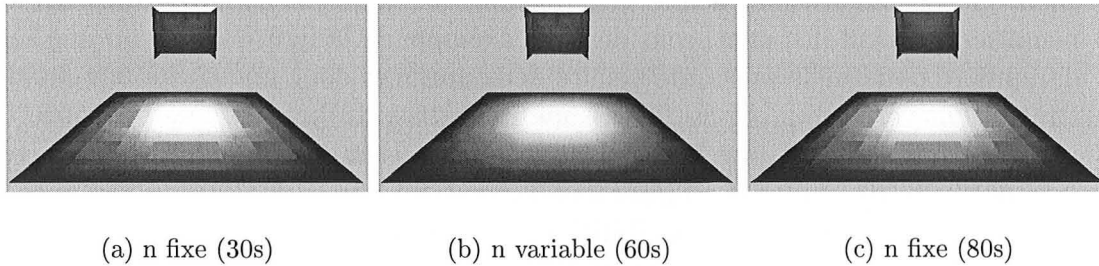


FIG. 2.8 – Effet de l'augmentation du nombre de points de quadrature (n) en cas d'erreur de visibilité.

Pour corriger ce phénomène, la seule solution est d'augmenter la précision des calculs, soit en utilisant les limites de visibilité¹¹ directement dans les calculs d'interaction, soit en augmentant le nombre de points dans la discrétisation. Nous avons implanté dans Candela une nouvelle approche basée sur cette dernière solution.

Lorsque nous détectons un problème de visibilité dans un calcul d'interaction nous refaisons automatiquement ce calcul en augmentant le nombre de points de quadrature sur l'émetteur. Nous considérons qu'un calcul d'interaction a un problème de visibilité si toutes les requêtes de visibilité, faites lors de l'évaluation de cette interaction ou lors de l'évaluation de la grille de test, ne sont pas identiques. Le nombre de points de quadrature que nous ajoutons peut être changé dans un fichier de configuration. Nous avons remarqué qu'en ajoutant le minimum, c'est-à-dire en ajoutant un point suivant chaque direction, on obtient des résultats tout à fait convaincants pour un minimum de calculs supplémentaires.

Cette heuristique augmente la quantité de travail à effectuer. Il est normal de remarquer sur les figures 2.8(a) et 2.8(b) que si on garde la même erreur locale sans et avec cette amélioration, le temps de calcul est plus long dans le dernier cas, mais la qualité est aussi nettement meilleure dans ce cas. De plus en diminuant l'erreur locale autorisée pour le calcul sans notre amélioration on remarque sur la figure 2.8(c) que le temps dépasse celui des deux cas précédents sans améliorer réellement le résultat visuel.

En conclusion, notre amélioration permet d'augmenter la précision des résultats, en affinant les calculs aux endroits où la solution en a le plus besoin.

2.3.3 Algorithmes de radiosité sans liens d'interaction

Nous avons vu dans cette section (2.2.3 (p. 51)) que la consommation mémoire était un des problèmes majeurs des algorithmes de radiosité à base d'ondelettes. Cette consommation mémoire est principalement due à la sauvegarde des liens d'interaction tout au long du calcul. Pour réduire l'impact de la consommation mémoire sur notre algorithme de

11. La détection des limites de visibilité correspond à un travail sur le maillage de discontinuité que nous effectuons actuellement au sein de l'équipe ISA.

radiosité, nous avons décidé de ne pas conserver les liens en mémoire au cours des calculs. Cette particularité de notre algorithme a été présentée et étudiée par Slimane MERZOUK et Christophe WINKLER dans leur thèse respective [Mer 97, Win 98].

Le fait de ne pas conserver les liens d'interaction en mémoire permet de faire l'équilibre entre le temps de calcul et la mémoire utilisée. En ne conservant pas les liens, on gagne de la place en mémoire, mais toute l'information qui n'est plus conservée en mémoire devra être recalculée à chaque fois qu'on en aura besoin au cours de l'algorithme. Dans leur article [Sta 98], Marc STAMMINGER *et al.* ont aussi présenté une approche pour ne pas conserver les liens d'interaction en mémoire dans un algorithme de radiosité hiérarchique. Ils ont démontré que l'impact sur le temps de calcul était négligeable par rapport au gain au niveau de la place gagnée en mémoire car les liens sont très rarement réutilisés au cours d'un algorithme de radiosité hiérarchique.

Notre implantation est étendue au cas des algorithmes de radiosité à base d'ondelettes. Le choix d'un tel algorithme a aussi des conséquences sur la structure de l'algorithme. La principale de ces conséquences réside dans le choix entre les méthodes de *gathering* et de *shooting* (voir 1.4 (p. 31)).

- La méthode de *gathering* calcule toute l'énergie transmise depuis chacune des surfaces de la scène vers toutes les autres à chaque itération de l'algorithme. Ainsi, tous les liens d'interaction sont utilisés au cours de chaque itération.
- La méthode de *shooting* calcule l'énergie qu'il reste à transmettre depuis la surface la plus énergétique (au sens de l'énergie qu'elle n'a pas encore transmise) vers toutes les autres surfaces de la scène. Au cours d'une itération donnée de l'algorithme, seuls les liens entre la surface émettrice et les autres surfaces de la scène sont utilisés. Après ce calcul, la surface émettrice est replacée en queue de la liste des surfaces qui doivent émettre et les liens ne seront réutilisés que lorsque cette surface sera revenue en tête de cette liste.

Les deux points précédents montrent que faire le choix de ne pas conserver les liens en mémoire est beaucoup plus cohérent si on choisit aussi d'utiliser un algorithme de *shooting* plutôt qu'un algorithme de *gathering*. Mais l'avantage cumulé du *shooting* et de la non conservation des liens ne s'arrête pas à ce simple constat sur les surfaces qui interagissent au cours d'une itération. Il existe aussi à un niveau plus fin dans la représentation hiérarchique des interactions.

- Avec la méthode de *gathering*, la distribution de l'énergie transportée le long des liens d'interaction ne varie pas beaucoup d'une itération à l'autre. En effet, l'énergie transportée d'un point à un autre de la scène n'augmente que grâce à l'apport énergétique reçu sur l'émetteur au cours des itérations précédentes. En conséquence, le niveau de subdivisions nécessaires pour calculer l'interaction est toujours au moins aussi fin que le niveau déjà calculé lors des itérations précédentes. Il est alors très intéressant de pouvoir repartir des liens précédents pour les réutiliser à cette itération en les affinant si nécessaire.
- Avec la méthode de *shooting*, l'énergie transportée le long des liens d'interaction n'est que l'énergie qui n'a pas encore été transmise au cours des itérations précédentes. Ainsi, la distribution de l'énergie transportée varie de façon très significative

d'une itération à l'autre pour une même surface émettrice. La part de l'énergie non encore transmise peut indifféremment être plus importante ou moins importante que lors d'une itération antérieure. En conséquence, il est impossible de prédire le niveau de subdivisions des liens qu'il sera nécessaire d'utiliser à un moment donné de l'algorithme. Les liens calculés lors des itérations antérieures ne peuvent pas être réutilisés simplement pour évaluer l'interaction courante.

Cette analyse montre que si on choisit d'utiliser un algorithme de *shooting* plutôt qu'un algorithme de *gathering* il est aussi plus intéressant de ne pas conserver les liens d'interaction en mémoire. En effet, l'information qui serait alors conservée en mémoire n'aurait que très peu de chance d'être réutilisée au cours de l'algorithme à part pour les premiers niveaux de subdivisions.

2.4 Bases d'ondelettes

Notre implantation de l'algorithme de radiosité à base d'ondelettes utilise tous les points détaillés dans la section précédente 2.3 (p. 51). Nous avons utilisé cette implantation pour faire une comparaison de différentes bases d'ondelettes

Les fonctions de base diffèrent suivant le mode de résolution de l'équation de radiosité. Nous disposons donc de deux familles de bases, une pour la méthode de GALERKIN et l'autre pour la méthode de collocation de points.

2.4.1 Ondelettes pour la méthode de GALERKIN

Pour la méthode de GALERKIN, nous avons décidé d'utiliser les bases d'ondelettes multiples (\mathcal{M}_n) présentées dans la version originale de la radiosité à base d'ondelettes [Gor 93].

Ces bases sont construites à partir des polynômes de LEGENDRE $P_n(x)$ de degrés n . Les polynômes de LEGENDRE ont la propriété d'être orthogonaux (par rapport au produit scalaire intégral) sur l'intervalle $[-1,1]$. Ils sont obtenus par la formule de récurrence suivante :

$$\begin{aligned} P_0(x) &= 1 \\ P_1(x) &= x \\ (n+1) P_{n+1}(x) &= (2n+1)x P_n(x) - n P_{n-1}(x). \end{aligned}$$

Et leur forme normalisée est :

$$\bar{P}_n(x) = \sqrt{n + \frac{1}{2}} P_n(x).$$

Pour optimiser les calculs, nous avons normalisé et paramétré chaque surface de la scène en $(u,v) \in [0,1]^2$. En posant $x = 2u - 1$ et en normalisant le résultat, on crée une base de fonctions orthonormales sur l'intervalle $[0,1]$:

$$\int_0^1 P_i(u) P_j(u) du = \begin{cases} 1 & \text{si } i = j. \\ 0 & \text{sinon.} \end{cases} \quad (2.2)$$

Dans notre implantation, nous nous sommes limités aux quatre premières bases pour $n \in \{0,1,2,3\}$. Les figures 2.9, 2.10, 2.11 et 2.12 illustrent les différentes fonctions formant les bases orthonormées constantes (1 fonction à partir du polynôme P_0), bi-linéaires (4 = 3 + 1 fonctions, polynômes P_0 et P_1), bi-quadratiques (9 = 4 + 5 fonctions, polynômes P_0 , P_1 et P_2) et bi-cubiques (16 = 9 + 7 fonctions, polynômes P_0 , P_1 , P_2 et P_3) que nous utilisons sur des supports normalisés carrés. Ces fonctions sont construites à partir du produit (en u et v) des polynômes de LEGENDRE suivants :

$$\begin{aligned} P_0(u) &= 1, \\ P_1(u) &= \sqrt{3}(2u - 1), \\ P_2(u) &= \sqrt{5}(6u^2 - 6u + 1), \\ P_3(u) &= \sqrt{7}(20u^3 - 30u^2 + 12u - 1). \end{aligned} \quad (2.3)$$

Ce qui donne comme expression pour les fonctions ondelettes multiples en deux dimensions :

$$\mathcal{M}_{ij}(u,v) = P_i(u)P_j(v). \quad (2.4)$$

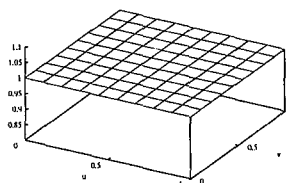


FIG. 2.9 — $\mathcal{M}_{00}(u,v)$ — Polynôme de LEGENDRE constant.

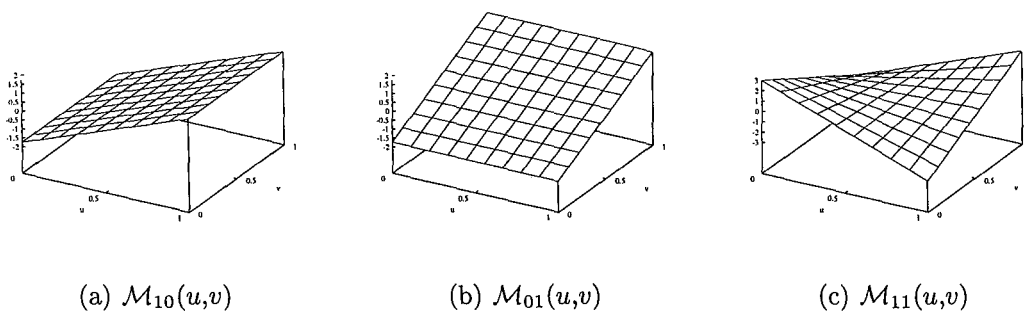


FIG. 2.10 — Polynômes de LEGENDRE bi-linéaires.

En utilisant l'équation 2.2, on montre simplement que les fonctions ondelettes multiples en dimension deux forment une base orthonormale qui vérifie donc :

$$\int_0^1 \int_0^1 \mathcal{M}_{ij}(u,v) \mathcal{M}_{kl}(u,v) du dv = \begin{cases} 1 & \text{si } i = k \text{ et } j = l. \\ 0 & \text{sinon.} \end{cases}$$

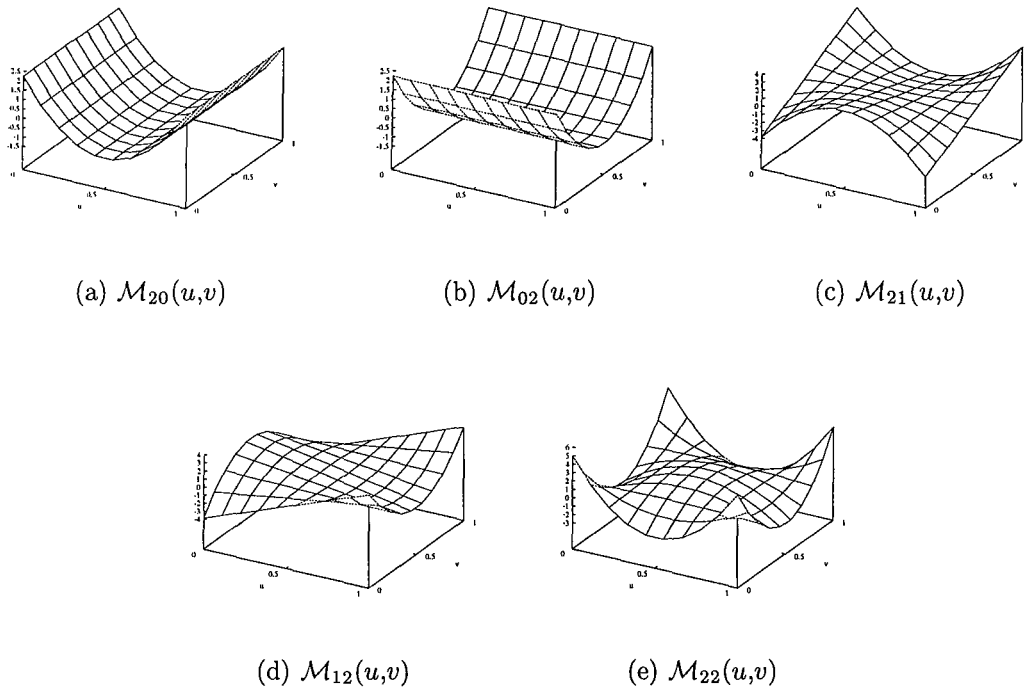


FIG. 2.11 – Polynômes de LEGENDRE bi-quadratiques.

La base bi-linéaire est formée par $\{\mathcal{M}_{00}, \mathcal{M}_{10}, \mathcal{M}_{01}, \mathcal{M}_{11}\}$, la base bi-quadratique reprend les 4 polynômes précédents plus $\{\mathcal{M}_{20}, \mathcal{M}_{02}, \mathcal{M}_{21}, \mathcal{M}_{12}, \mathcal{M}_{22}\}$ et la base bi-cubique se compose de $\{\mathcal{M}_{30}, \mathcal{M}_{03}, \mathcal{M}_{31}, \mathcal{M}_{13}, \mathcal{M}_{32}, \mathcal{M}_{23}, \mathcal{M}_{33}\}$ et des 9 polynômes précédents.

Sur des supports triangulaires (triangles rectangles), la base orthonormale de polynômes vérifie :

$$\int_0^1 \int_0^{1-u} \mathcal{M}_{ij}(u,v) \mathcal{M}_{kl}(u,v) du dv = \begin{cases} \frac{1}{2} & \text{si } (i,j) = (k,l). \\ 0 & \text{sinon.} \end{cases} \quad (2.5)$$

2.4.2 Bases pour la méthode de collocation de points

La méthode de collocation nécessite des fonctions qui prennent la valeur 1 en un point de collocation et 0 sur tous les autres points. Pour répondre à ce besoin, nous utilisons les polynômes de LAGRANGE. Un polynôme de LAGRANGE $L_n(x)$ de degrés n est une fonction qui passe par $n+1$ points connus (x_i, y_i) , $i \in [0, n]$. L'expression générale d'un tel polynôme est la suivante :

$$L_n(x) = \sum_{i=0}^n y_i \frac{(x-x_0) \cdots (x-x_{i-1})(x-x_{i+1}) \cdots (x-x_n)}{(x_i-x_0) \cdots (x_i-x_{i-1})(x_i-x_{i+1}) \cdots (x_i-x_n)}. \quad (2.6)$$

Si on applique ce principe en deux dimensions sur des supports normalisés, on obtient comme base constante $\mathcal{N}_1(u,v) = L_0(u)L_0(v) = 1$, fonction identique à $\mathcal{M}_{00}(u,v)$ illustrée

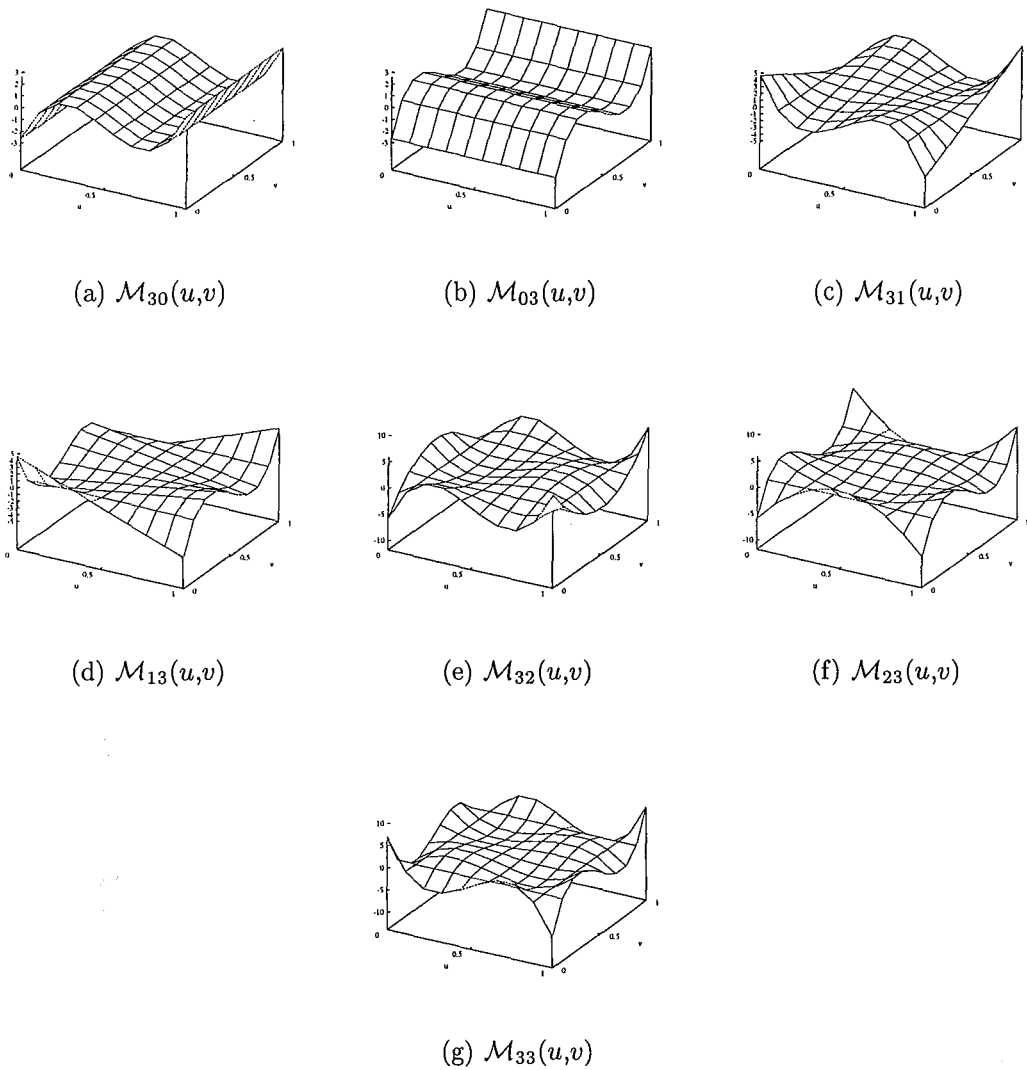


FIG. 2.12 – Polynômes de LEGENDRE bi-cubiques.

par la figure 2.9. Cette base d'ondelettes correspond à la base de HAAR aussi utilisée dans la méthode de radiativité hiérarchique classique [Han 91].

La base bi-linéaire (figure 2.13) s'exprime en fonction des 4 polynômes :

$$\begin{aligned} \mathcal{N}_2^{(0,0)}(u,v) &= (1-u)(1-v), \\ \mathcal{N}_2^{(1,0)}(u,v) &= u(1-v), \\ \mathcal{N}_2^{(0,1)}(u,v) &= (1-u)v, \\ \mathcal{N}_2^{(1,1)}(u,v) &= uv, \end{aligned}$$

où $\mathcal{N}_2^{(s,t)}(u,v)$ représente la fonction bi-linéaire qui est égale à 1 au point $(u=s, v=t)$ et à 0 pour les autres points de collocation (soit $(u=s, v=s)$, $(u=t, v=t)$ et $(u=t, v=s)$). Les points de collocation sont choisis de façon à former une grille régulière de taille $n \times n$

où n est l'ordre des polynômes de LAGRANGE utilisés.

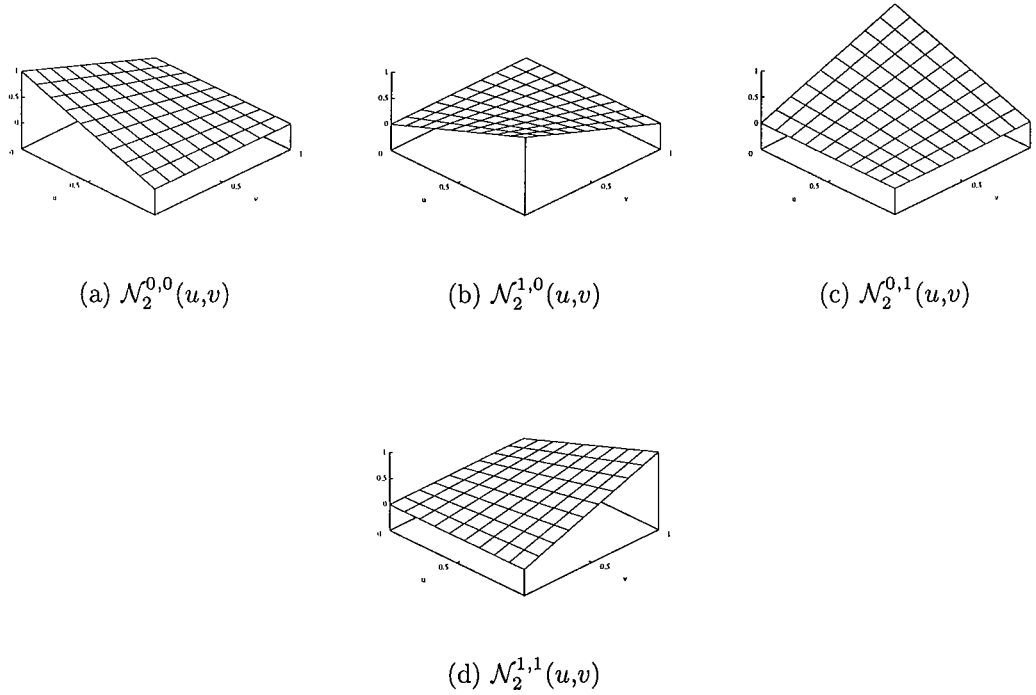


FIG. 2.13 – Base linéaire de polynômes de LAGRANGE.

La base de collocation bi-quadratique (figure 2.14) que nous utilisons est décrite par les 9 polynômes :

$$\begin{aligned}
 \mathcal{N}_3^{(0,0)}(u,v) &= (1-u)(1-2u)(1-v)(1-2v) \\
 \mathcal{N}_3^{(\frac{1}{2},0)}(u,v) &= 4u(1-u)(1-v)(1-2v) \\
 \mathcal{N}_3^{(1,0)}(u,v) &= -u(1-2u)(1-v)(1-2v) \\
 \mathcal{N}_3^{(0,\frac{1}{2})}(u,v) &= (1-u)(1-2u)4v(1-v) \\
 \mathcal{N}_3^{(\frac{1}{2},\frac{1}{2})}(u,v) &= 4u(1-u)4v(1-v) \\
 \mathcal{N}_3^{(1,\frac{1}{2})}(u,v) &= -u(1-2u)4v(1-v) \\
 \mathcal{N}_3^{(0,1)}(u,v) &= -(1-u)(1-2u)v(1-2v) \\
 \mathcal{N}_3^{(\frac{1}{2},1)}(u,v) &= -4u(1-u)v(1-2v) \\
 \mathcal{N}_3^{(1,1)}(u,v) &= u(1-2u)v(1-2v),
 \end{aligned}$$

qui prennent respectivement la valeur 1 aux points $(0,0)$, $(\frac{1}{2},0)$, $(1,0)$, $(0,\frac{1}{2})$, $(\frac{1}{2},\frac{1}{2})$, $(1,\frac{1}{2})$, $(0,1)$, $(\frac{1}{2},1)$ et $(1,1)$.

Pour des supports normalisés à formes triangulaires, la base bi-linéaire est engendrée par trois polynômes $\{\mathcal{N}_2^{(0,0)}, \mathcal{N}_2^{(1,0)}, \mathcal{N}_2^{(0,1)}\}$ et la base bi-quadratique est composée des cinq fonctions $\{\mathcal{N}_3^{(0,0)}, \mathcal{N}_3^{(\frac{1}{2},0)}, \mathcal{N}_3^{(1,0)}, \mathcal{N}_3^{(0,\frac{1}{2})}, \mathcal{N}_3^{(0,1)}\}$.

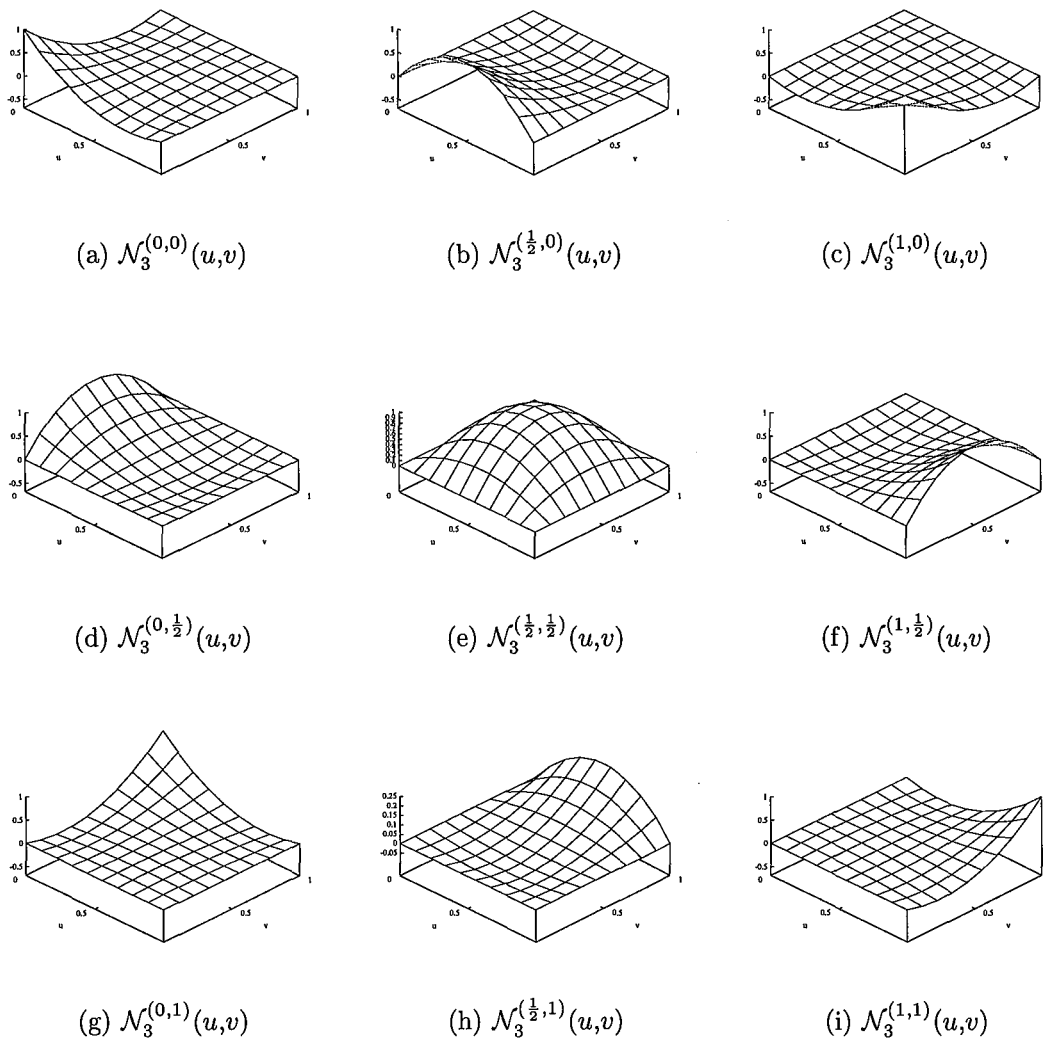


FIG. 2.14 – Base de polynômes de LAGRANGE quadratique.

2.5 Comparaison de différentes bases

Dans cette section, nous présentons les résultats expérimentaux obtenus lors des comparaisons que nous avons effectuées entre les différentes bases d'ondelettes. Nous avons divisé cette présentation en trois sections. Dans la première nous décrivons le protocole expérimental que nous avons utilisé lors de nos tests. Dans la deuxième nous exposons nos résultats. La dernière section nous permettra de faire une étude de ces résultats et de les comparer aux résultats des études précédentes.

2.5.1 Protocole expérimental

2.5.1.1 Bases d'ondelettes

Comme nous venons de le voir, nous disposons dans Candela d'un grand nombre de bases d'ondelettes différentes. Les tests que nous avons menés sur ces bases ont donné des résultats très similaires avec les méthodes de GALERKIN et de collocation. Nous n'avons pas fait de distinction dans nos résultats entre les deux méthodes. Nous avons préféré focaliser notre étude sur les différences qui apparaissent quand on change l'ordre des fonctions de base. Nous présentons ici les résultats obtenus avec les ondelettes multiples (GALERKIN) \mathcal{M}_1 (HAAR = constant), \mathcal{M}_2 (linéaire) et \mathcal{M}_3 (quadratique).

Nous n'avons pas implanté dans Candela les bases d'ondelettes connues sous le nom de « flatlets » (\mathcal{F}_n). Les fonctions utilisées pour ces bases sont des fonctions constantes par morceaux. Même si ces fonctions peuvent être d'ordre quelconque, elles ne donneront jamais une meilleure approximation que la base de HAAR avec plus de subdivisions. Notre but principal étant d'obtenir la meilleure approximation possible de la fonction de radiosité, nous avons préféré concentrer nos recherches sur des bases offrant plus de continuité dans les résultats.

2.5.1.2 Scènes de test

Pour tester l'efficacité des différentes bases d'ondelettes, nous les avons utilisées les unes après les autres sur plusieurs scènes de tests. Les quatre scènes que nous avons utilisées ont été choisies car elles couvrent la grande majorité des configurations que l'on peut retrouver lors de simulations de propagation de la lumière.

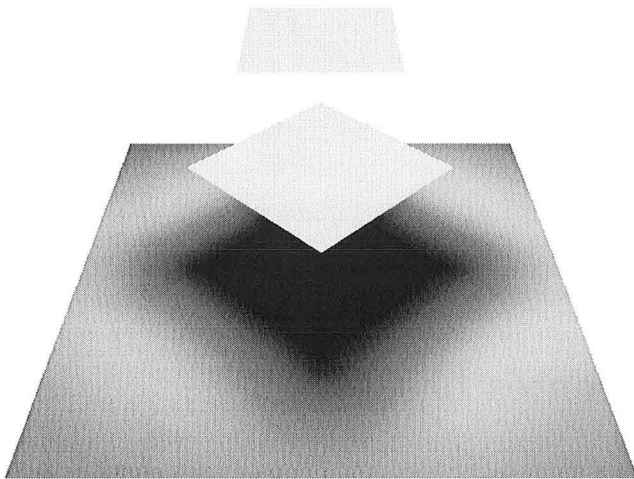


FIG. 2.15 – *Scène du bloqueur*

La scène du bloqueur : elle a pour but de vérifier l'efficacité des bases d'ondelettes pour approcher de fortes discontinuités dues à des changements de visibilité.

La scène du bloqueur est représentée sur la figure 2.15. Une surface éclairante est placée au dessus d'une surface jouant le rôle de bloqueur elle même au dessus d'une plus grande surfaces réceptrice. Les surfaces sont placées de telle façon que l'ombre sur la surface réceptrice n'est pas parallèle à ses bords.

Ce que l'on cherche à obtenir sur cette scène, c'est une ombre aussi pré-

cise que possible sur la surface réceptrice et avec un minimum d'artefacts. La profondeur des subdivisions nécessaire pour atteindre une bonne approximation sera un critère important d'efficacité pour ce test.

La scène du tube : elle a pour but de vérifier l'efficacité des bases d'ondelettes lors de la propagation de la lumière et plus particulièrement lors des propagations secondaires.

La scène du tube est représentée sur la figure 2.16. Une surface éclairante est placée au bas d'un tube 5 fois plus haut que large. La lumière se propage depuis cette surface le long du tube. Vu la hauteur du tube, le haut de ce dernier n'est éclairé que par les réflexions secondaires le long de ses bords.

Ce que l'on cherche à obtenir sur cette scène, c'est un éclairage très continu et très « propre » même au niveau du sommet du tube. L'apparition de discontinuités dues à la grande différence d'énergie entre les premières propagations et les suivantes doit être évitée au maximum.



FIG. 2.16 – Scène du tube

La scène de la salle à manger : elle a pour but de vérifier l'efficacité des bases d'ondelettes sur une scène de complexité modeste mais regroupant tous les phénomènes difficiles à simuler : ombres très marquées, propagations secondaires importantes, grandes surfaces éclairantes (les murs).

La scène de la salle à manger est représentée sur la figure 2.17. Cette scène proposée par Peter SHIRLEY au 5^{ème} Workshop sur le rendu¹² représente une pièce avec une table et quatre chaises. Une surface éclaire toute la scène depuis le plafond. Des ombres très marquées apparaissent sous la table, partie qui n'est éclairée que par les réflexions secondaires. Le sol est la principale surface permettant d'éclairer (par réflexion) le dessous de la table. Cette surface est relativement grande, ce qui suppose une subdivision relativement fine pour atteindre un bon niveau de précision.

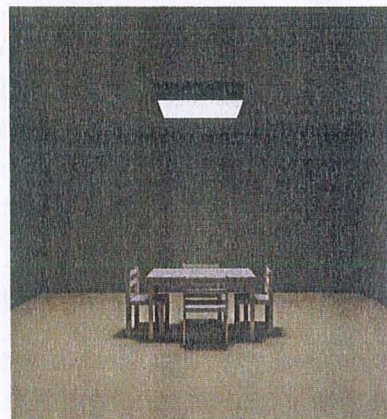


FIG. 2.17 – Scène de la salle à manger (402 surfaces initiales)

Ce que l'on cherche à obtenir sur cette scène, est une solution globale de très grande qualité. Les points importants dans ce test sont : la rapidité de convergence, le nombre de subdivisions et le nombre de mailles finales.

La scène de la salle de cours : elle a le même but que la scène de la salle à manger avec une complexité plus grande. Elle ajoute aux difficultés de la scène précédente une distribution non régulière de la géométrie.

La scène de la salle de cours est représentée sur la figure 2.18. Cette scène a aussi été proposée par Shirley. Elle représente une pièce avec une table, 32 chaises et 4 surfaces

12. <http://www.cs.utah.edu/~shirley/>

éclairantes au plafond. La particularité de cette scène est que toutes les chaises ont été placées de façon à ne jamais présenter la même configuration géométrique par rapport au sol.

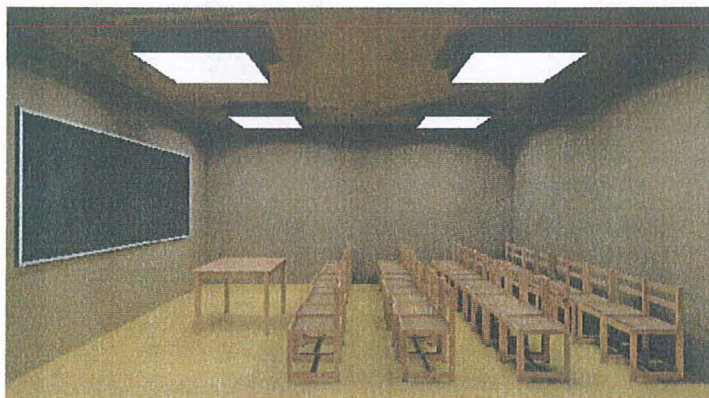


FIG. 2.18 – Scène de la salle de cours (3153 surfaces initiales)

Ce que l'on cherche à obtenir sur cette scène, c'est à nouveau, une solution globale de très grande qualité. Le point important supplémentaire qu'il faut surveiller, au cours de ce test, est la qualité de toutes les ombres au niveau des pieds des chaises. Bien sûr, la qualité des inter-réflexions, le temps nécessaire à la simulation et le nombre de subdivisions restent des points très repré-

sentatifs de l'efficacité de l'algorithme et il ne faut pas les négliger.

Autre scènes de test : nous avons utilisé nos fonctions ondelettes sur d'autres scènes de test issues d'applications du monde réel. Cependant nous souhaitons ici nous concentrer sur une étude précise de l'impact de l'utilisation des ondelettes d'ordre élevé par rapport aux fonctions d'ordre moins élevé. Les résultats que nous avons obtenus sur des scènes beaucoup plus complexes issues d'applications du monde réel sont, quand à eux, présentés dans le chapitre 4 (p. 137) à la fin de cette thèse.

2.5.1.3 Visualisation des résultats

Toutes les images que nous présentons pour illustrer nos résultats ont été obtenues directement à partir des valeurs issues de nos simulations contrairement à celles présentées dans la plupart des travaux sur la radiosité. Nous n'avons utilisé aucun post-traitement pour améliorer l'aspect visuel de nos résultats. Les images représentent exactement la fonction de radiosité telle qu'elle a été évaluée lors des itérations de l'algorithme.

La continuité visuelle de la fonction n'est due qu'aux subdivisions décidées dans la fonction oracle et au pouvoir d'approximation propre des fonctions ondelettes d'ordre élevé. Nous n'avons pas non plus modifié le maillage pour le rééquilibrer et le rendre plus facile à utiliser lors d'un éventuel lissage du résultat.

Pour afficher nos résultats nous avons utilisé des cartes graphiques. Malheureusement les cartes actuelles ne peuvent pas afficher des variations de couleurs autres que des variations linéaires. Les résultats obtenus avec les fonctions ondelettes d'ordre plus élevé que les fonctions linéaires n'ont donc pas pu être affichés directement. Pour pouvoir envoyer aux cartes graphiques des données qui représentent correctement les variations des fonctions quadratiques (\mathcal{M}_3 et \mathcal{N}_3), nous avons décidé de subdiviser chaque maille en quatre (lors de l'utilisation de ces fonctions). Le résultat affiché dans ce cas est donc une approximation linéaire des variations quadratiques de la fonction de radiosité.

Les derniers tests que nous avons effectués ont montré que cette opération ne permet

pas toujours de bien profiter du pouvoir d'approximation de fonctions ondelettes quadratiques. Ainsi, certains artefacts ne sont dus qu'au « mauvais » affichage du résultat. Nous explorons actuellement les possibilités d'un affichage interactif à base d'un lancé de rayons dans un algorithme parallèle. Cette solution permettra d'afficher exactement la valeur de la fonction de radiosit  que quel que soit l'ordre des fonctions de base.

2.5.1.4  valuation des erreurs

Afin d'obtenir une erreur num rique lors de nos tests, nous avons commenc  par calculer des solutions de r f rence pour chaque sc ne de test. Chaque solution de r f rence a  t  calcul e en utilisant les fonctions ondelettes lin aires et une m thode de GALERKIN (fonction \mathcal{M}_2). Pour avoir une solution fiable, nous avons fortement augment  la pr cision des calculs aussi bien au niveau du seuil de pr cision dans l'oracle de subdivision qu'au niveau de l'aire minimale autoris e pour la subdivision des mailles. Le seuil de l'oracle a  t  choisi 10 fois plus petit que le plus petit seuil utilis  lors de nos tests et l'aire minimale a  t  prise 16 fois plus petite que lors de nos tests. Pour chacune des bases d'ondelettes, nous avons v rifi  que les simulations convergeaient bien vers ces solutions de r f rence lorsque l'on augmentait la pr cision et ce jusqu'  une pr cision sup rieure   toutes les pr cisions des r sultats que nous exposons.

Pour chaque simulation, nous avons calcul  la diff rence  nerg tique avec la solution de r f rence. Cette diff rence a pu  tre calcul e pr cis ment, gr ce   la repr sentation fonctionnelle des deux solutions. La valeur calcul e est la suivante :

$$\mathcal{D} = \int_S |B_{ref}(\vec{x}) - B_{simu}(\vec{x})| d\vec{x}$$

o  S est l'ensemble des surfaces de la sc ne, $B_{ref}(\vec{x})$ est la valeur de la fonction de radiosit  au point \vec{x} pour la solution de r f rence et $B_{simu}(\vec{x})$ la valeur de la fonction pour la simulation.

Pour toujours avoir une valeur comparable d'une simulation   une autre et d'une sc ne   une autre, nous avons divis  \mathcal{D} par l' nergie totale contenue dans la sc ne. Nous avons ainsi obtenu, pour chaque sc ne et pour chaque pr cision, une approximation de l'erreur par la formule suivante :

$$\mathcal{E} = \frac{\int_S |B_{ref}(\vec{x}) - B_{simu}(\vec{x})| d\vec{x}}{\int_S |B_{ref}(\vec{x})| d\vec{x}}$$

Nous appelons \mathcal{E} l'**erreur globale**. Une erreur globale de 10^{-1} signifie donc que la diff rence  nerg tique entre la solution de la simulation et la solution de r f rence repr sente 10 % de l' nergie totale comprise dans la sc ne.

La figure 2.19 montre les diff rents r sultats visuels que nous obtenons pour trois valeurs de l'erreur globale sur la sc ne du bloqueur.

D'apr s nos exp rimentations, cette mesure de l'erreur est tr s coh rente et permet d'obtenir des r sultats visuels comparables pour toutes les sc nes. Nous avons notamment constat  qu'une valeur de 10^{-1} de l'erreur globale donnait toujours un r sultat visuel tr s m diocre similaire   celui de la figure 2.19(a). Une valeur de l'erreur globale de 10^{-2} donne

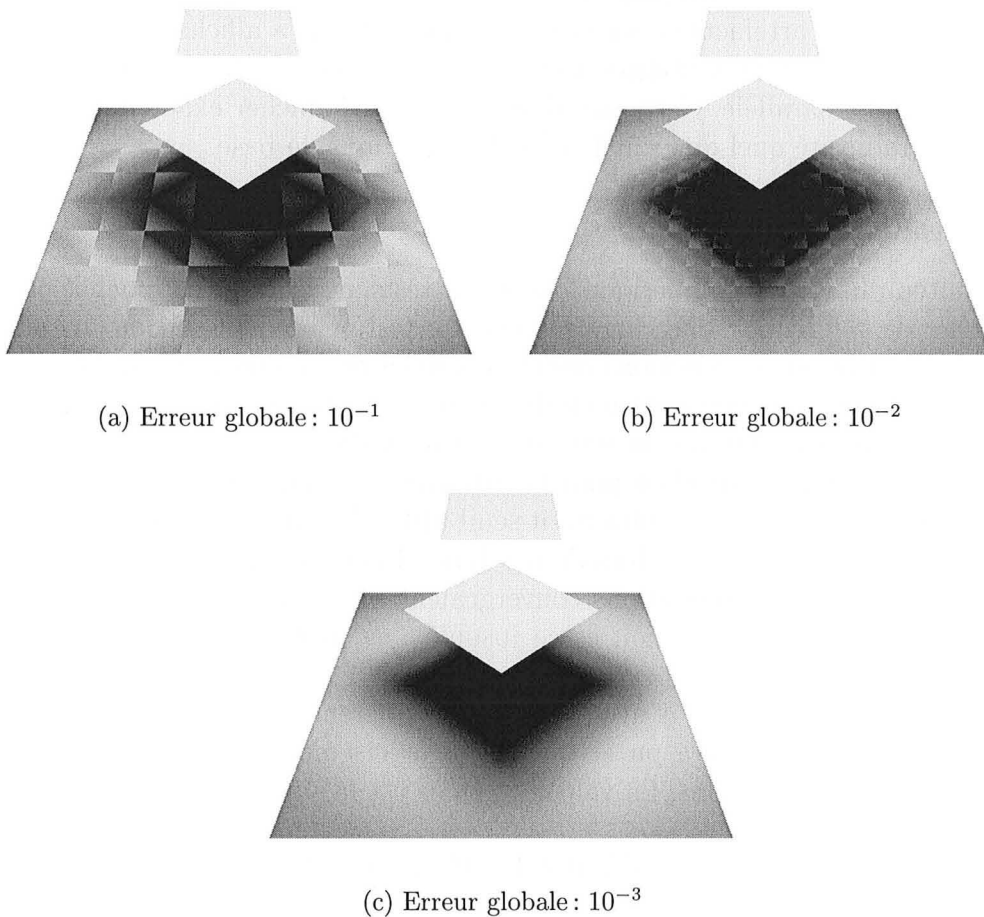


FIG. 2.19 – Résultat d'une simulation avec différentes valeurs de l'erreur globale

toujours un résultat visuel satisfaisant mais sur lequel il reste des artefacts au niveau des limites d'ombres : voir la figure 2.19(b). Enfin, une valeur de l'erreur globale de 10^{-3} procure toujours un résultat visuel parfait comme le montre la figure 2.19(c).

En fait, nous avons constaté que la meilleure valeur pour l'erreur globale était approximativement $5 \cdot 10^{-3}$. Avec cette valeur, la différence énergétique avec la solution de référence ne représente que 0.5 % de l'énergie de la scène et le résultat visuel est très bon. Une valeur plus petite suppose que l'on effectue des calculs beaucoup plus précis pour n'améliorer que faiblement la qualité visuelle de la simulation. Une valeur plus élevée fournit un résultat où les artefacts sont visibles au premier coup d'œil.

2.5.1.5 Rapport entre erreur globale et erreur locale

Lors d'un calcul de radiosité, la fonction oracle s'appuie sur un seuil de précision pour décider de subdiviser ou de ne pas subdiviser une interaction. Ce seuil de précision est toujours utilisé comme une valeur locale de l'erreur autorisée. C'est-à-dire que si l'erreur commise au niveau même de l'interaction est supérieure à ce seuil, on décide de subdiviser. Ce seuil permet donc d'assurer que, pour chaque interaction, l'erreur commise n'a pas

dépassé une certaine valeur.

Le calcul d'erreur que nous avons présenté ci-dessus fournit une valeur de l'erreur globale. C'est-à-dire que la valeur de l'erreur que nous testons mesure une différence sur toute la scène et non plus de façon locale sur une interaction. Notre mesure permet donc de détecter des erreurs dues aux multiples réflexions successives de la lumière et aussi de détecter une propagation éventuelle de l'erreur au fil des itérations.

Comme le seul paramètre modifiable avant les calculs est le seuil de la fonction oracle, il est important de connaître la relation qu'il existe entre l'erreur globale et l'erreur locale. Nous avons effectué des tests dans ce sens et nos résultats nous ont conduit aux mêmes conclusions que Marc STAMMINGER *et al.* dans [Sta 98]. Nous avons, en effet, constaté que l'erreur locale et l'erreur globale étaient fortement corrélées. La figure 2.20 montre cette corrélation pour les différentes scènes de test utilisées.

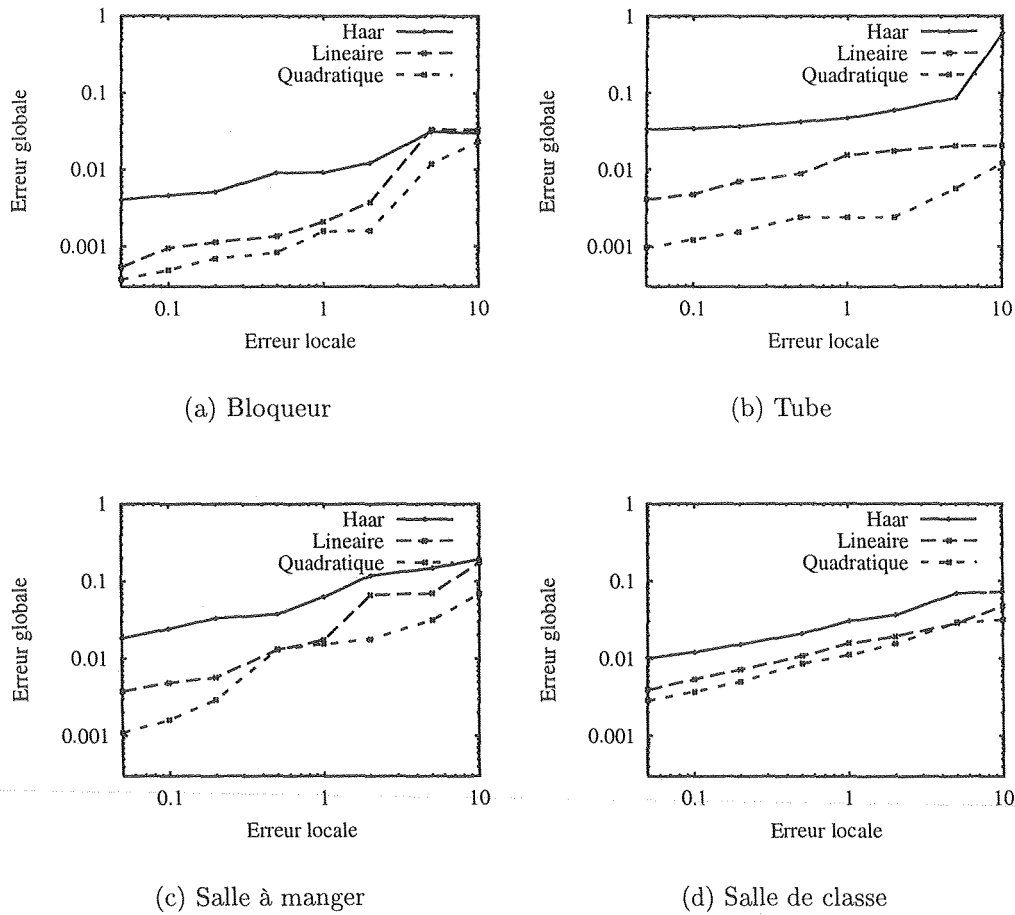


FIG. 2.20 – Erreur globale en fonction de l'erreur locale.

Pour les 4 scènes, on peut remarquer sur les figures 2.20(a,b,c et d) que la progression de l'erreur globale en fonction de l'erreur locale suit grossièrement une évolution linéaire. Ce résultat montre qu'agir sur l'erreur locale permet d'agir proportionnellement sur l'erreur globale et confirme la corrélation entre les deux erreurs.

2.5.1.6 Détail techniques

Pour toutes nos expérimentations, nous avons utilisé la même plate-forme logicielle Candela. Nous avons fait tourner tous nos calculs sur la même machine une Octane de SGI avec une cpu à 225 MHz et 256 Mo de RAM.

Afin d'effectuer une comparaison aussi « impartiale » que possible des différentes bases d'ondelettes, nous avons utilisé un algorithme de visibilité à base de lancé de rayons et non notre algorithme de visibilité hybride [Alo 99]. L'algorithme de visibilité hybride utilise beaucoup de temps pour calculer des *bitmaps* qui sont ensuite partagées pour effectuer différentes requêtes de visibilité. Le temps nécessaire pour effectuer les requêtes sur les *bitmaps* est négligeable par rapport au temps de création de ces *bitmaps*. Cet algorithme a donc tendance à avantager les méthodes de résolution nécessitant plus de requêtes de visibilité. C'est justement le cas des algorithmes à base d'ondelettes où le nombre de requêtes est d'autant plus important que l'ordre de fonctions utilisées est élevé.

2.5.2 Résultats

2.5.2.1 Comparaison visuelle des trois bases d'ondelettes

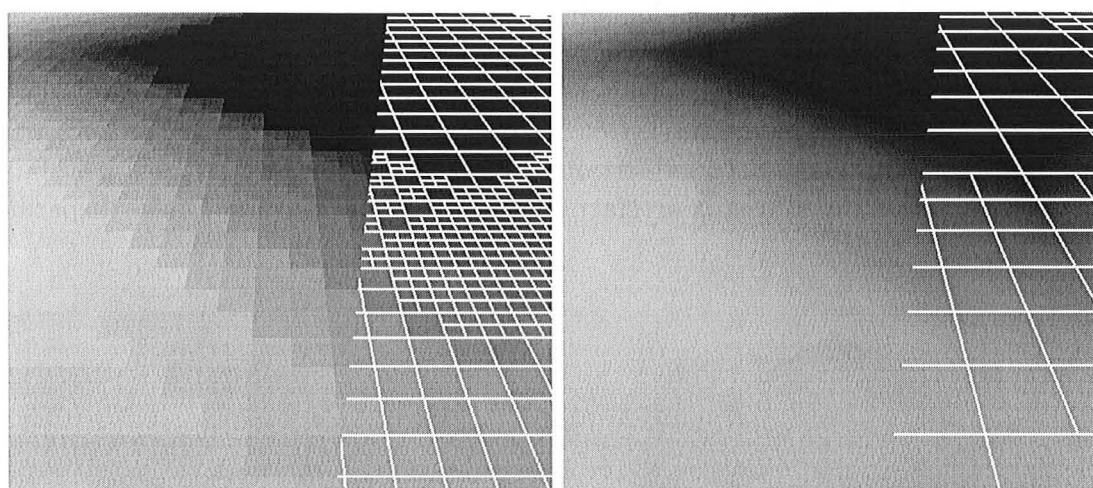
Le premier test que nous avons effectué avait pour objectif de voir si l'utilisation d'ondelettes d'ordre élevé donnaient un résultat visuel de meilleure qualité que les bases d'ordre moins élevé. Dans leur étude expérimentale [Wil 97a], Andrew WILLMOTT et Paul HECKBERT avaient conclu que les ondelettes d'ordre élevé étaient incapables de fournir un bon résultat visuel au niveau des limites d'ombres. Or, la représentation correcte des limites d'ombres a beaucoup d'importance quand on veut faire de la synthèse d'images, car la netteté de ces zones a un très gros impact sur l'impression visuelle générale obtenue lors de l'affichage.

Nos premières expériences ont donc été focalisées sur ce point. Pour l'étudier nous avons utilisé la scène du bloqueur où une lampe surfacique éclaire une grande surface avec un bloqueur placé entre les deux et projetant son ombre sur la surface illuminée. Cette scène est très bien adaptée au test que nous voulons effectuer car l'éclairage de la surface du bas se découpe suivant trois zones. Une zone totalement illuminée, une zone de pénombre et une zone totalement sombre. Les limites entre ces zones ne sont pas alignées suivant les axes de la surface et elles correspondent exactement aux limites d'ombres difficiles à approcher avec une représentation fonctionnelle.

Nous avons comparé le comportement des trois bases d'ondelettes sur cette scène. Pour cela, nous avons effectué une simulation ne comportant qu'une itération de l'algorithme, c'est-à-dire que seule la source surfacique a transmis son énergie vers le reste de la scène. Pour avoir des résultats comparables, nous nous sommes arrangés pour que le temps de la simulation soit exactement le même dans les trois cas (66 s).

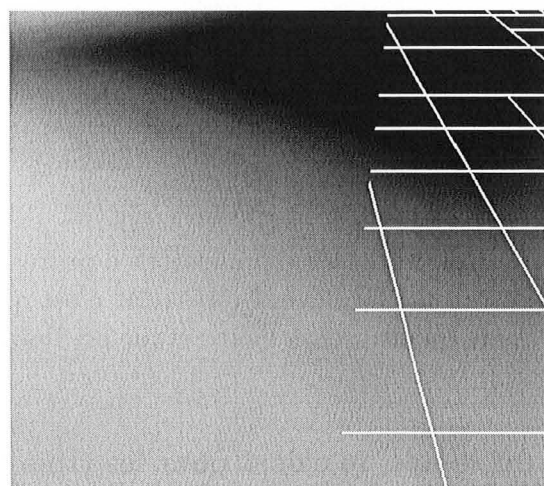
La figure 2.21 montre le résultat obtenu en calculant la radiosité avec chacune des trois bases d'ondelettes ainsi que le maillage associé à chaque calcul. Deux éléments importants sautent aux yeux en observant ces images :

- le résultat visuel brut est nettement plus continu et donc plus satisfaisant avec les fonctions linéaires et quadratiques qu'avec la base de HAAR.



(a) Haar

(b) Linéaire



(c) Quadratique

FIG. 2.21 – *Comparaison visuelle de nos trois bases d'ondelettes*

- plus les fonctions ondelettes sont d'ordre élevé, plus la solution obtenue est compacte. C'est-à-dire que le maillage est d'autant moins fin que les fonctions sont d'ordre élevé. Ce point est très important parce que ce phénomène se remarque non seulement dans les zones où la fonction de radiativité est relativement continue, mais aussi et surtout au niveau des limites de discontinuité dues aux frontières des ombres portées. Pour un temps équivalent et avec notre algorithme, nous obtenons donc un résultat radicalement différent de celui qu'avaient obtenu Andrew WILLMOTT et Paul HECKBERT dans [Wil 97a].

Les résultats obtenus avec la base de HAAR ne sont généralement pas affichés dans leur forme « brute » comme nous l'avons fait. En effet, il est habituel d'utiliser une sorte de

post-processing, tel que le « Gouraud shading » pour lisser la fonction de radiosité entre les différentes mailles d'une surface originale. La figure 2.22 montre l'effet de l'utilisation du Gouraud shading sur le résultat obtenu avec la base de HAAR (Gouraud shading appliqué au résultat de la figure 2.21). Comme on peut le remarquer sur cette figure, cet algorithme peut faire disparaître un certain nombre de discontinuités mais il le fait au prix de l'introduction de nouveaux artefacts qui eux sont impossibles à faire disparaître.

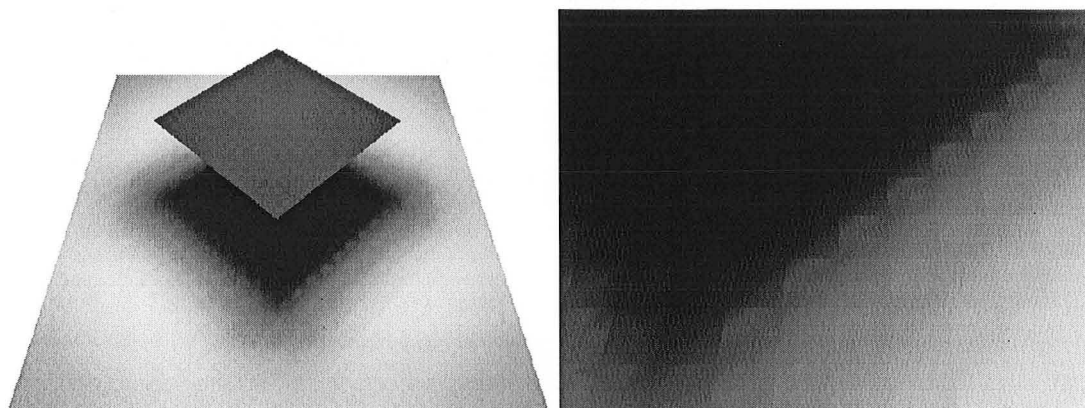


FIG. 2.22 – Effet de l'utilisation du Gouraud shading sur le résultat obtenu avec la base de HAAR

À en juger par la figure 2.21 les bases d'ondelettes d'ordre élevé sont plus efficaces que les bases d'ondelettes d'ordre moins élevé. Ce résultat n'est que qualitatif et nécessite une confirmation par une étude quantitative. Cette étude est l'objet des deux sections qui suivent.

2.5.2.2 Comparaison du temps de calcul pour les trois bases d'ondelettes

La figure 2.23 présente la comparaison du temps de calcul pour les trois bases d'ondelettes en fonction de l'erreur globale. Les tests ont été effectués sur nos quatre scènes de test afin d'avoir une vue globale indépendante de la géométrie initiale.

Le point le plus important qui ressort de cette étude expérimentale est, qu'avec notre implantation de l'algorithme de radiosité à base d'ondelettes, l'efficacité de l'algorithme est d'autant plus grande que l'ordre des fonctions de base est élevé. Les fonctions d'ordre élevé fournissent des résultats de meilleure qualité et elles permettent d'accélérer les calculs de façon significative. Pour arriver à un résultat visuel satisfaisant sur la scène de la salle de cours (suivant le critère numérique décrit à la section 2.5.1.4 (p. 71), c'est-à-dire pour passer en dessous de $5 \cdot 10^{-3}$ en erreur globale), les ondelettes quadratiques ont besoin de 10^4 s (voir la figure 2.23(d)). Dans le même temps, la base de HAAR n'arrive à atteindre qu'une valeur de 10^{-2} pour l'erreur globale. Cette scène de test est notre scène la plus difficile car elle contient énormément de limites d'ombres dont il est très dur de faire une approximation. C'est justement sur ce genre de scènes que les ondelettes d'ordre élevé n'avaient que des résultats très médiocres (et même souvent mauvais par rapport aux autres fonctions) dans les études expérimentales précédentes [Wil 97a].

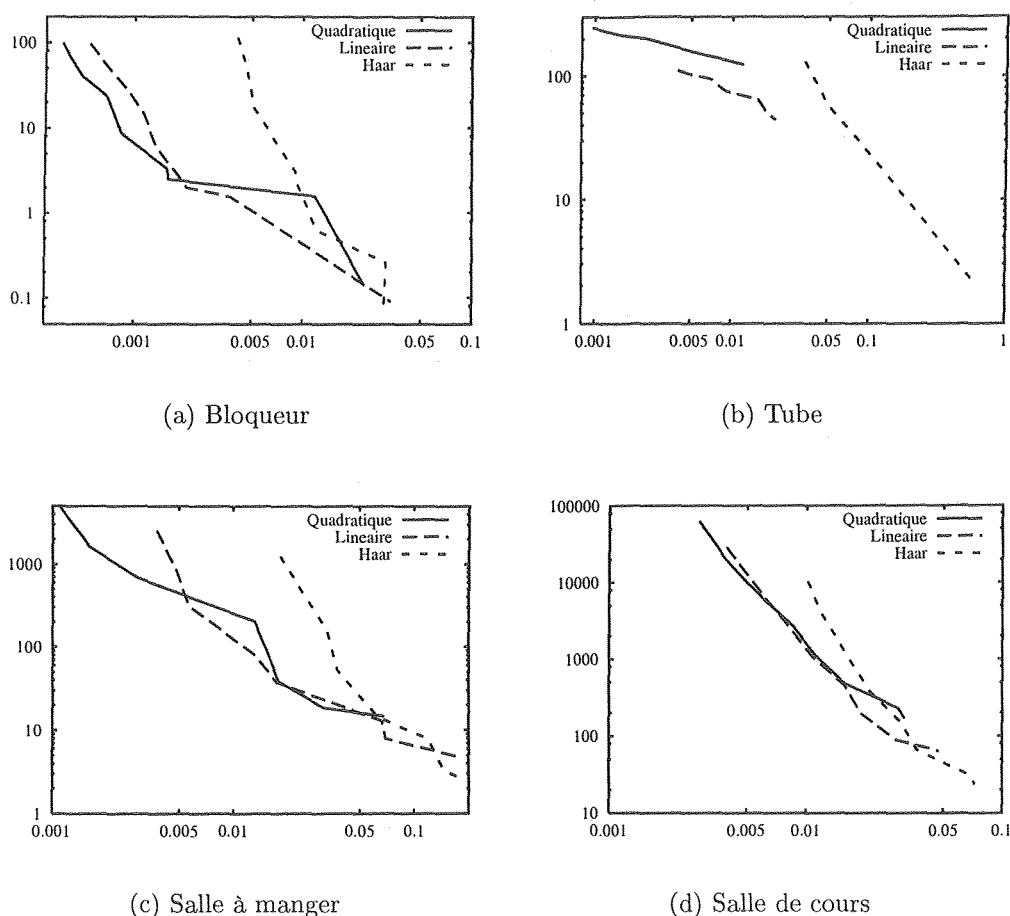


FIG. 2.23 – Temps de calcul (ordonnée en s) en fonction de l'erreur globale (abscisse).

Une deuxième remarque que l'on peut faire en étudiant la figure 2.23 est que l'avantage des fonctions d'ordre élevé est plus grand si la qualité du résultat demandé est plus précise et si la scène est plus complexe. Plus on demande de la précision dans le résultat plus la différence de temps de calcul avec les fonctions d'ordre peu élevé s'accroît.

À l'inverse, pour des approximations de qualité médiocre les ondelettes linéaires sont plus efficaces que les ondelettes quadratiques. Le même constat s'applique avec la base de HAAR qui devient plus efficace que les bases linéaires et quadratiques pour des calculs très rapides et de très faible qualité.

De façon générale, on remarque qu'il existe des *zones de compétences* pour chaque base d'ondelettes. Dans ces zones, la base concernée est plus efficace à tous points de vue que les autres bases : la base de HAAR est la plus efficace pour des simulations dont l'erreur globale reste supérieure à 10^{-1} , ce qui correspond à une simulation où il subsiste beaucoup d'artefacts visibles, mais dont le temps de calcul est très petit. La base linéaire dépasse les autres en efficacité dans une zone dont l'erreur globale va de 10^{-1} à $5 \cdot 10^{-3}$ et la base quadratique est la meilleure pour des simulations très précises dont l'erreur globale est inférieure à $5 \cdot 10^{-3}$.

2.5.2.3 Comparaison de la mémoire utilisée pour les trois bases d'ondelettes

Lors de nos tests, nous avons effectué un suivi de la quantité de mémoire utilisée lors des calculs avec les différentes bases d'ondelettes sur nos quatre scènes de test. Les courbes sur la figure 2.24 montrent l'utilisation mémoire en fonction de l'erreur globale lors des simulations successives que nous avons effectuées.

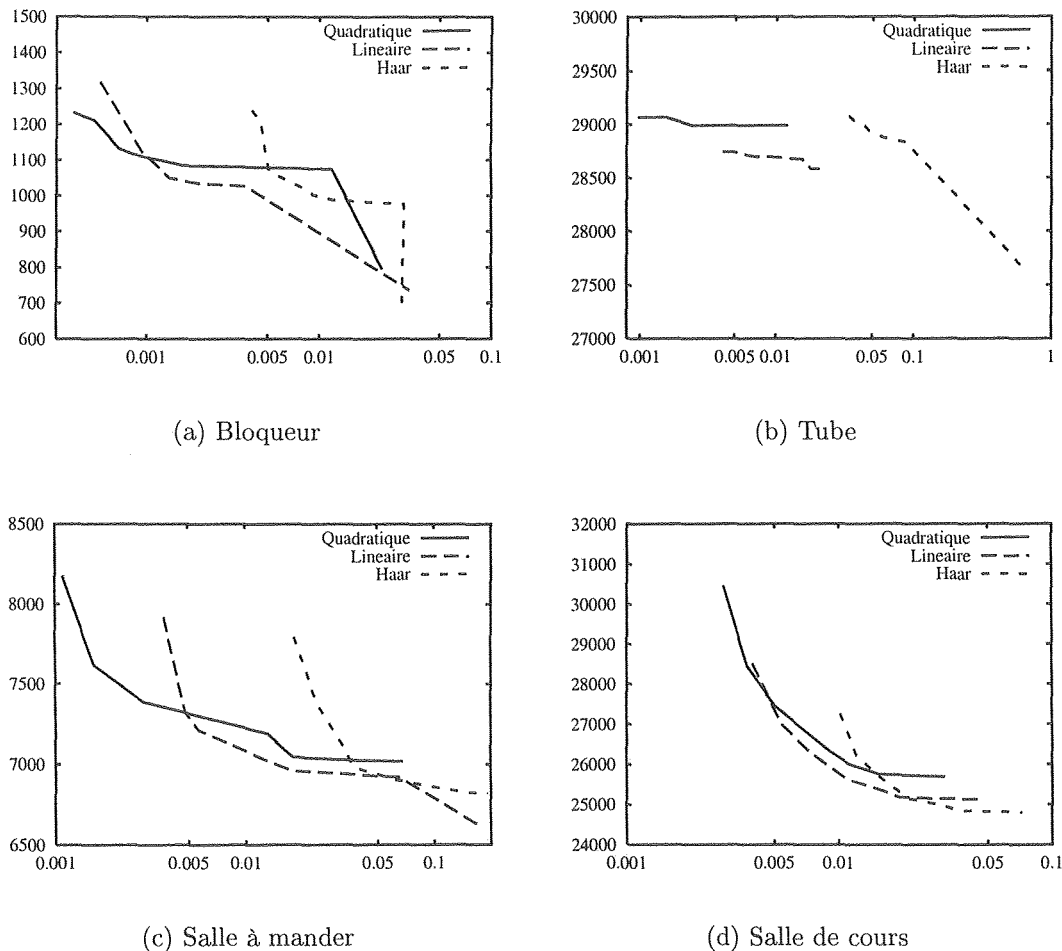


FIG. 2.24 – Mémoire maximum utilisé (ordonnée en kB) en fonction de l'erreur globale (abscisse)

Comme on peut le voir sur la figure 2.24, pour des simulations avec une très grande précision (erreur globale inférieure à $5 \cdot 10^{-3}$), les bases d'ondelettes d'ordre élevé utilisent *moins* de mémoire que les bases d'ordre moins élevé. Ce phénomène est particulièrement évident sur nos scènes de test les plus complexes (voir figure 2.24(c) et figure 2.24(d)).

D'un autre côté, pour des simulations avec une précision plus faible, cette hiérarchie est inversée. Pour des valeurs de l'erreur globale assez élevées, la base de HAAR et la base linéaire consomment moins de mémoire que la base quadratique. Une fois encore, chaque base d'ondelettes a une zone de compétence où elle est plus efficace et moins consommatrice en mémoire que les autres bases. Pour des simulations très approximatives,

les ondelettes de HAAR sont les plus efficaces en terme de consommation mémoire. Pour des simulations plus précises c'est la base linéaire qui devient la plus efficace. Finalement, la base quadratique prend le dessus lorsque l'on fait des simulations avec une très grande précision.

Un résultat très intéressant qui ressort de notre étude est la façon dont le coût en mémoire pour une base d'ondelettes donnée varie entre les différentes simulations. En effet, ce coût augmente soudainement lorsque l'on cherche à faire passer l'erreur globale en dessous d'un certain seuil. Ce résultat est particulièrement bien visible sur les figures 2.24(c) et 2.24(d). Un tel résultat laisse à penser qu'il existe une précision maximum pour chaque base d'ondelettes au delà de laquelle la base perd toute son efficacité et conduit à des simulations difficiles à gérer en terme de mémoire. Il faut cependant relativiser ce résultat par le fait que le changement brusque de pente des courbes de la figure 2.24 est accentué par l'utilisation d'une échelle logarithmique sur l'axe des abscisses pour suivre l'erreur globale, alors que l'axe des ordonnées suivant la consommation mémoire utilise une échelle linéaire. De plus, l'augmentation de la consommation mémoire reste petite par rapport au total de la mémoire utilisée : entre 10 % et 20 % de la mémoire totale. Le changement de pente est cependant suffisamment important pour que l'on puisse en conclure qu'il sera très difficile de passer en dessous d'un certain seuil de précision. D'après nos courbes, il paraît clair que ce seuil existe pour chaque base d'ondelettes mais qu'il varie suivant la base : plus les fonctions de la base sont d'ordre élevé plus le seuil est bas. Étant donné que ce phénomène apparaît de façon similaire pour toutes les bases d'ondelettes utilisées lors de nos tests nous pensons que c'est un effet général associé à l'utilisation des ondelettes et qu'il s'appliquera à toutes les bases d'ondelettes.

Le fait que les bases d'ondelettes d'ordre élevé soient plus efficaces que les bases d'ondelettes d'ordre moins élevé pour faire une approximation de la fonction de radiosité, est un phénomène logique que les études théoriques [Gor 93, Arv 94] avaient annoncé. En effet, les ondelettes d'ordre élevé offre un outil plus performant que les ondelettes d'ordre moins élevé pour faire une approximation de fonctions complexes. Elles permettent de bénéficier d'un espace fonctionnel de dimension plus élevée et elles ont la propriété d'avoir un moment plus grand, ce qui leur permet d'avoir une représentation plus compacte des fonctions complexes. Grâce à ces propriétés, les ondelettes d'ordre élevé nécessitent moins de coefficients pour représenter une fonction complexe que les ondelettes d'ordre moins élevé. Elles utilisent donc moins de place en mémoire lors des simulations avec une grande précision. Nos expérimentations fournissent donc un résultat pratique en rapport avec les attentes théoriques.

Par ailleurs, il était aussi prévisible que les fonctions ondelettes d'ordre peu élevé seraient très efficaces pour des simulations avec une faible précision. En effet, les simulations avec une faible précision ne cherchent pas à prendre en compte toute la complexité de la fonction de radiosité (par définition même de la faible précision). Elles ont pour but d'obtenir rapidement une représentation relativement simple de la fonction de radiosité. Pour arriver à cette représentation simplifiée, l'espace fonctionnel peut être relativement réduit et l'ordre des fonctions n'a pas besoin d'être très élevé. Il est donc facile et rapide, pour des bases de fonctions simples telles que les ondelettes d'ordre peu élevé, d'effectuer cette approximation.

2.5.3 Discussion et comparaison de nos résultats avec ceux des études précédentes

Bien que nous aboutissions à des conclusions opposées de celles de Andrew WILLMOTT et Paul HECKBERT dans leur étude [Wil 97a, Wil 97b] nous voulons souligner le fait que nos résultats sont néanmoins consistants avec les leurs.

En effet, dans leur étude Andrew WILLMOTT et Paul HECKBERT ont utilisé les ondelettes avec un algorithme classique de radiosité hiérarchique. Les ondelettes d'ordre élevé souffrent donc chez eux de la quantité de mémoire nécessaire pour conserver les liens d'interaction. La conséquence directe de cette algorithme était que les ondelettes d'ordre élevé étaient rendues totalement inutilisables en dehors des simulations avec une faible précision. Or, d'après les résultats expérimentaux que nous venons d'exposer, les ondelettes d'ordre peu élevé utilisent effectivement moins de mémoire et sont plus rapides que les ondelettes d'ordre élevé lors de simulations avec une faible précision.

Notre étude peut de ce point de vue être considérée comme une extension de l'étude de Andrew WILLMOTT et de Paul HECKBERT aux simulations de grande précision. De telles simulations de grande précision sont en fait rendues possibles par le fait de ne plus stocker les liens d'interaction en mémoire. Une fois que l'on a éliminé le stockage des liens, la mémoire nécessaire se résume à la mémoire nécessaire au stockage du maillage. Or, l'oracle de subdivision assure que le maillage soit optimal pour représenter la fonction de radiosité.

De plus, en concentrant l'oracle sur les variations de la fonction de radiosité sur la surface réceptrice et plus sur l'interaction, nous pouvons exploiter pleinement le pouvoir d'approximation des fonctions ondelettes d'ordre élevé. Ainsi, le maillage résultat est mieux adapté aux variations de la fonction et moins profond aux endroits où ces variations ne nécessitent pas de subdivisions.

2.6 Conclusion

Dans ce chapitre, nous avons présenté notre implantation des bases d'ondelettes dans l'algorithme de radiosité. Avec cette implantation nous avons pu mener un grand nombre d'expérimentations pour tester différentes bases d'ondelettes. En plus des ondelettes classiques pour la méthode de GALERKIN, nous avons testé de nouvelles bases d'ondelettes adaptées à la méthode de résolution par collocation de points.

Nos expériences nous ont permis de prouver que pour des simulations de grande précision les ondelettes d'ordre élevé donnaient de meilleurs résultats que celles d'ordre moins élevé. Nous avons montré que ces résultats étaient meilleurs à tous points de vue. Non seulement, les qualités visuelle et numérique des simulations se rapproche beaucoup plus de celles des solutions de références, mais de plus, les calculs durent moins longtemps et nécessitent moins de place mémoire avec les bases d'ordre élevé.

Notons que nous avons fait très attention pour faire nos tests à ce que notre implantation ne désavantage pas les bases d'ordre peu élevé. Nous avons, par exemple, pris en compte toutes les optimisations relatives à la base de HAAR pour ne pas la pénaliser par rapport aux autres bases. Notons aussi que le fait de ne pas conserver les liens d'interaction

en mémoire est indépendant de la base choisie et n'influence en aucun cas les résultats pour une base particulière.

Bien que tous les résultats ici correspondent à des tests menés sur des scènes de relativement faible complexité (moins de 3 000 polygones), nous avons aussi utilisé notre implantation de l'algorithme de radiosité à base d'ondelettes sur des scènes du monde réel. Ces dernières sont de complexité telle qu'il aurait été difficile de les utiliser pour faire l'ensemble des tests. Afin de rester exhaustif, nous avons donc décidé de nous limiter ici aux scènes pour lesquelles nous pouvions faire l'ensemble des expérimentations. Nous avons rassemblé tous les tests menés sur les scènes issues d'applications industrielles à la fin de cette thèse dans le chapitre 4 (p. 137).

Un résultat important de notre étude est que le choix optimal pour la base de fonctions dépend de la précision. Les fonctions d'ordre peu élevé sont les plus efficaces pour effectuer des simulations de faible précision alors que les fonctions d'ordre plus élevé sont meilleures pour les simulations de plus grande précision. En fait, un certain degré de précision pour chaque base d'ondelettes pour lequel la base concernée dépasse toutes les autres en efficacité, aussi bien en temps de calcul qu'en mémoire nécessaire. Bien que nos tests se soient limités aux bases de HAAR, \mathcal{M}_2 et \mathcal{M}_3 nous pensons que ce phénomène sera aussi valable pour des bases d'ordre plus élevé telles que les bases $\mathcal{M}_4, \mathcal{M}_5, \dots$ et que pour des simulations encore plus précises la base \mathcal{M}_4 sera plus performante que la base \mathcal{M}_3 et ainsi de suite.

Le niveau de précision qui correspond, d'après notre expérience, à des simulations visuellement acceptables se trouve à la limite entre les zones de compétence des bases \mathcal{M}_2 et \mathcal{M}_3 . Ainsi, la base \mathcal{M}_2 est généralement suffisante pour toutes les simulations destinées à la visualisation. D'autre part, la base \mathcal{M}_2 présente l'avantage de fournir une solution qui varie linéairement sur les mailles finales et qui peut donc être affichée directement par les cartes graphiques.

Dans le futur, nous voulons explorer la possibilité d'utiliser plusieurs bases d'ondelettes au cours d'une même simulation. Dans cette approche, l'idée est d'utiliser la base de HAAR avec très peu de calculs pour les interactions ne nécessitant qu'une précision relativement faible (interaction transportant peu d'énergie) et de passer à une base de degré plus élevé, \mathcal{M}_2 ou même \mathcal{M}_3 , pour des interactions plus importantes. Cette solution pourrait donner de très bons résultats avec un algorithme de *shooting*, car les premières interactions transportent beaucoup d'énergie alors que les suivantes en transportent de moins en moins et nécessitent de moins en moins de précision. Contrairement à la critique émise il y a quelques années sur nos travaux *shooting is not dead*.

Chapitre 3

Maillage virtuel

« Toute substance composée, dans le monde, se compose de parties simples, et il n'existe absolument rien que le simple dans le monde. »

Blaise PASCAL, *Pensées*

Sommaire

3.1	Introduction	84
3.2	État de l'art	90
3.2.1	Trois raisons pour utiliser des carrés normalisés	90
3.2.2	Recherches antérieures pour gérer les surfaces complexes	92
3.2.3	Nos contributions	94
3.3	Principe du maillage virtuel	95
3.3.1	Passage des primitives quelconques aux carrés normalisés	96
3.3.2	Fonction de radiosité sur le maillage virtuel	102
3.3.3	Déduire la radiosité sur les primitives originales	107
3.4	Maillage virtuel et radiosité à base d'ondelettes	108
3.4.1	Algorithme	108
3.4.2	Coefficients de <i>push-pull</i>	109
3.4.3	Primitives auto-émettrices	114
3.4.4	Gérer la hiérarchie	114
3.5	Applications	115
3.5.1	Primitives planes quelconques	116
3.5.2	Quadriques	116
3.6	Résultats expérimentaux	125
3.6.1	Résultats avec des surfaces planes à bords complexes	126
3.6.2	Résultats avec les surfaces paramétriques	129
3.7	Conclusion	135

Résumé

Dans ce chapitre, nous introduisons une méthode générale permettant d'effectuer le calcul de radiosit  dans des sc nes compos es d'une grande vari t  de surfaces param triques. Cette m thode permet notamment de g rer efficacement les primitives planes avec des bords quelconques et les surfaces param triques courbes de degr  peu  lev . Contrairement aux pr c dentes m thodes qui n cessitaient une triangulation des primitives complexes initiales, notre m thode permet de b n ficier pleinement de toute la richesse et de la grande compacit  de la repr sentation math matique des primitives g om triques. Le c ur de notre algorithme est le *maillage virtuel*. C'est une abstraction des donn es g om triques initiales qui permet d'illuminer des primitives complexes exactement comme des triangles ou des parall logrammes. Le maillage virtuel est un ensemble de carr s normalis s sur lesquels les primitives g om triques initiales sont projet es tout en pr servant leurs propri t s  nerg tiques. Les valeurs de radiosit  sont calcul es sur ces nouveaux supports tout au long de la simulation et sont re-projet es sur les primitives originales   la fin des calculs.

Pour d montrer toute la puissance de notre m thode, nous l'avons implant e dans notre algorithme de radiosit    base d'ondelettes d crit dans le chapitre pr c dent. Nous l'avons utilis e pour g rer des primitives planes   bords complexes et des morceaux de quadriques.

Nous donnons des r sultats qualitatifs et quantitativs sur un certain nombre d'objets et d'environnements mod lis s pour des simulations de r alit  virtuelle. Ces r sultats montrent qu'en int grant de fa on exacte les primitives complexes dans le processus de r solution, le maillage virtuel permet d'effectuer un rendu r aliste pour des sc nes de grande complexit  g om trique. En effet, le maillage virtuel permet d'acc l rer les calculs et fournit une simulation de meilleure qualit  gr ce   une mod lisation plus naturelle.

3.1 Introduction

Les sc nes cr ees avec les outils de mod lisation actuels renferment g n ralement une grande vari t  de primitives g om triques. Le point commun entre la plupart de ces primitives est que ce sont g n ralement des surfaces param triques d crites au moyen de deux param tres. Ce param trage permet de les mailler   volont  et donc de les afficher facilement quelle que soit la pr cision que l'on d sire. Des exemples connus de ces surfaces sont les NURBS, les cylindres, les sph res, les c nes et tous les morceaux de plans que l'on retrouve g n ralement sous forme de polygones. Ces surfaces sont le plus souvent d coup es suivant les fronti res issues de l'intersection d'un certain nombre d'entre elles ou de la combinaison CSG des volumes leur correspondant. De plus, dans un futur proche, on saura retrouver   partir des points acquis au moyen de scanners 3D un ensemble de surfaces param triques plus ou moins complexes d crivant parfaitement la surface ainsi scann e. Ce mode de repr sentation r duira consid rablement le nombre des primitives g om triques par rapport au nombre important de triangles que l'on g n re   l'heure actuelle pour repr senter ces surfaces. Dans ce chapitre, nous pr sentons une nouvelle m thode tr s puissante et tr s efficace permettant d'effectuer un calcul de radiosit  directement sur les surfaces param triques.

Une hypothèse de base de l'algorithme de radiosit  original est que les primitives g om triques composant les sc nes   illuminer sont planes [Gor 84]. Malheureusement, aucune des d couvertes et am liorations dans le domaine n'a r ellement permis de s'affranchir totalement de cette hypoth se. Par exemple, dans les algorithmes de radiosit  hi rarchique introduits par Pat HANRAHAN *et al.* dans [Han 91], les mailles finales sont toujours des triangles ou des parall logrammes. Le calcul de radiosit  par la m thode de GALERKIN pr sent e par Harold ZATZ dans [Zat 93] fait exception   la r gle en proposant une solution totalement ind pendante de la g om trie initiale gr ce   des calculs d'interaction bas s sur des approximations par quadrature. Cependant, les implantations de cette m thode dans des algorithmes de radiosit  hi rarchique se sont toujours limit es   l'utilisation de primitives g om triques planes. Comme nous l'avons vu au chapitre pr c dent, les algorithmes de radiosit    base d'ondelettes [Gor 93, Sch 94a] font partie de cette cat gorie et n'ont jamais donn  lieu   des implantations ind pendantes de la g om trie.   l'heure actuelle, tous les algorithmes permettant de simuler l'illumination globale dans des environnements diffus sur des primitives g om triques courbes utilisent d'une fa on ou d'une autre une approximation g om trique pour g rer ces primitives. Cette approximation peut  tre faite de deux fa ons :

- en maillant les primitives courbes au moyen de primitives planes simples (g n ralement des triangles). Il r sulte de ce type d'approximation une multiplication importante du nombre de primitives initiales, ce qui rend le mod le beaucoup plus complexe.
- en faisant une approximation g om trique au moment de calculer les coefficients d'interaction pour effectuer les transferts d' nergie [Arq 95].

Quelle que soit l'approximation g om trique utilis e, la fonction de radiosit  est  valu e au prix d'une certaine erreur. Dans aucun des deux cas, les donn es g om triques ne sont utilis es de fa on exacte pour effectuer les calculs. Par cons quent,   aucun moment et avec aucune des solutions actuelles l'int gration de primitives courbes dans un algorithme de radiosit  n'est apparue comme une extension logique et naturelle de la radiosit  sur des surfaces polygonales.

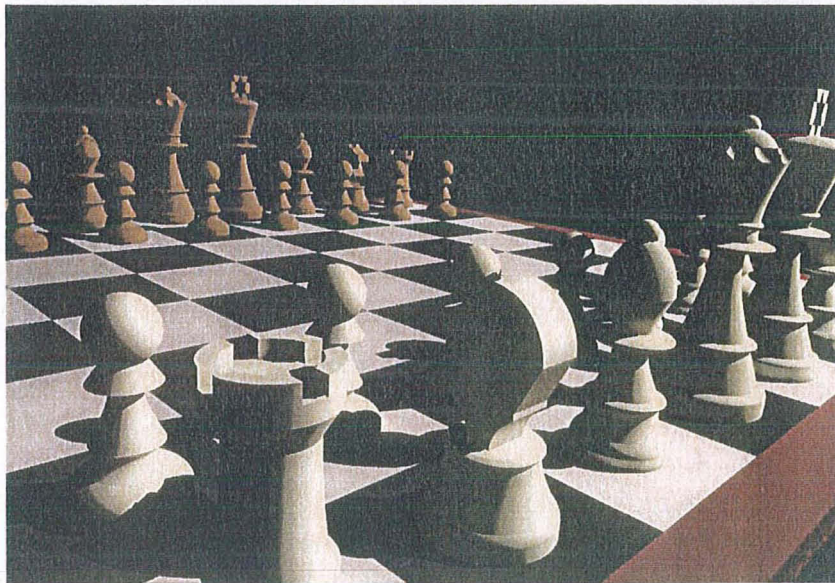
Dans ce chapitre, nous pr sentons le *maillage virtuel*, une solution innovante et efficace pour effectuer le calcul de radiosit  sur des primitives g om triques courbes. Le maillage virtuel permet de combiner la puissance des algorithmes   base de subdivisions adaptatives de la g om trie avec celle de l'analyse fonctionnelle tout en  vitant d'effectuer un maillage pr alable des primitives initiales. Gr ce   notre solution, nous b n ficioons au sein d'un m me algorithme de :

- la description g om trique pr cise et tr s compacte des surfaces param triques. Nous avons ainsi la possibilit  de mod liser de fa on beaucoup plus r aliste des objets pr sentant des surfaces courbes ou avec des bords complexes.
- l'utilisation tr s efficace des bases de fonctions hi rarchiques de degr   lev  (comme nous l'avons pr sent  au chapitre pr c dent). Ces fonctions trouvent toute leur puissance math matique gr ce   la formulation de GALERKIN des algorithmes de r solution adaptatifs.

D'autre part, comme notre algorithme ne fait aucune approximation lors de l'utilisation



(a) La salle de la géométrie



(b) Le jeu d'échecs

FIG. 3.1 – La salle de la géométrie que nous avons introduite dans le 4^{ème} étage du Soda Hall a été illuminée grâce à notre algorithme à base d'ondelettes d'ordre élevé en utilisant le maillage virtuel. Le rendu réaliste de cette scène — modélisée au moyen de surfaces quadriques et incluant le jeu d'échecs — aurait été impossible avec les algorithmes antérieurs et avec une rapidité et une qualité équivalente. En effet, le nombre de primitives initiales aurait dépassé la capacité de tous les algorithmes existants.

des primitives géométriques complexes, nous évitons la quasi-totalité des erreurs géométriques classiques des algorithmes de radiosit . Nous  liminons notamment un des facteurs d'erreur d crits par James ARVO *et al.* dans [Arv 94] en partant d'une  quation qui n'est pas fauss e par les approximations g om triques initiales. Nous pensons que le maillage virtuel est la solution efficace pour pouvoir introduire et utiliser des primitives courbes dans les algorithmes de radiosit . Gr ce   cette innovation, il est possible d'effectuer un rendu interactif de tr s grande qualit  sur des sc nes contenant des surfaces complexes en un temps tr s raisonnable (voir la figure 3.1), alors que jusqu'  pr sent l'obtention de la rapidit  de simulation et de la qualit  du r sultat n' tait pas envisageable.

Tous les algorithmes permettant d'obtenir une repr sentation de la fonction de radiosit  au moyen d'approches par  l ments finis supposent que les calculs puissent  tre effectu s sur des supports carr s normalis s (des carr s dont la longueur des c t s est  gale   1) ou  ventuellement sur des triangles rectangles normalis s (des demi-carr s normalis s coup s suivant une diagonale) comme repr sent s sur la figure 3.2.

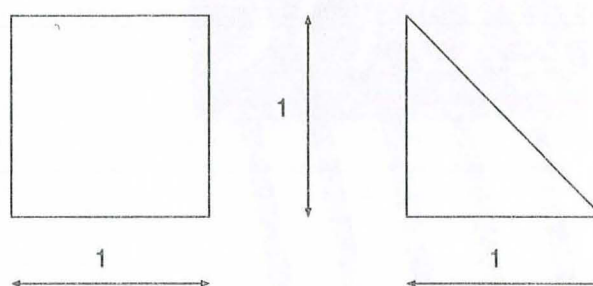


FIG. 3.2 – Supports normalis s pour une repr sentation fonctionnelle.

Le choix et l'utilisation de ces supports sont particuli rement importants pour :

- construire une base de fonction ;
- d finir des r gles optimales pour les calculs de quadrature ;
- maintenir la coh rence entre les niveaux de repr sentation dans le cas d'une repr sentation hi rarchique.

Dans notre m thode, nous construisons une deuxi me repr sentation abstraite des primitives g om triques initiales   base de carr s normalis s que nous appelons : « maillage virtuel ». Sans cette repr sentation, il serait impossible ou totalement inefficace d'effectuer un calcul de radiosit .

Le processus de cr ation et d'utilisation du maillage virtuel dans un algorithme de radiosit  est un processus en trois  tapes. Ces trois  tapes sont r sum es sur la figure 3.3. Chaque  tape correspond exactement au passage d'une vignette   la suivante,   savoir :

- *projeter* les primitives g om triques initiales (les primitives courbes et les primitives planes avec des bords quelconques) sur des carr s normalis s et constituer ainsi le maillage virtuel —  tape repr sent e par le passage de la figure 3.3(a)   la figure 3.3(b),
- * tendre* la fonction de radiosit  sur toute la surface du maillage virtuel, puis calculer la radiosit  en utilisant le maillage virtuel comme nouveau support g om trique —  tape repr sent e par le passage de la figure 3.3(b)   la figure 3.3(c),
- *r cup rer* les valeurs de radiosit    partir des valeurs calcul es sur le maillage virtuel pour les appliquer sur les surfaces originales —  tape repr sent e par le passage de la figure 3.3(c)   la figure 3.3(d).



FIG. 3.3 – *Illumination d'une scène contenant un tore troué.*

La première étape constitue un réel défi scientifique car il s'agit de trouver des fonctions de projection appropriées permettant de conserver les propriétés importantes de la fonction de radiosit . En fait, les fonctions de projection doivent permettre de maintenir une coh rence entre les propri t s  nerg tiques de la radiosit  sur les surfaces originales et sur les supports virtuels. Pour avoir cette propri t , la projection doit  tre inversible et elle doit conserver les rapports d'aire. Ainsi, l' nergie sur une primitive initiale sera toujours proportionnelle   l' nergie sur le support virtuel associ . Avec cette propri t , les fonctions de projection permettent d'utiliser la m thode de GALERKIN hi rarchique avec des fonctions de base d'ordre  lev . Elles permettent aussi de r utiliser   chaque niveau de la hi rarchie un grand nombre d'attributs qui ne d pendent que de la g om trie des surfaces.  tant donn  que cette propri t  ne peut  tre obtenue que pour des surfaces param trables, nous avons restreint notre utilisation de surfaces complexes aux surfaces totalement param trables et nous ne consid rons que ces surfaces dans le reste de ce chapitre.

Un avantage majeur du maillage virtuel est qu'il peut  tre associ    tous les algorithmes existants de radiosit . Ainsi, nous pourrions utiliser notre maillage avec le maillage de discontinuit  et avec les algorithmes hi rarchiques ou encore avec tous les algorithmes bas s sur des m thodes de r solution par  l ments finis. Pour d montrer l'efficacit  de notre approche, nous l'avons implant e dans notre algorithme de radiosit    base d'ondelettes d crit dans le chapitre pr c dent. Nous nous sommes restreints dans notre implantation   deux types de surfaces param triques : les surfaces planes   bords quelconques (et particuli rement les polygones quelconques qui repr sentent la grande majorit  des primitives g om triques des mod les existant   l'heure actuelle) et les surfaces quadriques (c'est- -dire les surfaces alg briques de degr  2). Nous avons implant  les surfaces quadriques car leur repr sentation math matique est relativement simple et qu'elles ont un tr s fort potentiel de mod lisation pour repr senter des objets du monde r el. Ainsi, B. NOURSE *et al.* indiquent dans [Nou 80] que 85 % des pi ces m caniques peuvent  tre parfaitement mod lis es par les quadriques naturelles,   savoir les plans, les c nes, les sph res et les cylindres. De plus, ces surfaces repr sentent un bon point de d part pour, plus tard,  tendre notre m thode   d'autres surfaces param triques plus g n rales.

Ce chapitre est organis  de la fa on suivante. Dans la premi re section, nous faisons un r capitulatif des travaux qui ont d j   t  effectu s pour g rer les surfaces complexes dans des algorithmes de radiosit . La section suivante est consacr e   la description des id es g n rales qui se cachent derri re le maillage virtuel pour les algorithmes d'illumination globale   base de radiosit . Cette description est faite ind pendamment de la m thode de r solution choisie pour calculer la fonction de radiosit . Dans la section 3.4 (p. 108), nous d crivons notre implantation du maillage virtuel dans notre algorithme de radiosit    base d'ondelettes d'ordre  lev . Dans la section 3.5 (p. 115), nous montrons comment obtenir des fonctions de projection conservant les rapports d'aire pour les deux types de surfaces auxquelles nous nous sommes int ress s : les surfaces planes   bords quelconques et les surfaces quadriques. Dans l'avant derni re section, nous pr sentons les r sultats qualitatifs et quantitatifs que nous avons obtenus lors de simulations de l'illumination globale dans des sc nes contenant ces types de surfaces. Nous montrons les diff rences d'efficacit  entre un algorithme de radiosit    base d'ondelettes classiques et notre algorithme avec

l'utilisation du maillage virtuel. Enfin, la dernière section nous permet de faire une brève conclusion et de donner des voies de recherche pour le futur, notamment l'utilisation d'autres améliorations pour encore accélérer et rendre plus génériques les algorithmes de radiosit .

3.2  tat de l'art

Comme nous venons de l'expliquer, la grande majorit  des algorithmes de radiosit  se limitent   l'utilisation de primitives planes simples pour d crire la g om trie des environnements. Cette restriction, qui va contre l'utilisation g n rique et efficace de la description math matique des surfaces, est motiv e par les contraintes li es   la repr sentation fonctionnelle de la radiosit . Dans cette section, nous rappelons tout d'abord pourquoi les seuls supports fonctionnels utilisables sont les carr s normalis s. Nous pr sentons ensuite les solutions qui ont  t  envisag es pour contourner ce probl me et pour pouvoir utiliser des surfaces plus complexes dans les algorithmes de radiosit . Enfin, nous disons en quoi le maillage virtuel est une bonne solution pour r soudre ce probl me.

3.2.1 Trois raisons pour utiliser des carr s normalis s

3.2.1.1 Construction de bases de fonctions en 2D

La r solution de l' quation de radiosit  par des algorithmes informatiques tels que ceux d crits dans les deux livres sur le sujet [Coh 93, Sil 94b], suppose toujours que l'on travaille sur des supports g om triques tr s simples. Cette supposition est non seulement un fait mais surtout une n cessit  car la repr sentation d'une fonction sur un domaine de d finition de forme quelconque n'est pas une chose ais e. La construction de bases de fonctions 2D n'est bien connue que pour des domaines de d finition carr s.

En effet, les solutions propos es dans la litt rature [Gor 93, Zat 93, Tro 93, Sch 94b] pour construire des bases de fonctions en 2D s'appuient sur l'utilisation d'un produit tensoriel de deux bases de fonctions en 1D (la figure 3.4 montre la construction d'une base bi-lin aire).

Cette construction utilise g n ralement des bases orthonormales de polyn mes 1D telles que les bases de LEGENDRE ou de JACOBI (la figure 3.4 correspond aux deux premiers polyn mes de LEGENDRE: $\phi_1(x) = 1$ et $\phi_2(x) = x$). Malheureusement, cette construction ne peut  tre utilis e que sur des supports de fonctions « ordinaires », c'est- -dire sur des parall logrammes ou des triangles. De plus, il n'existe pas de solution simple pour construire facilement des bases de fonctions sur des supports plus complexes.

Il r sulte de cette construction que les domaines de d finition des fonctions 2D sont des carr s. Bien qu'il soit facile d' tendre cette construction au cas des parall logrammes et des triangles (voir la section 3.3.1 (p. 96)), les bases ainsi d finies ne sont jamais adapt es   des supports plus complexes.

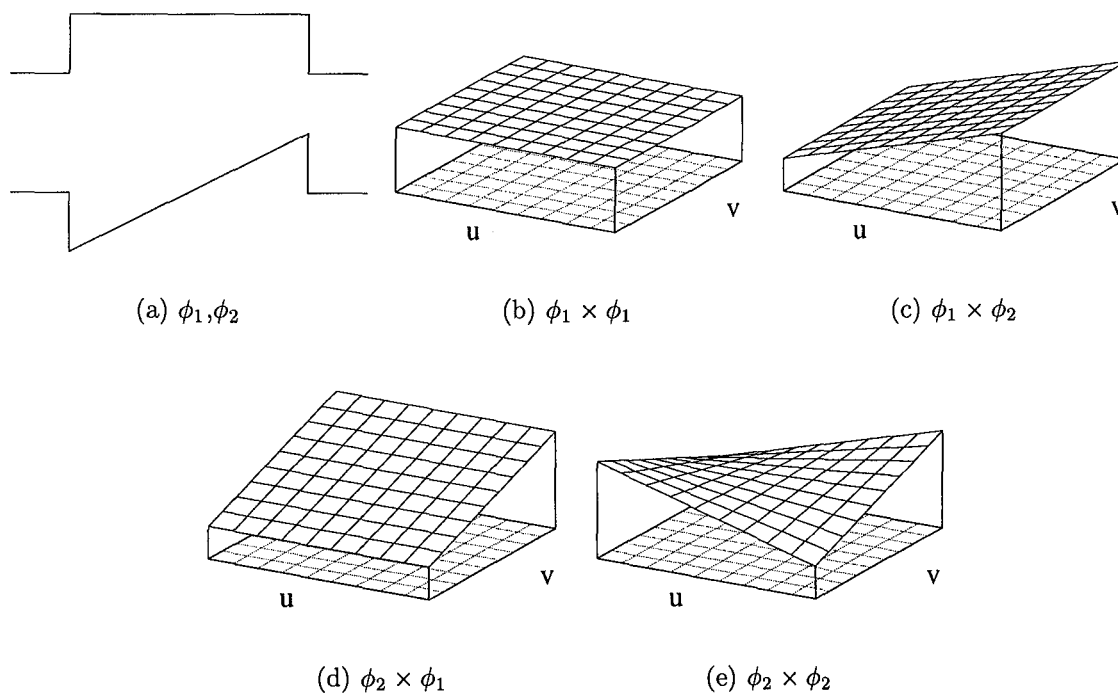


FIG. 3.4 – Construction d'une base linéaire en 2D

3.2.1.2 Calculs de quadrature en 2D

Il n'existe pas de formule analytique pour évaluer l'interaction entre 2 surfaces quelconques. On est donc obligé d'avoir recours à des méthodes d'approximation numérique pour ces calculs. Les méthodes numériques d'approximation d'intégrales sont connues sous le nom de méthodes de quadrature et se basent sur une discrétisation du noyau de l'intégrale pour approcher cette dernière par une somme pondérée :

$$H = \int_X h(x) dx \approx \tilde{H} = \sum_k \omega_k h(x_k).$$

Le but des calculs de quadrature est d'atteindre la plus grande précision possible avec un minimum de points échantillons. Des solutions optimales sont connues pour le cas de la 1D (voir dans [Pre 92]). Ces solutions donnent les points de quadrature (les x_k) et les poids associés (les ω_k) pour les principales bases de fonctions. Ces solutions peuvent facilement être étendues au cas des parallélogrammes en effectuant le produit de deux ensembles de points de quadrature :

$$H_{2D} = \int_X \int_Y h(x,y) dx dy \approx \tilde{H}_{2D} = \sum_k \sum_l \omega_k \omega_l h(x_k, y_l).$$

Par contre il est très difficile, voir impossible, de trouver des règles pour effectuer des quadratures optimales sur des surfaces quelconques.

3.2.1.3 Bases de fonctions hiérarchiques en 2D

Nous avons vu dans la section 1.4.2.1 (p. 34) que les algorithmes hiérarchiques supposent que l'on sait maintenir la cohérence entre toutes les représentations de la fonction à tous les niveaux de la hiérarchie. Pour cela, il faut savoir transférer l'énergie d'un niveau aux niveaux voisins de façon simple et efficace. Nous verrons dans la section 3.4 (p. 108) que cette propriété est très bien remplie par les fonctions ondelettes à condition que les supports des fonctions se subdivisent en conservant certaines propriétés géométriques. Seuls des supports très simples tels que des carrés peuvent assurer cette propriété et il est impossible de la généraliser au cas de supports complexes.

3.2.2 Recherches antérieures pour gérer les surfaces complexes

À cause des trois principaux problèmes évoqués ci-dessus, l'utilisation des méthodes de rendu réaliste à base de radiosit  s'est g n ralement limit e   des sc nes ne contenant que des parall logrammes et des triangles. Quelques recherches ont cependant  t  effectu es pour int grer des surfaces plus complexes dans les algorithmes de radiosit .

3.2.2.1 Solutions pour g rer les polygones

Tr s peu de recherches ont  t  faites pour int grer directement les polygones dans les calculs de radiosit  avec des fonctions d'ordre  lev . Seuls Kadi BOUATOUCH et Sumanta PATTANAİK ont pr sent  dans [Bou 95] une m thode pour int grer le maillage de discontinuit  directement dans un algorithme de radiosit    base d'ondelettes multiples. Dans leur approche, ils proposent, en effet, une solution pour effectuer les calculs d'interaction avec des ondelettes multiples sur des supports polygonaux. Cependant, leur algorithme n cessite le calcul d'un grand nombre de coefficients de transfert relativement compliqu s pour maintenir la coh rence dans la repr sentation hi rarchique de la fonction de radiosit . Nous verrons plus loin que notre approche permet d' viter ces calculs.

La solution la plus directe pour g rer les polygones dans les calculs de radiosit  avec des fonctions d'ordre  lev  consiste en fait   contourner le probl me en effectuant une premi re passe de l'algorithme pour trianguler les surfaces polygonales. Cette solution pr sente beaucoup d'inconv nients :

- Elle augmente sensiblement le nombre de surfaces initiales et donc aussi la complexit  de l'algorithme qui est li e au carr  de ce nombre.
- La triangulation  limine la possibilit  d'effectuer un traitement hi rarchique sur les surfaces originales. Seule une repr sentation hi rarchique sur chaque triangle g n r  est possible. Une telle triangulation diminue donc l'efficacit  des approches hi rarchiques et notamment des approches   base d'ondelettes en pr -d coupant de fa on arbitraire les surfaces.
- Le d coupage arbitraire des surfaces originales g n re des discontinuit s qui se retrouveront dans la repr sentation de la fonction de radiosit .
- La triangulation peut g n rer des triangles avec des formes quelconques et notamment des triangles tr s allong s. Ces derniers peuvent cr er des artefacts lors de

l'utilisation du Z-buffer et peuvent s'avérer très pénalisants lors de l'algorithme de radiosité car il sont difficiles à détecter lors des tests de visibilité.

Une autre solution, proposée par Daniel BAUM *et al.* dans [Bau 91] pour gérer les polygones dans l'algorithme de radiosité, consiste à effectuer un pré-traitement générant des triangles bien adaptés au calcul de radiosité. Cette solution permet d'éviter certains des inconvénients cités ci-dessus mais elle ne s'applique pas aux algorithmes hiérarchiques et ne peut donc pas être utilisée dans les algorithmes de radiosité à base d'ondelettes.

3.2.2.2 Solutions pour gérer les surfaces courbes

Le problème de la gestion des surfaces complexes dans l'algorithme de radiosité est encore plus difficile pour les surfaces courbes que pour les polygones. En effet, comme nous venons de le voir, une solution largement utilisée pour les polygones consiste à trianguler ces surfaces pour contourner le problème. Malheureusement, il n'est pas possible de représenter une surface courbe au moyen d'un ensemble de triangles. Trianguler une surface courbe implique donc nécessairement une approximation géométrique et une certaine perte d'information sur la topologie initiale.

Toutes les solutions proposées jusqu'à présent pour intégrer les surfaces courbes dans l'algorithme de radiosité s'appuient sur une certaine approximation géométrique. Cette approximation peut être basée sur le maillage en surfaces plus simples de la surface initiale, comme c'est le cas avec la triangulation, mais elle peut aussi s'appuyer sur la représentation mathématique de la surface.

En suivant cette dernière voie, Stephan SCHAEFER a proposé dans [Sch 97], une extension en 3D du maillage adaptatif pour introduire les surfaces courbes dans les approches hiérarchiques. Sa méthode appelée « object based meshing » permet d'éviter le maillage initial en utilisant comme première approximation d'une surface courbe sa simple boîte englobante. Puis, à chaque étape de l'algorithme hiérarchique, cette approximation polygonale est améliorée en utilisant des nouveaux points pris sur la boîte englobante et projetés sur la surface originale. Comme les éléments de surfaces aux différents niveaux de la hiérarchie ne représentent pas exactement la même surface, il est nécessaire de redéfinir les coefficients de transfert entre chaque niveau. Un autre point faible de cette méthode est qu'il est très difficile d'évaluer précisément l'impact de cette approximation géométrique sur le résultat final.

Marc STAMMINGER *et al.* ont proposé une autre approche en suivant la même voie, la « bounded radiosity method » [Sta 97a]. Leur solution consiste à prendre en compte la représentation mathématique exacte des surfaces lors des calculs d'interaction sans recourir à un échantillonnage sur des points de référence. Cependant, leur algorithme se base sur un autre type d'approximation : l'oracle de subdivision s'appuie sur l'évaluation de bornes minimum et maximum du facteur de forme, calculées en utilisant une estimation du champ des normales à la surface originale et replacées sur la boîte englobante de la primitive géométrique initiale. Cette solution, qui a l'avantage d'utiliser directement la géométrie initiale, souffre du manque de précision fournie à l'oracle car la différence entre les deux bornes du facteur de forme et d'autant plus importante que la surface présente un grand rayon de courbure.

3.2.2.3 Considérations sur le *clustering*

La méthode classique de radiosité hiérarchique a une complexité en $O(k^2 + n)$, où k est le nombre de surfaces initiales et n le nombre de mailles finales. D'après ce qui précède, le terme k peut augmenter fortement lorsque l'on maille les primitives initiales complexes pour en faire des approximations au moyen de surfaces plus simples. Le terme k^2 prend donc une importance considérable dans la complexité de l'algorithme lorsque l'on veut gérer des surfaces complexes. Plusieurs algorithmes ont été étudiés pour éliminer ce terme de la complexité de l'algorithme. Brian SMITS *et al.* [Smi 94] ainsi que François SILLION [Sil 95a] ont proposé des approches par regroupement de surfaces ou « *clustering* » alors que Holly RUSHMEIER *et al.* [Rus 93] se sont tournés vers la simplification de la géométrie pour résoudre ce problème. Bien que les approches à base de *clustering* puissent être utilisées avec des primitives géométriques courbes, les seules implantations qui ont été faites se sont limitées aux primitives planes. De plus, comme cela est indiqué par Jean-Marc HAZENFRATZ *et al.* dans leur récente étude [Has 99], l'utilisation des méthodes de *clustering* sur des objets qui ont été préalablement maillés sans conserver des connaissances sur leur topologie, s'avère particulièrement inefficace.

Il apparaît donc important d'éviter d'augmenter le nombre de surfaces initiales si l'on veut gérer efficacement des primitives courbes dans un algorithme de radiosité.

3.2.2.4 Défauts des méthodes actuelles

Comme nous venons de le présenter, les meilleures méthodes proposées jusqu'à présent pour gérer efficacement les surfaces complexes dans un algorithme de radiosité ont encore quelques défauts :

- L'approche traditionnelle qui consiste à mailler la surface complexe pour la transformer en un ensemble de surfaces plus simples génère d'importants artefacts géométriques et engendre des problèmes de discontinuité. De plus, elle multiplie le nombre de surfaces initiales, ce qui complique les calculs de visibilité. Enfin, les surfaces générées ont souvent des formes mal adaptées aux calculs de radiosité (surfaces très allongées).
- Toutes les méthodes hiérarchiques proposées supposent que le niveau hiérarchique le moins précis n'est composé que de surfaces planes simples. Aucune d'elles ne travaille directement à tous les niveaux sur la géométrie exacte des primitives originales.
- Étant donné que la subdivision hiérarchique des surfaces complexes ne découle pas directement des algorithmes de subdivision utilisés sur les carrés, un grand nombre de valeurs géométriques telles que l'aire, le champ vectoriel des normales, les boîtes englobantes, etc., doivent être recalculées et stockées pour chaque élément de surface de la hiérarchie. Une telle contrainte ralentit considérablement l'algorithme de radiosité et diminue son efficacité.

3.2.3 Nos contributions

Avant de présenter dans les détails notre approche du problème grâce au maillage virtuel, nous présentons ici les avantages de notre méthode par rapport aux approches dont

nous venons de parler. Le but du maillage virtuel est de pouvoir procéder à l'illumination de surfaces paramétriques au moyen des méthodes de radiosité classiques.

- **Contribution théorique** — le maillage virtuel permet d'intégrer de façon naturelle les primitives courbes dans tous les algorithmes classiques de radiosité. De plus, il permet de faire le lien entre les méthodes de GALERKIN hiérarchiques utilisées en radiosité et les surfaces paramétriques, alors que ces approches étaient jusque là incompatibles. Enfin, contrairement à la méthode de radiosité hiérarchique classique, notre implantation du maillage virtuel bénéficie pleinement du fort pouvoir de compression des bases de fonctions ondelettes d'ordre élevé dont nous avons prouvé l'efficacité dans le chapitre 2 (p. 41).
- **Généricité** — étant donné que le maillage virtuel fournit une représentation « normalisée » de la géométrie initiale au moyen d'un ensemble de fonctions de projection bijectives qui permettent de faire des allers et retours entre les supports réels et virtuels, notre méthode peut être implantée dans n'importe quel algorithme de radiosité. Par exemple, son utilisation dans un algorithme à base de *clustering* permettrait une meilleure manipulation des primitives géométriques complexes à l'intérieur même des *clusters* et accélérerait les calculs pour des transferts énergétiques entre des surfaces proches. En d'autres termes, le maillage virtuel permet de concentrer la puissance de calcul et les efforts de développement aux endroits où l'algorithme en a réellement besoin.
- **Apport en précision** — en conservant tout au long du processus de résolution une représentation virtuelle de la géométrie qui correspond exactement à la géométrie initiale, le maillage virtuel permet d'éviter la plupart des erreurs dues aux approximations géométriques. Ces erreurs dégradaient fortement les résultats de radiosité quand l'algorithme était utilisé sur des surfaces complexes telles que les surfaces courbes.
- **Apport en efficacité** — en utilisant directement dans l'algorithme des représentations plus naturelles des surfaces, le maillage virtuel permet d'atteindre des gains d'efficacité allant de 40 % à 70 % en nombre de surfaces initiales. Ces gains engendrent directement une diminution du temps de calcul qui représente entre 50 % et 70 % du temps total de simulation. Grâce à ces améliorations en efficacité, le maillage virtuel permet d'effectuer en un temps raisonnable et avec des capacités de calcul d'ores et déjà existantes, des simulations de radiosité à base d'ondelettes de grande précision sur des modèles de très grande taille.

3.3 Principe du maillage virtuel

Dans la section précédente, nous avons donné les principales raisons pour lesquelles les algorithmes de radiosité travaillent généralement sur des scènes ne contenant que des primitives géométriques simples. Pour résumer ces raisons, on peut dire que ce choix est dû au besoin omniprésent dans l'algorithme de radiosité d'effectuer des calculs sur les surfaces de l'environnement en se ramenant à des domaines normalisés.

Nous montrons dans cette section qu'il est possible de gérer aussi efficacement que les triangles et les parallélogrammes toutes les primitives planes (avec ou sans trous et présentant des bords quelconques) ainsi que les surfaces courbes paramétriques. Pour cela, il faut supprimer l'étape classique de triangulation et la remplacer par la projection de ces surfaces sur des morceaux de plans. En d'autres termes, il faut enrichir la représentation géométrique au moyen d'une deuxième description des primitives initiales. Cette deuxième description est virtuelle car elle ne correspond pas directement aux surfaces réelles, mais elle est surtout mathématiquement beaucoup plus simple que la description originale. C'est sur cette deuxième description que tous les calculs surfaciques seront effectués tout au long de l'algorithme.

Dans cette section, nous présentons les principes généraux de notre méthode. Cette présentation est faite indépendamment de tout algorithme de résolution particulier pour le calcul de radiosité. Nous présentons ensuite toutes les implications qui gravitent autour de son utilisation. La première partie de cette section est consacrée à la description de l'algorithme permettant de passer des primitives quelconques aux mailles virtuelles planes. Nous illustrons le principe général au moyen d'un exemple simple dont la primitive originale est un morceau de tore. Dans la deuxième section, nous expliquons comment étendre la fonction de radiosité à toute la surface normalisée de chaque support virtuel. Enfin, dans la dernière section, nous voyons comment déduire, à partir des valeurs calculées sur le maillage virtuel, la radiosité sur les primitives originales.

3.3.1 Passage des primitives quelconques aux carrés normalisés

L'idée principale qui se cache derrière notre méthode de maillage virtuel est de calculer la fonction radiosité non pas directement sur les primitives géométriques originales mais sur des supports normalisés plans qui ont une correspondance exacte avec les surfaces initiales. Afin de conserver un sens physique aux mailles virtuelles, les fonctions de projection permettant de passer des surfaces réelles aux mailles virtuelles doivent conserver les propriétés énergétiques des surfaces. Comme nous le montrons dans les sections 3.4 (p. 108) et 3.6 (p. 125) cette propriété d'invariance est cruciale lorsque l'on utilise des bases de fonctions non constantes dans un algorithme hiérarchique basé sur l'approche de GALERKIN et elle permet d'augmenter de façon très importante la vitesse des calculs. Or, comme nous l'avons démontré au chapitre précédent, les ondelettes d'ordre élevé sont réellement très efficaces et nous sommes convaincus qu'il est indispensable de pouvoir les utiliser dans tous les futurs développements en radiosité.

Du point de vue mathématique, étant donné que la radiosité mesure une puissance par unité de surface, dire que les fonctions de projection conservent les propriétés énergétiques des surfaces revient à dire qu'elles conservent les rapports d'aire entre les primitives originales et les mailles virtuelles. Cette propriété suppose donc que le Jacobien¹³ de chaque fonction de projection est constant. Du fait que la projection directe des primitives complexes sur des carrés normalisés en conservant les propriétés énergétiques n'est pas toujours

13. Les définitions et les propriétés précises du Jacobien de fonctions en 2D pourront être trouvées dans [Kap 84].

possible, nous avons découpé ce processus en trois étapes distinctes :

- trouver une fonction de projection à Jacobien constant qui projette la primitive complexe S sur un morceau de plan (voir la section 3.3.1.1 (p. 97)),
- inclure l'image S' de S dans un parallélogramme S'' (voir la section 3.3.1.2 (p. 99)),
- projeter le parallélogramme S'' sur un carré normalisé \tilde{S} : la maille virtuelle (voir la section 3.3.1.3 (p. 101))

3.3.1.1 Passage des primitives courbes à des domaines plans

L'étape décrite dans cette section n'est nécessaire que pour les primitives géométriques courbes. Trouver une fonction de projection qui « aplatit » une surface courbe et qui a un Jacobien constant est un point clé de notre méthode. Cette fonction permet d'effectuer tous les calculs de radiosité sur un support géométrique plus simple tout en préservant les propriétés énergétiques de la primitive de départ. Cette section nous permet de donner une méthode générale pour trouver ce genre de fonctions et nous illustrons cette méthode par un exemple simple.

Le cas général : Considérons une surface paramétrique courbe décrite par ses équations :

$$\mathbf{x}(\theta, \varphi) = \begin{pmatrix} x(\theta, \varphi) \\ y(\theta, \varphi) \\ z(\theta, \varphi) \end{pmatrix},$$

où θ et φ sont les « paramètres naturels » de la surface (ce sont, par exemple, la hauteur et l'angle d'ouverture pour la description d'un cylindre).

Ce paramétrage correspond en fait à une fonction de projection $\mathcal{F} : (\theta, \varphi) \mapsto (x, y, z)$ dont le Jacobien est égal à :

$$J(\mathcal{F}) = \left\| \frac{\partial \mathbf{x}}{\partial \theta} \times \frac{\partial \mathbf{x}}{\partial \varphi} \right\|.$$

Malheureusement, ce Jacobien n'est généralement pas constant. Trouver une fonction de projection ayant un Jacobien constant revient donc à trouver un nouveau paramétrage $\mathbf{x}(u, v)$ tel que le Jacobien $J(\Psi)$ de la fonction de projection $\Psi : (u, v) \mapsto (x, y, z)$ soit constant.

On peut alors exprimer θ et φ en fonction des paramètres naturels u et v . Dans le cas général, on a :

$$\begin{pmatrix} \theta \\ \varphi \end{pmatrix} = \begin{pmatrix} f(u, v) \\ g(u, v) \end{pmatrix}.$$

Mais pour simplifier les calculs, on essaye toujours d'utiliser ce nouveau paramétrage en imposant à la fonction f de ne dépendre que d'un paramètre. On est donc ramené à trouver des paramètres u et v tels que :

$$\begin{pmatrix} \theta \\ \varphi \end{pmatrix} = \begin{pmatrix} f(u) \\ g(u, v) \end{pmatrix} \text{ et } J(\Psi) = \left\| \frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} \right\| = c = \text{cste.}$$

En utilisant les propriétés sur les dérivés des fonctions composées, on en déduit qu'il faut :

$$\left\| \frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} \right\| = \left| \frac{\partial \theta}{\partial u} \frac{\partial \varphi}{\partial v} \right| \left\| \frac{\partial \mathbf{x}}{\partial \theta} \times \frac{\partial \mathbf{x}}{\partial \varphi} \right\| = c. \quad (3.1)$$

Étant donné que

$$\frac{\partial \theta}{\partial u} = f'(u) \quad \text{et} \quad \frac{\partial \varphi}{\partial v} = \frac{\partial g}{\partial v}(u, v)$$

si on utilise le théorème sur la dérivée des fonctions opposées, l'équation (3.1) devient (au signe près que l'on absorbe dans la constante c) :

$$\frac{\partial v}{\partial \varphi}(\theta, \varphi) = \frac{1}{\frac{\partial g}{\partial v}(u, v)} = \frac{1}{c} |f'(u)| \left\| \frac{\partial \mathbf{x}}{\partial \theta} \times \frac{\partial \mathbf{x}}{\partial \varphi} \right\|$$

En intégrant cette équation par rapport à φ et en remplaçant respectivement θ et φ par $f(u)$ et $g(u, v)$, on obtient donc :

$$v = \frac{1}{c} |f'(u)| \int_0^{g(u, v)} \left\| \frac{\partial \mathbf{x}}{\partial \theta}(f(u), t) \times \frac{\partial \mathbf{x}}{\partial \varphi}(f(u), t) \right\| dt. \quad (3.2)$$

A priori tous les paramétrages qui satisfont à l'équation (3.2) sont appropriés pour définir notre fonction de projection $\psi : (u, v) \mapsto (x, y, z)$. On peut généralement trouver un grand nombre de tels paramétrages.

Pour pouvoir effectuer efficacement toutes les opérations surfaciques nécessaires dans l'algorithme de radiosité (calcul du maillage virtuel à partir des primitives originales, passage des informations de visibilité des primitives originales au maillage virtuel, passage des informations topologiques — normales — ou physiques — couleur — de la géométrie initiale au maillage virtuel, reconstruction de la fonction de radiosité sur les primitives originales à partir des données stockées sur le maillage virtuel, etc.), nous avons besoin de pouvoir faire des allers et retours entre les trois paramétrages : (x, y, z) , (θ, φ) et (u, v) . Nous avons donc besoin de connaître et de pouvoir évaluer facilement les valeurs des fonctions de projection inverses. Bien que cela puisse paraître compliqué, toutes les fonctions de projection sont en fait bijectives (sauf éventuellement en un certain nombre de points dont la mesure au sens de LEBESGUE est nulle), ce qui permet d'utiliser des approximations numériques très efficaces quand les fonctions inverses sont difficiles à exprimer mathématiquement. De plus, toutes les fonctions de projection que nous utilisons par la suite font appel aux fonctions circulaires et hyperboliques pour lesquelles la projection inverse $(x, y, z) \mapsto (\theta, \varphi)$ peut être exprimée de façon exacte.

Un exemple concret, un tore : Considérons le morceau de tore présenté sur la figure 3.5(a). Mathématiquement, ce morceau de tore correspond au paramétrage suivant :

$$\begin{cases} x(\theta, \varphi) = \cos \theta (\cos \varphi + 2), \\ y(\theta, \varphi) = \sin \theta (\cos \varphi + 2), \\ z(\theta, \varphi) = \sin \varphi, \end{cases} \quad \theta \in \left[0, \frac{\pi}{2}\right), \varphi \in [0, \pi).$$

Choisissons par exemple comme fonction $f(u) = u/2$ et comme constante $c = 1$ (nous rappelons que d'après les conventions précédentes : $\theta = f(u)$, $\varphi = g(u,v)$ et $\mathcal{J}(\Psi) = c$). L'équation (3.2) nous donne donc une contrainte sur la fonction g :

$$2v = \sin(g(u,v)) + 2g(u,v). \quad (3.3)$$

On peut tout de suite remarquer que dans ce cas g ne dépend que du paramètre v . On peut donc écrire : $g(u,v) = g(v)$. Avec l'équation (3.3), on peut déterminer g de façon numérique et on peut donc la réinjecter dans le paramétrage du morceau de tore pour obtenir un paramétrage qui le projette sur un plan tout en conservant ses propriétés énergétiques :

$$(u,v) \mapsto \left(\cos \frac{u}{2} (\cos g(v) + 2), \sin \frac{u}{2} (\cos g(v) + 2), \sin g(v) \right).$$

De plus, il est facile de calculer de façon exacte le passage $(x,y,z) \mapsto (\theta,\varphi)$ ainsi que le passage inverse. Il est donc aisé, une fois ce nouveau paramétrage établi, de passer successivement de l'un à l'autre des trois paramétrages : (x,y,z) , (θ,φ) et (u,v) .

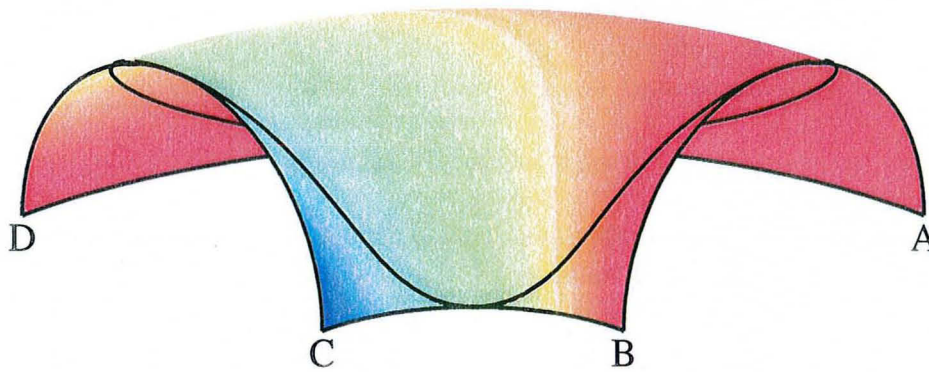
Le résultat de la projection de notre morceau de tore au moyen de la fonction $\psi^{-1} : (x,y,z) \mapsto (u,v)$ est présenté sur la figure 3.5(b). La courbe tracée sur la figure permet de comparer cette projection à Jacobien constant avec la projection naturelle à Jacobien non constant obtenue en utilisant la fonction $\mathcal{F}^{-1} : (x,y,z) \mapsto (2\theta,\varphi)$ et présentée sur la figure 3.5(c).

La simplicité de la fonction de projection à Jacobien constant pour le tore ne doit pas laisser croire que toutes les surfaces présentant un paramétrage naturel « simple » ont une fonction de projection à Jacobien constant « simple ». En effet, comme nous le verrons plus en détail dans la section 3.5 (p. 115), même des surfaces aussi naturelles que des ellipsoïdes peuvent nécessiter des fonctions de projection relativement compliquées.

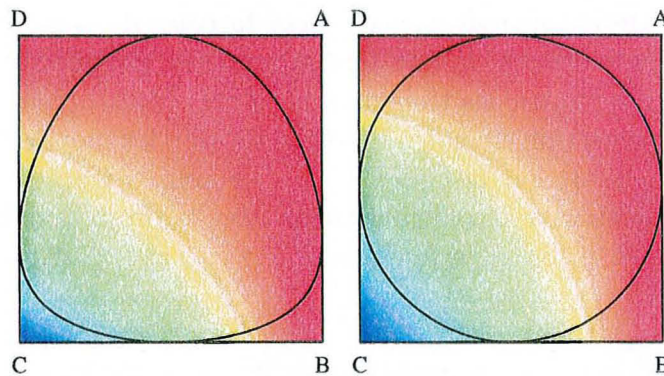
3.3.1.2 Passage des domaines plans quelconques aux parallélogrammes englobants

Nous venons de voir qu'il est possible de remplacer une surface courbe S par sa représentation sur un morceau de plan S' . Malheureusement, la primitive plane S' ainsi générée n'a aucune raison d'être un carré parfait. Bien au contraire, cette primitive peut avoir des bords quelconques, elle peut contenir des trous, avoir une forme polygonale ou même présenter des contours courbes. Si on considère, par exemple, la surface délimitée par la courbe tracée sur le tore de la figure 3.5(a) on constate que sa projection à Jacobien constant présentée sur la figure 3.5(b) a une forme de patatoïde. Dans la grande majorité des cas, il n'est pas possible de projeter une surface courbe en conservant ses propriétés énergétiques sur un carré normalisé. Le cas idéal d'une projection sur un carré peut cependant se présenter dans un certain nombre de situations tel que c'est le cas lorsque l'on considère le morceau de tore complet de la figure 3.5(a).

Par ailleurs, comme nous l'avons déjà souligné dans la section précédente, les surfaces planes à bords quelconques ne sont pas bien gérées dans les algorithmes classiques de radiosité.



(a) Morceau de tore



(b) Projection à Jacobien constant

(c) Une projection avec les paramètres naturels: Jacobien non constant

FIG. 3.5 – Projection d'un morceau de tore avec une courbe dessinée dessus.

L'idée que nous proposons ici pour manipuler les primitives planes à bords quelconques est de travailler sur des « domaines étendus » S'' englobant les primitives planes. Les domaines étendus doivent vérifier trois propriétés pour être utilisables efficacement dans l'algorithme de radiosité :

- ils doivent englober entièrement les primitives planes,
- ils doivent minimiser les vides,
- ils doivent avoir une forme bien adaptée au calcul de radiosité.

La troisième condition associée aux considérations de la section précédente impose que les domaines étendus soient des parallélogrammes. Pour satisfaire au deuxième critère ci-dessus, nous avons décidé de toujours choisir le parallélogramme englobant d'aire minimale. Plusieurs parallélogrammes peuvent répondre à ces critères. On choisit de conserver celui dont les angles sont les plus proches de $\pi/2$ pour éviter d'avoir une trop forte distorsion lorsque l'on le projette sur un carré normalisé.

L'algorithme 6 décrit notre approche pour construire le parallélogramme englobant.

Pour traiter le cas d'une primitive non polygonale, on commence par en faire une approximation polygonale en discrétisant ses contours. Il suffit donc de faire varier le paramètre de façon discrète. On calcule ensuite le parallélogramme englobant en utilisant cette approximation. À la fin du processus, on utilise les points extrêmes de la primitive originale pour étirer le parallélogramme jusqu'à ce qu'il englobe tous ces points. Cette méthode ne donne pas toujours le meilleur résultat possible mais elle est largement suffisante et le parallélogramme ainsi généré convient toujours très bien pour les calculs de radiosit . Un point cl  de notre algorithme est que sa complexit  est en $O(n)$ o  n est le nombre de points initiaux utilis s pour d crire le contour de la primitive. En effet, l'enveloppe convexe d'une cha ne de points peut  tre calcul e avec une complexit  $O(n)$ gr ce   l'algorithme d crit par Herbert EDELSBRUNNER dans [Ede 87], et l'algorithme de Christian SCHWARZ *et al.* d crit dans [Sch 95] permet de calculer le parall logramme englobant d'aire minimale pour un polygone convexe avec une complexit  $O(n)$.

Algorithme 6 Construction du parall logramme englobant

ENTR ES: Primitive p
SORTIES: Parall logramme *r sultat*

```

1: si  $p \langle \rangle$  Polygone alors
2:    $p' = \text{approximation\_polygonale}(p)$ 
3: sinon
4:    $p' = p$ 
5: fin si
6:  $env = \text{enveloppe\_convexe}(p')$  // Edelsbrunner [Ede 87]
7:  $parall = \text{parall logramme\_englobant}(env)$  // Schwarz [Sch 95]
8: si  $p \langle \rangle$  Polygone alors
9:    $extr = \text{extr mums}(p)$ 
10:   $result = \text{parall logramme\_ tendu}(parall)$  tel que  $extr \in result$ 
11: sinon
12:   $result = parall$ 
13: fin si
14: retourne  $result$ 

```

Consid rions, par exemple, la primitive de la figure 3.6(a) : le parall logramme de la figure 3.6(b) a des angles trop  loign s de $\pi/2$, celui de la figure 3.6(c) n'a pas deux c t s adjacents qui co ncident avec des ar tes de l'enveloppe convexe comme l'impose l'algorithme de Christian SCHWARZ *et al.* C'est donc le parall logramme de la figure 3.6(d) qui est choisi comme parall logramme englobant.

3.3.1.3 Passage des parall logrammes aux carr s normalis s

La derni re  tape du processus de cr ation du maillage virtuel est la projection du parall logramme S'' sur un carr  normalis  \tilde{S} . Cette  tape est relativement simple car elle

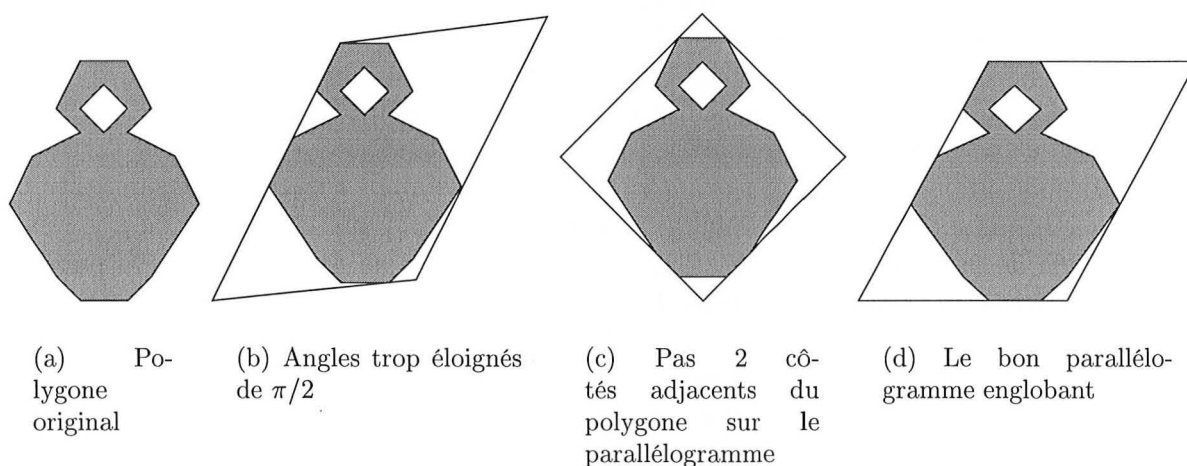


FIG. 3.6 – Exemples de parallélogrammes englobants.

peut être effectuée en utilisant une fonction de projection bi-linéaire Ψ dont l'inverse est donné par l'équation ci-dessous :

$$(\theta, \varphi) = \Psi^{-1}(u, v)_{\text{carré}} = P_0 + u\overrightarrow{P_0P_1} + v\overrightarrow{P_0P_3},$$

où les points P_0, P_1, P_2, P_3 sont ceux présentés sur la figure 3.7. Le Jacobien d'une telle projection est constant et égal à l'aire du parallélogramme S'' .

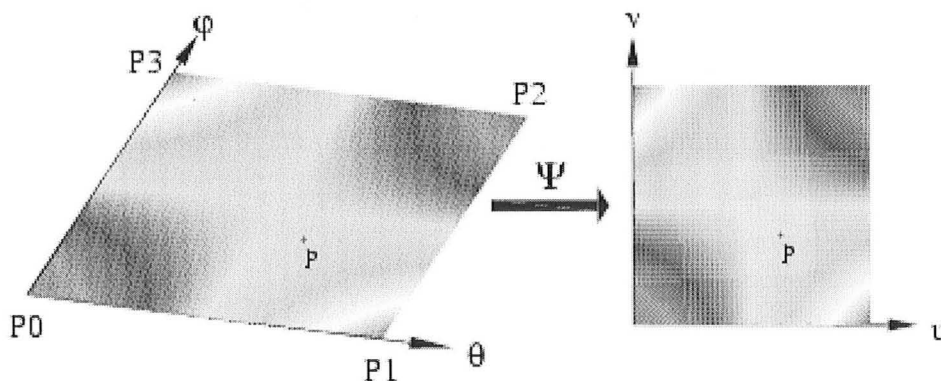


FIG. 3.7 – Projection d'un parallélogramme sur un carré normalisé.

3.3.2 Fonction de radiosit  sur le maillage virtuel

En g n ral, les points des mailles virtuelles normalis es \tilde{S} ne correspondent pas tous   des points des primitives originales S . Avant de calculer la fonction de radiosit  sur le maillage virtuel, il convient d' tendre la d finition de la fonction de radiosit  de son support naturel S   tous les points de \tilde{S} .

3.3.2.1 Définir la fonction de radiosité sur \tilde{S}

La solution que nous avons choisie pour définir la fonction de radiosité en dehors des points de S est représentée sur la figure 3.8.

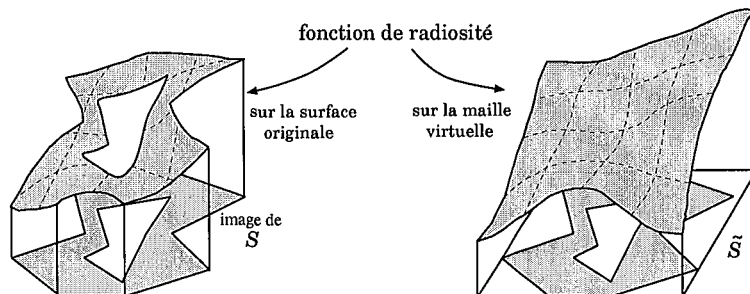


FIG. 3.8 – Étendre la fonction de radiosité de façon continue.

Cette extension de la définition de la fonction de radiosité obéit à deux principes que nous nous sommes fixés :

- La fonction étendue correspond exactement à la fonction de radiosité originale sur tous les points intérieurs à S ;
- La fonction étendue est aussi simple et aussi continue que possible aux frontières de S et sur les points extérieurs à S .

Le premier de ces deux principes est évident car nous voulons avoir la possibilité de relire les valeurs de la fonction de radiosité sur la primitive originale S à partir de celles calculées sur \tilde{S} . Le deuxième principe est moins évident mais il est tout aussi important car, dans un algorithme hiérarchique, la fonction de radiosité calculée en un point quelconque du domaine peut influencer la représentation en d'autres points. Il faut donc calculer la radiosité avec la même précision en tous les points du domaine et notamment sur les points extérieurs à S .

Étant donné que les discontinuités de la fonction de radiosité et de ses dérivées ne sont dues qu'à des variations de la visibilité entre les objets de la scène (comme cela est expliqué par Paul HECKBERT dans [Hec 92]), nous avons cherché, pour satisfaire au deuxième principe, à ne pas introduire de nouvelles discontinuités de visibilité dans notre extension. Pour faire cela, nous avons défini une *fonction de visibilité étendue* \tilde{V} de la façon suivante. La visibilité entre un point q dans l'environnement et un point p sur la maille virtuelle \tilde{S} est égale à la visibilité entre q et \tilde{p} , où \tilde{p} est le point le plus proche de p sur la primitive originale S :

$$\begin{aligned}\tilde{V}(q,p) &= V(q,\tilde{p}) \\ \|\tilde{p} - p\| &= \min_{p' \in S} \|p' - p\|.\end{aligned}$$

Bien sûr, si p est un point de S alors $\tilde{p} = p$. Dans ce cas, la fonction de visibilité étendue est égale à la fonction de visibilité naturelle.

La fonction de radiosité sur la maille virtuelle est alors définie comme la fonction de radiosité classique calculée comme si la maille virtuelle avait remplacé la primitive originale. Le seul changement est que la fonction de visibilité V de l'équation de radiosité 1.4

est remplacée par la fonction de visibilité étendue \tilde{V} :

$$B(x) = E(x) + \rho(x) \int_S B(y) K(x,y) dy$$

$$K(x,y) = G(x,y) \tilde{V}(x,y).$$

Cette nouvelle définition de la fonction de visibilité n'est peut être pas la meilleure dans toutes les situations mais elle a l'avantage d'être simple à mettre en œuvre. De plus, elle s'est montrée tout à fait pertinente sur l'ensemble des tests que nous avons effectué. De même que pour le calcul du parallélogramme englobant, la recherche du point \tilde{p} , le plus proche de p sur S , se fait en utilisant une approximation polygonale de \tilde{S} si la projection S' sur un plan de la primitive originale n'est pas un polygone.

3.3.2.2 Utilisation de \tilde{S} pour le calcul de radiosité

Maintenant que l'on a défini la façon d'étendre la fonction de radiosité à toute la maille virtuelle, on peut utiliser cette définition pour calculer la radiosité sur cette maille \tilde{S} . Rappelons encore une fois que le but ultime est de relire les valeurs de la fonction de radiosité sur la primitive originale S à partir de celles calculées sur \tilde{S} . Il faut donc calculer la fonction sur \tilde{S} de telle façon qu'elle ait un réel sens physique sur S . Intuitivement, l'idée est d'utiliser la primitive originale uniquement quand cela est vraiment nécessaire et de profiter de la définition très simple de la maille virtuelle aussi souvent que cela est possible. On peut alors distinguer trois situations différentes :

- S peut être un récepteur,
- S peut être un bloqueur,
- S peut être un émetteur.

Récepteur : Quand la primitive S reçoit de l'énergie, on utilise la maille virtuelle \tilde{S} exactement comme si c'était la surface réceptrice. Tous les points de \tilde{S} sont traités exactement avec la même précision, qu'ils soient situés à l'intérieur ou à l'extérieur de la primitive originale S . De cette façon, la fonction de radiosité reste cohérente sur toute la surface de la maille virtuelle et elle peut être utilisée dans un algorithme hiérarchique sans que les points extérieurs à S ne viennent perturber ceux à l'intérieur de S .

Bloqueur : Lorsque la primitive S est considérée en tant que bloqueur, elle intervient dans le calcul de la visibilité entre deux points et il faut alors utiliser sa définition initiale pour faire les calculs. La maille virtuelle n'a aucun sens dans les calculs de visibilité.

Pour détecter efficacement si la primitive originale est bloquante dans une interaction, nous procédons par étapes :

- Si la primitive est courbe, elle est généralement définie comme le morceau d'un support dont on connaît exactement une représentation paramétrique. On utilise alors cette représentation pour calculer l'intersection du rayon qui nous intéresse avec le support de la primitive, puis, si cette intersection existe, on la projette au moyen de notre fonction de projection à Jacobien constant sur la maille virtuelle.

- Si la primitive est plane, on utilise directement les coordonnées 2D de l'intersection du rayon avec la primitive, coordonnées que l'on projette sur la maille virtuelle.
- Une fois que l'on connaît les coordonnées de l'intersection sur la maille virtuelle, on utilise la carte trapézoïdale d'un arrangement de segments dans le plan, définie respectivement par Jean-Daniel BOISSONNAT et Mariette YVINEC dans [Boi 98] et par Olivier DEVILLERS *et al.* dans [Dev 92] et dont un exemple est présenté sur la figure 3.9, pour savoir si le point est à l'intérieur ou à l'extérieur de la primitive originale.

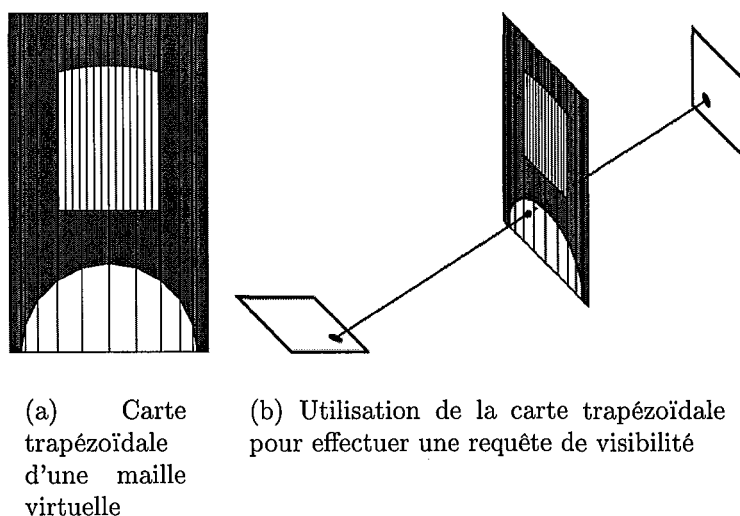


FIG. 3.9 – Carte trapézoïdale d'un arrangement de segments dans le plan.

Les segments utilisés pour construire cette carte sont ceux définissant le contour de la primitive originale. Nous avons enrichi l'algorithme de construction pour intégrer les segments qui ne sont pas des segments de droite. Pour cela, nous utilisons pour chaque segment courbe ses points extrêmes définis dans les directions de la maille virtuelle.

Pour pouvoir faire nos requêtes de façon efficace, on stocke, lors de la création de la carte trapézoïdale, un booléen pour chaque trapèze. Ce booléen est positionné à *vrai* si le trapèze est à l'intérieur de la primitive originale et à *faux* s'il est à l'extérieur.

En utilisant un algorithme aléatoire, la carte trapézoïdale peut être construite en un temps $O(n \log n)$, où n est le nombre de segments initiaux. Une fois construite, la carte permet de répondre aux requêtes de visibilité en un temps $O(\log n)$. Puisque l'algorithme de construction est aléatoire, nous mélangeons les segments avant de commencer à construire la carte.

Les requêtes de visibilité dans notre algorithme sont des requêtes point à point comme nous l'avons vu dans le chapitre précédent. Pour effectuer une requête de visibilité entre deux points, on utilise alors la procédure ci-dessus avec le rayon défini par ces deux points.

Émetteur : Quand la primitive S est considérée en tant qu'émetteur, il faut faire attention au fait que la radiosité accumulée sur la maille virtuelle \tilde{S} représente plus d'énergie

que l'énergie réellement reçue par la primitive originale. En effet, les points situés en dehors de la maille originale ont accumulé de l'énergie qui n'existe pas en réalité. Il faut donc éviter de repropager cette énergie dans la scène car sinon on violerait le principe de conservation de l'énergie en créant artificiellement une augmentation énergétique dans l'environnement. Afin d'éviter de propager ce surplus d'énergie, nous utilisons la fonction stockée sur la maille virtuelle mais nous pondérons les poids utilisés dans les calculs de quadrature pour qu'ils intègrent ce phénomène.

Étant donné que les poids de quadrature en 1D sont des valeurs positives et que leur somme est égale à 1, on peut considérer qu'ils représentent chacun la longueur d'une zone d'influence pour le point de quadrature correspondant. Nous avons représenté cette zone en 1D sur la figure 3.10(a). Cette constatation peut être transposée en 2D (voir figure 3.10(b)) ; le poids du point de quadrature $p_{i,j}$ est $\omega_i\omega_j$ ¹⁴.

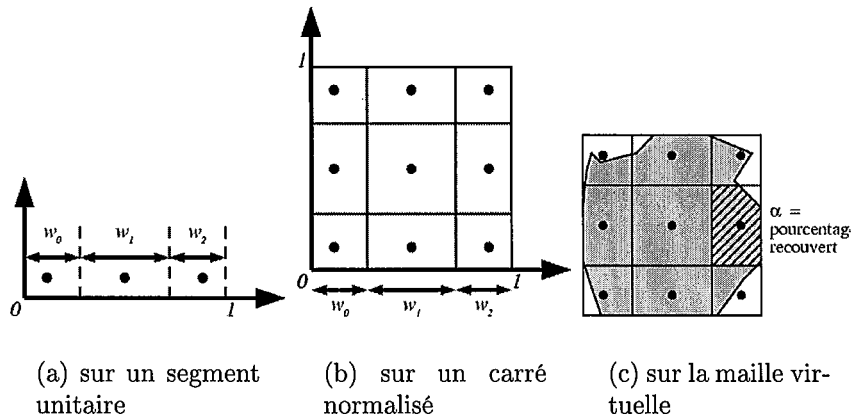


FIG. 3.10 – Poids des points de quadrature peuvent être vus comme l'aire de leur zone d'influence

Pour intégrer le phénomène du surplus énergétique directement dans les poids de quadrature, nous proposons de multiplier ces derniers par la fraction :

$$\frac{\text{aire de la section de la maille virtuelle à l'intérieur de la zone d'influence}}{\text{aire de la zone d'influence elle-même}}$$

Par exemple, sur la figure 3.10(c), le poids du point de quadrature dans la zone hachurée devrait être $\omega_1\omega_2$. Comme la fraction de l'aire effectivement occupée par la maille originale est égale à α , nous utilisons comme poids pour ce point $\alpha\omega_1\omega_2$ dans tous les calculs d'émission.

La solution que nous proposons n'est pas idéale car elle ne permet pas de prendre en compte la représentation complexe des fonctions d'ordre élevé. Cependant, pour des calculs d'interaction de faible précision, cette solution est extrêmement efficace car elle est simple à implanter et très rapide dans son utilisation. De plus, pour des calculs de grande précision, cette définition permettra de tendre vers la solution exacte. En effet, pour une

14. On notera que les zones d'influence des points de quadrature ne correspondent pas aux zones définies par le diagramme de VORONOÏ des points de quadrature.

grande précision, les subdivisions successives auront tendance à découper plus fortement les zones où la fonction varie fortement et l'importance du facteur α sera donc diminuée. Pour les zones où la surface originale remplit entièrement ou ne remplit pas du tout la zone d'influence, les calculs par cette méthode sont exacts.

Si, dans certaines situations, un point de quadrature tombe en dehors de la surface originale, nous l'utilisons malgré tout sans rien changer dans nos calculs.

L'algorithme 7 montre un pseudo code de notre algorithme d'émission en utilisant la maille virtuelle.

Algorithme 7 Pseudo code de notre algorithme d'émission à partir d'une primitive complexe en utilisant la maille virtuelle

ENTRÉES: Primitive S /* émetteur complexe */ , Primitive R /* récepteur */

```

1:  $M = \text{Maille\_Virtuelle}(S)$ 
2: pour tout interaction  $S \rightarrow R$  faire
3:   pour tout  $q_i = \text{point\_quadrature}(M)$  faire
4:      $\alpha_i \leftarrow \text{fraction\_zone\_influence}(S, M, q_i)$ 
5:      $q'_i = \text{point\_le\_plus\_proche}(q_i, S)$ 
6:     pour tout  $p_j = \text{point\_quadrature}(R)$  faire
7:        $p'_j = \text{point\_le\_plus\_proche}(p_j, R)$ 
8:        $V(q'_i, p'_j) = \text{visibilité}(p'_j, q'_i)$ 
9:        $G(q_i, p_j) = \text{valeur\_du\_noyau}(p_j, q_i)$ 
10:       $B_r + = \alpha_i w_i w_j B_s(q_i) V(q'_i, p'_j) G(q_i, p_j)$ 
11:    fin pour
12:  fin pour
13: fin pour

```

3.3.3 Déduire la radiosité sur les primitives originales

Les deux étapes que nous venons de décrire permettent de calculer la fonction de radiosité sur le maillage virtuel avec la précision et le niveau de convergence définis dans l'algorithme utilisé. Ceci étant, nous avons à notre disposition la valeur exacte de la radiosité en chaque point du maillage hiérarchique pour chacune des mailles virtuelles. La visualisation des primitives originales illuminées, c'est-à-dire avec les couleurs correspondant à la fonction de radiosité ainsi calculée, est très simple à partir de ce point. En effet, il suffit pour chaque primitive S de reprojeter la partie qui coïncide effectivement avec des points de S de la maille virtuelle \tilde{S} correspondante en utilisant la fonction de projection inverse $\tilde{S} \rightarrow S'' \rightarrow S$. La primitive originale peut alors être envoyée au moteur d'affichage dans le bon format en utilisant les valeurs ainsi récupérées sur la maille virtuelle.

L'affichage des surfaces courbes au moyen des cartes graphiques actuelles pose encore quelques problèmes car ces dernières n'ont que la possibilité d'afficher des triangles. Mais la description mathématique des primitives courbes permet de les envoyer à la carte

graphique sous une forme triangulée avec la précision désirée. On peut alors développer un algorithme intelligent qui génère automatiquement des niveaux de détails pour afficher les surfaces courbes. Les chercheurs de l'équipe ISA expérimentent actuellement ce genre d'algorithmes qui permettent d'avoir une précision d'affichage presque infinie sans jamais atteindre les limites des matériels graphiques.

3.4 Maillage virtuel et radiosit      base d'ondelettes

Le maillage virtuel s  pare les donn  es g  om  triques de la repr  sentation de la fonction de radiosit  . Il fournit ainsi une nouvelle repr  sentation abstraite de la g  om  trie. Cette nouvelle repr  sentation est li  e aux donn  es initiales pour pouvoir les interroger chaque fois qu'il est n  cessaire d'avoir des informations physiques sur la g  om  trie (normale, visibilit  , intersection, etc.). Comme cette repr  sentation est largement ind  pendante de la nature de l'algorithme de radiosit   utilis  , elle peut   tre implant  e dans n'importe quel algorithme existant. En particulier, elle est tout    fait adapt  e pour   tre utilis  e dans un algorithme de radiosit   hi  rarchique utilisant une m  thode de GALERKIN. Nous avons justement implant   le maillage virtuel dans notre algorithme de radiosit      base d'ondelettes d  crit au chapitre 2 (p. 41). Nous pr  sentons ici les points particuliers pour lesquels il a fallu modifier l'algorithme pour l'adapter    cette nouvelle repr  sentation de la g  om  trie.

3.4.1 Algorithme

L'algorithme 8 pr  sente dans ses grandes lignes notre algorithme de radiosit      base d'ondelettes lorsqu'il est associ   au maillage virtuel. Il correspond exactement    l'algorithme de SOUTHWELL d  crit dans le chapitre pr  c  dent. Nous avons soulign   dans l'algorithme 8 les diff  rences avec l'algorithme classique et, comme on peut le constater, elles sont tr  s peu nombreuses. Nous revenons sur ces quelques diff  rences dans les lignes qui suivent.

Comme nous l'avons d  j   soulign  , notre implantation de l'algorithme de radiosit   associe un algorithme de *shooting* qui ne stocke pas les liens aux fonctions de base les plus g  n  rales actuellement utilis  es : les ondelettes d'ordre   lev  . Les ondelettes sont des fonctions hi  rarchiques qui permettent de maintenir une repr  sentation de la fonction de radiosit   avec plusieurs niveaux de pr  cision en gardant une r  elle coh  rence entre tous ces niveaux. Or, les ondelettes sont r  ellement efficaces quand les subdivisions successives permettant de cr  er les niveaux hi  rarchiques engendrent des mailles de formes identiques et conservent toujours les m  mes rapports g  om  triques entre deux niveaux successifs. Le maillage virtuel permet pr  cis  ment d'apporter cette propri  t   pour des primitives complexes qui n'auraient pas pu   tre utilis  es facilement autrement.

Lorsque l'on travaille avec des surfaces courbes, ces derni  res peuvent pr  senter des grandes variations dans leur courbure et donc avoir un champ de normales tr  s   tendu. Il est alors important de pouvoir augmenter l'ordre des fonctions de base pour mieux approcher les variations de la fonction de radiosit   et pour, ainsi, r  duire le nombre de

Algorithme 8 Algorithme de radiosité à base d'ondelettes avec maillage virtuel.

```

1: pour tout  $i \in \{1, \dots, n\}$  faire
2:    $\tilde{S}_i \leftarrow \text{Maille Virtuelle}(S_i)$ 
3:    $\tilde{B}_i \leftarrow \tilde{E}_i, \Delta\tilde{B}_i \leftarrow \tilde{E}_i$ 
4: fin pour
5: tant que  $\sum_k \int_{S_{k,\lambda}} \Delta\tilde{B}_k > \varepsilon$  faire
6:   Choisir  $i$  tel que  $\tilde{S}_i$  ait la plus grande valeur d'énergie non propagée  $\Delta\tilde{B}_i$ 
7:    $\text{Push/Pull-Résidus}(\tilde{S}_i)$ 
8:   pour tout  $j \in \{1, \dots, n\}$  faire
9:      $\tilde{E}_j \leftarrow \text{HR-Transfert}(S_i \rightarrow \tilde{S}_j, \Delta\tilde{B}_i)$ 
10:     $\tilde{B}_j \leftarrow \tilde{B}_j + \tilde{E}_j, \Delta\tilde{B}_j \leftarrow \Delta\tilde{B}_j + \tilde{E}_j$ 
11:   fin pour
12:    $\Delta\tilde{B}_i \leftarrow 0$ 
13: fin tant que
14: pour tout  $i \in \{1, \dots, n\}$  faire
15:    $\text{MiseÀJourHiérarchie}(\tilde{S}_i)$ 
16:    $\text{Affiche}(S_i)$ 
17: fin pour

```

mailles finales nécessaires. En effet, nous avons pu constater, lors de nos tests, que lorsque les surfaces qui composent la scène deviennent de plus en plus complexes — au sens de leur courbure — les résultats sont d'autant meilleurs que l'ordre des ondelettes est élevé. Or, comme nous l'avons montré dans la section précédente, le surcoût dû à l'utilisation des ondelettes linéaires (\mathcal{M}_2) ou des ondelettes quadratiques (\mathcal{M}_3) par rapport aux ondelettes de HAAR (\mathcal{M}_1) est très faible comparé au gain que ces dernières peuvent apporter.

À ce stade, on peut légitimement se demander si l'utilisation de fonctions de projection à Jacobien constant est réellement indispensable. En effet, pourquoi ne pas utiliser les fonctions de projection naturelles et recalculer le Jacobien chaque fois que cela s'avère nécessaire durant les calculs de propagation énergétique ou pour maintenir la cohérence entre les niveaux de la hiérarchie? Nous montrons dans la section suivante que le Jacobien constant (qui apparaît à la ligne 7 de l'algorithme 8) est primordial en terme d'efficacité et que le fait de contourner cette difficulté fait perdre tout son intérêt au maillage virtuel. Deux autres points de l'algorithme ont été modifiés pour intégrer le maillage virtuel. Nous revenons respectivement sur chacun d'eux dans les sections 3.4.3 (p. 114) pour les surfaces auto-émettrices (ligne 8) et 3.4.4 (p. 114) pour la gestion de la hiérarchie (ligne 9).

3.4.2 Coefficients de *push-pull*

Durant les calculs de radiosité, les informations sont évaluées et transmises entre les différents niveaux de la hiérarchie. Pour maintenir la cohérence entre ces niveaux, nous

avons besoin de pouvoir transférer l'information entre les niveaux en vérifiant deux propriétés principales :

- Il n'y a pas de perte d'information quand on passe d'un niveau de faible précision au niveau de précision supérieure. Cette propriété est indispensable pour pouvoir retrouver l'information sur le niveau initial à une étape ultérieure.
- Il y a conservation de la quantité d'énergie quand on dégrade une représentation pour passer au niveau de précision inférieur. Cette propriété permet de maintenir à chaque niveau la même quantité d'énergie et assure ainsi que le principe de conservation de l'énergie est respecté dans toute la hiérarchie.

Les bases de fonctions ondelettes permettent, justement, de transférer les informations fonctionnelles d'un niveau à l'autre de la hiérarchie en respectant ces propriétés. Elles utilisent pour cela des coefficients de transfert entre les fonctions de base de deux niveaux successifs appelés coefficients de *push-pull*. Ces coefficients sont calculés au moyen d'intégrales sur les domaines des fonctions considérées.

Nous montrons ici en quoi le fait de travailler à Jacobien constant est extrêmement important pour le calcul et l'utilisation des coefficients de *push-pull*.

Pour étudier cette propriété, rappelons que les fonctions ondelettes sont obtenues par homothéties et translations d'une fonction initiale :

$$\phi_i^m(x) = \sqrt{2^m} \phi(2^m x - i). \quad (3.4)$$

Définissons alors $\tilde{B}^m(x)$ comme étant la projection de la fonction de radiosité $B(x)$ dans une base d'ondelettes. On a alors :

$$\tilde{B}^m(x) = \sum \alpha_i^m \phi_i^m(x), \quad (3.5)$$

où m est le niveau de représentation dans la hiérarchie. Les α_i^m sont des poids associés aux fonctions de base ondelettes ϕ_i^m .

En utilisant les équations 3.4 et 3.5, on voit qu'il est possible de calculer des coefficients $\tilde{h}_{i,j}^m$ et $h_{i,j}^m$ qui permettent de transférer de l'information fonctionnelle d'un niveau à l'autre grâce aux relations suivantes :

$$\begin{aligned} \alpha_j^{m+1} &= \sum_i \tilde{h}_{i,j}^m \alpha_i^m, \\ \alpha_i^{m-1} &= \sum_j h_{i,j}^m \alpha_j^m. \end{aligned}$$

Les fonctions ondelettes « classiques » forment des bases orthonormales ; on peut donc exprimer les coefficients de *push-pull* comme suit :

$$\begin{aligned} h_{i,j}^m &= \langle \phi_i^m(x), \phi_j^{m+1}(x) \rangle_{S_i^m}, \\ \tilde{h}_{i,j}^m &= \langle \phi_i^m(x), \phi_j^{m+1}(x) \rangle_{S_j^{m+1}}, \end{aligned} \quad (3.6)$$

où S_i^m représente le support de la fonction ϕ_i^m et \mathcal{A}_j^m l'aire de ce support.

Comme les fonctions ondelettes classiques sont définies sur des domaines normalisés, on peut prouver que les coefficients de *push-pull* $\tilde{h}_{i,j}^m$ et $h_{i,j}^m$ ne dépendent pas du niveau m . Par exemple, pour $h_{i,j}^m$:

$$\begin{aligned} h_{i,j}^m &= \langle \phi_i^m(x), \phi_j^{m+1}(x) \rangle_{S_i^m} \\ &= 2^m \sqrt{2} \langle \phi(2^m x - i), \phi(2^{m+1} x - j) \rangle_{S_i^m} \\ &= 2^m \sqrt{2} \int_{S_i^m} \phi(2^m x - i) \phi(2^{m+1} x - j) dx \\ &= \sqrt{2} \int_{[0,1]} \phi(u) \phi(2u + 2i - j) du \quad (u = 2^m x - i) \\ &= \langle \phi(u), \phi_{2i-j}^1(u) \rangle_{[0,1]} \\ &= h_{2i-j}. \end{aligned}$$

En utilisant la même approche, on montre que $\tilde{h}_{i,j}^m = \tilde{h}_{2i-j}$. Cette propriété est extrêmement importante lors de l'implantation des ondelettes dans un algorithme de radiosité car elle permet de ne pas recalculer les coefficients de *push-pull* à chaque niveau de la hiérarchie. Les coefficients peuvent ainsi être calculés une fois pour toutes au début de l'algorithme et stockés au premier niveau de la hiérarchie. Nous verrons dans la section 3.6 (p. 125) que le fait de ne pas devoir calculer ces coefficients de façon dynamique à chaque fois que l'on veut descendre d'un niveau dans la hiérarchie peut apporter un facteur 3 pour la rapidité de l'algorithme. Faisons aussi remarquer que cette propriété est d'autant plus importante que l'ordre des fonctions est élevé. En effet, plus l'ordre des fonctions est élevé, plus le nombre de fonctions de base est important à chaque niveau hiérarchique. Ainsi, le nombre des coefficients, qui est le carré du nombre de fonctions, peut devenir très grand si l'ordre des fonctions est élevé.

Dans notre cas, les fonctions de base sur les supports originaux ne sont pas les fonctions ondelettes classiques : les supports ne sont pas normalisés et les bases de fonctions n'ont pas de raison d'être orthonormales. L'expression des coefficients de *push-pull* $\tilde{h}_{i,j}^m$ et $h_{i,j}^m$ est donc modifiée de la façon suivante :

$$\begin{bmatrix} h_{0,0}^m & \cdots & h_{0,j}^m & \cdots & h_{0,n_2}^m \\ \vdots & \ddots & \vdots & & \vdots \\ h_{i,0}^m & \cdots & h_{i,j}^m & \cdots & h_{i,n_2}^m \\ \vdots & & \vdots & \ddots & \vdots \\ h_{n_1,0}^m & \cdots & h_{n_1,j}^m & \cdots & h_{n_1,n_2}^m \end{bmatrix} = \left(\mathcal{A}_{k,l}^m \right)^{-1} \left(\mathcal{H}_{i,j}^m \right)$$

et

$$\begin{bmatrix} \tilde{h}_{0,0}^m & \cdots & \tilde{h}_{0,j}^m & \cdots & \tilde{h}_{0,n_2}^m \\ \vdots & \ddots & \vdots & & \vdots \\ \tilde{h}_{i,0}^m & \cdots & \tilde{h}_{i,j}^m & \cdots & \tilde{h}_{i,n_2}^m \\ \vdots & & \vdots & \ddots & \vdots \\ \tilde{h}_{n_1,0}^m & \cdots & \tilde{h}_{n_1,j}^m & \cdots & \tilde{h}_{n_1,n_2}^m \end{bmatrix} = \left(\tilde{\mathcal{A}}_{k,l}^m \right)^{-1} \left(\tilde{\mathcal{H}}_{i,j}^m \right),$$

avec les matrices $\mathcal{A}_{k,l}^m$ est $\tilde{\mathcal{A}}_{k,l}^m$ définies de la façon suivante :

$$\mathcal{A}_{i,j}^m = \begin{bmatrix} \langle \phi_0^m, \phi_0^m \rangle_{S_0^m} & \cdots & \langle \phi_0^m, \phi_j^m \rangle_{S_0^m} & \cdots & \langle \phi_0^m, \phi_{n_2}^m \rangle_{S_0^m} \\ \vdots & \ddots & \vdots & & \vdots \\ \langle \phi_i^m, \phi_0^m \rangle_{S_i^m} & \cdots & \langle \phi_i^m, \phi_j^m \rangle_{S_i^m} & \cdots & \langle \phi_i^m, \phi_{n_2}^m \rangle_{S_i^m} \\ \vdots & & \vdots & \ddots & \vdots \\ \langle \phi_{n_1}^m, \phi_0^m \rangle_{S_{n_1}^m} & \cdots & \langle \phi_{n_1}^m, \phi_j^m \rangle_{S_{n_1}^m} & \cdots & \langle \phi_{n_1}^m, \phi_{n_2}^m \rangle_{S_{n_1}^m} \end{bmatrix}$$

et

$$\tilde{\mathcal{A}}_{i,j}^m = \begin{bmatrix} \langle \phi_0^m, \phi_0^m \rangle_{S_0^m} & \cdots & \langle \phi_0^m, \phi_j^m \rangle_{S_j^m} & \cdots & \langle \phi_0^m, \phi_{n_2}^m \rangle_{S_{n_2}^m} \\ \vdots & \ddots & \vdots & & \vdots \\ \langle \phi_i^m, \phi_0^m \rangle_{S_0^m} & \cdots & \langle \phi_i^m, \phi_j^m \rangle_{S_j^m} & \cdots & \langle \phi_i^m, \phi_{n_2}^m \rangle_{S_{n_2}^m} \\ \vdots & & \vdots & \ddots & \vdots \\ \langle \phi_{n_1}^m, \phi_0^m \rangle_{S_0^m} & \cdots & \langle \phi_{n_1}^m, \phi_j^m \rangle_{S_j^m} & \cdots & \langle \phi_{n_1}^m, \phi_{n_2}^m \rangle_{S_{n_2}^m} \end{bmatrix},$$

et les matrices $\mathcal{H}_{i,j}^m$ et $\tilde{\mathcal{H}}_{i,j}^m$ définies comme suit :

$$\mathcal{H}_{i,j}^m = \begin{bmatrix} \langle \phi_0^m, \phi_0^{m+1} \rangle_{S_0^m} & \cdots & \langle \phi_0^m, \phi_j^{m+1} \rangle_{S_0^m} & \cdots & \langle \phi_0^m, \phi_{n_2}^{m+1} \rangle_{S_0^m} \\ \vdots & \ddots & \vdots & & \vdots \\ \langle \phi_i^m, \phi_0^{m+1} \rangle_{S_i^m} & \cdots & \langle \phi_i^m, \phi_j^{m+1} \rangle_{S_i^m} & \cdots & \langle \phi_i^m, \phi_{n_2}^{m+1} \rangle_{S_i^m} \\ \vdots & & \vdots & \ddots & \vdots \\ \langle \phi_{n_1}^m, \phi_0^{m+1} \rangle_{S_{n_1}^m} & \cdots & \langle \phi_{n_1}^m, \phi_j^{m+1} \rangle_{S_{n_1}^m} & \cdots & \langle \phi_{n_1}^m, \phi_{n_2}^{m+1} \rangle_{S_{n_1}^m} \end{bmatrix}$$

et

$$\tilde{\mathcal{H}}_{i,j}^m = \begin{bmatrix} \langle \phi_0^m, \phi_0^{m+1} \rangle_{S_0^{m+1}} & \cdots & \langle \phi_0^m, \phi_j^{m+1} \rangle_{S_j^{m+1}} & \cdots & \langle \phi_0^m, \phi_{n_2}^{m+1} \rangle_{S_{n_2}^{m+1}} \\ \vdots & \ddots & \vdots & & \vdots \\ \langle \phi_i^m, \phi_0^{m+1} \rangle_{S_0^{m+1}} & \cdots & \langle \phi_i^m, \phi_j^{m+1} \rangle_{S_j^{m+1}} & \cdots & \langle \phi_i^m, \phi_{n_2}^{m+1} \rangle_{S_{n_2}^{m+1}} \\ \vdots & & \vdots & \ddots & \vdots \\ \langle \phi_{n_1}^m, \phi_0^{m+1} \rangle_{S_0^{m+1}} & \cdots & \langle \phi_{n_1}^m, \phi_j^{m+1} \rangle_{S_j^{m+1}} & \cdots & \langle \phi_{n_1}^m, \phi_{n_2}^{m+1} \rangle_{S_{n_2}^{m+1}} \end{bmatrix}.$$

Le point très important de ces calculs est que l'évaluation des produits scalaires n'est pas évidente dans le cas où les primitives géométriques ne sont pas des carrés normalisés. En effet, le produit scalaire $\langle f, g \rangle$ est égal à :

$$\langle f, g \rangle_S = \int_S f(x)g(x) dx.$$

Or, dans notre cas, les supports S_i^m ne sont pas les supports sur lesquels sont définies les ondelettes classiques. On est donc obligé de projeter toutes les surfaces sur des supports normalisés pour pouvoir effectuer les calculs. Après projection, on obtient :

$$\begin{aligned} \langle \phi_i^{m_1}, \phi_j^{m_2} \rangle_{S_i^{m_1}} &= \int_{S_i^{m_1}} \phi_i^{m_1}(\theta, \phi) \phi_j^{m_2}(\theta, \phi) d\theta d\phi, \\ &= \int_{\tilde{S}_i} \tilde{\phi}_i(u, v) \tilde{\phi}_j^{m_2 - m_1}(u, v) \mathcal{J}_{\theta, \phi}^{m_1}(u, v) dudv, \end{aligned} \quad (3.7)$$

où $\tilde{S}_i = [0, 1]^2$ est le support normalisé de la fonction ondelette « classique » $\tilde{\phi}_i$ et $\mathcal{J}_{\theta, \phi}^m(u, v)$ est le Jacobien de la fonction de projection inverse $\Psi^{-1} : (\theta, \phi) \mapsto (u, v)$ qui est égal à :

$$\mathcal{J}_{\theta, \phi}^m(u, v) = \begin{vmatrix} \frac{\partial \theta}{\partial u} & \frac{\partial \phi}{\partial u} \\ \frac{\partial \theta}{\partial v} & \frac{\partial \phi}{\partial v} \end{vmatrix}.$$

On constate ici que l'efficacité de l'algorithme de radiosité dépend fortement du calcul des coefficients de *push-pull*. En effet, ce calcul, qui demande pour chaque coefficient l'évaluation de l'équation (3.7), peut être extrêmement coûteux et pénaliser fortement l'algorithme s'il n'est pas optimisé correctement.

Quand la fonction Ψ est à Jacobien constant, le Jacobien peut être sorti de l'intégrale dans l'équation (3.7) :

$$\langle \phi_i^{m_1}, \phi_j^{m_2} \rangle_{S_i^{m_1}} = \mathcal{J}_{\theta, \phi}^m \int_{[0, 1]^2} \tilde{\phi}_i(u, v) \tilde{\phi}_j^{m_2 - m_1}(u, v) dudv.$$

On est alors ramené au cas usuel où les coefficients ne dépendent pas du niveau auquel les transferts doivent être effectués. Si on considère les fonctions ondelettes classiques (ondelettes multiples \mathcal{M}_n), on travaille alors directement sur des bases orthonormales et l'équation des coefficients de *push-pull* est alors la même que l'équation (3.6), à une constante près.

Le maillage virtuel, tel que nous l'avons présenté dans la section précédente, est donc suffisamment générique pour que ses avantages agissent à tous les niveaux de l'algorithme de radiosité, et ce quel que soit l'algorithme choisi. Il permet, en effet, de toujours se ramener au cas le plus simple des primitives géométriques carrées.

En conclusion :

Les fonctions de projection à Jacobien constant représentent un point essentiel pour l'efficacité du maillage virtuel dans l'algorithme de radiosité à base d'ondelettes.

3.4.3 Primitives auto-émettrices

Le maillage virtuel permet de garder la géométrie initiale « intacte ». Il est alors possible qu'une surface courbe ait besoin de *shooter* de l'énergie sur elle-même. Plus précisément, toute surface courbe qui présente au moins un « îlot » concave sera auto-émettrice. Pour gérer une telle surface lorsqu'elle émet sur elle-même, on crée temporairement un *leurre* (voir la figure 3.11). Le *leurre* est une copie simplifiée de la surface. En fait, on ne copie que le premier niveau hiérarchique et on fait comme si la surface ainsi créée était une nouvelle surface n'ayant jamais été subdivisée et ne renfermant aucune énergie.

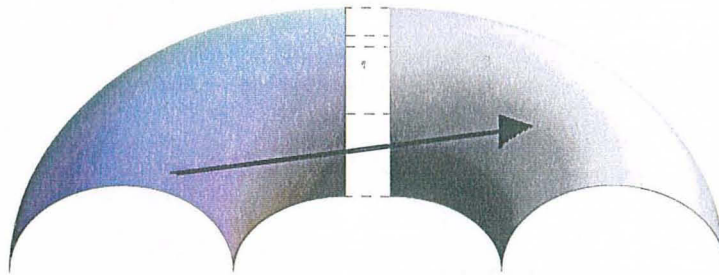


FIG. 3.11 – Surface auto-émettrice.

On procède ensuite en trois étapes. Tout d'abord, la primitive courbe *shoote* son énergie sur le *leurre* qu'on vient de créer exactement comme si ce dernier était une surface de la scène. Puis, l'énergie récupérée sur le *leurre* est ajoutée à celle de la surface originale, en rajoutant, dans la hiérarchie de cette dernière, de nouvelles subdivisions si cela est nécessaire. Finalement, le *leurre* est détruit et l'algorithme continue normalement.

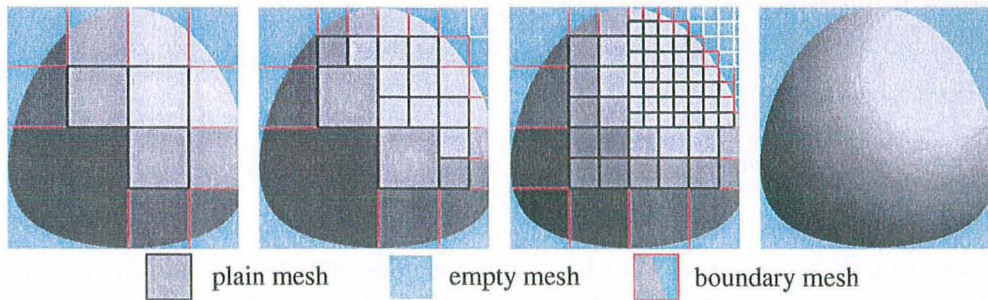


FIG. 3.12 – Subdivisions hiérarchiques de la maille virtuelle associée à la surface délimitée par la courbe de la figure 3.5(a).

3.4.4 Gérer la hiérarchie

Considérons maintenant que l'image d'une primitive courbe par une fonction de projection à Jacobien constant soit le support de l'approximation à base d'ondelettes d'une fonction de radiosit . Afin de b n ficier de la simplicit  de l'expression des coefficients de *push-pull* sur cette maille virtuelle, nous l'utilisons dans notre algorithme comme si elle  tait le support r el de la fonction de radiosit . De cette fa on, tous les calculs fonctionnels

sont effectués sur un carré normalisé et les subdivisions hiérarchiques se font très naturellement (voir la figure 3.12). En effet, il suffit d'utiliser un simple algorithme de *quadtree* pour générer quatre nouvelles mailles à chaque niveau ; ces mailles serviront directement de mailles virtuelles pour les parties de la primitive originale qui se retrouveront projetées à l'intérieur.

La représentation hiérarchique de la fonction de radiosité suppose que l'on ait la même précision sur la totalité du support. Il ne faut surtout pas diminuer la précision aux endroits où la maille virtuelle ne correspond pas à des points de la primitive originale. En effet, l'étape de *push-pull* fait que des valeurs de radiosité sur des points en dehors de la primitive originale peuvent avoir une influence sur les valeurs des points à l'intérieur de cette primitive. Cette remarque nous conforte dans le fait qu'il ne faut rien changer à l'algorithme quand on utilise le maillage virtuel et confirme donc la simplicité de son utilisation.

Deux cas particuliers peuvent se présenter lors des subdivisions successives de la maille virtuelle :

- Après subdivision une sous-maille peut ne pas contenir de zone correspondant à la primitive originale. On mémorise alors le fait que cette maille n'interviendra pas dans la phase d'émission. On continue cependant à calculer la fonction de radiosité sur cette sous-maille comme pour les autres mailles.
- Une maille issue d'une subdivision peut être entièrement remplie par la primitive originale. Dans ce cas, on se retrouve dans la situation standard d'un parallélogramme et on conserve cette information pour simplifier les calculs d'émission dans la suite de l'algorithme.

L'utilisation des mailles virtuelles dans le processus de subdivision hiérarchique permet de ne pas avoir recours à une stratégie de maillage compliquée qui générerait de nouvelles surfaces complexes à chaque subdivision. Ces nouvelles surfaces seraient elles-mêmes difficiles à gérer. De plus, grâce à sa stratégie élémentaire de subdivision, le maillage virtuel permet de réduire à la fois le nombre de primitives initiales et le nombre de mailles finales comme nous le montrons dans la section 3.6 (p. 125).

3.5 Applications

En utilisant les idées présentées dans la section 3.3 (p. 95), on peut se servir du maillage virtuel pour gérer une grande variété de primitives complexes. Nous illustrons ici cette utilisation sur deux types de surfaces très importantes :

- Les surfaces planes, avec ou sans trous et pouvant avoir plusieurs composantes connexes. Nous imposons aux contours de ces surfaces de n'être composés que de segments de droites et de morceaux de coniques.
- Les morceaux de surfaces algébriques de degré 2, c'est-à-dire les surfaces quadriques. Nous imposons d'utiliser les mêmes restrictions pour les contours des projections sur les mailles virtuelles de ces surfaces que pour les contours des surfaces planes.

3.5.1 Primitives planes quelconques

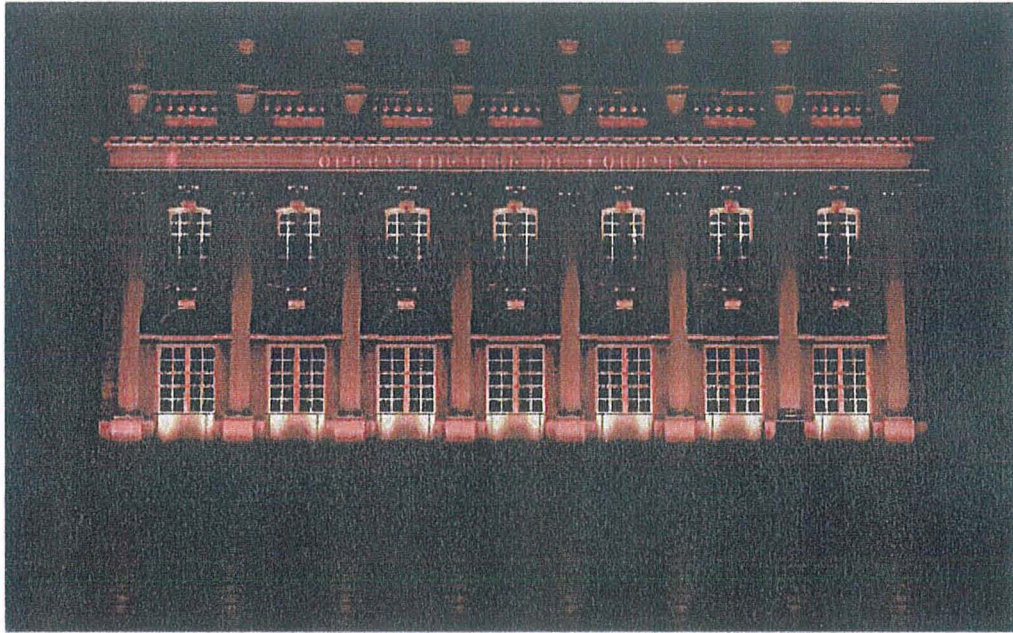
La gestion des surfaces planes ne requiert rien de plus que l'application des principes exposés dans la section 3.3.1.2 (p. 99) : chaque primitive plane est englobée par un parallélogramme d'aire minimale et chaque parallélogramme est projeté sur un carré normalisé pour générer les mailles virtuelles. Toutes les opérations de l'algorithme de radiosité, à base d'ondelettes sont, ensuite, effectuées sur ces nouveaux domaines normalisés.

Nous avons implanté cette application fondamentale du maillage virtuel dans notre algorithme de radiosité à base d'ondelettes et nous avons obtenu d'excellents résultats. En effet, dans le domaine de la modélisation architecturale, les surfaces planes à bords complexes sont très souvent utilisées. Jusqu'à présent, pour pouvoir les intégrer dans des algorithmes de radiosité, on les triangulait. Avec le maillage virtuel, on peut non seulement éviter cette étape de triangulation mais on peut surtout utiliser directement ces surfaces dans l'algorithme de radiosité. La figure 3.13 montre une image de l'Opéra de la place Stanislas à Nancy. Le modèle original de cette scène comprend 32 429 triangles et parallélogrammes. Si on utilise le maillage virtuel, ce nombre tombe à 17 272. Nous montrons dans la section 3.6 (p. 125) que ce gain permet de diminuer fortement le nombre de mailles finales et surtout le temps nécessaire à la simulation de l'éclairage.

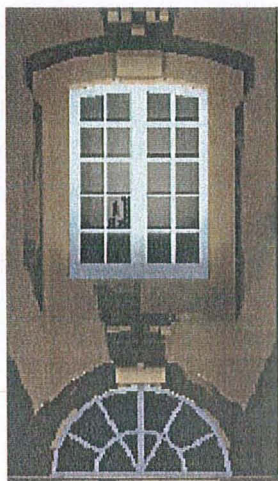
3.5.2 Quadriques

La relative simplicité mathématique des quadriques et leur grand potentiel en terme de modélisation géométrique font que ces surfaces commencent à trouver de nouvelles applications dans les domaines de la modélisation et de la visualisation. Récemment, de nouveaux moteurs de modélisation à base de surfaces algébriques de degré peu élevé, et notamment de quadriques, ont vu le jour. De plus, un certain nombre de recherches s'orientent actuellement autour de l'approximation de surfaces paramétriques de degré élevé, incluant les B-Splines et les NURBS, en surfaces quadriques (voir les travaux de B. LACOLLE et N. SZAFRAN dans [Lac 99]). Ces recherches vont dans le sens d'une généralisation des techniques de triangulation en utilisant des surfaces plus complexes pour permettre à ces approximations d'intégrer des notions de continuité aussi bien pour la surface que pour ses dérivées.

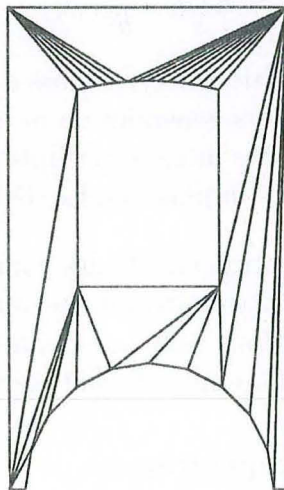
Il apparaît donc très intéressant de pouvoir illuminer directement les surfaces quadriques dans un algorithme de radiosité. De plus, comme nous le montrons dans ce qui suit, le maillage virtuel rend cette application relativement facile à implanter et surtout très efficace dans la pratique. Dans cette section, nous montrons comment trouver des fonctions de projection à Jacobien constant pour les morceaux de surfaces quadriques. Étant donné que les projections pour la plupart des quadriques utilisent des fonctions spéciales issues de l'étude des coniques, nous commençons par faire une digression pour regarder le problème en 2D. Nous donnons ensuite une taxinomie des différentes quadriques avant de détailler la construction des fonctions de projection pour tous les types de quadriques. Pour plus de précision, nous donnons en annexe C (p. 185) une description complète de la construction et de l'utilisation des quadriques dans notre plate-forme logicielle Candela. Les annexes A (p. 165) et B (p. 179) donnent les prérequis nécessaires pour la compréhension et l'utilisation des fonctions de projection.



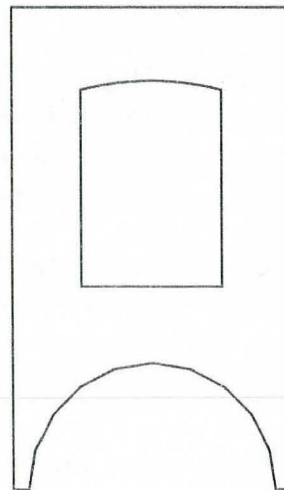
(a) La scène de l'Opéra de la place Stanislas à Nancy



(b) Détail de la figure (a)



(c) Triangulée : 32 triangles



(d) Maille virtuelle : 1 maille

FIG. 3.13 – Utilisation des surfaces planes complexes en architecture.

3.5.2.1 Préliminaires en 2D

Regardons de plus près le cas d'une ellipse dont les axes sont orthogonaux. L'équation générale d'une telle ellipse :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \quad \begin{cases} x = a \cos \theta, \\ y = b \sin \theta, \end{cases} \quad b \geq a > 0, \quad \theta \in [0, 2\pi).$$

Si on pose $\theta = f(u)$. On peut alors, exactement comme cela est expliqué dans la section 3.3.1.1 (p. 97), trouver une fonction de projection à Jacobien constant $\psi : u \mapsto (x, y)$ (avec $\mathcal{J}(\psi) = b$) en posant :

$$u = \frac{1}{b} \int_0^T \sqrt{a^2 \sin^2 t + b^2 \cos^2 t} dt. \quad (3.8)$$

Cette intégrale représente, en fait, la longueur de l'arc d'ellipse pour θ allant de 0 à T . C'est une *intégrale elliptique*¹⁵ qui a été largement étudiée au *XIX^{ième}* siècle pour résoudre des problèmes en astronomie (son nom vient justement du fait qu'elle exprime la longueur de l'arc d'une ellipse). On peut utiliser sa forme canonique

$$E(\alpha, m) = \int_0^\alpha \sqrt{1 - m \sin^2 t} dt$$

pour la réécrire de la façon suivante :

$$u = E\left(\theta, \frac{b^2 - a^2}{b^2}\right). \quad (3.9)$$

Constatons que si l'ellipse est en fait un cercle alors $a = b$ et comme $E(\alpha, 0) = \alpha$, on a immédiatement $u = \theta$. Pour des valeurs générales de m et de α , il existe des algorithmes très efficaces pour calculer les valeurs des intégrales elliptiques et de leurs fonctions inverses (se référer à l'annexe A (p. 165) pour de plus amples détails).

Les quadriques étant la généralisation en 3D des coniques, il est logique que les intégrales elliptiques que nous venons d'introduire nous servent pour construire leur fonctions de projection à Jacobien constant. Nous pouvons maintenant nous intéresser au cas de ces surfaces en commençant par en faire une classification.

3.5.2.2 Taxinomie des surfaces quadriques

Les surfaces quadriques sont les surfaces algébriques de degré 2. L'équation générale d'une quadrique est donnée par $f(\vec{x}) = 0$ où $\vec{x} = (X, Y, Z)$ et f est un polynôme du second degré :

$$f(\vec{x}) = aX^2 + a'Y^2 + a''Z^2 + 2b'XY + 2b'XZ + 2bYZ + 2cX + 2c'Y + 2c''Z + d$$

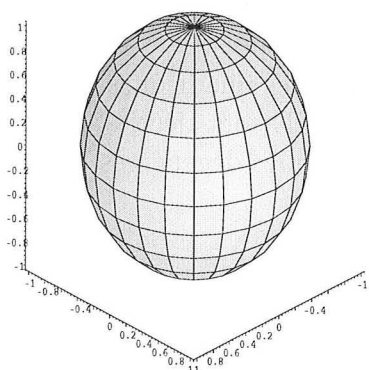
15. Une description relativement complète de l'étude des intégrales elliptiques est donnée en annexe A (p. 165).

Le polynôme f définit une forme quadratique sur \mathbb{R}^3 . On peut donc déduire trois valeurs propres et trois vecteurs propres de la définition de f . Toutes les quadriques s'écrivent, sous formes réduites, dans le repère formé par leurs vecteurs propres. Ces formes réduites donnent une classification en fonction de la nullité des valeurs propres.

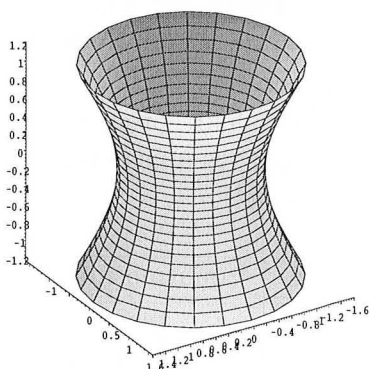
Le tableau 3.1 et la figure 3.14 montrent cette classification des quadriques en 9 types tels qu'ils ressortent de l'analyse des valeurs propres et tels qu'ils sont décrits dans l'annexe C (p. 185) (nous n'avons pas classé les quatre types quadriques dégénérées, les plans, les points, l'ensemble vide et l'espace \mathbb{R}^3).

Type	Équation implicite	Paramétrage
<i>Ellipsoïde</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1$ $a \geq b \geq c \geq 0$	$x = a \cos \theta \cos \varphi$ $y = b \sin \theta \cos \varphi$ $z = c \sin \varphi$
<i>Hyperboloïde à une nappe</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1$ $a \geq b \geq c \geq 0$	$x = a \cos \theta \cosh \varphi$ $y = b \sin \theta \cosh \varphi$ $z = c \sinh \varphi$
<i>Hyperboloïde à deux nappes</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = -1$ $a \geq b \geq 0$	$x = a \cos \theta \sinh \varphi$ $y = b \sin \theta \sinh \varphi$ $z = c \cosh \varphi$
<i>Cône elliptique</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 0$ $b \geq a \geq 0$	$x = a \cos \theta \varphi$ $y = b \sin \theta \varphi$ $z = c \varphi$
<i>Paraboloïde elliptique</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} - z = 0$ $b \geq a \geq 0$	$x = a \cos \theta \varphi$ $y = b \sin \theta \varphi$ $z = \varphi^2$
<i>Paraboloïde hyperbolique</i>	$\frac{x^2}{a^2} - \frac{y^2}{b^2} - z = 0$	$x = a \theta$ $y = b \varphi$ $z = \theta^2 - \varphi^2$
<i>Cylindre elliptique</i>	$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ $a \geq b \geq 0$	$x = a \cos \theta$ $y = b \sin \theta$ $z = \varphi$
<i>Cylindre parabolique</i>	$x^2 = 2pz$	$x = \theta$ $y = \varphi$ $z = \frac{\theta^2}{2p}$
<i>Cylindre hyperbolique</i>	$x^2 = 2p$	$x = 2p$ $y = \theta$ $z = \varphi$

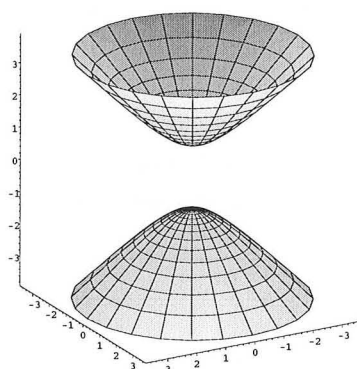
TAB. 3.1 – Classification des surfaces quadriques.



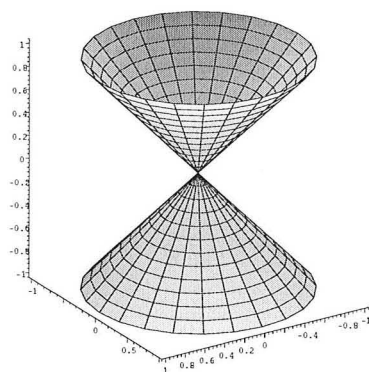
(a) ellipsoïde



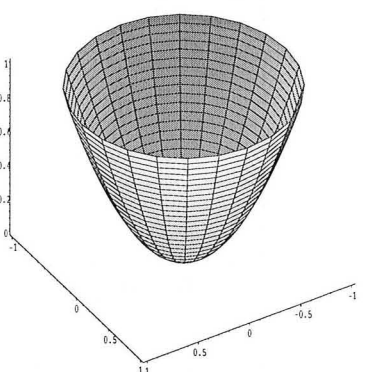
(b) Hyperboloïde à une nappe



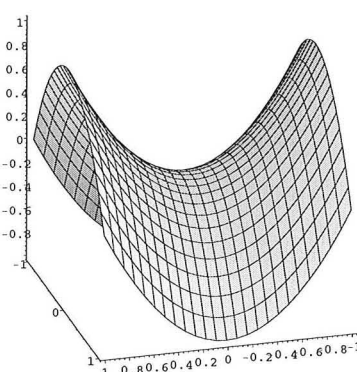
(c) Hyperboloïde à deux nappes



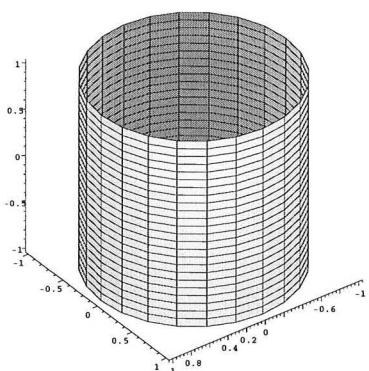
(d) Cône elliptique



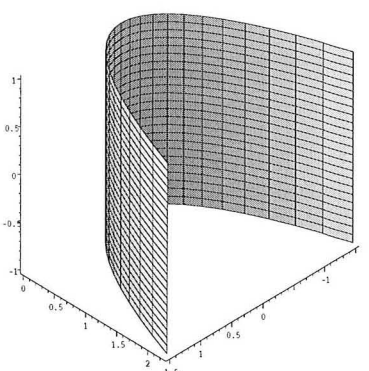
(e) Parabolôïde elliptique



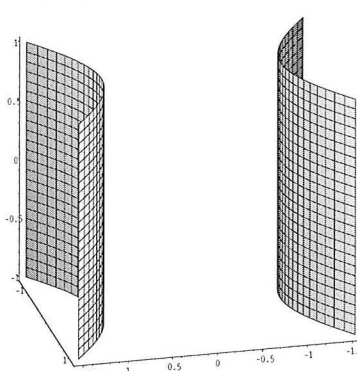
(f) Parabolôïde hyperbolique



(g) Cylindre elliptique



(h) Cylindre parabolique



(i) Cylindre hyperbolique

FIG. 3.14 – 9 types de quadriques dans leur forme normalisée

3.5.2.3 Fonctions de mapping

Pour chaque type de quadrique, nous avons cherché une fonction de projection à Jacobien constant et nous les détaillons maintenant.

Une description précise de la construction des fonctions de projection est donnée en annexe C (p. 185). Les notations qui suivent utilisent parfois les notions introduites dans les annexes A (p. 165) et B (p. 179).

1) Ellipsoïde :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1, \quad \begin{cases} x = a \sin \theta \sin \varphi, \\ y = b \cos \theta \sin \varphi, \\ z = c \cos \varphi. \end{cases}$$

Soit $\theta = f(u)$ et $\varphi = g(u, v)$. Alors, si $a \geq b \geq c$, on prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(E \left(\theta, \frac{b^2 - a^2}{b^2} \right), \sin \varphi \sqrt{1 - o^2 \sin^2 \varphi} + \frac{1}{o} \arcsin(o \sin \varphi) \right) \end{aligned} \quad \mathcal{J} = \frac{bc}{2}$$

avec

$$o^2 = 1 - \frac{a^2 b^2}{c^2 (a^2 \sin^2 \theta + b^2 \cos^2 \theta)} \geq 0.$$

Si, dans l'équation de l'ellipsoïde, $a = b$ (cette surface s'appelle *prolate spheroid* en anglais), l'équation de projection se simplifie considérablement car dans ce cas o est une constante. Ainsi, u et v ne dépendent chacun que d'un des deux paramètres de la surface. Quand $b = c$ (cette surface s'appelle *oblate spheroid* en anglais), un changement de paramétrage conduit à une simplification similaire. Enfin, si $a = b = c$ (on est dans le cas d'une sphère), on peut prendre comme fonction de projection : $\Psi : (\theta, \varphi) \longmapsto (\theta, \arcsin \varphi)$. Ces trois exemples de simplification montrent qu'il peut être important de détecter et de traiter à part certains cas particulier pour augmenter l'efficacité de l'algorithme. Ce phénomène est d'autant plus important que des simplifications sont généralement possibles pour les surfaces les plus couramment utilisées en modélisation.

La figure 3.15 montre les effets d'un paramétrage à Jacobien constant sur une sphère et un ellipsoïde. On notera que la projection du paramétrage de l'ellipsoïde sur la maille virtuelle donne une surface plane dont les contours sont relativement complexes. Il faut faire attention dans ce genre de cas quand on étend la fonction de radiosité à toute la maille virtuelle (c'est-à-dire sur le parallélogramme englobant la projection) car les points en dehors de la projection n'ont pas toujours une correspondance physique avec un point de l'environnement (ils correspondent parfois à des coordonnées à valeurs complexes — au sens mathématique du terme). La solution est alors de redéfinir une correspondance physique pour ces points imaginaires ; toute correspondance qui maintient le Jacobien constant convient.

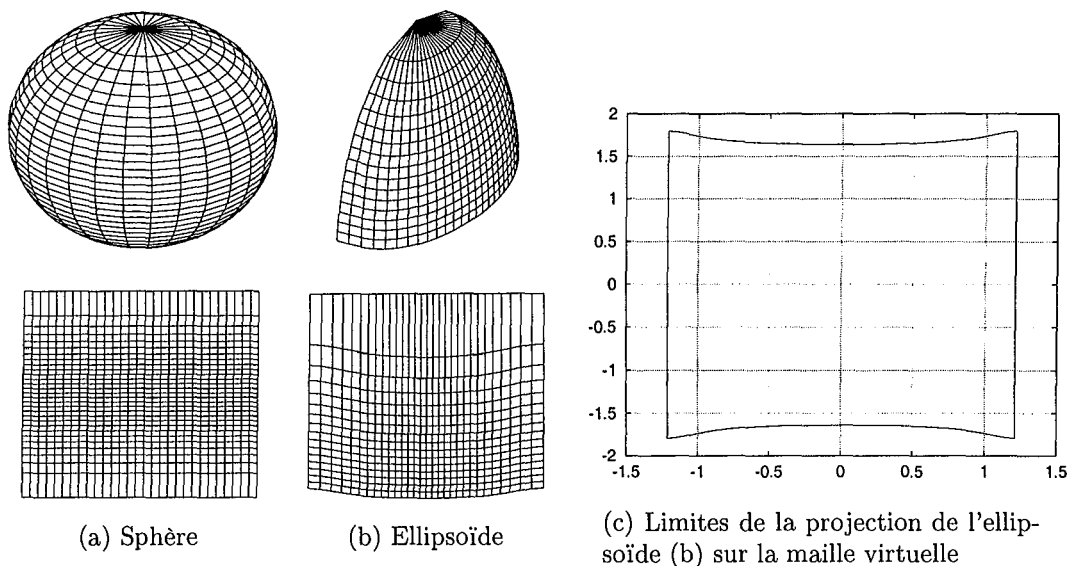


FIG. 3.15 – Exemple de projection à Jacobien constant sur la maille virtuelle pour une sphère et pour un ellipsoïde. Pour les figures (a) et (b), la rangée du haut correspond à la subdivision à Jacobien constant et celle du bas à la projection de cette subdivision dans le repère du paramétrage naturel en (θ, φ) .

2) Hyperboloïde à une nappe :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1, \quad \begin{cases} x = a \sin \theta \cosh \varphi, \\ y = b \cos \theta \cosh \varphi, \\ z = c \sinh \varphi. \end{cases}$$

Si $a \geq b \geq c$, on prend comme fonction de projection :

$$\Psi : (\theta, \varphi) \longrightarrow (u, v)$$

$$(\theta, \varphi) \longmapsto \left(E\left(\theta, \frac{b^2 - a^2}{b^2}\right), \sinh \varphi \sqrt{1 + p^2 \sinh^2 \varphi} + \frac{1}{p} \operatorname{arcsinh}(p \sinh \varphi) \right) \quad \mathcal{J} = \frac{bc}{2}$$

avec

$$p^2 = 1 + \frac{a^2 b^2}{c^2 (a^2 \sin^2 \theta + b^2 \cos^2 \theta)}$$

Notons que si $a = b$, alors p est une constante. Ainsi, comme pour l'ellipsoïde, u et v ne dépendent chacun que d'un des deux paramètres de la surface. On peut donc simplifier les équations en fonction de ce résultat. Si $b = c$, un simple changement de variable nous conduit aux mêmes conclusions et par conséquent aux mêmes simplifications.

3) Hyperboloïde à deux nappes :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = -1, \quad \begin{cases} x = a \sin \theta \sinh \varphi, \\ y = b \cos \theta \sinh \varphi, \\ z = c \cosh \varphi. \end{cases}$$

On prend comme fonction de projection (en reprenant les notations de l'hyperboloïde à une nappe où $p^2 = 1 + \frac{a^2 b^2}{c^2(a^2 \sin^2 \theta + b^2 \cos^2 \theta)}$):

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(E\left(\theta, \frac{b^2 - a^2}{b^2}\right), \cosh \varphi \sqrt{1 + p^2 \sinh^2 \varphi} + \frac{1}{p} \operatorname{arcsinh}(p \cosh \varphi) \right) \quad \mathcal{J} = \frac{bc}{2} \end{aligned}$$

Les mêmes remarques que pour l'hyperboloïde à une nappe peuvent être faites à propos des simplifications envisageables si $a = b$ ou $b = c$.

4) Cône :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 0, \quad \begin{cases} x = a \cos \theta \varphi, \\ y = b \sin \theta \varphi, \\ z = c \varphi. \end{cases}$$

On prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(E\left(\theta, \frac{c^2(b^2 - a^2)}{b^2(a^2 + c^2)}\right), \varphi^2 \right) \quad \mathcal{J} = \frac{b}{2} \sqrt{a^2 + c^2} \end{aligned}$$

Si $a = b$, on est dans le cas d'un cône à base circulaire. L'équation de la fonction de projection se simplifie, alors, de la façon suivante: $\Psi : (\theta, \varphi) \longmapsto (\theta, \varphi^2)$.

5) Paraboloïde elliptique :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - z = 0, \quad \begin{cases} x = a \cos \theta \sqrt{\varphi}, \\ y = b \sin \theta \sqrt{\varphi}, \\ z = \varphi. \end{cases}$$

Si $b \geq a$, on prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(E\left(\theta, \frac{b^2 - a^2}{b^2}\right), \left(\varphi + \frac{a^2 b^2}{4(b^2 \cos^2 \theta + a^2 \sin^2 \theta)}\right)^{3/2} \right) \quad \mathcal{J} = \frac{b}{3} \end{aligned}$$

6) Paraboloïde hyperbolique :

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} - z = 0, \quad \begin{cases} x = a \cosh \theta \sqrt{\varphi}, \\ y = b \sinh \theta \sqrt{\varphi}, \\ z = \varphi. \end{cases}$$

On prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(u, \left(\varphi + \frac{a^2 b^2}{4\sqrt{b^2 \cosh^2 \theta + a^2 \sinh^2 \theta}}\right)^{3/2} \right) \quad \mathcal{J} = 1 \end{aligned}$$

7) **Cylindre elliptique :**

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \quad \begin{cases} x = a \cos \theta, \\ y = b \sin \theta, \\ z = \varphi. \end{cases}$$

On prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(E\left(\theta, \frac{b^2 - a^2}{b^2}\right), \varphi \right) \quad \mathcal{J} = b \end{aligned}$$

Tout comme pour l'ellipsoïde, l'hyperboloïde et le cône, ces équations se simplifient énormément quand $a = b$ car, alors, on peut prendre $\Psi : (\theta, \varphi) \longmapsto (\theta, \varphi)$. Ces deux exemples ne sont pas les seuls où l'on peut obtenir des simplifications mais ce sont les plus « spectaculaires ».

Un bon exemple de l'utilisation des cylindres est le cas de scènes modélisées en 2,5D. En effet, les scènes modélisées en 2D et auxquelles on a donné du volume en les élevant suivant l'axe z utilisent toutes des cylindres pour représenter la partie d'élévation. Un exemple d'une telle scène est présenté sur la figure 3.16 où les chiffres du deuxième millénaire sont modélisés grâce à des surfaces planes à bords complexes (segments et morceaux de coniques) et l'élévation est obtenue grâce à des cylindres. Cette scène ne contient que 22 surfaces alors qu'il en faudrait plusieurs milliers pour la représenter correctement avec des triangles.

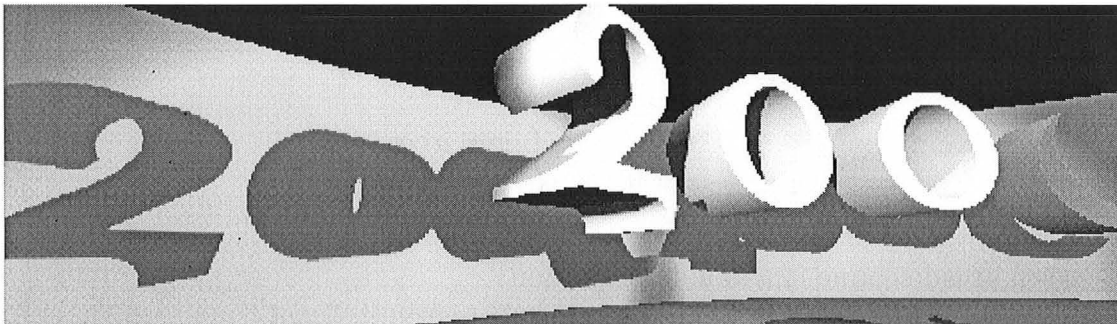


FIG. 3.16 – *Illumination d'une scène composée de 22 surfaces dont 9 surfaces planes à bords complexes et 13 morceaux de cylindres.*

8) **Cylindre parabolique :**

$$x^2 = 2pz, \quad \begin{cases} x = a\theta, \\ y = \frac{a^2}{2p}\theta^2, \\ z = \varphi. \end{cases}$$

On prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(\frac{a}{2}\theta \sqrt{1 + \left(\frac{a}{p}\theta\right)^2} + \frac{p}{2} \sinh^{-1} \frac{a}{p}\theta, \varphi \right) \quad \mathcal{J} = 1 \end{aligned}$$

9) Cylindre hyperbolique :

$$x^2 = 2p, \quad \begin{cases} x = a \cosh \theta, \\ y = b \sinh \theta, \\ z = \varphi. \end{cases}$$

On prend comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto \left(-ibE\left(i\theta, \frac{a^2 + b^2}{b^2}\right), \varphi\right) \quad \mathcal{J} = 1 \end{aligned}$$

10) Tore circulaire : Le tore circulaire n'est pas une quadrique, mais vu la fréquence de son utilisation en modélisation et vu la simplicité de sa représentation paramétrique nous avons cherché, pour lui aussi, une fonction de projection à Jacobien constant. La représentation paramétrique du tore circulaire est la suivante :

$$\begin{cases} x = a \cos \theta (\cos \varphi + b) \\ y = a \sin \theta (\cos \varphi + b) \\ z = a \sin \varphi \end{cases}$$

On peut donc prendre comme fonction de projection :

$$\begin{aligned} \Psi : (\theta, \varphi) &\longrightarrow (u, v) \\ (\theta, \varphi) &\longmapsto (\theta, \sin \varphi + b\varphi) \quad \mathcal{J} = 1 \end{aligned}$$

Malgré la très grande simplicité du tore circulaire, la gestion d'un tore quelconque est beaucoup plus compliquée. Nous nous sommes, pour l'instant, limités dans notre implantation au cas du tore circulaire.

3.6 Résultats expérimentaux

Dans cette section, nous présentons les résultats que nous avons obtenus en utilisant le maillage virtuel dans des scènes contenant des surfaces planes à bords complexes et des surfaces quadriques. Tous les tests ont été effectués avec la plate-forme logicielle Candela. On évite ainsi d'avoir des résultats biaisés par les différences d'implantation. D'autre part, nous avons toujours utilisé la même machine, une Origin2000 munie de 64 processeurs MIPS R10000 cadencés à 195 MHz et disposant d'une mémoire partagée de 24 Go. Nous avons profité de l'implantation parallèle de notre algorithme (implantation faite par Xavier CAVIN *et al.* et exposée dans [Cav 98]) pour pouvoir faire nos calculs plus rapidement. Le nombre de processeurs que nous avons utilisés varie suivant les tests et nous donnons pour chacun d'eux une description précise de la configuration matérielle qui a été retenue. De même que pour les processeurs, nous n'avons jamais utilisé la totalité de la mémoire disponible pour effectuer nos tests.

Comme nous l'avons déjà fait lors de la présentation de nos résultats sur l'utilisation des bases d'ondelettes, nous ne donnons pas, dans cette section, la totalité des résultats que

nous avons obtenus sur des scènes issues d'applications du monde réel. Nous souhaitons, en effet, nous concentrer sur l'impact de l'utilisation du maillage virtuel dans un algorithme de radiosit . Nous voulons notamment prouver l'efficacit  de l'utilisation respective des surfaces planes   bords complexes et des surfaces quadriques directement dans les calculs de radiosit . Dans les deux sections qui suivent nous pr sentons respectivement les r sultats obtenus avec les surfaces planes complexes (section 3.6.1 (p. 126)) et ceux obtenus avec les quadriques (section 3.6.2 (p. 129)). Nous r servons le chapitre 4 (p. 137) pour pr senter l'ensemble de nos r sultats sur des sc nes issues d'applications du monde r el.

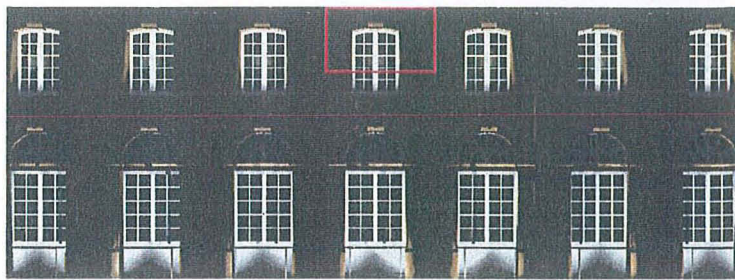
3.6.1 R sultats avec des surfaces planes   bords complexes

3.6.1.1 Consid rations g n rales

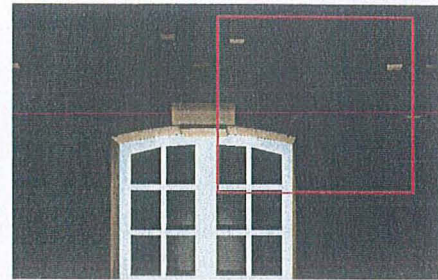
La sc ne que nous avons utilis e pour  tudier et pr senter l'impact de l'utilisation de surfaces planes   bords complexes dans un algorithme de radiosit  est la sc ne de la fa ade de l'Op ra sur la place Stanislas de Nancy d j   voqu e dans la section 3.5.1 (p. 116). Les images sur lesquelles nous nous appuyons pour pr senter nos r sultats proviennent de la zone de maillage situ e pr s d'une fen tre du premier  tage (figures 3.17(a) et 3.17(b)). Nous nous limitons   la pr sentation des r sultats qualitatifs et quantitatifs pour la sc ne de l'Op ra dans un souci de clart  et parce que les tests que nous avons effectu s sur les autres sc nes (d crites dans le chapitre 4 (p. 137)) nous ont conduits, proportionnellement, aux m mes conclusions.

Afin de faire une comparaison objective de notre algorithme du maillage virtuel avec les pr c dentes solutions propos es pour g rer les surfaces planes, nous avons d cid  d'utiliser une version polygonale de la sc ne de l'Op ra. L'algorithme de triangulation que nous avons utilis  est bas  sur le *tesselator* de la librairie graphique OPEN-GL. Cet algorithme g n re des triangles de forme tr s allong e relativement mal adapt s au calcul de radiosit  — voir figure 3.17(e) —. Il est possible d'utiliser des algorithmes de triangulation qui g n rent des triangles de forme mieux adapt e aux simulations (tels que l'algorithme de triangulation de DELAUNAY contrainte). Cependant, ces algorithmes g n rent g n ralement beaucoup plus de triangles que l'algorithme bas  sur OPEN-GL (qui en g n re un nombre minimum). Or, on a d j  fait remarquer que l'algorithme de radiosit   tait tr s d pendant du nombre de surfaces initiales, notamment pour les calculs de visibilit . Nous avons donc retenu l'algorithme bas  sur OPEN-GL pour ne pas biaiser nos comparaisons.

Pour que nos r sultats soient coh rents entre eux, nous avons toujours utilis  notre algorithme de radiosit  avec les m mes param tres de simulation : m me taille minimum de d coupage des surfaces et m me seuil pour l'oracle de subdivision. Nous avons, d'autre part, toujours pris le m me crit re d'arr t : un taux de convergence. Et pour que ce taux de convergence ait la m me signification dans toutes les simulations, nous avons d fini la convergence comme le rapport entre l' nergie initiale   propager dans la sc ne et l' nergie restant   propager   un instant donn . Ce taux repr sente donc un param tre global ind pendant de la g om trie.



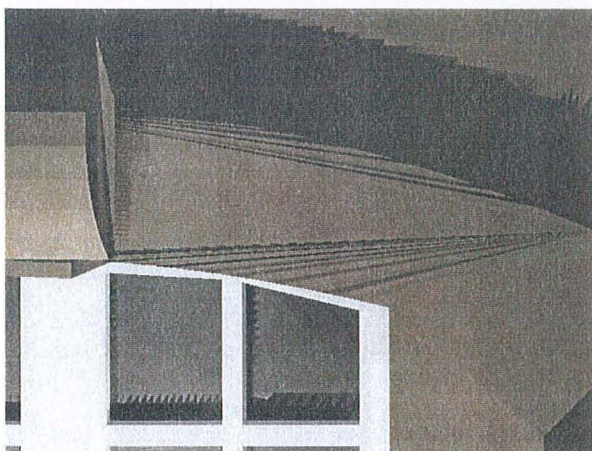
(a) Localisation des exemples de maillage



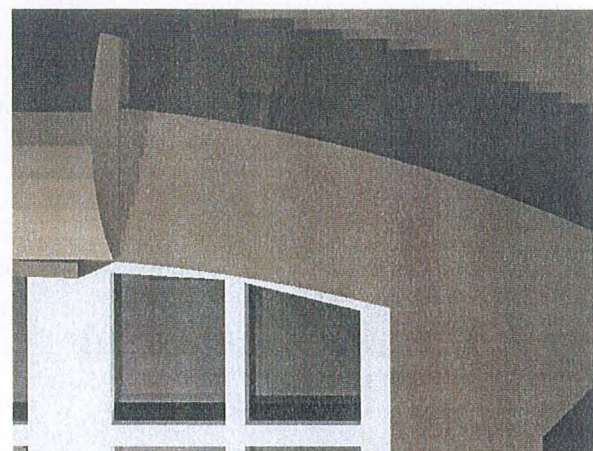
(b) Fenêtre du premier étage

Scène triangulée

Scène avec le maillage virtuel

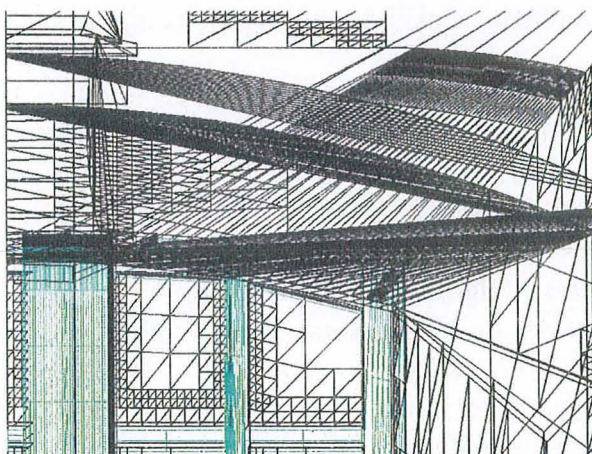


(c)

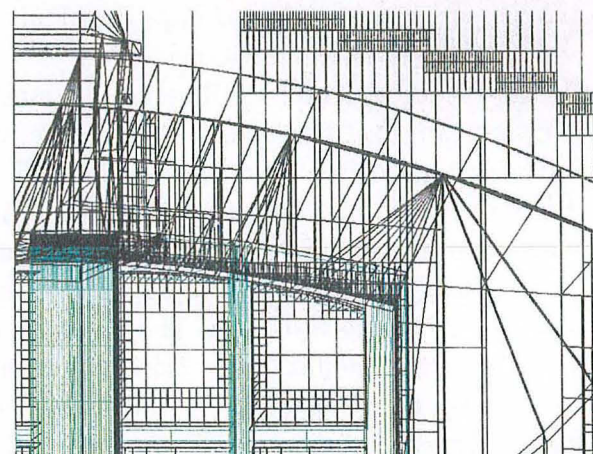


(d)

Illumination



(e)



(f)

Subdivisions

FIG. 3.17 – Utilisation des surfaces planes à bords complexes.

3.6.1.2 Résultats qualitatifs

L'observation de la figure 3.17 nous amène à une première constatation : l'utilisation du maillage virtuel pour gérer les polygones complexes permet d'obtenir des simulations visuellement plus satisfaisantes que lorsque l'on passe par une étape de triangulation. En effet, le fait de séparer la représentation de la fonction de radiosité de la géométrie sur laquelle elle s'applique et d'utiliser des mailles normalisées comme nouveaux supports permet de faire une meilleure approximation de la radiosité.

- Les figures 3.17(c) et 3.17(d) nous permettent d'apprécier l'efficacité du découpage créé lorsque l'on utilise le maillage virtuel pour intégrer directement les polygones complexes dans les calculs de radiosité. Nous pouvons remarquer, sur la figure 3.17(c), les nombreux triangles allongés engendrés par la triangulation basée sur OPEN-GL. Avec cette triangulation, les subdivisions n'arrivent pas à suivre correctement l'ombre du chambranle de la fenêtre sur le carreau de droite. En revanche, notre approche génère un maillage régulier, mieux adapté pour faire une approximation de la fonction de radiosité comme le montre la figure 3.17(d).
- La fonction de radiosité représentée sur la figure 3.17(e) a quelques défauts. Nous remarquons, notamment, des traits noirs sur la partie du bâtiment entourant la fenêtre. Ils sont dus aux problèmes posés par les triangles très allongés dans les calculs de visibilité lorsque l'on utilise notre algorithme avec le Z-Buffer. D'autre part, l'ombre projetée sur le carreau droit de la fenêtre est fortement crénelée, à cause de la triangulation initiale de la fenêtre. Ces problèmes n'apparaissent pas sur la figure 3.17(f).

3.6.1.3 Résultats quantitatifs

En plus des résultats qualitatifs présentés ci-dessus, nous nous sommes intéressés aux performances quantitatives du maillage virtuel pour gérer les polygones complexes. Nous avons pour cela comparé le temps de calcul et la mémoire utilisée entre les deux algorithmes avec et sans le maillage virtuel.

De façon évidente, l'utilisation du maillage virtuel diminue fortement le nombre de surfaces initiales. Le tableau 3.2 montre, en effet, que le fait de trianguler les surfaces initiales¹⁶ augmente leur nombre de 88 % pour la scène de l'Opéra. Il est, par ailleurs, intéressant de constater que le nombre de mailles finales a aussi largement diminué (49 %), mais dans une proportion, néanmoins, moindre que pour les surfaces initiales.

Ce dernier résultat était prévisible. En effet, l'algorithme de radiosité à base d'on-delettes subdivise fortement la géométrie initiale. Le grand nombre de mailles dans la scène après la simulation est principalement dû à ces subdivisions qui sont d'autant plus concentrées que la fonction de radiosité à représenter est compliquée. L'impact de la forme initiale des surfaces n'est donc plus la principale source de subdivisions géométriques. Malgré cela, il ressort du tableau 3.2 que notre algorithme fournit un processus de subdivision plus efficace puisque nous atteignons la convergence avec moins de subdivisions.

Le fait que notre algorithme fournisse une stratégie de subdivision plus efficace apparaît aussi dans les temps de calcul présentés dans le tableau 3.2. En effet, le temps pour

16. Nous n'avons pas triangulé les parallélogrammes.

Primitives géométriques utilisées	Nombre de primitives initiales	Nombres de mailles finales	Temps de calcul	Nombre de surfaces ayant shooté leur énergie
Triangles & parallélogrammes	32 429	819 109	476 s	9 358
Polygones	17 272	550 159	241 s	6 776
Ratio	1,88	1,49	1,98	1,38

TAB. 3.2 – Résultats numériques obtenus pour l'illumination de la scène de l'Opéra avec 16 processeurs et 1 Go de mémoire vive. Le temps de calcul et le nombre de surfaces ayant shooté leur énergie correspondent à un taux de convergence de l'algorithme de 99%.

atteindre la convergence avec la base de données triangulée est quasiment deux fois plus long que celui nécessaire avec le maillage virtuel. Notre définition du taux de convergence nous a aussi permis de suivre son évolution au fil des itérations. La figure 3.18 montre cette évolution et confirme le fait que le maillage virtuel permet d'aller, environ, deux fois plus vite que l'algorithme avec triangulation quelle que soit la convergence désirée.

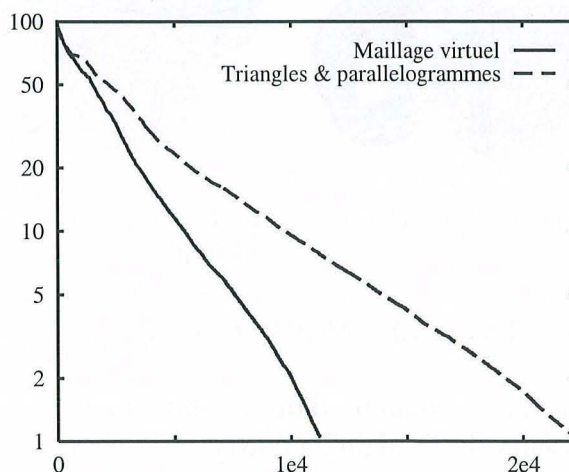


FIG. 3.18 – Taux de convergence en fonction du temps de simulation pour la scène de l'Opéra.

La dernière remarque que l'on peut faire est que le gain en temps est supérieur au gain en nombre de surfaces. Ceci se comprend aisément si on se rappelle que l'algorithme de radiosit  dépend fortement du nombre de surfaces initiales notamment pour les requêtes de visibilité qui sont les plus consommatrices en temps de calcul.

3.6.2 Résultats avec les surfaces paramétriques

3.6.2.1 Considérations g n rales

La sc ne que nous avons utilis e pour  tudier l'impact du maillage virtuel pour g rer les surfaces quadriques dans l'algorithme de radiosit  est la sc ne de la th i re. Cette sc ne,

qui est composée de 12 surfaces quadriques (des ellipsoïdes, des cylindres, des cônes et des hyperboloïdes à une nappe) et de 7 polygones complexes, a été créée grâce à un modèle CSG issu du logiciel de modélisation à base de surfaces paramétriques SGDL¹⁷. Le modèle original est représenté sur la figure 3.19(a) alors que sa version maillée en surfaces planes est représentée sur la figure 3.19(b). Cette dernière version, qui est une approximation de relativement faible précision du modèle original, comporte 1 392 primitives dont la plupart sont des parallélogrammes et des triangles. Chaque point du modèle maillé est associé à la normale qui lui correspond réellement sur la théière originale. Nous avons ainsi pu interpoler les normales sur toutes les surfaces durant les calculs. La scène utilisée pour la simulation comprend en plus une source ponctuelle blanche dont la lumière se réfléchit sur un plan rouge pour venir éclairer la face cachée de la théière. Cette scène permet de tester l'efficacité des surfaces quadriques pour la phase d'illumination directe mais aussi pour les inter-réflexions.

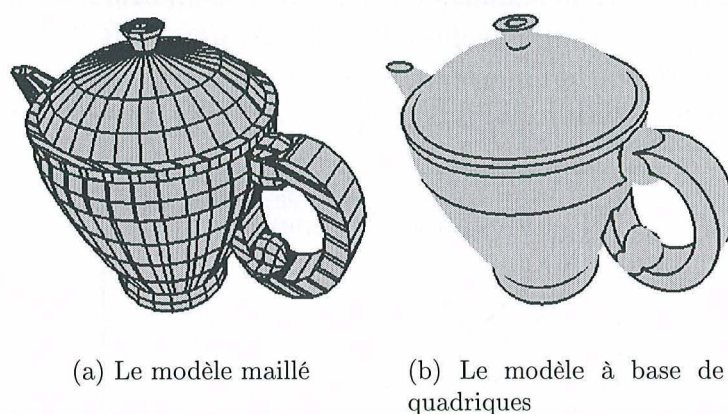


FIG. 3.19 – *Modèle de la théière.*

Tous les résultats que nous donnons dans la suite pour la scène de la théière ont été obtenus sur 4 processeurs de l'Origin2000 avec une mémoire maximum autorisée de 256 Mo.

Afin de bien étudier l'impact de la projection à Jacobien constant lors de l'utilisation du maillage virtuel sur cette scène, nous ne nous sommes pas limités à la comparaison des simulations pour la scène originale et pour la scène maillée. Nous avons fait une troisième série de simulations en conservant le modèle original mais en utilisant la projection définie par les paramètres naturels des quadriques, projection qui n'est généralement pas à Jacobien constant.

Enfin, pour chacune des trois situations ci-dessus, nous avons testé trois différentes bases d'ondelettes : les ondelettes de HAAR (\mathcal{M}_1), les ondelettes multiples d'ordre 2 (\mathcal{M}_2) et les ondelettes multiples d'ordre 3 (\mathcal{M}_3). Nous avons démontré, dans le chapitre précédent, le fort pouvoir d'approximation des fonctions ondelettes et nous avons voulu vérifier avec ces tests que ce pouvoir s'appliquait aussi sur des surfaces paramétriques — intuitivement on s'attend à avoir des résultats d'autant meilleurs que les primitives sont complexes

17. <http://www.sgd1.com>

car le pouvoir d'approximation des ondelettes peut alors être utilisé au maximum de son potentiel.

3.6.2.2 Résultats qualitatifs

La figure 3.20 présente les résultats visuels que nous obtenons pour nos 9 tests. Ces résultats correspondent tous au même taux de convergence (tel qu'il est défini dans la section précédente) de 99 % de l'algorithme de radiosité utilisé avec les mêmes paramètres de simulation.

On constate sur cette figure 3.20 que les résultats sont visuellement équivalents pour les deux séries de simulations avec le modèle original. Par contre, la simulation est loin d'être visuellement correcte avec la scène maillée. Dans ce dernier cas, l'interpolation des normales permet de récupérer la continuité de la fonction aux endroits où la fonction de visibilité ne change pas. Par contre, de gros artefacts apparaissent aux abords des discontinuités dues aux limites d'ombres. Ces artefacts sont dus au fait que, par rapport à une surface plane, une source ponctuelle est, soit totalement visible, soit totalement invisible. Ainsi, la meilleure discontinuité de visibilité que l'on peut obtenir sur cette scène est définie par le maillage initial de la théière. Ce phénomène explique l'effet de crénelage que l'on observe sur toutes les vignettes de la rangée du haut.

Une deuxième constatation que l'on peut faire directement à partir de la figure 3.20 est que la simulation sur le modèle à base de quadriques est visuellement d'autant meilleure que les fonctions utilisées sont d'ordre élevé. Ce résultat n'est pas surprenant car nous avons vu dans le chapitre précédent que pour un même seuil de l'oracle de subdivision, les ondelettes d'ordre élevé donnaient un résultat de meilleure qualité physique et visuelle que les ondelettes d'ordre moins élevé. Ce résultat s'applique, logiquement, aussi sur les surfaces quadriques. De façon contradictoire, nous n'observons pas d'amélioration visuelle pour le modèle maillé lorsque l'ordre des fonctions augmente. Bien que le modèle maillé représente une approximation relativement grossière de la théière originale, ce dernier phénomène est sans doute dû au fait que les surfaces qui composent le modèle sont trop nombreuses pour la complexité de la fonction de radiosité. Cette dernière varie très peu sur chaque surface et les ondelettes d'ordre élevé ne font qu'augmenter la complexité des calculs sans pouvoir faire bénéficier de leur efficacité pour faire des approximations de fonctions complexes.

La dernière remarque qualitative concerne la précision du modèle. Le fait d'avoir la possibilité d'utiliser des surfaces complexes directement dans les calculs de radiosité permet de ne pas avoir recours à une approximation géométrique. Les surfaces quadriques permettent de conserver toutes les informations topologiques et de les prendre en compte dans les calculs. L'effet de crénelage, qui est un peu atténué par l'interpolation des normales dans le modèle de la théière, est beaucoup plus ennuyeux sur le modèle du jeu d'échecs présenté sur la figure 3.21. Là encore, les problèmes de visibilité engendrent des erreurs rendant impossible une visualisation de grande qualité.

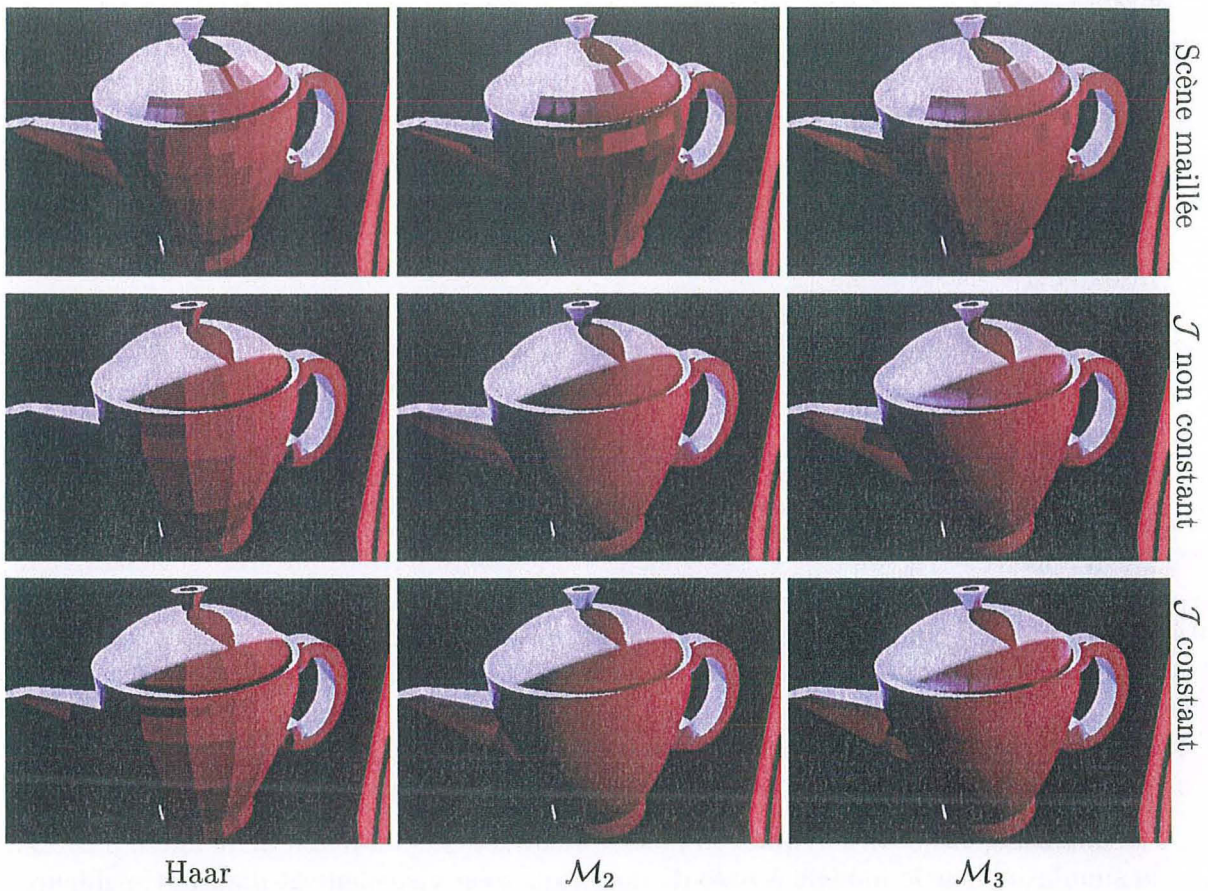


FIG. 3.20 – Résultats sur la scène de la théière. La rangée du haut correspond à l'utilisation d'un algorithme classique avec la scène maillée, la rangée du milieu correspond à l'algorithme avec le maillage virtuel mais avec des fonctions de projection définies par les paramètres naturels pour lesquels le Jacobien peut ne pas être constant et la rangée du bas correspond à l'utilisation du maillage virtuel.

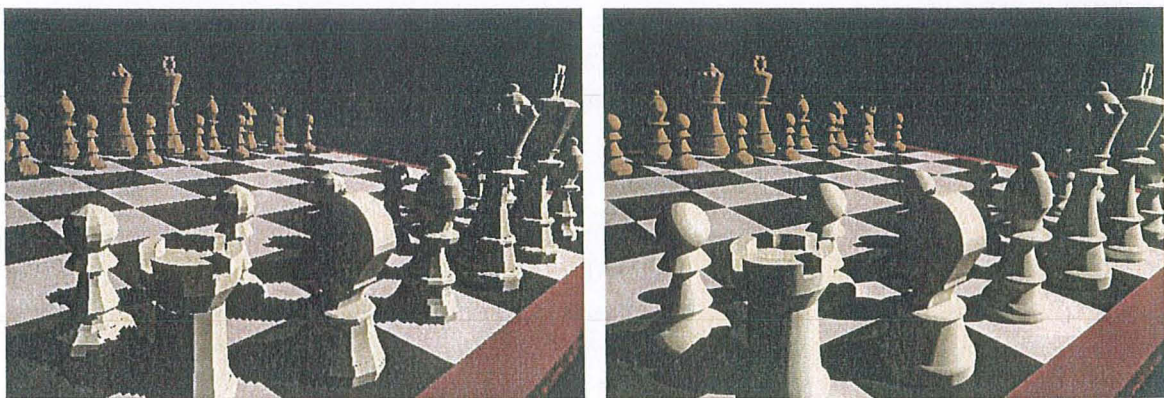


FIG. 3.21 – Jeu d'échecs maillé (a) et modélisé à base de quadriques (b).

3.6.2.3 Résultats quantitatifs

Les résultats quantitatifs que nous avons obtenus pour la scène de la théière sont rapportés dans le tableau 3.3.

Primitives géométriques utilisées	Fonctions ondelettes utilisées	Nombre de primitives initiales	Nombre de mailles finales	Temps de calcul	Nombre de surfaces ayant <i>shooté</i> leur énergie
Polygones avec interpolation des normales	\mathcal{M}_1	1 392	13 509	260 s	733
	\mathcal{M}_2		8 962	321 s	714
	\mathcal{M}_3		8 983	501 s	734
Quadriques avec projection « naturelle »	\mathcal{M}_1	19	85 945	554 s	19
	\mathcal{M}_2		27 811	256 s	
	\mathcal{M}_3		20 536	298 s	
Quadriques avec projection à $\mathcal{J} = \text{constante}$	\mathcal{M}_1	19	52 897	257 s	19
	\mathcal{M}_2		26 335	204 s	
	\mathcal{M}_3		19 675	180 s	

TAB. 3.3 – Résultats numériques obtenus pour l'illumination de la scène de la théière avec 4 processeurs et 256Mo de mémoire vive. Le temps de calcul et le nombre de surfaces ayant *shooté* leur énergie correspondent à un taux de convergence de l'algorithme de 99%. La projection « naturelle » correspond à l'utilisation du maillage virtuel avec les paramètres naturels des quadriques.

Afin de faciliter la lecture des résultats, nous avons reporté les temps de calcul que nous avons mesurés lors de nos tests sur les graphiques de la figure 3.22 et le nombre de mailles finales sur la figure 3.23.

La première constatation que l'on peut faire en regardant la figure 3.22 est que l'algorithme le plus rapide est l'algorithme du maillage virtuel à Jacobien constant avec les fonctions ondelettes multiples d'ordre 3 (figure 3.22(c), \mathcal{M}_3 : temps = 180s). Ce résultat est très important car il confirme l'hypothèse que nous avons faite tout au long de cette thèse :

Plus les algorithmes utilisés sont implantés et utilisés de façon générique avec un niveau d'abstraction élevé, plus ils permettent de converger rapidement vers une solution de très haute qualité sur des scènes de grande complexité.

Si on considère les différences de temps de calcul pour les trois simulations de la figure 3.22(c) avec le maillage virtuel et une fonction de projection à Jacobien constant, on constate que le temps de calcul diminue régulièrement quand l'ordre des ondelettes augmente. Ceci prouve que notre algorithme permet d'exploiter au maximum le pouvoir des fonctions d'ordre élevé pour faire rapidement une approximation de très bonne qualité d'une fonction de radiosité qui varie beaucoup. On remarque d'ailleurs sur la figure 3.23(c) que le nombre de mailles finales nécessaires pour représenter la fonction de radiosité avec

la même précision est 2,5 fois plus petit avec les ondelettes \mathcal{M}_3 qu'avec les ondelettes de HAAR, ce qui confirme la conclusion précédente.

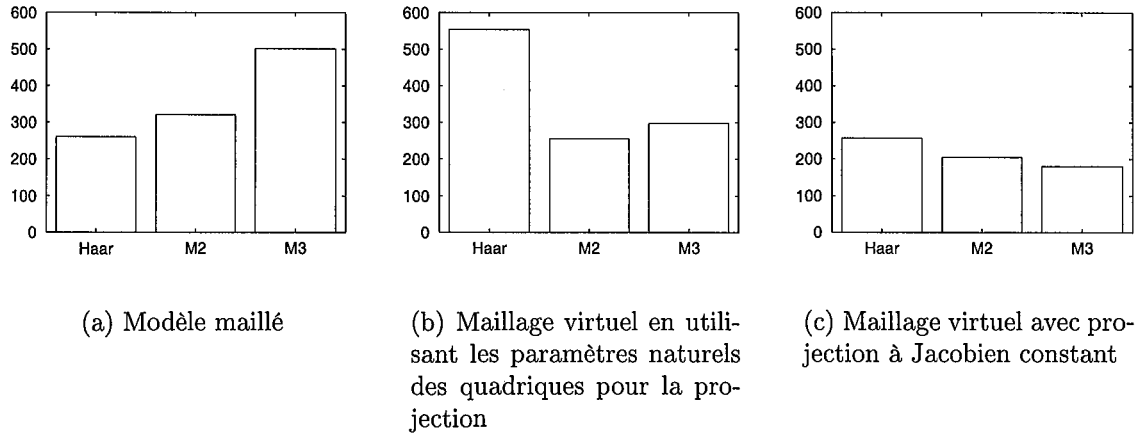


FIG. 3.22 – Temps de calcul (sec) pour les tests sur la théière

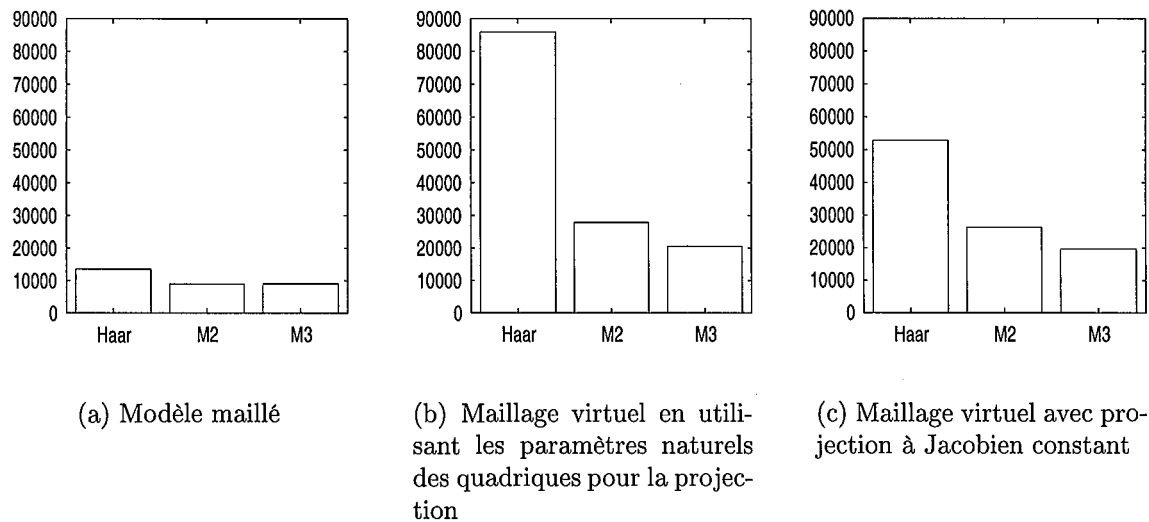


FIG. 3.23 – Nombre de mailles finales pour les tests sur la théière

La deuxième constatation concerne la simulation effectuée sur la scène maillée (figure 3.22(a)). Pour cette scène, le temps de calcul augmente avec l'ordre des fonctions de base. Cette constatation rejoint la remarque que nous avons faite dans la section précédente à propos du nombre de surfaces initiales important, comparé à la précision demandée pour la fonction de radiativité. Dans ce cas, les ondelettes d'ordre élevé ne peuvent pas apporter une amélioration de l'approximation. Par contre, elles impliquent des calculs d'interaction plus compliqués et donc plus longs que ceux avec des fonctions d'ordre moins élevé.

Le cas des simulations avec le maillage virtuel en utilisant la fonction de projection définie par les paramètres naturels est un peu plus compliqué (figure 3.22(b)). Les paramètres

naturels ne fournissent généralement pas une fonction de projection à Jacobien constant. Il faut donc calculer les coefficients de *push-pull* numériquement à chaque niveau de la hiérarchie. Étant donné que les coefficients de *push-pull* sont multipliés et additionnés entre eux lors de l'étape de *push-pull*, il est nécessaire de les calculer avec une grande précision. Comme les coefficients sont calculés par quadrature et que ces calculs sont relativement longs, le temps global de l'algorithme dépend fortement de la précision demandée. Pour être le plus juste possible, nous avons choisi la précision minimum générant un résultat visuel comparable à celui obtenu avec la projection à Jacobien constant. Malgré cette faible précision (qui peut engendrer des artefacts dans d'autres simulations) les temps de calcul sont de 25 % à 50 % plus longs qu'avec la projection à Jacobien constant. Évidemment pour une précision plus grande le temps de calcul augmente encore et pénalise d'autant plus la projection à Jacobien constant. Enfin, il est intéressant de constater que le temps de calcul, pour ces simulations, ne varie pas régulièrement suivant l'ordre des fonctions de base. Avec les ondelettes de HAAR, le temps est quasiment deux fois plus long car il faut nettement plus de subdivisions — donc plus de calculs de coefficients de *push-pull* — pour atteindre la précision demandée. En effet, on peut remarquer sur la figure 3.23(b) que le nombre de mailles finales est environ 3 fois plus élevé avec les ondelettes de HAAR qu'avec les autres bases. Par contre, le gain en nombre de subdivisions avec les ondelettes multiples d'ordre 3 n'est pas suffisant pour compenser la complexité du calcul des coefficients de *push-pull* et le temps est légèrement supérieur à celui avec les ondelettes d'ordre 2.

De façon générale, on peut constater sur la figure 3.23 que la place mémoire (qui est quasiment proportionnelle au nombre de mailles finales) diminue quand l'ordre des fonctions augmente sur nos trois séries de tests. Ce résultat confirme celui donné dans le chapitre précédent.

Enfin, il est à noter que la scène maillée génère toujours beaucoup moins de mailles finales que les autres simulations. Ce résultat, qui peut paraître surprenant, découle directement de la remarque qualitative que nous avons faite dans la section précédente. En effet, la majorité des subdivisions sont générées lors de la phase d'illumination directe pour représenter correctement les limites d'ombres qui correspondent à des variations violentes de la fonction de radiosit . Or, sur la scène maill e, les erreurs de visibilité engendr es par l'approximation g om etrique   base de surfaces planes font que ces limites d'ombres correspondent exactement aux contours des surfaces initiales. Il n'est donc pas n ecessaire, d'apr es les crit eres de l'oracle, de subdiviser d'avantage ces surfaces. En conclusion, la diminution du nombre de mailles finales n'est que le r esultat d'une erreur g om etrique, erreur qui g en ere des artefacts visuels importants.

3.7 Conclusion

Nous venons de pr esenter, dans ce chapitre, une nouvelle m ethode qui permet de g erer efficacement les surfaces param etriques dans les algorithmes de radiosit . Notre m ethode est bas ee sur le maillage virtuel qui est une abstraction de la g eom etrie initiale. Gr ace au maillage virtuel, il est possible de g erer directement des surfaces param etriques complexes dans l'algorithme de radiosit  sans recourir ni   une  tape de triangulation ni   une

quelconque approximation géométrique.

Nous avons montré que notre méthode était très générique et nous avons donné un algorithme général applicable pour n'importe quel type de surface paramétrique et avec n'importe quel algorithme de radiosité. Nous avons ensuite décrit précisément son implantation pour deux grandes familles de surfaces, les surfaces planes à bords complexes et les surfaces quadriques, dans notre algorithme de radiosité à base d'ondelettes. Cette implantation de l'algorithme de radiosité est la première qui permette de faire le lien entre les méthodes de GALERKIN hiérarchiques et les surfaces courbes.

Les résultats que nous avons obtenus avec notre implantation du maillage virtuel sont excellents : le maillage virtuel permet de gagner en précision et en efficacité, il fournit des résultats de meilleure qualité visuelle et physique et il permet de diminuer fortement les erreurs dues aux approximations géométriques faites en amont de l'algorithme de radiosité. Notre méthode permet de diminuer simultanément le temps de calcul et la mémoire nécessaire lors de simulations de grande qualité.

Grâce à sa formulation générique, le maillage virtuel n'est pas limité à l'algorithme de radiosité avec la méthode de GALERKIN. Nous envisageons, d'ailleurs, de l'implanter dans une méthode de radiosité à base de *clustering*.

Nous pensons que le maillage virtuel peut être utilisé pour d'autres applications : nous travaillons actuellement à son utilisation en collaboration avec un algorithme de maillage de discontinuité. Nous espérons ainsi bénéficier de la précision de ce dernier et de l'efficacité du maillage virtuel pour aller un cran plus loin dans la précision et le réalisme de nos simulations.

Une autre direction de recherche que nous aimerions engager concerne les surfaces algébriques de degré élevé. Nous pensons, cependant, que les fonctions de projection à Jacobien constant risquent de devenir rapidement trop compliquées pour être utilisables dans les calculs de radiosité. Il sera alors, peut-être, plus efficace de faire des approximations de ces surfaces (notamment les B-Splines ou le NURBS) au moyen de quadriques. Des recherches ont déjà été faites dans ce sens par B. LACOLLE et N. SZAFRAN et sont présentées dans [Lac 99].

Chapitre 4

Expérimentations

« C'est par l'expérience que progressent la science et l'art »

ARISTOTE, *Méthaphysique*

Sommaire

4.1	Protocole expérimental	138
4.1.1	Caractéristiques matérielles et logicielles	139
4.1.2	Critères de performance	139
4.1.3	Scènes de test	141
4.2	Résultats	146
4.2.1	Modélisation plus naturelle	147
4.2.2	Simulations plus efficace	148
4.2.3	Moins de place en mémoire	148
4.2.4	Simulations plus rapides	149
4.2.5	Simulations très réalistes	150
4.3	Autres solutions	155
4.3.1	Algorithme parallèles	156
4.3.2	<i>Clustering</i>	156
4.3.3	<i>Face cluster</i>	157

Les techniques de radiosité sont issues d'un domaine de recherche très actif depuis une dizaine d'années. De nombreuses extensions et améliorations sont présentées chaque année par la communauté scientifique. Pourtant, peu de comparaisons ont été faites entre les différentes solutions possibles pour aboutir au résultat sur des scènes du monde réel.

Par ailleurs, seules quelques rares publications très intéressantes ont été faites sur l'analyse formelle des problèmes rencontrés et des erreurs inhérentes au processus de résolution [Arv 95, Arv 94]. Malheureusement, celles-ci ne permettent pas de comparer quantitativement deux alternatives concrètes de résolution.

Pour vérifier l'efficacité de notre algorithme de radiosité à base d'ondelettes avec le maillage virtuel, nous avons effectué des tests sur des scènes qui ne sont pas spécialement

prévues pour des simulations numériques. Les scènes que nous avons choisies sont presque toutes issues d'applications du monde réel et ont, en général, une utilité dans le monde industriel. Nous avons comparé nos simulations avec des résultats obtenus avec des algorithmes plus classiques. Les scènes utilisées ici sont nettement plus complexes que les scènes de tests classiques et nous n'avons effectué que quelques comparaisons pour des raisons de faisabilité :

- **matérielle** : certaines simulations demandent trop de mémoire et ne sont matériellement pas faisables ;
- **temporelle** : certaines simulations ne sont pas faisables en un temps raisonnable si on n'utilise pas les algorithmes les plus performants ;
- **logicielle** : certaines scènes ont été modélisées avec des surfaces courbes et n'ont pas d'équivalent sous forme de polygones ;

Nous avons décidé, à l'instar de Andrew WILLMOTT et Paul HECKBERT [Wil 97a], de définir précisément un protocole expérimental afin de pouvoir tirer des conclusions pertinentes de nos expériences. Les choix des critères de performance et des scènes de tests précisent les domaines de validité des résultats et nous permettent de garder un jugement assez critique et objectif lors de l'analyse des résultats.

Dans ce chapitre, nous commençons par présenter le protocole expérimental dans la section 4.1 (p. 138). Il regroupe les différents critères de performance utilisés qui vont servir à juger du résultat et décrit les scènes de test choisies pour leurs caractéristiques différentes. Puis, nous exposons, dans la section 4.2 (p. 146), les résultats qualitatifs et quantitatifs que nous avons obtenus sur les nouvelles scènes de tests. Enfin, nous terminons, dans la section 4.3 (p. 155), par présenter les autres approches qui ont été développées pour illuminer des scènes industrielles de grande complexité.

4.1 Protocole expérimental

Les expérimentations des différents algorithmes doivent être faites avec une extrême rigueur en vue de tirer des conclusions pertinentes. Il faut se méfier des *effets de bords* qui peuvent apparaître lors des comparaisons des simulations. Ainsi, il est très important d'effectuer tous les tests dans un environnement ayant toujours les mêmes caractéristiques matérielles et logicielles.

Par ailleurs, pour être pertinentes, les expérimentations doivent suivre un protocole expérimental permettant de cerner exactement les points que l'on veut tester. Nous avons basé notre protocole sur deux choix déterminants :

- **les critères de performance**, qui nous permettent de juger et de quantifier les résultats obtenus. Tout en restant prudent par rapport à leur exploitation, ces chiffres nous aident à avoir un regard plus objectif sur nos conclusions ;
- **les scènes de test**, qui doivent être un échantillon représentatif des scènes issues d'applications du monde réel.

Dans cette section, nous présentons tout d'abord les caractéristiques matérielles et logicielles avec lesquelles nous avons effectué nos tests dans la section 4.1.1 (p. 139). Puis,

dans la section 4.1.2 (p. 139), nous présentons les critères de performances qui vont nous permettre d'étudier nos résultats et éventuellement de les comparer avec d'autres. Enfin, dans la section 4.1.3 (p. 141), nous décrivons précisément les scènes de tests que nous utilisons.

4.1.1 Caractéristiques matérielles et logicielles

L'utilisation de la plate-forme Candela nous garantit une unité dans le style de programmation comme dans les structures utilisées. Ainsi, les résultats ne sont que peu influencés par la *griffe* du programmeur, comme cela aurait pu être le cas avec l'utilisation d'une plate-forme constituée de l'assemblage de parties de codes récupérées dans la communauté scientifique. Nous garantissons donc une cohérence de programmation pour les différentes alternatives étudiées.

D'autre part, nous avons utilisé une seule et unique machine pour tous nos tests : une machine Silicon Graphics de type ORIGIN2000 munie de 64 processeurs MIPS R10000 cadencés à 195 MHz et disposant d'une mémoire partagée de 24 Go. Pour tous nos tests, nous nous sommes limités à 32 processeurs et 4 Go avec l'implantation parallèle de notre algorithme faite par Xavier CAVIN [Cav 98].

Enfin, suite aux résultats du chapitre 2.5, nous avons décidé de faire nos tests sur ces scènes de grande taille avec les ondelettes linéaires. Ces ondelettes sont les mieux adaptées à la valeur de l'erreur globale que nous avons choisie et pour obtenir des résultats visuellement corrects. De plus elles fournissent des résultats adaptés aux capacités d'affichage des cartes graphiques. Nous n'avons pas fait de tests comparatifs avec différentes bases d'ondelettes car ils auraient été beaucoup trop longs et ils auraient nécessité beaucoup trop de ressources matérielles.

4.1.2 Critères de performance

Deux idées nous ont guidé lors du choix des critères de performance :

1. Ils doivent tout d'abord être en adéquation avec notre volonté d'étudier l'impact des alternatives de résolution sur des scènes conséquentes pour permettre de juger de la qualité du résultat en insistant sur le contrôle de l'erreur, tant du point de vue numérique que visuel. De plus, ces critères mettent en avant la robustesse des méthodes de résolution, à travers l'utilisation de la mémoire et des temps de calcul.
2. Par ailleurs, nous souhaitons pouvoir les exploiter de manière facile et sûre. C'est pourquoi nous avons choisi des critères mesurables, afin de pouvoir quantifier nos résultats. Pourtant, puisqu'il n'existe pas de méthode mathématique universelle pour mesurer la perception humaine des solutions calculées, nous nous sommes résolus à juger les aspects visuels des solutions.

4.1.2.1 Contrôle numérique de l'erreur

Nous avons vu dans le chapitre 2.5.1.5 (p. 72) que l'erreur locale relative au seuil fixé dans les fonctions oracles et l'erreur globale relative à la différence avec une solution de référence sont fortement corrélées. Comme nos tests sont faits sur des scènes du monde

réel, il n'existe pas de solution de référence utilisable. Nous avons donc décidé de contrôler l'erreur uniquement par le réglage du paramètre local. En utilisant les résultats du chapitre 2.5.1.5 (p. 72), nous avons réglé le paramètre local pour qu'il corresponde à une erreur globale d'environ $5 \cdot 10^{-3}$ ce qui donne un résultat visuel de grande qualité où les artefacts sont très rares.

Pour pouvoir comparer les résultats entre eux et pour assurer la cohérence de nos tests, nous avons mené tous les calculs jusqu'à une convergence de 95 % de l'algorithme. Nous avons donc arrêté les itérations de l'algorithme quand l'énergie résiduelle restant à émettre dans la scène représente moins de 5 % de l'énergie initiale.

4.1.2.2 Contrôle visuel de l'erreur

Le contrôle numérique de l'erreur (même quand il est indépendant du point de vue) ne peut pas refléter toute la complexité de la perception humaine. Certaines approches ont été proposées pour prendre en compte la sensibilité humaine des spectres, mais aucune n'est vraiment convaincante.

Nous nous sommes donc résolus à utiliser un critère empirique et subjectif. Nous jugeons de la qualité du résultat par rapport à la perception que nous avons d'images de sa visualisation sur écran.

Après ces deux critères qui permettent de juger de la qualité du résultat obtenu, les deux critères suivants caractérisent les performances du processus de résolution.

4.1.2.3 Utilisation de la mémoire

Tous les programmes ont besoin de mémoire vive pour pouvoir s'exécuter. Cette mémoire est limitée et devient donc un facteur limitatif pour les programmes en général, et pour ceux de synthèse d'images en particulier.

Suite aux progrès techniques en informatique de ces dernières années, et à la banalisation des images de synthèse en tout genre, les scènes qui sont simulées sont de plus en plus détaillées et complexes. Les besoins en mémoire sont ainsi décuplés et rendent cette ressource critique.

Nous avons décidé de prendre en compte la mémoire totale utilisée lors des calculs. Cette mémoire ne prend pas en compte la taille initiale du programme, qui ne varie pas suivant l'approche choisie. Comme notre algorithme ne stocke rien d'autre en mémoire que les structures spatiales et fonctionnelles, la mémoire utilisée lors des calculs est proportionnelle au nombre de mailles finales générées par l'algorithme. En suivant le nombre de surfaces générées et en connaissant la mémoire nécessaire par surface, cela nous permet de juger de la possibilité d'utiliser les approches de radiosité à base d'ondelettes pour des applications du monde réel. De plus, la mémoire nécessaire par surface étant fortement dépendante de l'implantation, nous avons choisi d'étudier les variations d'utilisation mémoire en suivant le nombre de surfaces générées.

4.1.2.4 Temps de calcul

Pour finir, le temps de calcul est le deuxième critère incontournable. D'une part, c'est un critère universel pour comparer différentes solutions. D'autre part, un algorithme n'est performant que s'il se termine en un temps raisonnable.

Le temps CPU nous donne le temps effectivement passé à faire les calculs. Son point fort est de ne pas être affecté par la charge de la machine, ni par le surcoût de la gestion de plusieurs processus par le système d'exploitation. Il donne donc des résultats constants, même si la machine est chargée par d'autres processus. Par contre, ce temps ne prend pas en compte la gestion des périphériques. Tous les temps d'attente, d'accès à la mémoire de la carte graphique, de lecture ou d'écriture sur les disques durs, etc., ne sont pas inclus. Donc ce temps CPU ne reflète pas exactement le temps de travail de l'ordinateur pour effectuer la résolution.

Pour contourner ce problème, nous avons mesuré le temps *utilisateur*, qui correspond au temps réel passé à faire les calculs. Cette mesure, qui prend en compte la gestion des périphériques, dépend fortement de la charge de la machine. Pour éviter de fausser nos résultats à cause de la charge due aux autres utilisateurs, nous avons utilisé la machine en mono-utilisateur. Le temps ainsi mesuré, même s'il ne correspond pas à strictement parler à des calculs effectifs des processeurs, est directement dû à l'algorithme de résolution.

4.1.3 Scènes de test

Un des objectifs de ce travail est de juger la capacité des méthodes de radiosité à base d'ondelettes à être utilisées pour des applications du monde réel. Ceci nous a poussé à choisir des scènes de test qui ne sont pas particulièrement adaptées aux méthodes de radiosité. Nous avons préféré utiliser certaines scènes qui proviennent d'applications réelles.

Les scènes utilisées par Simon GIBSON et R. J. HUBBOLD dans [Gib 96] font référence en terme de complexité pour des simulation de radiosité. Leur taille est comparable à celle de nos scènes de tests (225 000 surfaces initiales au maximum) mais leur géométrie est uniquement constituée de triangles et de parallélogrammes. Par opposition, la géométrie dans nos scènes est constituée de surfaces quelconques, c'est-à-dire planes ou courbes, convexes ou concaves et même trouées. De plus, l'orientation de ces surfaces est généralement quelconque, bien qu'elle soit majoritairement alignée aux axes dans certaines scènes architecturales.

4.1.3.1 Opéra de Nancy

L'Opéra de Lorraine, sur la Place Stanislas à Nancy, a été modélisé en 1992 pour pouvoir simuler de nouveaux projets d'éclairage lors du renouvellement de l'illumination de la place. Cette scène est donc entièrement issue d'une application du monde réel.

Le modèle géométrique a été réalisé avec un modeleur du commerce (Arc+) par les techniciens de la ville de Nancy et les architectes du CRAI¹⁸. La modélisation a été faite

18. Centre de recherche en architecture et ingénierie, laboratoire de l'École d'Architecture de Nancy, dont une présentation est accessible à l'adresse <http://www.crai.archi.fr/>.

à partir de relevés architecturaux divers. Le résultat est une scène détaillée représentant fidèlement la réalité. L'opéra est composée de 34 614 surfaces initiales (figure 4.1).

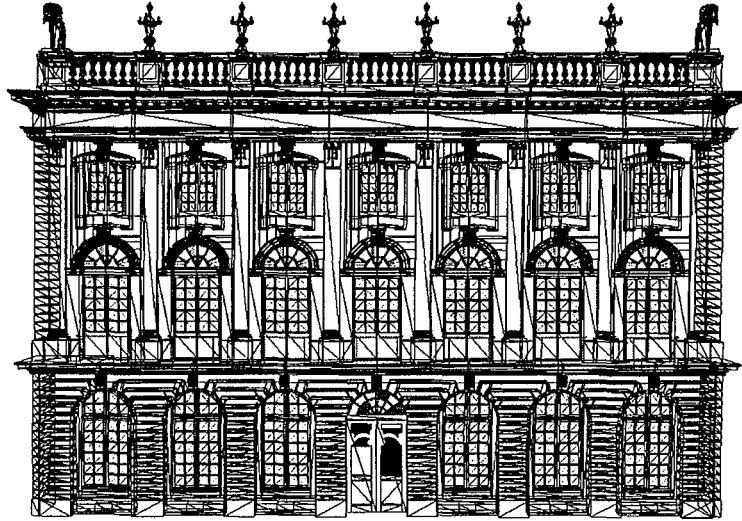
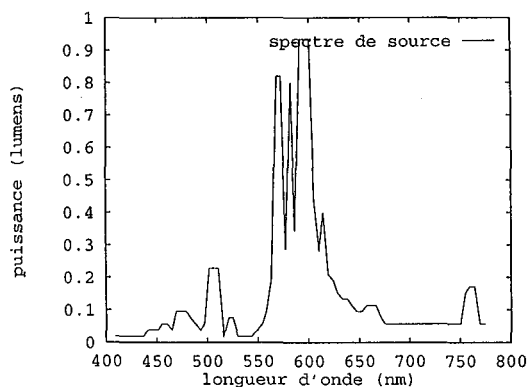


FIG. 4.1 – Modèle de l'Opéra de la place Stanislas.

Cette scène représente une façade de bâtiment, assez plane dans son ensemble. La majorité des surfaces sont orientées dans la même direction. Cette configuration fait que les inter-réflexions ont une faible influence sur le résultat, en comparaison à celle de l'illumination directe.

Le projet d'éclairage a été défini par une équipe de conception. Celle-ci a choisi les sources et leur positionnement par rapport à une ambiance qu'elle voulait donner à l'ensemble architectural. Il est donc évident que le placement des sources n'a pas été fait par rapport à la simulation informatique, ni pour la faciliter. De plus, les caractéristiques photométriques des sources proviennent de mesures de gonio-photomètres. Il faut donc prendre en compte les données directionnelles des 98 sources éclairant le bâtiment.



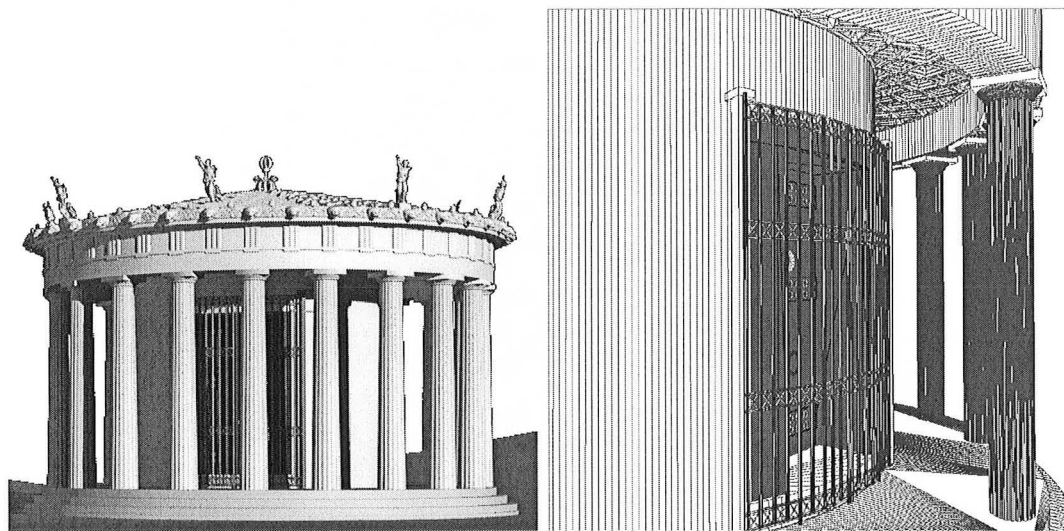
Pour finir, les données spectrales, que ce soit pour les sources ou les matériaux, proviennent elles aussi de mesures faites à l'aide d'un spectro-photomètre. Il faut donc faire particulièrement attention à la représentation des grandeurs spectrales, pour bien prendre en compte les pics spectraux des sources. Le spectre de la figure ci-contre montre la complexité des données spectrales décrivant les données photométriques de la scène

de l'Opéra. Ces données ont pu être prise en compte dans Candela grâce à une représentation fonctionnelle très générique des base spectrales.

4.1.3.2 Tholos de Delphes

La tholos de Delphes est un des trois temples circulaires recensés actuellement. Il fait partie du sanctuaire de Marmaria, sur le site archéologique de Delphes. Ce temple et les autres bâtiments du sanctuaire, tous détruits, ont été modélisés par le CRAI en vue de la restitution du sanctuaire en synthèse d'image.

La tholos a elle aussi été modélisée avec un modelleur du commerce, PDMS, à partir de relevés effectués par les archéologues. De plus, certains détails du bas-relief et quelques sculptures ont été acquis à l'aide d'un capteur laser. La scène totale illustrée par la figure 4.2 (a), au niveau de résolution le plus précis, contient environ 913 000 surfaces. Cette complexité rend actuellement son utilisation en radiosité difficile.



(a) Modèle total.

(b) Détail du modèle simplifié.

FIG. 4.2 – *Tholos de Delphes*.

Nous utilisons donc une base de données simplifiée et partielle de la tholos pour nos tests. Cette base est composée majoritairement de polygones qui ne sont pas des parallélogrammes. Parmi les 7 778 surfaces initiales, 4 512 sont des polygones complexes, 2 099 des parallélogrammes et 1 167 des triangles.

L'éclairage de cette tholos est dû à la lumière naturelle. Comme ce temple se situe en Grèce et que le marbre blanc est très réfléchissant, le soleil représente pratiquement toute l'énergie lumineuse apportée. L'influence de la voûte céleste peut être considérée comme nulle. Ainsi, nous avons modélisé la lumière du jour par une lumière directionnelle, d'intensité constante.

Dans la simulation de cette scène, les inter-réflexions ont une très grande importance. En effet, de nombreuses surfaces ne sont pas directement éclairées. Leur seul apport de lumière vient donc des inter-réflexions.

4.1.3.3 Reality Center de Cortaillod

La scène du Reality Center de Cortaillod a été modélisée par nos soins pour servir de démonstration à Silicon Graphics dans son Reality Center à Cortaillod. Cette scène représente un environnement immersif de visualisation avec une pièce contenant des chaises pour les utilisateurs, un pupitre de commande, trois projecteurs tri-tubes, des éclairages d'ambiance et un écran sphérique de 6 mètres de diamètre (voir figure 4.3(b)).

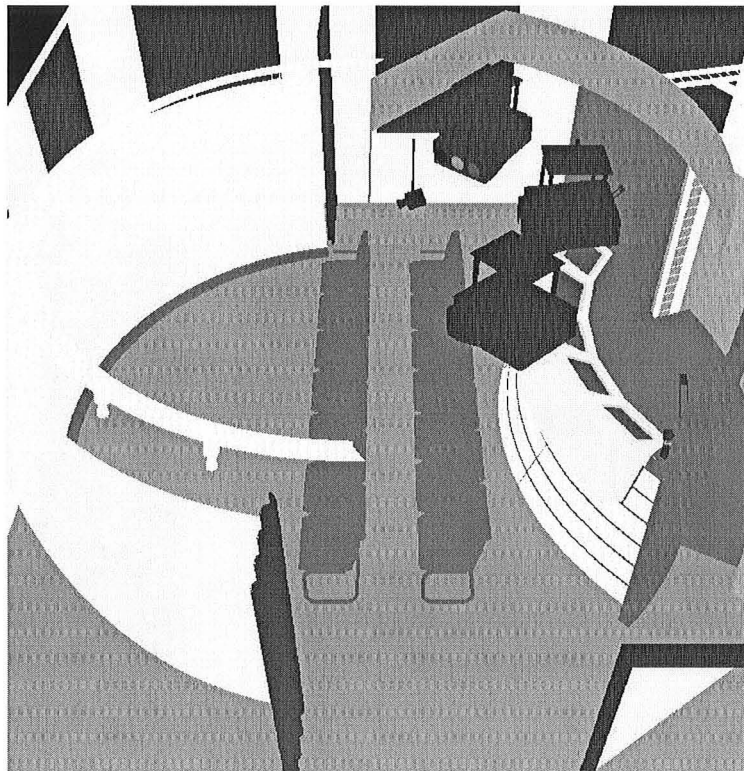


FIG. 4.3 – *La scène du Reality Center de Cortaillod.*

Le projet a débuté à partir des plans architecturaux (sur papier voir figure 4.3(a)) du Reality Center de Cortaillod. En utilisant ces plans, nous avons réalisé un premier modèle 3D grossier de la pièce (modèle en polygones). Ce travail a été réalisé par Jean-Pierr LAURENT de la société Neoxy.

Le modèle comporte un grand nombre d'objets courbes (écran sphérique, murs cylindriques, pupitre conique, chaises en ellipsoïdes et tores, ...). Ainsi le modèle 3D en polygones comporterait plus de 100 000 surfaces dans une version relativement précise, alors qu'il en comporte moins de 3 000 quand il est modélisé à base de quadriques.

La particularité de cette scène est donc qu'elle est majoritairement constituée de surfaces quadriques et qu'il aurait été impossible de d'utiliser la modélisation avec des polygones pour faire des simulations précises et réalistes.

4.1.3.4 Soda Hall de Berkeley

La scène de test du Soda Hall représente le bâtiment du département informatique de l'université de Berkeley. Nous avons récupéré la scène sous format VRML¹⁹, et l'avons utilisée directement telle quelle, sans traitement préalable.

Cette base est devenue incontestablement une référence dans le domaine, pour sa complexité en terme de nombre de surfaces. En effet, seuls deux exemples de simulation de la base complète en radiosité existent à notre connaissance dans des publications. Une des simulations a été faite par S. TELLER *et al.* [Tel 94] sur le bâtiment complet, mais sans mobilier, c'est-à-dire sur 39 979 polygones initiaux. La seconde réalisation, de Thomas A. FUNKHOUSER [Fun 96], portait sur cinq étages, mobilier compris, pour un total de 280 836 surfaces initiales. Il faut toutefois noter que ces deux simulations ont été faites avec un noyau de radiosité fondé sur un partitionnement de l'espace qui tirait particulièrement profit de l'agencement des surfaces. Celles-ci sont en effet très majoritairement parallèles aux axes et regroupées dans de nombreuses pièces. Cela permet de simplifier énormément les calculs de visibilité grâce à un pré-traitement adéquat.

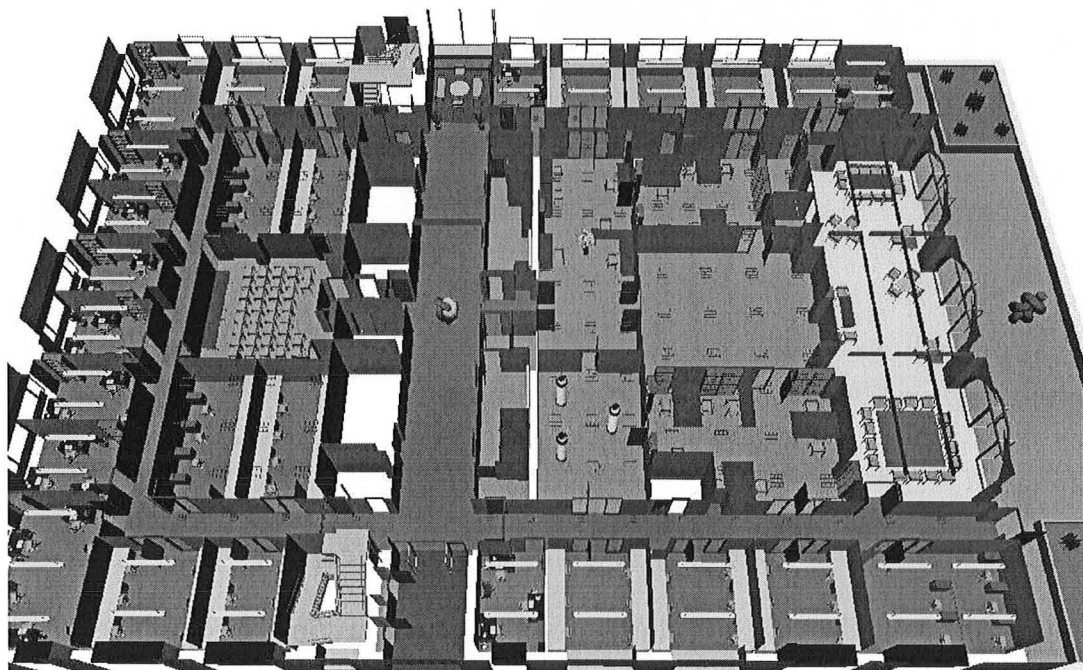


FIG. 4.4 – L'étage du Soda Hall de Berkeley (VRML).

Nous nous sommes restreints au 4^{ème} étage, mobilier compris, pour un total de 272 450 primitives si on ne s'autorise que des triangles et des parallélogrammes. La scène comporte 216 sources de lumières ponctuelles et 905 surfaces éclairantes servant à modéliser des néons et des sources surfaciques.

L'utilisation d'une base de cette envergure pour des calculs de radiosité à base d'ondelettes est un challenge. Nous étions d'autant plus intéressés pour utiliser cette base de

19. L'adresse web est <http://www.cs.berkeley.edu/~kofler/>

données que nous n'utilisons aucun pré-traitement de visibilité, ce qui prouve que notre algorithme est particulièrement robuste.

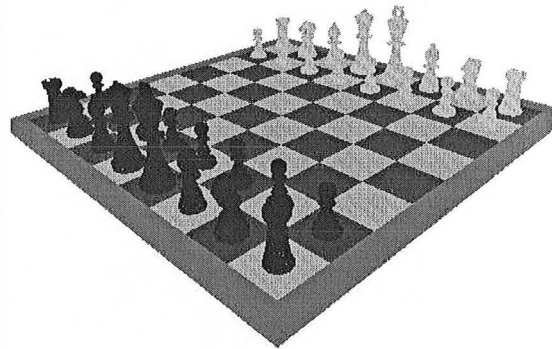
4.1.3.5 Salle de la géométrie

La salle de la géométrie regroupe un ensemble d'objets géométriques modélisés à base de quadriques que nous avons inclus dans une pièce particulière (la pièce 420A pour ceux qui connaissent) du 4^{ème} étage du Soda Hall. Cette ensemble contient un jeu d'échecs, une théière (comme celle utilisée pour les tests du chapitre 3.6), plusieurs bancs, des jarres et d'autres objets purement géométriques.

La figure 4.5 montre le modèle de la salle de la géométrie ainsi qu'un zoom sur le jeu d'échecs. Ce dernier est composé de 555 surfaces dont 264 quadriques alors qu'il aurait fallu plusieurs dizaines de milliers de triangles pour en avoir une représentation approximative ne permettant pas de zoomer comme sur la figure 4.5(b).



(a) La salle de la géométrie.



(b) Vue rapprochée du jeu d'échecs.

FIG. 4.5 – *Salle de la géométrie*

L'intérêt de ce modèle est d'étudier le comportement et l'impact d'objets complexes dans un modèle initialement prévu pour faire des simulations. De plus, ce modèle étant inclus dans celui du Soda Hall, il comporte un très grand nombre de surfaces et il nous donne l'occasion de tester le maillage virtuel (et notamment l'utilisation de surfaces quadriques complexes) de façon effective sur un environnement de taille industrielle.

4.2 Résultats

Les résultats que nous exposons ici regroupent l'ensemble des tests que nous avons effectués sur ces scènes. Étant données les contraintes matérielles et logicielles, ces résultats sont souvent partiels et ne concernent généralement qu'une partie des scènes présentées ci-dessus. L'ensemble de ces résultats forme néanmoins un tout cohérent qui confirme

l'efficacité du maillage virtuel pour simuler l'illumination globale au moyen de notre algorithme.

Notre présentation est décomposée en trois sections. Dans la première section 4.2.1 (p. 147), nous exposons les résultats relatifs à la modélisation et à l'impact du maillage virtuel sur la qualité et l'efficacité des modèles géométriques. Dans la deuxième section 4.2.2 (p. 148), nous donnons les résultats numériques relatifs aux simulations, c'est-à-dire que nous exposons l'effet du maillage virtuel sur le temps de calcul, la mémoire et la qualité numérique du résultat. Enfin, nous terminons par présenter nos résultats visuels par un grand nombre d'images dans la section 4.2.5 (p. 150).

4.2.1 Modélisation plus naturelle

4.2.1.1 Utilisation des surfaces planes quelconques

Les données géométriques de nos modèles ont une grande complexité inhérente. Nous avons déjà vu que la scène de l'Opéra pouvait être fortement simplifiée si on s'autorise l'utilisation de polygones complexes. Il en est de même pour la scène de la Tholos dont la plupart des surfaces ne sont ni des triangles ni des parallélogrammes. Enfin, le Soda Hall, qui a pourtant été modélisé spécialement pour des simulations, comporte lui aussi un grand nombre de surfaces planes relativement complexes.

Le tableau 4.1 montre que l'utilisation de polygones quelconques permet de diminuer fortement le nombre de primitives initiales même dans des scènes où toutes les surfaces sont, a priori, « simples ». Ainsi, le nombre de primitives dans le Soda Hall augmente de 37 % si on ne s'autorise que les triangles et les parallélogrammes comme primitives de base et ce gain est le plus petit remarqué sur les trois scènes testées.

Scène	Nombre de surfaces initiales		Ratio
	Triangles & parallélogrammes	polygones	
Opéra	32 429	17 272	1,88
Tholos	11 087	7 778	1,43
Soda Hall	272 450	199 550	1,37

TAB. 4.1 – Effet de l'utilisation de polygones complexes sur le nombre de primitives initiales

4.2.1.2 Modélisation à base de surfaces quadriques

Nous n'avons pas de résultats numériques similaires à ceux exposés ci-dessus pour les surfaces quadriques. En effet, une surface quadrique ne peut pas être modélisée de façon exacte par des polygones (sauf pour les plans qui sont des quadriques dégénérées). Pour comparer la modélisation à base de quadriques avec celle à base de polygones, il faut donc faire une approximation du modèle à base de quadriques. Selon la précision de cette approximation, le nombre de polygones générés peut varier énormément : une sphère peut être représentée grossièrement par un cube ou beaucoup plus précisément par plusieurs centaines de triangles.

Ainsi, nous avons décidé de ne pas remodeliser les scènes de la salle de la géométrie et du Reality Center qui sont fondamentalement modélisées à base de quadriques et dont nous ne pouvons pas donner d'approximation absolue à base de polygones. La scène de l'Opéra, quant à elle, ne contient aucune surface courbe. Nous l'avons donc laissée sous forme de polygones. Nous n'avons pas non plus remodelisé la Tholos.

Enfin, le seul modèle que nous avons décidé de remodeliser en utilisant des quadriques est le Soda Hall.

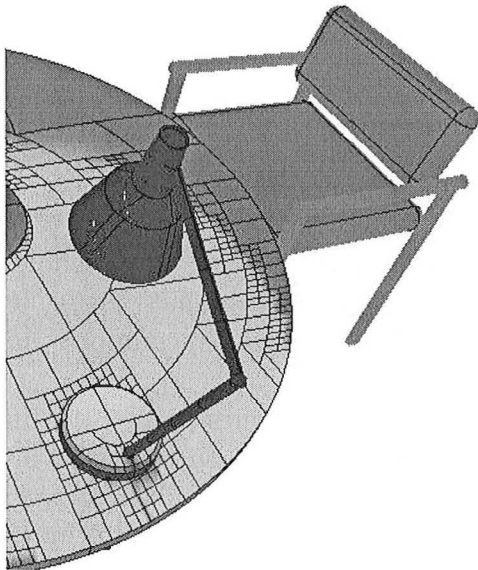


FIG. 4.6 – Exemple de mobilier du Soda Hall remodelisé à base de quadriques.

des primitives quadriques à l'origine puis maillées en triangles pour exporter le modèle en VRML).

Ce choix peut paraître surprenant quand on voit le modèle car il ne semble pas contenir un grand nombre de surfaces courbes. Pourtant, parmi le mobilier, nous avons pu remodeliser un certain nombre d'objets dont les contours présentaient des parties courbes comme par exemple les lampes de bureau, les tables rondes et les fauteuils comme nous l'avons illustré sur la figure 4.6. Le résultat que nous obtenons après cette remodelisation est surprenant, en effet, le nombre de primitives initiales chute de 199 550 à 70 645 soit pratiquement d'un facteur 3. Ceci prouve l'importance de la modélisation et notamment de la possibilité d'utiliser les surfaces dans leur format original (les surfaces que nous avons remodelisées dans le Soda Hall ont très certainement été modélisées avec

4.2.2 Simulations plus efficace

4.2.3 Moins de place en mémoire

Reprenons les résultats numériques relatifs aux nombres de surfaces initiales et comparons les aux nombres de mailles finales. La figure 4.7 montre qu'en plus de diminuer le nombre de primitives initiales (et donc le temps pour effectuer les requêtes de visibilité), l'utilisation des polygones complexes réduit aussi sensiblement le nombre de mailles finales. Notons, que le gain en nombre de mailles finales est sensiblement inférieur au gain en nombre de primitives initiales (de 9 à 49 % contre 37 à 88 % pour les primitives initiales). Nous avons déjà remarqué et expliqué ce phénomène dans le chapitre 3.6.1.

Comme nous l'avons déjà expliqué dans le chapitre 3.6, le surcoût dû à la gestion des primitives complexes est proportionnel au nombre de primitives initiales et donc négligeable devant le nombre de mailles finales. Ainsi, la mémoire nécessaire diminue avec l'utilisation des primitives complexes.

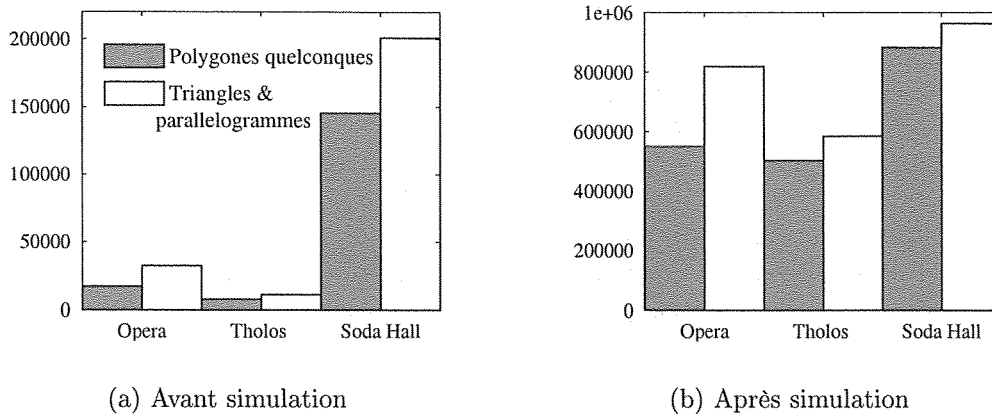


FIG. 4.7 – Nombre de mailles initiales et finales dans 3 de nos scènes de test, avec et sans l'utilisation des polygones quelconques

Le tableau 4.2 reprend ces résultats et les complète par des résultats sur le temps de simulation et sur le nombre d'itérations nécessaires à la convergence de l'algorithme pour les scènes de l'Opéra, du Soda Hall et de la salle de la géométrie. Notons, dès maintenant, que le nombre de surfaces finales diminue encore nettement dans les simulations du Soda Hall lorsque l'on utilise le modèle à base de quadriques. Ceci confirme le résultat précédent, à savoir que la mémoire nécessaire diminue lorsque les primitives géométriques autorisées augmentent en complexité.

Scène	Primitives géométriques	Nombre de primitives initiales	Nombre de mailles finales	Temps de simulation (secondes)	Nombre d'itérations
Opera	Triangles & parallélog.	32 429	819 109	476 s	9 358
	Polygones	17 272	550 159	241 s	6 776
Soda Hall	Triangles & parallélog.	272 450	1 124 585	44 585 s	9 010
	Polygones	199 550	927 428	30 232 s	7 008
	Quadriques	70 645	623 600	12 730 s	6 535
Salle de la géométrie	Quadrics	71 957	827 627	13 411 s	6 642

TAB. 4.2 – Résultats numériques pour les calculs d'illumination de l'Opéra, et du 4^{ème} étage du Soda Hall.

4.2.4 Simulations plus rapides

Le tableau 4.2 montre aussi que le temps de simulation diminue lorsque les primitives augmentent en complexité. Ce gain peut devenir très important si on passe d'un modèle

triangulé à un modèle à base de quadriques : on gagne un facteur 4 dans le temps de simulation du 4^{ème} étage du Soda Hall avec le modèle à base de quadriques. Il est intéressant de remarquer que ce gain ne s'explique pas seulement par la diminution du nombre d'itérations qui ne diminue que de 50 %. Le fait que chaque interaction transporte plus d'énergie (puisque les surfaces sont plus « grandes ») ne suffit donc pas à expliquer la diminution du temps de calcul. Cette diminution résulte, en fait, majoritairement de la diminution du nombre de primitives initiales (ces deux nombres sont relativement bien corrélés dans le tableau) qui influe fortement sur le temps des requêtes de visibilité. On notera, une fois de plus, l'importance du calcul de la visibilité sur l'efficacité de l'algorithme.

Afin de valider ces résultats quelle que soit la convergence demandée, nous avons suivi l'évolution du temps de simulation en fonction du taux de convergence pour les scènes à base de triangles et de parallélogrammes et celles à base de polygones. La figure 4.8 montre cette évolution pour les deux scènes de l'Opéra et du Soda Hall. Elle confirme le résultat que nous avons déjà présenté pour la scène de l'Opéra à savoir que le temps de calcul est de l'ordre de deux fois plus grand avec une scène triangulée qu'avec la même scène modélisée à base de polygones quelconques. Et surtout, ces courbes montrent que ce résultat est général et qu'il ne dépend pas du taux de convergence demandé.

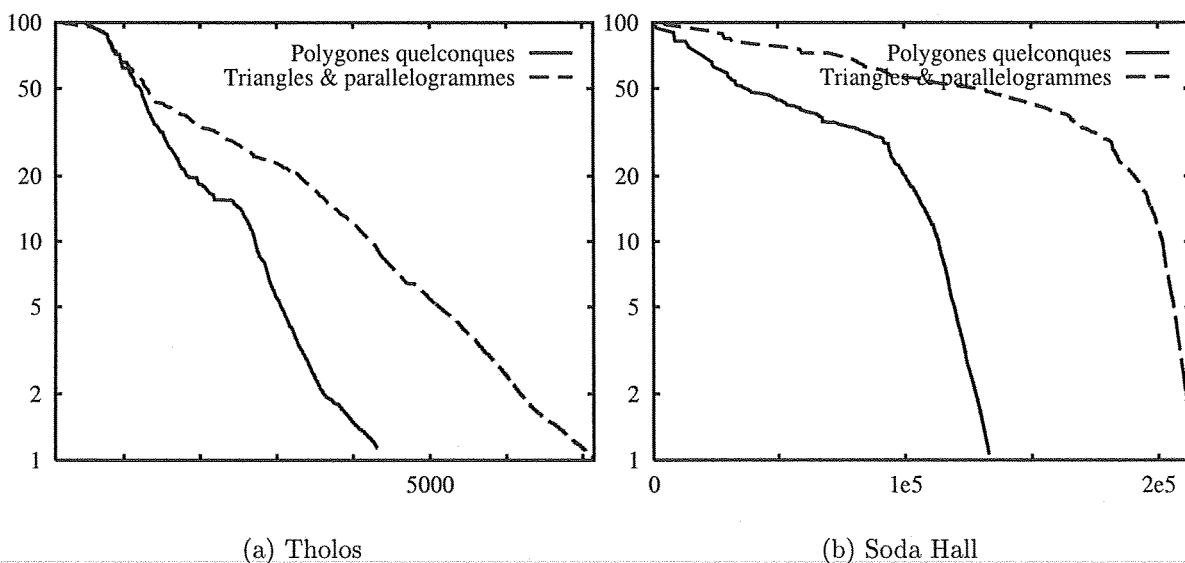


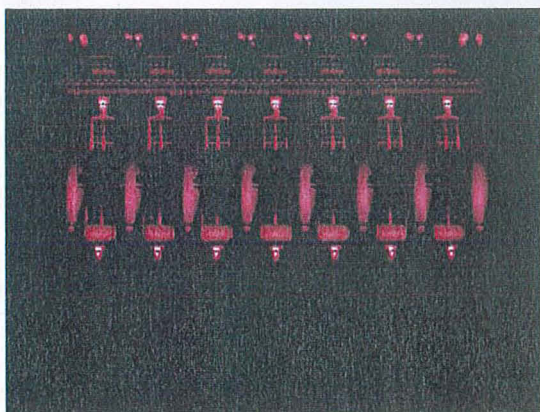
FIG. 4.8 – Taux de convergence (énergie non encore transmise divisée par l'énergie initiale) en fonction du temps de calcul (en s).

4.2.5 Simulations très réalistes

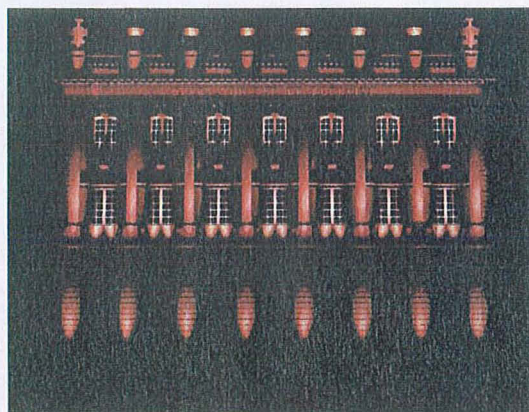
Nous présentons, maintenant, les images obtenus avec nos scènes de test. Ces images sont toutes des prises de vue instantanées faites lors de navigation en temps réel à l'intérieur des environnements que l'on peut visualiser interactivement.

4.2.5.1 Opéra de la place Stanislas

Les simulations de l'Opéra de la place Stanislas présentées sur la figure 4.9 (a) et (b) ont permis à la ville de Nancy de choisir entre différents projets d'éclairage. Le réalisme de la solution que nous avons simulée est illustré par la figure 4.9 (c) et (d) où il est difficile de discerner la photo de la simulation.



(a) Un projet tout en couleur



(b) Le projet retenu par la mairie de Nancy



(c) Une photo de l'Opéra

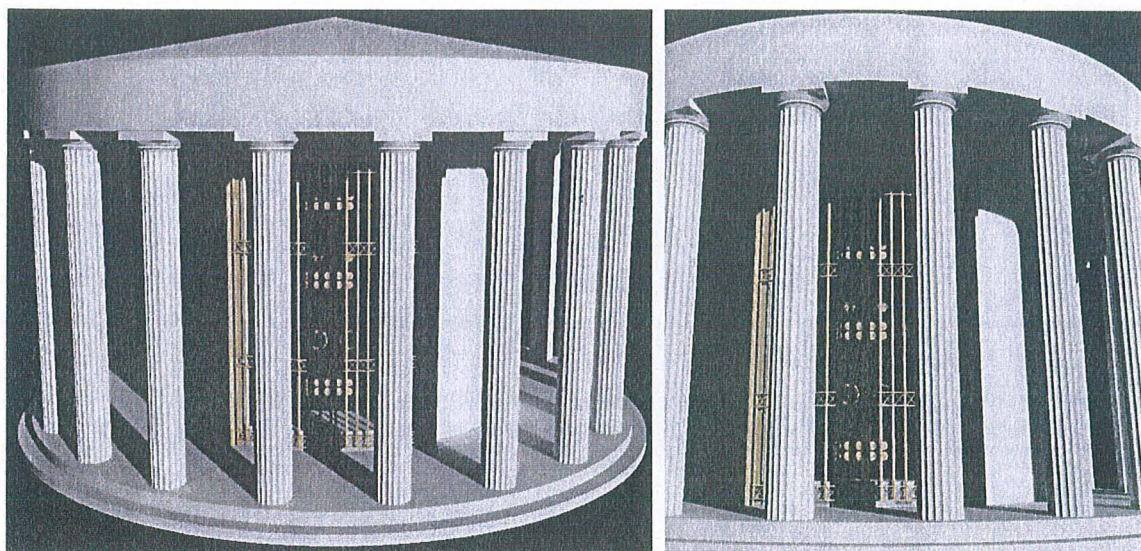


(d) Image simulée réincrûstée.

FIG. 4.9 – *Nos simulations.*

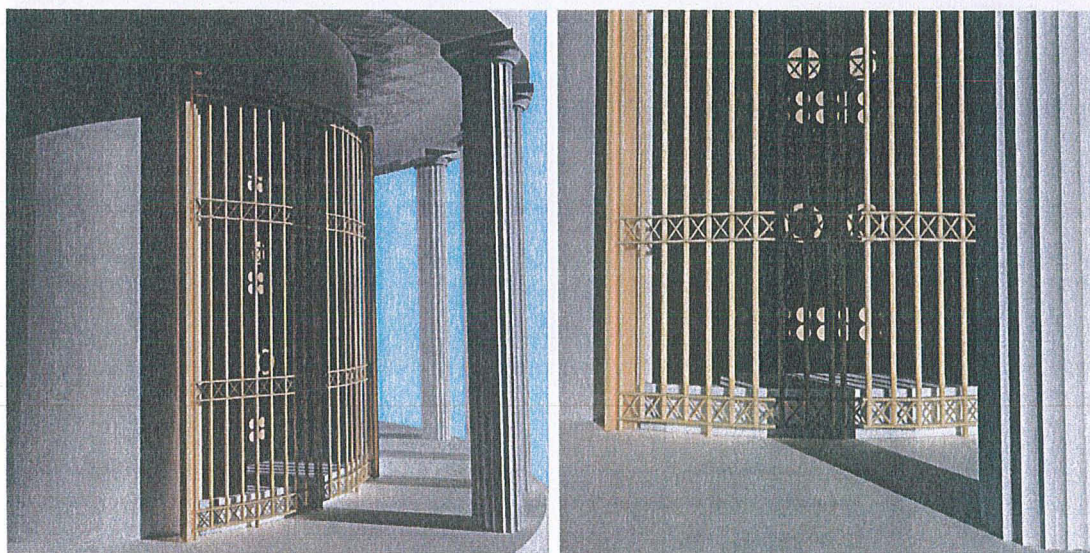
4.2.5.2 Tholos de Delphes

La figure 4.10 nous permet d'apprécier la précision de nos simulations. Cette précision se remarque à deux niveaux : la qualité de l'éclairage des sculptures du plafond par inter-réflexions et la qualité des ombres décelées au travers de la grille.



(a) Vue générale

(b) Vue de face en contre plongée



(c) Vue latérale avec réflexions sur les sculptures du plafond

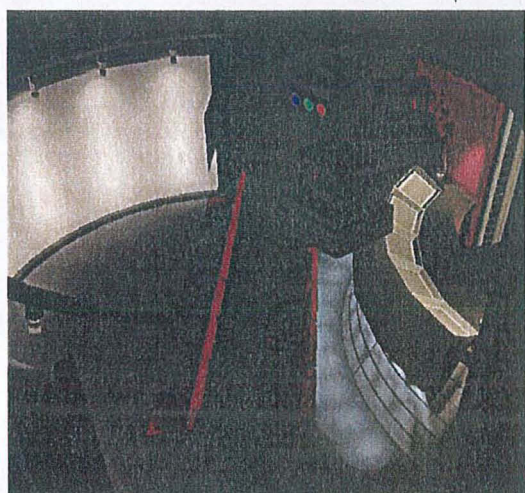
(d) Zoom de face avec une grande précision des ombres portées

FIG. 4.10 – Simulations d'un éclairage naturel de la Tholos

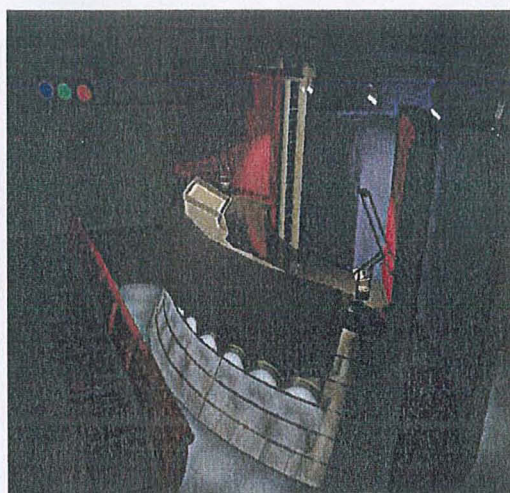
4.2.5.3 Reality Center

La simulation du Reality Center de Cortaillod présentée sur la figure 4.11 avait pour objectif de présenter à des industriels les techniques de radiosité permettant de réaliser des simulations réalistes. Ces simulations peuvent servir de support au prototypage virtuel, à l'éclairage, au marketing, à l'urbanisme, à l'architecture et à bien d'autres domaines encore.

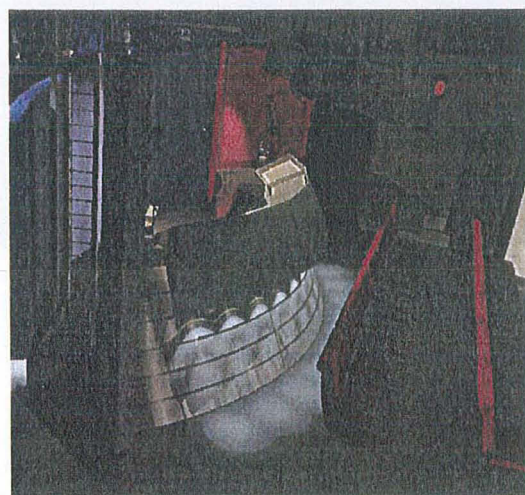
Cet objectif a été rempli lors des Monster Weeks qui ont eu lieu du 15 mai 2000 au 16 juin 2000 et durant lesquels la simulation a été présentée à un grand nombre d'industriels partenaires de SGI dans le Reality Center de Cortaillod.



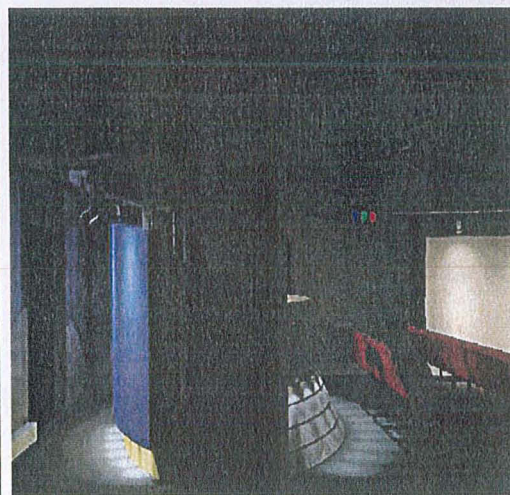
(a) Vue générale



(b) Le pupitre de commande



(c) Effet des sources de lumières et des réflexions



(d) Vue latérale

FIG. 4.11 – *Simulations du Reality Center de Cortaillod*

4.2.5.4 4^{ème} étage du Soda Hall

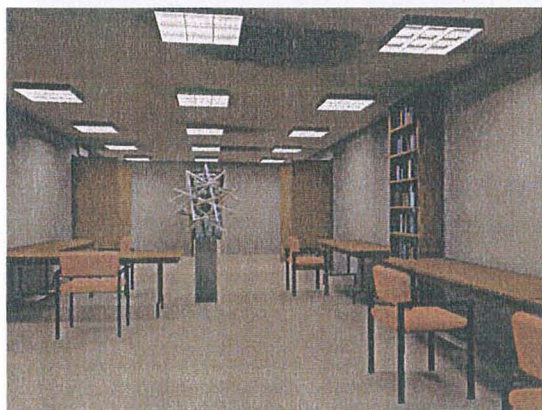
La scène du 4^{ème} étage du Soda Hall n'est pas une scène industrielle. Elle a été spécialement conçue pour des simulations numériques et son intérêt réside dans la taille du modèle. Les images de la figure 4.12 montrent la précision avec laquelle nous avons réussi à simuler la totalité de l'étage avec son mobilier. Nous n'avons utilisé aucune connaissance a priori de la scène pour cette simulation qui a été faite directement à partir du modèle récupéré sur le WEB.



(a) Vue générale



(b) Vue extérieure



(c) Vue intérieur



(d) Vue intérieure d'une pièce

FIG. 4.12 – Simulation du 4^{ème} étage du Soda Hall

4.2.5.5 Salle de la géométrie

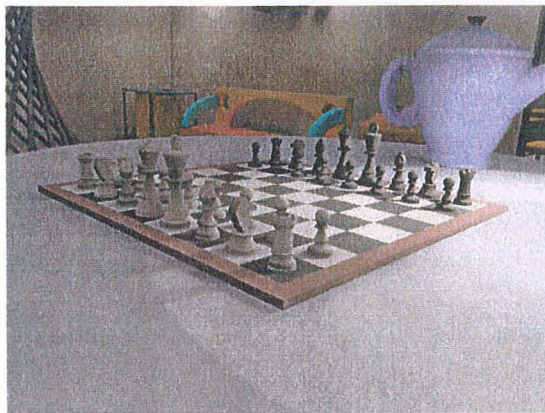
Les images de la salle de la géométrie (figure 4.13) montrent que la qualité de nos simulations n'est pas dépendante de la complexité du modèle. Ainsi, la salle remplie de surfaces quadriques a été simulée avec la même précision (et la même efficacité d'après le tableau 4.2) que le reste de l'étage. De plus, les surfaces quadriques n'ont en rien perturbé la simulation du reste de l'étage.



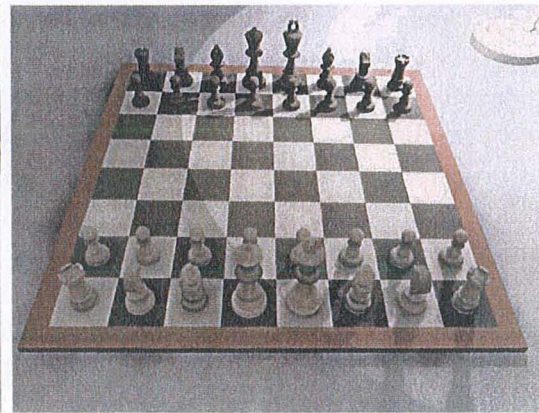
(a) Vue générale



(b) Vue rapprochée



(c) Un zoom



(d) Le jeu d'échecs

FIG. 4.13 – Simulation dans le Soda Hall avec la salle de la géométrie.

4.3 Autres solutions

Les résultats que nous venons de présenter prouvent la grande efficacité de notre algorithme pour simuler l'illumination globale dans des scènes du monde réel. D'autres solutions ont été proposées pour faire ces simulations et nous les rappelons et nous les analysons dans cette section.

Nous commençons par voir les implantations parallèles de l'algorithme de radiosité qui ont été proposées dans la littérature. Ensuite nous parlons du *clustering* qui est une généralisation de l'algorithme hiérarchique. Finalement, nous étudions le *face cluster*, une approche développée récemment pour gérer des scènes de grande taille mais présentant certaines caractéristiques géométriques très précises.

4.3.1 Algorithme parallèles

La plupart des implantations parallèles de la méthode de radiosité [Cha 98] sont basées sur l'algorithme de radiosité classique, qui a une complexité quadratique en temps et en mémoire, comme nous l'avons vu précédemment. Ces solutions ne s'appliquent pas à des scènes du monde réel. Néanmoins, il existe quelques travaux consacrés à la radiosité hiérarchique et qui ont eu pour but de traiter des scènes de taille industrielle.

Dans [Fun 96], Thomas A. FUNKHOUSER décrit un système pour effectuer des calculs de radiosité hiérarchique pour des modèles polygonaux de très grande taille en utilisant plusieurs processus concurrents. La solution proposée est une version parallèle des techniques de partitionnement et d'ordonnancement proposées par Set TELLER *et al.* [Tel 94], qui sont fortement dépendantes de la géométrie axiale et dense en occlusions du modèle (une surface donnée interagit avec un faible nombre d'autres surfaces).

Cette approche est étendue à des environnements à géométrie non-axiale par Daniel MENEVEAUX *et al.* [Men 98], en gardant des idées très similaires. Cependant, l'hypothèse de géométrie dense en occlusions y demeure une contrainte forte.

Notons que les techniques de partitionnement et d'ordonnancement dépendent fortement de la géométrie du modèle considéré : il faut que les surfaces initiales puissent facilement être organisées en *clusters* lors de la création de la scène, et que des informations de visibilité pré-calculées entre les *clusters* soient utilisées pour créer des groupes séparés de *clusters* avec peu d'interactions mutuelles. Cette caractéristique est essentielle pour une exécution efficace des calculs de radiosité avec ces méthodes.

Ainsi, les précédentes versions parallèles d'algorithmes de radiosité hiérarchique ne peuvent effectuer des calculs de radiosité sur des scènes de grande taille sans aucune hypothèse restrictive sur la géométrie de la scène. L'algorithme de radiosité parallèle proposé par Xavier CAVIN dans sa thèse [Cav 00] pour bénéficier de l'architecture à mémoire partagée des ORIGIN2000 est la solution que nous avons utilisée pour l'associer à nos optimisations. Cette solution ne fait aucune hypothèse sur la géométrie des données initiales et permet de profiter au maximum de la puissance des algorithmes décrits dans ce mémoire. Grâce à l'efficacité que nous avons obtenue en conjuguant nos algorithmes nous pouvons faire des simulations dans des scènes industrielles et nous utilisons cette nouvelle technologie issue de nos travaux de recherche pour effectuer un transfert industrielle de Candela.

4.3.2 Clustering

Nous avons déjà fait remarquer que l'algorithme de radiosité à base d'ondelettes permettait de réduire la complexité de l'algorithme de n^2 à $O(k^2 + n)$, où k est le nombre de primitives géométriques initiales et n est le nombre de mailles finales. Cette amélioration

de l'algorithme résulte de la représentation hiérarchique des ondelettes. Malheureusement, elle ne fait pas disparaître le terme en k^2 qui peut devenir très pénalisant pour l'algorithme lorsque celui-ci est utilisé pour faire des simulations sur des scènes de très grande taille.

Afin de pouvoir s'attaquer à des scènes de taille industrielle, des solutions ont été proposées pour faire disparaître ce terme en k^2 . Ainsi, les approches par regroupement de surfaces ou *clustering* proposées par Brian SMITS *et al.* dans [Smi 94] puis par François SILLION dans [Sil 95a] permettent de généraliser la notion hiérarchique à tout l'environnement en regroupant les surfaces pour diminuer le nombre d'interactions. Grâce à ces méthodes, la complexité de l'algorithme est ramenée à $O(k \log k + n)$ ce qui améliore nettement l'efficacité de l'algorithme et le rend utilisable pour des scènes de beaucoup plus grande taille. Avec ce gain d'efficacité, la méthode de *clustering* a même pu être adaptée au problème de la radiance par François SILLION *et al.* [Sil 95b].

Pour arriver à un tel résultat, il faut créer une hiérarchie qui englobe la totalité de l'environnement puis qui se subdivise en utilisant des stratégies prenant en compte la proximité des surfaces les unes par rapport aux autres telles que la hiérarchie des volumes englobants proposée par J. GOLDSMITH et J. SALMON dans [Gol 87]. L'ensemble de toutes les primitives géométriques forme donc le premier *cluster* de niveau 0 qui se subdivise en niveaux hiérarchiques où les primitives sont regroupées en sous-*clusters* et ceci de façon récursive. Quand un sous-*cluster* ne contient plus qu'une seule primitive géométrique, on gère la hiérarchie sous cette dernière exactement comme on le faisait dans l'algorithme classique de radiosité hiérarchique. L'algorithme revient alors à faire interagir le *cluster* de niveau 0 avec lui-même (comme pour les primitives courbes concaves qui doivent *shooter* leur énergie sur elles-mêmes) autant de fois que cela est nécessaire pour atteindre la convergence désirée.

Les approches à base de *clustering* semblent donc être très prometteuses pour augmenter encore la complexité des scènes que l'on peut traiter. Or, nous avons expliqué dans le chapitre 3 que l'énorme avantage du maillage virtuel est qu'il ne fait aucune supposition sur l'algorithme de radiosité et qu'il peut être implanté dans n'importe quelle solution existante. Il est donc tout à fait possible d'envisager d'intégrer le maillage virtuel dans un algorithme de radiosité à base de *clustering*. Une telle implantation permettrait, par ailleurs, de résoudre le problème indiqué par Jean-Marc HAZENFRATZ *et al.* dans leur récente étude [Has 99], à savoir que l'utilisation des méthodes de *clustering* sur des objets qui ont été préalablement maillés sans conserver des connaissances sur leur topologie s'avère particulièrement inefficace.

4.3.3 Face cluster

Récemment, Andrew WILLMOTT *et al.* ont proposé une nouvelle approche pour traiter des scènes de taille extrêmement grande mais présentant des caractéristiques bien particulières. Leur solution, le *face cluster*, présentée dans [Wil 99], consiste à créer une hiérarchie au dessus des primitives géométriques de base (comme pour le *clustering*) en prenant en compte leur coplanarité. Cette très intéressante approche leur a permis de simuler l'illumination globale dans une scène contenant 2,7 millions de primitives initiales en 2 minutes seulement. Ce résultat peut sembler surprenant quand on sait que les autres

approches peinent à traiter des scènes contenant plusieurs centaines de milliers de surfaces et que les temps de calcul se compte en heures. Cependant, ce résultat s'explique par les caractéristiques de la scène simulée par Andrew WILLMOTT *et al.* lors de leurs expérimentations. En effet, leur scène représente une pièce relativement simple contenant une table et une statue. La particularité de cette scène est que le sol contient à lui seul plus d'un million de triangles pour rendre un aspect rugueux. De même, la statue contient énormément de détails et donc un très grand nombre de surfaces. Cette scène est donc parfaitement adaptée au *face cluster* qui crée une scène de quelques centaines de *clusters* initiaux avec chacun une lourde hiérarchie. Les calculs de simulation sont ensuite faits à des niveaux relativement élevés et ce n'est qu'après la simulation que l'énergie collectée sur chaque *cluster* est redescendue aux niveaux les plus bas sur les surfaces initiales.

Cette méthode ne donnerait aucun résultat convaincant sur la scène du Soda Hall ou sur celle du Reality Center où les surfaces ne peuvent que très rarement être regroupées suivant leur coplanarité. De plus, la création de la hiérarchie au dessus des surfaces suppose que l'on sait à l'avance quelles surfaces vont être regroupées pour former un *cluster* ce qui est le cas pour la scène de la pièce mais qui est impossible dans toutes nos scènes de test.

Enfin, la solution du *face cluster* souffre du fait que la hiérarchie résulte d'une construction purement géométrique. En effet, aucune relation mathématique ne relie les niveaux les uns aux autres et il n'est donc pas possible d'appliquer un algorithme de radiosité à base d'ondelettes avec le *face cluster* car on ne peut pas calculer de coefficients de *push-pull*.

La solution du *face cluster* nous apparaît donc comme une solution très efficace et très intéressante mais seulement pour un certain type de scènes présentant des caractéristiques bien précises. Il peut notamment s'avérer particulièrement efficace sur des scènes générées par des relevés au laser.

Conclusion

« Dans le noir toutes les couleurs s'accordent »

Francis BACON, *Essais*

Nous avons présenté dans ce travail le maillage virtuel qui est une abstraction géométrique pour la radiosité à base d'ondelettes. Ce travail nous a amené à étudier précisément l'influence des bases de fonctions ondelettes sur la solution de radiosité avant de nous intéresser plus particulièrement à l'aspect géométrique.

Notre travail est parti de la constatation que les recherches en radiosité se sont faites de façon empirique. Ainsi, les fondements théoriques basés sur des formalismes mathématiques se sont greffés sur des algorithmes existants. Par exemple, la formalisation de l'utilisation de bases de fonctions de degré élevé et celle de la représentation hiérarchique se sont faites en même temps grâce aux bases d'ondelettes qui n'ont été introduites que bien après que ces deux théories aient été développées. On remarque, cependant, que la formalisation mathématique permet généralement une meilleure compréhension des algorithmes et qu'elle donne des outils pour optimiser les implantations et l'utilisation des théories sous-jacentes.

L'efficacité des algorithmes découle en grande partie de la compréhension des processus mathématiques qui les animent. Pourtant, nous constatons que la grande majorité des implantations n'utilisent que très partiellement la totalité de la description formelle des processus qui interviennent dans la résolution. Par exemple, bien que les ondelettes formalisent les approches hiérarchiques, les implantations de ces dernières restent majoritairement calquées sur la méthode originale de radiosité hiérarchique qui ne permet pas l'utilisation de fonctions d'ordre élevé. Par ailleurs, les rares implantations qui ont cherché à prendre en compte les nouvelles théories l'ont généralement fait en ajoutant à un algorithme existant des morceaux de code pour compléter les programmes sans repenser réellement la structure des programmes pour qu'elle s'adapte pleinement à ces nouvelles théories.

Dans un souci de formalisation et de généralisation de l'algorithme de radiosité, nous nous sommes intéressés à deux points centraux de l'algorithme. Nous avons tout d'abord repris la théorie des ondelettes pour l'intégrer dans un nouvel algorithme spécialement conçu autour de la notion fonctionnelle. Ceci nous a permis de bien séparer la représentation fonctionnelle de la représentation géométrique. Grâce à cela, nous avons pu nous intéresser plus précisément à la représentation géométrique. Nous avons notamment défini un nouveau formalisme : le maillage virtuel. Ce formalisme permet de généraliser la

notion de primitives géométriques jouant le rôle de support de fonctions, ce qui donne la possibilité d'utiliser des primitives beaucoup plus complexes que des triangles et des parallélogrammes pour peu qu'on sache proprement les manipuler de façon théorique.

Dans ce mémoire, nous avons présenté les résultats de nos recherches pour généraliser et formaliser l'algorithme de radiosit .

Ce m moire a commenc  par l' tude th orique de la propagation de l' nergie lumineuse dans un environnement donn . Nous sommes parti de consid rations physiques pour d finir l' quation de radiance qui est une g n ralisation de l' quation de radiosit . Nous nous sommes notamment attach    bien d finir les hypoth ses faites, pour conna tre exactement le domaine de validit  de notre approche. Puis, le formalisme math matique nous a permis d' tablir le syst me d' quations   r soudre pour trouver la solution   l' quation pr c dente. Pour finir, nous avons  tudi  les aspects algorithmiques des notions d finies pendant les deux  tapes pr c dentes. Ces aspects ont d taill  la provenance des donn es initiales auxquelles il faut adapter l'algorithme de r solution, ainsi que les relations qui unissent les notions impliqu es dans le processus de r solution. Cette  tude algorithmique a aussi affin  certains probl mes, comme l'utilisation de base de fonctions pour projeter l' quation et la repr sentation de la g om trie.

Puis, nous avons pr sent  notre implantation de l'algorithme de radiosit  avec les sp cificit  associ es   l'utilisation des bases de fonctions ondelettes. Nous avons, pour cela, red fini un certain nombre de points cl  de l'algorithme de radiosit  pour qu'il int grent parfaitement la notion fonctionnelle. Nous avons notamment repens  l'oracle de subdivision, le calcul de la visibilit  et le processus it ratif de r solution. Ce dernier point nous a permis d' viter un des  cueils classiques des algorithmes   base d'ondelettes,   savoir le stockage des liens d'interaction en m moire ce qui rend les fonctions d'ordre  lev  inutilisables pour des simulations sur des sc nes de grande taille. Une fois l'algorithme repens  pour s'adapter   la notion fonctionnelle, nous avons pr sent  les sp cificit  des bases d'ondelettes et nous avons notamment d velopp  de nouvelles bases pr vues pour la m thode de r solution par collocation de points. Cette m thode est tr s efficace pour faire des simulations visuellement parfaites. Nous avons termin  notre pr sentation par une  tude comparative des diff rentes bases. Cette  tude nous a permis de montrer que pour chaque base d'ondelettes il existe une zone de comp tence dans laquelle la base concern e d passe toutes les autres en efficacit    tous points de vue : qualit , rapidit , taille. Cette zone de comp tence correspond   des solutions de plus en plus pr cises lorsque l'ordre des fonctions augmente. Nous avons ainsi pu d duire que les fonctions lin aires  taient les mieux adapt es pour faire des simulations dont l'objectif est la visualisation alors qu'il valait mieux utiliser des fonctions d'ordre plus  lev  pour des simulations destin es   faire des mesures physiques.

Ensuite, nous avons d taill  les r sultats de nos recherches sur la repr sentation g om trique. Nous avons d crit le principe d'un nouveau formalisme : le maillage virtuel. Le maillage virtuel permet de g n raliser la s paration entre les primitives g om triques et les supports de la fonction de radiosit . Pour cela, il d finit une abstraction de la g om trie   base de carr s normalis s parfaitement adapt e   la repr sentation fonctionnelle. Cette abstraction permet de revenir   la g om trie initiale chaque fois que cela est n cessaire tout en conservant les propri t s  nerg tiques des fonctions que l'on projette dessus. Nous

avons décrit le processus général pour utiliser le maillage virtuel dans un algorithme de radiosité. Comme cette description est indépendante de l'algorithme choisi nous avons montré plus précisément son implantation dans notre algorithme de radiosité à base d'ondelettes. Le maillage virtuel est un concept général qui permet théoriquement de gérer n'importe quelle surface paramétrique. Pour confirmer la puissance de ce concept, nous avons implanté le maillage virtuel pour gérer deux grands types de surfaces paramétriques : les surfaces planes à bords quelconques (dont l'ensemble très important des polygones) et les surfaces quadriques. Enfin, nous avons donné les résultats expérimentaux que nous avons obtenus avec ces nouvelles primitives géométriques. Nous avons notamment montré que le maillage virtuel permettait de réduire sensiblement le nombre de primitives initiales tout en augmentant la précision des modèles géométriques. Cette réduction nous a permis de diminuer fortement le temps des simulations car les requêtes de visibilité se font, alors, dans un environnement de moins grande taille. De plus le maillage virtuel permet de réduire par la même occasion la place nécessaire en mémoire. Enfin, nous avons montré que la complexité des surfaces utilisées grâce au maillage virtuel rendait encore plus important le fort pouvoir d'approximation des fonctions de degré élevé. En effet, elles seules peuvent facilement rendre compte des variations continues de la fonction sur des surfaces dont la courbure varie elle aussi de façon continue. Ainsi, les surfaces paramétriques et notamment les quadriques sont apparues comme une voie tout à fait prometteuse pour faire des modélisations de grande précision avec un minimum de primitives.

Nous avons fini notre mémoire en rassemblant dans une même partie tous les résultats des tests que nous avons effectués sur des scènes issues d'applications du monde réel. Ces scènes qui sont beaucoup plus complexes que les scènes de tests classiques ne peuvent pas être utilisées pour faire des expérimentations exhaustives. Ces résultats nous ont permis de confirmer par la pratique la très grande efficacité théorique des algorithmes décrits dans ce mémoire. Enfin, nous avons présenté quelques autres approches qui ont été proposées pour gérer des scènes du monde réel dans un algorithme de radiosité.

Nous n'avons pas abordé dans ce mémoire la visualisation des résultats. Or un des intérêts des méthodes de radiosité est leur indépendance du point de vue, et donc la possibilité de les visualiser en temps réel. Ce problème fait l'objet de recherches actuelles dans l'équipe ISA. Nous essayons notamment de prendre en compte de nouvelles primitives géométriques, car les cartes graphiques sont limitées à l'affichage de triangles et il faut donc les développer de nouveaux algorithmes de bas niveau pour afficher des surfaces plus complexes comme des surfaces courbes.

Poursuite de la recherche

Les résultats que nous avons obtenus et présentés dans ce mémoire incitent à continuer les recherches sur la résolution de l'équation de radiosité. Nous souhaitons notamment améliorer l'utilisation des bases d'ondelettes en développant un nouvel algorithme capable de sélectionner dynamiquement, au fil des itérations, la meilleure base d'ondelettes pour effectuer le calcul d'une interaction donnée. Nous voulons aussi étendre l'utilisation du maillage virtuel à de nouveaux types de surfaces tel que les NURBS et l'utiliser pour améliorer l'efficacité des subdivisions géométriques dans les algorithmes de *clustering*.

Algorithme à base d'ondelettes dynamiques

Pour améliorer l'utilisation des bases de fonctions ondelettes nous suggérons de développer un algorithme utilisant plusieurs bases d'ondelettes au cours d'une même simulation. Nous pourrions, ainsi, développer un algorithme qui choisit dynamiquement au cours des calculs la meilleure base d'ondelettes pour une interaction donnée. Cet algorithme utiliserait la base de HAAR pour faire les calculs d'interaction ne nécessitant que peu de précision et il les ondelettes d'ordre plus élevé, linéaires, quadratiques, cubiques ou même quartiques, pour les interactions de plus grande précision. La difficulté de cette algorithme réside dans l'implantation des critères de sélection permettant de choisir la base d'ondelettes adaptée. Le choix et le réglage de ces critères demandera une analyse approfondie pour déterminer où et quand l'algorithme a besoin de précision. Il faudra commencer par définir ce qu'on entend réellement par précision et voir comment on peut détecter au cours de l'algorithme les erreurs qui influent effectivement sur le résultat final. L'intérêt d'une telle approche et qu'elle permettrait de concentrer les calculs là où il y en a le plus besoin et de rester très approximatif (donc de faire très peu de calculs) là où la précision n'est pas réellement nécessaire. Le problème majeur pour développer un tel algorithme est de trouver un oracle permettant de décider automatiquement l'ordre des fonctions à utiliser.

Nouvelles surfaces paramétriques.

Pour étendre l'utilisation du maillage virtuel, nous suggérons de l'utiliser sur de nouvelles surfaces paramétriques telles que les surfaces algébriques de degré élevé (notamment les B-Splines ou le NURBS). En effet, les surfaces algébriques de degré élevé sont de plus en plus utilisées pour la modélisation industrielle et permettent une manipulation facile et très pratique aboutissant à des modèles très précis d'objets extrêmement compliqués. Nous pensons, cependant, que les fonctions de projection à Jacobien constant pour ces surfaces risquent de devenir rapidement trop compliquées pour être utilisables dans les calculs de radiosité.

Il sera alors, peut-être, plus efficace de faire des approximations de ces surfaces au moyen de quadriques — des recherches ont déjà été faites dans ce sens par B. LACOLLE et N. SZAFRAN dans [Lac 99]. Ces recherches visent à généraliser les techniques de triangulation en utilisant des surfaces plus complexes pour permettre à ces approximations d'intégrer des notions de continuité aussi bien pour la surface que pour ses dérivées.

Clustering

Nous avons déjà fait remarquer que le maillage virtuel était indépendant de l'algorithme de radiosité utilisé, et nous pensons qu'il serait intéressant de l'essayer avec les algorithmes ayant prouvé leur efficacité. Nous voulons notamment étendre notre algorithme pour qu'il intègre la notion de *cluster* ou inversement intégrer le maillage virtuel dans un algorithme à base de *cluster* déjà existant. Dans les deux cas un certain nombre de problèmes apparaîtront :

1. Comment regrouper les surfaces sachant que la géométrie de ces dernières sera de plus en plus complexes? Il faudra sans doute utiliser les boîtes englobantes pour gérer les nouvelles surfaces.

2. Comment gérer les nouveaux calculs de visibilité et les échanges d'énergie entre *clusters* alors même que la géométrie contenue dans ces *clusters* sera des plus complexe?
3. Dans le cas de l'algorithme à base d'ondelettes, comment calculer les coefficients de *push-pull* pour qu'il s'adapte à la totalité de la hiérarchie?

Maillage de discontinuité

Nous travaillons actuellement à l'utilisation du maillage virtuel en collaboration avec un algorithme de maillage de discontinuité. Nous espérons ainsi bénéficier de la précision de ce dernier et de l'efficacité du maillage virtuel pour aller un cran plus loin dans la précision et le réalisme de nos simulations. L'idée de l'algorithme que nous développons actuellement est de considérer le maillage de discontinuité comme délimitant des surfaces à bords complexes. La figure 14 montre, par exemple, que l'ombre du tétraèdre sur le parallélogramme crée deux polygones qui peuvent être gérés séparément grâce au maillage virtuel. Cette solution permet d'intégrer exactement les limites d'ombres dans la solution

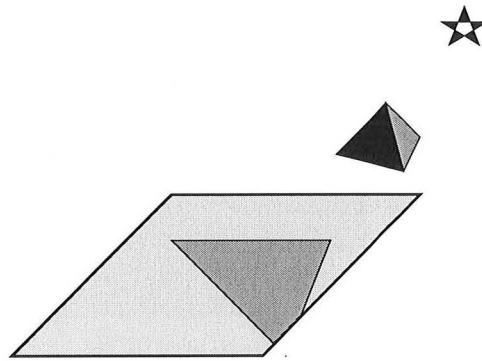


FIG. 14 – *Le maillage de discontinuité.*

finale sans augmenter trop fortement le nombre de primitives.

Annexe A

Intégrale elliptique.

Une *intégrale elliptique générale* est définie de la façon suivante :

$$\int R(x,y),$$

avec R une fonction rationnelle en x et y (y^2 est un polynôme cubique ou quartique en x dont tous les zéros sont disjoints). Le nom « elliptique » vient du fait que la longueur d'un arc d'ellipse s'exprime au moyen d'une intégrale elliptique. Ces intégrales font leur apparition dès le VII^{ème} siècle en astronomie pour calculer la trajectoire des planètes.

A.1 Intégrales elliptiques de première espèce

A.1.1 Définitions

Considérons m un paramètre. Soit φ un angle tel que $0 \leq \varphi \leq \pi/2$. On définit alors les *intégrales elliptiques de première espèce* comme les fonctions intégrales dont l'équation se ramène à :

$$F(\varphi, m) = \int_0^\varphi \sqrt{1 - m \sin^2 \theta}^{-1} d\theta = \int_0^{\sin \varphi} \frac{dt}{\sqrt{(1-t^2)(1-mt^2)}}.$$

Ces fonctions ont les propriétés suivantes :

- Valeurs particulières.

$$F(\varphi, 0) = \varphi, \quad F(i\varphi, 0) = i\varphi, \\ F(\varphi, 1) = \ln \tan \left(\frac{\pi}{4} + \frac{\varphi}{2} \right), \quad F(i\varphi, 1) = i \arctan (\sinh \varphi).$$

- Amplitude négative. $F(-\varphi, m) = -F(\varphi, m)$.

- Paramètre négatif. Si m est positif :

$$F(\varphi, -m) = \frac{1}{\sqrt{1+m}} \left(F\left(\frac{\pi}{2}, \frac{m}{1+m}\right) - F\left(\frac{\pi}{2} - \varphi, \frac{m}{1+m}\right) \right).$$

– Paramètre réciproque. Soit θ tel que $\sin \theta = \sqrt{m} \sin \varphi$. Alors :

$$F(\varphi, m) = \frac{1}{\sqrt{m}} F\left(\theta, \frac{1}{m}\right).$$

– Amplitude imaginaire. If $\tan \theta = \sinh \varphi$:

$$F(i\varphi, m) = iF(\theta, 1 - m).$$

Montrons par exemple comment on prouve cette dernière propriété

$$\begin{aligned} F(i\varphi, m) &= \int_0^{i\varphi} \frac{d\theta}{\sqrt{1 - m \sin^2 \theta}}, && \text{(changement } \theta = iv) \\ &= i \int_0^\varphi \frac{dv}{\sqrt{1 + m \sinh^2 v}}, && \text{(changement } \sinh v = \tan x \Rightarrow dv = \frac{dx}{\cos x}) \\ &= i \int_0^\theta \frac{dx}{\sqrt{1 - (1 - m) \sin^2 x}}, \\ &= iF(\theta, 1 - m), \end{aligned}$$

où $\tan \theta = \sinh \varphi$.

A.1.2 AGM et transformations de LANDEN

Les transformations de LANDEN partent du résultat suivant. Si $x \sin \lambda = \sin 2\mu - \lambda$, alors :

$$F(\lambda, x^2) = \frac{2}{x+1} F\left(\mu, \frac{4x}{(x+1)^2}\right).$$

(ceci est obtenu par simple changement de variable.)

On peut maintenant définir la *transformation de LANDEN ascendante* par la formule de récurrence suivante :

$$\begin{cases} (1 + \sin \alpha_n)(1 + \cos \alpha_{n+1}) = 2, \\ \sin(2\varphi_{n+1} - \varphi_n) = \sin \alpha_n \sin \varphi_n, \end{cases}$$

avec $\varphi = \varphi_0$ et $\alpha = \alpha_0$. Le point intéressant de cette transformation est que : $\alpha_{n+1} > \alpha_n$ et $\varphi_{n+1} < \varphi_n$ et que $\alpha_n \rightarrow \frac{\pi}{2}$ quand $n \rightarrow \infty$. On déduit des résultats A.1.2 et A.1.2 que :

$$F(\varphi, \sin^2 \alpha) = \frac{2}{1 + \sin \alpha} F(\varphi_1, \sin^2 \alpha_1).$$

De façon plus générale :

$$F(\varphi, \sin^2 \alpha) = 2^n \prod_{s=0}^{n-1} (1 + \sin \alpha_s)^{-1} F(\varphi_n, \sin^2 \alpha_n) = \prod_{s=1}^n (1 + \cos \alpha_s) F(\varphi_n, \sin^2 \alpha_n).$$

Par récurrence, on peut partir d'un angle donné pour se rapprocher suffisamment de $\frac{\pi}{2}$. Alors :

$$\begin{aligned} F(\varphi, \sin^2 \alpha) &\simeq \prod_{s=1}^N (1 + \cos \alpha_s) F(\varphi_N, 1), \\ &\simeq \prod_{s=1}^N (1 + \cos \alpha_s) \ln \tan \left(\frac{\pi}{4} + \frac{\varphi_N}{2} \right). \end{aligned}$$

La transformation de LANDEN descendante est presque la même :

$$\begin{cases} (1 + \sin \alpha_{n+1})(1 + \cos \alpha_n) = 2, \\ \tan(\varphi_{n+1} - \varphi_n) = \cos \alpha_n \tan \varphi_n. \end{cases}$$

Cette fois-ci, $\alpha_{n+1} < \alpha_n$ et $\varphi_{n+1} > \varphi_n$, et $\alpha_n \rightarrow 0$ quand $n \rightarrow \infty$. On a :

$$F(\varphi, \sin^2 \alpha) = 2^{-n} \prod_{s=1}^n (1 + \sin \alpha_s) F(\varphi_n, \sin^2 \alpha_n).$$

Donc, quand $\alpha_n = 0$ pour une certaine précision :

$$F(\varphi, \sin^2 \alpha) = 2^{-N} \varphi_N \prod_{s=1}^N (1 + \sin \alpha_s).$$

Plutôt que de calculer α_n à chaque étape, il est plus courant et plus intéressant d'utiliser le processus AGM (Arithmetic-Geometric Mean) de GAUSS. L'idée est de partir du triplet (a_0, b_0, c_0) et de calculer récursivement (a_n, b_n, c_n) de la façon suivante :

$$a_n = \frac{1}{2}(a_{n-1} + b_{n-1}), \quad b_n = \sqrt{a_{n-1}b_{n-1}}, \quad c_n = \frac{1}{2}(a_{n-1} - b_{n-1}).$$

On arrête le processus à la $N^{\text{ième}}$ étape quand $c_N = 0$ à la précision des nombres recherchés près. Le point intéressant de cette approche est qu'elle peut joliment être associée aux transformations de LANDEN. En effet, si on considère la transformation de LANDEN descendante, soit $(a_0, b_0, c_0) = (1, \cos \alpha, \sin \alpha)$. Alors :

$$a_n = \prod_{s=1}^n (1 + \sin \alpha_s)^{-1}, \quad \frac{b_n}{a_n} = \cos \alpha_n, \quad \frac{c_n}{a_n} = \sin \alpha_n.$$

Il est alors clair que :

$$F(\varphi, \sin^2 \alpha) \simeq \frac{\varphi_N}{2^N a_N}.$$

En utilisant la transformation de LANDEN ascendante, et en prenant $(a'_0, b'_0, c'_0) = (1, \sin \alpha, \cos \alpha)$, on a alors :

$$a'_n = \frac{1}{2^n} \prod_{s=0}^{n-1} (1 + \sin \alpha_s), \quad \frac{b'_n}{a'_n} = \sin \alpha_n, \quad \frac{c'_n}{a'_n} = \cos \alpha_n.$$

A.1.3 Inverses des intégrales elliptiques de première espèce

Soit $u = F(\varphi, m)$ et m , on définit les *fonctions elliptiques de JACOBI* (ou fonction elliptiques inverses) de la façon suivante :

$$\begin{aligned}\operatorname{sn}(u|m) &= \sin \varphi, \\ \operatorname{cn}(u|m) &= \cos \varphi, \\ \operatorname{dn}(u|m) &= \sqrt{1 - m \sin^2 \varphi}.\end{aligned}$$

Ces fonctions ont beaucoup de propriétés intéressantes :

- Valeurs particulières. On a :

$$\begin{aligned}\operatorname{sn}(u|0) &= \sin u, & \operatorname{cn}(u|0) &= \cos u, & \operatorname{dn}(u|0) &= 1, \\ \operatorname{sn}(u|1) &= \tanh u, & \operatorname{cn}(u|1) &= \operatorname{dn}(u|1) &= \frac{1}{\cosh u}.\end{aligned}$$

- Paramètre négatif. Si m est positif, soit $\mu = \frac{m}{1+m}$, $v = u\sqrt{1+m}$, alors :

$$\begin{aligned}\operatorname{sn}(u|-m) &= \frac{1}{\sqrt{1+m}} \frac{\operatorname{sn}(v|\mu)}{\operatorname{dn}(v|\mu)}, \\ \operatorname{cn}(u|-m) &= \frac{\operatorname{cn}(v|\mu)}{\operatorname{dn}(v|\mu)}, \\ \operatorname{dn}(u|-m) &= \frac{1}{\operatorname{dn}(v|\mu)}.\end{aligned}$$

- Paramètre réciproque. Si m est positif, soit $\mu = \frac{1}{m}$ et $v = u\sqrt{m}$. alors :

$$\begin{aligned}\operatorname{sn}(u|m) &= \frac{1}{\sqrt{m}} \operatorname{sn}(v|\mu), \\ \operatorname{cn}(u|m) &= \operatorname{dn}(v|\mu), \\ \operatorname{dn}(u|m) &= \operatorname{cn}(v|\mu).\end{aligned}$$

- Formule d'addition. On a :

$$\begin{aligned}\operatorname{sn}(a+b|m) &= \frac{\operatorname{sn}(a|m) \operatorname{cn}(b|m) \operatorname{dn}(b|m) + \operatorname{sn}(b|m) \operatorname{cn}(a|m) \operatorname{dn}(a|m)}{1 - m \operatorname{sn}^2(a|m) \operatorname{sn}^2(b|m)}, \\ \operatorname{cn}(a+b|m) &= \frac{\operatorname{cn}(a|m) \operatorname{cn}(b|m) - \operatorname{sn}(a|m) \operatorname{dn}(a|m) \operatorname{sn}(b|m) \operatorname{dn}(b|m)}{1 - m \operatorname{sn}^2(a|m) \operatorname{sn}^2(b|m)}.\end{aligned}$$

- Argument complexe. On a les transformations complexes suivantes :

$$\begin{aligned}\operatorname{sn}(iu|m) &= i \frac{\operatorname{sn}(u|1-m)}{\operatorname{cn}(u|1-m)}, \\ \operatorname{cn}(iu|m) &= \frac{1}{\operatorname{cn}(u|1-m)}, \\ \operatorname{dn}(iu|m) &= \frac{\operatorname{dn}(u|1-m)}{\operatorname{cn}(u|1-m)}.\end{aligned}$$

Montrons, par exemple, comment on prouve la propriété $\operatorname{sn}(iu|m) = i \frac{\operatorname{sn}(u|1-m)}{\operatorname{cn}(u|1-m)}$. Comme nous l'avons déjà vu, $F(i\varphi, m) = iF(\theta, 1-m)$, avec $\tan \theta = \sinh \varphi$. Soit $u = F(\theta, 1-m)$. Donc $F(i\varphi, m) = iu$. Et :

$$\begin{aligned} \operatorname{sn}(iu|m) &= \operatorname{sn}(F(i\varphi, m)|m), \\ &= \sin i\varphi, \\ &= i \sinh \varphi, \\ &= i \tan \theta, \\ &= i \frac{\sin \theta}{\cos \theta}, \\ &= i \frac{\operatorname{sn}(u|1-m)}{\operatorname{cn}(u|1-m)}. \end{aligned}$$

Pour calculer numériquement sn , cn , dn , l'idée est de partir de u et α , de calculer les a_n, b_n en utilisant la méthode AGM et d'utiliser le fait que :

$$F(\varphi, \sin^2 \alpha) = u = \frac{\varphi_N}{2^N a_N}$$

pour calculer φ_N . On calcule ensuite $\varphi_{N-1}, \dots, \varphi_1, \varphi_0$ en utilisant :

$$\tan(\varphi_{n+1} - \varphi_n) = \frac{b_n}{a_n} \tan \varphi_n.$$

Et on obtient comme résultat :

$$\operatorname{sn}(u|m) = \sin \varphi_0, \quad \operatorname{cn}(u|m) = \cos \varphi_0, \quad \operatorname{dn}(u|m) = \sqrt{1 - c_0^2 \sin^2 \varphi_0}.$$

Notons que l'on peut aussi calculer les fonctions inverses en définissant la transformation de LANDEN pour ces fonctions. En effet, en utilisant l'équation $(1 + \sin \alpha)(1 + \cos \alpha_1) = 2$ de la transformation de LANDEN ascendante, on obtient :

$$\mu = \sin^2 \alpha_1 = \frac{4\sqrt{m}}{(1 + \sqrt{m})^2}.$$

Et aussi :

$$u = F(\varphi, \sin^2 \alpha) = \frac{2}{1 + \sin \alpha} F(\varphi_1, \sin^2 \alpha_1) = \frac{2}{1 + \sin \alpha} u_1,$$

donc :

$$u_1 = u \frac{1 + \sqrt{m}}{2}.$$

Si on prend la relation $\sin(2\varphi_1 - \varphi) = \sqrt{m} \sin \varphi$, on obtient :

$$\tan \varphi = \frac{\sin 2\varphi_1}{\sqrt{m} + \cos 2\varphi_1}.$$

Puis :

$$\operatorname{sn}(u|m) = \sin \varphi = \frac{\tan \varphi}{\sqrt{1 + \tan^2 \varphi}}.$$

En effectuant des remplacements et en développant l'équation on a donc :

$$\operatorname{sn}(u|m) = (1 + \sqrt{1 - \mu}) \frac{\operatorname{sn}(u_1|\mu) \operatorname{cn}(u_1|\mu)}{\operatorname{dn}(u_1|\mu)}.$$

De la même façon, on obtient :

$$\operatorname{dn}(u|m) = \frac{1 - \sqrt{1 - \mu} \operatorname{dn}^2(u_1|\mu) + \sqrt{1 - \mu}}{\mu \operatorname{dn}(u_1|\mu)}.$$

On a clairement, $u_1 < u$ et $\mu > m$ (quand $m \neq 1$) c'est donc bien une transformation ascendante. On s'arrête quand μ est suffisamment proche de 1.

A.1.4 Les fonctions Theta

Considérons les définitions de fonctions suivantes données par D. LAW DEN dans [Law 89] :

$$\theta_1(z, q) = 2 \sum_{n=0}^{\infty} (-1)^n q^{((n+1)/2)^2} \sin(2n+1)z,$$

$$\theta_2(z, q) = 2 \sum_{n=0}^{\infty} q^{((n+1)/2)^2} \cos(2n+1)z,$$

$$\theta_3(z, q) = 1 + 2 \sum_{n=1}^{\infty} q^{n^2} \cos 2nz,$$

$$\theta_4(z, q) = 1 + 2 \sum_{n=1}^{\infty} (-1)^n q^{n^2} \cos 2nz.$$

On peut aussi écrire $\theta(z, q) = \theta(z|\tau)$, où $q = e^{i\pi\tau}$.

Une autre façon de définir les fonctions elliptiques de JACOBI est de le faire grâce aux fonctions Theta. En effet, soit :

$$\operatorname{sn}' u = \frac{\theta_3(0)}{\theta_2(0)} \cdot \frac{\theta_1(z)}{\theta_4(z)},$$

$$\operatorname{cn}' u = \frac{\theta_4(0)}{\theta_2(0)} \cdot \frac{\theta_2(z)}{\theta_4(z)},$$

$$\operatorname{dn}' u = \frac{\theta_4(0)}{\theta_3(0)} \cdot \frac{\theta_3(z)}{\theta_4(z)},$$

où $z = u/\theta_3^2(0)$. En utilisant une propriété des fonctions Theta :

$$\begin{aligned} \frac{d}{du} \operatorname{sn} &= \frac{dz}{du} \frac{\theta_3(0)}{\theta_2(0)} \theta_4^2(0) \frac{\theta_2(z)\theta_3(z)}{\theta_4^2(z)}, \\ &= \frac{\theta_4^2(0)}{\theta_2(0)\theta_3(0)} \frac{\theta_2(z)\theta_3(z)}{\theta_4^2(z)}, \\ &= \operatorname{cn} u \operatorname{dn} u. \end{aligned}$$

Soit $x = \operatorname{sn} u$. En utilisant les relations ci-dessus on a :

$$\frac{dx}{du} = \operatorname{cn} u \operatorname{dn} u = \sqrt{(1-x^2) \left(1 - \frac{\theta_2^4(0)}{\theta_3^4(0)} x^2\right)}.$$

Donc, si on pose $m = \theta_2^4(0)/\theta_3^4(0)$, on a :

$$u = \operatorname{sn}^{-1}(x) = \int_0^x \sqrt{(1-t^2)(1-mt^2)}^{-1} dt.$$

Et la fonction sn' est la même que celle définie précédemment.

Beaucoup de propriétés des fonctions elliptiques de JACOBI peuvent maintenant être obtenues en utilisant les propriétés des fonctions Theta. Considérons par exemple la formule d'addition. En divisant l'égalité :

$$\theta_1(x+y)\theta_4(x-y)\theta_2(0)\theta_3(0) = \theta_1(x)\theta_4(x)\theta_2(y)\theta_3(y) + \theta_2(x)\theta_3(x)\theta_1(y)\theta_4(y)$$

par

$$\theta_4(x+y)\theta_4(x-y)\theta_4^2(0) = \theta_4^2(x)\theta_4^2(y) - \theta_1^2(x)\theta_1^2(y)$$

et en posant $x = u/\theta_3^2(0)$ et $y = v/\theta_3^2(0)$ on arrive à :

$$\operatorname{sn}(u+v) = \frac{\operatorname{sn} u \operatorname{cn} v \operatorname{dn} v + \operatorname{sn} v \operatorname{cn} u \operatorname{dn} u}{1 - m \operatorname{sn}^2 u \operatorname{sn}^2 v}.$$

Une autre façon plus directe de prouver cela est de procéder comme suit. Tout d'abord, trouver une condition sur $\varphi_1, \varphi_2, \varphi_3$ pour que :

$$F(\varphi_1, m) = F(\varphi_2, m) + F(\varphi_3, m).$$

Ceci découle du théorème d'addition d'EULER, qui dit que la condition pour que

$$\int_0^a \frac{dx}{\sqrt{P(x)}} = \int_0^b \frac{dx}{\sqrt{P(x)}} + \int_0^c \frac{dx}{\sqrt{P(x)}},$$

où $P(x) = 1 + l_1 x^2 + l_2 x^4$, donc :

$$a = \frac{b\sqrt{P(c)} + c\sqrt{P(b)}}{1 - l_2 b^2 c^2}.$$

Dans notre cas (A.1.4) est satisfait si :

$$\sin \varphi_1 = \frac{\sin \varphi_2 \cos \varphi_3 \sqrt{1 - m \sin^2 \varphi_3} + \sin \varphi_3 \cos \varphi_2 \sqrt{1 - m \sin^2 \varphi_2}}{1 - m \sin^2 \varphi_2 \sin^2 \varphi_3},$$

et on obtient donc directement le théorème d'addition pour sn .

A.1.5 Approximations des fonctions elliptiques

Dans l'équation de LEGENDRE

$$F(\varphi, m) = \operatorname{sn}^{-1}(\sin \varphi, m) = \int_0^\varphi \frac{d\theta}{\sqrt{1 - m \sin^2 \theta}},$$

où $x = \sin \varphi$. En développant le noyau de l'intégrale suivant les puissances croissantes de m , on trouve :

$$\sqrt{1 - m \sin^2 \theta}^{-1} = 1 + \frac{1}{2} \sin^2 \theta m + \frac{3}{8} \sin^4 \theta m^2 + \frac{5}{16} \sin^6 \theta m^3 + \frac{35}{128} \sin^8 \theta m^4 + O(m^5).$$

En intégrant terme à terme, on trouve :

$$\begin{aligned} F(\varphi, m) &= \varphi + \frac{1}{4}(\varphi - \cos \varphi \sin \varphi)m + \frac{3}{64} \left[3\varphi - \cos \varphi \sin \varphi (3 + 2 \sin^2 \varphi) \right] m^2 \\ &\quad + \frac{5}{768} \left[15\varphi - \cos \varphi \sin \varphi (15 + 10 \sin^2 \varphi + 8 \sin^4 \varphi) \right] m^3 \\ &+ \frac{35}{49152} \left[105\varphi - \cos \varphi \sin \varphi (105 + 70 \sin^2 \varphi + 56 \sin^4 \varphi + 48 \sin^6 \varphi) \right] m^4 + O(m^5). \quad (\text{A.1}) \end{aligned}$$

L'équation (A.1) est équivalente à :

$$\begin{aligned} u = \operatorname{sn}^{-1}(x, m) &= \sin^{-1} x + \frac{1}{4}(\sin^{-1} x - x\sqrt{1-x^2})m \\ &\quad + \frac{3}{64}(3 \sin^{-1} x - 3x\sqrt{1-x^2} - 2x(1-x^2)^{3/2})m^2 + \dots \end{aligned}$$

Ces séries peuvent, maintenant, être inversées pour développer $\operatorname{sn} u$ suivant les puissances de m . L'approximation en $O(m)$ est facile :

$$u = \sin^{-1} x + O(m),$$

donc

$$\operatorname{sn}(u|m) = \sin u + O(m).$$

Pour l'approximation du second ordre on utilise le fait que :

$$\sin^{-1} x = u - \frac{1}{4}(\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2).$$

Ce qui donne :

$$\begin{aligned} \operatorname{sn}(u|m) &= \sin \left(u - \frac{1}{4}(\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2) \right), \\ &= \sin u - \frac{1}{4} \cos u (\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2), \\ &= \sin u + \frac{1}{4} \cos u (\sin u \cos u - u)m + O(m^2), \end{aligned}$$

après la substitution $x = \sin u + O(m)$ dans le terme en $O(m)$. Les termes d'ordre plus élevé peuvent être obtenus en répétant cette procédure. Nous obtenons finalement :

$$\begin{aligned} \operatorname{sn}(u|m) &= \sin u + \frac{1}{4} \cos u (\sin u \cos u - u)m \\ &\quad - \frac{1}{64} \left[\sin u \cos^2 u (1 - 16 \cos^2 u) - 3u \cos u (1 - 4 \cos^2 u) + 2u^2 \sin u \right] m^2 \\ &\quad + \frac{1}{768} \left[3 \sin u \cos^2 u (4 - 31 \cos^2 u + 88 \cos^4 u) - 3u \cos u (3 - 52 \cos^2 u + 80 \cos^4 u) \right. \\ &\quad \left. - 3u^2 \sin u (1 + 18 \cos^2 u) + 2u^3 \cos u \right] m^3 + O(m^4). \end{aligned}$$

Il est aussi facile de prouver que :

$$F(\varphi, m) = \varphi + \frac{m}{6} \varphi^3 + \frac{m}{120} (9m - 4) \varphi^5 + O(\varphi^7).$$

Et comme :

$$\begin{aligned} \frac{d}{du} \operatorname{sn}(u|m) &= \operatorname{cn}(u|m) \operatorname{dn}(u|m), & \frac{d}{du} \operatorname{cn}(u|m) &= -\operatorname{sn}(u|m) \operatorname{dn}(u|m), \\ \frac{d}{du} \operatorname{dn}(u|m) &= -m \operatorname{sn}(u|m) \operatorname{cn}(u|m) \end{aligned}$$

nous pouvons calculer les dérivées d'ordre supérieur en appliquant ces résultats plusieurs fois d'affilée. Comme $\operatorname{sn}(0|m) = 0$, $\operatorname{cn}(0|m) = \operatorname{dn}(0|m) = 1$, en utilisant le théorème de MACLAURIN, le développement en séries entières de ces fonctions peut être déterminé en $u \approx 0$. On a :

$$\begin{aligned} \operatorname{sn}(u|m) &= u - \frac{1+m}{6} u^3 + \frac{1+14m+m^2}{120} u^5 + O(u^7), \\ \operatorname{cn}(u|m) &= 1 - \frac{1}{2} u^2 + \frac{1+4m}{24} u^4 + O(u^6), \\ \operatorname{dn}(u|m) &= 1 - \frac{m}{2} u^2 + \frac{m(4+m)}{24} u^4 + O(u^6). \end{aligned}$$

A.2 Intégrales elliptiques de seconde espèce

A.2.1 Définitions

Une *intégrale elliptique de seconde espèce* est de la forme :

$$\int \frac{t^2}{\sqrt{(A_1 t^2 + B_1)(A_2 t^2 + B_2)}} dt$$

Toutes les intégrales de seconde espèce peuvent être ramenées à l'évaluation d'une fonction epsilon de JACOBI (sous la forme de Legendre, plus de détails sont donnés par M. ABRAMOWITZ et C. STEGUN dans [Abr 72]) :

$$E(\phi, m) = \int_0^{\sin \phi} \sqrt{\frac{1 - mt^2}{1 - t^2}} dt = \int_0^{\phi} x \sqrt{1 - m \sin^2 \theta} d\theta.$$

Avec le changement de variable suivant $w = F(\theta, m)$, et comme $dw = \frac{d\theta}{\operatorname{dn}(w|m)}$, on écrit aussi :

$$E(\varphi, m) = E(u|m) = \int_0^u \operatorname{dn}^2(w|m) dw.$$

Ces intégrales ont les propriétés suivantes :

- Valeurs particulières. On a :

$$\begin{aligned} E(\varphi, 0) &= \varphi, & E(i\varphi, 0) &= i\varphi, \\ E(\varphi, 1) &= \sin \varphi, & E(i\varphi, 1) &= i \sinh \varphi. \end{aligned}$$

- Amplitude négative. $E(-\varphi, m) = -E(\varphi, m)$.

- Amplitude imaginaire. If $\tan \theta = \sinh \varphi$, donc

$$E(i\varphi, m) = i \left(F(\theta, 1-m) - E(\theta, 1-m) + \tan \theta \sqrt{\cos^2 \theta + m \sin^2 \theta} \right).$$

Ce qui peut se réécrire

$$E(iu|m) = i \left(u - E(u|1-m) + \frac{\operatorname{dn}(u|1-m) \operatorname{sn}(u|1-m)}{\operatorname{cn}(u|1-m)} \right).$$

- Paramètre négatif. Si m est positif :

$$E(u|-m) = \sqrt{1+m} E\left(u\sqrt{1+m} \middle| \frac{m}{m+1}\right) - m \frac{\operatorname{sn}\left(u\sqrt{1+m} \middle| \frac{m}{1+m}\right) \operatorname{cn}\left(u\sqrt{1+m} \middle| \frac{m}{1+m}\right)}{\operatorname{dn}\left(u\sqrt{1+m} \middle| \frac{m}{1+m}\right)}.$$

- Paramètre réciproque. On a :

$$E(u|m) = \sqrt{m} E\left(u\sqrt{m} \middle| \frac{1}{m}\right) - (m-1)u.$$

Ce qui est équivalent à :

$$E(\varphi, m) = \sqrt{m} E\left(\theta, \frac{1}{m}\right) - \frac{m-1}{\sqrt{m}} F\left(\theta, \frac{1}{m}\right),$$

où $\sin \theta = \sqrt{m} \sin \varphi$.

Notons que comme $E(u|m) = \int_0^u \operatorname{dn}^2(w|m) dw$, on peut prendre la dérivée de l'équation ci-dessus :

$$\operatorname{dn}^2(u|m) = m \operatorname{dn}^2\left(u\sqrt{m} \middle| \frac{1}{m}\right) - (m-1),$$

qui se simplifie de la façon suivante :

$$\operatorname{sn}(u|m) = \frac{1}{\sqrt{m}} \operatorname{sn}\left(u\sqrt{m} \middle| \frac{1}{m}\right).$$

Les propriétés des fonctions Theta permettent de démontrer le théorème d'addition pour E :

$$E(u+v|m) = E(u|m) + E(v|m) - m \operatorname{sn}(u|m) \operatorname{sn}(v|m) \operatorname{sn}(u+v|m).$$

Ce qui revient à dire que si $F(\mu, m) = F(\phi, m) + F(\psi, m)$, alors :

$$E(\phi, m) + E(\psi, m) - E(\mu, m) = m \sin \phi \sin \psi \sin \mu.$$

Notons aussi qu'en combinant deux des propriétés ci-dessus on peut prouver que :

$$E(i\mu, t) = i \left(\frac{1}{\sqrt{t}} F(o, t-1) - \sqrt{t} E(o, t-1) + \frac{t \sinh(2\mu)}{2\sqrt{1+t \sinh^2 \mu}} \right),$$

où $\tan o = \sqrt{t} \sinh \mu$.

Pour calculer $E(\varphi, \sin^2 \alpha)$, on peut utiliser la transformation de LANDEN ascendante. Après de long calculs, on obtient :

$$E(\varphi_n, \sin^2 \alpha_n) = (1 + \sin \alpha_n) E(\varphi_{n+1}, \sin^2 \alpha_{n+1}) + (1 - \sin \alpha_n) F(\varphi_{n+1}, \sin^2 \alpha_{n+1}) - \sin \alpha_n \sin \varphi_n.$$

Quand α_n est suffisamment proche de $\frac{\pi}{2}$, on a :

$$E(\varphi_N, \sin^2 \alpha_N) = E(\varphi_N, 1) \simeq \sin \varphi_N.$$

Et de là, on peut calculer $E(\varphi, \sin^2 \alpha)$.

A.2.2 Inverses des fonctions intégrales de seconde espèce

Le calcul des inverses des fonctions intégrales de seconde espèce est beaucoup plus complexe que pour les fonctions de seconde espèce et peu de recherche ont réellement donné des résultats sur ce sujet.

Soit $v = E(\varphi, m)$ et m , on définit les fonctions inverses :

$$\begin{aligned} s_1(v|m) &= \sin \varphi, \\ c_1(v|m) &= \cos \varphi, \\ d_1(v|m) &= \frac{1}{\sqrt{1 - m \sin^2 \varphi}}. \end{aligned}$$

Ces fonctions ont les propriétés suivantes :

- Valeurs spéciales. On a :

$$\begin{aligned} s_1(v|0) &= \sin v, & c_1(v|0) &= \cos v, \\ s_1(v|1) &= v, & c_1(v|1) &= \sqrt{1 - v^2}. \end{aligned}$$

- Paramètre négatif. On a :

$$s_1(-u|m) = -s_1(u|m), \quad c_1(-u|m) = c_1(u|m),$$

Comme

$$\frac{dE^{-1}}{dv} = \frac{1}{\frac{dv}{d\varphi}} = \frac{1}{\sqrt{1 - m \sin^2 \varphi}} = d_1(v|m),$$

on peut calculer

$$\frac{d}{dv} s_1(v|m) = c_1(v|m) \frac{dE^{-1}}{dv} = c_1(v|m) d_1(v|m).$$

Et de même :

$$\begin{aligned} \frac{d}{dv} c_1(v|m) &= -s_1(v|m) d_1(v|m), \\ \frac{d}{dv} d_1(v|m) &= m c_1(v|m) s_1(v|m) d_1^4(v|m). \end{aligned}$$

A.2.3 Approximations

Dans la forme de LEGENDRE :

$$E(\varphi, m) = \int_0^\varphi \sqrt{1 - m \sin^2 \theta} d\theta.$$

En développant l'intégrale suivant les puissances croissantes de m :

$$\sqrt{1 - m \sin^2 \theta} = 1 - \frac{1}{2} \sin^2 \theta m - \frac{1}{8} \sin^4 \theta m^2 - \frac{1}{16} \sin^6 \theta m^3 - \frac{5}{128} \sin^8 \theta m^4 + O(m^5).$$

On intègre terme à terme et on obtient :

$$\begin{aligned} E(\varphi, m) &= \varphi + \frac{1}{4} (\cos \varphi \sin \varphi - \varphi) m + \frac{1}{64} \left[\cos \varphi \sin \varphi (3 + 2 \sin^2 \varphi) - 3\varphi \right] m^2 \\ &\quad + \frac{1}{768} \left[\cos \varphi \sin \varphi (15 + 10 \sin^2 \varphi + 8 \sin^4 \varphi) - 15\varphi \right] m^3 \\ &+ \frac{5}{49152} \left[\cos \varphi \sin \varphi (105 + 70 \sin^2 \varphi + 56 \sin^4 \varphi + 48 \sin^6 \varphi) - 105\varphi \right] m^4 + O(m^5). \quad (\text{A.2}) \end{aligned}$$

L'équation (A.2) est équivalente à :

$$v = s_1^{-1}(x, m) = \sin^{-1} x + \frac{1}{4} (x \sqrt{1 - x^2} - x) m + \frac{1}{64} (3x \sqrt{1 - x^2} + 2x(1 - x^2)^{3/2} - 3 \sin^{-1} x) + \dots$$

Cette série peut maintenant être inversée et développée en série une série entière $s_1(v|m)$ suivant les puissances croissantes de m . L'approximation en $O(m)$ est facile :

$$v = s_1^{-1}(x, m) + O(m),$$

donc

$$s_1(v|m) = \sin v + O(m).$$

Pour l'approximation au second ordre, on utilise le fait que :

$$\sin^{-1} x = v + \frac{1}{4}(\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2).$$

Et par conséquent,

$$\begin{aligned} s_1(v|m) &= \sin \left(v + \frac{1}{4}(\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2) \right), \\ &= \sin v + \frac{1}{4} \cos v (\sin^{-1} x - x\sqrt{1-x^2})m + O(m^2), \\ &= \sin v + \frac{1}{4} \cos v (v - \sin v \cos v)m + O(m^2), \end{aligned}$$

en substituant $x = \sin v + O(m)$ dans le terme en $O(m)$. Les termes d'ordre plus élevé peuvent être obtenus en répétant cette procédure. Nous terminons par :

$$\begin{aligned} s_1(v|m) &= \sin v + \frac{1}{4} \cos v (v - \sin v \cos v)m \\ &\quad - \frac{1}{64} \left[3 \sin v \cos^2 v (5 - 4 \cos^2 v) - 3v \cos v (5 - 4 \cos^2 v) + 2v^2 \sin v \right] m^2 \\ &\quad - \frac{1}{768} \left[\sin v \cos^2 v (174 - 307 \cos^2 v + 148 \cos^4 v) - 3v \cos v (59 - 114 \cos^2 v + 60 \cos^4 v) \right. \\ &\quad \left. + 3v^2 \sin v (11 - 18 \cos^2 v) + 2v^3 \cos v \right] m^3 + O(m^4). \end{aligned}$$

Il est aussi facile de prouver que :

$$E(\varphi, m) = \varphi - \frac{m}{6} \varphi^3 + \frac{m}{120} (4m - 3) \varphi^5 + O(\varphi^7).$$

En procédant de la même façon que ci-dessus, on trouve :

$$\begin{aligned} s_1(v|m) &= v - \frac{1-m}{6} v^3 + \frac{1-13m+14m^2}{120} v^5 + O(v^7), \\ c_1(v|m) &= 1 - \frac{1}{2} v^2 + \frac{1-4m}{24} v^4 + O(v^6). \end{aligned}$$

Ceci peut aussi être obtenu en utilisant les dérivées de s_1, c_1, d_1 que l'on trouve grâce à :

$$s_1(v|m) = s_1(0|m) + s_1'(0|m)v + \frac{1}{2} s_1''(0|m)v^2 + \dots,$$

avec $s_1(0|m) = 0$ et $c_1(0|m) = d_1(0|m) = 1$.

Annexe B

Projections en 2D

Nous donnons dans cette annexe des fonctions de projection à Jacobien constant pour toutes les courbes coniques. Les coniques se divisent en trois catégories : les ellipses, les paraboles et les hyperboles. Pour chacune de ces catégories, deux cas sont à envisager : le cas où les axes de la conique sont perpendiculaires entre eux et le cas où ils ne le sont pas.

B.1 Ellipse dont les axes sont orthogonaux.

Considérons l'équation canonique suivante d'une ellipse :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \quad \begin{cases} x = a \cos t, \\ y = b \sin t. \end{cases}$$

En supposant que $t = g(v)$. La norme du vecteur vitesse vaut :

$$\mathcal{V} = \sqrt{\left(\frac{\partial x}{\partial v}\right)^2 + \left(\frac{\partial y}{\partial v}\right)^2} = g'(v) \sqrt{a^2 \sin^2 t + b^2 \cos^2 t}.$$

Comme :

$$\frac{\partial v}{\partial t} = \frac{\partial}{\partial t} g^{-1}(t) = \frac{1}{g'(g^{-1}(t))} = \frac{1}{g'(v)},$$

on a alors :

$$\frac{\partial}{\partial t} g^{-1}(t) = \frac{1}{\mathcal{V}} \sqrt{a^2 \sin^2 t + b^2 \cos^2 t}.$$

Supposons par exemple que $b \geq a$. On cherche alors \mathcal{V} constant. Prenons le, par exemple, égal à b , alors :

$$v = g^{-1}(t) = \frac{1}{b} \int_0^t \sqrt{a^2 \sin^2 l + b^2 \cos^2 l} dl = E\left(t, \frac{b^2 - a^2}{b^2}\right).$$

g est donc l'inverse de la fonction elliptique de seconde espèce. On peut, alors écrire :

$$\begin{cases} x = ac_1 \left(v \left| \frac{b^2 - a^2}{b^2} \right. \right), \\ y = bs_1 \left(v \left| \frac{b^2 - a^2}{b^2} \right. \right). \end{cases}$$

Si $a \geq b$, on peut prendre :

$$\begin{cases} x = as_1 \left(v \sqrt{\frac{a^2-b^2}{a^2}} \right), \\ y = bc_1 \left(v \sqrt{\frac{a^2-b^2}{a^2}} \right), \end{cases}$$

où $v = E\left(t, \frac{a^2-b^2}{a^2}\right)$.

B.2 Ellipse dont les axes ne sont pas orthogonaux

Prenons l'équation canonique d'une ellipse :

$$\frac{x^2}{a^2} + \frac{(x \cos \theta + y \sin \theta)^2}{b^2} = 1, \quad \begin{cases} x = a \cos t, \\ y = \frac{b}{\sin \theta} \sin t - a \cot \theta \cos t. \end{cases}$$

Alors,

$$\begin{aligned} f(t) &= \frac{1}{\sin \theta} \int_0^t \sqrt{a^2 \sin^2 l + b^2 \cos^2 l + ab \sin 2l \cos \theta} dl, \\ &= \frac{1}{\sin \theta} \int_0^t \sqrt{\frac{a^2 + b^2}{2} + \frac{\lambda}{2} \cos(2l + \beta)} dl, \\ &= \frac{1}{\sin \theta} \int_0^t \sqrt{\frac{a^2 + b^2 + \lambda}{2} - \lambda \sin^2 \left(l + \frac{\beta}{2}\right)} dl, \\ &= \frac{\sqrt{C_1}}{\sin \theta} \int_0^t \sqrt{1 - \frac{\lambda}{C_1} \sin^2 \left(l + \frac{\beta}{2}\right)} dl, \end{aligned}$$

où :

$$\tan \beta = \frac{2ab \cos \theta}{a^2 - b^2}, \quad \lambda^2 = a^4 + b^4 + 2a^2b^2 \cos 2\theta, \quad C_1 = \frac{a^2 + b^2 + \lambda}{2} \geq 0.$$

En prenant $\beta \in]-\pi/2, \pi/2[$. Alors λ est du signe de $b - a$. Supposons maintenant que $b > a$ de telle façon que $\lambda > 0$. On peut, alors, faire le changement de variable suivant : $l + \frac{\beta}{2} = s$. Ce qui donne :

$$\begin{aligned} f(t) &= \frac{\sqrt{C_1}}{\sin \theta} \int_{\frac{\beta}{2}}^{\frac{\beta}{2}+t} \sqrt{1 - \frac{\lambda}{C_1} \sin^2 s} ds, \\ &= \frac{\sqrt{C_1}}{\sin \theta} \left(E\left(t + \frac{\beta}{2}, \frac{\lambda}{C_1}\right) - E\left(\frac{\beta}{2}, \frac{\lambda}{C_1}\right) \right). \end{aligned}$$

Notons que $\frac{\lambda}{C_1} \leq 1$.

Prenons, maintenant, $t = g(v)$. On doit alors avoir, comme précédemment :

$$\frac{\partial}{\partial t} g^{-1}(t) = \frac{1}{v} f'(t).$$

Et si on choisit $\mathcal{V} = \frac{\sqrt{C_1}}{\sin \theta}$. On peut prendre :

$$v = g^{-1}(t) = \frac{\sin \theta}{\sqrt{C_1}} f(t) + E\left(\frac{\beta}{2}, \frac{\lambda}{C_1}\right) = E\left(t + \frac{\beta}{2}, \frac{\lambda}{C_1}\right).$$

On en déduit directement que $s_1(v|\lambda/C_1) = \sin(t + \beta/2)$, et donc :

$$\begin{aligned} x &= a \cos t = a \cos\left(t + \frac{\beta}{2} - \frac{\beta}{2}\right), \\ &= a \left[\cos \frac{\beta}{2} \cos\left(t + \frac{\beta}{2}\right) + \sin \frac{\beta}{2} \sin\left(t + \frac{\beta}{2}\right) \right], \\ &= a \left[\cos \frac{\beta}{2} c_1\left(v \middle| \frac{\lambda}{C_1}\right) + \sin \frac{\beta}{2} s_1\left(v \middle| \frac{\lambda}{C_1}\right) \right]. \end{aligned}$$

De la même façon :

$$y = \left(\frac{b}{\sin \theta} \cos \frac{\beta}{2} - a \cot \theta \sin \frac{\beta}{2} \right) s_1\left(v \middle| \frac{\lambda}{C_1}\right) - \left(\frac{b}{\sin \theta} \sin \frac{\beta}{2} + a \cot \theta \cos \frac{\beta}{2} \right) c_1\left(v \middle| \frac{\lambda}{C_1}\right).$$

B.3 Hyperbole dont les axes sont orthogonaux

L'équation canonique d'une hyperbole :

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1, \quad \begin{cases} x = a \cosh t, \\ y = b \sinh t. \end{cases}$$

La longueur d'un arc sur cette hyperbole vaut donc :

$$f(t) = \int_0^t \sqrt{b^2 + (a^2 + b^2) \sinh^2 l} \, dl.$$

En utilisant le changement de variable suivant : $l = -io$, on obtient :

$$f(t) = -i \int_0^{it} \sqrt{b^2 - (a^2 + b^2) \sin^2 o} \, do = -ibE\left(it, \frac{a^2 + b^2}{b^2}\right).$$

Si on prend $t = g(v)$ et $\mathcal{V} = b$, avec :

$$v = g^{-1}(t) = \frac{1}{b} f(t) = -iE\left(it, \frac{a^2 + b^2}{b^2}\right).$$

Comme $E\left(it, \frac{a^2 + b^2}{b^2}\right) = iv$, on peut utiliser :

$$\begin{cases} x = a \cosh t = a \cos(it) = ac_1\left(iv \middle| \frac{a^2 + b^2}{b^2}\right), \\ y = b \sinh t = -bi \sin(it) = -bis_1\left(iv \middle| \frac{a^2 + b^2}{b^2}\right). \end{cases}$$

On peut maintenant simplifier la formule pour $g^{-1}(t)$ de telle façon que l'angle soit un réel et que l'argument soit plus petit que 1. Tout d'abord :

$$E\left(it, \frac{a^2 + b^2}{b^2}\right) = \frac{\sqrt{a^2 + b^2}}{b} E\left(i\mu, \frac{b^2}{a^2 + b^2}\right) - \frac{a^2}{b\sqrt{a^2 + b^2}} F\left(i\mu, \frac{b^2}{a^2 + b^2}\right),$$

où $\sin(i\mu) = \frac{\sqrt{a^2 + b^2}}{b} \sin(it)$, c'est-à-dire que $\sinh \mu = \frac{\sqrt{a^2 + b^2}}{b} \sinh t$. De plus,

$$F\left(i\mu, \frac{b^2}{a^2 + b^2}\right) = iF\left(\lambda, \frac{a^2}{a^2 + b^2}\right),$$

$$E\left(i\mu, \frac{b^2}{a^2 + b^2}\right) = iF\left(\lambda, \frac{a^2}{a^2 + b^2}\right) - iE\left(\lambda, \frac{a^2}{a^2 + b^2}\right) + i \tan \lambda \sqrt{1 - \frac{a^2}{a^2 + b^2} \sin^2 \lambda},$$

où $\tan \lambda = \sinh \mu = \frac{\sqrt{a^2 + b^2}}{b} \sinh t$. En reprenant les équations, on aboutit à :

$$g^{-1}(t) = \frac{b}{\sqrt{a^2 + b^2}} F\left(\lambda, \frac{a^2}{a^2 + b^2}\right) - \frac{\sqrt{a^2 + b^2}}{b} E\left(\lambda, \frac{a^2}{a^2 + b^2}\right) + \frac{a^2 + b^2}{2b} \frac{\sinh(2t)}{\sqrt{a^2 \sinh^2 t + b^2 \cosh^2 t}}.$$

B.4 Hyperbole dont les axes ne sont pas orthogonaux.

La représentation canonique d'une hyperbole :

$$\frac{x^2}{a^2} - \frac{(x \cos \theta + y \sin \theta)^2}{b^2} = 1, \quad \begin{cases} x = a \cosh t, \\ y = \frac{b}{\sin \theta} \sinh t - a \cot \theta \cosh t. \end{cases}$$

Alors :

$$\begin{aligned} f(t) &= \frac{1}{\sin \theta} \int_0^t \sqrt{a^2 \sinh^2 l + b^2 \cosh^2 l - ab \sinh 2l \cos \theta} dl, \\ &= \frac{1}{\sin \theta} \int_0^t \sqrt{\frac{b^2 - a^2}{2} + \frac{\lambda}{2} \cosh(2l + \beta)} dl, \\ &= \frac{1}{\sin \theta} \int_0^t \sqrt{\frac{b^2 - a^2 + \lambda}{2} + \lambda \sinh^2 \left(l + \frac{\beta}{2}\right)} dl, \\ &= \frac{\sqrt{C_1}}{\sin \theta} \int_0^t \sqrt{1 + \frac{\lambda}{C_1} \sinh^2 \left(l + \frac{\beta}{2}\right)} dl, \end{aligned}$$

où

$$\tanh \beta = -\frac{2ab \cos \theta}{a^2 + b^2}, \quad \lambda = \sqrt{a^4 + b^4 - 2a^2 b^2 \cos 2\theta}, \quad C_1 = \frac{b^2 - a^2 + \lambda}{2} \geq 0.$$

En faisant le changement de variable suivant : $l + \frac{\beta}{2} = -is$, on obtient :

$$\begin{aligned} f(t) &= -i \frac{\sqrt{C_1}}{\sin \theta} \int_{i\frac{\beta}{2}}^{it + \frac{\beta}{2}} \sqrt{1 - \frac{\lambda}{C_1} \sin^2 s} ds, \\ &= i \frac{\sqrt{C_1}}{\sin \theta} \left(E\left(i\frac{\beta}{2}, \frac{\lambda}{C_1}\right) - E\left(it + i\frac{\beta}{2}, \frac{\lambda}{C_1}\right) \right). \end{aligned}$$

On remarque facilement que :

$$1 \leq \frac{\lambda}{C_1} \leq \frac{a^2 + b^2}{b^2}.$$

En prenant $t = g(v)$ et en choisissant $\mathcal{V} = \frac{\sqrt{C_1}}{\sin \theta}$ et on prend :

$$v = g^{-1}(t) = \frac{\sin \theta}{\sqrt{C_1}} f(t) - iE\left(i\frac{\beta}{2}, \frac{\lambda}{C_1}\right) = -iE\left(it + i\frac{\beta}{2}, \frac{\lambda}{C_1}\right).$$

On a, alors, $\sin(it + i\beta/2) = s_1(iv|\lambda/C_1)$, qui se réécrit : $\sinh(t + \beta/2) = -is_1(iv|\lambda/C_1)$ et donc $\cosh(t + \beta/2) = c_1(iv|\lambda/C_1)$. Enfin :

$$\begin{aligned} x &= a \cosh t = a \cosh\left(t + \frac{\beta}{2} - \frac{\beta}{2}\right), \\ &= a \left[\cosh \frac{\beta}{2} \cosh\left(t + \frac{\beta}{2}\right) - \sinh \frac{\beta}{2} \sinh\left(t + \frac{\beta}{2}\right) \right], \\ &= a \left[\cosh \frac{\beta}{2} c_1\left(iv\left|\frac{\lambda}{C_1}\right.\right) + i \sinh \frac{\beta}{2} s_1\left(iv\left|\frac{\lambda}{C_1}\right.\right) \right]. \end{aligned}$$

Et de même :

$$y = -i \left(\frac{b}{\sin \theta} \cosh \frac{\beta}{2} + a \cot \theta \sinh \frac{\beta}{2} \right) s_1\left(iv\left|\frac{\lambda}{C_1}\right.\right) - \left(\frac{b}{\sin \theta} \sinh \frac{\beta}{2} + a \cot \theta \cosh \frac{\beta}{2} \right) c_1\left(iv\left|\frac{\lambda}{C_1}\right.\right)$$

On peut, alors, une fois encore, simplifier un peu $g^{-1}(t)$:

$$\begin{aligned} g^{-1}(t) &= \sqrt{\frac{C_1}{\lambda}} F\left(\alpha, \frac{\lambda - C_1}{\lambda}\right) - \sqrt{\frac{\lambda}{C_1}} E\left(\alpha, \frac{\lambda - C_1}{\lambda}\right) \\ &\quad + \frac{\lambda}{2} \frac{\sinh(2t + \beta)}{\sqrt{C_1(\lambda - C_1) \sinh^2\left(t + \frac{\beta}{2}\right) + C_1^2 \cosh^2\left(t + \frac{\beta}{2}\right)}}, \end{aligned}$$

où $\tan \alpha = \sqrt{\lambda/C_1} \sinh(t + \beta/2)$.

B.5 Parabole dont les axes sont orthogonaux

L'équation canonique d'une parabole :

$$y^2 = 2px, \quad \begin{cases} x = \frac{p}{2} \cot^2 t, \\ y = p \cot t \end{cases}$$

La longueur d'un arc sur cette parabole vaut donc :

$$\begin{aligned} f(\varphi) &= p \int_{\pi/2}^{\varphi} \frac{dt}{\sin^3 t}, \\ &= \frac{p}{2} \left(\ln \left(\frac{\sin \varphi}{1 + \cos \varphi} \right) - \frac{\cos \varphi}{\sin^2 \varphi} \right). \end{aligned}$$

Une solution est d'utiliser :

$$\begin{cases} x = \frac{p}{2} \cot^2(f^{-1}(\varphi)), \\ y = p \cot(f^{-1}(\varphi)). \end{cases}$$

B.6 Parabole dont les axes ne sont pas orthogonaux.

L'équation :

$$y^2 = 2p(x \cos \theta + y \sin \theta), \quad \begin{cases} x = \frac{p \cot^2 t}{2 \cos \theta} - p \cot t \tan \theta, \\ y = p \cot t \end{cases}$$

Alors :

$$f(\varphi) = \frac{p}{\cos \theta} \int_0^\varphi \frac{1}{\sin^3 t} \sqrt{1 - \sin \theta \sin 2t} dt.$$

On prend comme substitution $u = \tan t$, ce qui donne :

$$f(\varphi) = -\frac{p}{\cos \theta} \int_{-\infty}^{\tan \varphi} \frac{\sqrt{1 - 2u \sin \theta + u^2}}{u^3} du.$$

En passant cette équation à *Maple* on obtient après quelques simplifications :

$$f(\varphi) = \frac{p}{2} \left[\frac{\sin \theta \sin \varphi - \cos \varphi}{\cos \theta \sin^2 \varphi} \sqrt{1 - \sin \theta \sin (2\varphi)} - \cos \theta \operatorname{arctanh} \left(\frac{\cos \varphi - \sin \theta \sin \varphi}{\sqrt{1 - \sin \theta \sin (2\varphi)}} \right) - \cos \theta \operatorname{arctanh} (\sin \theta) - \tan \theta \right].$$

Notons que $\operatorname{arctanh} x = \frac{1}{2} \ln \left(\frac{1+x}{1-x} \right)$.

Annexe C

Les quadriques

Nous donnons dans cette annexe des fonctions de projection à Jacobien constant pour toutes les surfaces quadriques. Nous commençons cette annexe par redéfinir ce qu'est une surface quadrique, puis nous donnons une classification de ces surfaces avant de donner pour chacune d'elles une fonction de projection à Jacobien constant.

C.1 Définition et notations

Considérons la quadrique (Q) dans un espace affine 3D $\mathcal{R} = (O, \vec{i}, \vec{j}, \vec{k})$. L'équation générale de (Q) est donnée par $f(\vec{x}) = 0$ où $\vec{x} = (X, Y, Z)$ et f est un polynôme du second degré :

$$f(\vec{x}) = \underbrace{aX^2 + a'Y^2 + a''Z^2 + 2b''XY + 2b'XZ + 2bYZ}_{\phi(\vec{x})} + 2cX + 2c'Y + 2c''Z + d \quad (\text{C.1})$$

Considérons maintenant les deux matrices H et A et le vecteur $\vec{X} = (X, Y, Z, 1)$ tels que $\phi(\vec{x}) = \vec{x}H^t\vec{x}$ et $f(\vec{x}) = \vec{X}A^t\vec{X}$. H et A sont définies comme suit :

$$H = \begin{bmatrix} a & b'' & b' \\ b'' & a' & b \\ b' & b & a'' \end{bmatrix} \quad \text{et} \quad A = \begin{bmatrix} a & b'' & b' & c \\ b'' & a' & b & c' \\ b' & b & a'' & c'' \\ c & c' & c'' & d \end{bmatrix}.$$

Posons, maintenant, δ et Δ les déterminant associés à ces deux matrices : $\delta = \det H$ et $\Delta = \det A$.

C.2 Classification générale

Considérons, maintenant, les vecteurs propres $(\vec{\varepsilon}_1, \vec{\varepsilon}_2, \vec{\varepsilon}_3)$ de H et les valeurs propres associées S_1, S_2, S_3 de H . On peut alors définir la matrice P de transformation du repère

\mathcal{R} dans le repère $\mathcal{E} = (0, \vec{\varepsilon}_1, \vec{\varepsilon}_2, \vec{\varepsilon}_3)$ par :

$$P = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{bmatrix} \begin{matrix} \vec{i} \\ \vec{j} \\ \vec{k} \end{matrix} \\ \varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3$$

Comme les deux bases \mathcal{R} et \mathcal{E} ainsi définies sont orthonormales, la matrice P est orthogonale et on a $P^{-1} = {}^tP$. On peut alors exprimer \vec{x}' la représentation du vecteur \vec{x} dans \mathcal{E} de la façon suivante $\vec{x} = P\vec{x}'$. Et, finalement, si D est la matrice de ϕ dans \mathcal{E} on a :

$$D = \begin{bmatrix} S_1 & 0 & 0 \\ 0 & S_2 & 0 \\ 0 & 0 & S_3 \end{bmatrix} \quad \text{avec} \quad D = {}^tPHP$$

et $\phi(\vec{x}) = \phi'(\vec{x}') = S_1X'^2 + S_2Y'^2 + S_3Z'^2$. Dans \mathcal{E} l'équation de (Q) se réécrit donc :

$$(Q) : S_1X'^2 + S_2Y'^2 + S_3Z'^2 + 2c_1X' + 2c'_1Y' + 2c \gg_1 Z' + d_1 = 0. \quad (\text{C.2})$$

Avec $\vec{c} = (c, c', c \gg)$ et $\vec{c}'_1 = (c_1, c'_1, c \gg_1)$ et en utilisant les équations C.1 et C.2, on en déduit que $\vec{c} = P\vec{c}'_1$ et $d_1 = d$.

C.2.1 Cas où les valeurs propres sont toutes non-nulles

Nous considérons ici le cas où $S_1, S_2, S_3 \neq 0 (\Leftrightarrow \delta \neq 0)$, on peut alors réécrire l'équation (Q) de la façon suivante :

$$(Q) : S_1 \left(X' + \frac{c_1}{S_1} \right)^2 + S_2 \left(Y' + \frac{c'_1}{S_2} \right)^2 + S_3 \left(Z' + \frac{c \gg_1}{S_3} \right)^2 + d_2 = 0.$$

En considérant le nouveau point

$$\vec{\omega} = -\frac{c_1}{S_1} \vec{\varepsilon}_1 - \frac{c'_1}{S_2} \vec{\varepsilon}_2 - \frac{c \gg_1}{S_3} \vec{\varepsilon}_3,$$

l'équation de (Q) dans le repère $(\vec{\omega}, \vec{\varepsilon}_1, \vec{\varepsilon}_2, \vec{\varepsilon}_3)$ devient (on peut prouver que $d_2 = \frac{\Delta}{\delta}$) :

$$(Q) : S_1X \gg^2 + S_2Y \gg^2 + S_3Z \gg^2 + \frac{\Delta}{\delta} = 0.$$

Deux cas différents peuvent alors se présenter.

C.2.1.1 Si S_1, S_2, S_3 ont tous le même signe

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} + \frac{Z^2}{c^2} = \begin{cases} 1 & \text{ellipsoïde réel} \\ -1 & \emptyset \\ 0 & \text{un point} \end{cases}$$

C.2.1.2 Si S_1, S_2, S_3 sont de signes différents

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} - \frac{Z^2}{c^2} = \begin{cases} 1 & \text{hyperboloïde à une nappe} \\ -1 & \text{hyperboloïde à deux nappes} \\ 0 & \text{cône} \end{cases}$$

C.2.2 Cas où une des valeurs propres est nulle

Nous considérons ici le cas où $S_1, S_2 \neq 0, S_3 = 0$

$$(Q) : S_1 \left(X' + \frac{c_1}{S_1} \right)^2 + S_2 \left(Y' + \frac{c'_1}{S_2} \right)^2 + 2c \gg_1 Z' + d_2 = 0,$$

$$d_2 = d - \frac{c_1^2}{S_1} - \frac{c'^2_1}{S_2}.$$

C.2.2.1 Premier cas : $c \gg_1 \neq 0$

$$\vec{\omega} = -\frac{c_1}{S_1} \vec{\varepsilon}_1 - \frac{c'_1}{S_2} \vec{\varepsilon}_2 - \frac{d_2}{2c \gg_1} \vec{\varepsilon}_3,$$

et l'équation de (Q) dans le repère $(\vec{\omega}, \vec{\varepsilon}_1, \vec{\varepsilon}_2, \vec{\varepsilon}_3)$ devient :

$$(Q) : S_1 X \gg^2 + S_2 Y \gg^2 + 2c \gg_1 Z \gg = 0.$$

Ici encore, on distingue deux cas particuliers.

Si S_1, S_2 sont de même signe

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} - Z = 0 \quad \text{paraboloïde elliptique}$$

Si S_1, S_2 sont de signe différent

$$\frac{X^2}{a^2} - \frac{Y^2}{b^2} - Z = 0 \quad \text{paraboloïde hyperbolique}$$

C.2.2.2 Deuxième cas : $c \gg_1 = 0$

$$\vec{\omega} = -\frac{c_1}{S_1}\vec{\varepsilon}_1 - \frac{c_2}{S_2}\vec{\varepsilon}_2,$$

et l'équation de (Q) dans le repère $(\vec{\omega}, \vec{\varepsilon}_1, \vec{\varepsilon}_2, \vec{\varepsilon}_3)$ devient :

$$(Q) : S_1 X \gg^2 + S_2 Y \gg^2 + d_2 = 0$$

Ce qui nous donne, pour la classification, deux cas distincts.

Si S_1, S_2 sont de même signe

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} = \begin{cases} 1 & \text{cylindre elliptique} \\ -1 & \emptyset \\ 0 & \text{ligne} \end{cases}$$

Si S_1, S_2 sont de signe différent

$$\frac{X^2}{a^2} - \frac{Y^2}{b^2} = \begin{cases} 1 & \text{cylindre hyperbolic} \\ 0 & \text{deux plans} \end{cases}$$

C.2.2.3 Si deux des valeurs propres sont nulles

Nous considérons ici le cas où $S_2, S_3 = 0, S_1 \neq 0$. Alors :

$$(Q) : S_1 \left(X' + \frac{c_1}{S_1} \right)^2 + 2c'_1 Y' + 2c \gg_1 Z' + d_2 = 0,$$

$$d_2 = d - \frac{c_1^2}{S_1}.$$

Posons :

$$\vec{u} = 2c'_1 \vec{\varepsilon}_2 + 2c \gg_1 \vec{\varepsilon}_3.$$

$$\vec{I} = \vec{\varepsilon}_1, \quad \vec{J} = \frac{\vec{u}}{\|\vec{u}\|}, \quad \text{et} \quad \vec{K} = \vec{I} \wedge \vec{J}.$$

Dans le repère $(O, \vec{I}, \vec{J}, \vec{K})$, on a alors :

$$(Q) : S_1 \left(X \gg + \frac{c_1}{S_1} \right)^2 + 2c_2 Y \gg + d_2 = 0,$$

$$c_2 = \|\vec{u}\|.$$

Premier cas : $c_2 \neq 0$

$$\vec{\omega} = -\frac{c_1}{S_1} \vec{I} - \frac{d_2}{c_2} \vec{J},$$

et l'équation de (Q) dans le repère $(\vec{\omega}, \vec{I}, \vec{J}, \vec{K})$ devient :

$$(Q) : S_1 X \gg^2 + c_2 Y \gg = 0.$$

Ce qui nous donne, pour la classification :

$$X^2 = 2pZ \quad \text{cylindre parabolique}$$

Deuxième cas : $c_2 = 0$

$$\vec{\omega} = -\frac{c_1}{S_1} \vec{I},$$

et l'équation de (Q) dans le repère $(\vec{\omega}, \vec{I}, \vec{J}, \vec{K})$ devient :

$$(Q) : S_1 X \gg^2 + d_2 = 0.$$

Ce qui nous donne, pour la classification :

$$X^2 = p \quad \text{deux plans}$$

C.3 Les quadriques et leur fonction de projection à Jacobien constant

C.3.1 Ellipsoïde

Considérons l'équation d'un ellipsoïde, $\vec{x} = (x, y, z)$:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1, \quad \begin{cases} x = a \sin \varphi \sin \psi, \\ y = b \cos \varphi \sin \psi, \\ z = c \cos \psi. \end{cases}$$

Soit $\varphi = f(u)$ et $\psi = g(u, v)$. Alors :

$$\frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} = f'(u) g'_v(u, v) \sin g(u, v) \begin{cases} bc \sin(f(u)) \sin(g(u, v)), \\ ac \cos(f(u)) \sin(g(u, v)), \\ ab \cos(g(u, v)). \end{cases}$$

Donc :

$$\mathcal{V} = \left\| \frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} \right\|,$$

$$= f'(u) g'_v(u, v) \sin(g(u, v)) \sqrt{b^2 c^2 \sin^2(f(u)) \sin^2(g(u, v)) + a^2 c^2 \cos^2(f(u)) \sin^2(g(u, v)) + a^2 b^2 \cos^2(g(u, v))}.$$

On veut que \mathcal{V} soit constant, posons par exemple $\mathcal{V} = -1$. Alors :

$$\frac{\partial v}{\partial \psi} = \frac{1}{g'_v(u, v)},$$

et on peut prendre

$$v = f'(u) \int_{\psi}^{\frac{\pi}{2}} \sin t \sqrt{b^2 c^2 \sin^2 \varphi \sin^2 t + a^2 c^2 \cos^2 \varphi \sin^2 t + a^2 b^2 \cos^2 t} dt.$$

Soit $o = \cos t$. Alors :

$$\begin{aligned} v &= f'(u) \int_0^{\cos \psi} \sqrt{b^2 c^2 \sin^2 \varphi (1 - o^2) + a^2 c^2 \cos^2 \varphi (1 - o^2) + a^2 b^2 o^2} do, \\ &= f'(u) \sqrt{b^2 c^2 \sin^2 \varphi + a^2 c^2 \cos^2 \varphi} \int_0^{\cos \psi} \sqrt{1 + \alpha o^2} do, \end{aligned}$$

avec

$$\alpha = \frac{a^2 b^2}{b^2 c^2 \sin^2 \varphi + a^2 c^2 \cos^2 \varphi} - 1.$$

De là, on sait que :

$$\begin{aligned} \int \sqrt{1 + k^2 t^2} dt &= \frac{1}{2} \left(t \sqrt{1 + k^2 t^2} + \frac{1}{k} \operatorname{arcsinh}(kt) \right), \\ \int \sqrt{1 - k^2 t^2} dt &= \frac{1}{2} \left(t \sqrt{1 - k^2 t^2} + \frac{1}{k} \operatorname{arcsin}(kt) \right). \end{aligned}$$

Si $a \geq b \geq c$, alors :

$$a^2 b^2 = a^2 b^2 (\sin^2 \varphi + \cos^2 \varphi) \geq b^2 c^2 \sin^2 \varphi + a^2 c^2 \cos^2 \varphi,$$

de telle façon que $\alpha \geq 0$. Ainsi :

$$v = \frac{1}{2} f'(u) \frac{ab}{\sqrt{\alpha + 1}} \left(\cos \psi \sqrt{1 + \alpha \cos^2 \psi} + \frac{1}{\sqrt{\alpha}} \operatorname{arcsinh}(\sqrt{\alpha} \cos \psi) \right).$$

C.3.2 Hyperboloïde à une nappe

Soit l'équation d'un hyperboloïde à une nappe

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1, \quad \begin{cases} x = a \sin \varphi \cosh \psi, \\ y = b \cos \varphi \cosh \psi, \\ z = c \sinh \psi. \end{cases}$$

Soit $\varphi = f(u)$ et $\psi = g(u, v)$. Alors :

$$\frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} = f'(u) g'_v(u, v) \cosh(g(u, v)) \begin{vmatrix} -bc \sin(f(u)) \cosh(g(u, v)), \\ ac \cos(f(u)) \cosh(g(u, v)), \\ -ab \sinh(g(u, v)). \end{vmatrix}$$

On veut que

$$\mathcal{V} = \left\| \frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} \right\|$$

soit constant. Prenons par exemple $\mathcal{V} = 1$. Alors :

$$\frac{\partial v}{\partial \psi} = f'(u) \cosh \psi \sqrt{b^2 c^2 \sin^2 \varphi \cosh^2 \psi + a^2 c^2 \cos^2 \varphi \cosh^2 \psi + a^2 b^2 \sinh^2 \psi}.$$

Soit $o = \sinh t$. Alors :

$$v = f'(u) \int_0^{\sinh \psi} \sqrt{b^2 c^2 \sin^2 \varphi (1 - o^2) + a^2 c^2 \cos^2 \varphi (1 - o^2) + a^2 b^2 o^2} do,$$

donc si $a \geq b \geq c$ et si

$$\alpha = \frac{a^2 b^2}{b^2 c^2 \sin^2 \varphi + a^2 c^2 \cos^2 \varphi} - 1,$$

$$v = \frac{1}{2} f'(u) \frac{ab}{\sqrt{\alpha + 1}} \left(\sinh \psi \sqrt{1 + \alpha \sinh^2 \psi} + \frac{1}{\sqrt{\alpha}} \operatorname{arcsinh}(\sqrt{\alpha} \sinh \psi) \right).$$

C.3.3 Hyperboloïde à deux nappes

Soit l'équation d'un hyperboloïde à deux nappes :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = -1, \quad \begin{cases} x = a \sin \varphi \sinh \psi, \\ y = b \cos \varphi \sinh \psi, \\ z = c \cosh \psi. \end{cases}$$

Soit $\varphi = f(u)$ et $\psi = g(u, v)$. Donc, de même que ci-dessus :

$$\frac{\partial v}{\partial \psi} = f'(u) \sinh \psi \sqrt{b^2 c^2 \sin^2 \varphi \sinh^2 \psi + a^2 c^2 \cos^2 \varphi \sinh^2 \psi + a^2 b^2 \cosh^2 \psi}.$$

On a donc :

$$v = \frac{1}{2} f'(u) \frac{ab}{\sqrt{\alpha + 1}} \left(\cosh \psi \sqrt{1 + \alpha \cosh^2 \psi} + \frac{1}{\sqrt{\alpha}} \operatorname{arcsinh}(\sqrt{\alpha} \cosh \psi) \right).$$

C.3.4 Paraboloïde elliptique

On utilise le paramétrage suivant :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - z = 0, \quad \begin{cases} x = a \cos \varphi \sqrt{\psi}, \\ y = b \sin \varphi \sqrt{\psi}, \\ z = \psi. \end{cases}$$

Soit $\varphi = f(u)$ et $\psi = g(u, v)$. Alors :

$$\mathcal{V} = \frac{1}{2} f'(u) g'_v(u, v) \begin{vmatrix} 2b \cos(f(u)) \sqrt{g(u, v)}, \\ 2a \sin(f(u)) \sqrt{g(u, v)}, \\ -ab. \end{vmatrix}$$

En prenant $\mathcal{V} = 2/3$. On peut alors prendre

$$v = f'(u) \sqrt{b^2 \cos^2 \varphi + a^2 \sin^2 \varphi} \left(\psi + \frac{a^2 b^2}{4(b^2 \cos^2 \varphi + a^2 \sin^2 \varphi)} \right)^{3/2}.$$

Si par exemple $b \geq a$, on a alors :

$$\cos \varphi = c_1 \left(u \left| \frac{b^2 - a^2}{b^2} \right. \right), \quad \sin \varphi = s_1 \left(u \left| \frac{b^2 - a^2}{b^2} \right. \right), \quad u = E \left(\varphi, \frac{b^2 - a^2}{b^2} \right),$$

puis :

$$f'(u) = \frac{b}{\sqrt{b^2 \cos^2 \varphi + a^2 \sin^2 \varphi}}.$$

Dans ce cas :

$$v = b \left(\psi + \frac{a^2 b^2}{4(b^2 \cos^2 \varphi + a^2 \sin^2 \varphi)} \right)^{3/2}, \quad \psi = \left(\frac{v}{b} \right)^{2/3} - \frac{a^2 b^2}{4(b^2 \cos^2 \varphi + a^2 \sin^2 \varphi)}.$$

C.3.5 Paraboloïde hyperbolique

Avec le paramétrage suivant :

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} - z = 0, \quad \begin{cases} x = a \cosh \varphi \sqrt{\psi}, \\ y = b \sinh \varphi \sqrt{\psi}, \\ z = \psi. \end{cases}$$

On a, comme précédemment :

$$v = f'(u) \sqrt{b^2 \cosh^2 \varphi + a^2 \sinh^2 \varphi} \left(\psi + \frac{a^2 b^2}{4\sqrt{b^2 \cosh^2 \varphi + a^2 \sinh^2 \varphi}} \right)^{3/2}.$$

Et comme on l'a déjà vu :

$$\cosh \varphi = c_1 \left(iu \left| \frac{a^2 + b^2}{b^2} \right. \right), \quad \sinh \varphi = -i s_1 \left(iu \left| \frac{a^2 + b^2}{b^2} \right. \right), \quad u = -i E \left(i\varphi, \frac{a^2 + b^2}{b^2} \right),$$

puis :

$$f'(u) = \frac{b}{\sqrt{b^2 \cosh^2 \varphi + a^2 \sinh^2 \varphi}}.$$

Et, dans ce cas :

$$v = b \left(\psi + \frac{a^2 b^2}{4\sqrt{b^2 \cosh^2 \varphi + a^2 \sinh^2 \varphi}} \right)^{3/2}, \quad \psi = \left(\frac{v}{b} \right)^{2/3} - \frac{a^2 b^2}{4(b^2 \cosh^2 \varphi + a^2 \sinh^2 \varphi)}.$$

C.3.6 Cône

Soit les équations suivantes :

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 0, \quad \begin{cases} x = a \cos \varphi \psi, \\ y = b \sin \varphi \psi, \\ z = c \psi. \end{cases}$$

Alors :

$$\frac{\partial \mathbf{x}}{\partial u} \times \frac{\partial \mathbf{x}}{\partial v} = g(u,v) f'(u) g'_v(u,v) \begin{vmatrix} bc \cos(f(u)), \\ ac \sin(f(u)), \\ -ab \end{vmatrix}$$

Et on peut prendre :

$$v = f'(u) \sqrt{b^2 c^2 \cos^2 \varphi + a^2 c^2 \sin^2 \varphi + a^2 b^2 \psi^2}.$$

C.3.7 Cylindres

Pour le cylindre elliptique

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \quad \begin{cases} x = a \cos \varphi, \\ y = b \sin \varphi, \\ z = \psi. \end{cases}$$

Puis on peut prendre :

$$v = f'(u) \sqrt{b^2 \cos^2 \varphi + a^2 \sin^2 \varphi} \psi.$$

Bibliographie

- [Abr 72] M. Abramowitz et C. Stegun. Elliptic integrals. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Table*, chapitre 17, pages 587–607. Dover, New York, 1972. (p. 173)
- [Alo 99] Laurent Alonso et Nicolas Holzschuch. Using graphics hardware to speed-up your visibility queries. *Journal of Graphics Tools*, 1999. To appear (available from <http://www.loria.fr/publications/1999/99-R-030/99-R-030.ps>). (pp. 58, 74)
- [Alo 00a] Laurent Alonso, Francois Cuny, Sylvain Petitjean et Jean-Claude Paul. The Virtual Mesh : A Geometric Abstraction for Efficiently Computing Radiosity. Rapport de recherche, 2000.
- [Alo 00b] Laurent Alonso, Sylvain Petitjean, François Cuny, Nicolas Holzschuch et Jean-Claude Paul. Constant jacobian mappings : Accurate wavelet radiosity on parametric surfaces. Rapport de recherche, 2000. (pp. 5, 32)
- [Arq 95] Didier Arques et Sylvain Michelin. A New Radiosity Approach for Regular Objects : Application to Ruled Surfaces. *Computer Graphics Forum (Eurographics '95)*, 14(3) :C299–C310, September 1995. (p. 85)
- [Arv 94] James Arvo, Kenneth Torrance et Brian Smits. A Framework for the Analysis of Error in Global Illumination Algorithms. *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 75–84, 1994. (pp. 15, 24, 79, 87)
- [Arv 95] James Arvo. The Role of Functional Analysis in Global Illumination. P. M. Hanrahan et W. Purgathofer, éditeurs, *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, pages 115–126, New York, NY, 1995. Springer-Verlag. (pp. 6, 18, 25, 137)
- [Aup 93] L. Aupperle et Pat Hanrahan. A Hierarchical Illumination Algorithm for Surfaces with Glossy Reflection. *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 155–162, 1993. (p. 3)
- [Bau 91] Daniel R. Baum, Stephen Mann, Kevin P. Smith et James M. Winget. Making Radiosity Usable : Automatic Preprocessing and Meshing Techniques for the Generation of Accurate Radiosity Solutions. *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 51–60, July 1991. (p. 93)
- [Bek 96a] Philippe Bekaert et Yves Willems. Error Control for Radiosity. *Rendering Techniques '96 (Proceedings of the Seventh Eurographics Workshop on Rendering)*, pages 153–164, New York, NY, 1996. Springer-Verlag/Wien. (p. 52)

- [Bek 96b] Philippe Bekaert et Yves D. Willems. Hirad: A Hierarchical Higher Order Radiosity Implementation. *Proceedings of the Twelfth Spring Conference on Computer Graphics (SCCG '96)*, Bratislava, Slovakia, June 1996. Comenius University Press. (p. 52)
- [Bey 92] G. Beyklin, R. Coifman et V. Rokhlin. *Wavelets in numerical analysis*. Wavelet and their Applications. Cambridge, 1992. pages 181-210. (p. 37)
- [Boi 98] Jean-Daniel Boissonnat et Mariette Yvinec. *Algorithmic Geometry*. Cambridge University Press, 1998. (p. 105)
- [Bou 95] K. Bouatouch et Sumanta N. Pattanaik. Discontinuity Meshing and Hierarchical Multiwavelet Radiosity. W. A. Davis et P. Prusinkiewicz, éditeurs, *Proceedings of Graphics Interface '95*, pages 109–115, San Francisco, CA, May 1995. Morgan Kaufmann. (p. 92)
- [Cat 75] Edwin E. Catmull. Computer display of curved surfaces. *Proceedings of the IEEE Conference on Computer Graphics, Pattern Recognition, and Data Structure*, pages 11–17, mai 1975. (p. 2)
- [Cav 98] Xavier Cavin, Laurent Alonso et Jean-Claude Paul. Parallel wavelet radiosity. *Proceedings of the Second Eurographics Workshop on Parallel Graphics and Visualisation*, pages 61–75, Rennes, France, September 1998. Eurographics. (pp. 5, 125, 139)
- [Cav 00] Xavier Cavin. *Simulation numérique parallèle et graphisme haute-performance pour la synthèse d'images réalistes*. Thèse de Doctorat, Institut National Polytechnique de Lorraine, 2000. (p. 156)
- [Caz 94] D. Cazier, D. Chamont, P. Deville et J. C. Paul. Modeling characteristics of light: A method based on measured light. *Pacific Conference on Computer Graphics and Applications*. International Academic Publishers, 1994. (p. 4)
- [Cha 97] David Chamont. *Architecture logicielle pour les méthodes de radiosit e*. Thèse de Doctorat, Universit e Henri Poincar e, Vandœuvre-l es-Nancy, France, November 1997. (p. 4)
- [Cha 98] Alan Chalmers et Erik Reinhard. Parallel and distributed photo-realistic rendering. *ACM SIGGRAPH '98 Course Notes - Course*, 1998. (p. 156)
- [Che 91] Shenchang Eric Chen, Holly E. Rushmeier, Gavin Miller et Douglass Turner. A Progressive Multi-Pass Method for Global Illumination. *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 164–174, July 1991. (p. 3)
- [Chr 94] Per H. Christensen, Eric J. Stollnitz, David H. Salesin et Tony D. DeRose. Wavelet Radiance. *Fifth Eurographics Workshop on Rendering*, pages 287–302, Darmstadt, Germany, June 1994. (p. 3)
- [Chr 96] Per H. Christensen, Eric J. Stollnitz, David H. Salesin et Tony D. DeRose. Global Illumination of Glossy Environments Using Wavelets and Importance. *ACM Transactions on Graphics*, 15(1):37–71, January 1996.
- [Chu 92] C. K. Chui. *An Introduction to Wavelets*, volume 1, s erie *Wavelet Analysis and its Applications*. Academic Press, Inc., 1992. (p. 37)

- [Coh 85] Michael Cohen et Donald P. Greenberg. The Hemi-Cube: A Radiosity Solution for Complex Environments. *Computer Graphics (ACM SIGGRAPH '85 Proceedings)*, volume 19, pages 31–40, August 1985.
- [Coh 93] Michael F. Cohen et John R. Wallace. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Boston, MA, 1993. (pp. 6, 20, 90)
- [Coo 82] R. Cook et K. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1(1):7–24, 1982. (p. 19)
- [Coo 94] R. Cook et K. Torrance. An inexpensive brdf model for physically-based rendering. *Computer Graphics Forum (Eurographics '94)*, 13(3):233–246, September 1994. (p. 19)
- [Cun 98a] Francois Cuny, Laurent Alonso, Christophe Winkler et Nicolas Holzschuch. Gestion efficace des polygones complexes pour la radiosit . *6 me Journ es de l'Association Fran aise d'Informatique Graphique, Dunkerque*. AFIG, d cembre 1998.
- [Cun 98b] Francois Cuny, Christophe Winkler, Laurent Alonso et Jean-Claude Paul. Wavelet Algorithms for Complex Models. Rapport de recherche, 1998. (p. 31)
- [Cun 99] Francois Cuny, Laurent Alonso, Christophe Winkler et Nicolas Holzschuch. Radiosite   base d'ondelettes sur des mailles quelconques. *Revue internationale de CFAO et d'informatique graphique*, 14(1):41–56, octobre 1999.
- [Cun 00] Francois Cuny, Laurent Alonso et Nicolas Holzschuch. A novel approach makes higher order wavelets really efficient for radiosity. volume 19, 2000. Available from <http://www.loria.fr/holzschu/Publications/paper20.pdf>. (p. 38)
- [Dau 92] Ingrid Daubechies. *Ten Lectures on Wavelets*, volume 61, s rie *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, 1992. (p. 37)
- [dB 92] M. de Berg et M. Overmars. Hidden surface removal for *c*-oriented polyhedra. *cga*, 1(5):247–268, 1992. (p. 5)
- [dB 94] M. de Berg, M. de Groot et M. Overmars. New results on binary space partitions in the plane. *Proceedings of 4th Scandinavian Workshop on Algorithm Theory*, volume 824, s rie *lncs*, pages 61–72. Springer-Verlag, 1994.
- [dB 95] M. de Berg. Linear size binary space partitions for fat objects. *Proceedings of 3rd ESA (Annual European Symposium on Algorithms)*, volume 979, s rie *lncs*, pages 252–263. Springer-Verlag, 1995.
- [Dev 92] Olivier Devillers, Monique Teillaud et Mariette Yvinec. Dynamic location in an arrangement of line segments in the plane. *Algorithms Review*, 2(3):89–103, 1992. (p. 105)
- [Dev 95] Pascal M. Deville et Jean-Claude Paul. Modelling the Spatial Energy Distribution of Complex Light Sources for Lighting Engineering. P. M. Hanrahan et W. Purgathofer,  diteurs, *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, pages 147–159, New York, NY, 1995. Springer-Verlag. (p. 4)
- [Dev 96] P. Deville. *Mod lisation et Simulation Des Propri tes Radiatives Des Sources Lumineuses*. Th se de Doctorat, Universite Henri Poincar , Nancy, France, 1996. (p. 4)

- [Dur 99a] Fredo Durand. *3D Visibility: Analytical Study and Applications*. Thèse de Doctorat, Université Joseph Fourier, Grenoble, France, July 1999.
- [Dur 99b] Fredo Durand, George Drettakis et Claude Puech. Fast and accurate hierarchical radiosity using global visibility. *ACM Transactions on Graphics*, 18(2) :128–170, 1999.
- [Ede 87] Herbert Edelsbrunner. *Algorithms in Combinatorial Geometry*, volume 10, série *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, novembre 1987. (p. 101)
- [Fol 90] James D. Foley, Andries van Dam, Steven K. Feiner et John F. Hughes. *Computer Graphics, Principles and Practice, Second Edition*. Addison-Wesley, Reading, Massachusetts, 1990. (p. 2)
- [Fun 96] Thomas A. Funkhouser. Coarse-Grained Parallelism for Hierarchical Radiosity Using Group Iterative Methods. *Computer Graphics Proceedings, Annual Conference Series, 1996 (ACM SIGGRAPH '96 Proceedings)*, pages 343–352, 1996. (pp. 145, 156)
- [Gib 96] Simon Gibson et R. J. Hubbold. Efficient hierarchical refinement and clustering for radiosity in complex environments. *Computer Graphics Forum*, 15(5) :297–310, December 1996. (pp. 46, 141)
- [Gla 95] Andrew S. Glassner. *Principles of Digital Image Synthesis*. Morgan Kaufmann, San Francisco, CA, 1995.
- [Gol 87] J. Goldsmith et J. Salmon. Automatic creation of object hierarchies for ray tracing. 1987. (p. 157)
- [Gor 84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg et Bennett Battaille. Modelling the Interaction of Light Between Diffuse Surfaces. *Computer Graphics (ACM SIGGRAPH '84 Proceedings)*, volume 18, pages 212–222, July 1984. (pp. 2, 7, 26, 85)
- [Gor 93] Steven J. Gortler, Peter Schroder, Michael F. Cohen et Pat Hanrahan. Wavelet Radiosity. *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 221–230, 1993. (pp. 6, 7, 36, 46, 49, 62, 79, 85, 90)
- [Gor 95] Steven J. Gortler. *Wavelet Methods for Computer Graphics*. Ph.D. thesis, Technical Report, Princeton, NJ, January 1995.
- [Gou 71] H. Gouraud. Continuous shading of curved surfaces. *IEEE Transactions on Computers*, C-20(6) :623–629, juin 1971. (p. 2)
- [Han 91] Pat Hanrahan, David Salzman et Larry Aupperle. A Rapid Hierarchical Radiosity Algorithm. *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 197–206, July 1991. (pp. 6, 34, 45, 49, 65, 85)
- [Has 99] J. M. Hasenfratz, C. Domez, F. Sillion et G. Drettakis. A practical analysis of clustering strategies for hierarchical radiosity. *Computer Graphics Forum (Proc. Eurographics '99)*, volume 18, pages C-221–C-232, September 1999. (pp. 94, 157)
- [He 91] Xiao D. He, Kenneth E. Torrance, Francois X. Sillion et Donald P. Greenberg. A Comprehensive Physical Model for Light Reflection. *Computer Graphics*

- (*ACM SIGGRAPH '91 Proceedings*), volume 25, pages 175–186, July 1991. (p. 19)
- [Hec 91] Paul Heckbert et J. M. Winget. Finite Element Methods for Global Illumination. Technical Report no. UCB/CSD 91/643, University of California at Berkeley, Berkeley, CA, 1991. (p. 3)
- [Hec 92] Paul Heckbert. Discontinuity Meshing for Radiosity. *Third Eurographics Workshop on Rendering*, pages 203–226, Bristol, UK, May 1992. (pp. 48, 103)
- [Hol 94] Nicolas Holzschuch, Francois Sillion et George Drettakis. An Efficient Progressive Refinement Strategy for Hierarchical Radiosity. *Fifth Eurographics Workshop on Rendering*, pages 343–357, Darmstadt, Germany, June 1994. (pp. 46, 49)
- [Hol 98] Nicholas Holzschuch et Francois. X. Sillion. An exhaustive error-bounding algorithm for hierarchical radiosity. *Computer Graphics Forum*, 17(4):197–218, December 1998. (p. 46)
- [Hol 00] Nicolas Holzschuch, François Cuny et Laurent Alonso. Wavelet radiosity on arbitrary planar surfaces. *Proc. of the 11th Eurographics Workshop on Rendering*, Brno, Czech Republic, 2000.
- [Imm 86] Dave S. Immel, Michael Cohen et Donald P. Greenberg. A Radiosity Method for Non-Diffuse Environments. *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, volume 20, pages 133–142, August 1986. (p. 19)
- [Kaj 86] James T. Kajiya. The Rendering Equation. *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, volume 20, pages 143–150, August 1986. (pp. 2, 24)
- [Kap 84] W. Kaplan. *Advanced Calculus*. Addison-Wesley, 1984. (p. 96)
- [Lac 99] B. Lacolle et N. Szafran. Approximating parametric surfaces by means of quadric patches. *Proc. of Fourth International Conference on Curves and Surfaces*, St. Malo, France, 1999. (pp. 116, 136, 162)
- [Law 89] D. Lawden. *Elliptic Functions and Applications*, volume 80, série *Applied mathematical sciences*. Springer-Verlag, 1989. (p. 170)
- [Lis 94] Dani Lischinski, Brian Smits et Donald P. Greenberg. Bounds and Error Estimates for Radiosity. *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 67–74, 1994. (p. 45)
- [Mal 89] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989. (p. 37)
- [Men 98] D. Meneveaux, K. Boauatouch, E. Maisel et R. Delmont. A new partitioning method for architectural environments. *Journal of Visualization and Computer Animation*, 9(4):195–213, October/November 1998. (p. 156)
- [Mer 97] Slimane Merzouk. *Architecture Logicielle et Algorithmes Pour la Resolution de L'Equation de Radiance*. Thèse de Doctorat, Institut National Polytechnique de Lorraine, 1997. (pp. 5, 6, 25, 31, 51, 61)
- [Nis 85] Tomoyuki Nishita et Eihachiro Nakamae. Continuous Tone Representation of Three-Dimensional Objects Taking Account of Shadows and Interreflection.

- Computer Graphics (ACM SIGGRAPH '85 Proceedings)*, volume 19, pages 23–30, July 1985.
- [Nou 80] B. Nourse, D. Hakala, R. Hillyard et P. Malraison. Natural quadrics in mechanical design. *Autofact West*, 1:363–378, 1980. (p. 89)
- [Ort 96a] Rachel Orti, Stephane Riviere, Fredo Durand et Claude Puech. Radiosity for Dynamic Scenes in Flatland with the Visibility Complex. *Computer Graphics Forum*, volume 15, pages C237–C248, September 1996. (p. 6)
- [Ort 96b] Rachel Orti, Stephane Riviere, Fredo Durand et Claude Puech. Using the Visibility Complex for Radiosity Computation. *Lecture Notes in Computer Science (Applied Computational Geometry: Towards Geometric Engineering)*, volume 1148, pages 177–190, Berlin, Germany, May 1996. Springer-Verlag.
- [Pau 95] J. C. Paul, P. M. Deville et C. Winkler. Modelling radiative properties of light sources and surfaces. *Journal of Visualization and Computer Animation*, 6(4):231–240, October-December 1995. (p. 4)
- [Pho 75] Bui-Tuong Phong. Illumination for computer generated pictures. *CACM June 1975*, 18(6):311–317, 1975. (p. 2)
- [Pre 92] William H. Press, Saul A. Teukolsky, William T. Vetterling et Brian P. Flannery. *Numerical Recipes in C, 2nd. edition*. Cambridge University Press, 1992. (p. 91)
- [Rus 87] Holly E. Rushmeier et Kenneth E. Torrance. The Zonal Method for Calculating Light Intensities in the Presence of a Participating Medium. *Computer Graphics (ACM SIGGRAPH '87 Proceedings)*, volume 21, pages 293–302, July 1987. (p. 21)
- [Rus 93] Holly E. Rushmeier, Charles Patterson et Aravindan Veerasamy. Geometric Simplification for Indirect Illumination Calculations. *Proceedings of Graphics Interface '93*, pages 227–236, San Francisco, CA, May 1993. Morgan Kaufmann. (p. 94)
- [Sch 93] Peter Schroder, Steven J. Gortler, Michael F. Cohen et Pat Hanrahan. Wavelet Projections for Radiosity. *Fourth Eurographics Workshop on Rendering*, number Series EG 93 RW, pages 105–114, Paris, France, June 1993.
- [Sch 94a] Peter Schroder. *Wavelet Algorithms for Illumination Computations*. Ph.D. thesis, Technical Report, Princeton, NJ, November 1994. (pp. 6, 7, 38, 42, 85)
- [Sch 94b] Peter Schroder, Steven J. Gortler, Michael F. Cohen et Pat Hanrahan. Wavelet Projections for Radiosity. *Computer Graphics Forum*, 13(2):141–151, June 1994. (p. 90)
- [Sch 95] Christian Schwarz, Jürgen Teich, Alek Vainshtein, Emo Welzl et Brian L. Evans. Minimal enclosing parallelogram with application. *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, pages C34–C35, 1995. (p. 101)
- [Sch 97] Stephan Schaefer. Hierarchical radiosity on curved surfaces. Julie Dorsey et Phillip Slusallek, éditeurs, *Rendering Techniques '97 (Proceedings of the Eighth Eurographics Workshop on Rendering)*, pages 187–192, New York, NY, 1997. Springer Wien. ISBN 3-211-83001-4. (p. 93)

- [Shi 94] Peter Shirley et Changyaw Wang. Direct Lighting Calculation by Monte Carlo Integration. P. Brunet et F. W. Jansen, éditeurs, *Photorealistic Rendering in Computer Graphics (Proceedings of the Second Eurographics Workshop on Rendering)*, pages 54–59, New York, NY, 1994. Springer-Verlag. (p. 3)
- [Sil 89] Francois Sillion et Claude Puech. A General Two-Pass Method Integrating Specular and Diffuse Reflection. *Computer Graphics (ACM SIGGRAPH '89 Proceedings)*, volume 23, pages 335–344, July 1989. (p. 3)
- [Sil 91] Francois Sillion, James R. Arvo, Stephen H. Westin et Donald P. Greenberg. A Global Illumination Solution for General Reflectance Distributions. *Computer Graphics (ACM SIGGRAPH '91 Proceedings)*, volume 25, pages 187–196, July 1991. (p. 19)
- [Sil 94a] Francois Sillion. Clustering and Volume Scattering for Hierarchical Radiosity Calculations. *Fifth Eurographics Workshop on Rendering*, pages 105–117, Darmstadt, Germany, June 1994. (p. 36)
- [Sil 94b] Francois Sillion et Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann, San Francisco, CA, 1994. (pp. 6, 20, 90)
- [Sil 95a] Francois Sillion. A Unified Hierarchical Algorithm for Global Illumination with Scattering Volumes and Object Clusters. *IEEE Transactions on Visualization and Computer Graphics*, 1(3), September 1995. (pp. 94, 157)
- [Sil 95b] Francois Sillion, George Drettakis et Cyril Soler. A Clustering Algorithm for Radiance Calculation in General Environments. P. M. Hanrahan et W. Purgathofer, éditeurs, *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, pages 196–205, New York, NY, 1995. Springer-Verlag. (p. 157)
- [Smi 94] Brian Smits, James Arvo et Donald Greenberg. A Clustering Algorithm for Radiosity in Complex Environments. *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 435–442, 1994. (pp. 35, 94, 157)
- [Sta 97a] M. Stamminger, P. Slusallek et H.-P. Seidel. Bounded radiosity - illumination on general surfaces and clusters. *Computer Graphics Forum (Eurographics '97 Proceedings)*, 16(3):C309–C317, 1997. Available from <http://www9.informatik.uni-erlangen.de/eng/research/pub1997>. (pp. 46, 93)
- [Sta 97b] M. Stamminger, P. Slusallek et H.-P. Seidel. A comparison of different refiners for hierarchical radiosity. B. Girod, H. Niemann et H.-P. Seidel, éditeurs, *3D Image Analysis and Synthesis '97*, 1997. Available from <http://www9.informatik.uni-erlangen.de/eng/research/tr1997>. (p. 54)
- [Sta 98] M. Stamminger, H. Schirmacher, P. Slusallek et H.-P. Seidel. Getting rid of links in hierarchical radiosity. *Computer Graphics Journal (Proc. Eurographics '98)*, 17(3):C165–C174, September 1998. (pp. 51, 61, 73)
- [Sta 99] Marc Stamminger, Annette Scheel, Xavier Granier, Frederic Perez-Cazorla, George Drettakis et Francois Sillion. Efficient glossy global illumination with interactive viewing. *Graphics Interface '99*, San Francisco, CA, June 1999. Morgan Kaufmann.

- [Sut 74] Ivan E. Sutherland, Robert F. Sproull et Robert A. Schumacker. A characterization of ten hidden-surface algorithms. *ACM Computing Surveys*, 6(1):1–55, mars 1974. (p. 2)
- [Tel 94] Seth Teller, Celeste Fowler, Thomas Funkhouser et Pat Hanrahan. Partitioning and Ordering Large Radiosity Computations. *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 443–450, 1994. (pp. 145, 156)
- [Tro 93] Roy Troutman et Nelson L. Max. Radiosity Algorithms Using Higher Order Finite Element Methods. *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 209–212, 1993. (p. 90)
- [Wal 87] John R. Wallace, Michael F. Cohen et Donald P. Greenberg. A Two-Pass Solution to the Rendering Equation : A Synthesis of Ray Tracing and Radiosity Methods. *Computer Graphics (ACM SIGGRAPH '87 Proceedings)*, volume 21, pages 311–320, July 1987. (pp. 3, 19)
- [War 92] Gregory J. Ward. Measuring and Modeling Anisotropic Reflection. *Computer Graphics (ACM SIGGRAPH '92 Proceedings)*, volume 26, pages 265–272, July 1992. (p. 19)
- [Wil 97a] Andrew Willmott et Paul Heckbert. An empirical comparison of progressive and wavelet radiosity. Julie Dorsey et Phillip Slusallek, éditeurs, *Rendering Techniques '97 (Proceedings of the Eighth Eurographics Workshop on Rendering)*, pages 175–186, New York, NY, 1997. Springer Wien. ISBN 3-211-83001-4. (pp. 43, 50, 51, 74, 75, 76, 80, 138)
- [Wil 97b] Andrew J. Willmott et Paul S. Heckbert. An empirical comparison of radiosity algorithms. Rapport no. CMU-CS-97-115, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, April 1997. Available from <http://www.cs.cmu.edu/radiosity/emprad-tr.html>. (pp. 6, 42, 43, 80)
- [Wil 99] Andrew Willmott, Paul Heckbert et Michael Garland. Face cluster radiosity. *Eurographics Workshop on Rendering*, June 1999. (p. 157)
- [Win 98] Christophe Winkler. *Expérimentation d'algorithmes de calcul de radiosité à base d'ondelettes*. Thèse de Doctorat, Institut National Polytechnique de Lorraine, 1998. (pp. 4, 5, 6, 13, 31, 51, 52, 61)
- [Zat 93] Harold R. Zatz. Galerkin Radiosity : A Higher Order Solution Method for Global Illumination. *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pages 213–220, 1993. (pp. 6, 7, 37, 49, 85, 90)

**AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL
POLYTECHNIQUE DE LORRAINE**

o0o

VU LES RAPPORTS ETABLIS PAR :

Monsieur KIRCHNER Claude, Directeur de Recherche, LORIA/UHP Nancy I,

**Monsieur SILLION François, Directeur de Recherche, IMAGIS/GRAVI/IMAG/ZIRST
Montbonnot,**

Monsieur GUITTON Pascal, Professeur, LABRI/Université de Bordeaux I.

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

Monsieur CUNY François

à soutenir devant un jury de l'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE,
une thèse intitulée :

"Radiosité à base d'ondelettes sur des surfaces paramétriques".

en vue de l'obtention du titre de :

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : **"INFORMATIQUE"**

Fait à Vandoeuvre, le 9 octobre 2000

Le Président de l'I.N.P.L.,

J. HARDY



NANCY BRABOIS
2, AVENUE DE LA
FORET-DE-HAYE
BOITE POSTALE 3
F - 5 4 5 0 1
VANDOEUVRE CEDEX

Résumé

Dans cette thèse, nous introduisons une méthode générale pour effectuer le calcul de radiosité dans des scènes constituées d'une grande variété de surfaces paramétriques. Contrairement aux approches précédentes qui nécessitaient une triangulation des surfaces en entrée, notre méthode bénéficie de la richesse de la représentation mathématique de la géométrie. Au coeur de notre méthode se trouve le *Maillage Virtuel*, une abstraction de la géométrie initiale permettant d'appliquer les algorithmes classiques d'illumination à des surfaces complexes exactement comme si ces surfaces étaient de simples triangles. Le maillage virtuel est constitué d'un ensemble de domaines normalisés sur lesquels les surfaces sont projetées grâce à une fonction permettant de préserver leurs propriétés énergétiques. Les valeurs de radiosité sont calculées sur ces domaines normalisés avant d'être re-projetées sur les surfaces originales.

Nous commençons par démontrer l'efficacité des fonctions ondelettes d'ordre élevé quand elles sont utilisées au sein d'un algorithme cohérent prévu pour gérer l'aspect fonctionnel de la fonction de radiosité. Ensuite, nous décrivons l'implantation dans cet algorithme du maillage virtuel pour les primitives planes à bords quelconques et les morceaux de surfaces quadriques. Nous donnons des exemples d'utilisation de notre méthode sur divers environnements de réalité virtuelle. Grâce à ces exemples nous montrons qu'en intégrant de façon exacte les surfaces courbes dans le processus de résolution, le maillage virtuel permet d'obtenir plus rapidement un rendu, de meilleure qualité et de façon plus naturelle qu'avec les méthodes précédentes.

Mots-clés: Synthèse d'images, illumination globale, radiance, radiosité, ondelettes, surfaces paramétriques, quadriques, maillage virtuel.

Abstract

In this thesis, we introduce a general-purpose method for computing radiosity on scenes made of a large variety of parametric surfaces (including arbitrary planar primitives and low-degree curved patches). By contrast with past approaches that require a tessellation of the input surfaces, our method takes advantage of the rich and compact mathematical representation of objects. At its core lies the *virtual mesh*, an abstraction of the input geometry that allows complex shapes to be illuminated as if they were simple primitives. The virtual mesh is a collection of normalized square domains to which the input surfaces are mapped while preserving their energetic properties. Radiosity values are then computed on these supports before being lifted back to the original surfaces.

To demonstrate the power of our method, we describe a high-order wavelet radiosity implementation that uses the virtual mesh. Before this description we show that, with an appropriated implementation, higher order wavelets are actually bringing an improvement over standard hierarchical radiosity and lower order wavelets. Then we describe the implementation of the virtual mesh on arbitrary planar primitives and quadric surface patches. Examples of objects and environments, designed for interactive applications or virtual reality, are presented. They prove that, by exactly integrating curved surfaces in the resolution process, the virtual mesh allows complex scenes to be rendered more quickly, more accurately and much more naturally than with previously known methods.

Keywords: Image synthesis, global illumination, radiance, radiosity, wavelets, parametric surfaces, quadrics, virtual mesh.