



**HAL**  
open science

# Improving users' product acceptability: an approach based on Bayesian networks and a simulated annealing algorithm

Giovanny Arbelaez Garces, Auguste Rakotondranaivo, Eric Bonjour

## ► To cite this version:

Giovanny Arbelaez Garces, Auguste Rakotondranaivo, Eric Bonjour. Improving users' product acceptability: an approach based on Bayesian networks and a simulated annealing algorithm. *International Journal of Production Research*, 2016, 54 (17), pp.5151-5168. 10.1080/00207543.2016.1156183 . hal-01522501

**HAL Id: hal-01522501**

**<https://hal.univ-lorraine.fr/hal-01522501>**

Submitted on 8 Apr 2019

**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.

## Title:

Improving users' product acceptability: an approach based on Bayesian networks and a Simulated annealing algorithm

## Authors:

Giovanny Arbelaez Garces<sup>1\*</sup>, Auguste Rakotondranaivo<sup>1</sup>, Eric Bonjour<sup>1</sup>

## Institutions and address

<sup>1</sup> Université de Lorraine / ENSGSI, ERPI, EA 3767,

8, rue Bastien Lepage, Nancy, 54010, France

{giovanny.arbelaez-garces} {at} univ-lorraine.fr

\*Corresponding author

## Acknowledgement

This work is financed by a grant (number CT65092012SU) from the French Ministry of Higher Education and Research (Ministère de l'Enseignement supérieur et de la Recherche, MESR). The models described in this paper were created using the GeNIe modeling environment. The optimization algorithm uses the SMILE reasoning engine, both tools are developed by the Decision Systems Laboratory, University of Pittsburgh.

*\*This document is a post-print authors' version for a final version please visit editors site.*

*You should cite this document as:*

Giovanny Arbelaez Garces, Auguste Rakotondranaivo, Eric Bonjour. Improving users' product acceptability: an approach based on Bayesian networks and a simulated annealing algorithm. *International Journal of Production Research*, Taylor & Francis, 2016, 54 (17), pp.5151-5168. [10.1080/00207543.2016.1156183](https://doi.org/10.1080/00207543.2016.1156183) . [hal-01522501](https://hal.archives-ouvertes.fr/hal-01522501)

# Improving users' product acceptability: an approach based on Bayesian networks and a simulated annealing algorithm

## Abstract

Developing products that are properly suited to users' needs and preferences in order to be accepted is one of the main challenges designers and engineers face constantly. Evaluating and improving users' product acceptability has become an important research question. Many approaches leave the acceptability evaluation question for the last phases of the New Product Development process (NPD), when an almost finished prototype is available and when there is no time left for important modifications. In the early phase of the NPD process, the project managers need models and methods to evaluate the potential acceptability of the new concept and if required, to define actions to improve this concept. In this paper, a method with two main goals is proposed to tackle this problem. Its first goal consists in evaluating an index of users' product acceptability. When this index is too low, the second goal concerns the optimal selection of the most appropriate actions (improvement scenario) to increase this previously assessed index and to optimize the supplementary cost. As information collected from users in the early phase is subject to uncertainty, the proposed method exploits the inference properties of Bayesian networks (BN) making it possible to make useful estimations of the acceptability index. Furthermore, the improvement scenarios are composed of actions that make it possible to improve different criteria composing the users' acceptability index. The improvement problem is formulated as an optimization problem to be solved by a simulated annealing (SA) algorithm. In order to illustrate its interest, the proposed method is applied to a real case concerning the design of a medical-stocking threading device.

**Keywords:** Acceptability evaluation; Project management; Optimization; Bayesian Networks; Simulated annealing.

## 1 Introduction

Identification and analysis of the needs and perceptions of the future users are fundamental in an innovation project. Therefore, the so-called 'user-centered' innovation approaches such as Livings Labs and generally all shared innovation systems (open innovation) promote the integration of the user through the design process (Chesbrough, 2003; Camargo et al., 2012). An increasing number of companies are adopting integrated co-design strategies in new product development to minimize the risk of their products not being accepted (Durugbo, 2014). This requires constructing and measuring user satisfaction (Jiang et al., 2015). In this perspective, acceptability estimation can be used for estimating the probability of success of the new concept (Luo, Su, and Lee, 2011). Identifying and estimating an acceptability index in the early phase of the development process (i.e. the conceptual design phase) requires appropriate methods and models. Methods for technology acceptability evaluation have been proposed (Davis, 1989; Venkatesh et al., 2003). They provide acceptability criteria but do not characterize the relationships that exist among criteria and with the global acceptability index. The project managers need to obtain a global vision of the different criteria that assess the potential success of the future product or system. After the concept generation activity, the stakeholders have to express their judgment about the new concept according to the identified criteria. The collected information is uncertain and incomplete because the test situation may be different to the future situation of usage, or the prototype does not match the final product properly (lack of fidelity or maturity). To tackle this problem concerning acceptability estimation, Bayesian networks seem to be appropriate modeling techniques since they provide visual representations of relationships and handle uncertain data. Moreover, they can be used according to two goals, prediction and diagnosis, that may be useful for the acceptability index estimation (prediction) and for the search for key criteria that greatly impact the acceptability index (diagnostic). When the project manager and the design team judge that the computed acceptability index is low, they have to determine a set of actions that may positively increase the users' perception of the new concept, that is, increase the value of concerned criteria. The increase in acceptability criteria will increase the acceptability index. The acceptability estimation with Bayesian networks should be useful to compute the impact of different sets of actions (improvement scenario) on the acceptability index. Determining the optimal list of actions that will enable to increase the acceptability index up to a target level while minimizing the cost of required actions is not a straightforward task because it may be difficult to find the right balance among the possible actions, their associated cost of performance (material, human and technical resources) and their effect over the acceptability index

(Alsawalqah, Kang, and Lee, 2014). This optimization problem is equivalent to the selection of  $k$  actions in a set of  $n$  actions and then the computation of the new acceptability index. Its complexity depends on the sum of the number of  $k$ -combination  $C_n^k$ , that is  $\sum_{0 \leq k \leq n} \binom{n}{k} = 2^n$ . When  $n$  increases, the running time of an enumeration procedure will drastically increase. A metaheuristic is useful for solving large problems ( $n > 20$  actions). Among the metaheuristics, Simulated annealing is appropriate to develop a heuristic to solve this problem. It is easy to find an initial (certainly bad) solution for this problem that is to select all actions and to define neighbors by adding or removing an action from the current combination. While this technique is unlikely to find the optimum solution, it can often find a very good solution. Consequently, this paper proposes a method to solve the product acceptability improvement problem. This method has two goals: assessing the design acceptability index and optimizing the selection of the most appropriate actions to improve the previously assessed index. The proposed optimization algorithm helps to select a combination of actions that improves the acceptability index to the desired level while controlling the associated cost of execution. The remainder of the paper is structured as follows. In section 2, we present an overview of the methods for acceptability evaluation, the properties related to Bayesian networks and the simulated annealing algorithm. Section 3 describes the proposed method. Section 4 presents a case study. We discuss a comparison of the proposed algorithm with other optimization algorithms in section 5. In section 6 we discuss the implications and limits of the proposed method. Finally, conclusions and directions for further research are presented in section 7.

## 2 Related Work

### 2.1 Acceptability evaluation

Several approaches concerning acceptability evaluation are discussed in the literature. There are a few that deal with the problem of using subjective and unconscious feelings about a product to translate them into concrete product attributes (Luo, Su, and Lee, 2011; Camargo, Wendling, and Bonjour, 2014). The latter for instance proposed an approach to integrate user’s perception and identify the importance and interaction of semantic attributes thus allowing designers to validate and adjust product concepts. Another kind of approach is related to the users’ needs assessment process. The proposed method based on Kano model presented in (Ben Rejeb, Boly, and Morel-Guimaraes, 2008), can be used as a decision aid tool for selecting consumer needs. Other approaches are based on the theory of technology acceptance, in which the users’ behavioral intention is used as a measure of acceptability. They derive from Ajzen and Fishbein’s Theory of Reasoned Action (Ajzen and Fishbein, 1980). The most influential from this trend is the Technology Acceptance Model (TAM) (Davis, 1989). This theory states that a user’s behavioral intention for a new technology is influenced by its perceived usefulness (PU) and its perceived ease of use (PEOU). PU was defined by the author as ‘the degree to which a person believes that using a particular system would enhance his or her job performance’ and PEOU as ‘the degree to which a person believes that using a particular system would be free from effort’. The TAM model was expanded first as TAM2 (Venkatesh and Davis, 2000) to include other factors influencing the PU. Factors from the organizational context of the user (social influence and facilitating conditions), were also added to the model that was renamed unified theory of acceptance and use of technology (UTAUT) (Venkatesh et al., 2003). Conversely, Nielsen’s model (Nielsen, 1993), distinguishes the acceptability factors of a system between two categories: Social acceptability and Practical acceptability. The former refers to the system’s compliance with social needs, whereas the latter is related to the technical environment of the proposed system (reliability, compatibility) as well as aspects related to its use (utility, usability). It evolved as the ISO 9241-210:2010 norm and could be used as a prescriptive tool in product design.

The previously described models lack of a vision of the interaction among their different criteria. Furthermore, those models are mostly used as measurement mechanisms once the technology has been deployed, thus requiring a finished prototype and a huge amount of user testing. As prototypes and users’ test data are scarce in the first stages of product development, some artificial intelligence techniques can help to deal with those drawbacks. There have been some proposals for evaluating and predicting acceptability using some of those techniques. Elazouni, Ali, and Abdel-Razek (2005), estimated the acceptability of a new formwork system for construction using neuronal networks. Luo, Su, and Lee (2011) proposed a combination of techniques (Radial Basis Function Networks, Support Vector Machine–Sequential Minimal Optimization and their ensembles). Along with Bayesian networks to predict acceptability of products, they tested the method with a car evaluation. Their method, however, requires large data-sets, making it suitable for design cases where previous projects (and user data) exist but is limited for radical innovation cases where no previous information is available.

The base of our proposed method rests on the usage of Bayesian networks and its inference properties. Compared to other techniques (neural networks, support vector machines, decision trees) a Bayesian network makes it possible to integrate different sources of information (experts’ knowledge, and data). It allows uncertainty and incomplete data to be handled. Acceptability evaluation allows designers to anticipate and act accordingly to avoid project risks. Work has already been proposed in this direction mainly in the health-care sector (Arbelaez-garces, Bonjour, and

Rakotondranaivo, 2014). However the optimization step to find the appropriate combination of actions for improving the product’s acceptability was not addressed. Therefore, this paper aims to present a method for acceptability evaluation that allows its initial level to be assessed and helps to identify and select the levers for acceptability that will improve the product design. More than calculating an exact level of acceptability our approach intends to encourage a systematic way of thinking in the design team helping them to ask the right questions and to find the solutions for improving the product’s acceptability.

## 2.2 Reminder on Bayesian Networks

Bayesian networks (BN) are directed acyclic graphs (DAG) that represent the probabilistic relationships between a group of variables (Pearl, 1988). They are generally composed of two parts: a quantitative part and a qualitative part. The qualitative part corresponds to the nodes that represent the states of random variables and the arcs pointing from a parent node to a child node represent the causal conditional dependency between the two nodes. The quantitative part refers to the relationship between a node and its parents. The latter can be represented by the probability of the node’s state provided by different probabilities of the parent’s node state. The former can be represented by a conditional probability table (CPT) of the node. Two types of inference reasoning are possible with Bayesian network: (1) diagnosis (backwards inference) which, given an observation, makes it possible to find the most probable cause among the hypotheses, and (2) prediction (top-down inference), which makes it possible to estimate the probability of an observation, given the assumptions (Jensen and Nielsen, 2007). The former inferences can be drawn when the BN is constructed. One of the algorithms available in the literature (Huang and Darwiche, 1996; Pearl, 1988; Jensen and Nielsen, 2007) can be used. When the BN is not constructed, there are algorithms for parameter learning to estimate the conditional probability laws assuming that the structure was fixed using experts’ knowledge (Dempster, Laird, and Rubin, 1977). Otherwise, structure learning algorithms determine the structure and estimate the CPT (Jensen and Nielsen, 2007). The inference properties we have described on Bayesian networks allow us to calculate and predict the acceptability level of a technology.

## 2.3 Optimization and Simulated Annealing

The analysis of the product acceptability improvement problem, and model development, formulates a combinatorial optimization problem. Small-sized optimization problems are easily solved by Exact Algorithms (EA) that explores all the possible combinations. For large-sized problems, those techniques suffer from exponential algorithmic complexity, thus requiring huge amounts of time to find the optimal combination. Heuristic-based optimization algorithms can find solutions with acceptable optimality (sub-optimal solutions) and can easily overcome those drawbacks. However, to our knowledge, there is no work providing an approximation algorithm for the problem of selecting the best actions to improve a product’s acceptability. Simulated Annealing (SA) (Kirkpatrick, Gelatt, and Vecchi, 1983) is a robust technique that provides excellent solutions to single and multiple objective optimization problems with a substantial reduction in computation time. SA is widely used for large combinatorial optimization problems. A global optimum is guaranteed if sufficiently large numbers of iterations are run. The underlying idea of the heuristic of SA arises from an analogy with certain thermo-dynamical processes (cooling of a melted solid). Kirkpatrick, Gelatt, and Vecchi (1983) pioneered its use for combinatorial problems. For a generic problem of the form  $\min F(x) \text{ s.t. } x \in X$ ; the basic principle of the SA heuristic can be described as follows. Starting from a current solution  $x$ , another solution  $y$  is generated by taking a stochastic step in some neighborhood of  $x$ . If this new proposal improves the value of the objective function, then  $y$  replaces  $x$  as the new current solution. Otherwise, the new solution  $y$  is accepted with a probability that decreases with the magnitude of the deterioration and in the course of iterations. The probability term is generally taken to be the Boltzman function of the form  $e^{-\Delta F(x)/T}$ , where  $T$  is the current temperature at the current step and  $\Delta F(x)$  is the difference between the current solution and the best solution so far. The reader can explore further concerning the SA technique in (Pirlot, 1996). SA has already been shown to be useful for optimization problems in engineering design (Lee and Wang, 1992) as well as in risk assessment (Bouaziz, Zamaï, and Duvivier, 2013). The idea of using a SA to optimize the options or scenarios provided by a Bayesian network has already been explored by Lu and Liechty (2007), whose work compared a linear programming and SA approach for a Bayesian portfolio selection. However, to our knowledge, there are no works exploring the problem of optimizing the actions for the improvement of product acceptability.

## 3 Proposed method

The proposed method is useful in the conceptual design phase activity and before the product development phase (Aoussat, Christofol, and Le Coq, 2000; Ullman, 1994). After the concept generation activity there are some studies concerning the feasibility from the technical and economical points of view. Our proposal will help the project manager

to complete these studies with the assessment of the users' product acceptability and if needed, the determination of actions that could improve the assessed acceptability level. Thus the aim of the proposed method is twofold: (i) to contribute to the concept validation process by estimating the acceptability of a new concept and (ii) to identify the optimal combination of actions to improve it (actions that have to be achieved in the product development phase). The first step of the method is to build a Bayesian Network (BN) to form the acceptability evaluation model. The second step consists in using this BN to make the initial acceptability assessment. Once the initial assessment is made, the Potential Acceptability Index (PAI) is generated. According to the initial PAI the project manager will decide whether improvement actions are required or not. If required, the third step recommends that the project manager evaluates different set of actions (improvement scenarios) and then select the ones that are the most suited to improve the users' product acceptability. Figure 1 shows these three steps that will be further explained in the next sections: the construction of the Bayesian Network, the initial acceptability assessment and acceptability improvement.

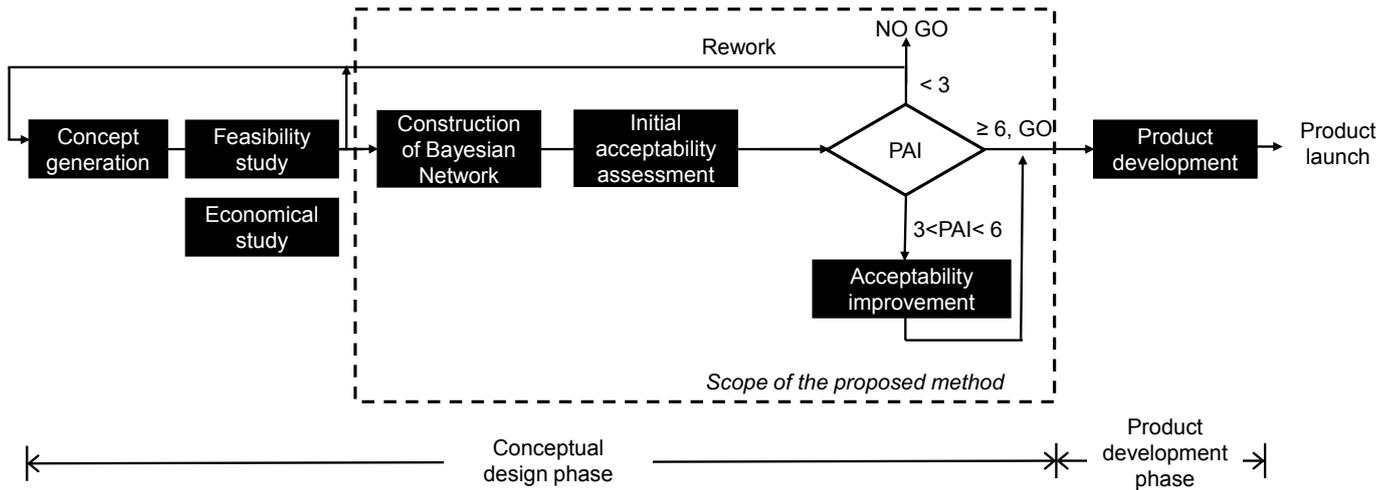


Figure 1: Overview of the method.

### 3.1 Construction of Bayesian Network

The first phase concerns the construction of the Bayesian network for evaluating the acceptability level. First the criteria for evaluation are identified and the structure is defined, next we collect the data to build the conditional probabilities of the BN and lastly the data is used for the acceptability assessment.

#### 3.1.1 Acceptability criteria identification

In this step we identify the criteria that are going to be evaluated for estimating the users' perceived acceptability of the product. There are different user acceptance models like TAR, TAM, TAM2 or UTAUT (Ajzen and Fishbein, 1980; Davis, 1989; Venkatesh et al., 2003; Venkatesh and Davis, 2000). Caine et al. (2006) analyzed the works applying those different models during a 50 years period ending in 2006. From their research they listed the most important criteria linked to product acceptance. They suggested that designers should take into account those criteria (cf. Table 1) during the product design to increase the chance of the product being accepted. We propose to use Table 1 as a framework for criteria selection. This list of generic criteria can be completed with criteria pertaining to the domain of application. The model has to be revised for each project, because, if the perceived privacy category might apply for a web application development, it could not be the case for a mechanical tool that does not collect any data. The concept generation phase can also help to define more specific criteria once the concept solution takes form.

Each criteria category has to be decomposed in measurable items. As an example for a new medical diagnosis application, two constructs explain the intention to use: perceived ease of use and perceived utility (usefulness). One can define items to evaluate each one of those constructs. The utility of the medical application rests on its diagnosis and accuracy capability, so perceived utility can be composed by 'diagnosis is accurate' (users' perception of device diagnosis accuracy) and completed by other constructs.

The initial structure of the network is defined by linking the criteria to the acceptability index node. If there is previous knowledge that there is a link between two or more criteria, that could be represented in the BN by linking those criteria nodes. The project manager can build the Bayesian network using any software Bayesian networks modeling software (GeNIe®, Bayesialab®, Netica®...). Each criterion is represented as a node and the links between the criteria (edges) are fixed as to represent the network structure. In order to validate the completeness of the retained

Table 1: Acceptance and adoption criteria suggested by the work of (Caine et al., 2006)

Criteria type	Rationale	Reference
Perceived utility	Users needs had to be take into account.	(Davis, 1989)
Perceived ease of use	Products need to be tested with end users to validate the ease of use.	(Davis, 1989)
Perceived compatibility	Compatibility with the knowledge and current uses by the users.	(Moore and Benbasat, 1991)
Perceived complexity	Facilitate product understanding.	(Rogers, 1995)
Perceived enjoyment	Pleasure and enjoyment level associated with product use.	(Brosnan, 1999)
Perceived image	Status gain with product use.	(Moore and Benbasat, 1991)
Perceived novelty	Perception of the modernity of the product.	(Gruen, 1960)
Perceived privacy	Respect for private life. Collected data is used to enhance user experience.	(Luarn and Lin, 2005)
Perceived comparative advantage	Perceived improvements against other existing products.	(Dillon and Morris, 1999)
Perceived risk	Perceived risks associated with product usage.	(Featherman and Pavlou, 2003)

criteria and network it is necessary to verify that the interest and knowledge of every stakeholder involved in the project is taken into account.

### 3.1.2 Data collection

In this step a questionnaire is used to evaluate all the identified criteria in the model. Participants are invited to test the product prototype and to evaluate it as regards each criterion on a pre-defined scale. 5- to 7-level Likert scale with extremes from ‘totally disagree’ to ‘completely agree’ could be used (Sinclair, 2005; Likert, 1932). A global acceptability index corresponding to the PAI must be included in the evaluation to serve as a final utility score. To evaluate the criteria, in addition to the questionnaire, some objective measures can be used. They can be collected using physiological measurements like electromyography (EMG), galvanic skin response (GSR), electroencephalography (EEG), heart rate (HR), temperature, and respiration rate that can support the evaluation of the anxiety criterion for example (Silva et al., 2013; Zhou et al., 2011). The advantage of physiological measurements in comparison with cognitive ones (self-reports, questionnaires, etc.) is that they deliver objective results (Jenkins, Brown, and Rutterford, 2009).

### 3.1.3 Definition of the probability laws of the structure

Once the data from the test has been collected the project manager can enter the data to the software where the BN was modeled. The data from the questionnaire is linked to each corresponding node. We use a parameter learning algorithm (Dempster, Laird, and Rubin, 1977) to build the probability distributions for each variable from the collected data.

## 3.2 Initial acceptability assessment

When the probability laws for each node have been defined we can use them to make an inference with the Lauritzen algorithm (Huang and Darwiche, 1996). The inference provides us with the initial assessment of the acceptability level. If the PAI level is greater than a specified limit (6 for example) defined by the project manager, then the project can be launched and go to development. Otherwise, if the level is too low (below 3 for example) the manager has to decide whether to stop the project or to go back to the concept generation (rework) to improve the solution. If the level is medium (greater than 3 but lower than 6 for example) the acceptability improvement step enables to search

for improvement actions that can be put into practice during the product development phase to improve the product acceptability.

### 3.3 Acceptability improvement

The aim of this phase is to model the impact of a series of actions to improve acceptability criteria and define the optimal combination of actions that minimize the improvement effort while helping to reach an acceptability level that is fixed by the project manager.

#### 3.3.1 Characterization of actions and their impacts on the acceptability index

When aiming to improve the acceptability level we need to know which actions among all those possible are the best choice. Each action is defined by its impact to each criterion of the acceptability model and the effort required to be put in practice (here modeled as a performance cost in €). The first step is to identify all possible actions that can be put into practice to improve the product acceptability. This task is generally better suited to be made in consensus with the design group and the manager because it has a creativity problem-solving component. It is also possible to collect users' feedback from the product tests and to use this as improvement recommendations. The group has to list all the criteria from the first phase and to identify actions that can be put into practice during the product development stage to improve them. Once the first list of actions is identified each action has to be characterized by its associated performance cost and its impact.

When modeling the actions we made the following assumptions:

- The relationship between the actions and the criteria is assumed to be modeled as a delta of contribution of the former to the latter.
- The cost of an action is not tied to its impact on the acceptability, *i.e* the most expensive action is not necessarily the best one.
- There is an additive property in the effects of actions (*i.e* the effects of multiple actions are accumulated).

From the set of all described (n) actions  $A = \{a_1, \dots, a_n\}$ , we define the impact of action  $a_i$  as a contribution to each criterion  $(\Delta a_{i1}, \dots, \Delta a_{im})$ . We define three levels of positive impact (1, 2, 3) and three levels of negative impact (-1, -2, -3). The final value of one criteria  $C_j$  is defined by the equation  $newC_j = MIN(currentC_j + \sum_n \Delta a_{ij} \times x_i; Max\ value)$ ; where:

- $currentC_j$  is the current estimated value of criteria  $C_j$  before the execution of actions
- $\Delta a_{ij}$  is the impact the action  $a_i$  to the criteria  $j$
- $x_i$  is a decision variable that is set to 1 if the action  $i$  is chosen or 0 if not
- $Max\ Value$  is the maximum allowed value of the evaluation scale.

For example, if there are two actions  $a_1$  and  $a_2$ , the first step is to associate the corresponding cost to each action and then for each criterion  $C_j$  to define how much the action contributes to the improvement of the criteria (1, 2, 3, or -1, -2, -3 if the impact is negative). If the impact of  $a_1$  to  $C_j$  is 2 and the impact of  $a_2$  is 1, and if both actions are put into practice ( $x_1, x_2$  set to 1) and if the current value of  $C_j$  is 3 the final value will be 6. If  $C_j$  is 5,  $a_1$  is -1 and  $a_2$  is 1 the final value of  $C_j$  will be 5. Here note that the positive impact of  $a_2$  is eliminated by  $a_1$ .

#### 3.3.2 Optimization problem formulation

Once the entire set of actions has been defined the problem is to minimize the function:

$$\min \sum_i^I x_i \times cost_i; \quad (1)$$

Subject to constraints:

$$\begin{aligned} PAI &\geq \text{Acceptability limit} \\ \forall i = 1, \dots, I; x_i &= 1 \text{ or } 0; \end{aligned}$$

where  $x_i$  is the decision variable,  $cost_i$  execution cost of action  $i$ , PAI is the computed acceptability index corresponding to the combination of actions, and the acceptability limit is set to the minimum level tolerated by the decision makers. This limit allows the project manager to set the acceptability that he wants to reach. PAI is calculated for each combination of actions, in the same way than in the initial assessment phase. For each criterion its new value is calculated by the  $newC_j$  equation according to the active actions. The new criteria values are then propagated to the Bayesian network and the PAI value for the combination is calculated. Figure 2 represents the optimization problem.

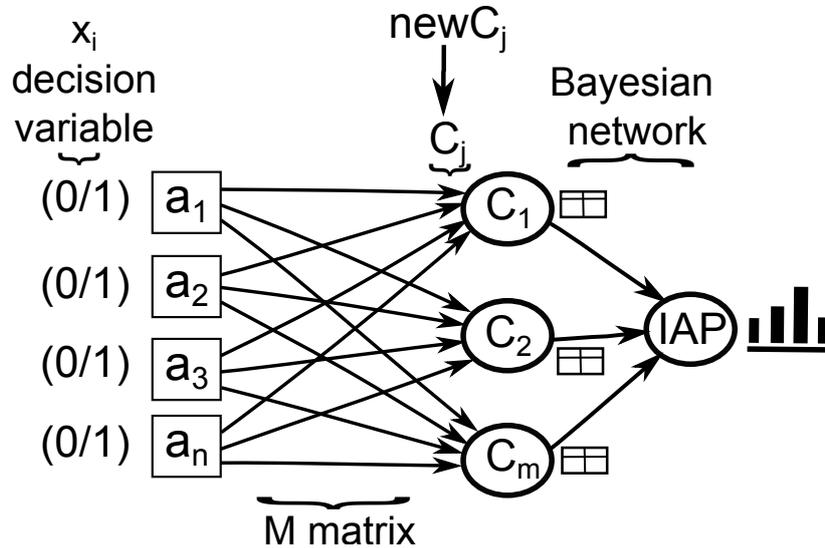


Figure 2: Optimization problem overview.

### 3.3.3 Optimization problem solving with Simulated Annealing

The analysis of the problem formulates a 0-1 integer, constrained, linear problem that can be solved using combinatorial optimization. However, the number of available actions determines the size of the solution space. Small-size problems can be solved using Exact Algorithms to explore the entire solution space (all possible combinations). But for large-size problems exact techniques suffer from exponential complexity. Finding the exact solution is NP-hard. Heuristic-based optimization algorithms manage to find solutions with an acceptable optimality (i.e. near the optimal solution). Among these algorithms SA has shown to be a robust technique. Moreover in this optimization problem, the generation of neighbors is easy. For these reasons our optimization algorithm is based on the SA. The pseudo-code in algorithm 1 resumes the implemented algorithm.

---

#### Algorithm 1 Optimize action combination

---

```

Begin with an initial configuration Comb and temperature t
while temperature t  $\geq$   $\alpha$  do
  Pick a neighboring configuration Comb'
  Let  $\Delta \leftarrow f(Comb') - f(Comb)$ 
  if  $\Delta \leq 0$  then
    Set Comb  $\leftarrow$  Comb'
  else
    Set Comb  $\leftarrow$  Comb' with probability  $e^{(-\Delta/t)}$ 
  end if
  Lower temperature t
end while
Return Comb

```

---

where Comb is the action combination;  $t$  the temperature;  $\alpha$  is the final temperature level; Comb' the neighboring combination;  $\Delta$  the change in the objective function. A neighbor combination is defined as the set of actions selecting aleatory one action and alternating the value of  $x_i$ , so the vector of one set of actions and its neighbor only differs by the value of one  $x_i$ . For the implementation of the annealed penalty the objective function  $f(Comb)$  is defined as the

objective function from the problem definition  $f_j^1(x)$  (equation 1) plus a penalty term defined as  $1/2t \times \sum_{j=1}^m f_j^2(x)$ , where  $f_j^2(x)$  is the constraint formulation. The algorithm starts with an initial combination of actions (set of all actions)  $Comb$  that is compared to a neighbor combination  $Comb'$ . If the function of the neighbor combination ( $f_j^1(Comb') + 1/2t \times \sum_{j=1}^m f_j^2(Comb')$ ) is lower than the cost of the function for  $Comb$  ( $f_j^1(Comb) + 1/2t \times \sum_{j=1}^m f_j^2(Comb)$ ), then  $Comb'$  is taken to be the new best answer. Otherwise,  $Comb'$  is only accepted with a probability given by a term that is a function of the difference between both answers and the temperature. The aim of sometimes accepting 'bad' answers is to allow the algorithm to explore the entire solution space and avoid getting caught in local minima. The exponential term allows the algorithm to accept bad answers at the beginning but to become restrictive as the temperature decreases. Once the solution is accepted the temperature is decreased and the cycle is repeated until the temperature reaches a minimum level. For the constraints implementation, we evaluated the use of a penalty function versus a hard constraint as discussed by Crama and Schyns (2003). We will prove through our experiments that an annealed penalty function as proposed by Michalewicz and Schoenauer (1996) works better for this problem than a hard constraint. The annealed penalty used is a function where the quadratic penalty weight is divided by the annealing temperature which is reduced as the search runtime increases. The penalty becomes quite high near the end of the search, and infeasible solutions tend to be rejected.

Once the algorithm is executed it provides a vector with an optimal combination among the set of actions that minimizes the performance cost while reaching an acceptability level set by a limit.

## 4 Case study

We applied the proposed method to an industrial case study concerning the development of a new product whose purpose is to facilitate the threading of compression stockings for elderly persons in medical homes. The product is addressed to the nurses who have to repeat the threading task several times a day for the different patients. The muscular effort required in this task has been associated as a source of repetitive strain injuries (RSI). The device thus intends to reduce this effort without disturbing the nurses' behavior because the time spent threading is also considered as a time for exchange between nurse and patient. It allows the nurses to ask questions about the emotional and health state of the patient.

In this project a few experimentations with low fidelity prototypes (Lim, Stolterman, and Tenenberg, 2008) were already made allowing it to evolve to the actual prototype. We can consider the prototype as an evolved version almost ready for the development phase. The reported data are based on the entire case study but some actions have different names due to confidentiality constraints.

### 4.1 Construction of Bayesian Network

In this first phase the evaluation models and data collection instruments were constructed by the project manager and the design team; the criteria were validated with the nurses and their managers.

#### 4.1.1 Acceptability criteria identification

The model for evaluating the product's acceptability was constructed using the criteria from Table 1 and the UTAUT model (Venkatesh et al., 2003) as reference. Table 2 shows the description of the nodes of the proposed acceptability model that makes it possible to evaluate this product (cf. Figure 3). This model was built using the GeNIe<sup>1</sup> software; it contains a Bayesian Networks reasoning and inference library called SMILE that we used in the next phase. From the model we can observe that there are five main categories of criteria: Perceived utility (PU), Perceived image (IM), Perceived ease of use (PEOU), Perceived sound (CO) and Perceived social influence (SI). Each one is composed by different factors. For example, perceived utility (PU) is related with the purpose of the device: 'facilitate the treading' (PU1) to 'increase treatment' (PU7). The perceived sound criteria were added by the group of experts because it seemed from the actual prototype to be something that could disturb the patient and they wanted to have the perception of the users. The criteria PU enables to group perceived comparative advantage factors and a question for the anxiety (perceived enjoyment). We gathered in the 'perceived ease of use' (PEOU) category the criteria relative to the perceived compatibility and complexity. The perceived novelty was associated with the 'perceived image' (IM) criteria. From the criteria in Table 1 the perceived privacy criteria were not included because the device did not collect any data. Finally, from the UTAUT model we added the 'perceived social influence' (SI) criteria because the product will be used in the social context of the medical home and the opinion of their colleagues and superiors was important to the nurses. All the criteria are linked to the intention to use the product here named as Potential Acceptability Index (PAI).

<sup>1</sup><https://dslpitt.org/genie/>

Table 2: Node description of the example network.

Criteria type	Code	Description of criteria
Perceived utility	PU1	Facilitate the threading
	PU2	Facilitate your job
	PU3	Reduce the unwanted effects for the patients (scratching,...)
	PU4	Reduce musculoskeletal disorder complications
	PU5	Reduce patient anxiety
	PU6	Reduce nurse anxiety
	PU7	Increase treatment compliance
Perceived image	IM1	Improve your professional image
	IM2	Improve the organization's image
	IM3	Provide an adequate device's image (color, form ...)
Perceived ease of use	PEOU1	Reduce fatigue related with threading
	PEOU2	Reduce muscle and joint pain associated with threading
	PEOU3	Decrease joint stiffness associated with the repeated donning
	PEOU4	Reduce muscle forces required when threading
	PEOU5	Make threading more comfortable
Perceived sound	CO1	The device sonority is adequate
Perceived social influence	SI1	My colleagues will support the use of this device
	SI2	My superiors will support the use of this device

The evaluation model was validated by the design team, the nurses, their superiors and an occupational therapist from the medical home.

#### 4.1.2 Data collection

In this step we used three data collection instruments: Firstly, a questionnaire adapted for the evaluation of the users' product perception (It contained an item for each criterion in the model using a 7-point Likert scale ranging from 'not at all' to 'very much' (Likert, 1932; Sinclair, 2005)). Secondly, an observation grid of the activity. Thirdly, a group meeting allowing the exchange with the participants. All the evaluation instruments were built and validated with the project manager.

The tests were undertaken during the month of July 2014 in a medical home in the east of France. A total of 18 care professionals participated in the test.

In the first part of the test we followed a group of nurses in the medical home during their morning shift while they used the device. After the test the nurses had to answer the questionnaire. The observation grid was used during the test to analyze the behavior during usage of both the patient and the nurse. It made it possible to identify gestures or expressions that could suggest fear or anxiety. At the end of the shift the group of nurses was gathered in a small meeting and the manager of the medical home was also invited to participate. The purpose of the meeting was to allow the group to give their opinions about the experiment and about the usage of the device. It also allowed the design team to collect a few suggestions that were used in the improvement phase.

The questionnaire had eighteen questions plus one conclusion question on the intention to use the product that corresponds to the Potential Acceptability Index (PAI). The seven point scale can be divided into three acceptability levels (or states in Figure 3): from 1 to 3, low level, 3.1 to 5 medium level and from 5.1 to 7, high level.

#### 4.1.3 Determination of the probability laws of the structure

The survey answers were fed to the modeled network. The data was associated to each node of the structure and then using the parameter learning algorithm (Dempster, Laird, and Rubin, 1977), the conditional probability tables (CPT) were built.

## 4.2 Initial acceptability assessment

Once the data is updated in the network the initial acceptability assessment was made. Figure 3 shows the value of the PAI node and Figure 4 shows the values for each composing criterion in the form of a radar. From these data we can observe that the device was rather attractive to users (PAI mean: 4.9 in a scale of 7); they evaluated it at the medium level. We could also deepen the analysis by visualizing the other criteria levels (the utility of the device, the usefulness...). From the radar we can also observe that four criteria (IM3, PU2, PU6, SI2) are better perceived than

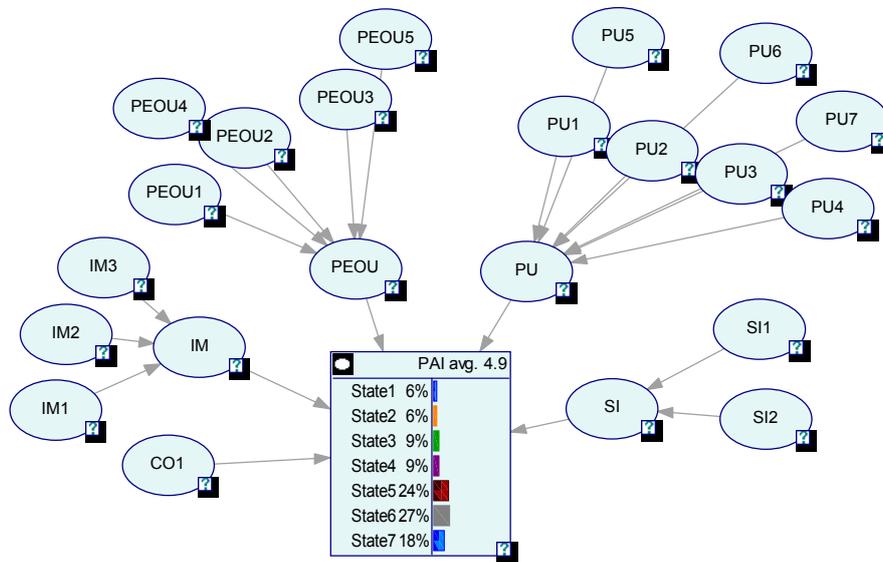


Figure 3: Bayesian network for the case study. Estimated acceptability (PAI) is at 4.9 on a scale of 7.

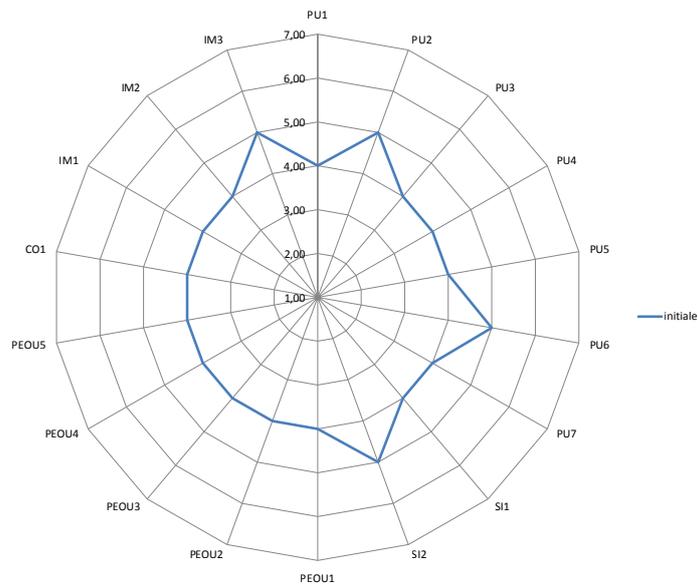


Figure 4: Radar with the node values for the initial acceptability assessment.

the others.

The initial assessment shown an acceptability level of 4.9 whereas the high value is 7. The project manager decided that 6 seems to be a good compromise between the high level and the assessed PAI. He decided to define the improvement target for the acceptability level to 6 (high level). The design team then started looking for actions that could be put into practice to improve the acceptability level.

### 4.3 Acceptability improvement

The team used the information and suggestions collected during the group meeting at the end of the test, as well as the observation grid. They helped in the definition of the improvement actions. From the initial assessment, the group had an overview of the entire criteria and their values. They listed all the actions that would impact and improve the criteria. Together they decided how each listed action was characterized by its impact to each criterion on the network (low, medium, high), (positive, negative).

#### 4.3.1 Characterization of actions and their impacts on the acceptability index

For this case the team identified and modeled twenty-five actions. They are listed in Table 3. Table 4 presents the impact from each action to each criterion and its associated execution cost. The cost was added by the project manager from his experience with the associated project cost.

Table 3: Modeled list of actions for the example.

Action	Description
a1	Increase engine power
a2	Increase opening distance
a3	Have a focus group in each center with the users to improve product perception
a4	Train users for an adequate use
a5	Define ergonomic operational modes
a6	Reduce product noise
a7	Improve pleasant appearance
a8	Add certification L2 (European level)
a9	Communicate evidence of improvement by product usages - marketing
a10	Communicate case of success - marketing
a11	Improve device professional appearance
a12	Reduce product weight
a13	Adapt the shape to the usage (ergonomics)
a14	Communicate the advantage of product usage in each client site - marketing
a15	Communicate the advantage of product usage in each client site to manager only
a16	Reduce initial distance for the stocking introduction
a17	Add color choices
a18	Add CE certification
a19	Add certification silver valley
a20	Present product in specialized showrooms
a21	Add certification L1 (national level)
a22	Add charging indicators
a23	Add charging and rest dock
a24	Increase power duration
a25	Add auto cleaning function

From Table 3, we can observe that there are actions that are directly related to product design: increase engine power, increase the opening distance (of the stocking device). There are others related to the users' perception of the product: certifications, training sessions, and marketing campaigns. Each action impacts as regards one or more criteria. The impact can be positive or negative. The  $a_1$  action (increase engine power) for example has a positive impact on the two criteria 'perceived utility' (PU) and 'perceived ease of use' (PEOU), but negative impact on the 'perceived sound' (CO1) of the device. It is the balance between impacts and costs that we intend to master in order to have a better product acceptability.

As we have already stated, the purpose of our work is to improve the acceptability level (here going from a level of 4.9 to a level greater or equal than 6) while minimizing the additional cost. We can accomplish an acceptability

Table 4: Matrix of action impact characterization for the case study.

	PU1	PU2	PU3	PU4	PU5	PU6	PU7	IM1	IM2	IM3	PEOU1	PEOU2	PEOU3	PEOU4	PEOU5	CO1	SI1	SI2	cost	K€
a1	3	2		3							3	2	2	3		-3				15
a2	2	2		2							3	3	3	3						3
a3					2	2									2	3				3
a4	1		2	1		1					-1				1					3
a5	3			3		3					2			2	1					3
a6					2	2										3				15
a7					2			-1	1	3										30
a8					3	2		3	3								1	1		15
a9					2	3		3	3								3	3		40
a10					2			3									3	1		40
a11					3	3		1	1	3										15
a12	2	1		2							2	1		2	1					30
a13	1							2	2	-1										10
a14								1	1								3	3		30
a15																	1	1		2
a16						3											3	3		3
a17					1			-1	-1	2										3
a18					2	3		1	2								2	2		5
a19					2	2			2									2		1.25
a20					1	2		3	3									2		10
a21					2	3		2	3								1	2		10
a22		2				1														3
a23		3						1	1								1	2		8
a24		2				2														5
a25		3	2			2		2	2								2	2		15

improvement by executing the twenty five improvement actions already listed. But that would incur a performance cost (here given in €) of 139000 €. Attempting to minimize the acceptability improvement cost makes it possible to reach the desired improvement while keeping the execution cost low.

### 4.3.2 Optimization problem solving with Simulated Annealing

In the optimization step the problem is to minimize the objective function given by equation 1. The optimization was done using the implemented SA algorithm (SA-p) described in the previous section.  $f^1(x)$  is defined as the change in the cost from the candidate vector, and  $f^2(x)$  verifies whether the constraint ‘PAI greater or equal to acceptability limit’ is fulfilled. We used the SA-p algorithm to solve the problem, results are shown in Table 5.

Table 6 shows the best improvement scenario for the 25 actions.

Table 5: Optimization results for the case study.

MIN cost ; ST PAI $\geq$ 6	SA-p 25
PAI	6.33
Cost (K€)	4 250 €
Number of iterations	12893
Solution found in (iteration number)	8959
Running time (s)	19.87

Table 6: Composition of the best improvement scenario for the optimization problem.

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25
$x_i$	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Table 6 presents the composition of the improvement scenario given by the optimization algorithm (‘1’ in the table indicates that the action  $a_i$  should be executed, ‘0’ that the action is not needed). We can observe that for the case with 25 actions, the answer is composed by an action that improved the perception of ease of use ( $a_5$ ) and an action that reinforced the confidence in the product ( $a_{19}$ ). The  $a_5$  action consist in defining ergonomic modes of operation, that means that the users are provided with clear instructions of how to handle and operate the device that would facilitate the ease of use. The certification ( $a_{19}$ ) concerns companies that are part of a professional cluster specialized in the market for seniors. It helps the product to be perceived as a better quality one as it shows that the company knows its clients. At first, when the actions were listed, the design team thought that the best improvement

to make was to increase the power of the motor (action  $a_1$ ) because it would allow the device to move faster during the threading. This action however is expensive compared to other actions providing a similar user satisfaction ( $a_2$  for example) but when combined with another action ( $a_4$ ) it added its impact to the impact of the former action. We

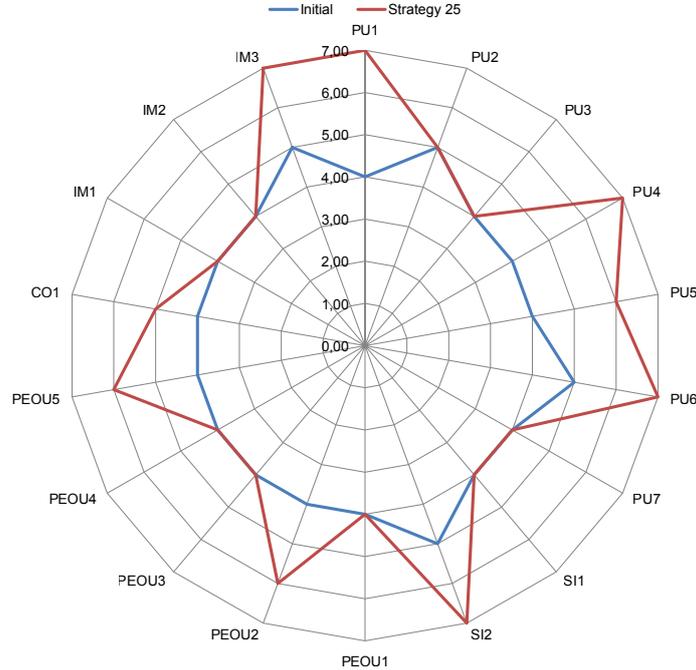


Figure 5: Criteria value distribution for 25 actions solution.

compared the value of each criterion for the solution set against its initial value. In Figure 5, we can observe that the solution set is focused on the improvement of the PU and PEOU criteria groups. The solution contained IM and SI criteria improvements as well. This combination helps to obtain the desired PAI level (6.33) and it required a budget of 4250 € for its execution.

The solution vector provided by the algorithm was the optimum solution according to the criteria used in the optimization problem. The algorithm however is not intended to replace the decision makers as they had to evaluate the solution provided and decide about its implementation. Once the decision-makers verified the combination of actions they decided to include those actions in the project planning to improve the users' product acceptability.

## 5 Algorithm performance

In order to compare the improvement obtained with the algorithm (SA-p) against an exact algorithm (EA) and a regular implementation of SA, we tested the 3 algorithms with 4 different problems sizes. These different problems were built from the real case study by truncating the real list of 25 actions to 10, 15, 20 and 25 (in this last case, the problem corresponds to the real case study). Due to the random nature of the SA algorithms we ran 20 trials for each of the SA experiments and obtained the average of the answer (only one for the EA). Four experiments were undertaken with 10, 15, 20 and 25 actions in order to compare when the solution space increases. The EA algorithm is a complete enumeration and evaluation of all possible combinations, the SA and SA-p algorithms correspond to the simulated annealing algorithm described by 1. Both algorithms differ in how they handle the constraints. The SA algorithm is the classic version as presented by Crama and Schyns (2003); the constraints are handled by a comparison. If the candidate solution does not fulfill the constraint, it is rejected. In the SA-p version the constraint is an annealed penalty function as described by Michalewicz and Schoenauer (1996). The candidate solution that does not satisfy the constraints might be accepted with a probability term. The algorithms were developed in Visual Studio using the SMILE engine for the Bayesian network and tested using an Intel Core i5-3470 3.20 GHz computer with 4 GHz of RAM memory. The parameters for the SA-p are set to  $\alpha = 0.999$  (cooling schedule with geometric decreasing), initial temperature is set to 400. The SA version does not behave well; the results are highly random with huge differences with the EA. Tables 7, 8 and 9, show the results for each one of the algorithms. Each of the experiments shows: the average PAI obtained estimated with the action combination proposed (PAI avg.), the cost of the combination, the number of iterations of the algorithm, the number of the iteration where the best solution was found, and the elapsed CPU time (running time). From the experiments we found without surprise that the EA always found the

Table 7: EA experiment results

<b>MIN cost; ST PAI <math>\geq</math> 6</b>	<b>EA 10</b>	<b>EA 15</b>	<b>EA 20</b>	<b>EA 25</b>
<b>PAI</b>	6.21	6.21	6.33	6.33
<b>Cost (€)</b>	6 000 €	6 000 €	4 250 €	4 250 €
<b>Number of iterations</b>	1024	32768	1048576	33554432
<b>Solution found in (iteration number)</b>	11	11	262161	262161
<b>Running time</b>	1.54(s)	52.19(s)	27.29(min)	14.23(h)
<b>Max PAI</b>	6.83	6.97	6.97	6.98
<b>Max Cost</b>	167 000 €	218 000 €	240 250 €	281 250 €

Table 8: SA experiment results

	<b>SA 10</b>		<b>SA 15</b>		<b>SA 20</b>		<b>SA 25</b>	
<b>MIN cost; ST. PAI <math>\geq</math> 6</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>
<b>PAI</b>	6.26	0.16	6.24	0.13	6.23	0.13	6.26	0.13
<b>Cost (€)</b>	34 950 €	30 062 €	35 250 €	25 951 €	35 475 €	27 266 €	33 563 €	19 873 €
<b>Number of iterations</b>	12893	0	12893	0	12893	0	12893	0
<b>Solution found in (iteration number)</b>	54.8	39.59	166.85	172.41	401.25	720.19	233.35	177.78
<b>Running time (s)</b>	20.39	0.26	20.3	0.21	20.17	0.06	19.99	0.23

Table 9: SAp experiment results

	<b>SA-p 10</b>		<b>SA-p 15</b>		<b>SA-p 20</b>		<b>SA-p 25</b>	
<b>MIN cost ; ST PAI <math>\geq</math> 6</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>	<b>avg</b>	<b>sd</b>
<b>PAI</b>	6.3	0.07	6.28	0.15	6.3	0.07	6.3	0.06
<b>Cost (K€)</b>	6 000 €	0 €	6 000 €	0 €	5 213 €	893 €	4 863 €	856 €
<b>Number of iterations</b>	12893	0	12893	0	12893	0	12893	0
<b>Solution found in (iteration number)</b>	8873	1313	9552	788	9555	312	9420	484
<b>Running time (s)</b>	19.95	0.18	19.97	0.12	20.43	0.3	19.94	0.07

best answer, but as the number of actions increase at a  $2^n$  rate the algorithm has to explore a greater solution space. Figure 6a, shows the solution space for 20 actions. In the bottom right corner are the actions that have a PAI greater or equal to 6 and a cost of less than 6000 €. This is the sub-optimal solutions region (it includes the global optimum). From Figure 6 we note that the space to explore is large and that only a few optimum answers are possible. The time required for the exploration of the solution space grows at an exponential rate. The SA-p algorithm runs exactly the same number of iterations (12893) and CPU time (20s) at each run. This algorithm found the exact answer easily when the solution space was small. When the solution space grows and due to its random nature it sometimes found the exact answer and at other times an answer that satisfied the criteria but was not the global optimum. Those sub-optimal answers however are not so far from the global one. When we observe the behavior after 20 runs and compare the difference between the answers found by the two algorithms, it is almost not significant (c.f. Table 10).

Table 10: Difference between the EA and SAp solutions

Problem size	10		15		20		25	
PAI	avg	sd	avg	avg	20	avg	avg	sd
$ Cost_{EA} - Cost_{SAp} $	0 €	0	0 €	0	963 €	893 €	613 €	856 €

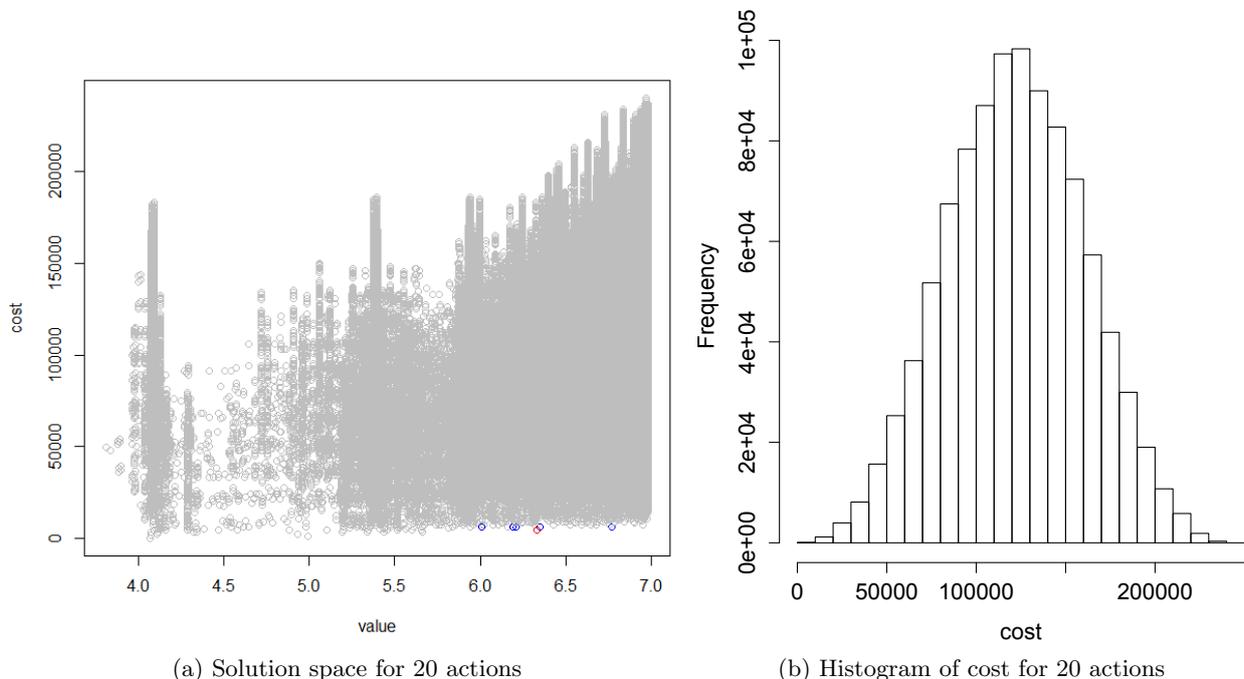


Figure 6: Solution space distribution for 20 actions

When we evaluate a distribution of the solution space (here limited to all the solutions that respect the constraint: PAI greater than 6) as shown in Figure 6b it is clear that a random solution has more probability to have a cost between 100k€ and 150k€ while the minimum solution has a small probability to be found in a random search. However, our modified version of the SA-p is able to find the optimal solution.

## 6 Discussion

The proposed method allows the project manager to estimate the acceptability level and to identify improvement actions for their product. The use of a Bayesian network allows the practitioners to better understand the influence of the different criteria on the acceptability level of their product. It also makes it possible to simulate how the changes in one or multiple criteria affect the level of the PAI. It is by the understanding of this impact that the design team and the project manager can identify actions to improve the acceptability level. The anticipation in the identification of the improvement actions helps to refine the specifications and requirements of the designed product before it goes into the production phase. Thus reducing the risk of delivering a product that does not respond to users' needs and expectations and making more effective the production system. In the case study, the improvement phase helped to

anticipate that a certification would be needed to improve the user's perception of a quality product. Anticipating that need helped the project manager to plan the production phase taking into account the requirements needed to fulfill the certification.

The definition and evaluation of the improvement actions are made possible by the use of the inference properties of BN. That would not be possible if only a simple weighted operator is used for the calculation of the PAI. Moreover, in the case study we found that the evaluation model (BN) was hierarchical and responded to a naive Bayes network (all criteria nodes were connected to the PAI). However, this is not the case for all the evaluation models in other domains. By our experimentation of the method we have found models where one or more criteria are linked between them. The structure of that type of problem is not suitable for multi criteria evaluation methods (e.g. AHP) but it is suitable for the use of a BN. In addition, the evaluation of all the possible improvement scenarios is a time consuming task that is reduced with the help of an optimization algorithm that searches for the best improvement scenario according to the improvement strategy defined by the project manager.

The proposed method can be partially compared to the acceptability estimation method proposed by Luo, Su, and Lee (2011) because they both make it possible to estimate the acceptability level of a product. However the intention of the evaluation is different in each approach. The approach proposed by Luo, Su, and Lee (2011) requires large data-sets and is based mainly on the historical data of the existing product. This makes the approach appropriate for the evaluation and improvement of existing products but is limited for new product design cases. Our proposed approach is not intended to deliver an exact level of acceptability (even if the indicator is generated and used) but to encourage systematic thinking and to help to understand the influence of the acceptability criteria and generating alternative scenarios to improve the product acceptability.

## 7 Conclusions and perspectives

This paper presents a method for acceptability evaluation and improvement that can be used in the first stages of product design. The method allows designers to estimate the acceptability level and to identify improvement actions for their product. Optimal combinations of those actions are evaluated according to the strategy defined by the project manager. The search is conducted using a modified version of the Simulated annealing algorithm. The resulting combination of actions allows the designers and the product manager to analyze the improvement scenarios to be put in practice. The proposed method exploits the properties of Bayesian networks for knowledge modeling and inference in order to deal with the lack of data that is a characteristic of this early phase. An illustrative case study related to the design of a medical stocking threading device shows the applicability of the method.

The case study shows that the initial assessment provides rich information for practitioners during the product design. On the other hand, the systematic reasoning put in practice by the method is an interesting approach for the search for improvement actions helping the design team to ask the right questions and to find appropriate solutions to improve the product acceptability.

In a regular design project there could be more than twenty improvement actions to be evaluated. We have shown that in this case the use of a heuristic algorithm proves to be more efficient than a complete search of the entire solution space. Our proposed algorithm shows its utility to practitioners for facilitating the evaluation and analysis of the acceptability improvement scenarios. Through our experiments we have demonstrated that the acceptability improvement problem is better solved by the modified SA algorithm when the constraints are handled using an annealed penalty function.

The proposed method has some limitations that need to be addressed in future work. The model used for acceptability assessment is based on the users' perception of the proposed solution. We evaluated perceived functionality and ease of use, for example. However, these factors are dependent on the perception of the solution. Good utility tends to directly impact good acceptance, poor utility does not indicate what could be operationally improved by the design team. The managers and designers have to define which actions or improvements need to be put into practice, but they do this helped by their experience and by the knowledge gathered with the users' test. Another approach is to directly relate product attributes to the evaluation. That would allow the design team to know which product characteristics have to be modified. Perceived utility can be decomposed in design parameters like force of the device, threading speed for example (Zhou, Ji, and Jiao, 2014; Zhou and Jiao, 2013). Determining design parameters linked to perceived criteria could be used to automatically propose improvement actions. Additionally, the subjectivity involved in the construction of the influence matrix can be reduced by exploring the use of fuzzy indicators.

On the other hand, our implementation of the simulated annealing algorithm has been configured with the parameters that worked better after the experiments, but this parameter tuning can also be improved using dynamic parameter tuning. Further work may explore the use of other evolutionary algorithms such as genetic algorithms, ant colony optimization or particle swarm optimization.

The optimization strategy presented in this paper, intends to improve the acceptability level while minimizing the effort required (cost). The presented strategy is only one of many that can be used. Managers can for example seek

the maximum acceptability level restricted to a project budget. Or they may improve the acceptability level while requiring that some criteria reach a specific level. Further work may deal with the implementation of those different strategies in order to be presented to the decision-makers who will be able to compare and select the one that best fits their interest.

## Acknowledgment

This work is financed by a grant (number CT65092012SU) from the French Ministry of Higher Education and Research (Ministère de l'Enseignement supérieur et de la Recherche, MESR). The models described in this paper were created using the GeNIe modeling environment. The optimization algorithm uses the SMILE reasoning engine, both tools are developed by the Decision Systems Laboratory, University of Pittsburgh.

## References

- Ajzen, I. and M. Fishbein (1980). *Understanding attitudes and predicting social behavior*. Prentice-Hall: Englewood Cliffs, NJ.
- Alsawalqah, H. I., S. Kang, and J. Lee (2014). "A method to optimize the scope of a software product platform based on end-user features". In: *Journal of Systems and Software* 98, pp. 79–106.
- Aoussat, A., H. Christofol, and M. Le Coq (2000). "The new product design - a transverse approach". In: *Journal of Engineering Design* 11(4), pp. 399–417.
- Arbelaez-garces, G., E. Bonjour, and A. Rakotondranaivo (2014). "Towards a method for acceptability analysis : application to healthcare innovation". In: *2014 International Conference on Engineering, Technology and Innovation (ICE)*, pp. 1–6.
- Ben Rejeb, H., V. Boly, and L. Morel-Guimaraes (2008). "A New Methodology Based on Kano Model for the Evaluation of a New Product Acceptability during the Front-End Phases". In: *32nd Annual IEEE International Computer Software and Applications Conference*. IEEE, pp. 619–624.
- Bouaziz, M. F., E. Zamaï, and F. Duvivier (2013). "Towards Bayesian network methodology for predicting the equipment health factor of complex semiconductor systems". In: *International Journal of Production Research* 51(15), pp. 4597–4617.
- Brosnan, M. J. (1999). "Modeling technophobia: A case for word processing". In: *Computers in Human Behavior* 15(2), pp. 105–121.
- Caine, K. E., M. O'Brien, S. Park, W. A. Rogers, A. D. Fisk, K. van Ittersum, M. Capar, and L. J. Parsons (2006). "Understanding Acceptance of High Technology Products: 50 Years of Research". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 50(18), pp. 2148–2152.
- Camargo, M., L. Wendling, and E. Bonjour (2014). "A fuzzy integral based methodology to elicit semantic spaces in usability tests". In: *International Journal of Industrial Ergonomics* 44(1), pp. 11–17.
- Camargo, M., R. Bary, N. Skiba, V. Boly, and R. Smith (2012). "Studying the implications and impact of smartphones on self-directed learning under a Living Lab approach". In: *International Journal of Product Development* 17, p. 119.
- Chesbrough, H. W. (2003). *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business Press.
- Crama, Y. and M. Schyns (2003). "Simulated annealing for complex portfolio selection problems". In: *European Journal of Operational Research* 150(3), pp. 546–571.
- Davis, F. D. (1989). "Perceived usefulness, perceived ease of use, and user acceptance of information technology". In: *MIS quarterly* 13(3), p. 319.
- Dempster, A., N. Laird, and D. Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the royal statistical society*. 39(1), pp. 1–38.
- Dillon, A. and M. Morris (1999). "Power, Perception and Performance: From Usability Engineering to Technology Acceptance with the P3 Model of User Response". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 43. 19, pp. 1017–1021.
- Durugbo, C. (2014). "Strategic framework for industrial product-service co-design: findings from the microsystems industry". In: *International Journal of Production Research* 52(10), pp. 2881–2900.
- Elazouni, A. M., A. E. Ali, and R. H. Abdel-Razek (2005). "Estimating the Acceptability of New Formwork Systems Using Neural Networks". In: *Journal of Construction Engineering and Management* 131(1), pp. 33–41.
- Featherman, M. S. and P. A. Pavlou (2003). "Predicting e-services adoption: A perceived risk facets perspective". In: *International Journal of Human Computer Studies* 59(4), pp. 451–474.
- Gruen, W. (1960). "Preference for new products and its relationship to different measures of conformity." In: *Journal of Applied Psychology* 44, pp. 361–364.

- Huang, C. and A. Darwiche (1996). *Inference in belief networks: A procedural guide*.
- Jenkins, S., R. Brown, and N. Rutterford (2009). “Comparing Thermographic, EEG, and Subjective Measures of Affective Experience During Simulated Product Interactions”. In: *International Journal of Design* 3(2), pp. 53–65.
- Jensen, F. V. and T. D. Nielsen (2007). *Bayesian Networks and Decision Graphs*. Information Science and Statistics. Springer New York: New York, NY.
- Jiang, H., C. Kwong, Y. Liu, and W. Ip (2015). “A methodology of integrating affective design with defining engineering specifications for product design”. In: *International Journal of Production Research* 58(8), pp. 2472–2488.
- Kirkpatrick, S., C. Gelatt, and M. Vecchi (1983). “Optimisation by Simulated Annealing”. In: *Science* 220(4598), pp. 671–680.
- Lee, S. and H.-P. Wang (1992). “Modified simulated annealing for multiple-objective engineering design optimization”. In: *Journal of Intelligent Manufacturing* 3, pp. 101–108.
- Likert, R. (1932). “A technique for the measurement of attitudes.” In: *Archives of psychology* 22(140), pp. 5–55.
- Lim, Y.-K., E. Stolterman, and J. Tenenbergs (2008). “The anatomy of prototypes”. In: *ACM Transactions on Computer-Human Interaction* 15(2), pp. 1–27.
- Lu, J. and M. Liechty (2007). “An empirical comparison between nonlinear programming optimization and simulated annealing (SA) algorithm under a higher moments Bayesian portfolio selection framework”. In: *Proceedings - Winter Simulation Conference*. IEEE, pp. 1021–1027.
- Luarn, P. and H.-H. Lin (2005). *Toward an understanding of the behavioral intention to use mobile banking*.
- Luo, S.-T., C.-T. Su, and W.-C. Lee (2011). “Constructing intelligent model for acceptability evaluation of a product”. In: *Expert Systems with Applications* 38(11), pp. 13702–13710.
- Michalewicz, Z. and M. Schoenauer (1996). *Evolutionary Algorithms for Constrained Parameter Optimization Problems*.
- Moore, G. C. and I. Benbasat (1991). “Development of an instrument to measure the perceptions of adopting an information technology innovation”. In: *Information Systems Research* 2(3), pp. 192–222.
- Nielsen, J. (1993). *Usability engineering*. Academic Press: San Diego, CA.
- Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.
- Pirlot, M. (1996). “General local search methods”. In: *European Journal of Operational Research* 92(96), pp. 493–511.
- Rogers, E. M. (1995). *Diffusion of innovations*. Free Press: New York.
- Silva, L. C. d. C. B., A. B. de Oliveira, D. C. Silva, L. C. Paschoarelli, and H. J. C. G. Coury (2013). “Evaluation of reusable cardboard box designs: Biomechanical and perceptual aspects”. In: *International Journal of Industrial Ergonomics* 43(2), pp. 154–160.
- Sinclair, M. A. (2005). “Subjective assessment”. In: *Evaluation of Human Work*. Ed. by E. N. Corlett, J. R. Wilson, and N Corlett. Third. CRC Press. Chap. 3.
- Ullman, D. G. (1994). *The mechanical design process*. 4th. McGraw-Hill.
- Venkatesh, V. and F. D. Davis (2000). “A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies.” In: *Management Science* 46(2), p. 186.
- Venkatesh, V., M. G. Morris, G. B. Davis, and F. D. Davis (2003). “User acceptance of information technology: toward a unified view”. In: *MIS Quarterly* 27(3), pp. 425–478.
- Zhou, F., Y. Ji, and R. J. Jiao (2014). “Prospect-Theoretic Modeling of Customer Affective-Cognitive Decisions Under Uncertainty for User Experience Design”. In: *Human-Machine Systems, IEEE Transactions on* 44(4), pp. 468–483.
- Zhou, F. and R. J. Jiao (2013). “An improved user experience model with cumulative prospect theory”. In: *Procedia Computer Science* 16, pp. 870–877.
- Zhou, F., X. Qu, M. G. Helander, and J. Jiao (2011). “Affect prediction from physiological measures via visual stimuli”. In: *International Journal of Human Computer Studies* 69(12), pp. 801–819.