

# Avantages et limites des modèles de caractéristiques dans la modélisation des exigences de variabilité

Raúl Mazo

## Résumé :

Les lignes de produit sont devenues le moyen principal de capitaliser et réutiliser les artefacts en ingénierie de systèmes. Les domaines d'application sont nombreux, allant de systèmes complexes tels que des véhicules automobiles, logiciels embarqués, systèmes d'information et progiciels de gestion intégrée à des systèmes de conception simple mais qui requièrent une forte dose de personnalisation. Le principe essentiel est de développer puis d'utiliser des modèles conceptuels de ligne de produit dans lesquels sont spécifiés les artefacts capitalisés et que l'on veut réutiliser au niveau des produits, ainsi que les contraintes d'agencement de ces éléments. Le concept commun à la plupart de ces modèles conceptuels, au moins dans la littérature, est celui de « caractéristique », la notation la plus connue en la matière étant FODA (Feature-Oriented Domain Analysis). Cet article présente la modélisation des caractéristiques en ingénierie des lignes de produits. Il montre en particulier comment représenter les exigences de variabilité dans le temps et dans l'espace d'un système en utilisant cette notation. S'appuyant sur l'exemple du système de freins électriques de stationnement l'article discute les avantages et limites des modèles de caractéristiques et propose quelques guides sur la conduite à tenir lorsque les limites de la notation sont atteintes.

**Mots clés :** modèles de caractéristiques, lignes de produit, exigences de variabilité, modélisation, avantages, limites

## 1. MODÈLES DE CARACTÉRISTIQUES

La modélisation des caractéristiques est l'approche la plus répandue en ingénierie de lignes de produits logiciels. Outre qu'elle offre un moyen simple de représenter les caractéristiques (*features* en anglais) communes et variables d'une famille de produits, elle offre un support efficace au raisonnement pour les activités d'ingénierie d'application, qu'il s'agisse de spécifier les exigences d'un nouveau système, de le concevoir, de le produire, de le configurer, ou même de le tester. Plus généralement, le concept de feature est très présent dans l'industrie des produits logiciels où, en particulier dans le cadre d'innovations, tout ne peut être ramené sur le terrain des exigences. Le concept de caractéristique a ceci d'avantageux qu'il est extrêmement vague. Une caractéristique est en effet définie comme étant un « aspect important ou distinctif d'un système, visible par l'utilisateur ou par rapport à la qualité » [5]. En d'autres termes, une exigence, une qualité, une fonction, un composant, un constituant, une propriété physique sont autant d'aspect d'un produit que l'on peut spécifier dans les modèles de caractéristiques. Le modèle de caractéristiques FODA [5] est probablement la notation qui a le plus retenu l'attention ces dernières années, comme en témoignent les nombreux dialectes dont elle est à l'origine, tels que les modèles de caractéristiques avec cardinalités et multiplicités [2,8] et des modèles de caractéristiques avec des attributs [1,5,6] – la liste n'est pas exhaustive.

### Modèles de caractéristiques basiques

Tous les modèles de caractéristiques représentent les caractéristiques sous la forme d'un graphe dont la structure élémentaire est un arbre où les caractéristiques sont décrites hiérarchiquement selon une relation d'affinement. Les modèles de caractéristiques basiques—à la FODA—permettent de modéliser quatre grandes catégories de relations entre les caractéristiques : obligatoire, optionnelle, à choix alternatif, à choix multiple. La figure 1 présente un exemple (voir Section 2) de modèle de caractéristiques selon cette structure d'arbre. La caractéristique à la racine de l'arbre est obligatoire. L'aspect variable ou optionnel des autres caractéristiques du produit ne peut être interprété qu'à la lueur des liens avec leurs parents et de la nature elle-même obligatoire ou optionnelle du parent.

- **Obligatoire :** lorsqu'une sous-caractéristique possède une relation obligatoire avec son parent, le produit ne peut présenter la caractéristique fille si et seulement si elle présente aussi la caractéristique parent. Cette relation va donc dans les deux sens : un produit qui aurait la caractéristique fille doit aussi présenter la caractéristique parent. Par exemple, ce type de relation permettra de spécifier que tout système de freinage doit inclure une boîte de vitesse.
- **Optionnelle :** lorsqu'une sous-caractéristique possède une relation optionnelle avec son parent cela signifie que le produit ne peut présenter cette caractéristique que si il présente aussi la caractéristique parent, mais le produit peut présenter la caractéristique parent sans la sous-caractéristique, c.-à-d. la sous-caractéristique est non-obligatoire. C'est au

moyen d'une relation optionnelle que l'on peut par exemple spécifier qu'un système de freinage électrique peut comprendre l'option d'un service d'assistance au démarrage en pente.

- **Choix alternatif** un ensemble de sous-caractéristiques est relié à une caractéristique parent par une relation de choix alternatif si et seulement si le produit ne peut présenter qu'une et une seule des sous-caractéristiques lorsqu'il possède la caractéristique parent. À l'inverse, aucune des sous-caractéristiques n'apparaît lorsque le produit ne présente pas la caractéristique parent. Par exemple, un système de freinage électrique doit s'implémenter soit de manière manuelle, soit de manière automatique, soit de manière assistée. Cela peut être spécifié au moyen d'un choix alternatif.

- **Choix multiple** : un ensemble de sous-caractéristiques est relié à une caractéristique parent par une relation à choix multiple si une ou plusieurs sous-caractéristiques peuvent être incluses dans un produit qui présente la caractéristique parent. Dans l'exemple du système de freinage électrique, on utilisera une relation de choix multiple pour indiquer qu'il peut être implémenté pour être conforme à une ou plusieurs des zones de réglementation (UE, USA, autres).

En plus des relations père-fils qui existent entre les caractéristiques, un modèle de caractéristiques peut aussi inclure des relations transversales entre les caractéristiques. Typiquement, ces sont des relations d'inclusion et d'exclusion, spécifiées le plus couramment dans la littérature sous forme de dépendances nommées respectivement « requiert » et « exclues » :

- **Requiert** : lorsqu'une caractéristique *A* requiert une caractéristique *B*, cela signifie que lorsqu'un produit présente la caractéristique *A*, alors il doit aussi présenter la caractéristique *B*. On utilisera une dépendance « requiert » pour spécifier qu'un système de freinage électrique incluant un service de freinage de stationnement doit aussi inclure un système de freinage assisté.

- **Exclut** : une caractéristique *A* exclut une caractéristique *B* signifie que l'inclusion de *A* dans un produit exclut *B* du même produit. Par exemple, comme les systèmes de freinage électrique ne peuvent pas inclure à la fois un système de freinage de stationnement traditionnel et un système de distribution de force réduit et dynamique, cela sera spécifié au moyen d'une dépendance « exclut » entre les deux caractéristiques.

Des contraintes plus complexes ont été proposées ultérieurement sous la forme de formules propositionnelles [1], par exemple « *A* and *B* requires not *C* », et sous la forme de contraintes booléennes, arithmétiques, symboliques et réifiées [9].

Néanmoins, une limite des modèles de caractéristique est le faible pouvoir d'expression lié à la nature fondamentalement booléenne du concept : un produit possède une caractéristique ou pas. Outre que cela oblige à démultiplier le nombre de caractéristiques spécifiées pour décrire des variables par ailleurs simples, cela rend difficile la modélisation des contraintes, qui deviennent ainsi de plus en plus complexes. Par ailleurs, il est impossible de spécifier des ensembles ou de trouver un équivalent au concept d'instanciation, alors que ceux-ci sont tout à fait commun dans le monde des bases de données et de l'orienté objet.

### Modèles de caractéristiques avec cardinalités

Afin d'introduire un équivalent aux concepts d'ensemble et d'instanciation, certains auteurs proposent d'étendre les modèles de caractéristiques FODA par des cardinalités UML [2,8]. Deux types de cardinalités peuvent en fait être distingués : les cardinalités de caractéristiques et les cardinalités de groupes de caractéristiques.

- Une **cardinalité de caractéristique** est un intervalle noté [*n*..*m*] associé à une caractéristique, dans lequel *n* est la borne inférieure et *m* la borne supérieure de l'intervalle, qui détermine le nombre d'instances de la caractéristique dans un produit. Par exemple, une automobile peut contenir 2, 3, 4 ou 5 portes. Cela se spécifie avec une cardinalité de caractéristiques [2..2][4..5] associée à la caractéristique « porte ».

- Une **cardinalité de groupe de caractéristique** est un intervalle noté sous la forme <*m*..*n*>, *m* étant la borne inférieure de l'intervalle et *n* la borne supérieure, limitant le nombre des caractéristiques de ce groupe que peut présenter un produit. Toutes les caractéristiques du groupe auquel la cardinalité s'applique doivent avoir le même père. En conséquence, une relation à choix alternatifs entre sous-caractéristiques est sémantiquement équivalente à une cardinalité <1..1> du groupe des caractéristiques. De la même manière, une relation à choix multiples correspond à une cardinalité <1..*n*> du groupe des caractéristiques, *n* étant le nombre total des caractéristiques du groupe.

### Modèles de caractéristiques avec attributs

Outre la dimension ensembliste, on peut avoir envie de représenter des caractéristiques mesurables de manière numériques. Pour ce faire, les modèles de caractéristiques ont été étendus avec un concept d'attributs qui permettent d'associer davantage d'informations aux caractéristiques. Ce principe trouve son origine dans la notation FODA [5] qui déjà proposait de spécifier des informations supplémentaires associées à chacune des caractéristiques : nom, description, synonymes,

source, type, etc. Plus tard, la notation FORM [6] fait référence aux caractéristiques non fonctionnelles auxquelles sont rattachées des attributs. Depuis, plusieurs autres notations ont proposé d'associer des attributs aux caractéristiques [1,2].

Il n'existe pas de consensus sur une manière unique de spécifier les attributs. Néanmoins, la plupart des notations conviennent qu'un attribut doit comporter au moins un nom, un domaine de valeur et une valeur qui lui sera attribuée ou non selon que la caractéristique à laquelle l'attribut est rattaché est présente dans le produit ou pas. Par exemple, si un produit possède un « Câble de traction », la caractéristique qui le définit est spécifiée plus précisément au moyen de l'attribut poids dont la valeur par défaut est de 20 grammes. Les modèles de caractéristiques avec attributs peuvent également comprendre des contraintes complexes entre les attributs et les caractéristiques. Par exemple : « Si l'attribut A de la caractéristique F1 est inférieur à une valeur X, alors la caractéristique F2 ne peut pas faire partie du produit ».

## 2. LE CAS DU SYSTÈME DES FREINS ÉLECTRIQUES DE STATIONNEMENT

Cette section présente le cas industriel du système des freins électriques de stationnement EPB (de l'anglais Electric Parking Brake). Ce modèle développé avec Renault, mais que l'on trouve aujourd'hui couramment dans les véhicules de toute marque, est une variation du système mécanique des freins de parking, qui assure l'immobilisation du véhicule lorsque le conducteur met le véhicule à l'arrêt complet et quitte le véhicule. Nous avons choisi cet exemple pour sa simplicité, parce qu'il s'agit d'une ligne réelle et complète de produits industriels (plus de détails sont présentés dans [3,7]) et parce que les exigences sur l'EPB présentent suffisamment de variabilité pour illustrer notre propos. Les sous-sections ci-dessous présentent les trois premières vues de ce système et leur modélisation en utilisant les notations présentées précédemment, et en les classant conformément au processus d'ingénierie système employé chez Renault.

- **Variabilité visible aux clients** (variabilité au niveau du véhicule). Un frein de stationnement propose trois types de services : manuel – le frein est commandé par le conducteur, soit par le levier classique ou par un interrupteur ; automatique – le système peut activer ou désactiver le frein lui-même en fonction de la situation ; assistée – il apporte des fonctions supplémentaires qui aident le conducteur dans d'autres situations (comme l'aide au démarrage en côte). Dans le cas d'un fonctionnement automatique et d'un fonctionnement assisté, le système peut décider de bloquer le frein de stationnement (par exemple, lorsque le conducteur quitte le véhicule ou lorsque le moteur est à l'arrêt). La fonction d'assistance au démarrage en pente est également une option pour les clients. Cette fonction peut être mise en œuvre par deux systèmes différents : le frein de stationnement électrique lui-même ou le système de freinage classique.

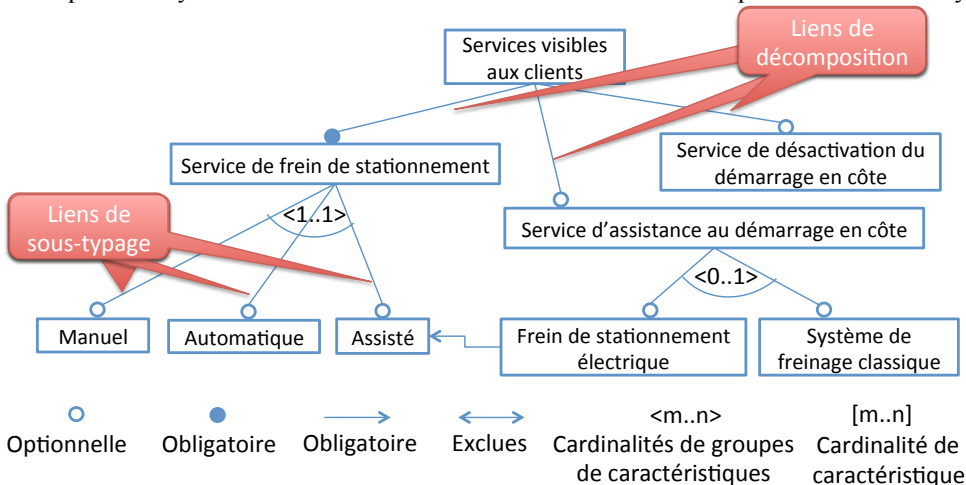


Figure 1 : Modèle de caractéristiques de la variabilité du système EPB visible aux clients

Néanmoins, le service d'assistance au démarrage en pente doit être désactivé (« Service de désactivation du démarrage en pente ») pour les systèmes qui ont à la fois un frein de stationnement électrique et qui sont installés dans un véhicule type « RemorqueW1 » (un type de remorque de Renault). Dans ce cas, il faut, soit écrire la contrainte de manière textuelle, soit créer une caractéristique artificielle avec deux sous-caractéristiques obligatoires. Ainsi, chaque caractéristique obligatoire doit être incluse par les caractéristiques « frein de stationnement électrique » et « RemorqueW2 » (voir Figure 2) respectivement. Pour compléter le modèle, la caractéristique artificielle père doit inclure la caractéristique correspondant au « Service de désactivation de démarrage en pente ».

L'autre limite rencontrée dans la spécification du système EPB avec un modèle de caractéristiques simple est due au fait que la nature des hiérarchies est complètement implicite et nécessite des annotations pour spécifier le type de lien (par exemple, décomposition, sous-typage, etc.) tel comme il est présenté dans la Figure 1.

- **Contexte** (environnement du véhicule). Les variantes dans l'environnement du véhicule font référence, par exemple, au type de « Boîte de vitesse », qui a un impact direct sur le comportement du système EPB en présence de certains types de remorques (« Remorque ») qui exigent que l'EPB soit désactivé, et à la réglementation qui varie selon les pays de commercialisation. Les autres attributs des véhicules, comme le modèle et le poids du véhicule, doivent aussi être pris en compte dans la gestion de variabilité du système EPB.

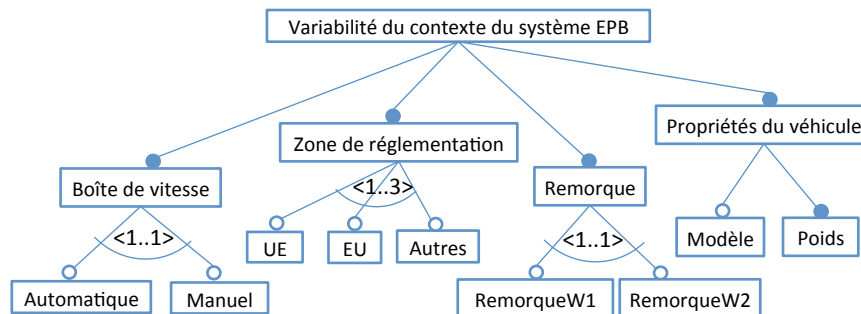


Figure 2 : Modèle de caractéristiques correspondant à la variabilité du contexte du système EPB

Les attributs associés aux différentes variantes sont un atout important des modèles de caractéristiques, car ils apportent plus d'informations sur l'impact des décisions sur les futurs systèmes. Par exemple, sur la performance (par exemple, force de freinage dissymétrie, temps de réponse sur frein), la réutilisation (par exemple, la gamme des véhicules), sur l'augmentation des coûts par rapport à une configuration de référence (par exemple, coûts supplémentaires d'ingénierie) ou sur les propriétés elles mêmes des produits (par exemple, modèle et poids des véhicules). Les attributs peuvent servir à révéler des informations au cours des processus de configuration et dérivation, et aider les ingénieurs à faire les bons choix, en permettant d'évaluer l'impact de leurs choix sur la configuration du système [4]. Néanmoins, il faut noter dans la Figure 2 que certains attributs, comme le modèle et le poids du véhicule, ont dû être modélisés sous forme de caractéristiques et non comme des attributs de caractéristiques. Cela est dû au fait qu'on ne dispose pas des parties des modèles auxquels ces attributs appartiennent. Cela permet aussi de faciliter la représentation graphique de certaines relations transversales impliquant ces deux attributs. Par exemple, le système doit être en mesure de désactiver l'assistant de démarrage en pente si une remorque est attachée au véhicule ou si le poids du véhicule est supérieur à une certaine valeur.

- **Alternatives de conception** (décisions de conception qui influent sur la totalité ou sur une partie de la solution technique). Les décisions de conception comprennent les principales alternatives de solutions (alternatives d'architecture), des choix sur la façon de répartir l'effort entre l'EPB et le système de freinage hydraulique principal (alternatives de distribution de force), des choix sur les architectures logiciels (alternatives d'architectures logiciels) et sur l'allocation de la fonction optionnelle de détection d'angle de pente (détection d'angle de pente). L'allocation de cette dernière fonction à un ordinateur spécifique exigerait évidemment que l'ordinateur existe déjà. Le modèle de caractéristiques correspondant aux alternatives de conception technique du système EPB est présenté dans la Figure 3.

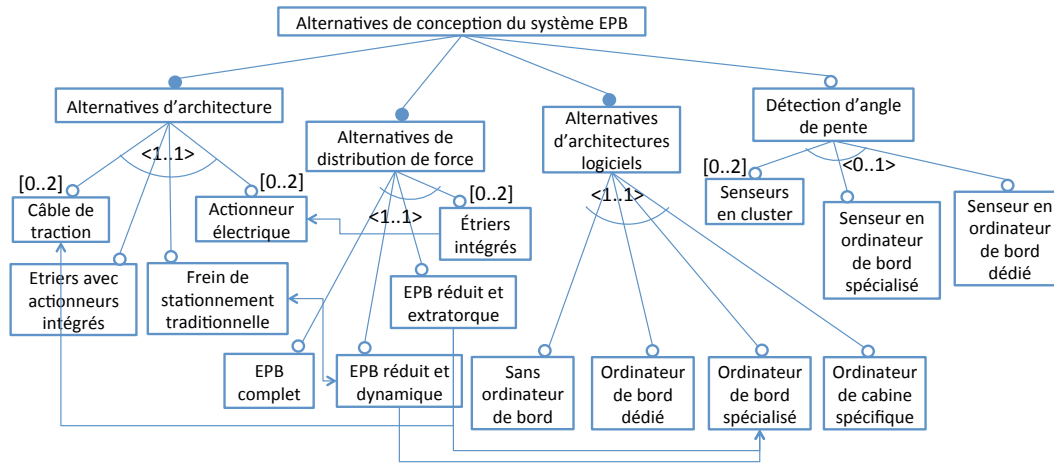


Figure 3 : Modèle de caractéristiques lié aux alternatives de conception du système EPB

Les composants qui donnent une solution aux alternatives d'architecture peuvent être de différents types et fournis par des différents fabricants. Choisir tel ou tel type de composant de telle ou une telle qualité et/ou de tel ou tel fabricant est aussi une forme de variabilité qu'on ne représente pas de manière correcte dans les modèles de caractéristiques, car il n'y a pas de séparation entre l'espace du problème (ce qui est présenté dans la Figure 3) et l'espace de la solution (les différents possibilités par rapport au fabricant, à la qualité, à la disponibilité en stock, etc.). Les logiciels utilisés pour détecter l'angle de pente peuvent aussi être implémentés de différentes manières et par différentes entreprises. Bien entendu, cette relation d'un à plusieurs entre une caractéristique et ses différentes implémentations n'est pas non plus représentable dans un modèle de caractéristiques car la relation entre les éléments de l'espace de problème et les éléments de l'espace de la solution n'est pas définie dans ces modèles.

L'utilisation de cardinalités individuelles est fort utile pour représenter le fait qu'une caractéristique puisse être dupliquée plusieurs fois dans certains produits. Néanmoins, on peut vite percevoir un problème de sémantique lorsque l'on spécifie des restrictions entre des caractéristiques avec cardinalités individuelles ou groupales. Ainsi, quand une dépendance « requiert » est positionnée entre la caractéristique « Étriers intégrés » et la caractéristique « Actionneur électrique », rien n'indique si la sémantique sous-jacente est qu'à chaque fois qu'il y a un Étrier intégré il faudrait aussi un Actionneur électrique, ou si pour chaque Étrier intégré il faudrait deux Actionneurs électriques.

Il faut noter aussi que la contrainte d'existence d'un ordinateur pour pouvoir allouer la fonction de détection d'angle de pente n'a pas été représentée dans la Figure 3. Cela impliquerait la création d'une caractéristique artificielle pour grouper les caractéristiques qui impliquent l'utilisation d'un ordinateur, et par conséquent l'apparition d'un nouveau niveau hiérarchique.

Il convient de noter que le caractère orthogonal de l'architecture de notre exemple (EPB) n'a pas pu être représenté dans la structure arborescente des modèles de caractéristiques en raison du fait que (i) la propagation des contraintes de variabilité entre les différents points de vue rend difficile la compréhension du modèle de caractéristiques, et (ii) les caractéristiques correspondant aux différentes vues doivent être représentés de manière ordonnée pour guider le processus de configuration selon le processus d'ingénierie des systèmes de Renault. Ainsi, comme expliqué précédemment, les contraintes entre les différentes vues n'ont pas été représentées dans les modèles de caractéristiques pour faciliter leur lecture.

### 3. CONCLUSIONS ET PERSPECTIVES

Les modèles de caractéristiques sont généralement utilisés pour exprimer la variabilité visible par les clients sous forme d'une collection structurée d'éléments obligatoires et facultatifs. Comme le montre Salinesi et al. [9], ces modèles sont intuitifs, grâce à leur nombre réduit de concepts et à la simplicité de leur représentation, et ils sont donc faciles à spécifier et à interpréter, notamment pour des débutants et des non-ingénieurs, même si des ambiguïtés peuvent surgir.

Cependant, la simplicité et la structure arborescente (formellement il s'agit en fait d'un graphe orienté acyclique ou DAG : *Directed Acyclic Graph*) des modèles de caractéristiques devient un handicap, dès lors que l'on souhaite spécifier des

ensembles, des phénomènes d'instanciation, ou des points de vues orthogonaux comme l'a illustré l'exemple (EPB). En fait, la structure arborescente devient un handicap en raison de la propagation des contraintes, que ce soit entre caractéristiques de différents niveaux, appartenant à des modèles orthogonaux, ou entre les différents points de vue: client, contexte, conception, comportement interne et architecture physique. Deux modèles sont orthogonaux s'il n'y a pas de relations hiérarchiques entre ses groupes de variables (i.e., entre ses groupes de caractéristiques quand il s'agit de modèles de caractéristiques).

Par ailleurs, les modèles de caractéristiques ne permettent pas de définir d'ordre dans le processus de configuration. La structure arborescente est en effet un moyen de simplifier la spécification et la lecture, mais elle ne fait que définir des contraintes entre caractéristiques : c.-à- d. n'indique qu'il faille parcourir l'ordre du haut vers le bas, en largeur (si toutefois cela avait un sens), en profondeur, ou autre lorsque l'on définit les caractéristiques d'un produit.

Une autre limite des modèles de caractéristiques est le fait que l'espace de solution n'est pas pris en compte dans la modélisation fonctionnelle. Ainsi, espace de problème et espace de solution sont regroupés dans le même modèle. En fait, cela rend difficile d'associer à une caractéristique plusieurs composantes de l'espace de solution, et à l'inverse enlève la possibilité de réutiliser le modèle qui représente l'espace de problème pour différents modèles de l'espace de solution. Pour illustrer ce dernier point, on peut prendre l'exemple d'une société qui dispose de plusieurs fournisseurs pour certains de ses composants et souhaite donc réutiliser le modèle représentant l'espace du problème pour les différents modèles représentant l'espace de solution.

Une autre limite des modèles de caractéristiques est que la sémantique des contraintes croisées entre les caractéristiques avec de cardinalités individuelles n'est pas bien définie ; comme présenté dans la section précédente. D'autres sujets qui devraient être abordés pour améliorer les modèles de caractéristiques sont brièvement présentés ci-dessous :

- La redondance pour représenter certaine information (par exemple, caractéristiques optionnelles et obligatoires peuvent aussi être représentées par des cardinalités individuelles [0..1] et [1..1], respectivement) ou la sémantique des relations de type « requiert » et « optionnelle » une fois transformés dans un langage mathématique. Ce dernier fait se produit, car après la transformation en langage mathématique, la hiérarchie disparaît. Néanmoins, la présence d'une hiérarchie (relations optionnelles) ou pas (relations de type requiert) constitue la seule différence entre elles.
- Le fait de pouvoir représenter dans un langage commun, pas forcément graphique, les contraintes complexes entre les caractéristiques ou celles où interviennent des attributs.
- Des questions telles que la gestion des remplacements et des évolutions, et la question récurrente : « Qu'est-ce qu'une caractéristique ? », restent ouvertes même si quelques travaux ont conduit à des progrès. Par exemple, Thum et al. [10] font la distinction entre caractéristiques « abstraites » (pour structurer le modèle en introduisant des éléments auxiliaires) et « concrète » (éléments réutilisables ayant un impact direct sur la mise en œuvre de la ligne de produits) ; néanmoins, il est difficile de savoir si une caractéristique est un composant de la ligne de produits ou tout simplement une décision à prendre dans le processus de configuration.

## RÉFÉRENCES

- [1] D. Batory : *Feature models, grammars, and propositional formulas* ; in Software Product Lines Conference, volume 3714 of Lecture Notes in Computer Sciences, pages 7–20. Springer-Verlag, 2005.
- [2] K. Czarnecki, S. Helsen et U. W. Eisenecker : *Formalizing cardinality-based feature models and their specialization* ; Software Process: Improvement and Practice, 10(1), pages 7–29, 2005.
- [3] C. Dumitrescu, R. Mazo, C. Salinesi et A. Dauron : *Bridging the gap between product lines and systems engineering: An experience in variability management for automotive model-based systems engineering* ; 17<sup>th</sup> International Software Product Line Conference (SPLC), Tokyo, Japon, 26-30 août 2013.
- [4] C. Dumitrescu, A. Dauron, C. Salinesi et R. Mazo : *La réutilisation en ingénierie de systèmes à base de modèles – Processus et activités IS pour l'adoption d'une approche lignes de produits* ; Génie Logiciel, n° 105, pages 52-59, Paris, juin 2013.
- [5] K. Kang, S. Cohen, J. Hess, W. Novak et S. Peterson : *Feature-oriented domain analysis (FODA) feasibility study* ; Technical Report CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University, novembre 1990.

- [6] K. Kang, S. Kim et J. Lee : *FORM: A feature-oriented reuse method with domain-specific reference architectures* ; Annals of Software Engineering, 5, pages 143-168, 1998.
- [7] R. Mazo, C. Salinesi et D. Diaz : *Recommendation heuristics for improving product line configuration processes* ; in Recommendation Systems in Software Engineering. M. Robillard, W. Maalej, R. Walker et T. Zimmermann (réds.), ISBN 978-3-642-45135-5, Springer-Verlag Berlin Heidelberg, 2014.
- [8] M. Riebisch, K. Böllert, D. Streitferdt et L. Philippow : *Extending feature diagrams with UML multiplicities* ; in Proceedings of the Sixth Conference on Integrated Design and Process Technology (IDPT), Pasadena, CA, juin 2002.
- [9] C. Salinesi, R. Mazo, O. Djebbi, D. Diaz et A. Lora-Michiels : *Constraints: The core of product line engineering* ; Fifth IEEE International Conference on Research Challenges in Information Science (RCIS), Guadeloupe, France, 19-21 mai 2011, IEEE Press, pp. 1-10.
- [10] T. Thum, Ch. Kastner, S. Erdweg et N. Siegmund : *Abstract features in feature modeling* ; Software Product Lines - 15<sup>th</sup> International Conference, SPLC 2011, Munich, Allemagne, 22-26 août 2011.



Raúl Mazo est Maître de conférences à l'Université Panthéon – Sorbonne et membre du Centre de Recherche en Informatique (CRI). Il a obtenu son diplôme d'Ingénieur en Informatique en 2005, puis son master recherche en 2008 et son doctorat en 2011, dans le même domaine. Il a travaillé dans le secteur privé et public en tant qu'analyste-développeur, ingénieur en télécommunications, chercheur et enseignant.

Raúl Mazo travaille sur les sujets suivants : l'ingénierie des exigences, le développement Web à base de composants réutilisables, les systèmes ERPs, l'ingénierie de lignes de produits (dynamiques) et la gestion de la variabilité de systèmes. Il a publié plus de 40 travaux dans des conférences et revues scientifiques, et participe actuellement à trois projets de recherche sur ces sujets. Raúl Mazo a eu quatre prix pour ses travaux de recherche dans des conférences scientifiques et deux distinctions pour son engagement et son travail dans l'enseignement supérieur.

Adresse : Centre de Recherche en Informatique (CRI), Université Paris 1 Panthéon – Sorbonne, 90 rue de Tolbiac, 75013 Paris - France

Courriel : raul.mazo@univ-paris1.fr