

Optimisation de la compilation de règles métier probabilistes à l'aide de PRM

Gaspard Ducamp, Philippe Bonnard, Christian de Sainte Marie, Christophe Gonzales, Pierre-Henri Wuillemin

▶ **To cite this version:**

Gaspard Ducamp, Philippe Bonnard, Christian de Sainte Marie, Christophe Gonzales, Pierre-Henri Wuillemin. Optimisation de la compilation de règles métier probabilistes à l'aide de PRM. 9èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilistes (JFRB 2018), May 2018, Toulouse, France. hal-01974967

HAL Id: hal-01974967
https://hal.science/hal-01974967

Submitted on 9 Jan 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.

Optimisation de la compilation de règles métier probabilistes à l'aide de PRM

Gaspard Ducamp [1,2], **Philippe Bonnard** [2],
Christian De Sainte Marie [2], **Christophe Gonzales** [1],
Pierre-Henri Wuillemin [1]

1. *LIP6 (UMR 7606), Sorbonne Université, 4 place Jussieu, 75005 Paris, France*
   prenom.nom@lip6.fr

2. *IBM France Lab, 9 rue de Verdun, 94250 Gentilly, France*
   philippe.bonnard@fr.ibm.com,csma@fr.ibm.com,gaspard.ducamp@ibm.com

RÉSUMÉ. *Largement adoptées depuis plus de 20 ans par le monde de l'industrie, les règles métiers (business rules) offrent la possibilité à des utilisateurs non-informaticiens de définir des politiques de prise de décision de manière simple et intuitive. Pour faciliter leurs utilisations, des systèmes à base de règles, dits « systèmes de gestion des règles métier », ont été développés, séparant la logique métier de l'application informatique. S'ils sont adaptés pour traiter des données structurées et complètes, ils ne permettent pas aisément de travailler sur des données probabilistes.*

Cette étude propose une nouvelle approche pour l'intégration de raisonnement probabiliste dans IBM Operational Decision Manager (ODM), le système de gestion des règles métier développé par IBM, notamment via l'introduction d'une notion de risque, complexifiant la compilation mais augmentant l'expressivité des règles métiers.

ABSTRACT. *Widely adopted for more than 20 years in industrial fields, business rules offer the opportunity to non-IT users to define decision-making policies in a simple and intuitive way. To facilitate their use, systems known as Business Rule Management Systems have been developed, separating the business logic from the application one. While suitable for processing structured and complete data, BRMS face difficulties when those are incomplete.*

This study proposes a new approach for the integration of probabilistic reasoning in IBM Operational Decision Manager (ODM), IBM's BRMS, especially through the introduction of a notion of risk, making the compilation phase more complex but increasing the expressiveness of business rules.

MOTS-CLÉS : *Règle métier, Système de Gestion de Règle Métier, Aide à la décision, Réseau Bayésien, Inférence Bayésienne, Modèle objet, Modèle Graphique Relationnel*

KEYWORDS: *Business Rule, Business Rule Management System, Decision Making, Bayesian Networks, Bayesian Inference, Object Model, Probabilistic Relational Models*

1. Présentation générale

Les systèmes de gestion des règles métier ont été introduits dans les années 90 de manière à faciliter l'édition, le débogage, le déploiement et l'exécution de politiques métier par une équipe experte sous la forme d'un ensemble de règles métiers de type "si condition(s) alors action(s)". Syntaxiquement proche du langage métier, ces dernières permettent de traduire aisément prises de décision et stratégies d'entreprise, les rendant accessibles à des utilisateurs experts non-informaticiens. Un modèle de données objet décrit les différents objets concernés par les règles, ces différents objets étant instanciés de manière dynamique dans une mémoire de travail lors de l'exécution du programme. C'est à partir de celle-ci que des règles seront ou non exécutées, cette décision se faisant grâce à des algorithmes de type Rete (Forgy, 1982 ; Berstel Da Silva, 2012 ; *RetePlus*, s. d.).

PROCOP (Probabilistic Rules Optimized ComPilation) est une thèse industrielle s'inscrivant dans la continuité de travaux déjà effectués au sein d'IBM France Lab se fondant sur une hybridation entre règles métiers et modèles graphiques probabilistes comme les réseaux bayésiens (El Ghali *et al*., 2012) ou les modèles relationnels probabilistes (Agli, 2017).

Un réseau bayésien est une représentation compacte d'une distribution de probabilité jointe sur un ensemble de variables aléatoires. Celles-ci y apparaissent sous forme de nœuds dans un graphe sans circuit orienté (DAG) dont l'absence d'arcs représente des indépendances conditionnelles. Ce type de structure est utilisée comme outil d'aide à la décision dans de nombreux systèmes experts et applications (Pearl, 1988 ; Weber *et al*., 2012).

Les modèles relationnels probabilistes (*Probabilistic Relational Model*, PRM) lient quant à eux les notions apportées par les réseaux bayésiens et le paradigme des langages orientés objets (Torti *et al*., 2010). Aux notions de variables aléatoires et de probabilités conditionnelles viennent s'ajouter celles de classes, de relations, d'interface, d'héritage et d'instanciations. Une telle expressivité permet notamment de répondre à des problèmes de réutilisabilité et de scalabilité des modèles graphiques (Medina-Oliva *et al*., 2010).

Plusieurs études internes dont (Ait-Kaci, Bonnard, 2011) et une thèse (Agli, 2017) ont montré qu'un couplage faible entre un système de règles métier développé par IBM, *IBM Operational Decision Manager* (ODM), et des modèles graphiques probabilistes (réseaux bayésiens dans un premier temps, PRM dans un second) permettait de raisonner sur des données probabilistes en introduisant la notion de règles de production probabilistes.

La figure 1 montre comment ces travaux s'insèrent dans la chaîne de compilation d'ODM. Différents arbres sémantiques y sont générés après avoir analysé et vérifié la syntaxe du fichier utilisateur décrivant les règles et les objets du modèle (*ARL*). En plus de produire celui décrivant les règles (*SemRuleset*) ces études proposaient d'utiliser le modèle objet afin de générer un modèle graphique (*SemPGM*), puis de

réécrire les expressions probabilisées des règles pour pouvoir faire des appels à un moteur d'inférence (JSmile, aGrUM, ...) à l'aide du fichier généré (*PGM file*) et d'en améliorer les algorithmes (Agli *et al.*, 2016). Une fois les règles réécrites les modèles sont compilés dans un langage proche de Java (IROS) avant d'être traduit en bytecode dans une archive (*JAR*), facilitant le déploiement et l'exécution sur les machines cibles.

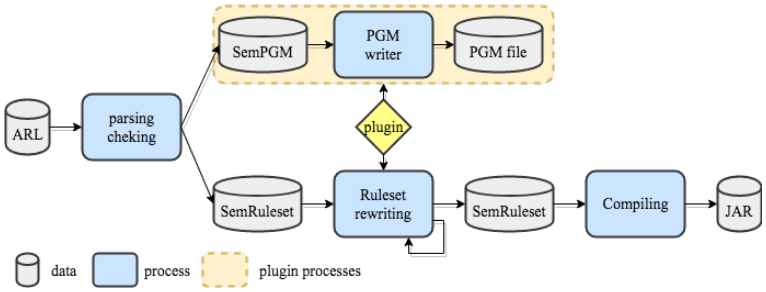


Figure 1. Représentation de la chaine de compilation d'ODM

Un des exemples développés dans la thèse mentionnée ci-dessus était celui d'une application pour une assurance, un organisme étant chargé de gérer les demandes de remboursements faites par ses clients en fonction de leurs natures (type et coût du remboursement). Un ensemble de professionnels de santé validant pour cela les requêtes émises par les clients.

La figure 2 montre un exemple de schéma de relation de classes PRM pour ce problème. Une classe HealthcareProfessional décrit les professionnels de santé à partir de plusieurs caractéristiques, notamment leur âge, leur localisation, leur sexe et la liste de clients qu'ils gèrent (Subscriber), eux-mêmes caractérisés par un âge et une liste de remboursements (Reimbursement).

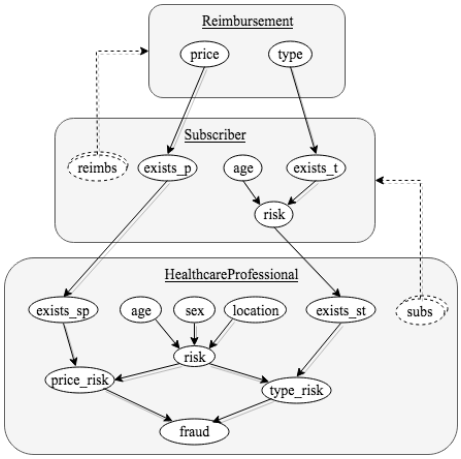


Figure 2. Exemple de schéma de relation de classes dans un PRM

Généré lors de la compilation, ce modèle permet de répondre à un premier ensemble de règles probabilistes. La règle ci-dessous, par exemple, indique qu'il faut marquer comme à risque les clients d'un professionnel de santé ayant une probabilité importante (>80%) de participer à de la fraude sur certains types de remboursements.

Exemple 1. Une règle métier probabiliste

```
1:  rule detectInvoiceAnomaly{
2:      when{
3:        hp: HealthcareProfessional(probability(type_risk==high)>.8);
4:        sub: Subscriber () in hp.subs;
5:      }
6:      then{
7:          update sub{risk=true;}
8:      }
9:  }
```

Les travaux précédents étant des études de faisabilités certaines limitations en surgissent :

– de telles règles sont difficiles à définir et à comprendre pour un utilisateur expert, le fait d'exprimer une probabilité sur des conditions particulières demandant un niveau de connaissance des modèles probabilistes utilisés important ;

– ODM offre la possibilité aux utilisateurs de définir différents types de conditions (à l'aide de filtres, d'agrégateurs et d'imbrications de conditions, par exemple). Ni ces constructions, plus complexes, ni leurs impacts sur les performances n'ont été étudiés ;

– l'intérêt de l'utilisation de l'algorithme IJTI (Agli *et al.*, 2016) vient de l'hypothèse qui a été faite que les modifications incrémentales de la mémoire de travail n'allaient pas avoir de gros impacts sur la structure de l'arbre de jonction utilisé pour les inférences, d'importantes économies de calculs pouvaient donc y être réalisées. IJTI est donc surtout adapté aux grandes structures connexes ;

– l'algorithme IJTI travaille avec des PRM, mais retranscrits sous la forme de réseaux bayésiens (dits « groundés »). L'intérêt des PRM en est fortement diminué puisque les redondances structurelles de ces réseaux groundés, qui sont encodées dans les PRM qui les ont générés, ne sont pas exploitées, ce qui réduit l'efficacité des inférences ;

– le modèle graphique construit lors de la compilation d'ODM prend en compte un modèle probabilisé des relations entre les objets de la mémoire de travail mais il ne s'adapte pas aux requêtes probabilistes définies dans les règles.

2. Vers une nouvelle définition des règles probabilistes

Pour répondre aux problématiques soulevées ci-dessus, nous souhaitons dans un premier temps redéfinir le traitement de l'incertitude dans l'expression des règles, en remplaçant les seuils de probabilités attachés à des variables particulières par une notion de risque acceptable sur l'évaluation de l'ensemble des conditions d'une règle.

Cela permettra à nos règles probabilistes d'être à la fois plus complexes et intuitives, mais demande une redéfinition de la phase de compilation d'ODM pour redistribuer le risque global sur chaque condition particulière. La partie action d'une règle ne sera donc exécutée que si l'ensemble des conditions est vérifiée avec une probabilité supérieure au risque défini.

Dans la règle ci-dessous la partie **then** ne sera exécutée que si la probabilité que les conditions c1 et c2 soient vraies est supérieure 80%. Les différents prédicats servant à conditionner les valeurs prises par les attributs (dire que *hp.type_risk* est supérieur à moyen ou que *hp.location* est égal à Paris ou Londres, par exemple).

Exemple 2. Un nouveau type de règle métier probabiliste

```
1:  rule detectInvoiceAnomaly {
2:      probability > .8;
3:      when {
4:        hp: HealthcareProfessional (predicat1(hp.type_risk) && predicat2(hp.location)); (c1)
5:        r: Reimbursement (predicat3(r.type)) in hp.sub.reimb;                          (c2)
6:      } then {…}
7:  }
```

Pour ce faire, il s'agit de générer le PRM non seulement à partir des modèles objets mais aussi de l'ensemble des règles (cf. Figure 3). Il faut pour cela étudier les différentes formes que peuvent prendre les conditions (agrégateurs, filtres, imbrications, …) et leurs prédicats. Une étude de faisabilité doit, de ce fait, être effectuée pour définir les limites théoriques et pratiques d'une telle méthode.

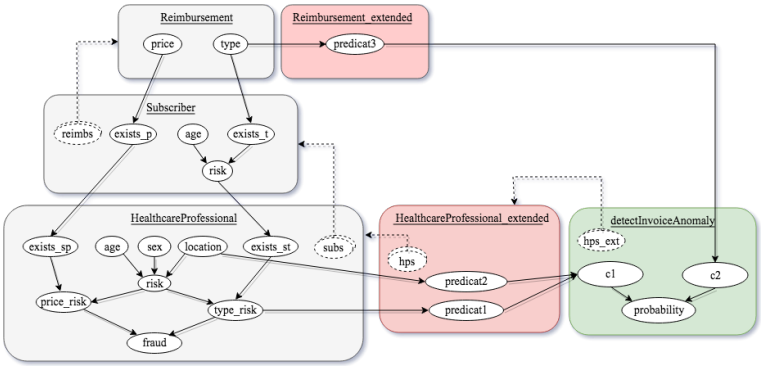


Figure 3. Exemple de PRM généré à la suite de l'analyse de la règle en Exemple 2

Les règles seront par la suite réécrites de manière à prendre en compte nos nouvelles requêtes probabilistes (sur l'attribut *probability* de la classe *detectInvoiceAnomaly*, par exemple). De telles manipulations pourraient cependant modifier l'efficience du pattern matching opéré par ODM pour la sélection de règles à effectuer, il conviendra donc d'en étudier l'impact.

Côté PRM, l'instanciation dynamique de classes représentant des règles (en vert dans la Figure 3) ainsi que la simplification du modèle généré lors de la compilation (les données sont-elles probabilistes ou connues ?) seront au cœur du travail.

Du fait de l'aspect fortement déterministe des modèles utilisateurs, nous ferons l'hypothèse que l'instanciation de nos PRMs donnera un grand nombre de petits sous-graphes disjoints (un par professionnel de santé, par exemple), ce qui nécessiterait de travailler sur un nouvel algorithme d'inférence profitant de cette redondance structurelle.

Ce travail de développement se fonde sur ce qui a été fait dans le plugin BIS (*Bayesian Insight Service*) développé dans le cadre de la thèse d'Agli (Agli, 2017). La figure 4 illustre comment notre nouveau module, PRIME (*Probabilistic Reasoning Insight ModulE*), s'insère dans la chaine de compilation d'ODM. Il intervient directement lors du processus de réécriture de l'arbre sémantique décrivant les règles (*SemRuleset*) mais, contrairement à BIS, étend donc la définition et l'optimisation du modèle graphique à partir des règles avant de réécrire celles-ci (via le processus *PRM enhancement*).

Une fois les règles réécrites le modèle graphique est sérialisé afin d'être utilisable par un moteur probabiliste en parallèle d'ODM.

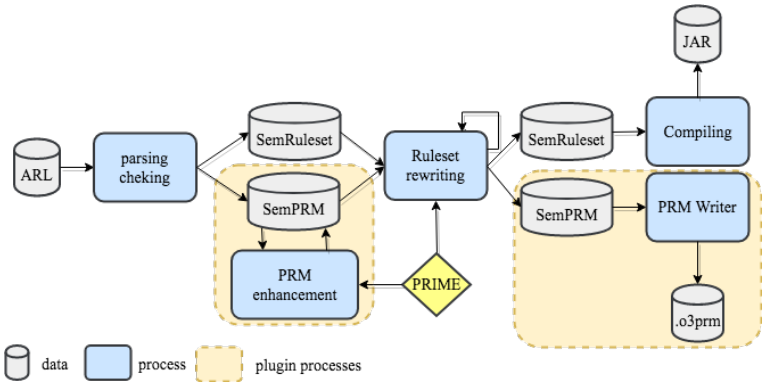


Figure 4. Utilisation du plugin PRIME dans la chaîne de compilation d'ODM

Dans le cadre de nos travaux, ODM utilise aGrUM [1] pour les calculs probabilistes et la manipulation des PRM. La complexité de la compilation ainsi que des problèmes de performances lors de l'exécution pourraient nécessiter de travailler sur une extension de cette libraire (agrégateurs avancés, nœuds fonctionnels, nœuds paramétriques, ...).

1. aGrUM (A GRaphical Universal Modeler) est une libraire C++ open source permettant la manipulation (apprentissage, modélisation, inférence) de données probabilistes et implémentant O3PRM : http://agrum.gitlab.io

3. Application et extensions

Une étude sur l'utilisation de modèles temporels via une extension des capacités de modélisation des PRM dans O3PRM [2] pourrait-être effectuée dans un second temps, ce qui permettrait un raisonnement analogue mais dans le cadre du traitement des événements complexes (Complex Event Processing, CEP ; cf. (Luckham, 2002)). L'utilisation de règles avec des expressions temporelles étant à la base des capacités de raisonnement de IBM Decision Server Insights, une application de CEP développée par IBM. Cette étude serait donc utile à la fois sur le plan théorique mais également sur le plan technique et industriel.

Un des avantages de l'expressivité des PRM est qu'il est possible de prendre en compte l'incertitude structurelle d'un modèle (Getoor *et al*., 2000). Ce faisant, une étude sur une extension prédictive d'ODM pourrait-être réalisée. Le système serait alors capable de raisonner au-delà de la mémoire de travail, permettant de définir de nouveaux types de règles (d'anticipation, par exemple).

Pour finir, ce travail devra fournir une extension du langage de définition de règles métier d'ODM aux concepts probabilistes. Syntaxiquement et sémantiquement proche du langage naturel celui-ci devra permettre à des non-informaticiens de définir des règles métiers probabiliste de manière intuitive.

Bibliographie

Agli H. (2017). *Raisonnement incertain pour les règles métiers*. Thèse de doctorat non publiée, Sorbonne Université.

Agli H., Bonnard P., Gonzales C., Wuillemin P.-H. (2016, juin). Incremental Junction Tree Inference. In *IPMU16*. Eindhoven, Netherlands. Consulté sur http://hal.upmc.fr/hal-01345418

Ait-Kaci H., Bonnard P. (2011). *Probabilistic production rules*. Rapport technique. IBM.

Berstel Da Silva B. (2012). *Verification of business rules program*. Thèse de doctorat non publiée, Université de Fribourg-en-Brisgau.

El Ghali A., Bonnard P., El Ghali K., Hromada D., Ait-Kaci H. (2012). *Règles de production et réseaux bayésiens pour l'extraction de mots clés*. Rapport technique.

Forgy C. L. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence*, vol. 19, n° 1, p. 17 - 37. Consulté sur http://www.sciencedirect.com/science/article/pii/0004370282900200

Getoor L., Koller D., Taskar B., Friedman N. (2000). Learning probabilistic relational models with structural uncertainty. In *Proc. of the aaai-2000 workshop on learning statistical models from relational data,technical report ws-00–06*, p. 13–20. Menlo Park, CA., AAAI Press.

2. O3PRM est un framework développé dans le cadre d'aGrUM permettant la manipulation de PRM : http://o3prm.gitlab.io

Luckham D. (2002). *The power of events: An introduction to complex event processing in distributed enterprise systems*. Addison Wesley.

Medina-Oliva G., Weber P., Levrat E., Iung B. (2010, juillet). Use of probabilistic relational model (PRM) for dependability analysis of complex systems. In *12th IFAC Symposium on Large Scale Systems: Theory and Applications, LSS 2010*, p. CDROM. Villeneuve d'Ascq, France. Consulté sur https://hal.archives-ouvertes.fr/hal-00549877

Pearl J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufman.

Reteplus. (s. d.). Consulté sur https://www.ibm.com/support/knowledgecenter/SSQP76_8.9.0/com.ibm.odm.dserver.rules.designer.run/optimizing_topics/tpc_opt_reteplusalgo.html

Torti L., Wuillemin P.-H., Gonzales C. (2010, septembre). Reinforcing the Object-Oriented Aspect of Probabilistic Relational Models. In *PGM 2010 - The Fifth European Workshop on Probabilistic Graphical Models*, p. 273-280. Helsinki, Finland. Consulté sur https://hal.archives-ouvertes.fr/hal-00627823

Weber P., Medina-Oliva G., Simon C., Iung B. (2012). Overview on bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, vol. 25, n° 4, p. 671 - 682. Consulté sur http://www.sciencedirect.com/science/article/pii/S095219761000117X (Special Section: Dependable System Modelling and Analysis)