

# Les ontologies pour l'optimisation

*Mina Ouabiba*

## 1 Introduction

L'optimisation globale est une branche particulièrement active des mathématiques appliquées car elle correspond à un besoin industriel. Plusieurs applications dans plusieurs domaines (automatique, biochimie, systèmes de production, etc) se ramènent à la recherche d'optima globaux. Nous nous intéressons à la résolution de problèmes continus non linéaires avec contraintes en utilisant un système basé sur la coopération de différents solveurs. Depuis quelques années, plusieurs systèmes coopératifs ont été développés. Mais, il n'existe pas d'architectures de coopération génériques [1].

L'ingénierie de connaissance s'est orientée vers la construction de bibliothèques de composants génériques réutilisables afin de minimiser le temps de développement, et de faciliter leur évolutions et leur maintenances. La plupart des méthodologies existantes (CommonKADS, Protégé II,..) décrivent une base de connaissances en utilisant les trois concepts: tâche, PSM (méthodes de résolution de problèmes) et domaine. Une telle séparation entre le traitement à faire, le raisonnement utilisé pour réaliser un traitement donné et les données du domaine d'application nous permet de voir la construction d'une base de connaissances comme un assemblage de ces trois composants préalablement décrits et testés (figure1).

Nous allons modéliser les méthodes coopératives d'optimisation non linéaires basées sur l'analyse d'intervalle [2] et les techniques de satisfaction de contraintes. Ce modèle intègre les trois concepts tâche, méthodes de résolution de problèmes et domaine, ainsi que toutes les relations sémantiques entre ces trois concepts. Ces relations seront décrites comme des entités indépendantes des composants qu'elles lient donc pouvant être elles mêmes réutilisées. la description de ce modèle sera réalisée à l'aide de la notion d'ontologie.

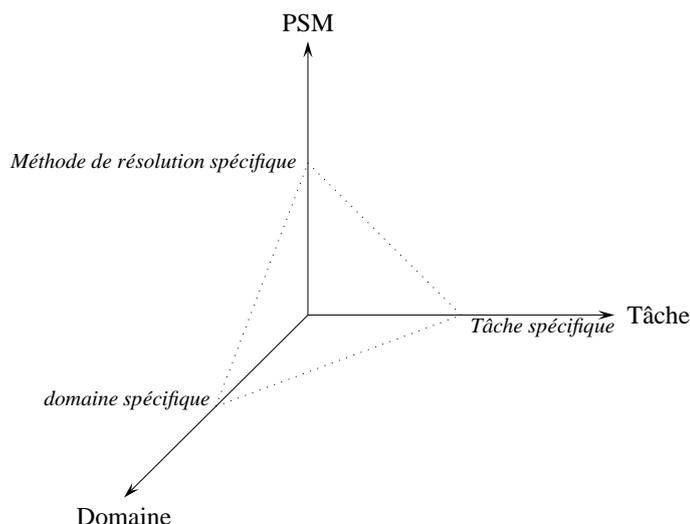


FIG. 18 – Application ( ensemble de composants )

## 2 Présentation du problème

Considérons une fonction  $f : D \subseteq \mathfrak{R}^n \rightarrow \mathfrak{R}$  appelée critère, ou fonction objectif. On se donne un ensemble de contraintes :

$$C = \{x \in \mathfrak{R}^n / g_i(x) = 0, i \in I, h_j(x) \leq 0, j \in J\}$$

où les fonctions  $g_i$  et  $h_j : \mathfrak{R}^n \rightarrow \mathfrak{R}$  sont définies sur  $\mathfrak{R}^n$ . On définit un problème d'optimisation par un triplet  $(f, C, D)$  ou par un couple  $(f, \Omega)$  tel que  $\Omega$  représente l'ensemble des points admissibles qui est défini par :

$$\Omega = \{x \in D / x \in C\}$$

**Définition 1 :** On appelle **solution optimale d'un problème d'optimisation**  $P=(f, C, D)$  (ou optimum global) une solution qui optimise ( maximise ou minimise )  $f(x)$  sur l'ensemble de toutes les solutions.

**Définition 2 :** On dit qu'un vecteur  $x$  est un optimum local de  $P = (f, C, D)$  si et seulement s'il existe un voisinage  $V(x) \subseteq \Omega$  de  $x$  tel que  $x$  soit un optimum global du problème.

Notre objectif est de construire une bibliothèque de composants génériques. Les approches orientées bibliothèque de composants qui ont été définies que ce soit en ingénierie de connaissances ou en génie logiciel décrivent souvent des composants de même type ou bien spécifiques à une application. Une bibliothèque qui intègre des composants de différents type et les informations permettant de les assembler élargirait le champ de son utilisation et

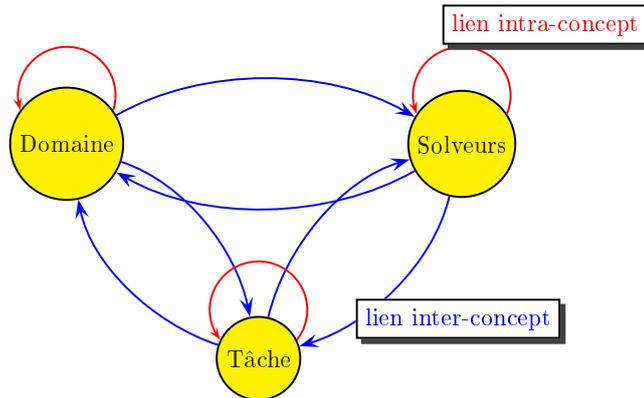


FIG. 19 – *Les concepts et les liens*

réutilisation. Les composants seront décrits de manière indépendante les uns des autres en utilisant la notion d’ontologie dans un souci de réutilisation et de partage.

Nous allons tout d’abord définir ce qu’est une ontologie. La définition la plus utilisée est celle de Gruber [3] “une ontologie est la spécification d’une conceptualisation” . Elle définit le vocabulaire et la sémantique associée qui permettent à deux agents de communiquer sur un domaine de connaissances donné. Concrètement, une ontologie est une représentation d’un domaine à l’aide d’une hiérarchie de relations de généralisation et de spécifications de concepts et un ensemble de relations sémantiques entre ces concepts.

### 3 Description du modèle de coopération

Notre modèle se base sur trois concepts qui sont la tâche, les méthodes de résolution de problèmes et le domaine.

- La tâche: spécifie une action et une méthode pour réaliser l’action qui définit le but poursuivi par la tâche.
- La méthode de résolution de problèmes: représente la méthode qui réalise une tâche. Elle peut être une primitive de résolution ou bien composée de sous tâches.
- Domaine: représente la description des données d’un domaine particulier.

Ces trois concepts sont liés pour construire une application. Afin de pouvoir réutiliser ces trois concepts, par exemple réutiliser une méthode de résolution pour réaliser plusieurs tâches dans différents domaines, des liens sémantiques inter-concepts et intra-concepts ont été définis (figure 2).

Un lien inter-concepts est un lien entre deux concepts différents par exemple la relation entre une tâche qui consiste à chercher un optimum local et une méthode d'exploration locale.

Un lien intra-concepts est un lien entre deux concepts de même type par exemple la relation qui définit la transformation symbolique de la fonction objectif et les fonctions qui définissent les contraintes d'un problème d'optimisation. Cette transformation permet de réduire le nombre d'occurrences des variables afin d'augmenter la précision des calculs d'intervalles.

Au niveau ontologique, une tâche est spécifiée par son nom, les données d'entrée, les données de sortie, son but c'est à dire le problème à résoudre ainsi que les liens qui lui sont attachés. La méthode de résolution de problèmes est spécifiée par son nom, son entrée, sa sortie, ses compétences (par exemple, une méthode locale ou globale) à résoudre un problème, toutes les informations sémantiques (par exemple la convergence d'une méthode d'optimisation nécessite la convexité de la fonction objectif) et les liens qui lui sont attachés. La description d'un domaine pour différentes tâches et différentes méthodes de résolution doit être faite suivant plusieurs niveaux de généralité. L'ontologie de domaine est décrite par les concepts d'un domaine tels que les relations sémantiques inter-concepts, les propriétés de ces relations ( binaire, transitive, etc ), les informations sémantiques ( contraintes linéaires, quadratiques, fonction objectif convexe, etc ) et les différents liens qui lui sont attachés. Un lien sémantique décrit une relation entre deux concepts qu'elle lie. Elle est dotée d'une sémantique et d'un comportement propre permettant aux entités liées de communiquer et de collaborer.

## 4 Conclusion

Nous avons décrit d'une manière générale notre modèle qui définit une bibliothèque permettant d'intégrer des tâches, des PSM et des domaines. Pour cela, nous nous sommes basés sur les travaux effectués dans le domaine du génie logiciel et de la représentation de connaissances.

## Références

- [1] *L. Granvilliers, E. Monfroy and F. Benhamou*, Symbolic-Interval Cooperation in Constraint Programming. In Proceedings of ISSAC'2001, 26th International Symposium on Symbolic and Algebraic Computation, 2001
- [2] *E. Hansen*, Global optimization using interval analysis, Marcel Dekker, 1992.
- [3] *T.R GruberE*, Toward Principles for the Design of Ontologies Used for Knowledge sharing. In Nicola Guarino and Roberto Poli, editors, Formal Ontology in Conceptual Analysis and Knowledge Representation . Kluwer Academic Publishers , 1993.

*Mina Ouabiba*

Institut de recherche en Informatique de Nantes  
Université de Nantes - Faculté des Sciences et Techniques  
2, rue de la Houssinière, BP 92208  
44322 Nantes Cedex 03  
France  
`ouabiba@irin.univ-nantes.fr`