

Thèse de Doctorat

Alejandro
REYES AMARO

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université de Nantes
sous le sceau de l'Université Bretagne Loire*

École doctorale : Sciences et technologies de l'information, et mathématiques

Discipline : Informatique et applications, section CNU 27

Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)

Soutenue le 23 janvier 2016

POSL: A Parallel-Oriented Solver Language

JURY

Rapporteurs :	M. Salvador ABREU , Professeur, Université d'Évora M. Christophe LECOUTRE , Professeur, Université d'Artois
Examineurs :	M. Arnaud LALLOUET , Ingénieur de recherche, Huawei Technologies Ltd. M. Frédéric LARDEUX , Maître de conférences, Université d'Angers
Directeur de thèse :	M. Éric MONFROY , Professeur, Université de Nantes
Co-encadrant de thèse :	M. Florian RICHOUX , Maître de conférences, Université de Nantes

Thèse de Doctorat

**Alejandro
REYES AMARO**

POSL: Un Langage Orienté Parallèle pour construire des Solveurs de contraintes

POSL: A Parallel-Oriented Solver Language

Résumé

La technologie multi-cœur et les architectures massivement parallèles sont de plus en plus accessibles à tous, à travers des technologies comme le Xeon Phi ou les cartes GPU. Cette stratégie d'architecture a été communément adoptée par les constructeurs pour faire face à la loi de Moore. Or, ces nouvelles architectures impliquent d'autres manières de concevoir et d'implémenter les algorithmes, pour exploiter complètement leur potentiel, en particulier dans le cas des solveurs de contraintes traitant de problèmes d'optimisation combinatoire. De plus, le temps de développement nécessaire pour coder des solveurs en parallèle est souvent sous-estimée, et concevoir des algorithmes efficaces pour résoudre certains problèmes consomme trop de temps. Dans cette thèse nous présentons le langage orienté parallèle POSL, permettant de construire des solveurs de contraintes basés sur des méta-heuristiques qui résolvent des Problèmes de Satisfaction de Contraintes. Le but de ce travail est d'obtenir un système pour facilement construire des solveurs et réduire l'effort de leur développement en proposant un mécanisme de réutilisation de code entre les différents solveurs. Il fournit aussi un mécanisme pour coder des stratégies de communication indépendantes des solveurs. Dans cette thèse, nous présentons aussi une analyse détaillée des résultats obtenus en résolvant plusieurs instances des CSPs. L'idée n'est pas d'améliorer l'état de l'art en terme d'efficacité sur ces instances de CSPs, mais de démontrer qu'il est possible de rapidement écrire des prototypes avec POSL afin d'expérimenter facilement différentes stratégies de communication.

Mots clés

CSP, méta-heuristiques, parallèle, communication entre processus, langage.

Abstract

The multi-core technology and massive parallel architectures are nowadays more accessible for a broad public through hardware like the Xeon Phi or GPU cards. This architecture strategy has been commonly adopted by processor manufacturers to stick with Moore's law. However, this new architecture implies new ways of designing and implementing algorithms to exploit their full potential. This is in particular true for constraint-based solvers dealing with combinatorial optimization problems. Furthermore, the developing time needed to code parallel solvers is often underestimated. In fact, conceiving efficient algorithms to solve certain problems takes a considerable amount of time. In this thesis we present POSL, a Parallel-Oriented Solver Language for building solvers based on meta-heuristic in order to solve Constraint Satisfaction Problems (CSP) in parallel. The main goal of this thesis is to obtain a system with which solvers can be easily built, reducing therefore their development effort by proposing a mechanism for code reusing between solvers. It also provides a mechanism to implement solver-independent communication strategies. We also present a detailed analysis of the results obtained when solving some CSPs. The goal is not to outperform the state-of-the-art solvers in terms of efficiency, but to show that it is possible to rapidly prototype with POSL in order to experiment different communication strategies.

Key Words

CSP, meta-heuristics, parallel, inter-process communication, language.