# Méthode de Recherches Successives pour la Résolution des Problèmes d'Optimisation

**Mohamed TOUNSI**
Ecole des Mines de Nantes,
4, Rue Alfred Kastler.
La Chantrerie, 44307 NANTES.
email: `mohamed.tounsi@emn.fr`

**Philippe DAVID**
Ecole des Mines de Nantes,
4, Rue Alfred Kastler
La Chantrerie, 44307 NANTES.
email: `philippe.david@emn.fr`

**Résumé**

Nous proposons une nouvelle méthode basée sur la méthode Russian Doll Search (RDS) pour la résolution de problèmes d'optimisation exprimés dans le cadre des problèmes de satisfaction de contraintes valuées (VCSP). La méthode RDS consiste à résoudre un problème de taille $N$ en $N$ recherches successives sur des sous-problèmes de taille $1, 2, \ldots, N$ en utilisant le résultat de chaque recherche pour produire une meilleure borne inférieure. Le principe de la méthode que nous proposons est d'introduire les variables lors des recherches successives non plus une à une mais par paquet de $k$ variables. Nous présentons deux variantes de notre méthode : une méthode de recherches successives avec un nombre de variables rentrantes fixe ($k$ est fixe) et une méthode dynamique ($k$ est variable au cours de la recherche).

## 1 Introduction

La méthode que nous proposons s'inspire de la méthode ***Russian Doll Search*** [VLS96]. L'objectif de RDS – qui résout un problème en $N$ étapes successives – est de fournir à chaque nouvelle étape une bonne borne inférieure, qui permet d'améliorer l'efficacité de la résolution des sous-problèmes suivants ; ce gain d'efficacité devant compenser le surcoût occasionné par les résolutions des $N$ sous-problèmes. Notre objectif est de déterminer s'il existe une solution intermédiaire entre RDS ($N$ sous-problèmes) et résolution directe du problème complet, qui permette d'améliorer le compromis entre nombre de sous-problèmes à résoudre et qualité des bornes inférieures obtenues (et donc efficacité de résolution des sous-problèmes). Notre méthode consiste à remplacer une recherche sur un problème de taille $N$ par $\lceil \frac{N-1}{K} \rceil + 1$ recherches successives sur des sous problèmes de taille croissante. Il s'agit à la première étape de résoudre un sous-problème à une variable, puis à chaque nouvelle étape d'introduire un paquet de $k$ variables dans la nouvelle résolution du sous-problème.

## 2 Principe de la Méthode de Recherches Successives

Le choix qui consiste à introduire à chaque fois dans la méthode RDS une nouvelle variable et lancer une nouvelle recherche peut constituer un handicap, car la résolution complète du problème correspond à la résolution successive de $N$ sous-problèmes. Le gain procuré par l'utilisation de bonnes bornes inférieures permettant un élagage efficace n'est en effet pas nécessairement supérieur au surcoût des $N$ résolutions. Dans chaque résolution on cherche une instanciation $V$ des variables du problème de valuation minimum $\varphi(V)$ telle que $BI \leq \varphi(V) \leq BS$, où $BI$ et $BS$ sont respectivement les bornes inférieure et supérieure du sous-problème courant. Suivant que la valeur de $k$ est fixe ou dynamique, deux variantes de la méthode de recherches successives peuvent en découler : *fixe* ou *dynamique*. Le principe de la version dynamique est de faire varier $k$ au cours du processus de résolution ; à chaque étape, la « meilleure » valeur de $k$ sera choisie. L'objectif n'est pas de trouver la séquence optimale pour un problème précis, mais de tenter d'analyser le comportement général de notre méthode, en fonction de l'état d'avancement de la résolution : existe-t-il des phases – début de la résolution (peu de variables), milieu de résolution, fin de résolution (presque toutes les variables) – pour lesquelles il est plus efficace de grouper les insertions ($k$ élevé), ou au contraire de les séparer ?

## 3 Expérimentations et Conclusion

Nous avons testé notre méthode sur 8 problèmes de planification journalière des prises de vue du satellite SPOT5 [ABB+95] ; nous avons tenté de trouver un meilleur compromis entre nombre de résolutions et gain d'efficacité. Un bon candidat semble être la valeur $k = 3$ (pour la variante « fixe »). Le fait que les gains ne soient pas importants pour les problèmes testés (environ 28%) est dû essentiellement au fait que la méthode RDS donne déjà de bons résultats sur ces problèmes. Nous pensons qu'une amélioration sensible de notre méthode est possible sur d'autres problèmes ou RDS échoue (ou est moins performante). Pour la variante « dynamique », nous avons réalisé peu de test pour en déduire des conclusions, mais il est clair que des expérimentations à plus grande échelle permettraient d'obtenir des résultats plus significatifs. Parallèlement, d'autres voies sont à explorer : étudier finement le comportement de notre méthode sur des problèmes aléatoires pour déterminer les meilleures valeurs de $k$ pour différentes catégories de problèmes (nombre de variables, densité du réseau de contraintes, dureté des contraintes...), coopération avec des méthodes approchées capables de faire croître le plus rapidement possible la borne inférieure.

## Références


[ABB+95] J. Agnèse, N. Bataille, E. Bensana, D. Blumstein, and G. Verfaillie. Exact and approximate methodds for the daily management of an earth observation satellite. In *The 5th ESA Workshop on artificial IOntelligence and Knowledge Based Systms for Space*, 1995.

[VLS96] G. Verfaillie, M. Lemaitre, and T. Schiex. Russian doll search for solving constraint optimization problems. In *AAAI*, Portland, Oregon, USA, 1996.