

Une heuristique de sélection de valeur dirigée par l'impact pour les WCSP

N. Levasseur¹, P. Boizumault², et S. Loudni³

GREYC, UMR60-72, Campus Côte de Nacre, boulevard du Maréchal Juin, BP 5186, 14032 Caen Cedex
{nicolas.levasseur,patrice.boizumault}@info.unicaen.fr loudni@iut3.unicaen.fr

Mots-clés : Weighted CSP, Impact, Heuristique de choix de valeur, Branch and bound, LDS

1 Introduction

Le formalisme des WCSP [1] [2] (Weighted Constraint Satisfaction Problem) est un cadre général pour modéliser et résoudre des problèmes d'optimisation sous contraintes. Les WCSP sont très souvent résolus par des méthodes arborescentes qu'elles soient complètes (Depth First Branch and Bound) ou partielles (Limited Discrepancy Search [3]) combinées avec des mécanismes de propagation (filtrage). Dans ces méthodes, les heuristiques de sélection dynamique de variable (MinDom/FutDeg [5]) et de valeur (MinAC) permettent de guider efficacement la résolution. Dans cet article, nous proposons une nouvelle heuristique de sélection de valeur basée sur la notion d'impact. Cette nouvelle heuristique améliore nettement MinAC sur les problèmes aléatoires ainsi que sur les sous-instances 6 du CELAR que nous avons testés.

1.1 Weighted Constraint Satisfaction Problem

Un WCSP est défini par un triplet (V, D, W) . $V = \{1, \dots, n\}$ est l'ensemble des *variables*, $D = \{D_1, \dots, D_n\}$ l'ensemble des *domaines* finis associés aux variables et W l'ensemble des *contraintes binaires molles* portant sur ces variables. Chaque contrainte $W_{ij} \in W$ représente une fonction $W_{ij} : D_i \times D_j \mapsto [0, k]$ où k correspond au coût maximum de violation du problème (i.e. *l'insatisfaction totale*). Une affectation complète (solution) $A=(a_1, \dots, a_n)$ est une affectation de l'ensemble des variables ; dans le cas contraire, nous parlons d'affectation partielle. Le coût de l'affectation complète $A=(a_1, \dots, a_n)$ est noté : $V(A) = \sum_{W_{ij} \in W} W_{ij}(a_i, a_j)$. Le problème consiste à trouver une affectation complète de coût minimal : $\min_{A \in D_1 \times D_2 \times \dots \times D_n} V(A)$.

1.2 Algorithme de recherche, filtrage et heuristiques dans les WCSP

Le principe de LDS est d'explorer uniquement les chemins de l'arbre de recherche ayant un nombre d'écarts borné au plus k (nommé discrepancy) par rapport à l'heuristique de sélection de valeur. A chaque noeud, ce nombre est incrémenté suivant le rang de la valeur choisie, dans le classement fourni par l'heuristique de sélection de valeur. Si ce nombre dépasse la valeur de k , l'algorithme backtrack, sinon une phase de filtrage est effectuée par un mécanisme tel que MFDAC [2]. Afin de filtrer, on associe à chaque valeur a d'un domaine d_i un compteur d'inconsistance. Ce compteur représente la somme des valuations des contraintes qui seront nécessairement violées lorsque l'affectation partielle sera étendue avec a_i . L'heuristique de sélection de valeur MinAC ordonne les valeurs suivant l'ordre croissant de ce compteur d'inconsistance en départageant les égalités suivant l'ordre croissant des valeurs (MinVal). La valeur du compteur d'inconsistance dépend fortement de l'algorithme de filtrage utilisé. L'heuristique de sélection de variable MinDom/FutDeg choisit la variable ayant le plus petit rapport de la taille du domaine sur le degré futur (égal pour une variable au nombre de contraintes portant sur celle-ci et ayant au moins une variable non affectée).

2 Sélection de valeur par l'impact

Soit H l'ensemble des solutions générées depuis le début de la recherche. On définit la notion d'impact pour a_i (notée $I(i, a)$) comme étant une estimation de la capacité de a_i à être présente dans des solutions optimales, autrement dit : $I(i, a) = \min_{A \in H \text{ et } a_i \in A} V(A)$. La pertinence de l'impact

dépend donc de la quantité et de la qualité des solutions présentes dans H . Pour diversifier les solutions et accroître la taille de H , on ajoute à celui-ci des solutions générées par des descentes gloutonnes. Chaque descente gloutonne est guidée par MinAC et seuls les compteurs d'inconsistance sont mis à jour par forward checking. Nous proposons l'heuristique, qui ordonne les valeurs suivant les impacts croissants et à égalité suivant les compteurs d'inconsistance croissants. Une autre notion d'impact basée sur la réduction des domaines des variables a été utilisée par Refalo dans le cadre des CSP [6].

La raison de ce choix, est de faire confiance à la propriété d'impact après que les descentes gloutonnes aient généré plusieurs solutions, avec l'hypothèse que plus la valuation d'une solution contenant a_i est proche de l'optimum, plus a_i a de chances de participer aux solutions optimales. Il faut donc trouver les affectations participant aux solutions de bonne qualité (voir optimale) du problème afin de guider la recherche vers ces zones.

3 Expérimentations

Nous avons comparé notre heuristique à MinAC sur des instances aléatoires de 50 variables ayant un domaine de taille 10, une densité (rapport entre le nombre de contraintes réelles et le nombre de contraintes possibles) de 80 et une dureté (pourcentage de tuples interdits par contrainte) de 50%. Les figures montrent la moyenne du temps de résolution sur 20 instances aléatoires générées avec les paramètres précédents. Nous avons également expérimenté notre heuristique sur la sous instance 3 du problème 6 du problème d'allocation de fréquence à des liens radio (RLFAP) [4] proposée par le Centre d'Electronique et de l'Armement (CELAR). L'algorithme de recherche utilisé est LDS avec l'heuristique de sélection de variable MinDom/FutDeg et le mécanisme de filtrage MFDAC.

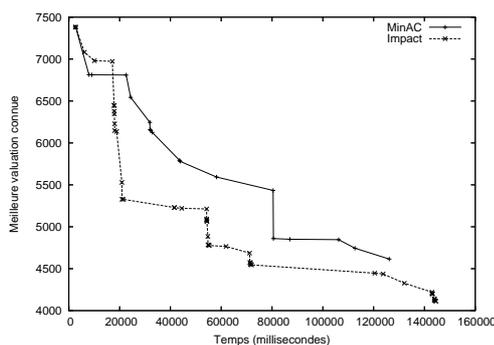


FIG. 1. Profil de performance sur l'instance 6-sub3 du celar

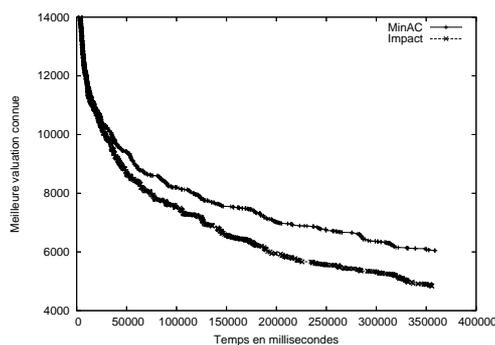


FIG. 2. Profil de performance sur les instances aléatoires $\langle 50, 10, 70, 80 \rangle$

Les résultats sur ces différentes instances sont prometteurs et demandent à être confortés sur l'ensemble des instances du CELAR. Nous travaillons actuellement sur l'extension de la notion d'impact aux heuristiques de sélection de variable.

Références

1. T. Schiex, H. Fargier et G. Verfaille : Valued Constraint Satisfaction Problems : Hard and Easy Problems. IJCAI, 631-639. (1995).
2. J. Larrosa et T. Schiex : In the quest of the best form of local consistency for Weighted CSP. IJCAI, 239-244. (2003).
3. W. D. Harvey et M. L. Ginsberg : Limited Discrepancy Search. IJCAI, 607-615. (1995).
4. B. Cabon, S. De Givry, L. Lobjois, T. Schiex, et al. : Radio Link Frequency Assignment. Constraints, 4(1), 79-89. (1999).
5. Barbara M. Smith et Stuart A. Grant : Trying Harder to Fail First. ECAI, 249-253. (1998).
6. Philippe Refalo : Impact-Based Search Strategies for Constraint Programming. CP, 557-571. (2004).