

Recherche Réactive pour l'ordonnancement de tâches sur deux processeurs dédiés

Méziane Aïder¹, Fatma Zohra Baatout¹, Mhand Hifi²

¹ LaRoMad, USTHB, Algérie

{m-aider, fbaatout}@usthb.dz

² EPROAD, UPJV, Amiens, France

hifi@u-picardie.fr

Mots-clés : *ordonnancement, optimisation, processeurs dédiés, recherche réactive.*

1 Introduction

Le problème d'ordonnancement de tâches sur deux processeurs dédiés (noté ST2P) est un problème d'optimisation combinatoire NP-difficile (cf. Hoogeveen *et al.* [4]), dont l'objectif est d'assigner les tâches disponibles à deux processeurs différents. Le problème étudié est un cas particulier de la famille de problèmes d'ordonnancement, où l'ensemble des tâches est divisé en trois groupes. On trouve des tâches qui doivent être exécutées sur le premier processeur, celles exécutées sur le second processeur tandis que le troisième groupe contient les tâches qui doivent être exécutées simultanément sur les deux processeurs.

Dans cette étude, nous nous intéressons à l'ordonnancement de tâches sur deux processeurs dédiés. L'objectif du problème est de minimiser le "makespan". Comme décrit dans Graham *et al.* [1], ce problème est noté $P2|f_i x_j, r_j|C_{\max}$. Une telle version du problème peut être rencontrée dans plusieurs applications réelles, comme la production et les transferts de données (cf. Manna et Chu [6]). Une instance du ST2P est représentée par un ensemble N de n tâches à exécuter sur deux processeurs dédiés (noté P_1 et P_2) de sorte qu'une tâche j arrivée à l'instant r_j , doit être exécutée sans préemption pendant sa durée d'exécution p_j et, C_j représente le temps de fin d'exécution de la j -ème tâche, i.e., C_{\max} désigne le makespan.

2 Une méthode de résolution réactive

Dans cette partie, nous résumons la méthode de recherche réactive proposée pour la résolution de l'ordonnancement de tâches sur deux processeurs dédiés. Le principe de base de cette méthode se résume par les étapes suivantes :

1. **Solution de départ** : Une première solution réalisable est obtenue en appliquant un algorithme glouton en s'appuyant sur un des principes de préférences utilisé pour la résolution de problèmes de type knapsack.
2. **Stratégie d'intensification** : Des opérateurs basés sur les 2-opt et 3-opt sont appliqués afin d'intensifier le processus de recherche autour d'une solution courante. Une recherche tabou est aussi appliquée afin d'éviter les cycles, d'accélérer le processus de recherche et d'améliorer, dans certains cas, la qualité des solutions obtenues.
3. **Stratégie de diversification** : Perturber la solution avec une méthode de destruction, puis de reconstruction, sachant que la reconstruction se fait selon le principe du glouton adapté à ce problème (cf., Hifi [2] et Hifi et Michrafy [3]).

Un tel processus (les étapes 2 et 3) est répété jusqu'à ce qu'un nombre maximum d'itérations soit atteint ; ensuite, la solution finale obtenue par la méthode est rendue comme la meilleure solution pour le ST2P.

3 Partie expérimentale

La méthode de résolution proposée dans cette étude (Recherche Réactive -RR) est évaluée sur deux ensembles d'instances, où chaque ensemble est composé de cinq groupes (comme suggéré dans [6]). Afin d'évaluer la performance du RR proposée, nous avons comparé ses résultats à ceux de l'algorithme génétique (noté GA) proposé dans [5] et la borne inférieure proposée dans [6] (comme elle a déjà été utilisée dans [5] pour évaluer la qualité des solutions produites par le GA).

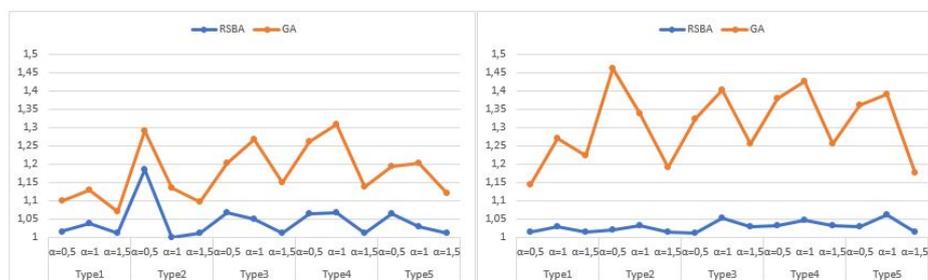


FIG. 1 – Variation des rapports d'approximation expérimentaux des GA et RR : (i) à gauche de la figure pour $n = 10$ et, (ii) à droite de la figure pour $n = 20$.

A partir de la figure 1, on peut observer le bon comportement de la méthode proposée sur la majorité des benchmarks ainsi que sa supériorité en terme de la qualité des solutions obtenues.

4 Conclusion

Dans cette étude, nous nous sommes intéressés à la résolution heuristique du problème d'ordonnement de tâches sur deux processeurs dédiés par l'application d'une recherche réactive. La méthode combine trois caractéristiques principales : une solution de départ construite par adaptation d'une procédure constructive à base d'un algorithme glouton, une méthode d'intensification introduite pour visiter l'ensemble de solutions locales et une stratégie de diversification utilisant l'opérateur de dégradation d'une solution et de reconstruction d'une solution partielle. Enfin, dans une partie expérimentale nous avons illustré la qualité des solutions obtenues par la méthode tout en les comparant aux solutions obtenues par l'un des plus récents algorithmes de la littérature.

Références

- [1] Graham, R.L., Lower, E.L, Lenstra, J.K., Rinnoy, A.H.G. Optimization and Approximation in Deterministic Sequencing and Scheduling Theory. A Survey. Annals of Discrete Mathematics.V5, 287–326, 1979.
- [2] Hifi, M. An iterative rounding search-based algorithm for the disjunctively constrained knapsack problem. Engineering Optimization,46(8), 1109–1122, 2014.
- [3] Hifi, M., Michrafy, M. A reactive local search-based algorithm for the disjunctively constrained knapsack problem. Journal of the Operational Research Society, 57(6), 718–726, 2006.
- [4] Hoogeveen, J.A., van de Velde, S.L. and Veltman, B. Complexity of scheduling multiprocessor tasks with prespecified processor allocations, Discrete Applied Mathematics, Vol. 55, 259–272, 1994.
- [5] Kacem, A., Dammak, A. A genetic algorithm to minimize the makespan on two dedicated processors. Control, Decision and Information Technologies (CoDIT), 400–404, 2014.
- [6] Manaa, A., Chu, C. Scheduling multiprocessor tasks to minimise the makespan on two dedicated processors. European Journal Industrial Engineering, 4(3), 265–279, 2010.