Analyse statistique d'hill climbers à voisinage large pour le problème d'ordonnancement linéaire

Ángel David Reyero Lobo[1], Nicolas Dupin[2]

[1] Université Paris-Saclay, LISN, 91405, Orsay, France
angel.reyero-lobo@universite-paris-saclay.fr
[2] Univ Angers, LERIA, SFR MATHSTIC, F-49000 Angers, France
nicolas.dupin@univ-angers.fr

Mots-clés : *méta-heuristiques, optimisation combinatoire, problème d'ordonnancement linéaire, analyse de paysage, hill climbers, maximums locaux, voisinages.*

1 Introduction

Le problème d'ordonnancement linéaire (LOP, Linear Ordering Problem) consiste à former une permutation d'un ensemble de N items, définissant un classement le plus représentatif issu de préférences entre couples d'items [5]. Le LOP s'applique en théorie du choix social, mais également en bio-informatique, median de permutations pour des requêtes de bases de données [6]. Face aux limites des approches exactes, le LOP a été résolu efficacement par des heuristiques de trajectoire, notamment par recherche taboue et VNS [5]. Les approches les plus efficaces à l'heure actuelle sont des méta-heuristiques de population, telles que [4]. Cet article analyse l'impact des choix de voisinages dans une recherche locale, en notant que les voisinages utilisés sont différents dans [6] et [5]. La méthodologie d'analyse suit [2], en comparant des schémas de recherche locale en partant de solutions initiales identiques et en considérant des voisinages de taille variable dont l'exploration considère le meilleur voisin, par résolution exacte.

2 Voisinages et exploration exacte

L'encodage d'une solution du LOP est un classement, la donnée d'une permutation, un tableau où les éléments sont classés consécutivement. La fitness d'une permutation se calcule ex-nihilo en temps $O(N^2)$, en sommant les $O(N^2)$ préférences considérées dans le classement.

Sur le recuit simulé proposé par [6], le voisinage considéré est d'intervertir deux items classés consécutivement. Mettre à jour la fitness par un tel déplacement se calcule en $O(1)$. A la manière des voisinages k-opt, on généralise ce voisinage en considérant comme voisinage l'ensemble des permutations obtenues d'une permutation initiale en modifiant l'ordre d'au plus k items consécutifs. L'exploration exacte de tels voisinages se ramène à des problèmes LOP de taille k sur les items considérés, que l'on peut calculer efficacement avec un solveur PLNE pour des tailles $N \leqslant 40$, comme analysé dans [1].

Sur les recherches locales considérées dans [5], un voisinage de ré-insertion est considéré, en déplaçant un item sans changer l'ordre partiel des autres items. Pour un item donné, l'exploration exacte d'un tel voisinage est en $O(N)$ avec un calcul amorti, un calcul en $O(N^2)$ permet de définir la meilleure insertion ou de prouver que la permutation est un minimum local du voisinage de ré-insertion. On ne considère pas de voisinage plus larges avec plusieurs ré-insertions simultanées.

3 Algorithmes de descente, "hill-climbers"

Pour les deux familles de voisinages, on définit des schémas de descente "Hill-Climbing" (HC), par exploration optimale de voisinage. Les ré-insertions optimales définissent des itérations en temps $O(N^2)$ pour aboutir à un maximum local.

Pour les voisinages k-opt avec $k > 2$, on considère des optimisations entrelacées par paquets de k items, en considérant les optimisations de 1 à k, de $k+1$ à $2k$, $\ldots$, et ensuite avec un décalage de $k/2$ (de $k/2$ à $3k/2-1$, puis de $3k/2$ à $5k/2-1$, $\ldots$). En considérant une telle séquence, on a un maximum local pour tous les voisinages de taille $3k/2$ au moins, mais toutes les optimisations sont utilisées itérativement pour améliorer la solution courante.

On ne fait pas varier la taille k des voisinages k-opt consécutifs, pour comparer l'impact de la profondeur de ces voisinages sur la qualité des maximums locaux. Les schémas considérés sont d'utiliser de tels hill climbers de taille fixe sur une solution initiale jusqu'à aboutir à un maximum local, de les appliquer sur le maximum local obtenu par ré-insertion sur la solution initiale, et d'itérer itérativement les hill climbers par k-opt consécutifs et par ré-insertion jusqu'à obtenir un maximum local pour tous les voisinages considérés.

4 Méthodologie d'analyse et résultats numériques

L'implémentation a été réalisée sous Julia, de manière à appeler Cplex ou CBC avec JuMP pour les optimisations PLNE. Les valeurs de k sont $k \in \{2, 3, 5, 10, 20\}$, des valeurs supérieures peuvent être envisagées pour les grandes instances. Les instances sélectionnées sont des instances diversifiées de xLOLIB ([5]) et des instances aléatoires et générées pour être représentatives de médians de permutations ([1]). Pour les instances xLOLIB, les meilleures solutions sont issues de [3, 4], les autres instances ont leur valeur optimale connue.

Pour chaque instance, 1000 permutations aléatoires sont générées comme solutions initiales, et on compare les qualités des minimas locaux obtenus par les différents hill-climbers en partant de ces mêmes solutions initiales. Les quartiles, les valeurs extrêmes et moyennes sont comparées sur les fitness des maximums locaux, le nombre moyen d'itération menant au maximum local est également fourni.

Les résultats numériques valident l'intérêt du voisinage de ré-insertion, dont la qualité des maximums locaux surpasse celle des voisinages k-opt consécutifs, particulièrement inefficaces quand k est petit. L'intérêt de grands voisinages k-opt consécutifs pour améliorer les maximums locaux de ré-insertion est également significatif. Les résultats montrent que les instances représentatives de médians de permutations représentatives de l'application bio-informatique [6] (générées par [1]) sont plus faciles pour de la recherche locale avec de meilleurs maximums locaux que les instances aléatoires, les instances xLOLIB sont encore plus difficiles.

Références

[1] N. Dupin. Integer linear programming reformulations for the linear ordering problem. In *International Conference on Optimization and Learning*, pages 74–86. Springer, 2022.

[2] N. Dupin and E. Talbi. Matheuristics to optimize refueling and maintenance planning of nuclear power plants. *Journal of Heuristics*, 27 :63–105, 2021.

[3] E. Garcia, J. Ceberio, and J. Lozano. Hybrid heuristics for the linear ordering problem. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pages 1431–1438. IEEE, 2019.

[4] L. Lugo, C. Segura, and G. Miranda. A diversity-aware memetic algorithm for the linear ordering problem. *Memetic Computing*, 14(4) :395–409, 2022.

[5] R. Martí and G. Reinelt. *The linear ordering problem : exact and heuristic methods in combinatorial optimization*, volume 175. Springer Science & Business Media, 2011.

[6] R. Milosz. Étude algorithmique et combinatoire de la méthode de kemeny-young et du consensus de classements. *Thèse de doctorat, Université de Montréal*, 2018.