

Algorithme mémétique guidé par l'apprentissage profond pour des problèmes de coloration de graphes

Olivier Goudet Cyril Grelier Jin-Kao Hao

LERIA, Université d'Angers, 2 Boulevard Lavoisier, 49045 Angers, France

{cyril.grelier,olivier.goudet,jin-kao.hao}@univ-angers.fr

Mots-clés : *Coloration de graphe, algorithme mémétique, réseau de neurones, métaheuristique*

1 Introduction

Étant donné un graphe $G = (V, E)$, avec V un ensemble de sommets et E un ensemble d'arêtes, un problème de coloration d'un tel graphe implique le partitionnement de l'ensemble des sommets V en différents sous-ensembles V_i en respectant un certain nombre de contraintes et d'objectifs d'optimisation. Le problème de k -coloration (k-COL) en est un des exemples les plus représentatifs. Il consiste à trouver une partition de V en k groupes de couleur V_i de telle sorte que $\forall u, v \in V_i, \{u, v\} \notin E$.

Les problèmes de coloration de graphes ont été étudiés de manière très intensive au cours des dernières décennies et de nombreuses méthodes de coloration ont été proposées dans la littérature. Une des familles de méthodes les plus efficaces, notamment pour le problème (k-COL) évoqué ci-dessus, sont les algorithmes mémétiques [2]. Ces algorithmes utilisent généralement une petite population ne comptant pas plus de 100 individus (voir uniquement 2 comme dans l'algorithme mémétique HEAD [4], état de l'art pour le problème k-COL). À chaque génération, un ou plusieurs nouveaux individus (solutions) sont créés par un opérateur de croisement appliqué à plusieurs individus de la population, avant d'être améliorés par une procédure de recherche locale

Dans un algorithme mémétique, l'opérateur de croisement permet de créer de nouveaux points de départ pour la procédure de recherche locale, qui sont censés être plus prometteurs et meilleurs qu'une initialisation purement aléatoire. Cependant, lorsque l'on utilise une telle procédure de diversification, il n'y a généralement aucun moyen de savoir à l'avance si le nouveau point de départ sera situé dans une zone réellement prometteuse qui mérite d'être examinée par la procédure de recherche locale. En effet, l'utilisation d'un croisement peut parfois ramener le processus de recherche vers une région déjà visitée de l'espace de recherche sans aucune chance d'amélioration ultérieure, ou vers une nouvelle région éloignée de l'optimum global.

2 Algorithme mémétique guidé par l'apprentissage profond

Dans ce travail, nous proposons un nouveau cadre d'algorithme mémétique qui combine un réseau de neurones profond avec les meilleurs outils des métaheuristiques "classiques" pour la coloration de graphes [3]. Afin de réaliser cette combinaison, nous revisitons une idée proposée par Boyan et Moore il y a vingt ans [1]. Dans cet article, les auteurs ont remarqué que la performance d'une procédure de recherche locale dépend de l'état à partir duquel la recherche commence et ont donc proposé d'utiliser un algorithme de régression pour prédire les résultats d'un algorithme de recherche locale en fonction de son point de départ. Une fois appris, ce modèle prédictif peut aider à sélectionner de nouveaux bons points de départ pour la recherche locale et ainsi accélérer le processus de recherche. Dans ce travail, nous revisitons cette

idée en utilisant des techniques modernes d'apprentissage profond pour mieux sélectionner des croisements prometteurs parmi les croisements possibles à chaque génération d'un algorithme mémétique. Une architecture de réseau de neurones de type *deep set* [7], permettant d'exploiter la propriété d'invariance par permutation des couleurs pour les problèmes de coloration, est utilisée.

Par ailleurs, comme l'apprentissage d'un réseau de neurones profond nécessite une grande quantité de données, notre algorithme mémétique est caractérisé par une grande population (environ 20 000 individus), évoluant en parallèle dans l'espace de recherche. Afin d'entraîner le réseau de neurones et calculer toutes les recherches locales en parallèle pour tous les individus de la population, le code est déployé sur une architecture de calcul GPU (*Graphic Processing Units*).

3 Expérimentation

Comme preuve de concept, nous appliquons cette approche pour résoudre le problème de k -coloration (k -COL) et le problème de coloration des sommets pondérés (WVCP) dans un graphe. Le problème k -COL est bien connu et étudié depuis longtemps. Le WVCP a récemment suscité beaucoup d'intérêt dans la littérature [6, 5].

L'approche proposée dans ce travail a été évaluée sur les benchmarks classiques DIMACS et COLOR pour ces problèmes de coloration. Les résultats de ces expériences montrent que l'algorithme est globalement compétitif par rapport aux meilleurs algorithmes de l'état de l'art pour les deux problèmes abordés. Bien que l'algorithme soit assez lent à converger, pour le problème de coloration des sommets, il peut atteindre la plupart des meilleurs résultats connus de la littérature pour les instances difficiles. Pour le problème de coloration pondérée, il peut trouver 14 nouvelles bornes supérieures pour des instances très difficiles, avec des améliorations parfois très significatives.

Une analyse des valeurs de fitness prédites et effectives après la recherche locale montre que le réseau de neurones peut aider, dans une certaine mesure, à trouver de nouveaux points de départ prometteurs à chaque génération, ce qui facilite la découverte de solutions de haute qualité dans l'espace de recherche.

Références

- [1] Justin Boyan and Andrew W Moore. Learning evaluation functions to improve optimization by local search. *Journal of Machine Learning Research*, 1(Nov) :77–112, 2000.
- [2] Philippe Galinier and Jin-Kao Hao. Hybrid evolutionary algorithms for graph coloring. *Journal of Combinatorial Optimization*, 3(4) :379–397, 1999.
- [3] Olivier Goudet, Cyril Grelier, and Jin-Kao Hao. A deep learning guided memetic framework for graph coloring problems. *Knowledge-Based Systems*, 258 :109986, 2022.
- [4] Laurent Moalic and Alexandre Gondran. Variations on memetic algorithms for graph coloring problems. *Journal of Heuristics*, 24(1) :1–24, 2018.
- [5] Bruno Nogueira, Eduardo Tavares, and Paulo Maciel. Iterated local search with tabu search for the weighted vertex coloring problem. *Computers & Operations Research*, 125 :105087, 2021.
- [6] Yiyuan Wang, Shaowei Cai, Shiwei Pan, Ximing Li, and Monghao Yin. Reduction and local search for weighted graph coloring problem. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2433–2441, 2020.
- [7] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.