

Approche hybride pour résoudre le problème du Bin Packing

BACHI Yasmine, MIHOUBI LamiaZohra , MOUSSAOUI Meroua, NOUALI Sarah, SAADI Khaoula

Ecole nationale Supérieure d'Informatique -ESI-Alger

Abstract

L'objectif du problème du bin packing (BPP) est de trouver le nombre minimal de boîtes nécessaire pour ranger un ensemble de n objets ayant des tailles connues, en respectant la capacité de chaque boîte. Ce problème est parmi les problèmes NP-difficile. Dans cet article, on propose un algorithme génétique hybride utilisant le recuit simulé. Les résultats expérimentaux ont montré l'efficacité de notre hybridation dans l'amélioration de la qualité de solution de l'algorithme génétique pour les classes 1 et 2 du benchmark Scholl.

Keywords: Bin packing, hybridation, AG, recuit simulé, WOA, ILWOA

1. Introduction

Le problème du bin packing à une dimension (BPP) est défini comme suit, étant donné un nombre illimité de boîtes avec une capacité fixe C , et un ensemble de n objets, chacun ayant un poids spécifique $0 \leq w_i \leq C$, on cherche à ranger les n objets dans un nombre minimal de boîtes, tout en respectant la capacité C .

1.1. Algorithme génétique (AG)

L'algorithme génétique est une heuristique de recherche initialement proposée par Holland [1] qui imite le processus de sélection naturelle. Elle appartient à la plus grande classe d'algorithmes évolutionnaires (EA), qui génèrent des solutions en utilisant des techniques inspirées de l'évolution naturelle, telles que la mutation, la sélection et le croisement.

La figure suivante représente les étapes de fonctionnement de notre AG, avec la spécification des méthodes utilisées pour implémenter chaque étape.

Les algorithmes génétiques sont connus par leur Rapidité et facilité d'utilisation. En effet, le AG est parmi les métaheuristiques les plus rapides, de plus, si la représentation vectorielle de l'individu est correcte, nous pouvons trouver une solution sans un travail d'analyse approfondi. Par contre, cette métaheuristique ne trouve pas toujours l'optimum et peut se retrouver avec le problème d'optimum local.

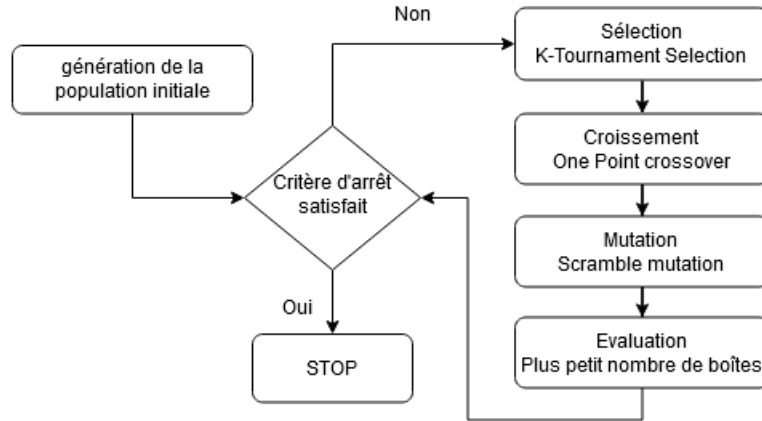


Figure 1: Processus de AG

1.2. Le Recuit Simulé (SA)

Le recuit simulé est un algorithme de recherche locale initialement introduit par Kirkpatrick et al [6] qui simule la fusion et le refroidissement dans le traitement des métaux. Le recuit simulé est généralement implémenté pour rechercher une solution optimale sur une petite zone, même s'il est parfois aussi performant que AG dans certains cas [1]. Cette métaheuristique sophistiquée empêche d'être piégé dans les minima locaux à l'aide d'un moteur de recherche aléatoire exprimé en termes de chaîne de Markov. Elle introduit des changements dans la solution pour améliorer la fonction objectif, mais conserve également des solutions qui, malgré les moins bonnes performances, répondent à certains critères. Mais l'un des inconvénients du recuit simulé est son temps d'exécution élevé.

1.3. Hybridation

Pour surmonter les inconvénients de GA et SA, plusieurs études proposent une hybridation entre les deux. Junghans et Darde [4] comparent entre AG et AG hybride avec SA modifiée (MSA). Le SA utilisé dans leur expérience a été modifié pour contrôler la réduction de la température. Ils ont découvert que le GA-MSA hybride offre une fiabilité plus élevée que le AG. Une autre recherche menée par Chen et Shahandashti dans [2] qui compare également le GA, SA, un hybride de GA-SA et MSA, où ils ont constaté que le GA-SA hybride est plus performante que AG, SA et MSA.

Dans cet article, nous avons hybridé le GA-SA avec quatre scénarios, à savoir le AG-RS hybride, le AGH-RS, le AG-2RS et le AG-MIX. Le schéma hybride AG-RS consiste à obtenir la meilleure solution en AG et à l'utiliser comme population initiale en SA, le AGH-RS utilise le même processus sauf que AG a été initialisé par plusieurs heuristiques. Le schéma AG-2RS consiste à inclure SA dans le AG après l'étape de mutation, et améliorer la solution de AG par le SA à nouveau. finalement le schéma AG-MIX utilise le même processus que

AG-2RS en initialisant le AG par plusieurs heuristiques. D'autres schémas ont été implémenté et seront inclus dans la comparaison, on cite le WOA-RS et le ILWOA-RS qui sont une hybridation de haut niveau des métaheuristiques WOA et ILWOA avec le recuit simulé.

1.4. Organisation du papier

On va tout d'abord définir le problème du bin packing, sa représentation mathématique, par la suite on présentera nos schémas hybrides proposés. Finalement, des résultats expérimentaux sont donnés et une comparaison des schémas hybrides est effectuée dans [num], suivi d'une conclusion.

2. Formulation du problème

2.1. Définition du problème

Le problème du bin packing à une dimension (BPP) est défini comme suit, étant donné un nombre illimité de boîtes avec une capacité fixe C , et un ensemble de n objets, chacun ayant un poids spécifique $0 \leq w_i \leq C$, on cherche à ranger les n objets dans un nombre minimal de boîte, tout en respectant la capacité C .

2.2. Formulation mathématique

Etant donné m boîtes de capacité C et n articles de volume v_i chacun. Soient:

$$x_{ij} = \begin{cases} 1 & \text{article } j \text{ rangé dans la boîte } i \\ 0 & \text{sinon} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{boîte } i \text{ utilisée} \\ 0 & \text{sinon} \end{cases}$$

La formulation du problème donne ainsi le programme linéaire suivant

$$(PN) \begin{cases} Z(\min) = \sum_{i=1}^m y_i \\ \sum_{i=1}^m x_{ij} = 1 \\ \sum_{j=1}^n v_j x_{ij} \leq C y_i \\ y_i \in \{0, 1\} \\ x_{ij} \in \{0, 1\} \end{cases}$$

La première contrainte signifie qu'un article j ne peut être placé qu'en une seule boîte La deuxième fait qu'on ne dépasse pas la taille d'une boîte lors du rangement

3. L'approche hybride proposée

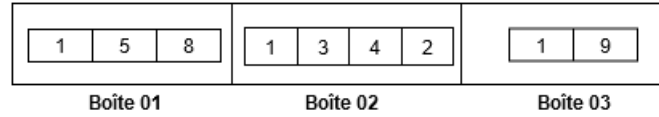
Pour le schéma de l'hybridation proposé dans cet article, il s'agit d'une hybridation classifiée selon la taxonomie de talbi comme suit **HRH globale, généraliste**, et qui utilise l'algorithme génétique comme méthode révolutionnaire, suivie par un Recuit Simulé pour l'intensification des résultats de ce dernier.

3.1. Encodage de la solution

3.1.1. Solution du bin packing

La solution sera représentée par:

- une liste de boîtes, où la boîte est représentée par une liste des objets qui y sont rangés, et l'objet est représenté par un entier désignant son poids. poids des objets qui y sont rangés, ainsi que le nombre de boîtes utilisées, ce qui représente le coût de la solution, et nous permettra d'évaluer la performance de l'algorithme en terme de la qualité de la solution.



- Le nombre de boîtes utilisées, ce qui représente le coût de la solution, et nous permettra d'évaluer la performance de l'algorithme en terme de la qualité de la solution.

3.1.2. Représentation chromosomique

Pour cette implémentation de l'AG, nous allons utiliser une nouvelle représentation chromosomique, proposée dans [5], On représente une solution du Bin packing comme suit:

- On suppose qu'on a 'n' articles à ranger donc on utilisera n boîtes au maximum.
- Chaque boîte est composée de "n" cellules, où chaque cellule ne peut contenir qu'un seul article.
- Chaque cellule à un numéro unique dans la solution.
- Si la cellule de l'ordre i de la boîte j est remplie par un objet, alors on aura plus le droit de ranger un objet dans toutes les cellules de l'ordre i des autres boîtes.
- La cellule "zero" contient le nombre de boîtes utilisés dans cette solution.

Dans le chromosome , on ne garde trace que des numéros de cellules où les objets sont stockés, donc la taille du chromosome sera de $1+n$, où n est le nombre d'objets à ranger.

Exemple: Supposons que nous avons quatre objets ($n=4$) de poids 2, 2, 4 et 4 respectivement, affectés à trois cases ($c_0=3$) comme indiqué ci-dessous :

Le chromosome correspondant est le suivant :

Ceci signifie que le nombre de boîtes utilisé est 3, le premier objet est rangé dans la première cellule (boîte 1) , le deuxième objet est rangé dans la 2ème cellule (boîte 1), le 3ème objet est rangé dans la 7ème cellule (boîte 2), et le 4ème objet est rangé dans la 12ème cellule (boîte 3) .

3.2. L'hybridation AG-RS

Une hybridation entre AG et RS permet à AG d'explorer un énorme espace de recherche et à RS d'exploiter des zones de recherche locales. Le RS commence généralement par une solution initiale aléatoire, dans la solution proposée, la meilleure solution de AG est utilisée comme configuration initiale, et le Recuit simulé va se charger de l'intensification de cette dernière, où il utilise un processus de recherche d'une solution optimale globale dans l'espace de la solution, grâce à son aspect aléatoire, et qui a été prouvé de guider l'algorithme vers l'optimum global. Donc, le GA est puissant pour obtenir une solution presque optimale sur la zone de recherche large tandis que SA est utile pour rechercher une solution dans la zone de recherche étroite.[3]

References

- [1] T. W. Al-Khateeb B. Solving 8-queens problem by using genetic algorithms, simulated annealing, and randomization method. *International Conference on Developments in eSystems Engineering (DeSE)*, page 187–191, 2013. doi: 10.1109/DeSE.2013.41.
- [2] S. S. Chen PH. Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints. page 18:434–443, 2009. doi: 10.1016/j.autcon.2008.10.007.
- [3] Gusti, Andreas, Tirana, M.Shochibul, and F.Wayan. Hybrid genetic algorithm and simulated annealing for function optimization. 2017. doi: 10.25126/jitecs.20161215.
- [4] D. N. Junghans L. Hybrid single objective genetic algorithm coupled with the simulated annealing optimization method for building optimization. page 187–191, 2015. doi: 10.1016/j.enbuild.2014.10.039.

- [5] N. Mohamadi. Application of genetic algorithm for the bin packing problem with a new representation scheme. 2010.
- [6] J. M. S. Kirkpatrick, C. D. Gelatt. Optimization by simulated annealing. 1983.