Optimisation par essaims particulaires pour un problème de placement de sphères

Mhand Hifi, Labib Yousef

EPROAD EA 4669, Université de Picardie Jules Verne
7 rue du Moulin Neuf - 80000 Amiens, France
{hifi,labib.yousef}@u-picardie.fr

Mots-clé : heuristique, optimisation, placement, sphère.

1 Introduction

Nous nous intéressons à la résolution du problème de placement de sphères dans un container. Une instance du problème est représentée par un ensemble N de n items (sphères), où chacun des items est caractérisé par son rayon ($r = 1$), et un container sphérique de rayon illimité ($R = \infty$). Le but du problème est de placer toutes les items (sphériques) disponibles dans le container de sorte à minimiser le rayon R du container. Ce problème possède de nombreuses application pratiques, en particulier dans le domaine de la santé (traitement des cancers), dans le domaine des matériaux intervenant dans les milieux granulaires, et en écologie lorsque certaines caractéristiques de forêts sont à préserver.

Dans ce travail, nous proposons une méthode hybride, combinant un algorithme d'optimisation par essaims particulaires et une procédure de recherche locale s'appuyant sur la résolution d'une séries de programmes non-linéaires.

2 Une méthode hybride pour la résolution du problème

Une représentation simple du problème peut être formulée comme suit :

$$\min \quad R$$

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \geq 2, \qquad \forall (i, j) \in N^2,\ i < j \tag{1}$$

$$\sqrt{x_i^2 + y_i^2 + z_i^2} \leq R - 1, \qquad \forall i \in N, \tag{2}$$

où R dénote le rayon du container à minimiser, l'équation (1) élimine le chevauchement entre deux sphères et la contrainte (2) assure que les sphères se situent à l'intérieur du container.

2.1 Un algorithme par essaims

L'algorithme par essaims (Kennedy *et al.* [4]) s'appuie sur des particules qui représentent des solutions potentielles pour le problème d'optimisation traité. Chacune des solutions est représentée par un vecteur déterminant une position actuelle. Un vecteur $\overrightarrow{x_i} = (x_{i1}, x_{i2}, \cdots, x_{id})$ permet d'évaluer la qualité de la solution courante par application d'une vitesse directionnelle $\overrightarrow{v_i} = (v_{i1}, v_{i2}, \cdots, v_{id})$ de dimension d. Par ailleurs, chaque solution (particule) i garde en mémoire sa meilleure position $\overrightarrow{P}^i_{best}$ auparavant calculée ainsi que la meilleure position $\overrightarrow{G}^k_{best}$ atteinte par l'ensemble des particules voisines.

A un instant t, le déplacement d'une particule i dans l'espace de recherche peut être obtenu par application de ce qui suit :

$$\overrightarrow{v}_i^t = \kappa \left(\overrightarrow{v}_i^{t-1} + c_1 r_1 \left[\overrightarrow{P}_{best_i}^{t-1} - \overrightarrow{x}^{t-1_i} \right] + c_2 r_i \left[\overrightarrow{G}_{best_k}^{t-1} - \overrightarrow{x}_i^{t-1} \right] \right) \tag{3}$$

$$\overrightarrow{x}_i^t = \overrightarrow{x}_i^{t-1} + \overrightarrow{v}_i^{t-1} \tag{4}$$

$$\kappa = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4 \times \varphi}|} \tag{5}$$

où les paramètres $c_1 = c_2 = 2.05$, $\varphi = c_1 + c_2 = 4.1$ et $\kappa = 0.729$ (voir. Kennedy *et al.* [4])

2.2 Une optimisation locale

Construire une solution revient à construire une configuration réalisable ou non réalisable. Une configuration réalisable conduit vers une solution réalisable pour le problème, qui peut être améliorée par une recherche locale. Dans l'autre cas, la configuration peut être aussi corrigée, puis améliorée en applicant une recherche locale. Dans notre cas, nous utilisons cette correction (ou amélioration) par application d'une optimisation continue (dont la procédure a été simplifiée par Wang [2]). Notons que durant le processus de correction (amélioration), la procédure cherche à réduire l'écart des chevauchements (soit entre les sphères ou entre les sphères et le container) et le faire tendre vers zéro (ou selon la précision escomptée).

3 Résultats préliminaires

La méthode proposée (notée Algo) a été testée sur un ensemble d'instances récemment utilisées dans M'Hallah *et al.* [3] (cette méthode sera notée H). La taille de la population utilisée dans ces tests a été fixée à 40 (même si d'autres valeurs ont été explorées pour régler la taille de cette population). Les solutions initiales de la population ont été générées par un algorithme glouton proposée par Hifi and Yousef [1].

n	H	Algo	Gap
45	4.41007453378598	4.40700316053521	*0.0030713733*
46	4.44190731948126	4.44178542926357	*0.0001218902*
47	4.47455137992898	4.47446547698958	*8.59029394E-05*
48	4.49712009262402	4.49632831111380	*0.0007917815*
49	4.52594554850145	4.51928918657509	*0.0066563619*
50	4.55221146893267	4.55095440529171	*0.0012570636*

TAB. 1 – Qualité des bornes de Algo versus les meilleures bornes de la littérature.

La table 1 affiche la moyenne des meilleures bornes réalisées sur 5 différents jets. A partir des résultats de cette table, on peut constater la supériorité de la méthode sur la majorité des instances considérées. D'autres résultats sur d'autres instances encore plus complexes sont en cours de réalisation.

Références

[1] M. Hifi and L. Yousef. A dichotomous search-based heuristic for the three-dimensional sphere packing problem. Cogent Engineering, 2(1) : 1-15, No 994257, 2015.

[2] H. Wang, W. Huang, Q. Zhang and D. Xu. An improved algorithm for the packing of unequal circles within a larger containing circle. European Journal of Operational Research, 141(2) :440–453, 2002.

[3] R. M'Hallah, A. Alkandari and N. Mladenovic. Packing unit spheres into the smallest sphere using vns and nlp. Computers & Operations Research, 40(2) :603–615, 2013.

[4] Maurice Clerc and James Kennedy. The particle swarm explosion, stability, and convergence in a multidimensional complex space. IEEE transactions on Evolutionary Computation, 6(1) :58–73, 2002.