VNS pour le Minimum Sitting Arrangement Problem

Alban Derrien[1], Borja Menéndez[1,2], Marc Sevaux[1], Eduardo G. Pardo[3], Abraham Duarte[2]

[1] Université de Bretagne-Sud, Lab-STICC, UMR 6285 CNRS, Lorient, France
{alban.derrien,marc.sevaux}@univ-ubs.fr

[2] Universidad Rey Juan Carlos, Dpto. Informática y Estadística, Madrid, Spain
{borja.menendez,abraham.duarte}@urjc.es

[3] Universidad Politécnica de Madrid, Dpto. Sistemas Informáticos, Madrid, Spain
eduardo.pardo@upm.es

Mots-clés : *graphe, réseaux sociaux, sitting arrangement, métaheuristiques, VNS*

1 Minimum Sitting Arrangement Problem (MinSAP)

Les graphes signés sont des graphes pour lesquels les arêtes ont un signe positif ou négatif. En psychologie sociale, les graphes signés modélisent des relations sociales. Les arêtes positives présentent alors l'amitié et les arêtes négatives les inimitiés entre les nœuds qui représentent les personnes [1].

Dans un travail récent, Kermarrec et al. ont formalisé le problème suivant : « Est il possible de *dessiner* un graphe signé tel que pour toutes personnes, ses amis sont plus proches que ses ennemis ? »[3]. Lorsque l'on se limite au dessin dans un espace euclidien de taille 1 (sur une ligne), le problème dans le cas général est NP-Complet [2]. Le problème de satisfaction peut alors s'écrire :

Définition (Sitting Arrangement Problem) *Étant donné un graphe signé $G = (V, E^+ \cup E^-)$ avec E^+ l'ensemble des arêtes signées positivement et E^- l'ensemble des arêtes signées négativement. Existe-il une permutation π de V telle que pour chaque $u \in V$: (i) il n'existe pas $(u_1, u) \in E^+, (u_2, u) \in E^-$ tel que $u_1 \underset{\pi}{<} u_2 \underset{\pi}{<} u$; (ii) il n'existe pas $(u_1, u) \in E^+, (u_2, u) \in E^-$ tel que $u_1 \underset{\pi}{>} u_2 \underset{\pi}{>} u$.*

Le problème de minimisation associé (*MinSAP*) minimise le nombre de fois où les conditions (i) et (ii) ne sont pas respectées [5].

Exemple (MinSAP) *Soit le graphe G représenté sur la figure 1 et $\pi = \{A, D, B, C\}$ un ordre sur V représenté par la figure 2 Cette solution a un coût de 1 car $D \underset{\pi}{<} B \underset{\pi}{<} C$: B qui est un ennemis de C est plus proche que D.*

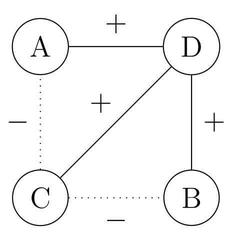

FIG. 1 – Exemple de problème pour le MinSAP, avec 4 noeuds

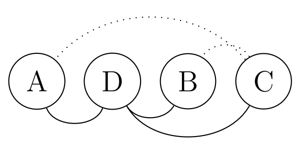

FIG. 2 – Représentation de l'ordre $\pi = \{A, D, B, C\}$

2 Variable Neighborhood Search (VNS)

VNS est une methaheuristique introduite par Mladenović and Hansen, elle exploite l'idée d'un changement de voisinage de façon systématique [4]. Elle est capable de trouver un optima local grâce à une phase de recherche locale ainsi que d'en sortir grâce à une diversification. L'heuristique a grandement évolué, avec de nombreuses extensions que nous appliquons dans notre travail. Par exemple, explorer les solutions par une perturbation aléatoire est un schéma appelé *Reduced VNS (RVNS)* ; combiner de l'aléatoire avec une exploration basique un *Basic VNS (BVNS)* ; explorer le voisinage avec de multiples recherches locales un *Variable Neighborhood Descent (VND)* ; et substituer l'explorations déterministe d'un *BVNS* par un *VND* donne la méthode appelée *General VNS (GVNS)*.

3 Contributions

Dans ce travail, nous avons étudié différentes approches du *VNS* pour résoudre le *MinSAP*. Le travail préliminaire que nous avons effectué montre que cette approche est très prometteuse.

Une solution valide du problème est un ordre représentant la position de chacun des nœuds sur la ligne. Une solution est un vecteur, permutation, de l'ensemble $\{1 \dots n\}$, où chaque valeur représente un nœud ; sa position dans le vecteur donnant sa position sur la ligne. Cette représentation intuitive permet de calculer le coût d'une solution et de mettre simplement en place des voisinages génériques tels que l'échange de deux éléments consécutifs dans la permutation (*Adjacent Pairwise Interchange*) ou de deux éléments quelconques (*Generalized Pairwise Interchange*).

Nous proposons, dans un premier temps, deux procédures constructives pour la solution initiale. La première est aléatoire permettant une grande diversité, la seconde crée la solution en suivant un algorithme glouton. Le critère glouton sélectionne le nœud qui maximise le nombre de connexion avec les nœuds préalablement sélectionnés. Dans le cas où de multiple nœuds peuvent être sélectionnés au vu de ce critère, un noeud est choisi aléatoirement parmi ceux minimisant le nombre de connections positives. Nous utilisons des voisinages génériques : un *Adjacent Pairwise Interchange* et un *Generalized Pairwise Interchange*. La diversité est réalisée en appliquant plusieurs fois ces voisinages aléatoirement.

Nous considérons également des heuristiques dédiées à chaque étape de l'algorithme :

- Modification de l'algorithme constructif initial pour obtenir de meilleures solutions,
- Étude de différents algorithmes de diversification :
  - Suppression des nœuds qui sont à l'origine du plus grand nombre d'erreurs,
  - Inversion de blocks consécutifs de nœuds...
- Optimisation de l'implémentation de la recherche locale,
- Utilisation d'une procédure d'insertion à la meilleure position dans le séquence courante...

Références

[1] D. Cartwright and F. Harary. Structural balance : a generalization of Heider's theory. *Psychological Review*, 1956.

[2] M. Cygan and M. Pilipczuk and M. Pilipczuk and J. Onufry Wojtaszczyk. Sitting closer to friends than enemies, revisited. *CoRR*, abs/1201.1869, 2012.

[3] A-M. Kermarrec and C. Thraves. Signed graph embedding : when everybody can sit closer to friends than enemies. *CoRR*, abs/1405.5023, 2014.

[4] N. Mladenović and P. Hansen. Variable neighborhood search *Computers & Operations Research*, 1997.

[5] E.G. Pardo and M. Soto and C. Thraves. Embedding signed graphs in the line - Heuristics to solve MinSA problem. *Journal of Combinatorial Optimization*, 2015.