Partitionnement multi-contraint d'hypergraphes valués avec sommets pré-fixés

François Galea, Lilia Zaourar
CEA, LIST, F-91191 Gif-sur-Yvette, France
{francois.galea,lilia.zaourar}@cea.fr

Mots-clés : *partitionnement, graphes, heuristiques*

1 Introduction

Le partitionnement d'hypergraphes trouve ses applications, entre autres, dans la parallélisation de calculs scientifiques [1, 3], la répartition de données dans les bases « Big Data » [4] ou la conception de circuits intégrés [2]. Ce problème étant en soi au moins aussi difficile que le partitionnement de graphes, les travaux cherchant à le généraliser pour inclure des contraintes de capacité multiples, la valuation d'hyperarêtes ou des sommets pré-fixés sont relativement peu abondants, et n'intègrent pas tous ces types de contraintes à la fois, car les heuristiques existantes ont du mal à trouver une solution réalisable. Nous proposons une méthode de partitionnement de type k-way multi-niveaux qui inclut toutes ces contraintes, permet d'obtenir une solution de qualité qui satisfait les contraintes. [1]

2 Problème abordé

L'hypergraphe à partitionner est défini par un couple (V, E) dont $V = \{v_1, v_2, ..., v_N\}$ est l'ensemble des sommets, et $E = \{e_1, e_2, ..., e_M\}$ est l'ensemble des hyperarêtes. Chaque hyperarête e_j est pondérée par une pondération w_j. L'hypergraphe est représenté par sa matrice d'incidence H telle que chaque coefficient h_{ij} vaut 1 si le sommet v_i est relié à l'hyperarête e_j, et 0 sinon. Le problème est de placer chaque sommet v_i sur une partition p_k en minimisant un coût qui dépend de la fragmentation sur les différentes partitions de chaque hyperarête. Le partitionnement est soumis à un ensemble de contraintes de capacités de ressources multiples : chaque partition p_k dispose d'une capacité c_{kr} pour la ressource r.

D'une manière simplifiée, le problème peut se formuler de la façon suivante :

$$
\begin{array}{lll}
\min & \sum\limits_{j} \sum\limits_{k} \sum\limits_{k'>k} w_j y_{jk} y_{jk'} & \\
\text{s.c.} & \sum\limits_{k} x_{ik} = 1 & \forall i \\
& h_{ij} x_{ik} \leq y_{jk} & \forall i, j, k \\
& \sum\limits_{i} q_{ir} x_{ik} \leq c_{kr} & \forall k, r \\
& x_{ik} \in \{0, 1\} & \forall i, k \\
& y_{jk} \geq 0 & \forall j, k
\end{array}
$$

où chaque variable x_{ik} vaut 1 si le sommet v_i est placé dans la partition p_k et 0 sinon, et à l'optimum, chaque variable y_{jk} prend la valeur 1 si l'hyperarête e_j a au moins une de ses extrémités dans la partition p_k et 0 sinon.

Il est possible que certaines valeurs x_{ik} soient fixées au préalable et ne puissent pas être modifiées au cours de la résolution.

1. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111, DeepHealth Project.

3 Méthode de résolution

Comme pour la plupart des méthodes de partitionnement de graphes et hypergraphes, la méthode de résolution est une heuristique multi-niveaux constituée de trois phases, comme illustrée dans la Figure 1. La première phase, dite d'aggrégation, consiste à appliquer à plusieurs

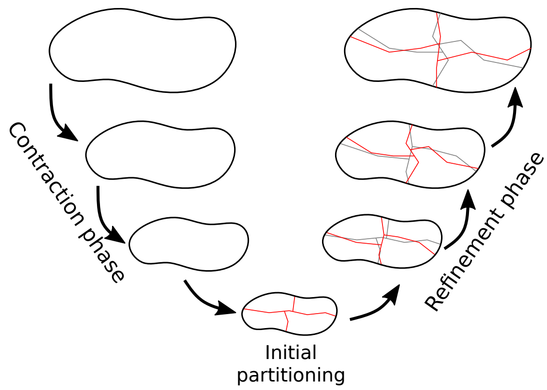

FIG. 1 – Partitionnement multi-niveaux classique

reprises une méthode de clustering pour agréger des sommets de l'hypergraphe afin d'obtenir un hypergraphe plus petit. La phase s'arrête lorsque l'on obtient un hypergraphe de taille suffisamment petite.

La seconde phase est le partitionnement initial, où l'on cherche une première solution réalisable.

Enfin, la troisième phase, dite de raffinement, consiste, en remontant tous les niveaux de clustering, à appliquer à chaque niveau la solution du niveau inférieur, puis à y appliquer une heuristique de raffinement qui essaye d'améliorer la solution courante en déplaçant des sommets entre partitions.

Dans notre étude, nous proposons une méthodologie pour le partitionnement initial qui construit une solution initiale tout au long de la phase d'aggrégation, et qui permet de maintenir en permanence une solution réalisable quel que soit le degré de liberté permis par les contraintes. Cette méthode, combinée à une heuristique de raffinement agressive, permet d'obtenir sur des hypergraphes de plusieurs millions de sommets, des solutions de qualité comparable aux solutions que les solveurs existants trouvent lorsque l'on désactive certaines contraintes.

Enfin, nous proposons des critères pour l'évaluation de solutions qui tiennent compte de tous les types de contraintes.

Références

[1] U. V. Catalyurek and C. Aykanat. Hypergraph-partitioning-based decomposition for parallel sparse-matrix vector multiplication. *IEEE Transactions on Parallel and Distributed Systems*, 10(7) :673–693, July 1999.

[2] Navaratnasothie Selvakkumaran, Abhishek Ranjan, Salil Raje, and George Karypis. Multi-resource Aware Partitioning Algorithms for FPGAs with Heterogeneous Resources. In *Proceedings of the 2004 ACM/SIGDA 12th International Symposium on Field Programmable Gate Arrays*, FPGA '04, pages 253–253, New York, NY, USA, 2004. ACM.

[3] Brendan. Vastenhouw and Rob H. Bisseling. A Two-Dimensional Data Distribution Method for Parallel Sparse Matrix-Vector Multiplication. *SIAM Review*, 47(1) :67–95, 2005.

[4] Wenyin Yang, Guojun Wang, Kim-Kwang Raymond Choo, and Shuhong Chen. HEPart : A balanced hypergraph partitioning algorithm for big data applications. *Future Generation Computer Systems*, 83 :250 – 268, 2018.