Triangle Width : de l'ordonnancement à la théorie des graphes

Luc Libralesso, Vincent Jost, Khadija Hadj Salem, Frédéric Maffray, Florian Fontan

Triangle Width : de l'ordonnancement à la théorie des graphes

Luc Libralesso[1], Vincent Jost[1], Khadija Hadj-Salem[2],
Frédéric Maffray[1], Florian Fontan[1]

[1] Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, 38000 Grenoble, France
[2] Université de Tours, LIFAT EA 6300, CNRS, ROOT ERL CNRS 7002, 64 avenue Jean Portalis, 37200 Tours
luc.libralesso@grenoble-inp.fr

Mots-clés : *Ordonnancement, visualisation de matrices, théorie des graphes*

1 Description du problème

Soit deux ensembles de tâches V et E et deux machines m_1, m_2 (machine "d'entrée", et machine "de sortie"). Chaque tâche $v \in V$ doit être placée sur la machine m_1 et chaque tâche $e \in E$ doit être placée sur la machine m_2. Les tâches E nécessitent que certains tâches de V soient déjà terminés avant de pouvoir être effectués. Nous souhaitons minimiser le temps d'exécution, et donc C_{max}.

Sur la Figure (1a) sont présentées les relations de dépendance entre les tâches de sortie V (sur les lignes, en rouge) et les tâches d'entrée E (sur les colonnes en vert). Par exemple, le tâche de sortie 4, nécessite les tâches d'entrée d, a avant de pouvoir commencer. Une telle solution réalisable est donnée par la Figure (2). Ce problème a été étudié dans le cadre de l'optimisation de cache dans des systèmes de vision embarquée [3].

Plus formellement, le problème peut se décrire par un hypergraphe encodant les relations de dépendance entre les tâches d'entrée et tâches de sortie ainsi que des temps d'exécution pour les différents tâches. Le problème est $\mathcal{NP}$-Difficile et nous nous intéresserons dans cet exposé au cas où l'ensemble des temps d'exécution est identique. Même avec cette contrainte, le problème reste $\mathcal{NP}$-Difficile.

(a) matrice d'incidence des dépendances (b) ordres ligne/colonne différents

FIG. 1 – Matrices d'incidence utilisant différents ordres

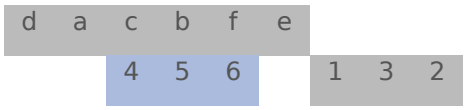

FIG. 2 – Exemple d'ordonnancement des tâches E sur la machine m_i et V sur la machine m_o

2 Permutation ligne/colonne sur une matrice

En visualisant nos instances avec cette forme matricielle, nous avons remarqué que les bonnes solutions partagent une certaine propriété (Figure (1b)). En effet, en choisissant l'ordre des lignes et colonnes de la matrice comme ordre sur les lignes/colonnes, nous observons un "triangle blanc" dans le coin supérieur droit de la matrice. En étudiant davantage ce phénomène, il est clair que minimiser le temps d'exécution revient à maximiser le triangle blanc de la matrice.

Nous pouvons donc re-poser le problème de la manière suivante : Trouver un ordre sur les lignes/colonnes qui maximise le triangle blanc dans le coin supérieur droit. Cette nouvelle formulation donne le nom de ce problème : "Triangle Width".

Nous proposons une application web où il est possible pour le lecteur de tester et d'essayer de résoudre quelques instances du problème [1] : `http://librallu.gitlab.io/hypergraph-viz/`.

3 Vers la théorie des graphes

Il est possible de formuler ce problème d'une troisième manière. Considérons un hypergraphe. Nous avons une réserve de carburant initiale b et souhaitons visiter l'ensemble des sommets et arêtes du graphe. Chaque visite de sommet coûte 1 unité de carburant. Une fois que l'on visite le dernier sommet adjacent à une arête, cela nous rapporte une unité de carburant. Quelle est la quantité minimale b pour visiter le graphe ?

Dans ce problème, il est possible de montrer certains résultats de complexité :

— Quand H est un graphe (chaque hyper-arête a cardinalité 2) et que les durées des tâches d'entrée sont plus petites que les durées des tâches de sortie, le problème peut être résolu en temps polynomial.

— Quand H n'est pas un graphe (il existe des hyper-arêtes de cardinalité différente de 2), le problème est $\mathcal{NP}$-Complet, même avec des durées unitaires.

— Déterminer s'il est possible de trouver une permutation avec un triangle de taille $|E|-1$ est $\mathcal{NP}$-Complet. En effet, il existe un résultat ([2]) qui indique que trouver une permutation de lignes/colonnes d'une matrice 0,1 pour la rendre triangulaire est $\mathcal{NP}$-Complet.

Dans [1] est présenté un problème proche de *Triangle Width*. Ce problème est défini comme une visite de graphe. Existe t-il une visite de graphe de telle sorte que nous ayons une réserve de carburant de k^2 et que chaque sommet coûte k à ajouter et que chaque arête ajoutée rapporte 1 unité de carburant. Pour ce problème, la complexité n'est pas encore connue.

Références

[1] Claudio Arbib, Michele Flammini, and Enrico Nardelli. How to survive while visiting a graph. *Discrete applied mathematics*, 99(1-3) :279–293, 2000.

[2] Guillaume Fertin, Irena Rusu, and Stéphane Vialette. Obtaining a triangular matrix by independent row-column permutations. In *International Symposium on Algorithms and Computation*, pages 165–175. Springer, 2015.

[3] Khadija Hadj Salem. *Optimisation du fonctionnement d'un générateur de hiérarchies mémoires pour les systèmes de vision embarquée*. PhD thesis, Grenoble Alpes, 2018.

1. L'application fonctionne sur PC, tablette et téléphone.