Approximation du problème de flot multiterminal entier maximum dans les graphes orientés

Cédric Bentz

CEDRIC-CNAM, 292, rue Saint-Martin, 75003 Paris
cedric.bentz@cnam.fr

Etant donné un graphe simple pondéré par des entiers et deux sommets particuliers s et t, le problème bien connu du flot maximum consiste à router le nombre maximum d'unités de flot entre s et t. De nombreux algorithmes polynomiaux ont été élaborés pour résoudre ce problème [1], et le théorème de Ford et Fulkerson [7] établit que la valeur d'un flot maximum entier est égale à la valeur d'une coupe minimum, c'est-à-dire à la valeur d'un ensemble d'arêtes de poids minimum dont la suppression sépare s de t. Malheureusement, ces propriétés ne sont pas conservées lorsque l'on étudie des généralisations de ce problème [4]. L'une d'entre elles est le *problème du flot multiterminal entier maximum* (PFMTEM) : étant donné un graphe simple pondéré par des entiers et un ensemble $T = \{t_1, \ldots, t_k\}$ de sommets *terminaux*, il consiste à router le nombre maximum d'unités de flot entre les terminaux. Ce problème se pose, par exemple, lorsque l'on veut router un nombre maximum de requêtes (unitaires) entre un ensemble prédéfini de nœuds d'un réseau.

Dans cette communication, nous nous intéressons à la difficulté et à l'approximation polynomiale (sous le ratio classique) de ce problème dans les graphes orientés. En particulier, nous identifions un nouveau paramètre intéressant, k_S, le nombre de terminaux solitaires (un terminal est dit *solitaire* s'il se trouve sur au moins un circuit ne contenant pas d'autre terminal).

Dans notre démarche d'approximation, nous utilisons la relation de dualité qui existe entre les relaxations continues des programmes linéaires modélisant PFMTEM et le *problème de la coupe multiterminale minimum* (PCMTM), qui consiste à sélectionner un ensemble d'arêtes de poids total minimum dont la suppression sépare t_i de t_j pour $i \neq j$. PCMTM généralise le problème de la coupe minimum, et a été amplement étudié dans le cas non orienté ([3], [4], [5], [12]). Le cas orienté a également reçu quelque attention : Garg et al. [9] montrent qu'il est $\mathcal{NP}$-difficile même pour $k = 2$ et donnent un algorithme polynomial $O(\log k)$-approché, et Naor et Zosin [10] donnent un algorithme polynomial 2-approché. Cependant, l'algorithme de Garg et al. a une propriété particulièrement intéressante : il calcule une coupe multiterminale dont la valeur est au plus $O(\log k)$ fois la valeur d'un flot multiterminal entier, et fournit donc une $O(\log k)$-approximation à la fois pour PCMTM et pour PFMTEM (alors que l'algorithme de Naor et Zosin ne fournit pas de solution approchée pour PFMTEM). Par ailleurs, Costa et al. [4] montrent que PFMTEM et PCMTM sont polynomiaux dans les graphes orientés acycliques en utilisant une simple réduction à un problème de flot maximum et de coupe minimum, respectivement. A notre connaissance, ce sont les seuls résultats concernant PFMTEM dans les graphes orientés. Dans les graphes non orientés, PFMTEM est polynomial dans les graphes semi-eulériens [8] et dans les arbres [2]. Cependant, le cas général orienté est "plus difficile" que le cas non orienté, puisque ce dernier s'y réduit polynomialement (voir [11, (70.9) à la page 1224]).

Nous montrons d'abord, en adaptant la preuve de [6, Théorème 3], que PFMTEM est $\mathcal{NP}$-difficile dans les graphes orientés, même si $k_S = 2$ et que toutes les capacités valent 1. Cela implique que, dans les graphes orientés, PFMTEM est fortement $\mathcal{NP}$-difficile, et n'admet pas de schéma d'approximation fortement polynomial si $\mathcal{P} \neq \mathcal{NP}$ (même pour $k_S = 2$). Ensuite, nous montrons que PFMTEM est polynomial lorsque $k_S = 0$, en adaptant une approche utilisée dans [4]. Enfin, nous utilisons ce dernier résultat pour améliorer l'algorithme approché proposé en [9], et obtenons un algorithme polynomial $O(\log k_S)$-approché pour $k_S \geq 2$. Nous montrons également, à l'aide d'une famille d'exemples, que l'analyse du ratio d'approximation de cet algorithme est fine, et nous fournissons un algorithme polynomial 2-approché très simple pour le cas (ouvert) $k_S = 1$. Par ailleurs, certains de nos résultats concernant PFMTEM s'appliquent aussi à PCMTM ou sont comparables à des résultats déjà connus pour PCMTM.

En guise de conclusion, nous proposons quelques questions ouvertes qui découlent naturellement des résultats obtenus dans cette étude : quelle est la complexité du cas $k_S = 1$? Existe-t-il un algorithme polynomial $O(1)$-approché pour PFMTEM dans les graphes orientés ?

Références

1. A.K. Ahuja, T.L. Magnanti et J.B. Orlin. Network Flows – Theory, Algorithms, and Applications. Prentice Hall, Englewood Cliffs, New Jersey (1993).
2. A. Billionnet et M.-C. Costa. Multiway cut and integer flow problems in trees. Proceedings CTW04, ENDM 17 (2004) 105–109.
3. D.Z. Chen et X. Wu. Efficient algorithms for k-terminal cuts on planar graphs. Algorithmica 38 (2004) 299–316.
4. M.-C. Costa, L. Létocart et F. Roupin. Minimal multicut and maximal integer multiflow : a survey. Eur. J. Op. Res. 162 (2005) 55–69.
5. E. Dahlhaus, D.S. Johnson, C.H. Papadimitriou, P.D. Seymour et M. Yannakakis. The complexity of multiterminal cuts. SIAM J. Computing 23 (1994) 864–894.
6. S. Even, A. Itai et A. Shamir. On the complexity of timetable and multicommodity flow problems. SIAM J. Computing 5 (1976) 691–703.
7. L.R. Ford et D.R. Fulkerson. Maximal Flow Through a Network. Canadian Journal of Mathematics 8 (1956) 339–404.
8. A. Frank, A. Karzanov et A. Sebö. On integer multiflow maximization, SIAM J. Discrete Mathematics 10 (1997) 158–170.
9. N. Garg, V.V. Vazirani et M. Yannakakis. Multiway cuts in directed and node weighted graphs. Proceedings ICALP'94 (1994) 487–498.
10. J. Naor et L. Zosin. A 2-approximation algorithm for the directed multiway cut problem. Proceedings FOCS'97 (1997) 548–553.
11. A. Schrijver. Combinatorial Optimization - Polyhedra and Efficiency. Algorithms and Combinatorics 24 (2003). Springer.
12. W.-C. Yeh. A Simple Algorithm for the Planar Multiway Cut Problem. Journal of Algorithms 39 (2001) 68–77.