Research Article **RJMCS 2017 1:8**

Research Journal of Mathematics and Computer Science
(ISSN:2576-3989)

An approximation algorithm for minimizing congestion in the single-source k-splittable flow

Chengwen Jiao*, Qi Feng, Weichun Bu

College of Science, Zhongyuan University of Technology, Zhengzhou, Henan 450007, People's Republic of China

ABSTRACT

In the traditional multi-commodity transmission networks, the number of paths each commodity can use is unrestricted, and the commodities can use arbitrary number of paths to transmit the flow. However, in the real transmission networks, too many paths will increase the total transmission cost of the network and also cause difficulties in the management of the network. In 2002, Baier[1] proposed the k-splittable flow problem, in which each commodity can only use a limited number of paths to transmit the flow. In this paper, we study the k-splittable multi-commodity transmission flow problem with the objective of minimizing congestion and cost. We propose an approximation algorithm with performance ratio $(\frac{3}{2}\cdot\frac{1}{1-(1/2)^{k_{min}}}, 1)$ for congestion and cost in the single-source case, in which $k_{\min}$ is the minimum value of the number of paths each commodity can use. The congestion reflects the total load of the network to some extent. The main aim of minimizing congestion is to distribute the demands of the commodities on the network in a balanced way, avoiding the case that some edge is used too much. By this way, the performance of the network as a whole can be guaranteed and more commodities can be served.

Keywords: k-splittable flow, congestion minimization, approximation algorithm

***Correspondence to Author:**
Chengwen Jiao
College of Science, Zhongyuan University of Technology Zhengzhou, Henan 450007, People's Republic of China

How to cite this article:
Chengwen Jiao et al., An approximation algorithm for minimizing congestion in the single-source k-splittable flow. Research Journal of Mathematics and Computer Science, 2017; 1:8.

eSciencePublisher

eSciPub LLC, Houston, TX USA.
Website: http://escipub.com/

Introduction

In the traditional multi-commodity flow problems, a directed graph $G=(V,E)$ is given with vertex set V and edge set E in which $|V|=n$ and $|E|=m$. For each edge $e \in E$, denote $u_e > 0$ and $c_e > 0$ be its arc capacity and unit cost, respectively. A set of commodities denoted by L needs to be transmitted in the network. Each commodity $l \in L$ has a certain amount of demand d_l to be transmitted from source node s_l to destination node t_l. The number of paths each commodity can use is not restricted, while in practice, large number of paths for each commodity may reduce the central management of the network. Baier[1] proposed the k-splittable flow problem. The only difference from the traditional multi-commodity flow problem is that the number of paths each commodity can use is restricted, that is, each commodity $l \in L$ can only use limited number of paths, say k^l to transmit its flow. If k^l =1, $\forall l \in L$, this problem is in fact the unsplittable flow problem (UFP) proposed by Kleinberg[2]. If $k^l \geq |E|, \forall l \in L$, it is reduced to the traditional maximum flow problem. For the multi-commodity flow problem, each commodity using only one path will make the network too loaded while too many paths will increase the difficulty in the management of the network. In the k-splittable flow problem, a mediate case is considered, that is $1<k^l<|E|, \forall l \in L$.

For the unsplittable flow problem, Kleinberg[2] introduced several optimization versions. In the "minimization congestion" version, the task is to find the smallest value λ such that there exists an unsplittable flow that uses at most a λ-fraction of the capacity of any edge. In the "maximum concurrent flow" problem, the aim is to maximize the routable fraction of the demand, i.e., find the maximal factor by which the given demands can be multiplied such that there still exists a feasible unsplittable flow satisfying the resulting demands. It can be proven that any solution to the minimum congestion problem of value λ can be turned into a solution to the maximum concurrent flow problem of value $1/\lambda$, and vice versa. The "minimize number of rounds" version asks for a partition of the set of commodities into a minimum number of subsets(rounds) and a feasible unsplittable flow for each subset. At last, the "maximum routable demands" problem is to find a feasible unsplittable flow for a subset of demands maximizing the sum of demands in the subset.

For the unsplittable flow problem, Erlebach.et.al. [3] proved that for arbitrary ε, obtaining an approximation algorithm of minimizing congestion and cost with performance ratio better than (2-ε,1) is NP-hard. The unsplittable flow problem is much easier if all commodities share a common single source. However, the resulting single-source unsplittable flow problem still remains strongly NP-hard. Some approximation algorithms with constant performance ratio of congestion have been developed for the single-source UFP. See references [4]-[6].

As for the k-splittable flow problem, researchers generalize the above optimization versions and there are a lot of study on the related problems. Baier.et.al. [7] solved the maximum Single- and Multi-commodity k-splittable flow problem using approximation algorithms. The authors proved that the maximum single-commodity k-splittable flow problem is NP-hard in the strong sense for directed graphs. Koch.et.al.[8] studied the single commodity maximum k-splittable flow problem. It is proved that when k is a constant, this problem is strongly NP-hard and obtaining an approximation algorithm with performance ratio better than $k/(k+1)$ is NP-hard. While when k is not a constant, obtaining an approximation algorithm with performance ratio better than 5/6 is NP-hard. Koch.et.al.[9] considered the same problem by giving an algorithm with two stages, first a packing step and second a routing step.

Kolliopoulos[10] researched the single-source 2-splittable minimization congestion problem, each commodity can only use at most two paths. Using the rounding down strategy, turn each commodity into two small commodities and by this way the 2-splittable flow problem is transformed into an unsplittable flow problem. Using the related approximation algorithm for the UFP, the authors obtained an unsplittable flow that satisfies the demands of the small commodities. Then by scaling a suitable constant, a 2-splittable flow satisfying the original demands with performance ratio of congestion and cost (2,1) is obtained. Salazar.et.al.[11] study the single-source k-splittable flow problem in which the number of paths each commodity can use is all equal to k. Authors designed an approximation algorithm for this problem by using rounding up strategy and obtained the performance ratio (1+1/k+1/(2k-1), 1) for congestion and cost.

In this paper, we consider the single-source k-splittable flow problem. We generalize the rounding down strategy used in [10] to the k-splittable flow and give an approximation algorithm with performance ratio $(\frac{3}{2}\cdot\frac{1}{1-(1/2)^{k_{\min}}},1)$ for congestion and cost. When $k_{\min}$ increases, the congestion ratio $\frac{3}{2}\cdot\frac{1}{1-(1/2)^{k_{\min}}}$ decreases, which is reasonable in the practice.

Problem description and the approximation algorithm

In this paper, we consider the single-source k-splittable flow problem. Denote the common single source node by s and suppose that the number of paths each commodity can use is at least 2. The balance condition holds in the network, that is, $d_{\max} \le u_{\min}$, where $d_{\max} := \max\{d_l : l \in L\}$, $u_{\min} := \min\{u_e : e \in E\}$. For analysis simplicity, we scale a suitable number, such as $1/u_{\min}$, to all the demand values and all the edge capacity values such that $d_{\max} \le 1 \le u_{\min}$ We also suppose that a feasible fractional flow satisfying all the demands of the commodities exists in the network, denote it by f_0. We propose an approximation algorithm, denote it by A. The main steps of algorithm A are as follows.

Algorithm A

Step1: For each commodity $l \in L$, transmit it into several small commodities.

If $d_l = 1$, let $q_1^l = -1,\ q_2^l = -1$, then $d_l = 2^{q_1^l} + 2^{q_2^l}$;

If $d_l < 1$, let $q_1^l = \lfloor \log_2 d_l \rfloor, q_2^l = \left\lfloor \log_2 d_l - 2^{q_1^l} \right\rfloor, \cdots,$

$q_j^l = \left\lfloor \log_2 d_l - 2^{q_1^l} - \cdots - 2^{q_{j-1}^l} \right\rfloor, \cdots$,

If there is an integer $j < k^l$ such that $d_l = 2^{q_1^l} + \cdots + 2^{q_j^l}$, the above iteration stops after j iterations and j negative integers $q_1^l,\ q_2^l, \cdots,\ q_j^l$ are obtained. Otherwise, the above iteration stops after proceeding k^l iterations, obtaining k^l negative integers $q_1^l,\ q_2^l, \cdots,\ q_{k^l}^l$.

The above iterations transform each commodity l into not larger than k^l small commodities with the same source node s and sink node t_l. The demands of the small commodities are all negative powers of 2 and we denote the total demands of the small commodities by $\underline{d}_l$.

Step2: For each commodity $l \in L$, find paths from f_0 for commodity l that having the maximum unit cost iteratively. Once a path is found, deleting its whole or partial flow from the path, until the remaining flow from s to t_l is $\underline{d}_l$. By this way, we obtain a feasible fractional flow that satisfies all the small commodities, denote it by f_1.

Step3: Beginning with f_1, adopt the related algorithm for the unsplittable flow problem and obtain an unsplittable flow that satisfies the demands of all the small commodities.

Step4: For each $l \in L$, multiply the constant $d_l / \underline{d}_l$ to the flow value of each path used by the small commodities corresponding to l such that the new total flow of these paths is equal to the original demand value d_l of commodity l.

For algorithm A, we have some remarks as follows.

- The feasible fractional flow f_0 can be obtained by using the classical maximum flow-minimum cost algorithm.

- In Step2, the fractional feasible flow f_1 can be obtained by proceeding finite number of shortest path algorithm. We can do it as follows: for each edge $e \in E$, define its weight by $w_e := W - c_e$ in which W is a constant such that $W > c_{\max}, c_{\max} := \max\{c_e, e \in E\}$. Then a shortest weight path is corresponding to a maximum unit cost path.

- In Step3, we can use the algorithm proposed in [5] to obtain an unsplittable flow satisfying the small commodities.

Algorithm analysis

First we estimate the ratio of d_l and $\underline{d}_l$, denote it by α_l, $\alpha_l := d_l / \underline{d}_l$. If $d_l = \underline{d}_l$, $\alpha_l = 1$. Otherwise, by Step1, we know that the commodity l must be transformed into k^l small commodities with the demand values $2^{q_1^l},\ 2^{q_2^l},\ \cdots,\ 2^{q_{k^l}^l}$

,respectively,and $\underline{d}_l = 2^{q_1^l} + 2^{q_2^l} + \cdots + 2^{q_{k^l}^l}$. By

the construction of $q_1^l,\ q_2^l, \cdots,\ q_{k^l}^l$, we have that

$q_1^l = \lfloor \log_2 d_l \rfloor > \log_2 d_l - 1$, further we have

$$2^{q_1^l} > \frac{1}{2} d_l \quad (3.1)$$

Since $q_2^l = \lfloor \log_2(d_l - 2^{q_1^l}) \rfloor > \log_2(d_l - 2^{q_1^l}) - 1$, we have

$$2 \cdot 2^{q_2^l} > d_l - 2^{q_1^l} \quad (3.2)$$

(3.1)+(3.2) we have that

$$2 \cdot (2^{q_1^l} + 2^{q_2^l}) > (\frac{1}{2} + 1) d_l \quad (3.3)$$

Now suppose that for $j > 2$, we have

$$2^{j-1} \cdot (2^{q_1^l} + \cdots + 2^{q_j^l}) > (\frac{1}{2} + 1 + \cdots + 2^{j-2}) d_l \quad (3.4)$$

Since $q_{j+1}^l = \lfloor \log_2(d_l - 2^{q_1^l} - \cdots - 2^{q_j^l}) \rfloor > \log_2(d_l - 2^{q_1^l} - \cdots - 2^{q_j^l}) - 1$

, we have

$$2^j \cdot 2^{q_{j+1}^l} > 2^{j-1} \cdot d_l - 2^{j-1}(2^{q_1^l} + \cdots + 2^{q_j^l}) \quad (3.5)$$

(3.4)+(3.5) we can obtain that

$$2^j \cdot (2^{q_1^l} + \cdots + 2^{q_{j+1}^l}) > (\frac{1}{2} + 1 + \cdots + 2^{j-1}) d_l \quad (3.6)$$

When proceeding the k^l-th iteration, we have

$$2^{k^l-1} \cdot (2^{q_1^l} + \cdots + 2^{q_{k^l}^l}) > (\frac{1}{2} + 1 + \cdots + 2^{k^l-2}) d_l \quad (3.7)$$

That is $2^{q_1^l} + \cdots + 2^{q_{k^l}^l} > \frac{1}{2^{k^l-1}}(\frac{1}{2} + 1 + \cdots + 2^{k^l-2}) d_l$,

by further computing we can have that

$$\underline{d_l} = 2^{q_1^l} + \cdots + 2^{q_{k^l}^l} > (1 - (\frac{1}{2})^{k^l}) d_l \quad (3.8)$$

Thus we have

$$\alpha_l = \frac{d_l}{\underline{d_l}} < \frac{1}{1 - (\frac{1}{2})^{k^l}} \quad (3.9)$$

By the above analysis, for each commodity $l \in L$, we obtain the upper bound on the ratio α_l between the original demand value d_l of commodity l and the total demand value of the small commodities corresponding to l. In order to get the feasible flow f_1 from f_0 that satisfies the small commodities, the algorithm find the paths with maximum unit cost iteratively, deleting the all or partial flow from the paths, until the remaining flow from s to t_l is exactly d_l. It is easy to see that for each $e \in E, f_1(e) \le f_0(\overline{e})$.

Reference [10] adopted the Theorem on the unsplittable flow problem in [5], that is Theorem 1 as follows.

Theorem 1[5]: Given an UFP instance where all demands are powers of 1/2 and an initial fractional flow solution, there is an algorithm, called POWER-ALG, which finds an unsplittable flow that violates the capacity of any edge by at most $d_{\max} - d_{\min}$ and whose cost is bounded by the cost of the initial fractional flow.

For Algorithm A we have the following important theorem:

Theorem 2: For any instance of a single-source k-splittable flow problem, suppose that $d_{\max} \le 1 \le u_{\min}$ and there is a fractional flow f_0 that satisfies all the demands of the commodities in the set L, Algorithm A can obtain a k-splittable flow f from f_0 that satisfies all the demands of the commodities such that the flow value $f(e)$ of each edge e satisfies $f(e) < u_e \cdot \frac{3}{2} \cdot \frac{1}{1-(1/2)^{k_{\min}}}$ in which $k_{\min} = \min\{k^l : l \in L\}$, and the cost $c(f)$ of f is not larger than the cost $c(f_0)$ of f_0.

Proof: By algorithm A we know that each original commodity is transformed into some small commodities. Suppose that $d_{\lfloor \max \rfloor}$ is the largest demand value of all the small commodities, it is easy to see that $d_{\lfloor \max \rfloor} \le \frac{1}{2}$. Since f_1 is the flow that satisfies all the small commodities, using the algorithm POWER-ALG, we can obtain an unsplittable flow that satisfies all the small commodities, denote the UFP by f_2. By theorem 1 we know that $c(f_2) \le c(f_1)$ and for each $e \in E$ we have $f_2(e) \le f_1(e) + d_{\lfloor \max \rfloor}$. In f_2 each small com modity use only one path to transmit the flow, denote R^l be the transmitting paths of the small commodities corresponding to commodity l. We have that

$$\underline{d_l} = \sum_{p \in R^l} f_2(p) \quad (3.10)$$

In which $f_2(p)$ is the flow value of path $p \in R^l$ in

f_2. Define $R_e^l := \{p \in R^l : e \in p\}$ by the paths of R^l that using the edge e. For $e \in E$ we have that $f_2(e) = \sum_{l \in L} \sum_{p \in R_e^l} f_2(p)$. In order to obtain a k-splittable flow that satisfies the demand of the original commodities, for each $l \in L$, multiply the path flow values of the paths in R^l by α_l, and we obtain a new flow, denote it by f. For each edge $e \in E$, we have that

$$
\begin{aligned}
f(e) &= \sum_{l \in L} \sum_{p \in R_e^l} \alpha_l \cdot f_2(p) = \sum_{l \in L} \alpha_l \sum_{p \in R_e^l} f_2(p) \\
&\le \max\{\alpha_l : l \in L\} \cdot \sum_{l \in L} \sum_{p \in R_e^l} f_2(p) \\
&= \max\{\alpha_l : l \in L\} \cdot f_2(e) \le \max\{\alpha_l : l \in L\} \cdot (f_1(e) + \frac{1}{2}) \\
&\le \max\{\alpha_l : l \in L\} \cdot (f_0(e) + \frac{1}{2}) \le \max\{\alpha_l : l \in L\} \cdot (u_e + \frac{1}{2} u_e) \\
&= \frac{3}{2} \cdot \max\{\alpha_l : l \in L\} \cdot u_e
\end{aligned}
$$

Since

$$\max\{\alpha_l : l \in L\} < \max\{\frac{1}{1-(1/2)^{k^l}} : l \in L\} = \frac{1}{1-(1/2)^{k_{\min}}},$$

we prove the first part of the Theorem.

Next we prove the second part of the Theorem. Define $c_p := \sum_{e \in p} c_e$ be the unit cost of path p. In Step2, in order to obtain the flow f_1 from f_0 that satisfies the small commodities, the algorithm finds the maximum unit cost paths from f_0 iteratively, decreases the whole or partial flow from f_0. Denote these paths with decreased flow by Q^l, we have that

$$\sum_{p \in Q^l} f_0(p) = d_l - \underline{d_l} \qquad (3.11)$$

In which $f_0(p)$ denotes the decreased flow value of p from f_0. Further we have

$$\sum_{l \in L} \sum_{p \in Q^l} f_0(p) \cdot c_p = c(f_0) - c(f_1) \qquad (3.12)$$

Let

$$c_{\max}(R^l) = \max\{c_p : p \in R^l\},\ c_{\min}(Q^l) = \min\{c_p : p \in Q^l\}$$

, by the algorithm we know that

$$c_{\max}(R^l) \le c_{\min}(Q^l) \qquad (3.13)$$

By the construction of f we have that

$$
\begin{aligned}
c(f) &= \sum_{l \in L} \sum_{p \in R^l} \alpha_l \cdot f_2(p) \cdot c_p = \sum_{l \in L} \sum_{p \in R^l} (f_2(p) \cdot c_p + (\alpha_l - 1) f_2(p) \cdot c_p) \\
&= c(f_2) + \sum_{l \in L} \sum_{p \in R^l} (\alpha_l - 1) f_2(p) \cdot c_p \le c(f_2) + \sum_{l \in L} c_{\max}(R^l) \sum_{p \in R^l} (\alpha_l - 1) f_2(p) \\
&= c(f_2) + \sum_{l \in L} c_{\max}(R^l)(\alpha_l - 1) \cdot \underline{d_l} = c(f_2) + \sum_{l \in L} c_{\max}(R^l)(d_l - \underline{d_l}) \\
&\le c(f_2) + \sum_{l \in L} c_{\min}(Q^l)(d_l - \underline{d_l}) = c(f_2) + \sum_{l \in L} c_{\min}(Q^l) \sum_{p \in Q^l} f_0(p) \\
&\le c(f_2) + \sum_{l \in L} \sum_{p \in Q^l} c_p \cdot f_0(p) = c(f_2) + c(f_0) - c(f_1) \le c(f_0)
\end{aligned}
$$

By theorem 2 we know that when $d_{\max} \le 1 \le u_{\min}$ (In fact this condition can be generalized to $d_{\max} \le u_{\min}$, by multiplying an approximate number, it can turn into $d_{\max} \le 1 \le u_{\min}$), algorithm A can find a k-splittable flow that satisfies the parh restrictions. We obtain the performance ratio of congestion and cost $(\frac{3}{2} \cdot \frac{1}{1-(1/2)^{k_{\min}}}, 1)$.

If for each $l \in L$, $k^l \le 2$, the congestion value is 2, which is the same as in [10]. When $k_{\min}$ increases, that is the minimum number of paths each commodity can use increases, the congestion value $\frac{3}{2} \cdot \frac{1}{1-(1/2)^{k_{\min}}}$ will decrease, which is reasonable in the practice.

Conclusions

In this paper, we consider the single-source multicommodity k-splittable minimizing congestion problem. In this problem, the number of paths each commodity can use may not be equal. We generalize the rounding down strategy used in [10] to this problem and obtain the congestion and cost performance ratio $(\frac{3}{2} \cdot \frac{1}{1-(1/2)^{k_{\min}}}, 1)$. Authors in [11] also consider the single-source k-splittable flow problem while the number of paths each commodity can use is equal to k. They obtained the congestion and cost performance ratio (1+1/k+1/(2k-1), 1). Although the congestion ratio is better than ours in some cases, the algorithm in [11] needs to adopt the algorithm in [12] which increases the difficulty of the algorithm.

For the k-splittable flow problem, most the current algorithms rely on the algorithms of the unsplittable flow and there are few results on the generalized multi-source case. In the future, we will go on to study the k-splittable flow problem. Analysis the characters of the k-splittable flow and design more effective algorithms.

Acknowledgement

This research is supported in part by The National Natural Science Foundation(Nos.11701595). This work is also supported by Young Backbone Teachers of Henan Colleges.2015 GGJS-193.

References

[1] G.Baier. Flows with path restrictions. Ph.D. Thesis, Technisce Universitat Berlin, Cuvillier Verlag Gottingen, 2003.

[2] J.Kleinberg. Single-source unsplittable flow. In Proceedings of the 37th Annual Symposium on Foundations of Computer Science, 1996, pp.68-77.

[3] T.Erlebach, A.Hall. NP-hardness of broadcast scheduling and inapproximability of single-source unsplittable min-cost flow. In: Proc.13th ACM-SIAM Symp, On discrete Algorithms, 2002.

[4] Y.Dinitiz, N.Garg, M.X.Goemans. On the single source unsplittable flow problem. Combinatorica 19, 1-25, 1999.

[5] S.G.Kolliopoulos and C.Stein. Approximation algorithms for single-source unsplittable flow. SIAM Journal on Computing, 31(3): 919-946, 2001.

[6] M.Skutella. Approximating the single-source unsplittable min-cost flow problem. Mathematical Programming, 91, 493-514, 2002.

[7] G.Baier, E.Kohler, and M.Skutella. On the k-splittable flow problem. Algorithmica, 42: 231-248, 2005.

[8] R.Koch., M.Skutella, and I.Spenke. Approximation and complexity of k-splittable flows. In: Approximation and Online Algorithms, Third International Workshop, WAOA, 244-257, 2005.

[9] R.Koch., M.Skutella, and I.Spenke. Maximum k-splittable s,t-flows. Theory of Computing Systems, 43(1): 56-66, 2006.

[10] S.G.Kolliopoulos. Minimum-cost single-source 2-splittable flow. Information Processing Letters, 94(1): 15-18, 2005.

[11] F.Salazar and M.Skutella. Single-source k-splittable min-cost flows. Operations research letters, 37:71-74,2006.

[12] M.Martens, F.Salazar, and M.Skutella.Representing single source multi-commodity flows as convex combinations of unsplittable flows. L.Arge, M.Hoffmann, E.Welzl(Eds), Algorithms-ESA 2007, Lecture Notes in Computer Sciences, Vol.4698, Springer, Berlin, 395-406, 2007.