Mechanism design for route allocation in multiple-commodity network*

(Extended Abstract)

Qipeng Liu
Institute of interdisciplinary information sciences
Tsinghua University, China
lqp1831@gmail.com

Yicheng Liu
Institute of interdisciplinary information sciences
Tsinghua University, China
61c@live.cn

Pingzhong Tang
Institute of interdisciplinary information sciences
Tsinghua University, China
kenshin@tsinghua.edu.cn

ABSTRACT

We consider the problem of allocating routes in multiple-commodity networks. In such networks, a user can directly download the ‘desired file from a server, or she can do this via a indirectly route to the server if others on the route choose to share their bandwidths. A key feature of such networks is that a user may benefit from exchanging her bandwidth with others to improve the download efficiency. However, she may be strategic about the amount of bandwidth he chooses to share with and may withhold her true bandwidth if it is optimal to do so. Our goal, described in terms of mechanism design, is to design a route allocation mechanism achieves maxmin and/or Pareto efficiency, subject to the participation as well as incentive compatible constraints. We make the following contributions. We first consider the setting where each user can only use a single route to download her file. We show that designing a feasible mechanism in this setting is NP-Complete. To circumvent this complexity, we then consider the setting where each user can use a collection of routes. We show that, optimal mechanisms that satisfy the participation constraints can be efficiently implemented via linear programming.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Economics, Theory, Algorithms

Keywords

mechanism design, strategy proof, minmax delay

*This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61361136003, 61303077, and a Tsinghua University Initiative Scientific Research Grant.

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*
Copyright © 2014, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

1. INTRODUCTION

Much of the work in multi-agent system research has focused on resource allocation (cf. eg. [3, Chapter 11]). In a resource allocation problem, the center wants to assign her resources to multiple agents so that certain desirable objective can be met. Quite often, the objective takes parameters that are private information of the agents, who might misreport this information if it is optimal for them to do so. As a result, resource allocation has been naturally casted as a problem of mechanism design and has been extensively studied for monetary scenarios such as auctions and non-monetary scenarios such as cake cutting.

In this paper, we consider a route allocation problem in multiple-commodity networks. For the network structures under consideration, there are several vertices known as *users*. Each user demands a file of certain size stored on a server[1]. The users themselves form a directed graph. Each user has a capacity, which denotes the maximum (traffic) flow that can go through that vertex. We assume this vertex capacity is private information of the user. Between each user and each server, there is an arc constrained by certain capacity, denoting the maximal flow that can go through the arc. Given such a network, a user can download her target file via any route to the destination server. Given the reported vertex capacities, a route allocation mechanism then allocates a route (or multiple routes, both of which we consider) and feasible flow within the route for each user. A user's utility is the negation of its time delay, calculated by

$$-\frac{\text{file size}}{\text{allocated flow}}.$$

The center's objective is the commonly used *minmax* criteria (see [2]), i.e., to minimize the maximum delay among all users. We also consider the implementation of Pareto efficient mechanisms.

Our problem encompasses many compelling applications in computer networks. One popular instance is the use of proxy for file downloading. Consider the following scenario:

Example 1. *User A wants to download a file from server 1 but has a slow direct connection. Similarly between user B and server 2. However, the cross connections between user A and server 2 as well as user B and server 1 are relatively fast. User A can thus benefit from downloading the file via user B while herself serves as proxy for user B to download from server 1.*

[1]All of our results can be generalized to the P2P network.

Clearly, after reconfiguration, both A and B are better off.

2. OUR CONTRIBUTIONS

Single Route We first investigate the "single-route" version of the problem: designing a mechanism that outputs for each user a *single* route from her vertex to destination server (denoted such route as $s-t$ route). We show that the problem of finding a feasible solution is NP-COMPLETE by reduction from the Partition problem [1]. This suggests that it is unlikely even to implement a feasible solution in polynomial time.

THEOREM 1. *The single-route problem is* NP-COMPLETE.

Multiple Route To circumvent the hardness result, we then investigate the "multiple-route" version of the problem: designing a mechanism that outputs for each user a *collection* of $s-t$ routes. We show that the problem is not easier than the "factional multi-commodity flow" problem. To our best knowledge, the later is not known to have a better solution than linear programming.

THEOREM 2. *The multiple-route problem is not easier than the* FACTIONAL MULTI-COMMODITY FLOW *problem.*

We then design an optimal mechanism for this problem via a linear programming. One difficulty with this approach is the multiplicity of optimal solutions. Among all these solutions, we identify a class that can be obtained by optimizing the flow for each user one by one, according to any ordering among users. This can be done in polynomial time by solving a series of linear programs. Based on the linear program, we propose several mechanisms that optimize different objectives. For all the mechanisms we propose, we show that they enjoy the property of *strategy-proofness*: no one can benefit from under-reporting her capacity, no matter what others do.

One technical difficulty of the proof is to compare two outputs when there is a big gap between a user's underreported bandwidth and her true bandwidth. In this case, the flow assignment could be completely different from the original one and existing network flow techniques do not apply. We identify a lexicographically largest solution and show a series of nice local properties of this solution.

In particular, we have the following results.

Technical details for multiple route

THEOREM 3.

- *There is a mechanism that is individually rational, incentive compatible and Pareto efficient.*
- *There is a mechanism that is individually rational, incentive compatible and Maxmin.*

We prove this by constructing the Mechanisms 1 and 2.

3. CONCLUSION

In this abstract, we consider a setting of mechanism design without money. The setting involves a file-sharing network structure where each node has a private capacity. Our setting encompasses a variety of real-world applications ranging from proxy routine protocol design to express carrier local routing. For two objectives, maxmin and Pareto efficiency, we propose a strategy-proof and individually rational mechanism respectively. We are currently working on extend these two mechanisms to more general settings.

Mechanism 1 An IR, PE and SP mechanism

Notations: G denotes the graph of the network; b_{ij} denotes the bandwidth between the i-th agent and j-th server; d_i denotes the number of i-th agent's sever; c_i denotes the total download size of i-th agent (used later); v_i denotes the capacity of i-th agent's bandwidth; x_i denotes the bandwidth i-th agent obtains; $f_{i,e}$ denotes the quantity of flow on edge e whose destination is agent i.

Given $G, \{b_{i,j}\}, \{(d_i, c_i)\}, \{v_i\}$, repeat the following procedure $n = |V|$ times:

maximize : x_i (the i-th time)

subject to

1. $x_i = x_{i,1} + x_{i,2} + \cdots + x_{i,n}$ for all $i = 1, \cdots, n$;
2. $x_i = x_{i,i} + \sum_{e'} f_{i,e'}$ (e' going into i) for all $i = 1, \cdots, n$;
3. $\sum_{e''} f_{i,e''} = \sum_{e'} f_{i,e'} + x_{i,j}$ (e' going into j and e'' going out of j) for all $i \neq j$;
4. $x_i \geq b_{i,d_i}$ for all i;
5. $\sum_j x_{j,i} + \sum_{e',j} f_{j,e'} \leq v_i$ (e' going into i) ;
6. $\sum_k x_{k,i} \leq b_{i,j}$ (where k have the following property: $d_k = j$), for all i, j;
7. for all $j \leq i-1$, set $x_i = x_i^*$ where x_i^* is the previous value we have got ;
8. all variables are non-negative;

Mechanism 2 Strategy-proof Minmax Mechanism

1. Binary search the minimized maximal downloading time D^* among the solution of the LP (listed below)
2. For all $p_i \in P$, p_i denotes the i-th agent.
   (a) if $\frac{c_i}{b_{id_i}} < D$, set $x_i' \leftarrow b_{id_i}$
   (b) else set $x_i' \leftarrow \frac{c_i}{D}$
3. For all $p_i \in P$, add constraint $x_i = x_i'$
4. Solve the new LP.

The LP is almost the same LP mentioned in Mechanism 1, while with one more constraint $\mathbf{x_i \geq c_i/D}$ **for all i** and requires no objective function. D denotes a given maximal downloading time.

4. REFERENCES

[1] B. Hayes. The easiest hard problem. *American Scientist*, 90(2):113–117, 2002.

[2] N. Nisan and A. Ronen. Algorithmic mechanism design (extended abstract). In *Proceedings of STOC*, pages 129–140, New York, NY, USA, 1999. ACM.

[3] Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game theoretic and Logical Fundations.* Cambridge Uni. Press, 2009.