Theoretical Computer Science 607 (2015) 221–246

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Randomized oblivious integral routing for minimizing power cost ☆

Yangguang Shi [a,b], Fa Zhang [c], Jie Wu [d], Zhiyong Liu [a,e,*]

[a] Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences, No. 6 Kexueyuan South Road, Haidian District, Beijing, China
[b] University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing, China
[c] Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, No. 6 Kexueyuan South Road, Haidian District, Beijing, China
[d] Department of Computer and Information Sciences, College of Science and Technology, Temple University, Suite 400, Carnell Hall, 1803 N. Broad Street, Philadelphia, PA, USA
[e] State Key Lab for Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, No. 6 Kexueyuan South Road, Haidian District, Beijing, China

ARTICLE INFO

Article history:
Received 13 November 2014
Received in revised form 6 June 2015
Accepted 4 July 2015
Available online 9 July 2015

Keywords:
Oblivious routing
Integral routing
Randomized algorithm
Competitive ratio
Energy efficiency

ABSTRACT

Given an undirected network $G(V, E)$ and a set of traffic requests $\mathcal{R}$, the minimum power-cost routing problem requires that each $R_k \in \mathcal{R}$ be routed along a single path to minimize $\sum_{e\in E}(l_e)^\alpha$, where l_e is the traffic load on edge e and α is a constant greater than 1. Typically, $\alpha \in (1, 3]$. This problem is important in optimizing the energy consumption of networks.

To address this problem, we propose a randomized oblivious routing algorithm. An oblivious routing algorithm makes decisions independently of the current traffic in the network. This feature enables the efficient implementation of our algorithm in a distributed manner, which is desirable for large-scale high-capacity networks.

An important feature of our work is that our algorithm can satisfy the integral constraint, which requires that each traffic request R_k should follow a single path. We prove that, given this constraint, no randomized oblivious routing algorithm can guarantee a competitive ratio bounded by $o(|E|^{\frac{\alpha-1}{\alpha+1}})$. By contrast, our approach provides a competitive ratio of $O(|E|^{\frac{\alpha-1}{\alpha+1}} \log^{\frac{2\alpha}{\alpha+1}} |V| \cdot \log^{\alpha-1} D)$, where D is the maximum demand of traffic requests. Furthermore, our results also hold for a more general case where the objective is to minimize $\sum_e (l_e)^p$, where $p \geq 1$ is an arbitrary unknown parameter with a given upper bound $\alpha > 1$.

The theoretical results established in proving these bounds can be further generalized to a framework of designing and analyzing oblivious integral routing algorithms, which is significant for research on minimizing $\sum_e (l_e)^\alpha$ in specific scenarios with simplified problem settings. For instance, we prove that this framework can generate an oblivious

☆ Some preliminary results of our work have appeared in a conference version [35] of this paper under the title "Oblivious integral routing for minimizing the quadratic polynomial cost".

* Corresponding author at: Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences, No. 6 Kexueyuan South Road, Haidian District, Beijing, China. Tel.: +86 010 62600634.

E-mail addresses: shiyangguang@ict.ac.cn (Y. Shi), zhangfa@ict.ac.cn (F. Zhang), jiewu@temple.edu (J. Wu), zyliu@ict.ac.cn (Z. Liu).

http://dx.doi.org/10.1016/j.tcs.2015.07.007
0304-3975/© 2015 Elsevier B.V. All rights reserved.

integral routing algorithm whose competitive ratio can be bounded by $O(\log^{\alpha}|V| \cdot \log^{\alpha-1} D)$ and $O(\log^{3\alpha}|V| \cdot \log^{\alpha-1} D)$ on expanders and hypercubes, respectively.
© 2015 Elsevier B.V. All rights reserved.

1. Introduction

In a minimum power-cost routing (MPR) problem, we are given a network $G(V, E)$ and a set of traffic requests $\mathcal{R} = \{R_1, R_2, \cdots, R_k, \cdots\}$. V and E represent the node set and edge set of G, respectively. Here we consider a typical case where G is undirected [6], i.e., each edge $e \in E$ is bidirectional. Each traffic request $R_k \in \mathcal{R}$ specifies its source-target pair $\{s_k, t_k\} \in V \times V$ and the demand (i.e., the volume of flow that needs to be routed) $d_k \geq 1$. Routing traffic requests along any edge $e \in E$ will incur a cost that grows superadditively with the load. Formally, let l_e be the flow routed along e, the corresponding cost will be a **power** function $f(l_e) = (l_e)^{\alpha}$, where α is a constant greater than 1 and is typically in the interval $(1, 3]$. The objective is to route every $R_k \in \mathcal{R}$ along a single path to minimize the overall cost $\sum_e f(l_e)$. In the following, we will also use an equivalent form of the overall cost, $\|\vec{l}\|_{\alpha}^{\alpha}$, where $\vec{l}$ represents the load vector composed of every l_e, and the operator $\|\cdot\|_{\alpha}^{\alpha}$ represents the α-th power of the α-*norm*.

The MPR problem is attracting great attention because of the emergence of energy conservation issues in data networks [6,8,21,28]. Research conducted by the U.S. Department of Energy [1] indicates that over 50 billion kWh of energy is annually consumed by data networks, whereas at least 40% of this can be saved if the electric power consumption[1] of network elements is in proportion to the actual traffic. For this reason, the speed scaling technique has become ubiquitous because it allows network devices to dynamically adjust their electric power consumption according to traffic. The electric power consumption of a network device with the capability of speed scaling can be characterized by the function $P(x) = x^q$ with $q > 1$, where x is the working speed and q is a constant, the value of which depends on the hardware. The value of q is usually assumed to be around 3 [12,24], while new studies indicate that it can be smaller. For instance, it will respectively take the values 1.11, 1.62, and 1.66 for Intel PXA 270, Pentium M770, and a TCP offload engine [41]. This implies that results of the MPR problem will help optimize the electric power consumption of the entire network.

In this paper, we investigate *oblivious routing* strategies [15,27,31,20,17,22,26] for the MPR problem. For an oblivious routing algorithm, each of its routing decisions is made independently of network traffic. This means that the routing paths for each $R_k \in \mathcal{R}$ are determined only using knowledge of the topology of the network G, the source-target pair $\{s_k, t_k\}$, and some random bits (if needed), in the absence of any information on the set $\mathcal{R} - R_k$, the value of d_k, or the load vector $\vec{l}$. An oblivious routing algorithm can be viewed as precomputing a routing "template" before any traffic request is known. In particular, for a *deterministic* oblivious routing strategy, the corresponding template specifies a unit flow $\mathtt{H}(u, v)$ for each node pair $\{u, v\}$ in G [15,17,31]. Then, each R_k will be routed according to the flow $d_k \cdot \mathtt{H}(s_k, t_k)$. By contrast, for a *randomized* oblivious routing strategy, the precomputed template contains a probabilistic distribution over a collection of unit flows $\{\mathtt{H}_1(u, v), \cdots, \mathtt{H}_i(u, v), \cdots\}$ for each $\{u, v\}$ [31]. In such case, each R_k will be routed according to the flow $d_k \cdot \mathtt{H}_i(s_k, t_k)$ with the corresponding probability $p_i(s_k, t_k)$, which implies that traffic requests with the same source-target pair can go through different paths.

An oblivious routing algorithm is attractive because of its simplicity of implementation. Since it allows for the routing strategy to be precomputed and stored in the routing table of every node, the oblivious routing algorithm can be efficiently implemented in a distributed manner [32]. It is especially significant for high-capacity network routers, where traffic requests will dynamically arrive on a transient timescale in the order of nanoseconds [42,37]. In such a circumstance, path selection based on real-time assessment of the traffic pattern is time-consuming, which implies that a routing algorithm depending on the current traffic may be inefficient. By contrast, oblivious routing algorithms can make timely routing decisions by simply generating random bits and looking up the routing tables, which will be a desirable feature when dealing with the issue of energy efficiency in large-scale high-capacity networks.

To the best of our knowledge, only a few oblivious routing algorithms have been designed to minimize $\|\vec{l}\|_{\alpha}^{\alpha}$, including [11,15,27]. These works, however, only consider the splittable version of MPR, where traffic requests can be partitioned into fractional flows. In this paper, we focus on the **unsplittable** version, which requires that each $R_k \in \mathcal{R}$ should follow a single path. Throughout this paper, we will refer to this requirement as the *integral constraint*. This constraint is important for many practical environments [3], especially for data networks where the frames are not arbitrarily divisible.

When the integral constraint exists, any deterministic oblivious routing algorithm will have to specify a fixed path for each source-target pair [17,20]. We prove that because of the superadditivity of the cost function, such a routing algorithm cannot provide a competitive ratio of $o(|E|^{\alpha-1})$, which implies a lower bound of $\Omega(|E|)$ on the competitive ratio for the typical case $\alpha \geq 2$. *Competitive ratio* here refers to the largest gap between the cost incurred by the oblivious routing algorithm and the cost associated with the optimal solution [27,32]. Such a lower bound indicates that randomization is required by oblivious routing strategies to guarantee a satisfactory performance.

[1] To avoid any ambiguity, throughout this paper we use *electric power consumption* to refer to the electrical power consumed by actual devices, whereas when we talk about the "power-cost", we mean a cost function in the form of $(l_e)^{\alpha}$.

Download English Version:

https://daneshyari.com/en/article/435666

Download Persian Version:

https://daneshyari.com/article/435666

Daneshyari.com