Fully Distributed Algorithms for Minimum Delay Routing Under Heavy Traffic

Sang-Woon Jeon, *Member, IEEE,* Kyomin Jung, *Member, IEEE,* and Hyunseok Chang, *Member, IEEE*

Abstract—We study a minimum delay routing problem in the context of distributed networks with and without partial load information. Even though a general minimum delay routing problem is NP hard, assuming uniformly distributed K source–destination (SD) pairs at random, we provide a lower bound on the average delay and demonstrate by simulation that it is tight for a certain classes of regularly deployed networks. We also show that some routing in a distributed manner is enough to achieve asymptotically optimal load balancing with high probability as K tends to infinity. In order to set such routing, however, each SD pair should know global load information, which is unrealistic for most networks. We propose novel predetermined path routing algorithms in which each SD pair chooses its routing path only among a set of predetermined paths. We then propose an efficient way of distributed construction for predetermined paths that are able to distribute traffic over a network. Our predetermined path routing algorithms work in a fully distributed manner with very limited load information or without any load information. In various network models, we demonstrate by simulation that the delay of the predetermined path routing algorithms quickly converges to that of the distributed routing with global load information.

Index Terms—Delay minimization, distributed routing, load balancing, low complexity routing algorithm, multi-source network, partial network information

1 Introduction

As demands for real-time applications increase, the best-effort paradigm of the past internet model has revealed limitations for providing an integrated service of voice, data, and video [1]–[4]. Many current applications not only require high data rates, but they also demand various quality-of-service requirements. In particular, recently developed voice over IP and real-time video streaming services require strict delay constraints [5], [6], hard to be guaranteed based on the naive best-effort paradigm.

The minimum delay routing problem has been studied over the past decades in the literature including both wired and wireless ad hoc networks [7]–[11]. In spite of surging importance of delivering real-time data, the minimum delay routing protocol is still unknown even for a simple class of networks [12]–[14] and, as a consequence, there is no theoretical framework to establish minimum delay routing applicable for general networks. For instance, the minimum delay routing problem for simple line networks has been shown to be NP hard [12], meaning that it is computationally untraceable as the network size increases. Due to such difficulties, many researchers have been focused on the development of efficient low-complexity routing algorithms under various network environments.

In order to reduce the overall network delay, the amount of load delivered by each communication link should be balanced to each other. Multi-path routing has been actively studied in this context since it has an innate advantage over single-path routing by distributing traffic into several routing paths; see [4], [9], [10], [12], [14]–[18] and the references therein. In [10], however, it has been pointed out that multi-path routing through several shortest paths cannot significantly relieve the load unbalance between the network center and the network boundary unless the number of used shortest paths is very large [9]. To resolve such a load unbalance, geometric routing algorithms detouring the network center has been proposed in [11], [19]. As pointed out in [10], the effect of multi-path routing can be enlarged by routing through disjoint paths and the on-demand construction of maximally disjoint routing paths has been studied in [15].

In practice, a central coordinator, which optimizes routing paths based on global network information, is unrealistic for most current networks due to the system complexity and feedback overhead. In the absence of a central coordinator, the works in [20]–[24] have studied decentralized routing in the context of game theory. When multiple users are able to access subsets of servers and wish to minimize their own delay, a Nash equilibrium of this game cannot generally achieve the global optimum [24]. Only the case in which the number of users is large enough and the server speeds are relatively bounded, it was shown that a Nash equilibrium approaches the minimum delay under linear delay functions. Without these assumptions, the ratio between a Nash equilibrium and the global optimum is

- *S.-W. Jeon is with the Department of Information and Communication Engineering, Andong National University, Andong 760-749, South Korea. E-mail: swjeon@anu.ac.kr.*
- *K. Jung is with the Department of Electrical and Computer Engineering, Seoul National University, Seoul 151-744, South Korea. E-mail: kjung@snu.ac.kr.*
- *H. Chang is with the Department of Electrical Engineering, KAIST, Daejeon 606-081, South Korea. E-mail: hyunseok.chang@kaist.ac.kr.*

Manuscript received 11 Oct. 2012; revised 26 Sep. 2013; accepted 19 Oct. 2013. Date of publication 3 Nov. 2013; date of current version 15 May 2014. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier 10.1109/TMC.2013.144

upper bounded by 5/2. For a general network with multiple source–destination (SD) pairs, this ratio is at most 4/3 under linear link delay functions and can be arbitrarily large under non-linear link delay functions [20]. To achieve such Nash equilibria, each user or SD pair should know the strategies of all the other users, which implies that each user should know at least the locations of all the other users. Therefore, it is hard to apply a game theoretic routing solution in a distributed manner if each user should set up its routing only with a partial view of network information.

The development of distributed routing algorithms that are able to effectively reduce the overall network delay have received great attention due to its practical importance such as vehicular networks, military networks, and internet [25]–[32]. One of the main challenges here is how to establish distributed routing paths under the existence of network ambiguity. The distributed construction of routing paths with and without limited node location information have been studied in [28], [30], [31] and the on-demand routing based on the distance vector has been studied in [26], [33]. Furthermore, establishing multiple disjoint routing paths is crucially important to enlarge the effect of multi-path routing. Motivated by this aspect, the on-demand distance vector routing in [26] has been extended to provide multiple disjoint routing paths in [33], [34] and traffic-splitting and load-aware on-demand multi-path routing has been studied in [15], [16].

In this paper, we study a minimum delay distributed routing problem in which each SD pair is only able to acquire a partial view of current load information or is not able to acquire any load information and set up its routing in a distributed manner. We mainly focus on the heavy traffic regime, where the delay is a primary performance measure, and propose efficient distributed routing algorithms that provide an order-optimal load balancing as the number of SD pairs increases. More specifically, the main contributions of this paper are as follows.

- We set a general *minimum delay routing problem* including all possible (multi-path) routing strategies for any network topology. Even though the problem itself is in general NP hard, we show a lower bound on the average delay. We demonstrate by simulation that this lower bound becomes tight for some regularly deployed networks.
- We set a *minimum delay distributed routing problem* for any network topology in which each SD pair sets its (multi-path) routing paths sequentially in a distributed manner. We prove that the routing solution of this problem turns out to provide an order-optimal load balancing in the limit of large number of SD pairs. We also prove that the single-path distributed routing only minimizing its own delay is enough to achieve this order-optimal load balancing, which can be easily implemented by Dijkstra's algorithm.
- We propose a novel *distributed method of multi-path construction* which we use predetermined paths in our routing algorithms. The proposed method works for any network topology and constructs multiple routing paths that are disjoint of each other and detour the network center. Hence, routing through predetermined paths can effectively relieve the load unbalance between communication links.
- We propose two *distributed predetermined path routing algorithms*, which works for any network topology. The first algorithm uses partial load information to set the minimum delay predetermined path and the second algorithm chooses one of the predetermined paths uniformly at random without any load information. We demonstrate by simulation that, with a small number of predetermined paths, the delays of our routing algorithms quickly converge to that of the minimum delay distributed routing which requires global load information.

This paper is organized as follows. In Section 2, we explain the underlying network model, the minimum delay routing problem, and basic design principles for distributed routing. In Section 3, we derive a lower bound on the delay achievable by solving the minimum delay routing problem. In Section 4, we set the minimum delay distributed routing problem and show its asymptotic optimality for load balancing. In Section 5, we propose several efficient distributed routing algorithms according to the amount of available load information. We simulate performance of the proposed routing algorithms on various network models in Section 6 and conclude this paper in Section 7.

2 System Model

In this section, we first define our network model and the delay measure used in this paper, and then explain basic design principles for distributed routing. Throughout the paper, for a given set $\mathcal{A}$, we will use $|\mathcal{A}|$ to denote the cardinality of $\mathcal{A}$ and I_E to denote the indicator function for an event E, which is one if E occurs or zero otherwise. We also use $\mathbb{R}_+$ and $\mathbb{Z}_+$ to denote the set of positive real numbers and the set of positive integers, respectively.

2.1 Setup

We consider a network of a connected directed graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ in which for each node pair (i,j) there is a directed path from node i to node j, where $i,j \in \mathcal{V}$ and $i \neq j$. The vertex set $\mathcal{V}$ represents the set of N nodes and the edge set $\mathcal{E}$ represents the set of communication links between the nodes. We assume that there exist K SD pairs in the network. The k-th source $s_k \in \mathcal{V}$ wishes to communicate at a rate of $r_k \in \mathbb{R}_+$ with the k-th destination $d_k \in \mathcal{V}$, $s_k \neq d_k$, where $k \in \{1,\cdots,K\}$. Here, each node can be a source or a destination of multiple SD pairs. Fig. 1 depicts an example network.

Now consider routing of the SD pairs over $\mathcal{G}$. Let $M_{i,j}$ denote the total number of distinct paths from node $i \in \mathcal{V}$ to node $j \in \mathcal{V}$, $i \neq j$, and $\mathcal{P}_{i,j}(m) \subseteq \mathcal{E}$ denote the m-th distinct path from node i to node j, where $m \in \{1,\cdots,M_{i,j}\}$.[1] For notational simplicity, let $M_k = M_{s_k,d_k}$ and $\mathcal{P}_k(m) = \mathcal{P}_{s_k,d_k}(m)$. That is, M_k and $\mathcal{P}_k(m)$ are the total number of distinct paths and the m-th distinct path of the k-th SD pair, respectively. Define $r_k(m) \in \{0\} \cup \mathbb{R}_+$ as the rate of the k-th SD pair routed through $\mathcal{P}_k(m)$, where $m \in \{1,\cdots,M_k\}$. Then the routing of the k-th SD pair can be represented as $\mathcal{R}_k = \{r_k(m)\}_{m=1}^{M_k}$ and

1. Throughout the paper, we only consider simple paths.

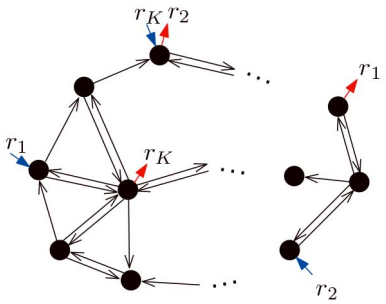


Fig. 1. Connected direct graph with K SD pairs.

the overall routing can be represented as $\mathcal{R} = \{\mathcal{R}_k\}_{k=1}^{K}$. In this paper, we only consider *feasible* routing that satisfies $\sum_{m=1}^{M_k} r_k(m) = r_k$ for all $k \in \{1, \cdots, K\}$.

Let us define the network delay (or cost), which will be a function of $\mathcal{R}$. Define the load of link $e \in \mathcal{E}$ as the aggregate rate routed through e. That is,

$$l_e = \sum_{k=1}^{K} \sum_{m=1}^{M_k} r_k(m) I_{e \in \mathcal{P}_k(m)}. \tag{1}$$

We assume a load-dependent link delay function $f_e{:}l_e \rightarrow \{0\} \cup \mathbb{R}_+$ that satisfies the following properties.

1) $f_e(x)$ is a continuous function of x.
2) $f_e(x)$ is a strictly increasing function of x.
3) $f_e(x)$ is a convex function of x.

Then the delay of the k-th SD pair is defined as

$$f_k(\mathcal{R}) = \sum_{m=1}^{M_k} r_k(m) \sum_{e \in \mathcal{P}_k(m)} f_e(l_e), \tag{2}$$

and the average delay over all SD pairs is defined as

$$f(\mathcal{R}) = \frac{1}{K} \sum_{k=1}^{K} f_k(\mathcal{R}). \tag{3}$$

By substituting (2) in (3), we have

$$\begin{aligned} f(\mathcal{R}) &= \frac{1}{K} \sum_{e \in \mathcal{E}} \sum_{k=1}^{K} \sum_{m=1}^{M_k} f_e(l_e) r_k(m) I_{e \in \mathcal{P}_k(m)} \\ &= \frac{1}{K} \sum_{e \in \mathcal{E}} f_e(l_e) l_e, \end{aligned} \tag{4}$$

where the second equality holds since l_e is given by $\sum_{k=1}^{K} \sum_{m=1}^{M_k} r_k(m) I_{e \in \mathcal{P}_k(m)}$.

Remark 1. The link delay function and the corresponding delay measure assumed in this paper have been broadly used in the literature; see [7], [8], [20], [22], [23], [35] and the references therein. For the above works, link delay function ($f_e(l_e)$ in (2)) is defined as the delay (or cost) per bit and then the delay of the k-th SD pair ($f_k(\mathcal{R})$ in (2)) is given by the aggregate delay of all bits caused by the k-th SD pair per second.

Remark 2. In order to analyze the impact of multi-path routing or load balancing in wireless ad hoc networks, a simple analytical model has been used in [9]–[11] and the references therein. Specifically, wireless ad hoc networks has been simplified to have point-to-point communication links between any two nodes located within a given distance [9]–[11]. A major drawback for this model is the lack of interference consideration, which is one of the key characteristics of wireless networks. Also, network dynamics cannot be captured by this model. Nonetheless, this simplified model provides a useful analytical tool for analyzing the effect of multi-path routing in large-scale networks and for designing geometric routing for load balancing. The considered graph-based model in this paper contains the above analytical model. In section 6, we will simulate the performance of several routing algorithms under similar assumptions used in [9]–[11] .

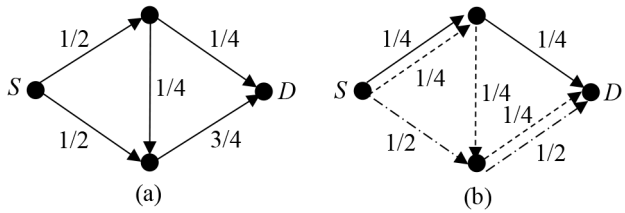


Fig. 2. Original routing (a), and equivalent representation with three paths (b).

Based on the average delay defined above, the minimum delay routing problem can be defined as the follow.

(P1) *The minimum delay routing problem is to find $\mathcal{R}^*$ such that*

$$\mathcal{R}^* = \arg\min_{\mathcal{R}} \{f(\mathcal{R})\}, \tag{5}$$

where the minimization is taken over all possible $\mathcal{R}$.

Remark 3. Note that any routing with infinitely many flows can be represented as $\mathcal{R}$ using the Ford–Fulkerson algorithm [36]. For instance, the routing in Fig. 2. (a) can be represented as the routing with three simple flows in Fig. 2. (b). Hence, $\mathcal{R}^*$ in (5) achieves the minimum delay over all possible routing.

2.2 Random Heavy Traffic Model

In this paper, we study the minimum delay routing problem (P1) in heavy traffic regime, in which minimizing the network delay is of importance to overall network performance. We assume that each SD pair is uniformly distributed at random and independent of each other. That is, a source is chosen over the N nodes at random and its destination is again chosen at random over the rest of the $N-1$ nodes. The rate of the k-th SD pair r_k is drawn from independent and identically distributed (i.i.d.) probability distribution with finite mean $\mu > 0$ and finite variance $\sigma^2 > 0$. We will be dealing with events which take place with high probability (whp), i.e., with probability converging to one as the number of SD pairs K tends to infinity. We will also use the notation $\doteq$, $\dot{\geq}$, and $\dot{\leq}$ to mean that the corresponding equality or inequality holds whp as $K \rightarrow \infty$.

2.3 Distributed Routing

In order to find the routing that minimizes the average delay in (P1), a central controller is needed to coordinate

the routing paths of all SD pairs. In practice, assuming to have such a central controller is, however, unrealistic for most cases. Furthermore, it is hard to acquire global load information at a central controller to set the minimum delay routing $\mathcal{R}^*$. Motivated by those reasons, our primary goal in this paper is to develop an efficient distributed routing algorithm based on the following design principles.

- **Decentralized processing**: When each SD pair is joining the network, it should be able to set up its routing path in a distributed manner.
- **Scalable load information**: Each SD pair should establish its routing path without any or only with a partial view of load information. Specifically, the required link load information at each SD pair should be scalable, i.e., the order of the number of required link loads should grow slowly compared to the network size N.
- **Load balancing**: Notice that minimizing the average delay is closely related to load balancing between links. Hence, when the link delay functions are similar to each other, the link loads should be also increased similar to each other as K increases. Specifically the multiplicative gap between the maximum and minimum link loads should be bounded below a certain fixed value independent of K with increasing K.

3 Lower Bound on Average Delay

In this section, we derive a lower bound on the average delay achieved by the minimum delay routing in (5). In the following lemma, we first derive upper and lower bounds on the aggregate rate of the SD pairs whose sources and destinations are equal to a specific node pair.

Lemma 1. *Consider a network $\mathcal{G}$ with K randomly distributed SD pairs. Then for sufficiently large K,*

$$\left|\sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)} - \frac{K\mu}{N(N-1)}\right| \le \sqrt{K\log K} \tag{6}$$

whp for all $i,j \in \{1,\cdots,N\}$ and $i \neq j$.

Since r_k and $I_{(s_k=i)\cap(d_k=j)}$ are independent of each other and also independent of different k, the random variable $r_k I_{(s_k=i)\cap(d_k=j)}$ is i.i.d. with mean $\mu/(N(N-1))$ and variance $\sigma^2/(N(N-1))$. Hence, from Chebychev's inequality,

$$\Pr\left[\left|\frac{1}{K}\sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)} - \frac{\mu}{N(N-1)}\right| \ge \epsilon\right] \le \frac{\sigma^2}{N(N-1)K\epsilon^2}. \tag{7}$$

Then from the union bound,

$$\Pr\left[\left|\frac{1}{K}\sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)} - \frac{\mu}{N(N-1)}\right| < \epsilon \text{ for } i,j \in \{1,\cdots,N\}, i \neq j\right] \tag{8}$$

is greater than $1-\sigma^2/(K\epsilon^2)$. By setting $\epsilon = \sqrt{(\log K)/K}$, we have

$$\left|\sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)} - \frac{K\mu}{N(N-1)}\right| \le \sqrt{K\log K} \tag{9}$$

for $i,j \in \{1,\cdots,N\}, i \neq j$ with probability greater than $1-\sigma^2/\log K$, which converges to one as K increases. In conclusion, Lemma 1 holds.

Lemma 1 shows that, in the limit of large K, the aggregate rate of the SD pairs whose sources and destinations are equal to a specific node pair is approximately given by $\frac{K\mu}{N(N-1)}$ whp. By using Lemma 1, we show a lower bound on $f(\mathcal{R}^*)$ in the following theorem. The basic idea is that we first consider a lower bound on the amount of load for each SD pair, that is given by $r_k \min_{m\in\{1,\cdots,M_k\}}\{|\mathcal{P}_k(m)|\}$ for the k-th SD pair. Then we argue that this entire amount of load $\sum_{i=1}^{K} r_k \min_{m\in\{1,\cdots,M_k\}}\{|\mathcal{P}_k(m)|\}$ is ideally distributed over the network.

Theorem 1. *Consider a network $\mathcal{G}$ with K randomly distributed SD pairs. If there exist $x_0 > 0$ and $e_0 \in \mathcal{E}$ such that $\min_{e\in\mathcal{E}}\{f_e(x)\} = f_{e_0}(x)$ for all $x \ge x_0$, then for sufficiently large K,*

$$f(\mathcal{R}^*) \ge \mu(1-\epsilon_1)\bar{h} f_{e_0}(K\mu(1-\epsilon_1)\bar{h}/|\mathcal{E}|) \tag{10}$$

whp, where

$$\bar{h} = \frac{1}{N(N-1)} \sum_{i,j\in\{1,\cdots,N\}, i\neq j} \min_{m\in\{1,\cdots,M_{i,j}\}}\{|\mathcal{P}_{i,j}(m)|\} \tag{11}$$

and $\epsilon_1 = \frac{\sqrt{(\log K)/K}}{\mu N(N-1)}$, which converges to zero as K increases.

Let l_e^* denote the load of link e assuming the minimum delay routing $\mathcal{R}^*$. Denote $h_k = \min_{m\in\{1,\cdots,M_k\}}\{|\mathcal{P}_k(m)|\}$, which is the minimum number of hops from s_k to d_k. Since $|\mathcal{P}_k(m)| \ge h_k$ for all $m \in \{1,\cdots,M_k\}$, the k-th SD pair imposes at least $h_k r_k$ amount of load on the network. Hence,

$$\begin{aligned}\sum_{e\in\mathcal{E}} l_e^* &\ge \sum_{k=1}^{K} h_k r_k \\ &\overset{(a)}{=} \sum_{k=1}^{K} \min_{m\in\{1,\cdots,M_{s_k,d_k}\}}\{|\mathcal{P}_{s_k,d_k}(m)|\} r_k \\ &\overset{(b)}{=} \sum_{i,j\in\mathcal{V}, i\neq j} \sum_{k=1}^{K} \min_{m\in\{1,\cdots,M_{s_k,d_k}\}}\{|\mathcal{P}_{s_k,d_k}(m)|\} r_k I_{(s_k=i)\cap(d_k=j)} \\ &\overset{(c)}{=} \sum_{i,j\in\mathcal{V}, i\neq j} \min_{m\in\{1,\cdots,M_{i,j}\}}\{|\mathcal{P}_{i,j}(m)|\} \sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)},\end{aligned} \tag{12}$$

where (a) follows from the definition of M_k and $\mathcal{P}_k(m)$, (b) follows since $I_{(s_k=i)\cap(d_k=j)}$ is one if $i=s_k$ and $j=d_k$ and zero otherwise for all $i,j \in \mathcal{V}, i \neq j$, and (c) follows since $\min_{m\in\{1,\cdots,M_{s_k,d_k}\}}\{|\mathcal{P}_{s_k,d_k}(m)|\} r_k I_{(s_k=i)\cap(d_k=j)} = \min_{m\in\{1,\cdots,M_{i,j}\}}\{|\mathcal{P}_{i,j}(m)|\} r_k I_{(s_k=i)\cap(d_k=j)}$ for all k. From Lemma 1, $\sum_{k=1}^{K} r_k I_{(s_k=i)\cap(d_k=j)} \dot{\ge} \frac{K\mu}{N(N-1)} - \sqrt{K\log K}$ for all $i,j \in \mathcal{V}, i \neq j$. Then

$$\sum_{e\in\mathcal{E}} l_e^* \dot{\ge} K\mu(1-\epsilon_1)\bar{h}. \tag{13}$$

Define $f_{\text{conv}}(x) = \text{conv}(\min_{e\in\mathcal{E}} f_e(x))$, where $\text{conv}(f(x))$ denotes the convex envelop of $f(x)$. Then the minimum average delay is given by

$$\begin{aligned} f(\mathcal{R}^*) &= \frac{1}{K}\sum_{e\in\mathcal{E}} f_e(l_e^*)l_e^* \\ &\overset{(a)}{\geq} \frac{1}{K} \min_{\sum_{e\in\mathcal{E}} l_e \geq K\mu(1-\epsilon_1)\bar{h}} \left\{\sum_{e\in\mathcal{E}} f_e(l_e)l_e\right\} \\ &\overset{(b)}{\geq} \frac{1}{K} \min_{\sum_{e\in\mathcal{E}} l_e \geq K\mu(1-\epsilon_1)\bar{h}} \left\{\sum_{e\in\mathcal{E}} f_{\text{conv}}(l_e)l_e\right\} \\ &\overset{(c)}{=} \frac{|\mathcal{E}|}{K} f_{\text{conv}}\left(\frac{K\mu(1-\epsilon_1)\bar{h}}{|\mathcal{E}|}\right)\frac{K\mu(1-\epsilon_1)\bar{h}}{|\mathcal{E}|} \\ &\overset{(d)}{\doteq} \mu(1-\epsilon_1)\bar{h} f_{e_0}\left(\frac{K\mu(1-\epsilon_1)\bar{h}}{|\mathcal{E}|}\right), \end{aligned} \tag{14}$$

where (a) follows from (12), (b) follows since $f_e(x) \geq f_{\text{conv}}(x)$ for all $e \in \mathcal{E}$ and $x \geq 0$, (c) follows since $l_e = \frac{K\mu(1-\epsilon_1)\bar{h}}{|\mathcal{E}|}$ achieves the minimum, and (d) follows from the assumption that there exist $x_0 > 0$ and $e_0 \in \mathcal{E}$ such that $\min_{e\in\mathcal{E}}\{f_e(x)\} = f_{e_0}(x)$ for all $x \geq x_0$. In conclusion, Theorem 1 holds.

Remark 4. Any set of non-negative increasing polynomial or exponential link delay functions satisfies the condition in Theorem 1. That is, the delay lower bound (10) is valid for a set of link delay functions $\{f_e(x) = c_e x^{\alpha_e}$ or $f_e(x) = c_e e^{\beta_e x}\}_{e\in\mathcal{E}}$, where $c_e > 0$, $\alpha_e \geq 1$, and $\beta_e \geq 0$. The condition in Theorem 1 is also satisfied if $f_e(x)$ is the same for all $e \in \mathcal{E}$.

We will demonstrate by simulation in Section 6 that the delay lower bound in Theorem 1 becomes tight as $K \to \infty$ for a certain class of regularly deployed networks. Furthermore, for this class of networks, the presented delay lower bound is achievable in a fully distributed manner, showing that there is no penalty due to the distributed routing.

4 Minimum Delay Distributed Routing

Recall the minimum delay routing problem (P1). From (1) to (4), the minimum delay routing problem (P1) can be represented as the following non-linear program:

$$\mathcal{R}^* = \arg\min_{\mathcal{R}} \left\{\sum_{e\in\mathcal{E}} f_e(l_e)l_e\right\} \tag{15}$$

subject to

$$\sum_{m=1}^{M_k} r_k(m) = r_k \text{ for } k \in \{1,\cdots,K\}, \tag{16}$$

$$l_e = \sum_{k=1}^{K}\sum_{m=1}^{M_k} r_k(m) I_{e\in\mathcal{P}_k(m)} \text{ for } e \in \mathcal{E}, \tag{17}$$

$$r_k(m) \in \{0\} \cup \mathbb{R}_+ \text{ for } k \in \{1,\cdots,K\}, m \in \{1,\cdots,M_k\}. \tag{18}$$

As mentioned before, a central controller is required to establish $\mathcal{R}^*$, which is unrealistic for most practical networks. Even assuming a central controller with global load information, (P1) is NP hard [12]–[14] and computationally untractable as the network size increases.

Instead, we will define the minimum delay distributed routing problem, which will be implementable in a fully distributed manner. We will show that, as K tends to infinity, both the minimum delay routing and the minimum delay distributed routing provide a bounded multiplicative gap whp between any link loads, if the link delay functions satisfy a certain similarity condition.

4.1 Minimum Delay Distributed Routing

We consider the following minimum delay distributed routing problem (P2) in which each SD pair is able to set its routing in a distributed manner based on the current load information of the network.

(P2) *For $k \in \{1,\cdots,K\}$, the minimum delay distributed routing is to find $\mathcal{R}_k^{**}$ such that*

$$\mathcal{R}_k^{**} = \arg\min_{\mathcal{R}_k} \left\{f\left(\mathcal{R}_k, \{\mathcal{R}_i^{**}\}_{i=1}^{k-1}\right)\right\}, \tag{19}$$

where the minimization is taken over all possible $\mathcal{R}_k$. Here, $\{\mathcal{R}_i^{**}\}_{i=1}^{k-1}$ denotes the routing of the previous $k-1$ SD pairs. Then the overall routing can be represented as $\mathcal{R}^{**} = \{\mathcal{R}_k^{**}\}_{k=1}^{K}$.

Similar to (P1), the minimum delay distributed routing problem (P2) is given by the following non-linear program:

$$\mathcal{R}_k^{**} = \arg\min_{\mathcal{R}_k} \left\{\sum_{e\in\mathcal{E}} f_e(l_e)l_e\right\} \tag{20}$$

subject to

$$\sum_{m=1}^{M_k} r_k(m) = r_k, \tag{21}$$

$$l_e = \sum_{m=1}^{M_k} r_k(m) I_{e\in\mathcal{P}_k(m)} + \sum_{i=1}^{k-1}\sum_{m=1}^{M_i} r_i^{**}(m) I_{e\in\mathcal{P}_i(m)} \text{ for } e \in \mathcal{E}, \tag{22}$$

$$r_k(m) \in \{0\} \cup \mathbb{R}_+ \text{ for } m \in \{1,\cdots,M_k\}, \tag{23}$$

where $\mathcal{R}_i^{**} = \{r_i^{**}(m)\}_{m=1}^{M_i}$ for $i \in \{1,\cdots,k-1\}$.

Unlike (P1), each SD pair sets its routing sequentially in a distributed manner in (P2). Also, each SD pair finds the routing strategy that minimizes the average delay (not its own delay) by considering the load information of previously assigned SD pairs.

4.2 Asymptotic Load Balancing

If the link delay functions have a similar tendency to each other, the minimum delay routing $\mathcal{R}^*$ may naturally achieve load balancing between the links by minimizing the average delay. The following theorem shows that not only the minimum delay routing $\mathcal{R}^*$ but also the minimum delay distributed routing $\mathcal{R}^{**}$ can achieve asymptotic load balancing whp in the limit of large K. More specifically, under a certain regularity condition of the link delay functions, the link loads increase with the same order of K whp as K increases. This result demonstrates that there is no penalty due to the distributed routing in the sense of asymptotic load balancing.

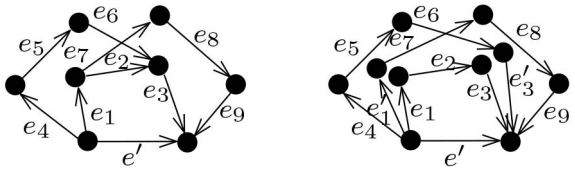


Fig. 3. Example of distinguishable paths between the head and the tail of e' on $\mathcal{G}$ (a), and $\mathcal{G}'$ (b), where the link delay functions of e_1 and e'_1 and the link delay functions of e_3 and e'_3 are the same.

Theorem 2. *Consider a network $\mathcal{G}$ with K randomly distributed SD pairs. Let $l^*_{e,K}$ and $l^{**}_{e,K}$ denote the load of $e \in \mathcal{E}$ when $\mathcal{R}^*$ and $\mathcal{R}^{**}$ are applied, respectively. If there exist $x_0 > 0$, $\epsilon_0 \in \mathcal{E}$, and $\delta \geq 1$ such that $f_{e_0}(\delta x) \geq f_e(x)$ for all $e \in \mathcal{E}$ and $x \geq x_0$, then for sufficiently large K,*

$$\frac{\max_{e\in\mathcal{E}}\{l^*_{e,K}\}}{\min_{e\in\mathcal{E}}\{l^*_{e,K}\}} \leq 2(1+\delta N(N-1))N(N-1)+\epsilon_2 \quad (24)$$

and

$$\frac{\max_{e\in\mathcal{E}}\{l^{**}_{e,K}\}}{\min_{e\in\mathcal{E}}\{l^{**}_{e,K}\}} \leq 2(1+\delta N(N-1))N(N-1)+\epsilon_2 \quad (25)$$

whp, where $\epsilon_2 = \frac{2(1+\delta N(N-1))N^2(N-1)^2}{\mu}\sqrt{\frac{\log K}{K}}$, which converges to zero as K increases.

To prove (24), we analyze upper and lower bounds on $l^*_{e,K}$, valid for all $e \in \mathcal{E}$. Hence, the derived upper and lower bounds hold for $\max_{e\in\mathcal{E}}\{l^*_{e,K}\}$ and $\min_{e\in\mathcal{E}}\{l^*_{e,K}\}$ and provide an upper bound on $\max_{e\in\mathcal{E}}\{l^*_{e,K}\}/\min_{e\in\mathcal{E}}\{l^*_{e,K}\}$.

First, consider an upper bound on $l^*_{e,K}$. From lemma 1, we have $\sum_{k=1}^{K} r_k \doteq K\mu + N(N-1)\sqrt{K\log K}$. By assuming that the routing paths of all SD pairs includes e,

$$l^*_{e,K} \doteq K\mu + N(N-1)\sqrt{K\log K} \quad (26)$$

for all $e \in \mathcal{E}$.

Second, consider a lower bound on $l^*_{e,K}$. We show that, $l^{**}_{e,K} \doteq c_0 K$ for all $e \in \mathcal{E}$, where $c_0 = \frac{\mu}{2(1+\delta N(N-1))N(N-1)}$. To prove this statement by contradiction, assume that $l^*_{e',K} \leq c_0 K$ for arbitrary $e' \in \mathcal{E}$. Let $\mathcal{P}'(1)$ to $\mathcal{P}'(M')$ denote the distinct paths from the head of e' to the tail of e' (see Fig. 3. (a)). Without loss of generality, denote $\mathcal{P}'(1) = \{e'\}$, which is the shortest path. Now consider the routing of the SD pairs whose sources and destinations are equal to the head and the tail of e', respectively. Let $r_{\max}$ denote the maximum aggregate rate of these SD pairs that satisfies $l^*_{e',K} \leq c_0 K$. To obtain an upper bound on $r_{\max}$, we first ignore the other SD pairs in the network. Then we define an auxiliary network $\mathcal{G}'$ from the original network $\mathcal{G}$ such that if link $e \in \mathcal{E}, e \neq e'$, appears in $\mathcal{P}'(2)$ to $\mathcal{P}'(M')$ more than one time, we put additional nodes and make the corresponding paths disjoint to each other (see Fig. 3. (b)). Then, routing over $\mathcal{G}'$ provides an upper bound on $r_{\max}$ since the delay routed through $\mathcal{P}'(m)$ on $\mathcal{G}'$ is less than or equal to that on $\mathcal{G}$ for all $m \in \{2, \cdots, M'\}$. Hence $r_{\max}$ is upper bounded as

$$r_{\max} \leq \max\left\{\sum_{m=1}^{M'} r(m)\right\} \quad (27)$$

subject to

$$r(1) = c_0 K, \quad (28)$$

$$\sum_{e\in\mathcal{P}'(m)} f_e(r(m)) \leq f_{e'}(c_0 K) \text{ for } m \in \{2, \cdots, M'\}, \quad (29)$$

$$r(m) \in \{0\} \cup \mathbb{R}_+ \text{ for } m \in \{2, \cdots, M'\}, \quad (30)$$

where $r(m)$ is the aggregate rate routed through $\mathcal{P}'(m)$. Obviously, the minimum delay routing $\mathcal{R}^*$ routes with a rate of $c_0 K$ through $\mathcal{P}'(1)$, which gives the condition (28). The condition (29) appears since $\mathcal{R}^*$ routes with a rate more than $c_0 K$ through $\mathcal{P}'(1)$ if $\sum_{e\in\mathcal{P}'(m)} f_e(r(m)) > f_{e'}(c_0 K)$ for some $m \in \{2, \cdots, M'\}$.

Now consider an upper bound on $r(m)$ in the limit of large K. From (28), we have $r(1) \doteq c_0 K$. Assume that $m \geq 2$. For the case where $r(m)$ is upper bounded by a constant independent of K, we have $r(m) \doteq \delta c_0 K$. For the case where $r(m)$ is an increasing function of K, from (29) and (30), we have $f_e(r(m)) \leq f_{e'}(c_0 K)$, which gives

$$f_e(r(m)) \leq f_{e_0}(\delta c_0 K) \quad (31)$$

if $K \geq x_0/(\delta c_0)$, where $e \in \cup_{m\in\{2,\cdots,M'\}}\mathcal{P}'(m)$. Here, the condition $f_{e_0}(\delta x) \geq f_e(x)$ for all $e \in \mathcal{E}$ and $x \geq x_0$ is used. From (31), we again have $r(m) \doteq \delta c_0 K$. This is because that the condition $f_{e_0}(\delta x) \geq f_e(x)$ for all $e \in \mathcal{E}$ and $x \geq x_0$ implies $\min_{e\in\mathcal{E}}\{f_e(x)\} = f_{e_0}(x)$ for all $x \geq x_0$. Hence,

$$\begin{aligned} r_{\max} &\leq \max\left\{\sum_{m=1}^{M'} r(m)\right\} \\ &\doteq c_0 K + (M'-1)\delta c_0 K \\ &\leq (1+\delta N(N-1))c_0 K \\ &= \frac{K\mu}{2N(N-1)}, \end{aligned} \quad (32)$$

where $M' \leq |\mathcal{E}| \leq N(N-1)$ is used.

However, from Lemma 1, for sufficiently large K, the aggregate rate of the SD pairs whose sources and destinations are equal to the head and the tail of e' is lower bounded by $\frac{K\mu}{N(N-1)} - \sqrt{K\log K}$ whp, which is larger than the upper bound on $r_{\max}$ presented in (32) in the limit of large K. This means that, for sufficiently large K, $\mathcal{R}^*$ cannot establish routing paths of all SD pairs whp while satisfying $l^*_{e',K} \leq c_0 K$. Since this statement holds for arbitrary $e' \in \mathcal{E}$, we have

$$l^*_{e,K} \doteq \frac{K\mu}{2(1+\delta N(N-1))N(N-1)} \quad (33)$$

for all $e \in \mathcal{E}$.

Finally, from (26) and (33), we have

$$\frac{\max_{e\in\mathcal{E}}\{l^*_{e,K}\}}{\min_{e\in\mathcal{E}}\{l^*_{e,K}\}} \doteq 2(1+\delta N(N-1))N(N-1)+\epsilon_2. \quad (34)$$

Recall the minimum delay distributed routing $\mathcal{R}^{**}$ in (20) to (23). The only difference from the minimum delay routing $\mathcal{R}^*$ in (15) and (18) is that each SD pair sets its routing sequentially in a distributed manner for this case.

Hence, the same bounds in (26) and (33) are still valid for $l^{**}_{e,K}$ and, as a result, the same bound in (34) holds for $\max_{e\in\mathcal{E}}\{l^{**}_{e,K}\}/\min_{e\in\mathcal{E}}\{l^{**}_{e,K}\}$. In conclusion, Theorem 2 holds.

Remark 5. Any set of link delay functions $\{f_e(x) = c_e x^{\alpha}\}_{e\in\mathcal{E}}$ or $\{f_e(x) = c_e e^{\beta x}\}_{e\in\mathcal{E}}$ satisfies the condition in Theorem 2, where $c_e > 0$, $\alpha \geq 1$, and $\beta \geq 0$. The condition in Theorem 1 is also satisfied if $f_e(x)$ is the same for all $e \in \mathcal{E}$. |

5 Distributed Routing Algorithms

In Section 4 we showed that $\mathcal{R}^{**}$ provides a bounded multiplicative gap whp between any link loads for a wide class of link delay functions, which is implementable in a fully distributed manner. However, the minimum delay distributed routing problem (P2) is still NP hard and therefore very challenging to establish the optimal routing as the network size increases.

In this section, we study distributed routing algorithms computable within polynomial time. We consider three possible scenarios regarding the amount of available load information for distributed routing: global load information, partial load information, and no available load information. For each of these three cases, we propose a polynomial time distributed routing algorithm.

5.1 Routing With Global Load Information

In the following, we propose a polynomial time distributed routing algorithm that is able to achieve asymptotic load balancing in the limit of large K. The key intuition is that a single-path routing minimizing its own delay is enough for each SD pair to achieve a bounded multiplicative gap whp between any link loads and, for this case, the minimum delay routing path can be found within polynomial time by using the well-known Dijkstra's algorithm with proper link costs. The following pseudo code describes the detailed routing algorithm, which uses Dijkstra's algorithm to find the minimum delay routing path.

Routing A

Result: $\mathcal{R}^{(A)}$.
Initialize $\{l_e = 0\}_{e\in\mathcal{E}}$;
for $k = 1{:}K$ **do**
  Set $\mathcal{P}_k(m_{\min})$ as the minimum cost path among $\{\mathcal{P}_k(m)\}_{m=1}^{M_k}$ obtained from Dijkstra's algorithm by assuming the cost of link e as $f_e(l_e + r_k)$ for all $e \in \mathcal{E}$;
  Set $\mathcal{R}_k^{(A)} = \{r_k^{(A)}(m)\}_{m=1}^{M_k}$, where $r_k^{(A)}(m) = r_k$ if $m = m_{\min}$ and $r_k^{(A)}(m) = 0$ otherwise;
  Update $l_e \rightarrow l_e + r_k$ for all $e \in \mathcal{P}_k(m_{\min})$;
end
Set $\mathcal{R}^{(A)} = \{\mathcal{R}_k^{(A)}\}_{k=1}^{K}$;

The next corollary shows that $\mathcal{R}^{(A)}$ again provides a bounded multiplicative gap whp between any link loads in the limit of large K.

Corollary 1. *Consider a network $\mathcal{G}$ with K randomly distributed SD pairs. Let $l_{e,K}^{(A)}$ denote the load of $e \in \mathcal{E}$ when $\mathcal{R}^{(A)}$ is applied. If there exist $x_0 > 0$, $\epsilon_0 \in \mathcal{E}$, and $\delta \geq 1$ such that $f_{e_0}(\delta x) \geq f_e(x)$ for all $e \in \mathcal{E}$ and $x \geq x_0$, then for sufficiently large K,*

$$\frac{\max_{e\in\mathcal{E}}\{l_{e,K}^{(A)}\}}{\min_{e\in\mathcal{E}}\{l_{e,K}^{(A)}\}} \geq 2(1+\delta N(N-1))N(N-1) + \epsilon_2, \quad (35)$$

where $\epsilon_2 = \frac{2(1+\delta N(N-1))N^2(N-1)^2}{\mu}\sqrt{\frac{\log K}{K}}$, which converges to zero as K increases.

The overall proof is almost the same as that in Theorem 2 and we only explain the differences here. Similar to (P1) and (P2), $\mathcal{R}^{(A)}$ is the solution of the following non-linear program:

$$\mathcal{R}_k^{(A)} = \arg\min_{\mathcal{R}_k}\left\{\sum_{e\in\mathcal{E}} f_e(l_e)l_e\right\} \quad (36)$$

subject to

$$\sum_{m=1}^{M_k} r_k(m) = r_k, \quad (37)$$

$$l_e = \sum_{m=1}^{M_k} r_k(m) I_{e\in\mathcal{P}_k(m)} + \sum_{i=1}^{k-1}\sum_{m=1}^{M_i} r_i^{(A)}(m) I_{e\in\mathcal{P}_i(m)} \text{ for } e \in \mathcal{E}, \quad (38)$$

$$r_k(m) \in \{0, r_k\} \text{ for } m \in \{1, \cdots, M_k\}. \quad (39)$$

The only difference from $\mathcal{R}^{**}$ is the condition (39) instead of (23). Therefore, the upper and lower bounds in Theorem 2 hold for $\mathcal{R}^{(A)}$. In conclusion, Corollary 1 holds.

5.2 Routing With and Without Partial Load Information

In this subsection, we consider the case where only a partial view of load information is available for each SD pair to set up its routing. We restrict each SD pair to access load information of only M chosen paths. The main challenge is how to efficiently reduce the average delay in a distributed manner by the help of this partial load information. Fortunately, we can find some hints from the previous results. Recall Routing A in which each SD pair simply routes through a single path minimizing its own delay (not the average delay), but the average delay can be efficiently reduced as K increases.

For each node pair (i, j), we set the same number of $M_p \in \mathbb{Z}_+$ predetermined paths from node i to node j, where $i, j \in \{1, \cdots, N\}$ and $i \neq j$. Let $\bar{\mathcal{P}}_{i,j}(m) \in \{\mathcal{P}_{i,j}(n)\}_{n=1}^{M_{i,j}}$ denote the m-th predetermined path from node i to node j, where $m \in \{1, \cdots, M_p\}$. We will propose distributed routing algorithms that only route through these predetermined paths. Hence, the construction of such predetermined paths is closely related to the delay performance. Based on Routing A, we construct predetermined paths as the following.

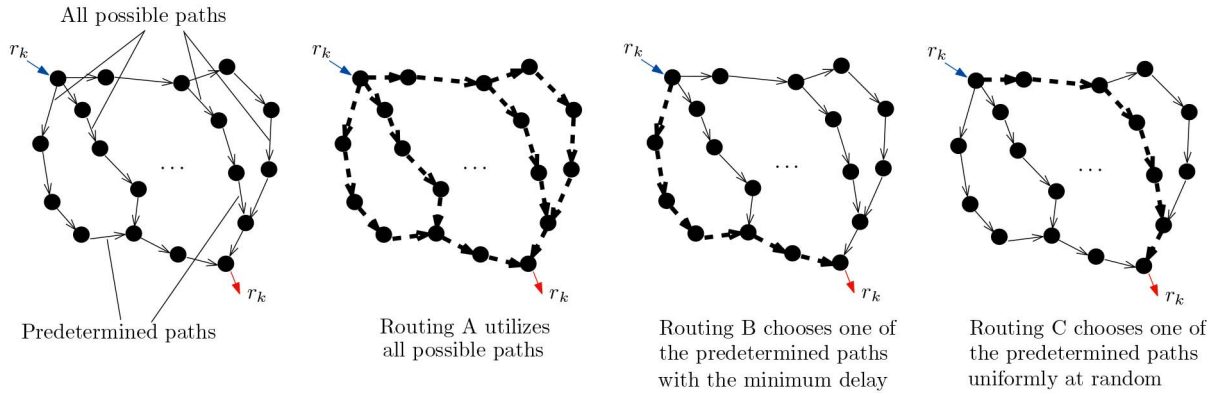


Fig. 4. Routing examples for Routing A, B, and C.

Construction of predetermined paths

Result: $\bar{\mathcal{P}}$.
for $i = 1{:}N$, $j = 1{:}N$, $i \neq j$ **do**
  for $m = 1{:}M_p$ **do**
    Generate $K-1$ *hypothetical* SD pairs uniformly at random and generate the K-th *hypothetical* SD pair whose source and destination are equal to node i and node j respectively;
    Set the routing of these K SD pairs using Routing A and denoted it by $\mathcal{R}^{(A)}$;
    Set $\bar{\mathcal{P}}_{i,j}(m) = \mathcal{P}_{i,j}(n)$ for n satisfying $r_K^{(A)}(n) = r_K$;
  end
  Set $\bar{\mathcal{P}}_{i,j} = \{\bar{\mathcal{P}}_{i,j}(m)\}_{m=1}^{M_p}$;
end
Set $\bar{\mathcal{P}} = \{\bar{\mathcal{P}}_{i,j}\}_{i,j\in\{1,\cdots,N\},i\neq j}$;

For the proposed construction, each pair of nodes can set up its predetermined paths in a fully distributed manner by separately generating hypothetical traffic independent of the other node pairs.

Remark 6. Although the real traffic may be different from the hypothetical traffic, we will demonstrate by simulation in Section 6 that the selected predetermined paths based on the hypothetical traffic can effectively reduce the overall delay for various network environments. The first reason for the above property is that the selected predetermined paths naturally detour the network center, which can be verified in Fig. 7. The second and more important reason is that these paths can be used to avoid congestion between multiple SD pairs while maintaining their own delay small, which can be verified in Fig. 8. The following Routing B and C will use these predetermined paths for routing SD pairs.

By setting $M_p = M$ and selecting the minimum delay path among the predetermined paths, each SD pair can successfully set up its routing from partial load information of only M paths. The following pseudo code describes the detailed routing algorithm, where the cost of a path is defined as the sum of the cost of each link in the path.

Routing B

Result: $\mathcal{R}^{(B)}$.
Fix $M_p = M$ and set $\bar{\mathcal{P}}$ using the proposed construction of predetermined paths;
Initialize $\{l_e = 0\}_{e\in\mathcal{E}}$;
for $k = 1{:}K$ **do**
  Set $\mathcal{P}_k(m_{\min})$ as the minimum cost path among $\{\bar{\mathcal{P}}_{s_k,d_k}(m)\}_{m=1}^{M}$ with the cost of link e as $f_e(l_e + r_k)$ for all $e \in \mathcal{E}$;
  Set $\mathcal{R}_k^{(B)} = \{r_k^{(B)}(m)\}_{m=1}^{M_k}$, where $r_k^{(B)}(m) = r_k$ if $m = m_{\min}$ and $r_k^{(B)}(m) = 0$ otherwise;
  Update $l_e \rightarrow l_e + r_k$ for all $e \in \mathcal{P}_k(m_{\min})$;
end
Set $\mathcal{R}^{(B)} = \{\mathcal{R}_k^{(B)}\}_{k=1}^{K}$;

Even without any load information, each SD pair can effectively reduce the average delay in a distributed manner by routing through one of the predetermined paths at random. The following pseudo code describes the detailed routing algorithm.

Routing C

Result: $\mathcal{R}^{(C)}$.
Set $\bar{\mathcal{P}}$ using the proposed construction of predetermined paths;
Initialize $\{l_e = 0\}_{e\in\mathcal{E}}$;
for $k = 1{:}K$ **do**
  Set $\mathcal{P}_k(m_{\min})$ as the path chosen uniformly at random among $\{\bar{\mathcal{P}}_{s_k,d_k}(m)\}_{m=1}^{M_p}$;
  Set $\mathcal{R}_k^{(C)} = \{r_k^{(C)}(m)\}_{m=1}^{M_k}$, where $r_k^{(C)}(m) = r_k$ if $m = m_{\min}$ and $r_k^{(C)}(m) = 0$ otherwise;
  Update $l_e \rightarrow l_e + r_k$ for all $e \in \mathcal{P}_k(m_{\min})$;
end
Set $\mathcal{R}^{(C)} = \{\mathcal{R}_k^{(C)}\}_{k=1}^{K}$;

Fig. 4 Illustrates the basic difference between Routing A, B, and C. Whereas Routing A utilizes all possible paths for multi-path routing, Routing B and C choose one of the predetermined paths for single-path routing.

Remark 7. Table 1 summarizes the required load information for routing of the k-th SD pair. As shown in the

TABLE 1
Required Load Information for Routing A, B, C

Algorithm	Required load information for the k-th SD pair
Routing A	$\{l_e\}_{e\in\{\mathcal{P}_{s_k,d_k}(m)\}_{m=1}^{N_k}}$
Routing B	$\{l_e\}_{e\in\{\bar{\mathcal{P}}_{s_k,d_k}(m)\}_{m=1}^{M}}$
Routing C	None

table, Routing A requires load information for all possible paths of the k-th SD pairs, i.e., N_k paths. If a network is connected, then $\{l_e\}_{e\in\{\mathcal{P}_{s_k,d_k}(m)\}_{m=1}^{N_k}}$ becomes load information of all links in a network. Routing B, however, only requires load information of M paths instead of N_k and as we will show in Section 6, Routing B can significantly reduce the required load information while providing the average delay similar to Routing A. Lastly, Routing C do not require load information for routing.

Remark 8. The basic philosophy for Routing C is similar to those of oblivious routing or Valiant's randomized routing [37]–[39] in the sense that they do not require the current state of a network. As proposed by Valiant [37], most oblivious or Valiant's routing algorithms first send a packet to a random node before it is sent to its destination [40]–[42]. However, there are two main drawbacks for the oblivious or Valiant's routing algorithms. The first is that delay can be large by choosing an intermediate relay node at random. The second is that depending on routing algorithms between the source node and an intermediate relay node (and an intermediate relay node to the destination) local congestion may still occur, see Fig. 8 for better understanding. As we will show in Section 6, routing based on predetermined paths can efficiently resolve these problems in the regime of a large number of SD pairs.

Remark 9. For mobile ad hoc networks, rotting based on predetermined paths is hard to apply due to the dynamics of topologies and/or channels. In this case, on-demand routing [15], [16] combining with randomized path selection, similar to Routing C, will be helpful for reducing delay.

In Section 6, we will show that the proposed predetermined path routing algorithms, Routing B and Routing C, can significantly reduce the required load information while providing the average delay similar to Routing A, which requires global load information for routing. Since the proposed predetermined path routing algorithms route through one of the predetermined paths, they can reduce the computational complexity for routing as well compared to Routing A.

6 Simulation Results

In this section, we evaluate the delay performance of the proposed routing algorithms. We consider two kinds of networks: random networks in which N nodes are deployed uniformly at random over a given network area and grid networks in which N nodes are regularly deployed over a given network area [43]–[45]. In simulation, we focus on the linear and quadratic delay functions, i.e., $f_e(x) = x$ and $f_e(x) = x^2$ for all $e \in \mathcal{E}$, respectively. Regrading the traffic pattern, we consider fixed rate of $r_k = 1$ ($\mu = 1$ and $\sigma^2 = 0$) and uniformly distributed rate of $r_k \sim \text{Unif}(0, 2)$ ($\mu = 1$ and $\sigma^2 = 1/3$) for all $k \in \{1 \cdots, K\}$.

Remark 10. Similar delay performance presented in Sections 6.1 and 6.2 can be obtained for various link delay functions and traffic patterns. e.g., polynomial (with higher degrees) or exponential link delay functions and Gaussian-distributed traffic pattern.

6.1 Random Networks

We construct two different types of random networks as the following manner.

- **Random geometric network**: N nodes are uniformly and independently distributed over the sphere in $\mathbb{R}^2$ of a unit area. There exist links (i, j) and (j, i) if the Euclidian distance between nodes i and j is less than or equal to a certain threshold $d_{\max} > 0$.
- **Random torus network**: The node distribution and link connection are the same as those assumed for random geometric network except that N nodes are distributed over the two dimensional square torus of a unit area and the distance between nodes are measured under the torus.

Remark 11. Due to the random construction, the resulting network may or may not be connected at each realization. However, the percolation theory shows that there exists $d_{\max} = \frac{c_{\max}}{\sqrt{N}}$ such that the resulting network is connected whp as $N \to \infty$, where $c_{\max} > 0$ is a constant independent of N [46], [47].

In simulation, we set $d_{\max} = \frac{c_{\max}}{\sqrt{N}}$ with appropriately chosen $c_{\max}$ that guarantees the network being connected whp as $N \to \infty$. The following two examples show the average delays of the proposed routing algorithms in the random networks. For comparison, we also consider the delay lower bound in Theorem 1 and the average delay of the shortest path routing in which each source routes through the shortest path without load information to the corresponding destination.

Experiment 1 (Delay for random geometric networks). *Fig. 5 plots the average delays for the random geometric network. From the figures in the top left and bottom left (or the top right and bottom right), the effect of the traffic pattern is marginal for overall delay performance. This is because that we focus on the regime of a large number of SD pairs and, due to the law of large numbers, the average rate only matters in this regime, which are the same for both deterministic and random traffic patterns, i.e., $\mu = 1$. For the linear delay function, the average delays in the figures in the top left and bottom left increase linearly with K. Similarly, for the quadratic delay function, the average delays in the figures in the top right and bottom right increase quadratically with K. The important thing is that the average delays between the routing algorithms show similar tendency for all four figures. Specifically, the results show that the average delays of the two predetermined path routings, Routing B and Routing*

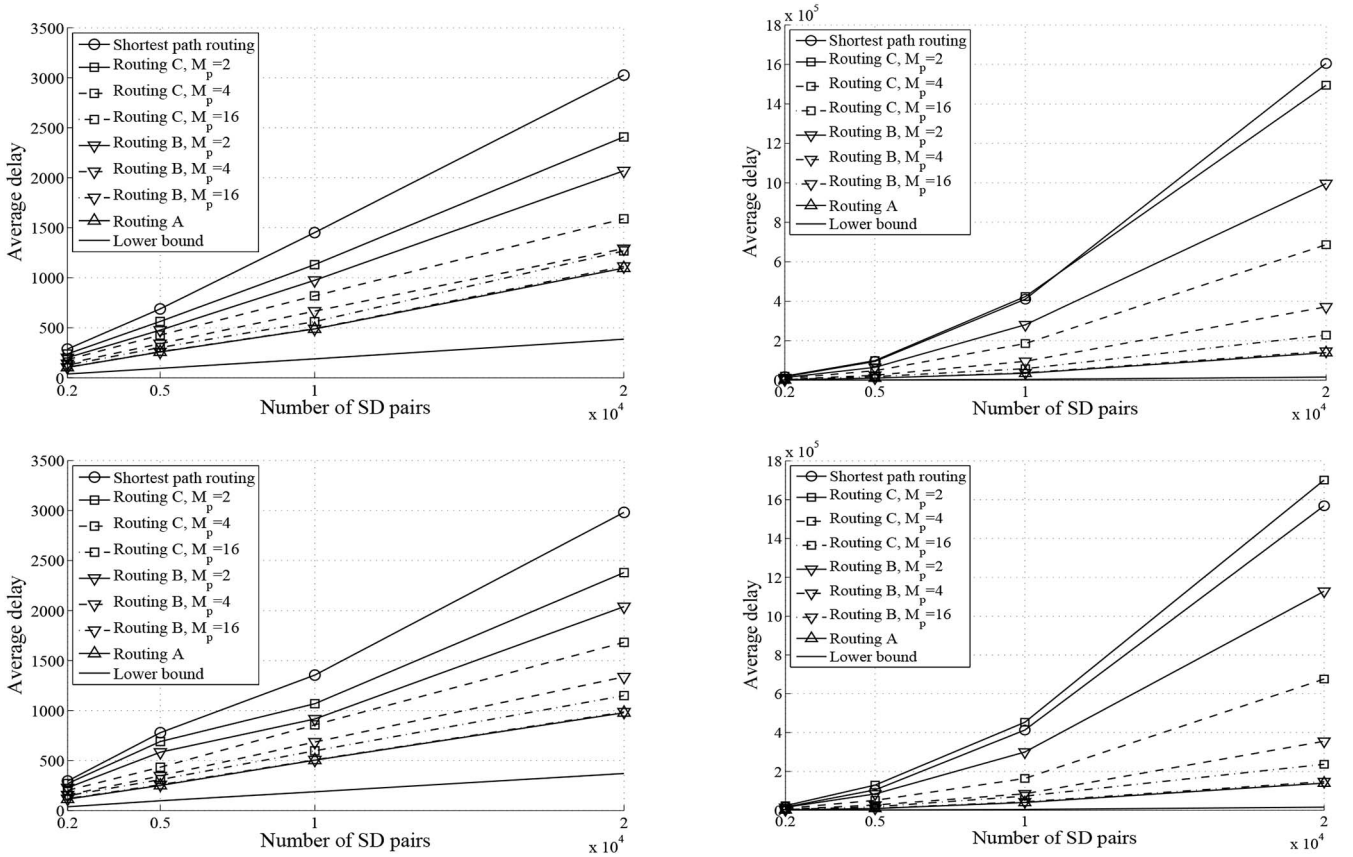


Fig. 5. Average delays for the random geometric network when $N = 500$ and $d_{\max} = 0.14$, where $f_e(x) = x$ and $r_k = 1$ (top left), $f_e(x) = x^2$ and $r_k = 1$ (top right), $f_e(x) = x$ and $r_k \sim \text{Unif}(0, 2)$ (bottom left), and $f_e(x) = x^2$ and $r_k \sim \text{Unif}(0, 2)$ (bottom right) were used.

C, converge to the average delay of Routing A with a small number of predetermined paths. This means that routing only through predetermined paths can significantly reduce the amount of required load information and computational complexity in distributed routing. From the average delays of Routing B and Routing C, we also know that a small amount of partial load information is helpful to reduce the delay compared with the case without any available load information.

Experiment 2 (Delay for random torus networks). *Fig. 6 plots the average delays for the random torus network. Similar to Experiment 1, the average delays between the routing algorithms show similar tendency for all four figures. However, unlike the random geometric network, there exists neither the network center nor the network boundary under the torus and, as a result, load unbalancing between the network center and the network boundary does not happen for this case as pointed out in [9], [10]. Even under the torus, however, the average delay performance is still almost the same as that of the random geometric network as shown in Fig. 5, see Remark 6 for the reason.*

For better understanding on the role of predetermined paths, we first illustrate typical trajectories of predetermined paths and then show the load distribution of the proposed routings in the following two figures.

Experiment 3 (Trajectory of predetermined path). *Fig. 7 plots typical trajectories of predetermined paths and the shortest paths of specific node pairs for the random geometric network. For simplicity, we only plot node locations and do not plot edges between nodes. As shown in the figure, predetermined paths naturally detour the network center in order to avoid the load concentration at the center [9], [10]. Hence the proposed construction of predetermined paths provides a simple and efficient way of geometrically detouring path construction, which has been studied in [11], [19]. More importantly, it provides maximally disjoint routing paths [15], [33], [34], which is able to spread the overall load over all links. This is the primary reason that the predetermined path routings achieve much smaller delay than the shortest path routing for both random geometric and random torus networks (see Experiments 1, 2), which can be also verified from the following figure.*

Experiment 4 (Load distribution). *Fig. 8 plots the histogram of the load distribution, i.e., histogram of $\{l_e\}_{e \in \mathcal{E}}$, for Routings A, B, C, and the shortest path routing in Experiment 1. For the shortest path routing, a large portion of links remains unused or delivers very small amount of load and these less-utilized links severely degrade the average delay performance. On the other hand, this problem is resolved for Routing A since each SD pair sets its routing path based on current load information of the entire network so that links with less amount of load are likely to be chosen, which gives a smaller link delay. From the load distributions of Routings B and C, we know that similar load balancing is indeed achievable by predetermined path routing with a small number of predetermined paths.*

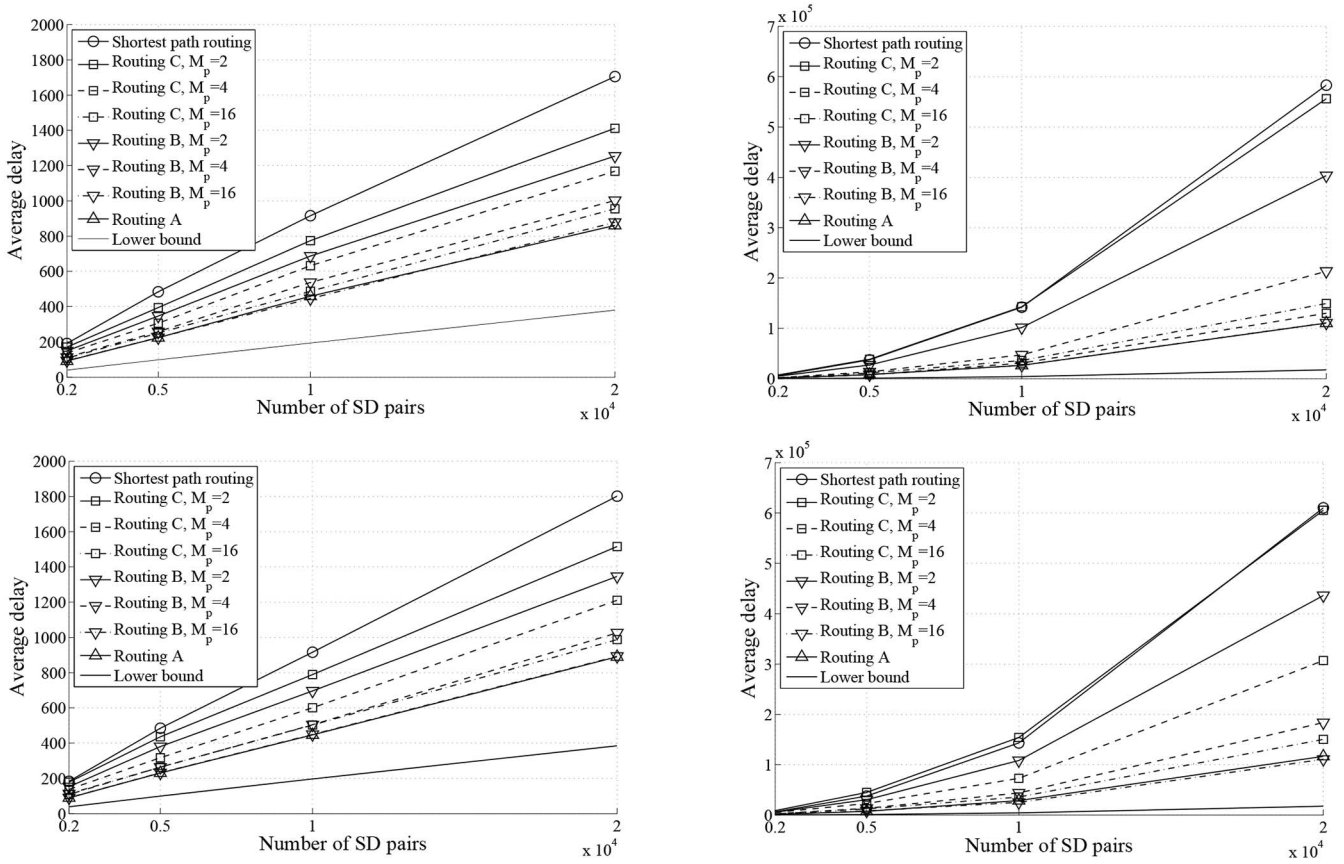


Fig. 6. Average delays for the random torus network when $N = 500$ and $d_{\max} = 0.14$, where $f_e(x) = x$ and $r_k = 1$ (top left), $f_e(x) = x^2$ and $r_k = 1$ (top right), $f_e(x) = x$ and $r_k \sim \text{Unif}(0, 2)$ (bottom left), and $f_e(x) = x^2$ and $r_k \sim \text{Unif}(0, 2)$ (bottom right) were used.

6.2 Grid Networks

We construct two different types of grid networks in the following manner.

- **Grid geometric network**: N nodes are deployed in a grid network of size $\sqrt{N} \times \sqrt{N}$. There exist links (i, j) and (j, i) if the Euclidian distance between nodes i and j is less than or equal to $d_{\max} > 0$.

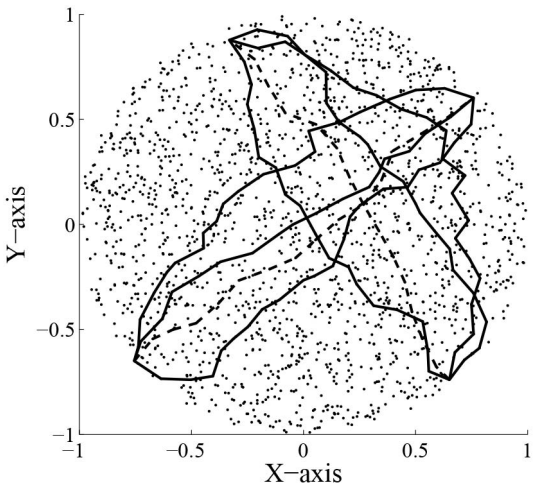


Fig. 7. Examples of trajectories of predetermined paths (solid curves) and the corresponding shortest paths (dashed curves) when $N = 2000$, $d_{\max} = 0.07$, $f_e(x) = x$, and $r_k = 1$.

- **Grid torus network**: The node deployment and link connection are the same as those assumed for grid geometric network except that the distance between nodes are measured under the grid torus network topology.

The average delay performance of grid networks is similar to that of random networks. The average delays of the two predetermined path routings, Routings B and C, quickly converges to the average delay of Routing A as the number of predetermined paths increases. Hence we can again significantly reduce the amount of required load information and computational complexity in grid networks by routing through a small number of predetermined paths. One noticeable thing is that the gap between the achievable delays of the proposed routings and its lower bound in Theorem 1 is smaller than that of random networks and even there is a case in which the lower bound becomes tight, which can be shown from the following example.

Experiment 5 (Delay of grid networks). *Fig. 9 plots the average delays for the grid geometric network and the grid torus network when the link delay functions are linear. Similarly, Fig. 10 plots the average delays when the link delay functions are quadratic. Due to the regularity of the node distribution and the fact that there only exist links between eight adjacent nodes for $d_{\max} = \sqrt{2}$, Routing A will impose similar amount of load on every link so that the gap between the achievable delay and its lower bound becomes smaller than that of the random networks. The same is true for Routings B*

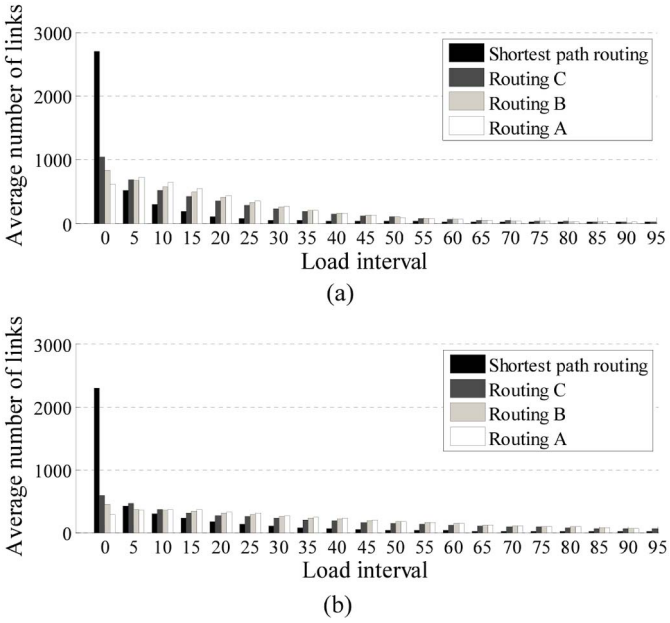


Fig. 8. Histogram on the load distribution for the random geometric network when $N = 500$, $d_{\max} = 0.14$, $M_p = 16$, $f_e(x) = x$, and $r_k = 1$, where $K = 10000$ was used for (a) and $K = 20000$ was used for (b). Here each bar at the x-axis a denotes the average number of links whose loads are in the interval $[a, a + 5)$.

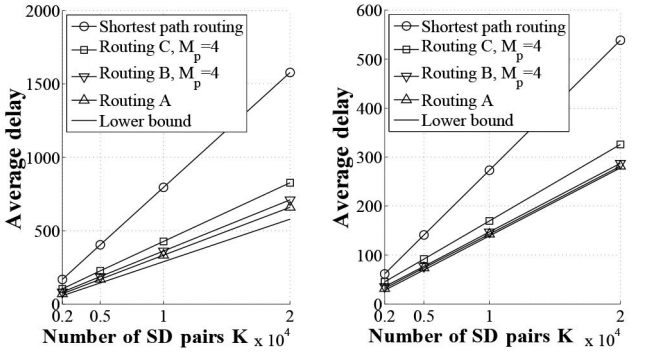


Fig. 9. Average delays for the grid geometric network (left) and for the grid torus network (right) when $N = 25 \times 25$, $d_{\max} = \sqrt{2}$, $f_e(x) = x$, and $r_k = 1$.

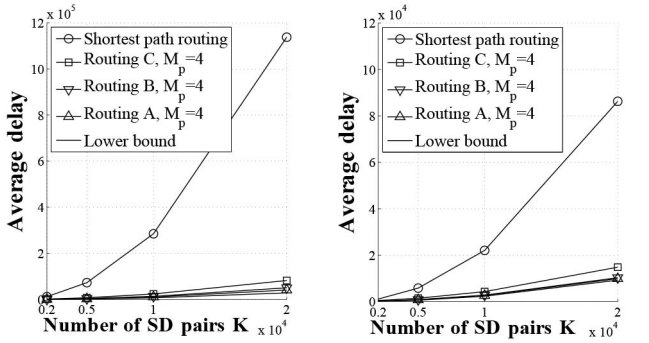


Fig. 10. Average delays for the grid geometric network (left) and for the grid torus network (right) when $N = 25 \times 25$, $d_{\max} = \sqrt{2}$, $f_e(x) = x^2$, and $r_k = 1$.

and C. Especially for the grid torus network, the gap becomes tight since there is neither the network center nor the network boundary so that every link delivers almost the same amount of load for this case.

7 Conclusion

In this paper, we studied the minimum delay routing problem for a network of direct graph having K uniformly distributed SD pairs at random. We mainly focused on the heavy traffic regime where there is a relatively large number of SD pairs compared to the network size. We first derived the lower bound on the average delay in the limit of large K and showed by simulation that it is tight for a certain class of grid networks. We then studied the minimum delay distributed routing problem in which each SD pair is able to set its routing in a distributed manner with and without partial load information. We showed that both the minimum delay routing and the minimum delay distributed routing achieve a bounded multiplicative gap whp between any link loads. We also proposed several predetermined path routing algorithms that are able to set their routing within polynomial time based on partial load information and without any load information. We demonstrated by simulation that, for a broad class of link delay functions and traffic patterns, the proposed routing algorithms efficiently reduce the average delay only with very limited load information and low computational complexity.

Acknowledgements

The research of the first author was funded in part by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2013. The research of the second author was funded in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2012R1A1A1014965). K. Jung is the corresponding author.

References

[1] Z. Sahinoglu and S. Tekinay, "On multimedia networks: Self-similar traffic and network performance," *IEEE Commun. Mag.*, vol. 37, no. 1, pp. 48–52, Jan. 1999.

[2] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the internet: Challenges and approaches," *Proc. IEEE*, vol. 88, no. 12, pp. 1855–1875, Dec. 2000.

[3] D. A. Menasce, "QoS issues in web services," *IEEE Internet Comput.*, vol. 6, no. 6, pp. 72–75, Nov./Dec. 2002.

[4] J. Chen, S.-H. G. Chan, and V. O. K. Li, "Multipath routing for video delivery over bandwidth-limited networks," *IEEE J. Select. Areas Commun.*, vol. 22, no. 10, pp. 1920–1932, Dec. 2004.

[5] W. Wang, S. C. Liew, and V. O. K. Li, "Solutions to performance problems in VoIP over a 802.11 wireless LAN," *IEEE Trans. Veh. Technol.*, vol. 54, no. 1, pp. 366–384, Jan. 2005.

[6] M. Chen, V. Leung, S. Mao, and Y. Yuan, "Directional geographical routing for real-time video communications in wireless sensor networks," *Elsevier Comput. Commun.*, vol. 30, pp. 3368–3383, Nov. 2007.

[7] R. G. Gallager, "A minimum delay routing algorithm using distributed computation," *IEEE Trans. Commun.*, vol. 25, no. 1, pp. 73–85, Jan. 1977.

[8] D. P. Bertsekas, E. M. Gafni, and R. G. Gallager, "Second derivative algorithms for minimum delay distributed routing in networks," *IEEE Trans. Commun.*, vol. 32, no. 8, pp. 911–919, Aug. 1984.

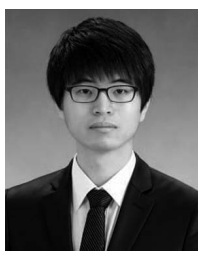
[9] P. Pham and S. Perreau, "Performance analysis of reactive shortest path and multi-path routing mechanism with load balance," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, Mar./Apr. 2003.

[10] Y. Ganjali and A. Keshavazian, "Load balancing in ad hoc networks: Single-path routing vs. multi-path routing," in *Proc. IEEE INFOCOM*, Hong Kong, China, Mar. 2004.

[11] L. Popa, A. Rostamizadeh, and R. M. Karp, "Balancing traffic load in wireless networks with curveball routing," in *Proc. ACM MobiHoc*, Montréal, QC, Canada, Sep. 2007.
[12] J. Gao and L. Zhang, "Load balanced short path routing in wireless networks," in *Proc. IEEE INFOCOM*, Hong Kong, China, Mar. 2004.
[13] Y. Bejerano, S. Han, and A. Kumar, "Efficient load-balancing routing for wireless mesh networks," *Comput. Netw.*, vol. 51, no. 10, pp. 2450–2466, Jul. 2007.
[14] R. Banner and A. Orda, "Multipath routing algorithms for congestion minimization," *IEEE Trans. Inf. Theory*, vol. 15, no. 2, pp. 413–424, Apr. 2007.
[15] S.-J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *Proc. IEEE ICC*, Helsinki, Finland, Jun. 2001.
[16] S.-J. Lee and M. Gerla, "Dynamic load-aware routing in ad hoc networks," in *Proc. IEEE ICC*, Helsinki, Finland, Jun. 2001.
[17] R. Laufer, H. Dubois-Ferri'ere, and L. Kleinrock, "Multirate anypath routing in wireless mesh networks," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009.
[18] K. Jung and D. Shah, "Low delay scheduling in wireless network," in *Proc. IEEE ISIT*, Nice, France, Jun. 2007.
[19] O. Souihli, M. Frikha, and M. B. Hamouda, "Load-balancing in MANET shortest-path routing protocols," *Ad Hoc Netw.*, vol. 7, pp. 431–442, Mar. 2009.
[20] T. Roughgarden and É. Tardos, "How bad is selfish routing?" *J. ACM*, vol. 49, pp. 236–259, Mar. 2002.
[21] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao, "Cooperation in wireless ad hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, Apr. 2003.
[22] J. Correa, A. Schulz, and N. Stier-Moses, "Selfish routing in capacitated networks," *Math. Oper. Res.*, vol. 29, pp. 961–976, Nov. 2004.
[23] R. Cole, Y. Dodis, and T. Roughgarden, "How much can taxes help selfish routing?" *J. Comput. Syst. Sci.*, vol. 72, pp. 444–467, May 2005.
[24] S. Suri, C. Tóth, and Y. Zhou, "Selfish load balancing and atomic congestion games," *Algorithmica*, vol. 29, pp. 79–96, Feb. 2007.
[25] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proc. IEEE INFOCOM*, Kobe, Japan, Apr. 1997.
[26] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. IEEE WMCSA*, New Orleans, LA, USA, Feb. 1999.
[27] M. R. Pearlman, Z. J. Haas, P. Sholander, and S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," in *Proc. ACM MobiHoc*, Boston, MA, USA, Aug. 2000.
[28] R. Jain, A. Puri, and R. Sengupta, "Geographical routing using partial information for wireless ad hoc networks," *IEEE Personal Commun.*, vol. 8, no. 1, pp. 48–57, Feb. 2001.
[29] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks," in *Ad Hoc Networking*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2001, pp. 139–172.
[30] S. Subramanian and S. Shakkottai, "Geographic routing with limited information in sensor networks," in *Proc. IEEE ISPN*, Los Angeles, CA, USA, Apr. 2005.
[31] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proc. ACM MobiCom*, San Diego, CA, USA, Sep. 2003.
[32] A. Kvalbein, C. Dovrolis, and C. Muthu, "Multipath load-adaptive routing: Putting the emphasis on robustness and simplicity," in *Proc. IEEE Int. Con. Netw. Protocols*, Princeton, NJ, USA, Oct. 2009.
[33] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," in *Proc. IEEE Int. Conf. Netw. Protocols*, Riverside, CA, USA, Nov. 2001.
[34] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi, "A framework for reliable routing in mobile ad hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, Apr. 2003.
[35] A. Orda, R. Rom, and N. Shimkin, "Competitive routing in multiuser communication networks," *IEEE/ACM Trans. Netw.*, vol. 1, no. 5, pp. 510–521, Oct. 1993.
[36] L. R. Ford, Jr., and D. R. Fulkerson, *Flows in Networks*. Princeton, NJ, USA: Princeton University Press, 1962.
[37] L. G. Valiant, "A scheme for fast parallel communication," *SIAM J. Comput.*, vol. 11, no. 2, pp. 350–361, 1981.
[38] B. Towles and W. J. Dally, "Worst-case traffic for oblivious routing functions," in *Proc. 14th Annu. ACM SPAA*, Manitoba, MB, Canada, Aug. 2002.
[39] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Räcke, "Optimal oblivious routing in polynomial time," in *Proc. 35th Annu. ACM STOC*, San Diego, CA, USA, Jun. 2003.
[40] B. M. Maggs, F. M. Auf Der Heide, B. Vocking, and M. Westermann, "Exploiting locality for data management in systems of limited bandwidth," in *Proc. 38th Annu. IEEE Symp. FOCS*, Miami, FL, USA, Oct. 1997.
[41] C. Harrelson, K. Hildrum, and S. Rao, "A polynomial-time tree decomposition to minimize congestion," in *Proc. 15th Annu. ACM Symp. SPAA*, Barcelona, Spain, Jun. 2003.
[42] C. Busch, M. Magdon-Ismail, and J. Xi, "Oblivious routing on geometric networks," in *Proc. 17th Annu. SPAA*, Las Vegas, NV, USA, Jul. 2005.
[43] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.
[44] M. Franceschetti, O. Dousse, D. Tse, and P. Thiran, "Closing the gap in the capacity of wireless networks via percolation theory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1009–1018, Mar. 2007.
[45] A. Özgür, O. Lévêque, and D. Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3549–3572, Oct. 2007.
[46] R. Meester and R. Roy, *Continuum Percolation*. Cambridge, U.K.: Cambridge University Press, 1996.
[47] M. Franceschetti and R. Meester, *Random Networks for Communication*. Cambridge, U.K.: Cambridge University Press, 2007.

Sang-Woon Jeon received the B.S. and M.S. degrees in Electrical Engineering from Yonsei University, Seoul, Korea in 2003 and 2006, respectively, and the Ph.D. degree in Electrical Engineering from KAIST, Deajeon, Korea in 2011. He is an assistant professor in the Department of Information and Communication Engineering at Andong National University since 2013. From 2011 to 2013, he was a Post-Doctoral Associate in the School of Computer and Communication Sciences at Ecole Polytechnique Federale de Lausanne (EPFL), Lausanne, Switzerland. His current research interests include network information theory and its application to wireless communications. Dr. Jeon won the Best Thesis Award from the EE Department at KAIST in 2012.

Kyomin Jung received the B.Sc. degree at Seoul National Univ. and the Ph.D. degree at MIT in 2003 and 2009, respectively. He is an Assistant Professor at Seoul National University Electrical and Computer Engineering department since September 2013. He was with the KAIST Computer Science department since June 2009. During his Ph.D., he was with Microsoft Research Cambridge in 2008, IBM T.J. Watson Research in 2007, and Bell Labs in 2006 as research internships. His current research interests include information network analysis, and machine learning. He was a gold medalist at the IMO (International Mathematical Olympiad) 1995, and a recipient of the excellent new faculty funding from NRF Korea in 2012.



Hyunseok Chang received the B.S. and M.S. degrees in electrical engineering from KAIST, Daejeon, Korea, in 2009 and 2011, respectively. His current research interests are include in the areas of information theory, communications, computer networking, and their applications.