A Tale of Two Metrics in Network Delay Optimization

Qingyu Liu, *Member, IEEE*, Lei Deng, *Member, IEEE*, Haibo Zeng, and Minghua Chen, *Senior Member, IEEE*

***Abstract*— We consider a single-unicast networking scenario where a sender streams a flow at a fixed rate to a receiver across a multi-hop network, possibly using multiple paths. Transmission over a link incurs a traffic-dependent link delay. We optimize network delay concerning two popular metrics, namely maximum delay and average delay. Well-known pessimistic results state that a flow cannot simultaneously achieve a maximum delay and an average delay both within bounded-ratio gaps to optimal. Instead, we pose an optimistic note on the fundamental compatibility of the two delay metrics. Specifically, we design two polynomial-time solutions each of which can deliver $(1-\epsilon)$-fraction of the flow with maximum delay and average delay simultaneously within $(1/\epsilon)$-ratio gap to optimal, for any $\epsilon \in (0,1)$. We prove that the ratio $(1/\epsilon)$ is at least near-tight. Moreover, our solutions can be extended to the multiple-unicast setting. In this setting, the two delay metrics of our solutions are both within a bounded-ratio gap of $(R/(R_{\min} \cdot \epsilon))$ to optimal, where R (resp. $R_{\min}$) is the aggregate (resp. minimum) flow rate requirement of all sender-receiver pairs. Hence we pose a similar optimistic note. Simulations based on real-world continent-scale network topology show that the empirical delay gaps observed under practical settings can be much smaller than their theoretical counterparts. In addition, our solutions can achieve over 10% reduction on the maximum delay and average delay simultaneously, only in the cost of losing 3% traffic, as compared to a conceivable delay-aware baseline without traffic loss. Our results can be of particular interest to delay-centric networking applications that can tolerate a small fraction of traffic loss, including cloud video conferencing that recently attracts substantial attention.**

***Index Terms*— Network delay optimization, maximum end-to-end delay, average end-to-end delay, Nash equilibrium.**

Manuscript received March 24, 2018; revised February 24, 2019 and September 10, 2019; accepted March 1, 2020; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor M. Andrews. Date of publication April 17, 2020; date of current version June 18, 2020. The work of Lei Deng was supported in part by the NSFC under Grant 61902256. The work of Minghua Chen was supported in part by the Start-Up Grant of City University of Hong Kong under Grant 9380118. A preliminary version of the work has been presented at the IEEE International Conference on Computer Communications, 2018. *(Corresponding authors: Qingyu Liu; Haibo Zeng; Minghua Chen.)*

Qingyu Liu and Haibo Zeng are with the Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24060 USA (e-mail: qyliu14@vt.edu; hbzeng@vt.edu).

Lei Deng is with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China, also with the Shenzhen Key Laboratory of Digital Creative Technology, Shenzhen 518060, China, and also with the Guangdong Province Engineering Laboratory for Digital Creative Technology, Shenzhen 518060, China (e-mail: ldeng@szu.edu.cn).

Minghua Chen was with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. He is now with the School of Data Science, City University of Hong Kong, Hong Kong (e-mail: minghua.chen@cityu.edu.hk).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TNET.2020.2983867

I. INTRODUCTION

A. Background and Motivation

WE CONSIDER a single-unicast network communication scenario where a sender streams a flow at a fixed rate to a receiver across a multi-hop network, possibly using multiple paths. Transmission over a link incurs a delay modeled as a *non-negative, non-decreasing, and differentiable* function of the link aggregate transmission rate. We study three fundamental network delay optimization problems, concerning two popular metrics, namely maximum delay and average delay. The three problems are (i) minimizing the maximum delay, denoted as MM, (ii) minimizing the average delay, denoted as SO, and (iii) finding the Nash equilibrium,[1] denoted as NE. Our focus is on understanding the fundamentals of the two delay metrics in terms of the three problems. In particular, we try to find a solution to each of the problems to obtain both a close-to-optimal maximum delay and a close-to-optimal average delay.

Our study is motivated by recent skyrocketing interests on supporting delay-centric traffic in data center networks [7], [8] and real-time communications such as cloud video conferencing [9], [10]. For example, it is reported that 51 million users per month attend WebEx meetings [11], 3 billion minutes of calls per day use Skype [12], and 75% of high-growth innovators use video collaboration [13]. As recommended by International Telecommunication Union (ITU) [14], for highly interactive tasks including video conferencing, it is desirable to keep the cross-network one-way delay as low as possible. A delay less than 150ms can provide a transparent interactivity while delays above 400ms are unacceptable [14].

End-to-end networking delay is mainly composed of processing delay, queuing delay, and propagation delay, all of which can be modeled as functions of flow rate in general. For example in many cases, queuing delay is the dominant factor, which can be estimated by the following function (for M/M/1 queue together with a FIFO server) [15]

$$\mathcal{D}(x) = \begin{cases} \dfrac{1}{c-x} & \text{if } c > x, \\ +\infty & \text{otherwise,} \end{cases} \tag{1}$$

where c is the capacity and x is the assigned traffic. The function in (1) does not allow the overhead case where

[1]Similar to earlier works in the area, e.g., [2]–[6], we consider a network game of a sender, a receiver, and infinitely many players. Each player chooses a fastest path under prevailing conditions to route its data from the sender to the receiver. The aggregate amount of data over players is equal to the flow rate requirement of this sender-receiver pair.

TABLE I

EXISTING RESULTS OF THE AVERAGE DELAY GAP (RESP. MAXIMUM DELAY GAP) OF SO, NE, AND MM, UNDER THE SINGLE-UNICAST SETTING. HERE THE GAP IS THE WORST-POSSIBLE RATIO COMPARING ACHIEVED DELAY WITH OPTIMAL. WE DENOTE $f_{\mathsf{SO}}(R)$ AS THE OPTIMAL SOLUTION TO SO, $f_{\mathsf{MM}}(R)$ AS THE OPTIMAL SOLUTION TO MM, AND $f_{\mathsf{NE}}(R)$ AS THE NASH EQUILIBRIUM, ALL SUBJECT TO A FLOW RATE OF R

	Average delay gap compared to optimal	Maximum delay gap compared to optimal
$f_{\mathsf{SO}}(R)$	1	$\gamma(\mathcal{L})$ [2], [3]
$f_{\mathsf{NE}}(R)$	$\sigma(\mathcal{L})$ [2], [3]; 1.33* [5]; 1.63*, 1.90* [6]	$\sigma(\mathcal{L})$ [2], [3]; $(\lvert V\rvert - 1)$ [4]; 1** [3], [18]
$f_{\mathsf{MM}}(R)$	$\sigma(\mathcal{L})$ [2], [3]	1

Note. Unless specified, gaps hold for general network topologies and arbitrary non-negative, non-decreasing, and differentiable delay functions.
*1.33 holds for linear delay functions, 1.63 for quadratic delay functions, and 1.90 for cubic delay functions, all independent of network topologies.
** This gap holds for a special network topology that has two nodes and multiple parallel links, further assuming linear delay functions.

assigned rate exceeds the capacity. Better but more complex delay formulas have been proposed to account for the overhead case and propagation delay [16]. In this paper, we do not restrict ourselves to specific link delay functions. Instead, we focus on arbitrary link delay functions that are assumed to be non-negative, non-decreasing, and differentiable, similar to the link delay models of many earlier works, e.g., [2]–[4], [17]. Thus our results can be applied widely, e.g., for problems with the delay modeled by the function in (1) or those in [16], because those link delay models satisfy our assumptions.

B. Network Delay Optimization

The two delay metrics are optimized by the three network delay optimization problems in different ways and are of strong practical relevance in delay-centric networking applications. For example in cloud video conferencing, the optimal solution to MM provides a good routing solution for streaming conferencing traffic of multiple sessions since maximum end-to-end delay is minimized. Hence all sessions can experience a delay within an optimal upper bound. SO minimizes average end-to-end delay and gives an optimal solution to efficiently utilize network resources from the view of the cloud. NE provides a routing solution to which delay-aware distributed rate control protocols converge without centrally administered servers. It is the best fair solution where conferencing sessions with the same sender and receiver experience the same delay.

For delay-centric applications, an ideal traffic routing approach should optimize both the maximum delay and the average delay, benefiting each user as well as the overall network simultaneously. In particular, (i) to ensure that each user has a satisfactory experience, maximum end-to-end delay shall be minimized or at least bounded above by the tolerant value (e.g., 400ms for video conferencing [14]). Also, (ii) it is desirable to provide a close-to-optimal average end-to-end delay, to ensure an efficient utilization of network resources.

Along this line, we summarize existing results on optimizing the two delay metrics in Table I. Details of existing studies refer to Section II. Briefly speaking, we observe that Correa *et al.* [2], [3] give the state-of-the-art study. For general network topologies and arbitrary link delay functions, they prove (i) the average delay of the optimal solution to MM, the average delay of the optimal solution to NE, and the maximum delay of the optimal solution to NE, are all bounded above by a ratio of $\sigma(\mathcal{L})$ compared to optimal, where $\sigma(\mathcal{L})$ is defined as

$$\sigma(\mathcal{L}) = \left(1 - \sup_{\mathcal{D}(\cdot)\in\mathcal{L}, 0\le x\le R} \left\{ \frac{x\cdot(\mathcal{D}(R)-\mathcal{D}(x))}{R\cdot\mathcal{D}(R)} \right\} \right), \quad (2)$$

where $\mathcal{L}$ is the set of link delay functions and R is the flow rate requirement. (ii) They also prove that the maximum delay of the optimal solution to SO is bounded above by a ratio of $\gamma(\mathcal{L})$ compared to optimal. Here $\gamma(\mathcal{L})$ is defined to be the smallest value meeting the following inequalities

$$\mathcal{D}(x) + x\cdot\mathcal{D}'(x) \le \gamma(\mathcal{L})\cdot\mathcal{D}(x), \quad \forall\mathcal{D}(\cdot)\in\mathcal{L}, \quad \forall x\in[0,R]. \quad (3)$$

Moreover, (iii) they prove that all of their proposed delay gaps are tight in the single-unicast setting. Because their problem-dependent gaps can be arbitrarily large for certain delay functions (e.g., for the queuing delay function in (1) or p-order polynomial function where p is arbitrarily large), and are tight [2], [3], we observe that no optimal solutions to MM, SO, or NE can simultaneously minimize the maximum delay and the average delay both within bounded-ratio gaps to optimal, for general network topologies and arbitrary link delay functions under the single-unicast setting.

C. Our Contributions

In this paper, we pose an optimistic note on the fundamental compatibility of the two delay metrics. Specifically, we propose a new approach to *minimize the maximum delay and the average delay, both within bounded-ratio gaps to optimal, at a cost of sacrificing the flow rate within a controllable level*. For easier reference, we use an (α,β) ($\alpha\in(0,1)$) bi-criteria average delay gap (resp. bi-criteria maximum delay gap) to refer to that the average delay (resp. maximum delay) of the solution must be within β times optimal, at a cost of only supporting α-fraction of the flow rate. Under the single-unicast setting, our results are summarized in Table II, where our bi-criteria delay gaps are constants independent to network topology and link delay function, and are at least near-tight.

Our approach is to either sacrifice a controllable portion of rates from the flow solutions (Theorems 3 and 4 for SF, i.e., Sacrificing Flow from SO), or directly solve the flow problems subject to a controllably smaller rate requirement (Theorems 5 and 6 for NE, Theorems 7 and 8 for MM). In this way, for each of the three problems including SO, NE, and MM under the single-unicast setting, we can obtain a solution that must achieve a maximum delay and an average delay both within a bi-criteria constant-ratio gap of $(1-\epsilon, 1/\epsilon)$ to optimal.

We further show that under the multiple-unicast setting, our approach also provides upbeat results, as it can still obtain a maximum delay and an average delay both within a bi-criteria bounded-ratio gap of $(1-\epsilon, R/(R_{\min}\cdot\epsilon))$ to optimal

TABLE II

BI-CRITERIA DELAY GAPS OF OUR PROPOSED SOLUTIONS UNDER THE SINGLE-UNICAST SCENARIO. A LOWER BOUND (α', β') OF THE BI-CRITERIA AVERAGE DELAY GAP (RESP. BI-CRITERIA MAXIMUM DELAY GAP) OF A FLOW SOLUTION DENOTES THAT THERE MUST EXIST AN INSTANCE WHERE THIS SOLUTION CAN SUPPORT α'-FRACTION OF THE FLOW RATE REQUIREMENT, AND ITS AVERAGE DELAY (RESP. MAXIMUM DELAY) MUST BE NO SMALLER THAN β' TIMES OPTIMAL. OUR SOLUTIONS $f_{\mathsf{SF}}[(1-\epsilon)R]$, $f_{\mathsf{NE}}[(1-\epsilon)R]$, AND $f_{\mathsf{MM}}[(1-\epsilon)R]$ ARE DEFINED IN SECTION III-D

	Average delay gap compared to optimal			Maximum delay gap compared to optimal		
	Bi-criteria gap	A lower bound of the gap	Proof	Bi-criteria gap	A lower bound of the gap	Proof
$f_{\mathsf{SF}}[(1-\epsilon)R]$	$(1-\epsilon, 1)$	$(1-\epsilon, 1)$	Theorem 3	$(1-\epsilon, 1/\epsilon)$	$(1-\epsilon, \lceil 1/\epsilon \rceil - 1)$	Theorem 4
$f_{\mathsf{NE}}[(1-\epsilon)R]$	$(1-\epsilon, 1/\epsilon)$	$(1-\epsilon, 1/\epsilon)$	Theorem 5	$(1-\epsilon, 1/\epsilon)$	$(1-\epsilon, \lceil 1/\epsilon \rceil - 1)$	Theorem 6
$f_{\mathsf{MM}}[(1-\epsilon)R]$	$(1-\epsilon, 1/\epsilon)$	$(1-\epsilon, 1/\epsilon)$	Theorem 7	$(1-\epsilon, 1)$	$(1-\epsilon, 1)$	Theorem 8

Note. ϵ can be any user-defined value that is greater than 0 and less than 1.

(refer to Table IV). Here R (resp. $R_{\min}$) is the aggregate (resp. minimum) flow rate requirement of all individual unicasts.

We highlight that SF, which is our rate-sacrificing approach from SO, is the first of its kind. It further suggests a new avenue for solving other delay-aware network optimization problems with theoretical performance guarantee. We believe that these results are of particular interest to delay-centric networking applications that can tolerate a small fraction of traffic loss, e.g., cloud video conferencing [19].

Simulations based on real-world continent-scale network topology verify our theoretical findings. We also observe that our solutions can achieve over 10% reduction on both delay metrics, only at a cost of losing 3% traffic, compared to a conceivable delay-aware baseline without traffic loss.

II. RELATED WORK

A. Studies on NE

There are lots of studies characterizing the average delay gap and the maximum delay gap of solutions to NE, as compared to optimal. To be succinct, under the single-unicast setting, representative average delay gaps of NE include $\sigma(\mathcal{L})$ [2], [3]; and representative maximum delay gaps of NE include $\sigma(\mathcal{L})$ [2], [3] and $(|V|-1)$ [4].

To be specific, many studies on NE exist to characterize the gap of the average delay of a Nash equilibrium to that of an optimal solution. First, for general network topologies but special link delay functions, this average delay gap is bounded by constant ratios. It is 1.33 for linear delay functions [5], 1.63 for quadratic delay functions [6], and 1.90 for cubic delay functions [6]. With linear delay functions, Christodoulou *et al.* [20] generalize the result of [5] that holds for a Nash equilibrium, and show that the gap is $4 \cdot (1+\epsilon)/(3-\epsilon)$ for any $0 \le \epsilon \le 1$, and is $(1+\epsilon)^2$ for any $\epsilon > 1$, in terms of an $(1+\epsilon)$-approximate Nash equilibrium. Next, for general network topologies and arbitrary link delay functions, this average delay gap is proven to be bounded by a problem-dependent ratio of $\sigma(\mathcal{L})$ [2], [3] that is defined in Equation (2). Basu *et al.* [17] generalize the result of [2], [3] and give the average delay gap of an $(1+\epsilon)$-approximate Nash equilibrium.

There also exist studies on NE to characterize the gap of the maximum delay of a Nash equilibrium to that of an optimal solution. To our best knowledge, it is first introduced by [21], followed by many studies, e.g. [18], [22], all for a specific network consisting of parallel links from a sender to a receiver, with linear delay functions. They show that the gap is minor under certain conditions, e.g., it is $\Omega(\log|E|/\log\log|E|)$ if links are identical [21], where $|E|$ is the number of links. Czumaj and Vöcking [18] also show that an equilibrium exists to minimize the maximum delay, i.e., the gap is one, if there are infinitely many players. Note that our study differ from those of [18], [21], [22] in two aspects: (i) for the network game model, we assume infinitely many players, while existing studies [18], [21], [22] can work with a finite number of players; (ii) for the network communication model, existing studies [18], [21], [22] focus on a specific two-node network with parallel links assuming linear link delay functions, while we consider general network topologies and arbitrary link delay functions. Under the network game model of infinitely many players for general network topologies and arbitrary link delay functions, Weitz [23] observes that any average delay gap of NE is also a maximum delay gap of NE in the single-unicast setting. Roughgarden [4] proves $(|V|-1)$ to be the maximum delay gap of NE, where $|V|$ is the number of nodes.

Note that NE can be formulated as a convex program and thus solved in polynomial time under our system model [3]. Hence the $(1+\epsilon)$-approximate Nash equilibrium is also polynomial-time achievable. However, considering the problem of finding a $(1+\epsilon)$-approximate Nash equilibrium with the minimal average delay, i.e., figuring out a $(1+\epsilon)$-approximate Nash equilibrium with its average delay gap to be 1, Basu *et al.* [17] prove that it can be solved in polynomial time if ϵ is large, but it is NP-hard if ϵ is small.

B. Studies on SO

Few study exists to characterize the maximum delay gap comparing the optimal solution to SO with the optimal, except for [2], [3], to our best knowledge. Correa *et al.* [2], [3] prove that the maximum delay of the optimal solution to SO is bounded by a ratio of $\gamma(\mathcal{L})$ as compared to the minimal, where $\gamma(\mathcal{L})$ is defined by the inequalities in (3). Note that SO can be formulated as a convex program, if link delay functions are convex [3].

C. Studies on MM

Almost all existing studies on MM focus on developing efficient algorithms. For example, Devetak *et al.* [24] design heuristic approaches yet without performance guarantee. For a special network where the link delay is a constant within a capacity, and is unbounded otherwise, Misra *et al.* [25] develop a $(1+\epsilon)$-approximation algorithm, and Zhang *et al.* [26]

develop a $(1+\epsilon)$-approximation algorithm further with reliability and differential delay constraints involved. To our best knowledge, studies [2], [3] are the only ones which (i) design approximation algorithms for MM considering arbitrary link delay functions, and (ii) characterize the average delay gap of the optimal solution to MM as compared to the minimal average delay. They prove that this average delay gap is bounded by $\sigma(\mathcal{L})$ which is defined in Equation (2).

It is uniquely challenging to solve MM, as Correa *et al.* [3] prove that (i) MM is NP-hard even when all link delay functions are linear; moreover, (ii) MM is APX-hard for arbitrary link delay functions, implying that there does not exist a polynomial-time algorithm which can obtain $(1+\epsilon)$-approximate solutions for any $\epsilon > 0$, unless P = NP.

Overall, for general network topologies and arbitrary link delay functions under the single-unicast scenario, solving NE, SO, and MM all can minimize the two delay metrics approximately, with problem-dependent approximation ratios ($\sigma(\mathcal{L})$ or $\gamma(\mathcal{L})$). However, as those ratios can be arbitrarily large for certain delay functions (e.g., for the queuing delay function in (1)), and are tight [2], [3], existing studies suggest that no optimal solutions to problems NE, SO, or MM can achieve a maximum delay and an average delay both within bounded-ratio gaps to optimal. Instead, in this paper we develop multiple solutions each of which can obtain a maximum delay and an average delay both within bi-criteria constant-ratio gaps to optimal.

III. System Model and Problem Formulation

A. Two Delay Metrics

We model the input network as a directed graph $G \triangleq (V, E)$ with $|V|$ nodes and $|E|$ links. Data transmission over a link $e \in E$ incurs a delay modeled as a function $\mathcal{D}_e(x_e)$ that is non-negative, non-decreasing, and differentiable with the aggregate link rate x_e. We consider a single-unicast networking scenario where a sender $s \in V$ streams a flow at a rate of $R > 0$ to a receiver $t \in V\backslash\{s\}$, possibly using multiple paths.

We define P as the set of all simple paths from s to t. A flow solution f is defined as the assigned rates over P, i.e., $f \triangleq \{x^p : x^p \geq 0, p \in P\}$, where x^p is the rate on path p. The total rate sent from s to t under flow f is defined as

$$|f| \triangleq \sum_{p\in P} x^p. \tag{4}$$

The aggregate assigned rate on link e under flow f is

$$x_e \triangleq \sum_{p\in P: e\in p} x^p, \tag{5}$$

and the delay of the path p under flow f is

$$d^p(f) \triangleq \sum_{e\in p} \mathcal{D}_e(x_e), \tag{6}$$

which is the sum of delays for all links belonging to p.

The *maximum delay* of flow f is the maximum path delay among all paths that have a positive rate,[2] defined as

$$\mathcal{M}(f) \triangleq \max_{p\in P: x^p>0} d^p(f). \tag{7}$$

[2]If a path has a positive rate, we call it *a flow-carrying path*.

The *total delay* of flow f is the sum of delays experienced by all flow units, defined as

$$\mathcal{T}(f) \triangleq \sum_{p\in P} d^p(f)\cdot x^p = \sum_{e\in E} \mathcal{D}_e(x_e)\cdot x_e. \tag{8}$$

The *average delay* of flow f is the ratio between the total delay and the total flow rate, defined as

$$\mathcal{A}(f) \triangleq \mathcal{T}(f)/|f|. \tag{9}$$

B. Three Network Delay Optimization Problems

MM: the Maximum delay Minimization problem is to minimize maximum delay subject to a flow rate requirement,

$$\mathsf{MM}: \quad \min_f \mathcal{M}(f) \quad \text{s.t. } |f| = R.$$

We define $f_{\mathsf{MM}}(R)$ as an arbitrary optimal solution to MM under rate requirement R. For convenience, $f_{\mathsf{MM}}(R)$ is called a *min-max flow*. We define $\mathcal{M}^*(R)$ as the optimal maximum delay that can be achieved for any feasible flow supporting a flow rate of R. Clearly, it is $\mathcal{M}^*(R) = \mathcal{M}(f_{\mathsf{MM}}(R))$.

SO: the System Optimization (average delay minimization) problem is to minimize average delay subject to a flow rate requirement. It is formulated as

$$\mathsf{SO}: \quad \min_f \mathcal{A}(f) \quad \text{s.t. } |f| = R.$$

We use $f_{\mathsf{SO}}(R)$ to denote an arbitrary optimal solution to SO under rate requirement R. The minimal average delay that can be achieved for flows supporting a rate of R is denoted as $\mathcal{A}^*(R)$. Similarly, we define the minimal total delay as $\mathcal{T}^*(R)$. Clearly $\mathcal{A}^*(R) = \mathcal{A}(f_{\mathsf{SO}}(R))$, and $\mathcal{T}^*(R) = R\cdot\mathcal{A}^*(R) = \mathcal{T}(f_{\mathsf{SO}}(R))$. We call $f_{\mathsf{SO}}(R)$ a *system-optimal flow*.

NE: to find a Nash Equilibrium flow subject to a flow rate requirement. We consider a non-cooperative network game of infinitely many players. Each player controls an infinitesimal amount of flow rate and acts selfishly to route its rate from the sender to the receiver to minimize the delay experienced. Then a Nash equilibrium of this game has the following definition.

Definition 1 [2]–[6]: A flow f is a *Nash equilibrium flow* (or in short a *Nash flow*) if for any pair of paths $p_1, p_2 \in P$ with $x^{p_1} > 0$, we have $d^{p_1}(f) \leq d^{p_2}(f)$.

Problem NE can be written as

$$\mathsf{NE}: \quad \text{find a Nash flow } f \text{ such that } |f| = R.$$

We use $f_{\mathsf{NE}}(R)$ to denote an arbitrary Nash flow under rate requirement R. By the definition of a Nash flow, all flow-carrying paths share the same path delay, i.e., we have $\mathcal{M}[f_{\mathsf{NE}}(R)] = \mathcal{A}[f_{\mathsf{NE}}(R)] = \mathcal{T}[f_{\mathsf{NE}}(R)]/R$. As proved in [5, Lemma 2.6], for link delay functions that are non-negative, non-decreasing, and differentiable, there exists a Nash flow and all Nash flows have the same average/maximum delay.

C. Extension to the Multiple-Unicast Setting

Later in Section VI, we extend our study from the single-unicast scenario to the multiple-unicast scenario. In this subsection, we first generalize notations and problem definitions of Sections III-A and III-B to the multiple-unicast setting.

Under the multiple-unicast setting, we are given K sender-receiver pairs $\{(s_k,t_k) : k = 1,2,\ldots,K\}$. For each $k \in \{1,2,\ldots,K\}$, we require $s_k \in V$ to stream a flow at a rate of $R_k > 0$ to $t_k \in V\backslash\{s_k\}$, and we denote all simple paths from s_k to t_k as P_k. For easier reference, we further define

$$R_{\min} \triangleq \min_{1\le k\le K} R_k, \quad R \triangleq \sum_{k=1}^{K} R_k, \quad P \triangleq \cup_{k=1}^{K} P_k.$$

A multiple-unicast flow $f = \{f^k : k = 1,2,\ldots,K\}$ is defined as the assigned rates over P, where each f^k is a single-unicast flow defined as the assigned rates over P_k. We define x_k^e as the aggregate assigned rate of unicast k over edge e,

$$x_k^e \triangleq \sum_{p\in P_k: e\in p} x^p, \quad \forall k = 1,2,\ldots,K.$$

We define $x_e \triangleq \sum_{k=1}^{K} x_k^e$. The delay of a path $p \in P_k$ under a single-unicast flow f^k is

$$d^p\left(f^k\right) \triangleq \sum_{e\in p} \mathcal{D}_e(x_e).$$

The total rate of a single-unicast flow f^k is defined as

$$\left|f^k\right| \triangleq \sum_{p\in P_k} x^p, \quad \forall k = 1,2,\ldots,K.$$

Similar to the delay metrics defined in Equations (7), (8), and (9), we define the following delay metrics respectively for each single-unicast flow f^k

$$\mathcal{M}\left(f^k\right) \triangleq \max_{p\in P_k: x^p>0} d^p\left(f^k\right), \quad \forall k = 1,2,\ldots,K,$$
$$\mathcal{T}\left(f^k\right) \triangleq \sum_{p\in P_k} d^p\left(f^k\right)\cdot x^p = \sum_{e\in E} \mathcal{D}_e(x_e)\cdot x_k^e, \quad \forall k=1,\ldots,K,$$
$$\mathcal{A}\left(f^k\right) \triangleq \mathcal{T}\left(f^k\right) / \left|f^k\right|, \quad \forall k = 1,2,\ldots,K.$$

We consider the definitions from existing studies [2]–[4] when extending problems MM, SO, and NE to the multiple-unicast setting. First, the maximum delay, total delay, and average delay of a multiple-unicast flow $f = \{f^k : k = 1,\ldots,K\}$ are defined by [2]–[4] as follows

$$\mathcal{M}(f) \triangleq \max_{1\le k\le K} \mathcal{M}\left(f^k\right), \quad \mathcal{T}(f) \triangleq \sum_{k=1}^{K} \mathcal{T}\left(f^k\right),$$
$$\mathcal{A}(f) \triangleq \sum_{k=1}^{K} \mathcal{T}\left(f^k\right) / \sum_{k=1}^{K} \left|f^k\right|.$$

Based on the three delay metrics, studies [2]–[4] define MM, SO, and NE in the multiple-unicast scenario as follows

$$\mathsf{MM} : \min_f \mathcal{M}(f) \quad \text{s.t. } \left|f^k\right| = R_k,\ \forall k = 1,2,\ldots,K,$$
$$\mathsf{SO} : \min_f \mathcal{A}(f) \quad \text{s.t. } \left|f^k\right| = R_k,\ \forall k = 1,2,\ldots,K,$$
$$\mathsf{NE} : \text{find a Nash flow } f \quad \text{s.t. } \left|f^k\right| = R_k,\ \forall k = 1,2,\ldots,K.$$

As discussed in Section II, existing results say that under the single-unicast setting, none of the optimal solutions to the three problems of MM, SO, and NE can simultaneously achieve a maximum delay and an average delay both within bounded-ratio gaps to optimal, for general network topologies and arbitrary link delay functions. In comparison, we pose an optimistic note that the maximum delay and the average delay can be optimized to be both within bi-criteria constant-ratio gaps to optimal. Moreover, our optimistic note can be extended to the multiple-unicast setting, as we can still minimize the two delay metrics to be both within bi-criteria bounded-ratio gaps to optimal. We define our approaches under the single-unicast setting in the following Section III-D, and generalize them to the multiple-unicast setting later in Section VI-B.

D. Optimizing Delays by Sacrificing Flow Rate

We consider the possibility of sacrificing flow rate requirement to simultaneously minimize the two delay metrics. More concretely, given any $\epsilon \in (0,1)$ and a flow rate requirement R, we first are interested in the following three problems that directly solve the corresponding network delay optimization problem under a reduced rate requirement of $(1-\epsilon)R$:

- $f_{\mathsf{MM}}[(1-\epsilon)R]$: an arbitrary optimal solution to MM subject to a flow rate requirement of $(1-\epsilon)R$;
- $f_{\mathsf{SO}}[(1-\epsilon)R]$: an arbitrary optimal solution to SO subject to a flow rate requirement of $(1-\epsilon)R$;
- $f_{\mathsf{NE}}[(1-\epsilon)R]$: an arbitrary optimal solution to NE subject to a flow rate requirement of $(1-\epsilon)R$.

As shown later in Section V, we are only able to prove bi-criteria constant-ratio gaps for $f_{\mathsf{MM}}[(1-\epsilon)R]$ and $f_{\mathsf{NE}}[(1-\epsilon)R]$, but not for $f_{\mathsf{SO}}[(1-\epsilon)R]$. This motivates us to design a flow sacrificing algorithm (Algorithm 1). It is the first to get a flow solution from SO, achieving a maximum delay and an average delay both within bi-criteria constant-ratio gaps to optimal:

- $f_{\mathsf{SF}}[(1-\epsilon)R]$: the solution by Sacrificing ϵ-fraction of the Flow rate from $f_{\mathsf{SO}}(R)$ using Algorithm 1.

Note that $f_{\mathsf{SF}}[(1-\epsilon)R]$ is *not* the system-optimal flow under a rate of $(1-\epsilon)R$. We present the algorithmic details of obtaining $f_{\mathsf{SF}}[(1-\epsilon)R]$ in the following Section IV.

IV. A Flow-Sacrificing Algorithm to SO Under the Single-Unicast Setting

In this section under the single-unicast setting, we develop the first algorithm to obtain $f_{\mathsf{SF}}[(1-\epsilon)R]$ which sacrifices ϵ-fraction of the flow rate from $f_{\mathsf{SO}}(R)$. $f_{\mathsf{SF}}[(1-\epsilon)R]$ can simultaneously minimize the maximum delay and the average delay, both within bi-criteria constant-ratio gaps to optimal.

A. Obtaining $f_{\mathsf{SF}}[(1-\epsilon)R]$ in Polynomial Time

Algorithm 1 gives details of obtaining $f_{\mathsf{SF}}[(1-\epsilon)R]$. In Algorithm 1, first, we obtain a path-based system-optimal flow $f_{\mathsf{SO}}(R)$ under rate requirement R (Line 4). Then the algorithm deletes $(\epsilon\cdot R)$ rate iteratively from $f_{\mathsf{SO}}(R)$ (Lines 6–13). In each iteration, we find the slowest flow-carrying path p_l and then delete the right amount of rate from it. The iteration terminates when $(\epsilon\cdot R)$ rate is deleted in total, and the remaining flow is $f_{\mathsf{SF}}[(1-\epsilon)R]$.

With an extra assumption that all link delay functions are convex, we can get a path-based system-optimal flow in polynomial time [3]: it is known that SO can be formulated as a convex program with a polynomial size using an edge-based

Algorithm 1 Obtain $f_{\mathsf{SF}}[(1-\epsilon)R]$

1: **input**: $G=(V,E)$, R, s, t, $\epsilon \in (0,1)$
2: **output**: $f_{\mathsf{SF}}[(1-\epsilon)R]$
3: **procedure**
4: $f_{\mathsf{SO}}(R)$ = System-Optimal-Flow(G,R,s,t)
5: $x^{\text{delete}} = \epsilon \cdot R$
6: **while** $x^{\text{delete}} > 0$ **do**
7: Find the slowest flow-carrying path $p_l \in P$
8: **if** $x^{p_l} > x^{\text{delete}}$ **then**
9: $x^{p_l} = x^{p_l} - x^{\text{delete}}$
10: $x^{\text{delete}} = 0$
11: **else**
12: $x^{\text{delete}} = x^{\text{delete}} - x^{p_l}$
13: $x^{p_l} = 0$
14: $f_{\mathsf{SF}}[(1-\epsilon)R]$ is defined by the remaining flow
15: **return** $f_{\mathsf{SF}}[(1-\epsilon)R]$

flow formulation [27]. The optimal solution to the convex program, which is an edge-based system-optimal flow, then can be converted to a path-based flow by a polynomial-time flow decomposition [28].[3] Note that the flow decomposition outputs at most $|E|$ paths and thus $f_{\mathsf{SO}}(R)$ contains at most $|E|$ flow-carrying paths [28]. Hence, the while loop (Lines 6–13) in Algorithm 1 terminates in at most $|E|$ iterations. Overall, Algorithm 1 can output the solution $f_{\mathsf{SF}}[(1-\epsilon)R]$ in a time polynomial to the network size $|V|$ and $|E|$, and its time complexity is even independent to the input ϵ.

B. Critical Delay Properties of $f_{\mathsf{SF}}[(1-\epsilon)R]$

For the proposed $f_{\mathsf{SF}}[(1-\epsilon)R]$, we observe that it has the following critical delay properties.

Lemma 1: Following Algorithm 1, we have

$$\mathcal{A}[f_{\mathsf{SF}}((1-\epsilon)R)] \leq \mathcal{A}^*(R), \tag{10}$$

$$\mathcal{M}[f_{\mathsf{SF}}((1-\epsilon)R)] \leq \mathcal{A}^*(R)/\epsilon. \tag{11}$$

Proof: Refer to Part A of supplementary materials. ∎

Lemma 1 implies that (i) the average delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ must be within a bi-criteria gap of $(1-\epsilon, 1)$ to optimal; (ii) the maximum delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ must be within a bi-criteria gap of $(1-\epsilon, 1/\epsilon)$ to optimal, due to that $\mathcal{A}^*(R) \leq \mathcal{M}^*(R)$ holds straightforwardly. In Theorems 3 and 4 of the following section, we further prove that both bi-criteria gaps are at least near-tight. Overall, Algorithm 1 generates a desirable single-unicast routing solution for delay-centric applications, in that its maximum delay is close-to-optimal and its average delay is optimal, at a cost of losing a small portion of the traffic rate.

V. Bi-Criteria Delay Gaps Under the Single-Unicast Setting

In this section under the single-unicast setting, we study the optimality gaps of the two delay metrics for the three network delay optimization problems, when we can sacrifice ϵ-portion of flow rate given any $\epsilon \in (0,1)$. Our results are summarized in Table II. Note that all flow solutions can support at least $(1-\epsilon)$-fraction of the flow rate requirement, which is straightforward according to their definitions given in Section III-D.

A. Comparing $f_{\mathsf{SO}}[(1-\epsilon)R]$ With the Optimal

Comparing the delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ that is the system-optimal flow supporting a rate of $(1-\epsilon)\cdot R$ with the optimal delay performance subject to a full rate requirement of R, we prove that its average delay must be no worse than optimal in Theorem 1, and its maximum delay is bounded above by a problem-dependent ratio to optimal in Theorem 2.

Theorem 1: In the single-unicast setting, we have the following for $\mathcal{A}(f_{\mathsf{SO}}[(1-\epsilon)R])$

$$\mathcal{A}(f_{\mathsf{SO}}[(1-\epsilon)R]) \leq \mathcal{A}^*(R). \tag{12}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{A}(f_{\mathsf{SO}}[(1-\epsilon)R]) = \mathcal{A}^*(R). \tag{13}$$

Proof: Since $f_{\mathsf{SO}}[(1-\epsilon)R]$ is optimal to SO with a rate requirement of $(1-\epsilon)\cdot R$, and $f_{\mathsf{SF}}[(1-\epsilon)R]$ is feasible to SO with a rate requirement of $(1-\epsilon)\cdot R$, we have

$$\mathcal{A}(f_{\mathsf{SO}}[(1-\epsilon)R]) \leq \mathcal{A}(f_{\mathsf{SF}}[(1-\epsilon)R]).$$

Then based on the inequality in (10), we can prove the inequality in (12). Given any $\epsilon \in (0,1)$, consider a simple two-node one-link network, where the link delay function is a constant. It is straightforward to verify that the equality in (13) holds in this simple example. ∎

Theorem 2: In the single-unicast setting, we have the following for $\mathcal{M}(f_{\mathsf{SO}}[(1-\epsilon)R])$

$$\mathcal{M}(f_{\mathsf{SO}}[(1-\epsilon)R]) \leq \gamma(\mathcal{L}) \cdot \mathcal{M}^*(R), \tag{14}$$

where $\gamma(\mathcal{L})$ is the same as that introduced by [2], [3], and is defined by the inequalities in (3).

Proof: Due to the non-decreasing property of all link delay functions, the following holds straightforwardly

$$\mathcal{M}^*([1-\epsilon]R) \leq \mathcal{M}^*(R).$$

This observation, together with the maximum delay gap result given by [2], [3], proves the inequality in (14). ∎

The problem-dependent maximum delay gap $\gamma(\mathcal{L})$ can be fairly large or even infinite in theory. For example, considering the queuing delay function in (1), following the definition of $\gamma(\mathcal{L})$ and assuming a rate requirement of $R < c$, we have

$$\gamma \cdot (c - x) \geq c, \quad \forall\, x \in [0, R],$$

which implies an infinitely large gap γ since R can be arbitrarily close to the capacity c. Although $\gamma(\mathcal{L})$ has been proven to be tight in [3] without sacrificing flow rate, whether it is tight remains unknown in our setting where we accept the cost of losing an ϵ-portion of the flow rate requirement.

[3] Given an edge-based flow, note that different flow decomposition could output different path-based flows, all of which have the same average delay but may have different maximum delays. When obtaining $f_{\mathsf{SF}}[(1-\epsilon)R]$ using Algorithm 1, any flow decomposition approach works in Line 4, because that all of our proved delay properties for $f_{\mathsf{SF}}[(1-\epsilon)R]$ in Lemma 1 hold for any path-based system-optimal flow.

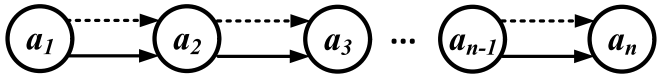


Fig. 1. A network with n nodes and $2(n-1)$ links. All upper dashed links have a constant delay of 1, while all lower solid links have a delay function of $\mathcal{D}(x)$ defined in (19). It is used to prove the near-tightness of gaps in Theorems 4 and 6.

B. Comparing $f_{\mathsf{SF}}[(1-\epsilon)R]$ With the Optimal

Drawbacks of $f_{\mathsf{SO}}[(1-\epsilon)R]$ motivate us to design a new approach (Algorithm 1) to sacrifice flow rate from the system-optimal flow, leading to the solution $f_{\mathsf{SF}}[(1-\epsilon)R]$. Comparing the delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ that supports a rate of $(1-\epsilon)\cdot R$ with the optimal delay performance subject to a full rate requirement of R, we show that its average delay is no worse than optimal in Theorem 3, and its maximum delay is within a constant-ratio gap to optimal in Theorem 4.

Theorem 3: In the single-unicast setting, we have the following for $\mathcal{A}(f_{\mathsf{SF}}[(1-\epsilon)R])$

$$\mathcal{A}(f_{\mathsf{SF}}[(1-\epsilon)R]) \leq \mathcal{A}^*(R). \tag{15}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{A}(f_{\mathsf{SF}}[(1-\epsilon)R]) = \mathcal{A}^*(R). \tag{16}$$

Proof: Inequality (15) comes from Lemma 1, and (16) holds in the same example introduced in Theorem 1. ∎

Theorem 4: In the single-unicast setting, we have the following for $\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R])$

$$\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R]) \leq (1/\epsilon)\cdot\mathcal{M}^*(R). \tag{17}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R]) \geq (\lceil 1/\epsilon\rceil - 1)\cdot\mathcal{M}^*(R). \tag{18}$$

Proof: Leveraging the inequality in (11), we have

$$\mathcal{M}[f_{\mathsf{SF}}((1-\epsilon)R)] \leq \mathcal{A}^*(R)/\epsilon \leq \mathcal{M}^*(R)/\epsilon.$$

Then we prove the inequality in (18) using Figure 1.

For any $\epsilon \in (0,1)$, it holds that $1/\epsilon > \lceil 1/\epsilon\rceil - 1$. Thus we can always find an $\alpha > 1$ such that $1/(\alpha\epsilon) = \lceil 1/\epsilon\rceil - 1$. We also denote $n = \lceil 1/\epsilon\rceil = 1/(\alpha\epsilon)+1$. Clearly, we have $n \geq 2$. We construct a network as in Figure 1, and assume $R = 1$, $s = a_1$, and $t = a_n$. All dashed links have a constant delay of 1, while all solid links have a delay function as below

$$\mathcal{D}(x) = \begin{cases} 0, & \text{if } 0 \leq x \leq \dfrac{n-2}{n-1}, \\ \left[\left(x - \dfrac{n-2}{n-1}\right) / \left(\dfrac{1-1/\alpha}{n-1}\right)\right]^2, & \text{if } x > \dfrac{n-2}{n-1}. \end{cases} \tag{19}$$

It is straightforward that $\mathcal{D}(\frac{n-2}{n-1}) = 0$, $\mathcal{D}(\frac{n-1-1/\alpha}{n-1}) = 1$, and $\mathcal{D}(R) = \mathcal{D}(1) > 1$.

We first show $\mathcal{M}^*(R) = 1$ by directly constructing a flow with a maximum delay of 1 and a flow rate of 1. We note that there are $(n-1)$ different $s-t$ paths containing exactly one dashed link. We assign $1/(n-1)$ rate on each of these $(n-1)$ paths. Hence, all solid links have a flow rate of $\frac{n-2}{n-1}$, leading to a link delay of $\mathcal{D}(\frac{n-2}{n-1}) = 0$. Therefore, all these $(n-1)$ flow-carrying paths have a path delay of 1, implying that achieved maximum delay is also 1.

We then prove that $\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R]) = n-1 = \lceil 1/\epsilon\rceil - 1$ by constructing $f_{\mathsf{SF}}[(1-\epsilon)R]$ following Algorithm 1. According to [5, Corollary 2.5], we can get $f_{\mathsf{SO}}(R)$ by obtaining $f_{\mathsf{NE}}(R)$ under new link delay functions $\hat{\mathcal{D}}_e(x_e) = \mathcal{D}_e(x_e) + x_e\cdot\mathcal{D}_e'(x_e)$. For our example in Figure 1, the new link delay function for dashed links is again the constant 1. But for solid links it is $\hat{\mathcal{D}}(x) = \mathcal{D}(x) + x\cdot\mathcal{D}'(x)$ where $\mathcal{D}(x)$ is defined in Equation (19). It is easy to verify that $\hat{\mathcal{D}}(x)$ is non-decreasing and continuous over $x \geq 0$ and strictly increasing over $x \in [\frac{n-2}{n-1}, \frac{n-1-1/\alpha}{n-1}]$. Also we have

$$\hat{\mathcal{D}}\left(\frac{n-2}{n-1}\right) = 0, \quad \hat{\mathcal{D}}\left(\frac{n-1-1/\alpha}{n-1}\right) > 1.$$

Thus it is fair to define $\lambda \in (\frac{n-2}{n-1}, \frac{n-1-1/\alpha}{n-1})$ such that $\hat{\mathcal{D}}(\lambda) = 1$. By the definition of Nash equilibrium and [5, Corollary 2.5], now we can claim that in $f_{\mathsf{SO}}(R)$ (or $f_{\mathsf{NE}}(R)$ but under new link delay functions equivalently), all solid links have a rate of λ and all dashed links have a flow rate of $(1-\lambda)$.

Then one possible path-based flow solution of $f_{\mathsf{SO}}(R)$ is: a rate of $(1-\lambda)$ is assigned to the path p_1 containing all dashed links whose delay is $n-1$, and a rate of λ is assigned to the path p_2 containing all solid links whose delay is $(n-1)\mathcal{D}(\lambda) < n-1$. Now following Algorithm 1, since p_1 is the slowest flow-carrying path and $1-\lambda > 1-\frac{n-1-1/\alpha}{n-1} = \epsilon = \epsilon\cdot R$, all $\epsilon\cdot R$ rate will be deleted from p_1. Besides, after deleting, path p_1 still contains a strictly positive rate, which implies

$$\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R]) = n-1.$$

Therefore, for this example, we have that

$$\frac{\mathcal{M}(f_{\mathsf{SF}}[(1-\epsilon)R])}{\mathcal{M}^*(R)} = n-1 = \lceil 1/\epsilon\rceil - 1,$$

which proves the inequality in (18). ∎

Different from $f_{\mathsf{SO}}[(1-\epsilon)R]$, $f_{\mathsf{SF}}[(1-\epsilon)R]$ is able to guarantee constant-ratio gaps both for the maximum delay and for the average delay as compared to their respective optimal performances. As discussed in Section IV-A, $f_{\mathsf{SO}}[(1-\epsilon)R]$, and hence $f_{\mathsf{SF}}[(1-\epsilon)R]$, can be obtained in polynomial time if link delay functions are convex. Thus for delay-sensitive applications, $f_{\mathsf{SF}}[(1-\epsilon)R]$ can be figured out quickly, and provides a solution where any user can experience a delay with known upper bound compared to optimal, and in the mean time the whole network will see an efficient utilization of resources since the average delay is also optimized.

C. Comparing $f_{\mathsf{NE}}[(1-\epsilon)R]$ With the Optimal

Compared with SO, NE has an advantage that it can be solved in a polynomial time even for non-convex link delay functions [3]. Comparing the delay of $f_{\mathsf{NE}}[(1-\epsilon)R]$ that is the Nash flow supporting a rate of $(1-\epsilon)\cdot R$ with the optimal delay performance subject to a full rate requirement of R, we observe that both its average delay and its maximum delay must be bounded by constant-ratio gaps to optimal, as introduced in our Theorems 5 and 6.

Theorem 5: In the single-unicast setting, we have the following for $\mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R])$

$$\mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R]) \leq (1/\epsilon)\cdot \mathcal{A}^*(R). \tag{20}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R]) = (1/\epsilon)\cdot \mathcal{A}^*(R). \tag{21}$$

Proof: This theorem is an easy adaptation of [5, Theorem 3.2]. Details refer to Part B of supplementary materials. ∎

Theorem 6: In the single-unicast setting, we have the following for $\mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R])$

$$\mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R]) \leq (1/\epsilon)\cdot \mathcal{M}^*(R). \tag{22}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R]) \geq (\lceil 1/\epsilon \rceil - 1)\cdot \mathcal{M}^*(R). \tag{23}$$

Proof: By leveraging Theorem 5 and the definition of a Nash flow, we can prove the inequality in (22)

$$\begin{aligned}\mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R]) &= \mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R]) \\ &\leq \frac{\mathcal{A}^*(R)}{\epsilon} \leq \frac{\mathcal{A}(f_{\mathsf{MM}}(R))}{\epsilon} \leq \frac{\mathcal{M}^*(R)}{\epsilon}.\end{aligned}$$

Inequality (23) holds for the same example of Figure 1 that is introduced in Theorem 4. The intuition is that $f_{\mathsf{NE}}[(1-\epsilon)R]$ shall route $(1-\epsilon)\cdot R$ rate to the path with all solid links. ∎

Although $f_{\mathsf{NE}}[(1-\epsilon)R]$ can obtain both a close-to-optimal maximum delay and a close-to-optimal average delay, we observe that its average delay gap is worse than that of $f_{\mathsf{SF}}[(1-\epsilon)R]$, as we compare Theorem 3 with Theorem 5. Thus, although the Nash flow is easy to be obtained even for non-convex delay functions and provides an alternate routing solution to simultaneously obtain a good maximum delay and a good average delay, its average delay performance is outperformed by that of $f_{\mathsf{SF}}[(1-\epsilon)R]$ in theory.

D. Comparing $f_{\mathsf{MM}}[(1-\epsilon)R]$ With the Optimal

Comparing the delay of $f_{\mathsf{MM}}[(1-\epsilon)R]$, the min-max flow supporting a rate of $(1-\epsilon)\cdot R$, with the optimal delay subject to a full rate requirement of R, we show that its average delay is bounded by a constant ratio to optimal in Theorem 7 and its maximum delay is no worse than optimal in Theorem 8.

Theorem 7: In the single-unicast setting, we have the following for $\mathcal{A}(f_{\mathsf{MM}}[(1-\epsilon)R])$

$$\mathcal{A}(f_{\mathsf{MM}}[(1-\epsilon)R]) \leq (1/\epsilon)\cdot \mathcal{A}^*(R). \tag{24}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{A}(f_{\mathsf{MM}}[(1-\epsilon)R]) = (1/\epsilon)\cdot \mathcal{A}^*(R). \tag{25}$$

Proof: Leveraging Theorem 5 and the definition of a Nash flow, we can prove the inequality in (24)

$$\begin{aligned}\mathcal{A}(f_{\mathsf{MM}}[(1-\epsilon)R]) &\leq \mathcal{M}(f_{\mathsf{MM}}[(1-\epsilon)R]) \\ &\leq \mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R]) \\ &= \mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R]) \leq \mathcal{A}^*(R)/\epsilon.\end{aligned}$$

We can follow a similar proof as that used to prove the equality in (21), in order to prove our equality in (25). The intuition is that $f_{\mathsf{NE}}[(1-\epsilon)R]$ in the instance used to prove the equality in (21) also minimizes the maximum delay under a rate requirement of $(1-\epsilon)\cdot R$. ∎

It is known that different path-based system-optimal flows supporting a flow rate of $(1-\epsilon)\cdot R$ may have different maximum delays. But according to Theorem 2, those maximum delays are always upper bounded by a problem-dependent ratio of $\gamma(\mathcal{L})$ compared to the optimal maximum delay subject to a full rate requirement of R. Similarly, it is known that different path-based min-max flows supporting a flow rate of $(1-\epsilon)\cdot R$ could have different average delays. But according to Theorem 7, those average delays are always upper bounded by a constant ratio of $(1/\epsilon)$ compared to the optimal average delay subject to a full rate requirement of R.

Theorem 8: In the single-unicast setting, we have the following for $\mathcal{M}(f_{\mathsf{MM}}[(1-\epsilon)R])$

$$\mathcal{M}(f_{\mathsf{MM}}[(1-\epsilon)R]) \leq \mathcal{M}^*(R). \tag{26}$$

Further, for any $\epsilon \in (0,1)$, there exists a single-unicast instance where the following holds

$$\mathcal{M}(f_{\mathsf{MM}}[(1-\epsilon)R]) = \mathcal{M}^*(R). \tag{27}$$

Proof: The inequality in (26) holds because all link delay functions are non-decreasing. The equality in (27) holds in the same example introduced in Theorem 1. ∎

For general network topologies and arbitrary link delay functions under the single-unicast setting, according to existing studies [2], [3], a flow cannot obtain a maximum delay and an average delay both within bounded-ratio gaps to optimal. In comparison, in this section we pose an optimistic note that the maximum delay optimization and the average delay optimization are "largely" compatible, because there exist many different flows, e.g., $f_{\mathsf{SF}}[(1-\epsilon)R]$, $f_{\mathsf{NE}}[(1-\epsilon)R]$, and $f_{\mathsf{MM}}[(1-\epsilon)R]$, that can obtain a maximum delay and an average delay both within a bi-criteria constant-ratio gap of $(1-\epsilon, 1/\epsilon)$ to optimal, for any user-defined $\epsilon \in (0,1)$.

VI. Extending Our Results to the Multiple-Unicast Setting

In the previous section, we design three single-unicast flow solutions, i.e., $f_{\mathsf{SF}}[(1-\epsilon)R]$, $f_{\mathsf{NE}}[(1-\epsilon)R]$, and $f_{\mathsf{MM}}[(1-\epsilon)R]$, for minimizing the two delay metrics. Now we generalize our solutions to the multiple-unicast networking scenario.

A. Existing Delay-Gap Results in the Multiple-Unicast Setting

For single-unicast networking, we have summarized existing delay gaps of MM, SO, and NE in Table I. Now we look at their delay performances in the multiple-unicast scenario, with corresponding results summarized in Table III.

A critical observation on $f_{\mathsf{NE}}(R)$ in the multiple-unicast setting is made in [3, Theorem 5.2]:

$$R_k \cdot \mathcal{A}\left(f_{\mathsf{NE}}^k(R)\right) \leq R \cdot \mathcal{A}\left(f_{\mathsf{NE}}(R)\right), \quad \forall k=1,2,\ldots,K. \quad (28)$$

Following a similar proof, we find that it is easy to generalize the inequalities in (28) from the Nash flow to an arbitrary multiple-unicast flow $f=\{f^k: k=1,2,\ldots,K\}$.

Lemma 2: In the multiple-unicast setting, for any network flow $f = \{f^k : k = 1,2,\ldots,K\}$ where $|f_k| = R_k$, $\forall k=1,2,\ldots,K$, and $|f|=R$, the following holds

$$\mathcal{A}\left(f^k\right) \leq (R/R_{\min}) \cdot \mathcal{A}(f), \quad \forall k=1,2,\ldots,K.$$

Proof: It follows a similar proof to [3, Theorem 5.2]. To be specific due to $\mathcal{T}(f^k) \leq \mathcal{T}(f)$, considering the definition of the total delay, the following holds for any $k=1,2,\ldots,K$

$$\mathcal{A}\left(f^k\right) \leq (R/R_k) \cdot \mathcal{A}(f) \leq (R/R_{\min}) \cdot \mathcal{A}(f). \quad (29)$$

■

With the inequalities in (28), Correa *et al.* [2], [3] prove that for NE, its average delay gap is $\sigma(\mathcal{L})$ and its maximum delay gap is $\sigma(\mathcal{L}) \cdot (R/R_{\min})$ in the multiple-unicast setting. Although Correa *et al.* [2], [3] do not extend their maximum delay gap of SO (resp. average delay gap of MM) from the single-unicast setting to the multiple-unicast setting, with Lemma 2 we can establish the following two theorems.

Theorem 9: In the multiple-unicast setting, comparing the average delay of $f_{\mathsf{MM}}(R)$ to optimal, we have

$$\mathcal{A}(f_{\mathsf{MM}}(R)) \leq \sigma(\mathcal{L}) \cdot (R/R_{\min}) \cdot \mathcal{A}^*(R).$$

Proof: It holds due to the following inequalities

$$\begin{aligned}
&\mathcal{A}(f_{\mathsf{MM}}(R)) \\
&\leq \mathcal{M}(f_{\mathsf{MM}}(R)) \leq \mathcal{M}(f_{\mathsf{NE}}(R)) \overset{(a)}{=} \mathcal{A}\left(f_{\mathsf{NE}}^i(R)\right) \\
&\overset{(b)}{\leq} (R/R_{\min}) \cdot \mathcal{A}(f_{\mathsf{NE}}(R)) \leq \sigma(\mathcal{L}) \cdot (R/R_{\min}) \cdot \mathcal{A}^*(R),
\end{aligned}$$

where in (a) we assume the maximum delay of $f_{\mathsf{NE}}(R)$ is achieved by the unicast i, and (b) comes from Lemma 2. ■

Theorem 10: In the multiple-unicast setting, comparing the maximum delay of $f_{\mathsf{SO}}(R)$ to optimal, we have

$$\mathcal{M}(f_{\mathsf{SO}}(R)) \leq \gamma(\mathcal{L}) \cdot (R/R_{\min}) \cdot \mathcal{M}^*(R).$$

Proof: It holds due to the following inequalities

$$\begin{aligned}
\mathcal{M}(f_{\mathsf{SO}}(R)) &\overset{(a)}{\leq} \gamma(\mathcal{L}) \cdot \mathcal{A}(f_{\mathsf{SO}}^i(R)) \overset{(b)}{\leq} \frac{\gamma(\mathcal{L}) \cdot R}{R_{\min}} \cdot \mathcal{A}(f_{\mathsf{SO}}(R)) \\
&\leq \frac{\gamma(\mathcal{L}) \cdot R}{R_{\min}} \cdot \mathcal{A}(f_{\mathsf{MM}}(R)) \leq \frac{\gamma(\mathcal{L}) \cdot R}{R_{\min}} \cdot \mathcal{M}(f_{\mathsf{MM}}(R)),
\end{aligned}$$

where (a) comes from [3, Theorem 4.2], assuming that the maximum delay of $f_{\mathsf{SO}}(R)$ is achieved by the unicast i, and (b) comes from Lemma 2. ■

Overall under the multiple-unicast scenario, solving problems MM, NE, and SO can simultaneously minimize the maximum delay and the average delay, giving approximation ratios which depend on link delay functions and flow rate requirements. However, as discussed in our Section II, those delay-function-dependent gaps can be arbitrarily large for certain delay functions, implying that none of the optimal solutions to the three problems can simultaneously optimize the two delay metrics within bounded-ratio gaps to optimal.

TABLE III

EXISTING RESULTS OF DELAY GAPS OF PROBLEMS SO, NE, AND MM, UNDER THE MULTIPLE-UNICAST SCENARIO [2], [3]

	Average delay gap compared to optimal	Maximum delay gap compared to optimal
$f_{\mathsf{SO}}(R)$	1	$\gamma(\mathcal{L}) \cdot (R/R_{\min})$
$f_{\mathsf{NE}}(R)$	$\sigma(\mathcal{L})$	$\sigma(\mathcal{L}) \cdot (R/R_{\min})$
$f_{\mathsf{MM}}(R)$	$\sigma(\mathcal{L}) \cdot (R/R_{\min})$	1

Comparing Table III with Table I, existing delay gaps under the multiple-unicast setting reduce to those under the single-unicast setting when $K=1$. Correa *et al.* [2], [3] prove that their gaps $\sigma(\mathcal{L})$ and $\gamma(\mathcal{L})$ are tight under the single-unicast setting, but do not discuss them under the multiple-unicast setting. It is unclear whether those gaps in Table III with the factor of $(R/R_{\min})$ involved are tight or not when $K \geq 2$, and it is an important future research direction.

Below we present some preliminary understandings, by proving that two conceivable methods cannot be used to construct multiple-unicast instances with tight delay gaps. Specifically, as the inequalities in (29) are critical when we characterize gaps, they should be close to equalities for instances achieving tight gaps. For the first part $\mathcal{T}(f^k) = R_k \cdot \mathcal{A}(f^k) \leq R \cdot \mathcal{A}(f) = \mathcal{T}(f)$, in order to make it close to an equality, $\mathcal{T}(f^i)$ is required to be close to 0 for any $i \neq k$. This motivates us to consider two conceivable cases. (i) In the first case, we assume that there exists an $i \neq k$ where R_i is close to 0. However, this R_i will definitely make $R_{\min}$ arbitrarily small. Therefore, the second part $(R/R_k) \cdot \mathcal{A}(f) \leq (R/R_{\min}) \cdot \mathcal{A}(f)$ of the inequalities in (29) will be far from an equality, implying that instances achieving tight gaps do not exist in the first case. (ii) In the second case, we assume that $\mathcal{A}(f^i)$ is close to 0 for each $i \neq k$. We consider a special instance of this case, where the network G is composed of K disjoint sub-networks $\{G_j, j=1,\ldots,K\}$. For each $i \neq k$, we assume G_i is a two-node one-link graph and the link delay function is a constant 0. It is clear that for this instance, the multiple-unicast problem SO (resp. NE, MM) in G is equivalent to its single-unicast counterpart in G_k. By the single-unicast tightness results from [2], [3], the largest maximum delay gap of $f_{\mathsf{SO}}(R)$ in this multiple-unicast instance is $\gamma(\mathcal{L})$. Again, this gap is much smaller than the proposed gap $\gamma(\mathcal{L}) \cdot (R/R_{\min})$. Similarly, it is easy to verify that the delay gaps of $f_{\mathsf{NE}}(R)$ and $f_{\mathsf{MM}}(R)$ are much smaller than the proposed ones in Table III for this instance. However, there might be other instances in the second case that have delay gaps close to the proposed ones in Table III, and we plan to investigate it in our future study.

B. Extending Our Results to the Multiple-Unicast Scenario

Similar to the respective definitions introduced in Section III-D, now we extend our solutions $f_{\mathsf{MM}}[(1-\epsilon)R]$ and $f_{\mathsf{NE}}[(1-\epsilon)R]$ to the multiple-unicast setting as follows

- $f_{\mathsf{MM}}[(1-\epsilon)R]$: an arbitrary optimal solution to MM, where the unicast session k is subject to a rate requirement of $(1-\epsilon)\cdot R_k$, for each $k=1,2,\ldots,K$;
- $f_{\mathsf{NE}}[(1-\epsilon)R]$: an arbitrary optimal solution to NE, where the unicast session k is subject to a rate requirement of $(1-\epsilon)\cdot R_k$, for each $k=1,2,\ldots,K$.

Based on their respective definitions, it is clear that we have

$$\left|f^k_{\mathsf{MM}}[(1-\epsilon)R]\right| = (1-\epsilon)\cdot R_k, \quad \forall k=1,2,\ldots,K,$$
$$\left|f^k_{\mathsf{NE}}[(1-\epsilon)R]\right| = (1-\epsilon)\cdot R_k, \quad \forall k=1,2,\ldots,K.$$

For the solution $f_{\mathsf{SF}}[(1-\epsilon)R]$, we have two different definitions in the multiple-unicast setting

- $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$: similar to Algorithm 1, it is a solution returned after we iteratively delete a total of $(1-\epsilon)\cdot R$ rate from the slowest flow-carry paths of $f_{\mathsf{SO}}(R)$;
- $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$: similar to Algorithm 1, it is a solution returned after we iteratively delete a total of $(1-\epsilon)\cdot R_k$ rate from the slowest flow-carrying paths of the unicast k of $f_{\mathsf{SO}}(R)$, for each $k=1,2,\ldots,K$.

It is clear that we have the following for $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$

$$|f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]| = (1-\epsilon)\cdot R,$$

i.e., $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$ can support $(1-\epsilon)$-fraction of the aggregate rate requirement of all unicasts. However, the sacrificed flow rate $(\epsilon\cdot R)$ may not be evenly distributed among individual unicasts, i.e., there may exist an unicast $k\in\{1,2,\ldots,K\}$ where $|f^k_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]| < (1-\epsilon)\cdot R_k$. As a comparison, according to the definition of $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$, it is clear that

$$\left|f^k_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]\right| = (1-\epsilon)\cdot R_k, \quad \forall k=1,2,\ldots,K.$$

For the average delay gap of $f_{\mathsf{NE}}[(1-\epsilon)R]$, we note that [5, Theorem 3.2] holds in the multiple-unicast setting.

Theorem 11: In the multiple-unicast setting, we have the following for $f_{\mathsf{NE}}[(1-\epsilon)R]$

$$\mathcal{A}(f_{\mathsf{NE}}[(1-\epsilon)R]) \le (1/\epsilon)\cdot\mathcal{A}^*(R),$$
$$\mathcal{M}(f_{\mathsf{NE}}[(1-\epsilon)R]) \le (1/\epsilon)\cdot(R/R_{\min})\cdot\mathcal{M}^*(R).$$

Proof: The average delay gap follows a similar proof to Theorem 5, and is an easy adaptation of [5, Theorem 3.2]. The maximum delay gap follows a similar proof to Theorem 6, with the help of Lemma 2. ■

For $f_{\mathsf{MM}}[(1-\epsilon)R]$, we have the following theorem.

Theorem 12: In the multiple-unicast setting, we have the following for $f_{\mathsf{MM}}[(1-\epsilon)R]$

$$\mathcal{A}(f_{\mathsf{MM}}[(1-\epsilon)R]) \le (1/\epsilon)\cdot(R/R_{\min})\cdot\mathcal{A}^*(R),$$
$$\mathcal{M}(f_{\mathsf{MM}}[(1-\epsilon)R]) \le \mathcal{M}^*(R).$$

Proof: The average delay gap follows a similar proof to Theorem 7, with the help of Lemma 2. The maximum delay gap holds straightforwardly, because all link delay functions are non-decreasing. ■

For $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$, we have the following theorem.

Theorem 13: In the multiple-unicast setting, we have the following for $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$

$$\mathcal{A}(f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]) \le \mathcal{A}^*(R),$$
$$\mathcal{M}(f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]) \le (1/\epsilon)\cdot\mathcal{M}^*(R).$$

TABLE IV

BI-CRITERIA DELAY GAPS OF OUR PROPOSED SOLUTIONS UNDER THE MULTIPLE-UNICAST SCENARIO

	Average delay gap compared to optimal	Maximum delay gap compared to optimal
$f_{\mathsf{NE}}[(1-\epsilon)R]$	$(1-\epsilon, 1/\epsilon)$	$(1-\epsilon, R/(\epsilon\cdot R_{\min}))$
$f_{\mathsf{MM}}[(1-\epsilon)R]$	$(1-\epsilon, R/(\epsilon\cdot R_{\min}))$	$(1-\epsilon, 1)$
$f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]^*$	$(1-\epsilon, 1)^*$	$(1-\epsilon, 1/\epsilon)^*$
$f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$	$(1-\epsilon, R/R_{\min})$	$(1-\epsilon, R/(\epsilon\cdot R_{\min}))$

Note. *: solution $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$ can only support $(1-\epsilon)$-fraction of the aggregate rate requirement R, while all the other three solutions can support $(1-\epsilon)$-fraction of the rate requirement R_k for each unicast $k=1,\ldots,K$.

Proof: The average delay gap follows a similar proof to Theorem 3. The maximum delay gap follows a similar proof to Theorem 4. ■

For $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$, we have the following theorem.

Theorem 14: In the multiple-unicast setting, we have the following for $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$

$$\mathcal{A}(f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]) \le (R/R_{\min})\cdot\mathcal{A}^*(R),$$
$$\mathcal{M}(f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]) \le (1/\epsilon)\cdot(R/R_{\min})\cdot\mathcal{M}^*(R).$$

Proof: Refer to Part C of supplementary materials. ■

We summarize our results in Table IV. We observe that $f_{\mathsf{SF\text{-}1}}[(1-\epsilon)R]$ can obtain a maximum delay and an average delay both within bi-criteria constant-ratio gaps to optimal, while all the other three solutions can obtain a maximum delay and an average delay both within bi-criteria problem-dependent-ratio gaps to optimal. Here the problem-dependent ratio only depends on flow rate requirements of individual unicasts, and is independent to link delay functions.

For general network topologies and arbitrary link delay functions, comparing existing results in Table III with ours in Table IV, we note that existing delay gaps depend on link delay functions, and hence can be unbounded; as a comparison, our bi-criteria delay gaps are independent to link delay functions, and hence are always bounded. We leave it as a future direction of analyzing the tightness of the delay gaps of our solutions under the multiple-unicast setting.

C. Our Solutions Can Solve Other Delay-Aware Network Flow Problems Approximately With Performance Guarantee

As shown in Table IV, $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$ can support $(1-\epsilon)$-fraction of the rate requirement of each unicast, but its bi-criteria approximation ratios of minimizing the two delay metrics are problem-dependent. In this section, we show that in fact the design of $f_{\mathsf{SF\text{-}2}}[(1-\epsilon)R]$ suggests a new avenue for solving many other delay-aware network flow problems, providing bi-criteria constant approximation ratios.

Different from the maximum delay and the average delay defined by [2]–[4], another practically meaningful way to define the maximum delay and the average delay of a multiple-unicast flow f is as follows

$$\bar{\mathcal{M}}(f) \triangleq \sum_{k=1}^{K} w_k\cdot\mathcal{M}\left(f^k\right), \quad \bar{\mathcal{A}}(f) \triangleq \sum_{k=1}^{K} w_k\cdot\mathcal{A}\left(f^k\right),$$

where $w_k\ge 0, k=1,2,\ldots,K$ is a weight associated with the delay performance of the unicast k. With the new definitions of the two delay metrics, we can have the following two problems

$$\bar{\mathsf{MM}}: \min_f \bar{\mathcal{M}}(f) \quad \text{s.t.} \quad \left|f^k\right| = R_k, \quad \forall k = 1, 2, \ldots, K, \quad (30)$$

$$\bar{\mathsf{SO}}: \min_f \bar{\mathcal{A}}(f) \quad \text{s.t.} \quad \left|f^k\right| = R_k, \quad \forall k = 1, 2, \ldots, K, \quad (31)$$

and we can define the following solution

- $f_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]$: similar to Algorithm 1, it is a solution returned after we iteratively delete a total of $(1-\epsilon)\cdot R_k$ rate from the slowest flow-carrying paths of the unicast k of $f_{\bar{\mathsf{SO}}}(R)$, for each $k = 1, 2, \ldots, K$.

It is clear that $|f^k_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]| = (1-\epsilon)\cdot R_k, \forall k = 1, 2, \ldots, K$. And we have the following theorem for it.

Theorem 15: In the multiple-unicast setting, we have the following for $f_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]$

$$\bar{\mathcal{A}}(f_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]) \leq \bar{\mathcal{A}}^*(R),$$
$$\bar{\mathcal{M}}(f_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]) \leq (1/\epsilon)\cdot \bar{\mathcal{M}}^*(R).$$

Proof: Follow a similar proof to those of Lemma 1, Theorem 4, and Theorem 14. ■

We observe that the design of our $f_{\bar{\mathsf{SF}}\text{-2}}[(1-\epsilon)R]$ suggests a method for developing bi-criteria constant-ratio approximation algorithms for optimizing multiple-unicast network flows that are sensitive both to the average delay and to the maximum delay of individual unicasts. We refer to our recent work [29], [30] for more results of leveraging this method to solve general multiple-unicast network delay optimization problems.

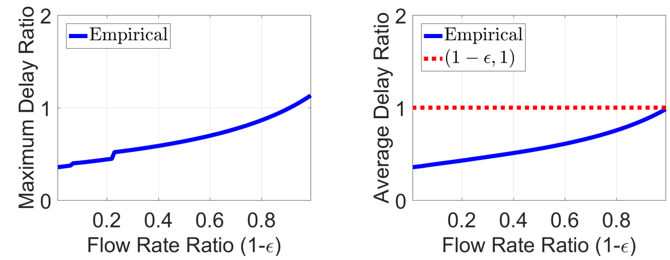
VII. Performance Evaluation

We simulate a real-world continent-scale network *GEANT* which is the main European research and education network with 45 nodes and 57 undirected links. The topology of GEANT can be found in [31] and Part D of supplementary materials. We assume link delay function to be the queuing delay function in (1), where the capacity $c_e \in \{5, 10, 20, 30, 100\}$ (Gbps) of a link $e \in E$ is set according to [31]. Each undirected link is treated as two directed links that operate independently and have identical capacities, as the setting in [15]. The edge-based system-optimal flow and Nash flow are modeled as convex programs and solved using *CVX* [32]. Other experiments including flow decomposition and Algorithm 1 are implemented in *C++*.

A. Comparing Our Solutions With the Optimal

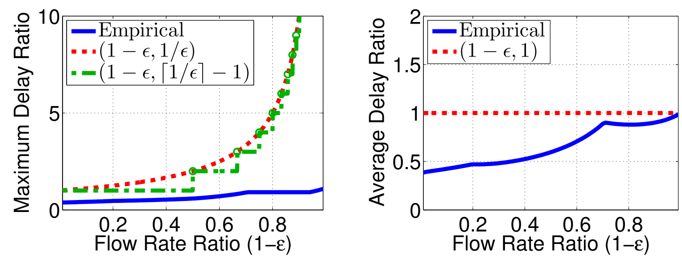
We assume that there are 1000 video conferencing sessions from the top-left node Iceland (IS) to the bottom-right node Israel (IL). The traffic demand for each session is 10Mbps, leading to a total flow rate requirement of $R = 10$Gbps.

Note that we cannot find $f_{\mathsf{MM}}(R)$ in our simulation since it is NP-hard and the network is a dense graph (there are more than 30 paths from IS to IL). Instead we find another feasible flow $\hat{f}_{\mathsf{MM}}(R)$ to approximate $f_{\mathsf{MM}}(R)$ efficiently, since $\mathcal{M}(\hat{f}_{\mathsf{MM}}(R))$ is no worse than 1.73% more of $\mathcal{M}(f_{\mathsf{MM}}(R))$ in average among extensive simulation instances. Details of obtaining $\hat{f}_{\mathsf{MM}}(R)$ refer to Part D of our supplementary materials.



(a) Maximum delay (b) Average delay

Fig. 2. Empirical and theoretical bi-criteria delay gaps of $f_{\mathsf{SO}}[(1-\epsilon)R]$ as compared to the optimal.



(a) Maximum delay (b) Average delay

Fig. 3. Empirical and theoretical bi-criteria delay gaps of $f_{\mathsf{SF}}[(1-\epsilon)R]$ as compared to the optimal.

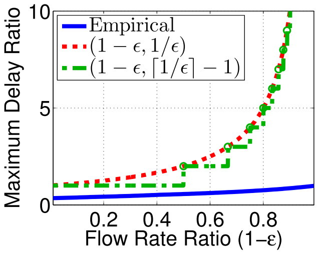
Compare the Maximum Delay of Our Solutions With the Optimal: As a benchmark, without sacrificing flow, we first obtain the ratio comparing the maximum delay of $f_{\mathsf{SO}}(R)$ to that of $\hat{f}_{\mathsf{MM}}(R)$, which is 1.128, and the ratio comparing the maximum delay of $f_{\mathsf{NE}}(R)$ to that of $\hat{f}_{\mathsf{MM}}(R)$, which is 1.017.

We increase the sacrificing ratio ϵ from 0.01 to 0.99 with a step of 0.01. Figure 2a presents the ratio comparing the maximum delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ to that of $\hat{f}_{\mathsf{MM}}(R)$. According to the figure, different from its theoretical maximum delay gap that is unbounded for the queuing delay function in (1), the empirical maximum delay gap of $f_{\mathsf{SO}}[(1-\epsilon)R]$ compared to optimal is very small. Besides, a larger sacrifice of the flow rate requirement leads to a smaller maximum delay performance.

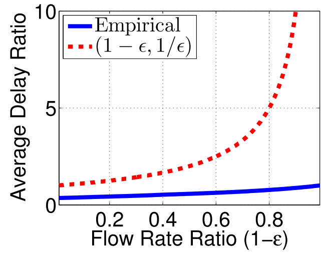
Figure 3a illustrates the ratio comparing the maximum delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ to that of $\hat{f}_{\mathsf{MM}}(R)$. Similar to Figure 2a, the empirical maximum delay gap of $f_{\mathsf{SF}}[(1-\epsilon)R]$ is substantially smaller than its theoretical counterpart. For an acceptable small loss in flow rate for video conferencing, e.g., $\epsilon = 0.03$, the maximum delay gap of $f_{\mathsf{SF}}[(1-\epsilon)R]$ is 1.007, which is better than that without flow rate loss, i.e., than 1.128.

Figure 4a shows the maximum delay results of $f_{\mathsf{NE}}[(1-\epsilon)R]$. For $\epsilon = 0.03$, the maximum delay gap is 0.953 which is better than 1.017 when we do not sacrifice rate requirement. We observe similar results of $\hat{f}_{\mathsf{MM}}[(1-\epsilon)R]$ in Figure 5a.

Compare the Average Delay of Our Solutions With the Optimal: Figure 2b gives the ratio comparing the average delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ to that of $f_{\mathsf{SO}}(R)$, where we empirically observe that the average delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ is non-decreasing with the flow rate ratio $(1-\epsilon)$, i.e., the minimal average delay is non-decreasing with the flow rate requirement. Figure 3b gives the ratio comparing the average delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ to that of $f_{\mathsf{SO}}(R)$, where it is obvious that the

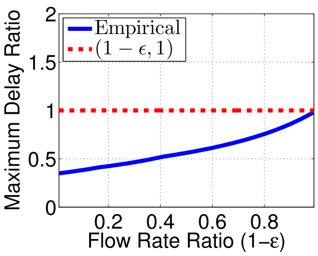


(a) Maximum delay

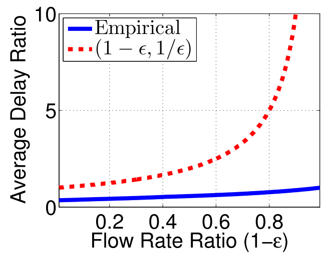


(b) Average delay

Fig. 4. Empirical and theoretical bi-criteria delay gaps of $f_{\mathsf{NE}}[(1-\epsilon)R]$ as compared to the optimal.



(a) Maximum delay



(b) Average delay

Fig. 5. Empirical and theoretical bi-criteria delay gaps of $\hat{f}_{\mathsf{MM}}[(1-\epsilon)R]$ as compared to the optimal.

average delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ is always no greater than that of $f_{\mathsf{SO}}(R)$. Interestingly, different from the average delay results of $f_{\mathsf{SO}}[(1-\epsilon)R]$ shown in Figure 2b, the average delay of $f_{\mathsf{SF}}[(1-\epsilon)R]$ is not monotonic with the flow rate ratio $(1-\epsilon)$. From Figure 4b (resp. Figure 5b), we can see that the empirical average delay gap of $f_{\mathsf{NE}}[(1-\epsilon)R]$ (resp. of $\hat{f}_{\mathsf{MM}}[(1-\epsilon)R]$) is much smaller than the theoretical counterpart $(1-\epsilon, 1/\epsilon)$.

Moreover, even for a small rate loss, for example $\epsilon = 0.03$, the average delay gap is 0.972 both for $f_{\mathsf{NE}}[(1-\epsilon)R]$ and $\hat{f}_{\mathsf{MM}}[(1-\epsilon)R]$. This again shows the benefit of controllably sacrificing flow rate: without sacrificing rate, the experimental average delay gap is 1.037 for both $f_{\mathsf{NE}}(R)$ and $\hat{f}_{\mathsf{MM}}(R)$.

Overall, we observe that we can obtain better performances both on the maximum delay and on the average delay when we can sacrifice a small portion (e.g., $\epsilon = 0.03$) of flow rate requirement. We remark that 3% traffic loss is very acceptable for video conferencing engines with loss protection/recovery and error resilience capabilities [19]. Besides, for our proposed solutions, their empirical delay gaps are much smaller than the theoretical counterparts.

B. Comparing Our Solutions With a Conceivable Baseline

In the previous section we compare the delays of our solutions that support $(1-\epsilon)$-fraction of the rate requirement with the optimal delay under full rate requirement. In this section, we further compare the delay performance of our solutions under reduced rate requirement with that of a conceivable delay-aware baseline under full rate requirement.

A Conceivable Baseline $f_{\mathsf{B}}(R)$: In order to stream R flow rate from a sender to a receiver with low end-to-end delay, a conceivable approach is to iteratively assign a small fraction of the flow rate to the fastest path till all the flow rate requirement are routed. We denote $f_{\mathsf{B}}(R)$ as the solution to this conceivable baseline. Specifically, in each iteration this conceivable approach (i) obtains the fastest sender-to-receiver path, then (ii) assigns a rate of $(\theta \cdot R)$ $(\theta \in (0,1))$ to the path, and (iii) updates the delay of all links $e \in E$.

It is clear that better delay performance can be achieved by the baseline with smaller θ (Table V gives an illustrative example). We set $\theta = 0.01$ in the following simulations. To be consistent, we also give simulation results of $f_{\mathsf{B}}[(1-\epsilon)R]$, which is the solution of the baseline under a smaller rate requirement of $(1-\epsilon) \cdot R$. In this section we do not evaluate $f_{\mathsf{MM}}[(1-\epsilon)R]$ due to the following two concerns: (i) it is time-consuming to obtain $f_{\mathsf{MM}}[(1-\epsilon)R]$ because of the NP-hardness of MM, and (ii) we set $\theta = 1\%$ for the baseline to obtain $f_{\mathsf{B}}[(1-\epsilon)R]$, where 1% is small enough for $f_{\mathsf{B}}[(1-\epsilon)R]$ to be a good approximation to $f_{\mathsf{MM}}[(1-\epsilon)R]$.

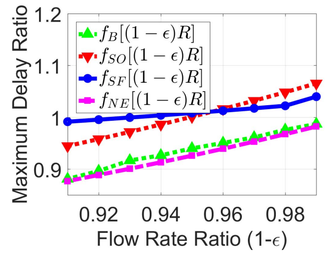
We assume that the sender is IS and the receiver is IL. First, for each $\epsilon \in (0, 0.1)$ with a step of 0.01 where R is fixed to be 10, Figure 6 gives the maximum delay ratio result and average delay ratio result, comparing the delay of solutions with a reduced rate requirement of $(1-\epsilon) \cdot R$ with that of the baseline $f_{\mathsf{B}}(R)$ under the full rate requirement of R. From Figure 6a (resp. Figure 6b) we observe that (i) solutions with reduced rate requirement obtain a smaller maximum delay (resp. smaller average delay) than $f_{\mathsf{B}}(R)$ that is under full rate requirement, as the maximum delay ratio (resp. average delay ratio) is below 1 for many instances; (ii) a smaller maximum delay gap (resp. smaller average delay gap) can be achieved as the ratio of the sacrificed flow rate requirement (i.e., ϵ) becomes larger; (iii) the maximum delay (resp. average delay) of $f_{\mathsf{SO}}[(1-\epsilon)R]$ and $f_{\mathsf{SF}}[(1-\epsilon)R]$ is larger than (resp. smaller than) that of $f_{\mathsf{NE}}[(1-\epsilon)R]$ and $f_{\mathsf{B}}[(1-\epsilon)R]$; and (iv) the maximum delay (resp. average delay) of $f_{\mathsf{NE}}[(1-\epsilon)R]$ is slightly smaller than that of $f_{\mathsf{B}}[(1-\epsilon)R]$.

Next, for each flow rate requirement $R \in (4, 14)$ with a step of 1 where ϵ is fixed to be 0.03, Figure 7 gives the maximum delay ratio result and average delay ratio result, comparing the delay of solutions supporting a flow rate of $(1-\epsilon) \cdot R$ with that of the baseline $f_{\mathsf{B}}(R)$. From the figure we observe

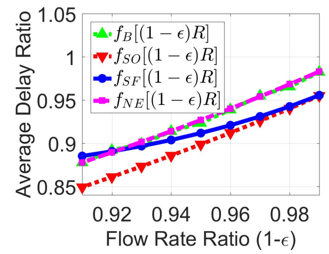
TABLE V

WE SET THE SENDER TO BE IS, THE RECEIVER TO BE IL, AND THE RATE REQUIREMENT TO BE 10. WE GIVE DELAY RESULTS OF THE CONCEIVABLE BASELINE WITH DIFFERENT θ

θ	0.01	0.05	0.1	0.2
Max-delay	0.63	0.65	0.68	1.17
Avg-delay	0.63	0.64	0.67	0.77

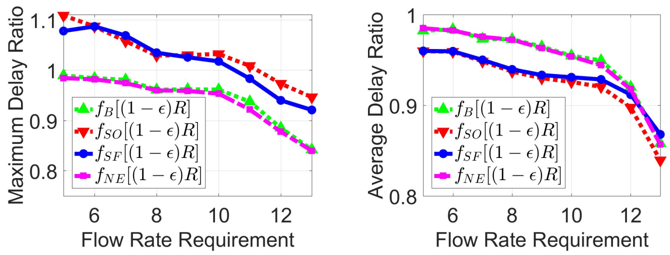


(a) Maximum delay



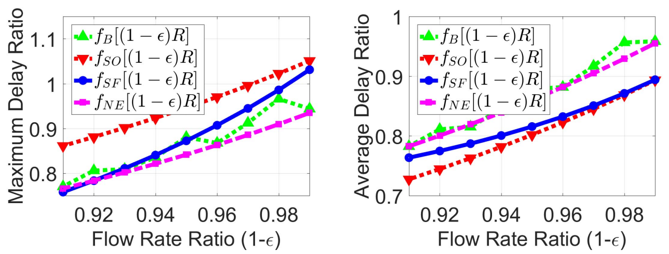
(b) Average delay

Fig. 6. Simulated delay ratio results with the approximation parameter ϵ, for instances of $s = \mathsf{IS}$, $t = \mathsf{IL}$, and $R = 10$. We compare our solutions with the baseline $f_{\mathsf{B}}(R)$.



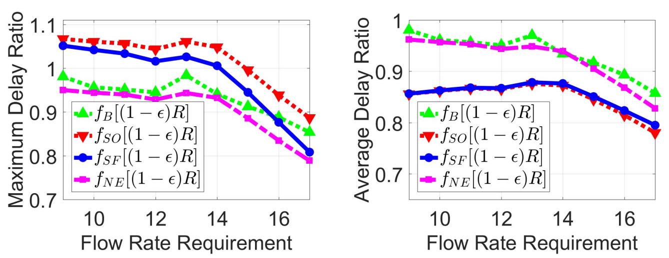
(a) Maximum delay (b) Average delay

Fig. 7. Simulated delay ratio results with the rate requirement R, for instances of $s = \mathsf{IS}$, $t = \mathsf{IL}$, and $\epsilon = 0.03$. We compare our solutions with the baseline $f_{\mathsf{B}}(R)$.



(a) Maximum delay (b) Average delay

Fig. 8. Simulated delay ratio results with the approximation parameter ϵ, for instances of $s = \mathsf{PT}$, $t = \mathsf{EE}$, and $R = 15$. We compare our solutions with the baseline $f_{\mathsf{B}}(R)$.



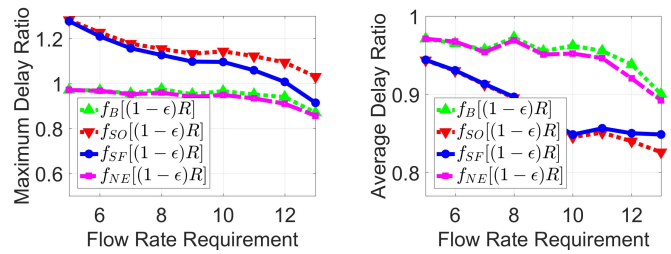
(a) Maximum delay (b) Average delay

Fig. 9. Simulated delay ratio results with the rate requirement R, for instances of $s = \mathsf{PT}$, $t = \mathsf{EE}$, and $\epsilon = 0.03$. We compare our solutions with the baseline $f_{\mathsf{B}}(R)$.

that (i) smaller maximum delay gap and smaller average delay gap can be achieved as the flow rate requirement (i.e., R) becomes larger; (ii) comparing the two solutions $f_{\mathsf{SO}}[(1-\epsilon)R]$ and $f_{\mathsf{SF}}[(1-\epsilon)R]$ with the other two solutions $f_{\mathsf{NE}}[(1-\epsilon)R]$ and $f_{\mathsf{B}}[(1-\epsilon)R]$, the maximum delay of the former is larger than that of the latter while the average delay of the former is smaller than that of the latter; and (iii) $f_{\mathsf{NE}}[(1-\epsilon)R]$ performs slightly better than $f_{\mathsf{B}}[(1-\epsilon)R]$, achieving both a smaller maximum delay and a smaller average delay.

We present simulation results of another sender-receiver pair (sender is the bottom-left Portugal, or PT in short, and receiver is the top-right Estonia, or EE in short) in Figures 8 and 9. We observe that Figure 8 (resp. Figure 9) is similar to Figure 6 (resp. Figure 7). We also simulate the sender-receiver pair $(\mathsf{IS}, \mathsf{EE})$ and give the delay results with the rate requirement in Figure 10, which is similar to Figures 7 and 9.

Overall, we observe that both a much smaller maximum delay and a much smaller average delay can be achieved if one can accept a small traffic loss. For example,



(a) Maximum delay (b) Average delay

Fig. 10. Simulated delay ratio results with the rate requirement R, for instances of $s = \mathsf{IS}$, $t = \mathsf{EE}$, and $\epsilon = 0.03$. We compare our solutions with the baseline $f_{\mathsf{B}}(R)$.

for a 3% traffic loss when the rate requirement is large, as shown in Figures 7 and 9, over 10% reduction can be obtained on both delay metrics for each of the four solutions under reduced rate requirement, as compared to $f_{\mathsf{B}}(R)$ that is under full rate requirement. Moreover, empirically $f_{\mathsf{SO}}[(1-\epsilon)R]$ and $f_{\mathsf{SF}}[(1-\epsilon)R]$ outperform $f_{\mathsf{B}}[(1-\epsilon)R]$ and $f_{\mathsf{NE}}[(1-\epsilon)R]$ in minimizing the average delay, but $f_{\mathsf{B}}[(1-\epsilon)R]$ and $f_{\mathsf{NE}}[(1-\epsilon)R]$ outperform $f_{\mathsf{SO}}[(1-\epsilon)R]$ and $f_{\mathsf{SF}}[(1-\epsilon)R]$ in minimizing the maximum delay. In addition, $f_{\mathsf{NE}}[(1-\epsilon)R]$ is slightly better than $f_{\mathsf{B}}[(1-\epsilon)R]$ in minimizing both delay metrics. The average delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ is smaller than that of $f_{\mathsf{SF}}[(1-\epsilon)R]$, while the maximum delay of $f_{\mathsf{SO}}[(1-\epsilon)R]$ is larger than that of $f_{\mathsf{SF}}[(1-\epsilon)R]$.

Finally we remark that our simulated sender-receiver pairs $((\mathsf{IS}, \mathsf{IL})$, $(\mathsf{PT}, \mathsf{EE})$, and $(\mathsf{IS}, \mathsf{EE}))$ are representative and serve our purpose well: These pairs are transcontinental, and there are many different sender-to-receiver paths. Therefore, the system-optimal flow, min-max flow, and Nash flow differ from each other. In comparison, randomly generating sender-receiver pairs suffers from a limitation that the number of paths from the sender to the receiver may be limited. Hence different delay-aware network flows may have similar flow rate assignments, leading to similar delay performances.

VIII. Conclusion

We consider that a sender streams a flow at a fixed rate to a receiver across a multi-hop network, and transmission over a link incurs a delay modeled as a non-negative, non-decreasing, and differentiable function of the link aggregate transmission rate. We optimize two popular network delay metrics, i.e., maximum delay and average delay. For general network topologies and arbitrary link delay functions, well-known pessimistic results state that a flow cannot simultaneously achieve a maximum delay and an average delay both within bounded-ratio gaps to optimal. But we design three solutions, all of which can deliver $(1-\epsilon)$-fraction of the flow with maximum delay and average delay simultaneously within a $(1/\epsilon)$-ratio gap to optimal, for any user-defined $\epsilon \in (0, 1)$. Hence, our results pose an optimistic note on the fundamental compatibility of the two delay metrics. The ratio $(1/\epsilon)$ is independent to the network topology and link delay function, and is at least near-tight. Furthermore, our solutions can be extended to the multiple-unicast setting, where each of them must obtain a maximum delay and an average delay both within a bounded-ratio gap of $(R/(R_{\min} \cdot \epsilon))$

to optimal. Thus our results pose a similar optimistic note in the multiple-unicast setting. Here R (resp. $R_{\min}$) is the aggregate (resp. minimum) flow rate requirement of all individual unicasts.

REFERENCES

[1] Q. Liu, L. Deng, H. Zeng, and M. Chen, "A tale of two metrics in network delay optimization," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 2123–2131.

[2] J. R. Correa, A. S. Schulz, and N. E. S. Moses, "Computational complexity, fairness, and the price of anarchy of the maximum latency problem," in *Int. Conf. Integer Program. Combinat. Optim.*, 2004, pp. 59–73.

[3] J. R. Correa, A. S. Schulz, and N. E. Stier-Moses, "Fast, fair, and efficient flows in networks," *Oper. Res.*, vol. 55, no. 2, pp. 215–225, Apr. 2007.

[4] T. Roughgarden, "The maximum latency of selfish routing," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, 2004, pp. 1–3.

[5] T. Roughgarden and É. Tardos, "How bad is selfish routing?" *J. ACM*, vol. 49, no. 2, pp. 236–259, Mar. 2002.

[6] T. Roughgarden, "The price of anarchy is independent of the network topology," *J. Comput. Syst. Sci.*, vol. 67, no. 2, pp. 341–364, Sep. 2003.

[7] M. Alizadeh *et al.*, "Data center TCP (DCTCP)," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 4, p. 63, Aug. 2010.

[8] B. Vamanan, J. Hasan, and T. N. Vijaykumar, "Deadline-aware datacenter TCP (D^2TCP)," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, p. 115, Sep. 2012.

[9] Y. Liu, D. Niu, and B. Li, "Delay-optimized video traffic routing in software-defined interdatacenter networks," *IEEE Trans. Multimedia*, vol. 18, no. 5, pp. 865–878, May 2016.

[10] M. H. Hajiesmaili, L. T. Mak, Z. Wang, C. Wu, M. Chen, and A. Khonsari, "Cost-effective low-delay design for multiparty cloud video conferencing," *IEEE Trans. Multimedia*, vol. 19, no. 12, pp. 2760–2774, Dec. 2017.

[11] *WebEx*. Accessed: Jul. 2017. [Online]. Available: https://blog.webex.com/2016/01/five-reasons-to-join-a-webex-now/

[12] *Skype*. Accessed: Jul. 2017. [Online]. Available: https://news.microsoft.com/bythenumbers/skype-calls

[13] *Cisco*. Accessed: Jul. 2017. [Online]. Available: http://www.cisco.com/c/dam/en/us/solutions/collateral/collaboration/mid%size-collaboration-solutions/high-growth-innovative-companies.pdf

[14] *Series G: Transmission Systems and Media, Digital Systems and Networks*, Int. Telecommun. Union, Geneva, Switzerland, 2003.

[15] B. Grimmer and S. Kapoor, "Nash equilibrium and the price of anarchy in priority based network routing," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[16] L. Kleinrock, "Analytic and simulation methods in computer network design," in *Proc. Spring Joint Comput. Conf.*, 1970, pp. 569–579.

[17] S. Basu, G. Yang, T. Lianeas, E. Nikolova, and Y. Chen, "Reconciling selfish routing with social good," in *Proc. Int. Symp. Algorithmic Game Theory*, 2017, pp. 149–157.

[18] A. Czumaj and B. Vöcking, "Tight bounds for worst-case equilibria," *ACM Trans. Algorithms*, vol. 3, no. 1, p. 4, Feb. 2007.

[19] I. M. Weinstein. (Jan. 2008). "Polycom's lost packet recovery (LPR) capability." Wainhouse Research. [Online]. Available: http://docs.polycom.com/global/documents/whitepapers/lost_packet_recovery_eval_report.pdf

[20] G. Christodoulou, E. Koutsoupias, and P. G. Spirakis, "On the performance of approximate equilibria in congestion games," *Algorithmica*, vol. 61, no. 1, pp. 116–140, Sep. 2011.

[21] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," *Comput. Sci. Rev.*, vol. 3, no. 2, pp. 65–69, May 2009.

[22] M. Mavronicolas and P. Spirakis, "The price of selfish routing," *Algorithmica*, vol. 48, no. 1, pp. 91–126, 2007.

[23] D. Weitz, "The price of anarchy," Univ. California, Berkeley, Berkeley, CA, USA, unpublished.

[24] F. Devetak, J. Shin, T. Anjali, and S. Kapoor, "Minimizing path delay in multipath networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–5.

[25] S. Misra, G. Xue, and D. Yang, "Polynomial time approximations for multi-path routing with bandwidth and delay constraints," in *Proc. 28th Conf. Comput. Commun.*, Apr. 2009.

[26] W. Zhang, J. Tang, C. Wang, and S. de Soysa, "Reliable adaptive multipath provisioning with bandwidth and differential delay constraints," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[27] M. Beckmann, C. B. McGuire, and C. B. Winsten, "Studies in the economics of transportation," Yale Univ., New Haven, CT, USA, Tech. Rep. 226, 1956.

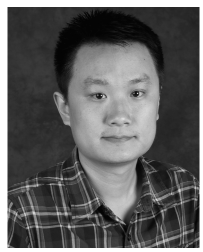
[28] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Can. J. Math.*, vol. 8, no. 3, pp. 399–404, 1956.

[29] Q. Liu, H. Zeng, and M. Chen, "Network utility maximization under maximum delay constraints and throughput requirements," in *Proc. 20th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2019, pp. 391–392.

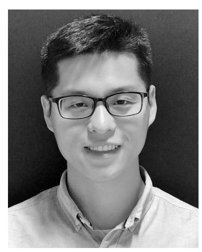
[30] Q. Liu, H. Zeng, M. Chen, and L. Liu, "Cost minimization in multipath communication under throughput and maximum delay constraints," in *Proc. IEEE Int. Conf. Comput. Commun.*, Jul. 2020, accepted for publication.

[31] (2017). *The Pan-European Research and Education Network*. [Online]. Available: https://geant3plus.archive.geant.net/Resources/Media_Library/Documents/GEANT_Project_TopologyMAR14_Web.pdf

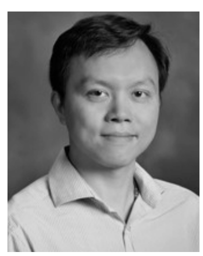
[32] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx



Qingyu Liu (Member, IEEE) received the M.S. degree in computer science and technology from Tsinghua University in 2014 and the Ph.D. degree in electrical and computer engineering from Virginia Tech in 2019. He was a Visiting Student of information engineering with The Chinese University of Hong Kong from August 2017 to December 2017. He is currently a Post-Doctoral Associate with Virginia Tech. His research interests are optimization and algorithm design in the fields of network communication and transportation.



Lei Deng (Member, IEEE) received the B.E. degree from the Department of Electronic Engineering, Shanghai Jiao Tong University, in 2012 and the Ph.D. degree from the Department of Information Engineering, The Chinese University of Hong Kong, in 2017. In 2015, he was a Visiting Scholar with the School of Electrical and Computer Engineering, Purdue University. He is currently an Assistant Professor with the College of Electronics and Information Engineering, Shenzhen University. He is also a member of the Shenzhen Key Laboratory of Digital Creative Technology and the Guangdong Province Engineering Laboratory for Digital Creative Technology. His research interests are digital creative technology, delay-constrained communications, and energy-efficient timely transportation.



Haibo Zeng received the Ph.D. degree in electrical engineering and computer sciences from the University of California at Berkeley. He was a Senior Researcher with General Motors Research and Development until October 2011 and an Assistant Professor with McGill University until August 2014. He is currently with the Department of Electrical and Computer Engineering, Virginia Tech. His research interests are embedded systems, cyber-physical systems, and real-time systems. He received the four best paper/best student paper awards in the above fields.

Minghua Chen (Senior Member, IEEE) received the B.E. and M.S. degrees from the Department of Electronic Engineering, Tsinghua University, and the Ph.D. degree from the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in 2006. He is currently a Professor with the School of Data Science, City University of Hong Kong. His research interests include energy systems such as smart power grids and energy-efficient data centers, intelligent transportation systems, distributed optimization, multimedia networking, wireless networking, delay-constrained network coding, machine learning in networked and societal systems, and capitalizing the benefit of data-driven prediction in algorithm/system design. He received several best paper awards, including the IEEE ICME Best Paper Award in 2009, the IEEE TRANSACTIONS ON MULTIMEDIA Prize Paper Award in 2009, and the ACM Multimedia Best Paper Award in 2012.