Delay-Based Back-Pressure Scheduling in Multihop Wireless Networks

Bo Ji, *Student Member, IEEE,* Changhee Joo, *Member, IEEE,* and Ness B. Shroff, *Fellow, IEEE*

***Abstract*—Scheduling is a critical and challenging resource allocation mechanism for multihop wireless networks. It is well known that scheduling schemes that favor links with larger queue length can achieve high throughput performance. However, these queue-length-based schemes could potentially suffer from large (even infinite) packet delays due to the well-known *last packet problem*, whereby packets belonging to some flows may be excessively delayed due to lack of subsequent packet arrivals. Delay-based schemes have the potential to resolve this last packet problem by scheduling the link based on the delay the packet has encountered. However, characterizing throughput-optimality of these delay-based schemes has largely been an open problem in multihop wireless networks (except in limited cases where the traffic is single-hop.) In this paper, we investigate delay-based scheduling schemes for multihop traffic scenarios with fixed routes. We develop a scheduling scheme based on a new delay metric, and show that the proposed scheme achieves optimal throughput performance. Further, we conduct simulations to support our analytical results, and show that the delay-based scheduler successfully removes excessive packet delays, while it achieves the same throughput region as the queue-length-based scheme.**

***Index Terms*—Throughput-optimal, Scheduling, Delay-based, Back-pressure, Fluid limit, Lyapunov approach**

I. Introduction

Link scheduling is a critical resource allocation component in multihop wireless networks, and also perhaps the most challenging. The seminal work of [1] introduces a joint adaptive routing and scheduling algorithm, called Queue-length-based Back-Pressure (Q-BP), that has been shown to be throughput-optimal, i.e., it can stabilize the network under any feasible load. This paper focuses on the settings with fixed routes, where the Q-BP algorithm becomes a scheduling algorithm. Since the development of Q-BP, there have been numerous extensions that have integrated it in an overall optimal cross-layer framework. Further, easier-to-implement queue-length-based scheduling schemes have been developed and shown to be throughput-efficient (see [2] and references therein). Some recent attempts [3]–[5] focus on designing real-world wireless protocols using the ideas behind these algorithms.

While these queue-length-based schedulers have been shown to achieve excellent throughput performance, they are usually evaluated under the assumption that flows have an infinite amount of data and keep injecting packets into the network. However, in practice, when accounting for multiple time scales [6]–[8], there also exist other types of flows that have a finite number of packets to transmit, which can result in the well-known *last packet problem*: consider a queue that holds the last packet of a flow, then the packet does not see any subsequent packet arrivals, and thus the queue length remains very small and the link may be starved for a long time, since the queue-length-based schemes give a higher priority to links with a larger queue length. In such a scenario with flow-level dynamics, it has also been shown in [6] that the queue-length-based schemes may not even be throughput-optimal.

Recent works in [9]–[14] have studied the performance of delay-based scheduling algorithms that use Head-of-Line (HOL) delays instead of queue lengths as link weights. One desirable property of the delay-based approach is that they provide an intuitive way around the last packet problem. The schedulers give a higher priority to the links with a larger weight as before, but now the weight (i.e., the HOL delay) of a link increases with time until the link is scheduled. Hence, if the link with the last packet is not scheduled at this moment, it is more likely to be scheduled in the next time. However, the throughput of the delay-based scheduling schemes is not fully understood, and has only been established for limited cases with single-hop traffic.

The delay-based approach was introduced in [9] for scheduling in Input-Queued switches. The results have been extended to wireless networks for single-hop traffic, providing throughput-optimal delay-based MaxWeight scheduling algorithms [11], [12], [15]. It has also been shown that delay-based schemes with appropriately chosen weight parameters provide good Quality of Service (QoS) [10], and can be used as an important component in a cross-layer protocol design [14]. The performance of the delay-based MaxWeight scheduler has been further investigated in a single-hop network with flow-level dynamics [13]. The results show that, when flows arrive at the base station carrying a finite amount of data, the delay-based MaxWeight scheduler achieves optimal throughput performance while its queue-length-based counterpart does not.

It should be noted that even for the multihop wireless networks with fixed routes, the scheduling problem is both important and challenging. There are many existing works focusing on such scenarios with fixed routes (see [16]–[18] for examples). *However, in multihop wireless networks, the throughput performance of these delay-based schemes has largely been an open problem.* To the best of our knowledge, even with the assumption of fixed routes, there are no prior

B. Ji is with Department of ECE at the Ohio State University. C. Joo is with School of ECE at UNIST, Korea. N. B. Shroff is with Departments of ECE and CSE at the Ohio State University.

Emails: ji@ece.osu.edu, cjoo@unist.ac.kr, shroff@ece.osu.edu.

This work was supported in part by ARO MURI Award W911NF-08-1-0238, and NSF Awards 1012700-CNS, 0721236-CNS, 0721434-CNS, and 1065136-CNS, and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science, and Technology (No. 2012-0003227).

A preliminary version of this work was presented at IEEE INFOCOM 2011.

works that employ delay-based algorithms to address the important issue of throughput-optimal scheduling in multihop wireless networks. Indeed, the problem becomes much more challenging in the multihop scenario. In [12], the key idea in showing throughput-optimality of the delay-based MaxWeight scheduler is to exploit the following property: after a finite time, there exists a linear relation between queue lengths and HOL delays in the fluid limits (which we formally define in Section III-A), where the ratio is the mean arrival rate. Hence, the delay-based MaxWeight scheme is basically equivalent to its queue-length-based counterpart, and thus achieves the optimal throughput. This property holds for the single-hop traffic. Since given that the exogenous arrival processes follow the Strong Law of Large Numbers (SLLN) and the fluid limits exist, the arrival processes are deterministic with constant rates in the fluid limits. *However, such a linear relation does not necessarily hold for the multihop traffic, since at a non-source (or relay) node, the arrival process may not satisfy SLLN and the packet arrival rate may not even be a constant, depending on the underlying schedulers dynamics.* To this end, we investigate delay-based scheduling schemes that achieve optimal throughput performance in multihop wireless networks.

Unlike previous delay-based schemes, we view the packet delay as a sojourn time in the network, and re-design the delay metric of the queue as the sojourn-time difference between the queue's HOL packet and the HOL packet of its previous hop (see Eq. (36) for the formal definition). Using this new metric, we can establish a linear relation between queue lengths and delays in the fluid limits. The linear relation then plays the key role in showing that the proposed Delay-based Back-Pressure (D-BP) scheduling scheme is throughput-optimal in multihop networks.

In summary, the main contributions of our paper are as follows:

- We devise a new delay metric for multihop wireless networks and develop the D-BP algorithm, under which a linear relation between queue lengths and delays in the fluid limits can be established. From this linear relation, we can show that D-BP achieves optimal throughput performance. To do this, we first re-visit throughput-optimality of Q-BP using fluid limit techniques. Further, we develop a simpler greedy approximation of D-BP for practical implementation.
- We provide extensive simulation results to evaluate the performance of the delay-based schedulers, including D-BP. Through simulations, i) we observe that the last packet problem can cause excessive delays for certain flows under Q-BP, while the problem is eliminated under D-BP. ii) We show that D-BP also achieves better fairness and prevents the flows that lack subsequent packet arrivals from starving. iii) Finally, we simulate the simpler greedy approximation algorithms of Q-BP and D-BP, and show that the delay-based approximation empirically achieves a throughput region that is no smaller than that of its queue-based counterpart.

The paper is organized as follows. In Section II, we present a detailed description of our system model. In Section III, we show throughput-optimality of Q-BP using fluid limit techniques, and extend the analysis to D-BP in Section IV. The discussions are further extended to the greedy algorithms in Section V. We evaluate the performance of delay-based schedulers through simulations in Section VI, and conclude our paper in Section VII.

II. System Model

We consider a multihop wireless network described by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E}$ denotes the set of links. Nodes are wireless transmitters/receivers and links are wireless channels between two nodes if they can directly communicate with each other. During a single time slot, multiple links that do not interfere with each other can be active at the same time, and each active link transmits one packet during the time slot if its queue is not empty. Let $\mathcal{S}$ denote the set of flows in the network. We assume that each flow has a single, fixed, and loop-free route. The route of flow s has an $H(s)$-hop length from the source to the destination, where each k-th hop link is denoted by (s,k). Let $H^{\max} \triangleq \max_{s\in\mathcal{S}} H(s) < \infty$ denote the length of the longest route over all flows. Note that the assumption of single route and unit link capacity is only for ease of exposition, and one can readily extend the results to more general scenarios with *multiple fixed routes and heterogeneous link rates*, applying the techniques used in this paper. To specify wireless interference, we consider the k-th hop of each flow s or link-flow-pair (s,k). Let $\mathcal{P}$ denote the set of all link-flow-pairs, i.e.,

$$\mathcal{P} \triangleq \{(s,k) \mid s \in \mathcal{S},\ 1 \le k \le H(s)\}.$$

The set of link-flow-pairs that interfere with (s,k) can be described as

$$I(s,k) \triangleq \{(r,j) \in \mathcal{P} \mid (r,j) \text{ interferes with } (s,k), \text{ or } (r,j) = (s,k)\}. \tag{1}$$

Note that the interference model we adopt is very general, and includes the class of the K*-hop* interference model[1]. A schedule is a set of (active or inactive) link-flow-pairs, and can be represented by a vector $\vec{M} \in \{0,1\}^{|\mathcal{P}|}$, where $|\cdot|$ denotes the cardinality of a set. Each element $M_{s,k}$ is set to 1 if link-flow-pair (s,k) is active, and 0 if link-flow-pair (s,k) is inactive. Slightly abusing the notation, we also use M to denote the set of active link-flow-pairs of $\vec{M}$, i.e., $M \triangleq \{(s,k) \in \mathcal{P} \mid M_{s,k} = 1\}$. A schedule $\vec{M}$ is said to be *feasible* if no two link-flow-pairs of $\vec{M}$ interfere with each other, i.e., $(r,j) \notin I(s,k)$ for all (r,j), (s,k) with $M_{r,j} = 1$ and $M_{s,k} = 1$. Let $\mathcal{M}_{\mathcal{P}}$ denote the set of all feasible schedules in $\mathcal{P}$, and let $Co(\mathcal{M}_{\mathcal{P}})$ denote its convex hull.

Let $A_s(t)$ denote the number of packet arrivals at the source node of flow s at time slot t. We assume that packets are of unit

[1] Under the K-hop interference model, two links within a K-hop "distance" interfere with each other and cannot be activated at the same time [19]. When $K = 1$, it is also called the *primary* or *node-exclusive* interference model. The 1-hop interference model has been known as a good representation for Bluetooth or FH-CDMA networks [20]–[23]. When $K = 2$, it is often used to model the ubiquitous IEEE 802.11 DCF (Distributed Coordination Function) wireless networks [22], [24]–[26].

length. Similar to [12], we assume that each arrival process $A_s(t)$ is a stationary and ergodic Markov chain with countable state space, and satisfies the Strong Law of Large Numbers (SLLN): That is, with probability one,

$$\lim_{t\to\infty} \frac{\sum_{\tau=0}^{t-1} A_s(\tau)}{t} = \lambda_s, \tag{2}$$

for each flow $s \in \mathcal{S}$, where λ_s denotes the mean arrival rate of flow s. We let $\vec{\lambda} \triangleq [\lambda_1, \lambda_2, \cdots, \lambda_{|\mathcal{S}|}]$ denote the arrival rate vector.

Let $Q_{s,k}(t)$ denote the number of packets at the queue of (s,k) at the beginning of time slot t. For notational ease, we also use $Q_{s,k}$ to denote the queue itself. We let $\vec{Q}(t) \triangleq [Q_{s,k}(t),\ (s,k) \in \mathcal{P}]$ denote the queue length vector at time slot t, and use $\|\cdot\|$ to denote the L_1-norm of a vector, e.g., $\|\vec{Q}(t)\| = \sum_{(s,k)\in\mathcal{P}} Q_{s,k}(t)$. Let $\Pi_{s,k}(t)$ denote the service of $Q_{s,k}$ at time slot t, which takes a value of either 1 if link-flow-pair (s,k) is active, or 0 otherwise, in our settings. We let $\Psi_{s,k}(t)$ denote the actual number of packets transmitted from $Q_{s,k}$ at time slot t. Clearly, we have $\Psi_{s,k}(t) \le \Pi_{s,k}(t)$ for all time slots $t \ge 0$. Let $P_{s,k}(t) \triangleq \sum_{i=1}^{k} Q_{s,i}(t)$ denote the cumulative queue lengths up to the k-th hop for flow s. By convention, we set $Q_{s,H(s)+1}(t) = 0$, and then we have $P_{s,H(s)+1}(t) = P_{s,H(s)}(t)$. The queue length evolves according to the following equations:

$$Q_{s,k}(t+1) = Q_{s,k}(t) + \Psi_{s,k-1}(t) - \Psi_{s,k}(t), \tag{3}$$

where we set $\Psi_{s,0}(t) = A_s(t)$.

Let $F_s(t)$ be the total number of packets that arrive at the source node of flow s until time slot $t \ge 0$, including those present at time slot 0, and let $\hat{F}_{s,k}(t)$ be the total number of packets that are served at $Q_{s,k}$ until time slot $t \ge 0$. By convention, we set $\hat{F}_{s,k}(0) = 0$ for all link-flow-pairs $(s,k) \in \mathcal{P}$. We let $Z_{s,k,i}(t)$ denote the sojourn time of the i-th packet of $Q_{s,k}$ in the network at time slot t, where the time is measured from the time when the packet arrives in the network (i.e., when the packet arrives at the source node), and let $W_{s,k}(t) = Z_{s,k,1}(t)$ denote the sojourn time of the HOL packet of $Q_{s,k}$ in the network at time slot t. We set $W_{s,0}(t) = 0$ for all $s \in \mathcal{S}$. Further, if $Q_{s,k}(t) = 0$, we set $W_{s,k}(t) = W_{s,k-1}(t)$. Letting $U_{s,k}(t) \triangleq t - W_{s,k}(t)$ denote the time when the HOL packet of $Q_{s,k}$ arrives in the network, we have that

$$U_{s,k}(t) = \inf\{\tau \le t \mid F_s(\tau) > \hat{F}_{s,k}(t)\}, \text{ for all } t \ge 0. \tag{4}$$

As in [27], a discrete-time queueing system is said to be *stable*, if the underlying Markov chain is *positive Harris recurrent*. When the state space is countable and all states communicate (as in the system that we consider in this paper), this is equivalent to the Markov chain being *positive recurrent*. The *throughput region* of a scheduling policy is defined as the set of arrival rate vectors for which the network remains stable under this policy. Further, the *optimal throughput region* (or *stability region*) is defined as the union of the throughput regions of all possible scheduling policies. We let Λ^* denote the optimal throughput region, which can be represented as

$$\Lambda^* \triangleq \{\vec{\lambda} \mid \exists \vec{\phi} \in Co(\mathcal{M}_{\mathcal{P}}) \text{ s.t. } \lambda_s \le \phi_{s,k}, \forall (s,k) \in \mathcal{P}\}. \tag{5}$$

An arrival rate vector is strictly inside Λ^*, if the inequalities above are all strict.

We summarize the notations in Appendix A for quick reference.

III. Queue-length-based Back-Pressure Algorithm

It has been shown in [1] that Q-BP stabilizes the network for any feasible arrival rate vector using stochastic Lyapunov techniques. Specifically, we can use a quadratic Lyapunov function to show that the function has a negative drift under Q-BP when queue lengths are large enough. In this section, we re-visit throughput-optimality of Q-BP using fluid limit techniques. The analysis will be extended later to prove throughput-optimality of the delay-based back-pressure algorithm.

To begin with, we define the *queue differential* $\Delta Q_{s,k}(t)$ as

$$\Delta Q_{s,k}(t) \triangleq Q_{s,k}(t) - Q_{s,k+1}(t), \tag{6}$$

and specify the back-pressure algorithm based on queue lengths as follows.

Queue-length-based Back-Pressure (Q-BP) algorithm:

$$\vec{M}^* \in \operatorname{argmax}_{\vec{M}\in\mathcal{M}_{\mathcal{P}}} \sum_{(s,k)\in\mathcal{P}} \Delta Q_{s,k}(t) \cdot M_{s,k}. \tag{7}$$

The algorithm needs to solve a MaxWeight problem with weights as queue differentials, and ties can be broken arbitrarily if there is more than one schedule that has the largest weight sum.

We establish the fluid limits of the system in the following subsection.

A. Fluid Limits

We define the process describing the behavior of the underlying system as $\mathcal{X} = (\mathcal{X}(t), t = 0, 1, 2, \cdots)$, where

$$\mathcal{X}(t) \triangleq \left((Z_{s,k,1}(t), \cdots, Z_{s,k,Q_{s,k}(t)}(t)), (s,k) \in \mathcal{P}\right).$$

We define the norm of $\mathcal{X}(t)$ as

$$\|\mathcal{X}(t)\| \triangleq \|\vec{Q}(t)\| + \|\vec{W}(t)\|. \tag{8}$$

Clearly, under Q-BP, the evolution of $\mathcal{X}$ forms a discrete-time Markov chain with countable state space. Let $\mathcal{X}^{(x)}$ denote a process $\mathcal{X}$ with an initial configuration such that

$$\|\mathcal{X}^{(x)}(0)\| = x. \tag{9}$$

The following Lemma was derived in [28] for continuous-time countable Markov chains, and it follows from more general results in [29] for discrete-time countable Markov chains.

Lemma 1 (Theorem 4 of [12]): Suppose there exist an $\epsilon > 0$ and a finite integer $T > 0$ such that for any sequence of processes $\{\frac{1}{x}\mathcal{X}^{(x)}(xT), x = 1, 2, \cdots\}$, we have

$$\limsup_{x\to\infty} \mathbf{E}\left[\frac{1}{x}\|\mathcal{X}^{(x)}(xT)\|\right] \le 1 - \epsilon. \tag{10}$$

Then, the Markov process $\mathcal{X}$ is positive recurrent.

A stability criteria of (10) leads to a fluid limit approach [30], [31] to the stability problem of queueing systems. Hence, we start our analysis by establishing the *fluid limit model* as in [12], [30]. We define the process $\mathcal{Y} \triangleq \left(A, F, \hat{F}, Q, P, \Pi, \Psi, W, U\right)$, and it is clear that a sample path of $\mathcal{Y}^{(x)}$ uniquely defines the sample path of $\mathcal{X}^{(x)}$. Then we extend the definition of $Y = A, F, \hat{F}, Q, P, \Pi, \Psi, W$ and U to continuous time domain as $Y(t) \triangleq Y(\lfloor t \rfloor)$ for each continuous time $t \geq 0$.

As in [12], we extend the definition of $F_s^{(x)}(t)$ to the negative interval $t \in [-x, 0)$ by assuming that the packets present in the initial state $\mathcal{X}^{(x)}(0)$ arrived in the past at some of the time instants $-(x-1), -(x-2), \cdots, 0$, according to their delays in the state $\mathcal{X}^{(x)}(0)$. By this convention, we have $F_s^{(x)}(-x) = 0$ for all $s \in \mathcal{S}$ and x, and $\sum_{s \in S} F_s^{(x)}(0) \leq x$ for all x.

Then, applying the techniques used in the proof for Theorem 4.1 of [30] or Lemma 1 of [12], we can show that with probability one, for any sequence of processes $\{\frac{1}{x_n}\mathcal{Y}^{(x_n)}(x_n \cdot)\}$, where $\{x_n\}$ is a sequence of positive integers with $x_n \to \infty$, there exists a subsequence $\{x_{n_j}\}$ with $x_{n_j} \to \infty$ as $j \to \infty$ such that the following convergences hold *uniformly over compact (u.o.c.)* intervals:

$$\frac{1}{x_{n_j}} \int_0^{x_{n_j} t} A_s^{(x_{n_j})}(\tau) d\tau \to \lambda_s t, \tag{11}$$

$$\frac{1}{x_{n_j}} F_s^{(x_{n_j})}(x_{n_j} t) \to f_s(t), \tag{12}$$

$$\frac{1}{x_{n_j}} \hat{F}_{s,k}^{(x_{n_j})}(x_{n_j} t) \to \hat{f}_{s,k}(t), \tag{13}$$

$$\frac{1}{x_{n_j}} Q_{s,k}^{(x_{n_j})}(x_{n_j} t) \to q_{s,k}(t), \tag{14}$$

$$\frac{1}{x_{n_j}} P_{s,k}^{(x_{n_j})}(x_{n_j} t) \to p_{s,k}(t), \tag{15}$$

$$\frac{1}{x_{n_j}} \int_0^{x_{n_j} t} \Pi_{s,k}^{(x_{n_j})}(\tau) d\tau \to \int_0^t \pi_{s,k}(\tau) d\tau, \tag{16}$$

$$\frac{1}{x_{n_j}} \int_0^{x_{n_j} t} \Psi_{s,k}^{(x_{n_j})}(\tau) d\tau \to \int_0^t \psi_{s,k}(\tau) d\tau. \tag{17}$$

Similarly, the following convergences (which are denoted by "$\Rightarrow$") hold at every continuous point of the limit function:

$$\frac{1}{x_{n_j}} W_{s,k}^{(x_{n_j})}(x_{n_j} t) \Rightarrow w_{s,k}(t), \tag{18}$$

$$\frac{1}{x_{n_j}} U_{s,k}^{(x_{n_j})}(x_{n_j} t) \Rightarrow u_{s,k}(t). \tag{19}$$

The above convergence properties follow directly from the Arzela-Ascoli Theorem and the structure of the model: that the arrival process satisfies the SLLN and that the sequence of the (scaled) departure process is uniformly bounded and uniformly equicontinuous.

Any set of limiting functions $(f, \hat{f}, q, p, \pi, \psi, w, u)$ is called a *fluid limit*. The family of these fluid limits is associated with our original stochastic network. The scaled sequences $\{\frac{1}{x_n}\mathcal{Y}^{(x_n)}(x_n \cdot)\}$ and their limits are referred to as a *fluid limit model* [27]. Since some of the limiting functions, namely $f_s, \hat{f}_{s,k}, q_{s,k}, p_{s,k}$ are Lipschitz continuous in $[0, \infty)$, they are absolutely continuous. Hence, at almost all points $t \in [0, \infty)$, the derivatives of these limiting functions exist. We call such points *regular* time.

We then present the *fluid model equations* of the system as follows.

$$\textstyle\sum_{s \in \mathcal{S}} f_s(0) \leq 1, \tag{20}$$

$$p_{s,k}(t) = \textstyle\sum_{i=1}^{k} q_{s,i}(t), \tag{21}$$

$$p_{s,k}(t) = f_s(t) - \hat{f}_{s,k}(t), \tag{22}$$

$$f_s(t) = f_s(0) + \lambda_s t, \tag{23}$$

$$u_{s,k}(t) = t - w_{s,k}(t), \tag{24}$$

$$\psi_{s,k}(t) \leq \pi_{s,k}(t), \tag{25}$$

$$\Delta q_{s,k}(t) = q_{s,k}(t) - q_{s,k+1}(t), \tag{26}$$

$$\frac{d}{dt} q_{s,k}(t) = \begin{cases} \psi_{s,k-1}(t) - \pi_{s,k}(t), & \text{if } q_{s,k}(t) > 0, \\ (\psi_{s,k-1}(t) - \pi_{s,k}(t))^+, & \text{otherwise,} \end{cases} \tag{27}$$

where $(z)^+ \triangleq \max(z, 0)$, and we set $\psi_{s,0} = \pi_{s,0} = \lambda_s$. Fluid model equations can be thought of as belonging to a fluid network which is the deterministic equivalence of the original stochastic network. Any set of functions satisfying the fluid model equations is called a *fluid model solution* of the system. It is easy to check that any fluid limit is a fluid model solution.

It is clear from (7) that Q-BP will not schedule link-flow-pair (s, k) if $Q_{s,k}(t) - Q_{s,k+1}(t) < 0$. Hence, if link-flow-pair (s, k) is scheduled, it must satisfy that $Q_{s,k}(t) - Q_{s,k+1}(t) \geq 0$. Moreover, the length of queue $Q_{s,k}$ can decrease by at most one within one time slot, and the length of queue $Q_{s,k+1}$ can increase by at most one within one time slot, due to the assumption of unit link capacity (a similar argument also holds with non-unit link rates). This implies that, if

$$Q_{s,k}(t) \geq Q_{s,k+1}(t) - 2 \tag{28}$$

initially holds for all (s, k) at time slot 0, then the inequality holds for every time slot $t \geq 0$. This further implies that

$$q_{s,k}(t) \geq q_{s,k+1}(t), \text{ i.e., } \Delta q_{s,k}(t) \geq 0, \tag{29}$$

for all (scaled) time $t \geq 0$, from the convergence of (14). We assume that at time slot 0, all queues on the route of each flow are empty except for the first queue, then it follows that (28) holds for all (scaled) time $t \geq 0$, and thus, $\Delta q_{s,k}(t) \geq 0$ holds for all time $t \geq 0$.

Remark: Note that we make the assumption of empty queues for ease of analysis. Even without this assumption, we can show that there exists a finite time $T > 0$ such that for all time $t \geq T$, (29) holds for all $(s, k) \in \mathcal{P}$. This can be proved by induction. The detailed proof can be found in our online technical report [32], but the basic idea is as follows: Consider a flow $\hat{s} \in \mathcal{S}$. We want to show that there exists a finite time $T_{\hat{s}} > 0$ such that for all time $t \geq T_{\hat{s}}$, (29) holds for all $(\hat{s}, k)$ with $k \in \{1, 2, \cdots, H(\hat{s})\}$.

1) First, we show that there exists a finite time $T_{\hat{s},1} > 0$ such that for all time $t \geq T_{\hat{s},1}$, (29) holds for link-flow-pair $(\hat{s}, 1)$. Suppose that (29) does not hold for $(\hat{s}, 1)$. Then Q-BP does not schedule $(\hat{s}, 1)$, i.e., $q_{\hat{s},1}(t)$ does not decrease and $q_{\hat{s},2}(t)$ does not increase. On the other hand, due to the exogenous arrivals at the source node of flow $\hat{s}$, $q_{\hat{s},1}(t)$ must increase with time. Hence, there must exist a finite time $T_{\hat{s},1}$ such that (29) holds for $(\hat{s}, 1)$ at time

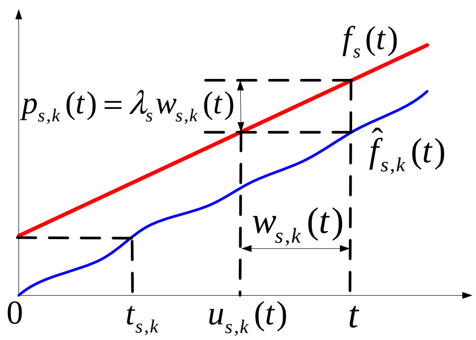


Fig. 1. Linear relation between queue lengths and delays in the fluid limits.

$T_{\hat{s},1}$. We can further show that (29) holds for all $t \geq T_{\hat{s},1}$ under Q-BP. This can be proved by contradiction.

2) Then, we discuss the induction step: Consider $k \in \{1, 2, \cdots, H(\hat{s})-1\}$. Suppose that for all time $t \geq T_{\hat{s},k} > 0$, (29) holds for $(\hat{s}, j)$ and for all $j \in \{1, 2, \cdots, k\}$, we show that there exists a finite time $T_{\hat{s},k+1} \geq T_{\hat{s},k}$ such that for all time $t \geq T_{\hat{s},k+1}$, (29) holds for $(\hat{s}, j')$ and for all $j' \in \{1, 2, \cdots, k+1\}$. For simplicity, we consider the case for which $k = 1$, and the general induction step follows similarly. Now, suppose that (29) does not hold for $(\hat{s}, 2)$, and we prove it by contradiction. Clearly, Q-BP will schedule only link-flow-pairs for which (29) holds (i.e., link-flow-pair $(\hat{s}, 1)$ in this case). Hence, the fluid limit model of the subsystem that consists of link-flow-pairs for which (29) holds must be stable, from the throughput-optimality of Q-BP (see Proposition 2). This, in particular, implies that $q_{\hat{s},1}$ is stable, which further implies that $q_{\hat{s},2}(t)$ must increase with time, because Q-BP keeps forwarding packets from $q_{\hat{s},1}$ to $q_{\hat{s},2}$ while not serving $q_{\hat{s},2}$. Hence, there must exist a finite time $T_{\hat{s},2} \geq T_{\hat{s},1}$ such that for all time $t \geq T_{\hat{s},2}$, (29) holds for $(\hat{s}, 2)$.

Hence, letting $T_{\hat{s}} \triangleq T_{\hat{s},H(\hat{s})}$, we have that for all time $t \geq T_{\hat{s}}$, (29) holds for all $(\hat{s}, k)$ with $k \in \{1, 2, \cdots, H(\hat{s})\}$. Since the above arguments can be applied to any flow $\hat{s} \in \mathcal{S}$, we can complete the proof by setting $T \triangleq \max_{\hat{s}\in\mathcal{S}} T_{\hat{s}}$.

B. Throughput-Optimality of Q-BP

Proposition 2: Q-BP can support any traffic with arrival rate vector that is strictly inside Λ^*.

Before giving the proof of Proposition 2, in the following lemma, we present a linear relation between cumulative queue length $p_{s,k}(t)$ and waiting time $w_{s,k}(t)$, which is used for proving Proposition 2.

Lemma 3: For any fixed $t_{s,k} > 0$, the two conditions $u_{s,k}(t_{s,k}) > 0$ and $\hat{f}_{s,k}(t_{s,k}) > f_s(0)$ are equivalent for every link-flow-pair $(s, k) \in \mathcal{P}$. Further, if the conditions hold, we have

$$p_{s,k}(t) = \lambda_s w_{s,k}(t), \tag{30}$$

for all $t \geq t_{s,k}$, with probability one.

Fig. 1 describes the relations between the variables.

Proof: Since the first part, i.e., that the two conditions are equivalent, is straightforward from the definition of fluid limits and (4), we focus on the second part, i.e., if $\hat{f}_{s,k}(t_{s,k}) > f_s(0)$, then (30) follow.

Suppose that $\hat{f}_{s,k}(t_{s,k}) > f_s(0)$. Then, by the definition of $u_{s,k}(t)$, we have $\hat{f}_{s,k}(t) = f_s(u_{s,k}(t))$, for all $t \geq t_{s,k}$. From (22), (23) and (24), we obtain that

$$\begin{aligned} p_{s,k}(t) &= f_s(t) - \hat{f}_{s,k}(t) \\ &= (f_s(0) + \lambda_s t) - (f_s(0) + \lambda_s u_{s,k}(t)) \\ &= \lambda_s \cdot (t - u_{s,k}(t)) \\ &= \lambda_s w_{s,k}(t). \end{aligned}$$

■

Proof of Proposition 2: We prove stability using standard Lyapunov techniques Let $V(\vec{q}(t))$ denote the Lyapunov function defined as

$$V(\vec{q}(t)) \triangleq \tfrac{1}{2} \textstyle\sum_{(s,k)\in\mathcal{P}} \left(q_{s,k}(t)\right)^2. \tag{31}$$

From the results of Lemmas 1 and 3, to show positive recurrence, we only need to prove that for any $\zeta > 0$, there exists a finite time $T_1 > 0$ such that for any fluid limit with $\|\vec{q}(0)\| \leq 1$, we have

$$\|\vec{q}(t)\| \leq \zeta, \tag{32}$$

for all time $t \geq T_1$. To show the above, it is sufficient to show that for any $\zeta_1 > 0$, there exists $\zeta_2 > 0$ such that $V(\vec{q}(t)) \geq \zeta_1$ implies $\frac{D^+}{dt^+} V(\vec{q}(t)) \leq -\zeta_2$ for any regular time $t \geq 0$, where $\frac{D^+}{dt^+} V(\vec{q}(t)) = \lim_{\delta\downarrow 0} \frac{V(\vec{q}(t+\delta)) - V(\vec{q}(t))}{\delta}$.

Suppose $\vec{\lambda}$ is strictly inside Λ^*, then there exists a vector $\vec{\phi} \in Co(\mathcal{M}_{\mathcal{P}})$ such that $\vec{\lambda} < \vec{\phi}$, i.e., $\lambda_s < \phi_{s,k}$ for all $(s, k) \in \mathcal{P}$. Since $\vec{q}(t)$ is differentiable, then for any regular time $t \geq 0$, we can obtain the derivative of $V(\vec{q}(t))$ as

$$\begin{aligned} &\tfrac{D^+}{dt^+} V(\vec{q}(t)) \\ &\overset{(a)}{=} \textstyle\sum_{(s,k)\in\mathcal{P}} q_{s,k}(t) \cdot (\psi_{s,k-1}(t) - \pi_{s,k}(t)) \\ &\overset{(b)}{\leq} \textstyle\sum_{(s,k)\in\mathcal{P}} q_{s,k}(t) \cdot (\pi_{s,k-1}(t) - \pi_{s,k}(t)) \\ &= \textstyle\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot \lambda_s \\ &\quad - \textstyle\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot \pi_{s,k}(t) \\ &= \textstyle\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot (\lambda_s - \phi_{s,k}) \\ &\quad + \textstyle\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot (\phi_{s,k} - \pi_{s,k}(t)), \end{aligned} \tag{33}$$

where (a) and (b) are from (27) and (25), respectively.

Note that $q_{s,k}(t) \leq H^{\max} \max_{(r,j)\in\mathcal{P}} \Delta q_{r,j}(t)$, for any $(s, k) \in \mathcal{P}$. Hence, we have $V(\vec{q}(t)) \leq \frac{1}{2}|S|H^{\max}(H^{\max} \max_{(r,j)\in\mathcal{P}} \Delta q_{r,j}(t))^2$. Let us choose $\zeta_3 = \sqrt{\frac{2\zeta_1}{|S|(H^{\max})^3}}$, then $V(\vec{q}(t)) \geq \zeta_1$ implies $\max_{(r,j)\in\mathcal{P}} \Delta q_{r,j}(t) \geq \zeta_3$. Since $\vec{\lambda} < \vec{\phi}(t)$ and $\Delta q_{s,k}(t) \geq 0$ for all $(s, k) \in \mathcal{P}$, then in the final result of (33), we can conclude that the first term is bounded as follows:

$$\begin{aligned} \textstyle\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot (\lambda_s - \phi_{s,k}) &\leq -\zeta_3 \min_{s,k}(\phi_{s,k} - \lambda_s) \\ &\triangleq -\zeta_2 < 0, \end{aligned}$$

and that the second term becomes non-positive due to the following. Since Q-BP chooses schedules that maximize the queue differential weight sum (7), then we have that

$$\vec{\pi}(t) \in \mathrm{argmax}_{\vec{\phi} \in Co(\mathcal{M}_{\mathcal{P}})} \sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot \phi_{s,k},$$

which implies that

$$\sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot \phi_{s,k} \leq \sum_{(s,k)\in\mathcal{P}} \Delta q_{s,k}(t) \cdot \pi_{s,k}(t),$$

for all $\vec{\phi} \in Co(\mathcal{M}_{\mathcal{P}})$. Therefore, this shows that $V(\vec{q}(t)) \geq \zeta_1$ implies $\frac{D^+}{dt^+}V(\vec{q}(t)) \leq -\zeta_2$. Then, it immediately follows that for any $\zeta > 0$, there exists a finite time $T_1 > 0$ such that for any fluid limit with $\|\vec{q}(0)\| \leq 1$, we have $\|\vec{q}(t)\| \leq \zeta$ for any time $t \geq T_1$. Also, we have

$$p_{s,k}(t) \leq \|\vec{q}(t)\| \leq \zeta, \tag{34}$$

for all $(s,k) \in \mathcal{P}$. Let us choose T_1 large enough, then it follows from (20), (22) and (34) that

$$\hat{f}_{s,k}(T_1) = f_s(T_1) - p_{s,k}(T_1) > f_s(0),$$

for all $(s,k) \in \mathcal{P}$ and for any time $t \geq T_1$. Hence, we have (30) from Lemma 3, and thus, we have

$$\begin{aligned} \|\vec{q}(t)\| + \|\vec{w}\| &\overset{(a)}{\leq} \|\vec{q}(t)\| + \frac{1}{\min_s \lambda_s}\|\vec{p}(t)\| \\ &\overset{(b)}{\leq} \left(1 + \frac{|S|H^{\max}}{\min_s \lambda_s}\right)\zeta \\ &\triangleq \epsilon_1, \end{aligned}$$

where (a) and (b) are from (30) and (34), respectively. We can make ϵ_1 arbitrarily small by choosing small enough ζ.

Now, consider any fixed sequence of processes $\{\frac{1}{x}\mathcal{X}^{(x)}(xt), x = 1, 2, \cdots\}$ (for simplicity also denoted by $\{x\}$). Hence, for any fixed $\epsilon_1 > 0$, we can always choose a large enough integer $T > 0$ such that for any subsequence $\{x_n\}$ of $\{x\}$, there exists a further (sub)subsequence $\{x_{n_j}\}$ such that

$$\lim_{j\to\infty} \frac{1}{x_{n_j}} \|\mathcal{X}^{(x_{n_j})}(x_{n_j}T)\| = \|\vec{q}(T)\| + \|\vec{w}(T)\| \leq \epsilon_1$$

almost surely. This in turn implies (for small enough ϵ_1) that

$$\limsup_{x\to\infty} \frac{1}{x}\|\mathcal{X}^{(x)}(xT)\| \leq \epsilon_1 \triangleq 1 - \epsilon < 1 \tag{35}$$

almost surely. This is because there must exist a subsequence of $\{x\}$ that converges to the same limit as $\limsup_{x\to\infty} \frac{1}{x}\|\mathcal{X}^{(x)}(xT)\|$.

One can readily show that the sequence $\{\frac{1}{x}\|\mathcal{X}^{(x)}(xT)\|, x = 1, 2, \cdots\}$ is uniformly integrable using standard techniques by invoking the Dominated Convergence Theorem and so the details are omitted here. Then, the almost sure convergence in (35) along with uniform integrability implies the following convergence in the mean:

$$\limsup_{x\to\infty} \mathbf{E}[\tfrac{1}{x}\|\mathcal{X}^{(x)}(xT)\|] \leq 1 - \epsilon.$$

Since the above convergence holds for any sequence of processes $\{\frac{1}{x}\|\mathcal{X}^{(x)}(x\cdot), x = 1, 2, \cdots\}$, the condition of (10) in Lemma 1 is satisfied. This completes the proof. ■

IV. Delay-based Back-Pressure Algorithm

A. Algorithm Description

In this section, we develop the Delay-based Back-Pressure (D-BP) policy, and in Section IV-B, we prove that it is throughput optimal. A similar delay-based approach has appeared first in [12] for single-hop networks. However, as mentioned earlier, when packets travel multiple hops before leaving the system, the analytical approach in [12] (i.e., using HOL delay in the queue as the metric) cannot capture queueing dynamics of multihop traffic and the resultant solutions cannot guarantee the linear relation. We will carefully design link weights using a new delay metric, and re-establish the linear relation between queue lengths and delays in the fluid limits for multihop traffic.

Recall that $W_{s,k}(t)$ denotes the sojourn time of the HOL packet of queue $Q_{s,k}(t)$ in the network, where the time is measured from the time when the packet arrives in the network. We define the delay metric $\hat{W}_{s,k}(t)$ as

$$\hat{W}_{s,k}(t) \triangleq W_{s,k}(t) - W_{s,k-1}(t), \tag{36}$$

and also define *delay differential* as

$$\Delta\hat{W}_{s,k}(t) \triangleq \hat{W}_{s,k}(t) - \hat{W}_{s,k+1}(t). \tag{37}$$

The relations between these delay metrics are illustrated in Fig. 2. We specify the back-pressure algorithm with the new delay metric as follows.

Delay-based Back-Pressure (D-BP) algorithm:

$$\vec{M}^* \in \mathrm{argmax}_{\vec{M} \in \mathcal{M}_{\mathcal{P}}} \sum_{(s,k)\in\mathcal{P}} \Delta\hat{W}_{s,k}(t) \cdot M_{s,k}. \tag{38}$$

D-BP computes the weight of (s,k) as the delay differential $\Delta\hat{W}_{s,k}(t)$ and solves the MaxWeight problem, i.e., finds a set of non-interfering link-flow-pairs that maximizes weight sum. Ties can be broken arbitrarily if there is more than one schedule that has the largest weight sum. An intuitive interpretation of the new delay metric $\hat{W}_{s,k}(t)$ is as follows. Note that the queue length $Q_{s,k}(t)$ is roughly the number of packets arriving at the source node of flow s during the time slots between $[U_{s,k}(t), U_{s,k}(t)+\hat{W}_{s,k}(t))$, and from the SLLN, $Q_{s,k}(t)$ is on the order of $\lambda_s\hat{W}_{s,k}(t)$ when $\hat{W}_{s,k}(t)$ is large. Hence, a large $\hat{W}_{s,k}(t)$ implies a large queue length $Q_{s,k}(t)$, and similarly, a large delay differential $\Delta\hat{W}_{s,k}(t)$ implies a large queue length differential $\Delta Q_{s,k}(t)$. Therefore, being favorable to the delay weight sum in (38) is in some sense "equivalent" to being favorable to the queue length weight sum in (7) as Q-BP. We later formally establish the linear relation between the fluid limits of queue lengths and delays in Section IV-B.

We highlight here that the last packet problem can be solved by the D-BP scheme using our proposed delay metric. Let us focus on the source nodes first. Suppose that at the source node of flow s, there are a finite number of packets waiting to be transmitted and there are no further packet arrivals. From the definition of (36) and the fact that $W_{s,0}(t) = 0$, we have $\hat{W}_{s,1}(t) = W_{s,1}(t)$. If some of the packets are stuck at the source node, the delay metric $\hat{W}_{s,1}(t)$ keeps increasing with time. On the other hand, $\hat{W}_{s,2}(t) = W_{s,2}(t) - W_{s,1}(t)$ is equal to the inter-arrival time between two packets and does not increase with time, in particular because some packets at

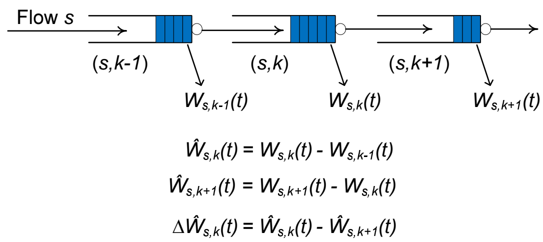


Fig. 2. Delay differentials using new delay metric.

the source node are not served. Hence, the delay differential $\Delta\hat{W}_{s,1}(t) = \hat{W}_{s,1}(t) - \hat{W}_{s,2}(t)$ also increases with time. This implies that under DBP, the increasing delay will eventually "push" all the packets that are waiting at the source node to the second-hop link. After all the packets leave the source node, we can observe similar procedure at the transmitting node of the second-hop link: since $Q_{s,1}(t) = 0$ and $W_{s,1}(t) = 0$, we have $\hat{W}_{s,2}(t) = W_{s,2}(t)$. Repeating the same argument, we can conclude that all the packets will ultimately be "pushed" to the destination node of flow s.

Recall that $U_{s,k}(t)$ denotes the time when the HOL packet of $Q_{s,k}$ arrives in the network (or the source node, rather than the current node). We let $U'_{s,k}(t)$ denote the time when the packet that arrives (in the network or the source node) immediately after the HOL packet of $Q_{s,k}$ arrives in the network. Let $B_{s,k}(t) \triangleq U'_{s,k}(t) - U_{s,k}(t)$ denote the inter-arrival time between the HOL packet of $Q_{s,k}$ and the packet that arrives immediately after it. Clearly, D-BP will not schedule link-flow-pair (s,k) if

$$\hat{W}_{s,k}(t) - \hat{W}_{s,k+1}(t) < 0.$$

Hence, if link-flow-pair (s,k) is scheduled, it must satisfy $\hat{W}_{s,k}(t) - \hat{W}_{s,k+1}(t) \geq 0$. Moreover, the delay $\hat{W}_{s,k}(t)$ can decrease by at most $B_{s,k}(t)$ within one time slot, and the delay $\hat{W}_{s,k+1}(t)$ can increase by at most $B_{s,k}(t)$ within one time slot, due to the assumption of unit link capacity (a similar argument also holds with non-unit link rates). Therefore, if inequality

$$\hat{W}_{s,k}(t) \geq \hat{W}_{s,k+1}(t) - 2B_{s,k}(t) \tag{39}$$

initially holds for all (s,k) at time slot 0, then the inequality holds for all time slot $t \geq 0$. This further leads to

$$\hat{w}_{s,k}(t) \geq \hat{w}_{s,k+1}(t), \text{ i.e., } \Delta\hat{w}_{s,k}(t) \geq 0, \tag{40}$$

for all (scaled) time $t \geq 0$, in the fluid limits, from the convergence of (18) and that $\frac{1}{x_{n_j}} B^{(x_{n_j})}_{s,k}(x_{n_j} t) \to 0$, as $x_{n_j} \to \infty$ (otherwise we will arrive a contradiction with the assumption on the arrival process, i.e., it satisfies the Strong Law of Large Numbers). Recall that we assume that all queues on each route are empty at time slot 0, except for the first queue, then (39) and (40) follow.

B. Throughput-Optimality

The following lemma provides the linear relation between queue lengths and delays in the fluid limits.

Lemma 4: For any fixed $t_{s,k} > 0$, if $\hat{f}_{s,k}(t_{s,k}) > f_s(0)$ for every link-flow-pair $(s,k) \in \mathcal{P}$, then we have

$$q_{s,k}(t) = \lambda_s \hat{w}_{s,k}(t), \tag{41}$$

for all $t \geq t_{s,k}$, with probability one.

Proof: It follows immediately from Lemma 3. ■

We emphasize the importance of (41). Lemma 4 implies that after a finite time (i.e., $\max_{(s,k)\in\mathcal{P}} t_{s,k}$), the queue lengths are λ_s times delays in the fluid limit model. Then the schedules of D-BP are very similar to those of Q-BP, which implies that D-BP achieves the optimal throughput region Λ^*. In the following, we show that the condition of Lemma 4 indeed holds, i.e., such a finite time exists.

Lemma 5: Consider a system under the D-BP policy. Then for $\vec{\lambda}$ strictly inside Λ^*, there exists a finite time $T > 0$ such that the fluid limits satisfy the following property with probability one,

$$\hat{f}_{s,k}(T) > f_s(0), \tag{42}$$

for all link-flow-pairs $(s,k) \in \mathcal{P}$.

We can prove Lemma 5 by induction following the techniques described in Lemma 7 of [12]. The formal proof is provided in Appendix B. We next outline an informal discussion, which highlights the main idea of the proof. First, we consider the base case. D-BP chooses one of the feasible schedules in $\mathcal{M}_{\mathcal{P}}$ (we omit the term "feasible" in the following, whenever there is no confusion) at each time slot. Each schedule receives a fraction of the total time and there must exist a schedule that receives at least $\frac{1}{|\mathcal{M}_{\mathcal{P}}|}$ fraction of the total time. Thus, after a large enough time $T_1 > 0$, there must exist a schedule $\vec{M}^*$ that is chosen for at least $\frac{T_1}{|\mathcal{M}_{\mathcal{P}}|}$ amount of time. The number of initial packets of $\vec{M}^*$ is bounded from (20), thus, for a large enough T_1, all initial "fluid" of at least one link-flow-pair of $\vec{M}^*$ must be completely served, i.e., $\hat{f}_{s,k}(T_1) > f_s(0)$, for at least one (s,k) with $M^*_{s,k} = 1$.

Next, we consider the inductive step. Suppose there exists a $T_l > 0$, such that for at least one subset $S_l \subset \mathcal{P}$ of cardinality l, we have

$$\hat{f}_{s,k}(T_l) > f_s(0), \tag{43}$$

for all $(s,k) \in S_l$. Then there exists $T_{l+1} \geq T_l$ such that

$$\hat{f}_{s,k}(T_{l+1}) > f_s(0), \tag{44}$$

holds for all link-flow-pairs (s,k) within at least one subset $S_{l+1} \subset \mathcal{P}$ of cardinality $l+1$. Since flows travel hop-by-hop, packets that have been served by one link must have been served by the link at the previous hop (of the flow that the packets belong to). Hence, if $(s,k) \in S_l$, we must have $(s,k-1) \in S_l$. Repeating the argument, if $(s,k) \in S_l$, we have $(s,i) \in S_l$ for $1 \leq i \leq k$. Let

$$\begin{aligned} S^*_l \triangleq \{(r,j) \mid\ & (r,j) \notin S_l, (r,j-1) \in S_l, \text{ for } j > 1; \\ & \text{or } (r,j) \notin S_l, \text{ for } j = 1\} \end{aligned} \tag{45}$$

denote the set of link-flow-pairs (r,j) such that $(r,j) \in \mathcal{P} \backslash S_l$ is the closest hop to the source of r. To avoid unnecessary complications, we discuss the induction step for $l = 1$. The generalization for $l > 1$ is straightforward. We show that for given S_1 and T_1, there exists a finite time $T_2 \geq T_1$ such that (44) with T_2 holds for at least two different link-flow-pairs.

Let $(\hat{s}, \hat{k})$ denote the link-flow-pair that satisfies (43) with T_1. Since $(\hat{s}, \hat{k}) \in S_l$ implies $(\hat{s}, i) \in S_l$ for all $1 \leq i \leq \hat{k}$, we must have $\hat{k} = 1$ and $S_1 = \{(\hat{s}, 1)\}$. From (45), we have that

$$S_1^* = \{(r,1) \mid r \in \mathcal{S} \backslash \{\hat{s}\}\} \cup N_{\hat{s}}, \tag{46}$$

where $N_{\hat{s}} = \{(\hat{s}, 2)\}$ if $H(\hat{s}) > 1$, and $N_{\hat{s}} = \emptyset$ if $H(\hat{s}) = 1$. We discuss only the case that $H(\hat{s}) > 1$, and the other case can be easily shown following the same line of analysis. Now suppose that

$$\hat{f}_{r,j}(t) \leq f_r(0), \text{ for all } (r,j) \in \mathcal{P} \backslash S_1, \text{ and all } t \geq 0, \tag{47}$$

i.e., for all the link-flow-pairs except those of S_1, the total amount of service up to time t is no greater than the amount of the initial fluid for all $t \geq 0$. We show that this assumption leads to a contradiction, which completes the induction step.

From the base case and Lemma 4, we have $q_{\hat{s},1}(t) = \lambda_{\hat{s}} \hat{w}_{\hat{s},1}(t)$ for all $t \geq T_1$. We view the subset of link-flow-pairs S_1 as a generalized system, and consider the time slots when there is at least one packet transmission from the outside of S_1, i.e., $(r,j) \in \mathcal{P} \backslash S_1$. For each such time slot, we say that the time slot is *unavailable* to S_1.

1) The number of such unavailable time slots is bounded from the above by x_{n_j}, since at every such time slot, at least one initial packet will be transmitted and the total number of initial packets is bounded by $\|\vec{Q}(0)\| \leq x_{n_j}$ from (9). Hence, the amount of (scaled) time unavailable to S_1 is bounded by $\|\vec{q}(0)\| \leq 1$.
2) Since the amount of (scaled) time unavailable to S_1 is bounded, there exists a sufficiently large $t \geq T_1$ such that the fraction of time that is given to $(r,j) \in \mathcal{P} \backslash S_1$ is negligible, and we must have $\hat{w}_{\hat{r},\hat{j}}(t) = \Theta(1)^2$ and $\Delta \hat{w}_{\hat{r},\hat{j}}(t) = \Theta(1)$ for $(\hat{r}, \hat{j}) \in \mathcal{P} \backslash (S_1 \cup S_1^*)$.
3) Then, we can restrict our focus on the generalized system S_1 to time $t \geq T_1$, and ignore the time that is unavailable to S_1. Then Q-BP and D-BP are in some sense "equivalent" in the generalized system S_1 for $t \geq T_1$ with the following properties: First, Q-BP will stabilize the system if the arrival rate vector is strictly inside Λ^*. Second, since the linear relation (41) holds for all link-flow-pairs in S_1 from Lemma 4, D-BP will schedule links similar to Q-BP and also stabilizes the generalized system S_1.
4) Now let us focus on S_1^*. Link-flow-pairs in S_1^* must have some initial fluid at $t \geq T_1$ because $S_1 \cap S_1^* = \emptyset$. On the other hand, the generalized network S_1 is stable. This implies that the delay metrics of link-flow-pairs in S_1^* should increase on the same order as we increase t, i.e., $\hat{w}_{r^*,j^*}(t) = \Theta(t)$ for $(r^*, j^*) \in S_1^*$. Then we have $\Delta \hat{w}_{r^*,j^*}(t) = \Theta(t)$, since $\hat{w}_{r^*,j^*+1}(t) = \Theta(1)$ from $(r^*, j^*+1) \in \mathcal{P} \backslash (S_1 \cup S_1^*)$ and 2). Since the delay differentials $\Delta \hat{w}_{s,k}(t)$ for all $(s,k) \in S_1$ and $\Delta \hat{w}_{\hat{r},\hat{j}}(t)$ for all $(\hat{r}, \hat{j}) \in \mathcal{P} \backslash (S_1 \cup S_1^*)$ are bounded above from stability of S_1 and 2), respectively, D-BP will choose link-flow-pairs in the set of S_1^* for most of time for a sufficiently large t. This implies that the amount of time unavailable to S_1 is $\Theta(t)$, which contradicts with our previous statement in 1) that the fraction of time that is given to $(r,j) \in \mathcal{P} \backslash S_1$ is negligible.

We provide the detailed proof of Lemma 5 in Appendix B.

We then present throughput-optimality of D-BP in the following proposition.

Proposition 6: D-BP can support any traffic with arrival rate vector that is strictly inside Λ^*.

Proof: We show the stability using fluid limits and standard Lyapunov techniques. From Lemmas 4 and 5, we obtain the key property for proving throughput-optimality of D-BP in Eq. (41), i.e., after a finite time, there is a linear relation between queue lengths and delays in the fluid limit model. We start with the following quadratic-form Lyapunov function,

$$V(\vec{q}(t)) \triangleq \tfrac{1}{2} \textstyle\sum_{(s,k) \in \mathcal{P}} \frac{(q_{s,k}(t))^2}{\lambda_s}. \tag{48}$$

Following the line of analysis in the proof for Proposition 2, we can show that for any $\zeta_1 > 0$, there exist $\zeta_2 > 0$ and a finite time $T > 0$ such that $V(\vec{q}(t)) \geq \zeta_1$ implies $\frac{D^+}{dt^+} V(\vec{q}(t)) \leq -\zeta_2$ for any regular time $t \geq T$, if the underlying scheduler maximizes $\sum_{s,k} \frac{\Delta q_{s,k}(t)}{\lambda_s} \cdot \pi_{s,k}(t)$. Then, by applying the linear relation (41), we can see that D-BP indeed satisfies such a condition, and obtain the results. We omit the detailed proof since it mirrors the derivations in Proposition 2. ∎

V. Greedy Algorithms

It is well known that the schemes (e.g., Q-BP and D-BP) based on the back-pressure techniques are complex to implement because they involve computing a MaxWeight component, which in general is NP-hard [19]. Hence, although D-BP operates efficiently and achieves the optimal throughput region, it could be difficult to implement in practice. Therefore, we are interested in simpler approximations of D-BP that can achieve a guaranteed fraction of the optimal performance. The Delay-based Greedy Maximal Scheduling (D-GMS) algorithm is a good candidate approximation algorithm. A Greedy Maximal Scheduling (GMS) algorithm [23], [26], [33], [34] (which is also known as Longest Queue First (LQF)) operates (in the scenarios with single-hop traffic) as follows: at each time slot t, starts with an empty schedule; first picks a link l with the maximum weight (e.g., queue length or delay); adds l into the schedule, and disables other links that interfere with l; next picks a link l' with the maximum weight from the remaining set of links, adds l' into the schedule, and disables other links that interfere with l'; and continues this process until all links are either chosen or disabled. All chosen links will be scheduled during time slot t. Note that any schedule obtained by GMS is maximal.

GMS has been extensively studied due to its low complexity [23], distributed implementations [35] (or distributed approximations [36]) and empirically observed good performance [22]. It was first shown in [33] that GMS is throughput-optimal in networks where the so-called *local pooling* condition is satisfied. The authors of [21], [34] generalize the idea of *local pooling* to *σ-local pooling*, where σ is a topological notion

[2] We use the standard order notation: $g(n) = o(f(n))$ implies $\lim_{n\to\infty}(g(n)/f(n)) = 0$; and $g(n) = \Theta(f(n))$ implies $c_1 \leq \lim_{n\to\infty}(g(n)/f(n)) \leq c_2$ for some constants c_1 and c_2.

Algorithm 1 Greedy Maximal Scheduling (GMS) Algorithm

procedure GMS($\mathcal{P}, x$)
  $M \leftarrow \emptyset$
  $\mathcal{P}' \leftarrow \mathcal{P}$
  while $\mathcal{P}' \neq \emptyset$ **do**
    pick a link-flow-pair (s, k) with maximum weight: $x(s,k) = \max_{(r,j)\in\mathcal{P}'} x(r,j)$
    $M \leftarrow M \cup \{(s,k)\}$
    $\mathcal{P}' \leftarrow \mathcal{P}' \backslash I(s,k)$
  end while
end procedure

depending on the underlying network topology and is called the *local pooling factor*. There, the authors show that GMS can achieve a σ-fraction of the optimal throughput region. On the other hand, in [37], [38], the *local pooling* condition is generalized to the scenarios with multihop traffic, i.e., GMS is throughput-optimal in networks where the *multihop local-pooling* condition is satisfied. Next, we will discuss the performance limits of D-GMS.

To generalize the GMS algorithm to settings with multihop traffic, we consider link-flow-pairs. We let $x(s,k)$ denote the weight of link-flow-pair $(s,k) \in \mathcal{P}$, and conclude the procedure of GMS in Algorithm 1. We then describe the operations of D-GMS and its queue-length-based counterpart (called Q-GMS) in the following.

Delay-based Greedy Maximal Scheduling (D-GMS) Algorithm: At each time slot t, the algorithm sets the weight of each link-flow-pair to the delay differential, i.e.,

$$x(s,k) \leftarrow \Delta\hat{W}_{s,k}(t), \text{ for all } (s,k) \in \mathcal{P}, \tag{49}$$

and finds its schedule in decreasing order of weight conforming to the underlying interference constraints, by applying Algorithm 1.

Queue-length-based Greedy Maximal Scheduling (Q-GMS) Algorithm: At each time slot t, the algorithm sets the weight of each link-flow-pair to the queue-length differential, i.e.,

$$x(s,k) \leftarrow \Delta Q_{s,k}(t), \text{ for all } (s,k) \in \mathcal{P}, \tag{50}$$

and finds its schedule by applying Algorithm 1.

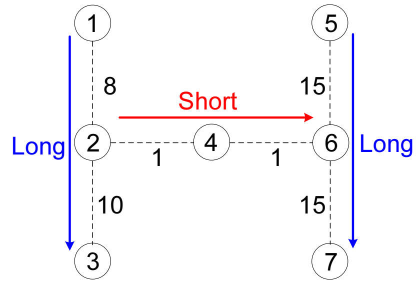
We characterize the throughput performance of D-GMS in the following proposition.

Proposition 7: The achievable throughput region of D-GMS is no smaller than that of Q-GMS.

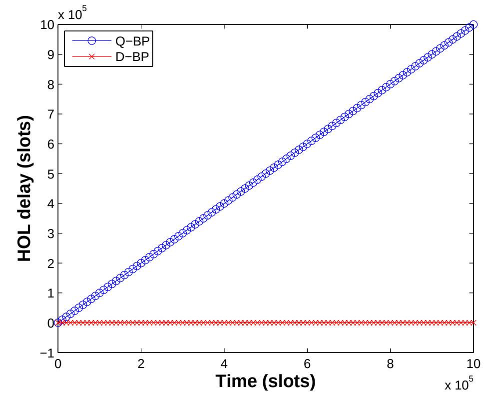
We omit the proof here, since it follows the similar line of analysis for D-BP to establish the linear relation between queue lengths and delays in the fluid limits, and the result can then be obtained by applying the techniques used in [37], [38].

VI. Numerical Results

In this section, we first highlight the *last packet problem* for the queue-length-based back-pressure algorithm. The last packet problem implies that flows that lack packet arrivals at subsequent time may experience excessive delays under Q-BP, which is later confirmed in the simulations. Then, we compare throughput and delay performance of Q-BP and D-BP in a grid network topology under the *2-hop* interference model. Finally, we compare throughput performance of Q-GMS and D-GMS in a size-6 ring network under the *1-hop* interference model.



(a) "H"-type network topology

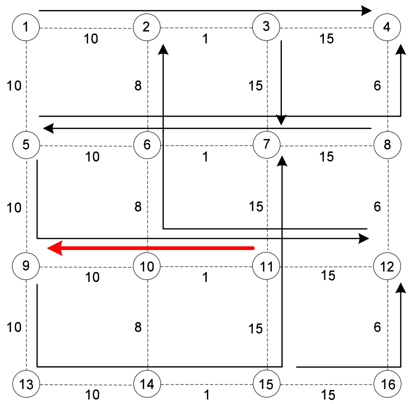


(b) HOL delay of short flow $(2 \to 4 \to 6)$ when $\lambda = 3$

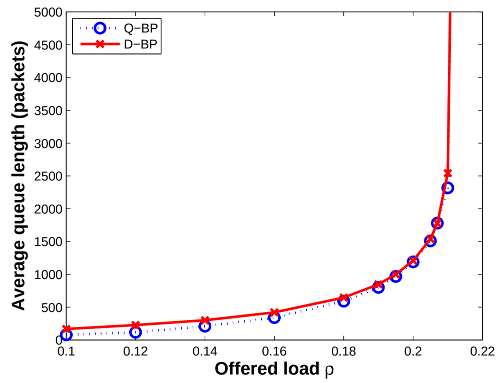
Fig. 3. Illustration of the last packet problem under Q-BP.

We first show the last packet problem of Q-BP through simulations. We observe that several last packets of a short flow (that carry a finite amount of data) may get stuck, which could cause excessive delays. We consider a scenario consisting of 7 nodes and 6 links as shown in Fig. 3(a), where nodes are represented by circles and links are represented by dashed lines with their associated link capacities[3]. We assume a time-slotted system. We establish three flows: one short flow $(2 \to 4 \to 6)$ and two long flows $(1 \to 2 \to 3)$ and $(5 \to 6 \to 7)$. The short flow arrives in the network with 10 packets at time 0. The long flows have an infinite amount of data and keep injecting packets at the source nodes following Poisson distribution with mean rate λ at each time slot. Numerical calculation shows that the feasible rate under the 2-hop interference should satisfy that $\lambda \leq 4.44$. We conduct our simulation for 10^6 time slots, and plot time traces of HOL delay of the short flow when $\lambda = 3$. Fig. 3(b) illustrates the results that the delay increases linearly with time under Q-BP, which implies that several last packets of the short flow are excessively delayed. On the other hand, D-BP succeeds in serving the short flow and keeps the delay close to 0. This also implies that certain flows whose queue lengths do not increase due to lack of future arrivals (or whose inter-arrival times between groups of packets are very large) may experience a large delay under Q-BP, which will be confirmed

[3]Unit of link capacity is packets per time slot.



(a) Grid network topology



(b) Average queue length

Fig. 4. Performance of scheduling algorithms for multihop traffic following Poisson distribution.

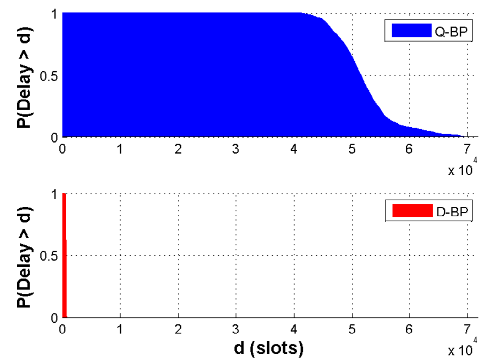


Fig. 5. Delay distribution of the bursty flow under $\rho = 0.2$.

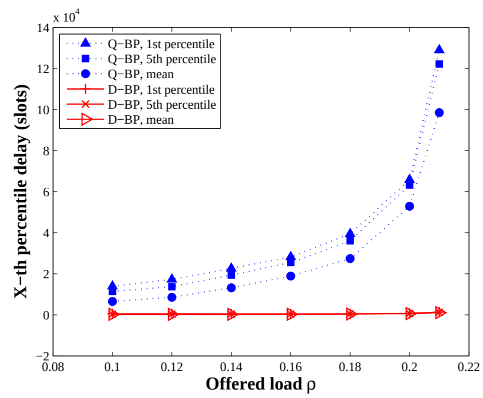


Fig. 6. Mean delay, the 1st and 5th percentile delay of the bursty flow $(11 \to 10 \to 9)$ over offered loads.

in the following simulations.

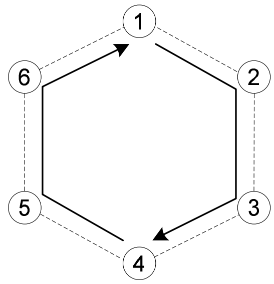
Next, we evaluate the throughput performance of different schedulers in a grid network that consists of 16 nodes and 24 links as shown in Fig. 4(a), where nodes and links are represented by circles and dashed lines, respectively, with link capacity. We establish 9 multihop flows that are represented by arrows. Let $\lambda_1 = 0.1$ and $\lambda_2 = 1$. At each time slot, there is a file arrival with probability $p = 0.01$ for flow $(11 \to 10 \to 9)$ (represented by the red thick arrow in Fig. 4(a)), and the file size follows Poisson distribution with mean rate[4] $\rho\lambda_1/p$. Note that flow $(11 \to 10 \to 9)$ has bursty arrivals with a small mean rate (we simply call it the bursty flow in the following part). All the other 8 flows have packet arrivals following Poisson distribution with mean rate $\rho\lambda_2$ at each time slot. Although these flows share the same stochastic property with an identical mean arrival rate $\rho\lambda_2$, uniform patterns of traffic are avoided by carefully setting the link capacities and placing the flows with different number of hops in an asymmetric manner.

We evaluate the scheduling performance by measuring average total queue lengths in the network over time. Fig. 4(b) illustrates average queue lengths under different offered loads to examine the performance limits of scheduling schemes. Each result represents an average of 10 simulation runs with independent stochastic arrivals, where each run lasts for 10^6 time slots. Since the optimal throughput region is defined as the set of arrival rates under which the queue lengths remain finite, we can consider the traffic load, under which the queue length increases rapidly, as the boundary of the optimal throughput region. Fig. 4(b) shows that D-BP achieves the same throughput region as Q-BP, thus supporting the theoretical results on throughput performance.

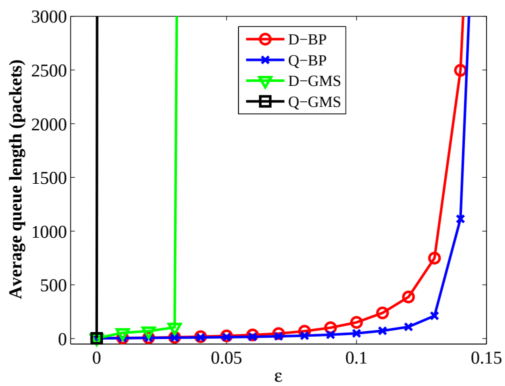
Although Q-BP and D-BP perform similarly in terms of the average queue length (or average delay due to Little's Law) over the network, the tail of the delay distribution of Q-BP could be substantially longer because certain flows are starved. This could cause enormous unfairness between flows, resulting in very poor QoS for certain flows.

Note that although a bursty flow is a long flow that has an infinite amount of data, the arrivals occur in a dispersed manner (i.e., the inter-arrival times between groups of packets are very large) and we can view this bursty flow as consisting of many short flows. Thus, we expect that the bursty flow may experience a very large delay under Q-BP. This is because the bursty flow lacks subsequent packet arrivals over long periods of time, which does not allow the queue-lengths to grow, and thus contributes to the long tail of the delay distribution. However, this phenomenon may not manifest itself in terms

[4]Note that given the network topology, it is hard to find the exact boundary of the optimal throughput region of scheduling policies in a closed form. Hence, we probe the boundary by scaling the amount of traffic. After we choose $\vec{\lambda}$, which determines the direction of traffic load vector, we run our simulations with traffic load $\rho\vec{\lambda}$ changing ρ, which scales the traffic loads.



(a) A size-6 ring network topology



(b) Average queue length

Fig. 7. Performance comparison of Q-BP, D-BP, Q-GMS and D-GMS for multihop traffic under the 1-hop interference model.

of a higher average delay for Q-BP, as can be observed in Fig. 4(b), because the amount of data corresponding to the bursty flow in the simulation is small compared to the other flows. On the other hand, D-BP can achieve better fairness by scheduling the links based on delays and not starving bursty or variable flows. We confirm this in the following observations.

Fig. 5 illustrates the effectiveness of using D-BP over Q-BP in terms of how each scheme affects the delay distribution of bursty flows. We set $\rho = 0.2$. The results show that the tail of the delay distribution under D-BP vanishes much faster than Q-BP. Further, we plot the mean delay, the 1st and 5th percentile delay[5] of the bursty flow over offered loads in Fig. 6. All these delays under D-BP are substantially less than under Q-BP, which implies that D-BP successfully eliminates the excessive packet delays. This confirms that, Q-BP causes a substantially long tail for the delay distribution of the network due to the starvation of the bursty flow, while D-BP overcomes this and achieves better fairness among the flows by scheduling the links based on delays.

Finally, we consider a size-6 ring network topology under the *1-hop* interference model as shown in Fig. 7(a), where links have unit link capacity. We simulate two flows: flow $(1 \to 2 \to 3 \to 4)$ and flow $(4 \to 5 \to 6 \to 1)$. It is known [21] that Q-GMS is not throughput-optimal in this network, as the *local pooling* condition is not satisfied (and thus the *multihop local pooling* is not satisfied from Lemma 7 of [38]). On the other hand, although D-GMS is at least as efficient as Q-GMS, it is not known whether D-GMS can achieve larger throughput in certain scenarios, e.g., in the network in Fig. 7(a).

To see these, we construct a traffic pattern using the idea in [34]. We consider packet arrivals in a frame of 12 time slots. Two flows have the same arrival pattern in each frame. We assume two arrival patterns for each frame. Starting with empty queues at time slot 0, in each frame, the number of exogenous packet arrivals at the source of each flow (i.e., nodes 1 and 4) follows pattern $P_1 = \{1, 0, 5, 0, 1, 0, 5, 0, 1, 0, 5, 0\}$ with probability ϵ, and pattern $P_2 = \{1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0\}$ with probability $(1-\epsilon)$, where $0 \leq \epsilon \leq 1$. The average arrival rate vector is then $\vec{\lambda} = (\frac{18}{12}\epsilon + \frac{4}{12}(1-\epsilon))\mathbf{e} = (\frac{1}{3} + \frac{7}{6}\epsilon)\mathbf{e}$, where $\mathbf{e}$ is a dimension-2 vector with all components equal to 1. It is easy to check that $\vec{\lambda}$ lies strictly inside the optimal throughput region when $0 \leq \epsilon < \frac{1}{7}$, while Q-GMS cannot stabilize the network under such a traffic pattern for all $\epsilon > 0$. Because under Q-GMS, when pattern P_2 occurs in a frame, all the packets arriving in this frame can be completely served and leave the network by the end of this frame, while pattern P_1 occurs, none of the packets arriving in this frame leaves the network by the end of this frame. We evaluate the performance of different scheduling policies under the above traffic pattern. For each policy under a fixed ϵ, we take the average over 10 independent experiments, with each run being 10^7 time slots. In Fig. 7(b), we can see that Q-BP and D-BP have finite average queue length for $0 \leq \epsilon < \frac{1}{7} = 0.143$ and thus achieve the maximum throughput. On the other hand, the average queue length increases linearly with ϵ under Q-GMS and D-GMS starting from $\epsilon = 0$ and $\epsilon = 0.04$, respectively. This implies that neither Q-GMS nor D-GMS is throughput-optimal in this setting, while D-GMS achieves larger throughput ($\epsilon < 0.04$). To fully characterize the performance limits of D-GMS is an interesting yet challenging problem.

VII. Conclusion

In this paper, we developed a throughput-optimal delay-based back-pressure scheduling scheme for multihop wireless networks with fixed routes. We introduced a new delay metric suitable for multihop traffic and established a linear relation between queue lengths and delays in the fluid limits, which plays a key role in the performance analysis and proof of throughput-optimality. Delay-based schemes provide a simple way around the well-known last packet problem that plagues queue-based schedulers, and thus avoid flow starvation. As a result, the excessively long delays that could be experienced by certain flows under queue-based scheduling schemes are eliminated without any loss of throughput. Nonetheless, in this paper, we have only considered the scheduling problem with fixed routes, albeit with multihop flows. The question of whether delay-based schemes under dynamic routing can achieve throughput-optimality is still very much open.

Appendix A
Summary of Notations

[5]Suppose there are N packets sorted by their delays from the largest to the smallest, the X-th percentile delay is defined as the delay of the $\lfloor \frac{NX}{100} \rfloor$-th packet. If $\frac{NX}{100} \leq 1$, it means the maximum delay. For example, if the delays are $[3, 2, 1, 1, 1]$, the 40th percentile delay is 2.

Symbol	Definition
$\mathcal{V}$	set of nodes
$\mathcal{E}$	set of links
$\mathcal{S}$	set of flows
$\mathcal{P}$	set of link-flow-pairs
$\mathcal{M}_{\mathcal{P}}$	set of feasible schedules
$Co(\mathcal{M}_{\mathcal{P}})$	convex hull of $\mathcal{M}_{\mathcal{P}}$
Λ^*	optimal throughput region
$H(s)$	# of hops on the route of flow s
$H^{\max}$	$\max_{s\in\mathcal{S}} H(s)$
$A_s(t)$	# of packet arrivals for flow s at time slot t
λ_s	mean arrival rate for flow s
$F_s(t)$	cumulative # of packet arrivals for flow s up to time slot t
$Q_{s,k}(t)$	queue length of $Q_{s,k}$ at time slot t
$\Pi_{s,k}(t)$	service at $Q_{s,k}$ at time t
$\Psi_{s,k}(t)$	# of packet departures at $Q_{s,k}$ at time slot t
$P_{s,k}(t)$	$\sum_{i=1}^{k} Q_{s,i}(t)$
$\hat{F}_{s,k}(t)$	cumulative # of packets served at $Q_{s,k}$ up to time slot t
$Z_{s,k,i}(t)$	sojourn time (in the network) of the i-th packet of $Q_{s,k}$ at time slot t
$W_{s,k}(t)$	sojourn time (in the network) of the HOL packet of $Q_{s,k}$ at time slot t, i.e., $Z_{s,k,1}(t)$
$U_{s,k}(t)$	time when the HOL packet of $Q_{s,k}$ arrives in the network, i.e., $t-W_{s,k}(t)$
$\hat{W}_{s,k}(t)$	$W_{s,k}(t)-W_{s,k-1}(t)$
$\Delta\hat{W}_{s,k}(t)$	$\hat{W}_{s,k}(t)-\hat{W}_{s,k+1}(t)$
$\Delta Q_{s,k}(t)$	$Q_{s,k}(t)-Q_{s,k+1}(t)$
$B_{s,k}(t)$	inter-arrival time (at the system or the source node) between the HOL packet of $Q_{s,k}$ and the packet that arrives immediately after it

TABLE I
SUMMARY OF NOTATIONS

APPENDIX B
PROOF OF LEMMA 5

Proof: We show that there exists a finite time $T>0$ such that the fluid limits satisfy $\hat{f}_{s,k}(T)>f_s(0)$ for all link-flow-pairs $(s,k)\in\mathcal{P}$. We prove this by induction. We show that there exists a finite time T with at least one link-flow-pair that satisfies the condition, and for a given set of link-flow-pairs satisfying the condition, at least one additional link-flow-pair will satisfy the condition by increasing T.

We first fix an arbitrary $\epsilon_1>0$ and define a constant $K_1 \triangleq \max_s H(s)+\left(\sum_s \lambda_s H(s)\right)\epsilon_1$. In the fluid limit model, we will have

$$f_s(\epsilon_1)=f_s(0)+\lambda_s\epsilon_1>f_s(0),\ \text{for all } s\in\mathcal{S}.$$

Since queue lengths are no greater than the injected amount of data, we have that $p_{s,k}(\epsilon_1)\le f_s(\epsilon_1)$ for all $(s,k)\in\mathcal{P}$, and thus,

$$\begin{aligned}\textstyle\sum_{(s,k)\in\mathcal{P}} p_{s,k}(\epsilon_1) &\le \textstyle\sum_{(s,k)\in\mathcal{P}} f_s(\epsilon_1)\\ &\le \textstyle\sum_s H(s)\left(f_s(0)+\lambda_s\epsilon_1\right)\\ &\le K_1,\end{aligned}\tag{51}$$

where the last inequality is from Eq. (20): $\sum_s f_s(0)\le 1$ and the definition of K_1. Now we show by induction that there exists a finite time T such that

$$\hat{f}_{s,k}(T)>f_s(0),\ \text{for all link-flow-pairs } (s,k).$$

Base Case: There exists $T_1>0$ such that for at least one link-flow-pair (s,k),

$$\hat{f}_{s,k}(T_1)\ge f_s(\epsilon_1).\tag{52}$$

Let $T_1\triangleq\epsilon_1+K_1$. Suppose that (52) does not hold, i.e., there exists at least one packet that arrives before time slot $\lfloor x_{n_j}\epsilon_1\rfloor+1$ and is not served by the end of time slot $\lfloor x_{n_j}T_1\rfloor$. Hence, at each time slot between $[\lfloor x_{n_j}\epsilon_1\rfloor+1, \lfloor x_{n_j}T_1\rfloor]$, there exists at least one schedule that has positive summed weight. Therefore, the schedule determined by D-BP must serve at least one packet in the original system, otherwise the summed weight of the schedule (that does not serve any packet) is zero, which is not the maximum over all the feasible schedules. Hence, we must have

$$\begin{aligned}\textstyle\sum_{(s,k)\in\mathcal{P}}\left(\hat{F}_{s,k}^{(x_{n_j})}(x_{n_j}T_1)-\hat{F}_{s,k}^{(x_{n_j})}(x_{n_j}\epsilon_1)\right)&\\ \ge \lfloor x_{n_j}T_1\rfloor-\lfloor x_{n_j}\epsilon_1\rfloor&,\end{aligned}$$

Dividing both sides of the above inequality by x_{n_j} and letting $x_{n_j}\to\infty$, we obtain

$$\textstyle\sum_{(s,k)\in\mathcal{P}}\left(\hat{f}_{s,k}(T_1)-\hat{f}_{s,k}(\epsilon_1)\right)\ge K_1.$$

Then, from (51), we have

$$\begin{aligned}\textstyle\sum_{(s,k)\in\mathcal{P}}\hat{f}_{s,k}(T_1)&\ge\textstyle\sum_{(s,k)\in\mathcal{P}}\hat{f}_{s,k}(\epsilon_1)+\sum_{(s,k)\in\mathcal{P}}p_{s,k}(\epsilon_1)\\ &=\textstyle\sum_{(s,k)\in\mathcal{P}}f_s(\epsilon_1).\end{aligned}$$

Therefore, $\hat{f}_{s,k}(T_1)\ge f_s(\epsilon_1)$ for at least one link-flow-pair (s,k).

Inductive Step: Suppose that there exist T_l and a subset $S_l\subseteq\mathcal{P}$ such that for all $(s,k)\in S_l$, we have

$$\hat{f}_{s,k}(T_l)\ge f_s(\epsilon_1).\tag{53}$$

Then there exists $T_{l+1}\ge T_l$, where $1\le l<\sum_s H(s)$, and a link-flow-pair $(\tilde{s},\tilde{k})\in\mathcal{P}\backslash S_l$ such that

$$\hat{f}_{\tilde{s},\tilde{k}}(T_{l+1})\ge f_{\tilde{s}}(\epsilon_1).\tag{54}$$

Further we define $S_{l+1}=S_l\cup\{(\tilde{s},\tilde{k})\}$.

We prove the inductive step for $l=1$. The generalization for $l>1$ is straightforward. Hence, we show that for given S_1 and T_1, there exists a finite $T_2>T_1$ such that (54) with T_2 holds for at least two different link-flow-pairs.

Let $(\hat{s},\hat{k})$ denote the link-flow-pair that satisfies (53) with T_1. Then, we have[6] $S_1=\{(\hat{s},1)\}$ and can specify the set S_1^* of link-flow-pairs $(s,k)\in\mathcal{P}\backslash S_1$ that is closest to the source of each flow from (46). We illustrate the case that $H(\hat{s})>1$, and the other case that $H(\hat{s})=1$ can be easily shown following the same line of analysis. Now, we have

$$\hat{f}_{\hat{s},1}(t)\ge f_{\hat{s}}(\epsilon_1),\ \text{for all } t\ge T_1.$$

For all the other link-flow-pairs, we observe that

$$\textstyle\sum_{(r,j)\in\mathcal{P}\backslash S_1}\left(f_r(\epsilon_1)-\hat{f}_{r,j}(T_1)\right)\le K_1.\tag{55}$$

Suppose that for all $t\ge T_1$, we have

$$\hat{f}_{r,j}(t)<f_r(\epsilon_1),\ \text{for all } (r,j)\in\mathcal{P}\backslash S_1.\tag{56}$$

In the following part, we provide a choice of $T_2>T_1$ such that assumption (56) leads to a contradiction, which completes the inductive step, and then the lemma follows by induction.

[6]Note that if $(s,k)\in S_l$, we must have $(s,k-1)\in S_l$. Hence, for $l=1$, we must have the first hop of a flow, i.e., $S_1=(\hat{s},1)$ for some $\hat{s}$.

We view each sample path $X^{(x_{n_j})}(t)$ after time slot $\lceil x_{n_j} T_1 \rceil$ as a generalized system with link-flow-pairs in $S_1 = \{(\hat{s}, 1)\}$. We say that a time slot is *unavailable* to S_1 when a packet from a link-flow-pair $(r, j) \in \mathcal{P} \backslash S_1$ is transmitted during the time slot. Let $h_{S_1}(t)$ denote the (scaled) amount of time unavailable to S_1 during the period of $(T_1, t]$ in the scaled system, for all $t > T_1$. For the scaled generalized system S_1, we obtain from (55) and (56) that

$$h_{S_1}(t) \leq \sum_{(r,j) \in \mathcal{P} \backslash S_1} \left(\hat{f}_{r,j}(t) - \hat{f}_{r,j}(T_1) \right) \leq K_1, \quad (57)$$

for all $t > T_1$. Since the time unavailable to S_1 is bounded, as time t increases, only link-flow-pairs in S_1 will be scheduled, which implies that the weight of link-flow-pairs of $\mathcal{P} \backslash S_1$ becomes negligible. This allows us to focus on S_1. Owing to Lemma 4 and the definition of S_1, the linear relation between queue lengths and delays holds for the link-flow-pair in S_1. Then, it can be easily shown following the same line of analysis of Proposition 6 that link-flow-pairs in S_1 are stable under D-BP[7]. Hence, for all $(s, k) \in S_1$, we have

$$q_{s,k}(t) \leq C_1, \text{ for all } t \geq T_1, \quad (58)$$

and thus

$$\hat{w}_{s,k}(t) \leq \frac{C_1}{\lambda_s}, \text{ for all } t \geq T_1, \quad (59)$$

for some constant C_1, which depends on T_1 and K_1 and does not depend on time t.

Recall that S_1^* denotes the set of link-flow-pairs that is closest to the source of each flow out of S_1 defined in (48). We choose t large enough such that for all $(s, k) \in S_1$ and $(r^*, j^*) \in S_1^*$,

$$\frac{C_1}{\lambda_s} - \left(t - \epsilon_1 - \frac{C_1}{\lambda_s} \right) < \left(t - \epsilon_1 - \frac{C_1}{\lambda_{r^*}} \right) - \epsilon_1. \quad (60)$$

From (56), there are packets that arrive at the source node by time ϵ_1 and have not been served at j-th hop by time t for all $(r, j) \in \mathcal{P} \backslash S_1$, we obtain that

$$t - \epsilon_1 \leq w_{r,j}(t) \leq t, \text{ for all } (r, j) \in \mathcal{P} \backslash S_1. \quad (61)$$

Since $(r^*, j^*), (r^*, j^*+1) \in P \backslash S_1$ for $(r^*, j^*) \in S_1^*$, we have

$$\hat{w}_{r^*, j^*+1}(t) = w_{r^*, j^*+1}(t) - w_{(r^*, j^*)}(t) \leq \epsilon_1, \quad (62)$$

for all $(r^*, j^*) \in S_1^*$. From (59), (61), and that $(r^*, j^* - 1) \in S_1$, we have

$$\hat{w}_{r^*, j^*}(t) \geq t - \epsilon_1 - \frac{C_1}{\lambda_{r^*}}, \quad (63)$$

for all $(r^*, j^*) \in S_1^*$. Then, we have

$$\begin{aligned} \Delta \hat{w}_{s,k}(t) &= \hat{w}_{s,k}(t) - \hat{w}_{s,k+1}(t) \\ &\overset{(a)}{\leq} C_1/\lambda_s - (t - \epsilon_1 - C_1/\lambda_s) \\ &\overset{(b)}{<} (t - \epsilon_1 - C_1/\lambda_{r^*}) - \epsilon_1 \\ &\overset{(c)}{\leq} \hat{w}_{r^*, j^*}(t) - \hat{w}_{r^*, j^*+1}(t) \\ &= \Delta \hat{w}_{r^*, j^*}(t) \end{aligned}$$

[7]Note that since Lemmas 4 and 5 hold for the generalized system S_1, Proposition 6 can be applied to S_1.

for all $(s, k) \in S_1$ and $(r^*, j^*) \in S_1^*$, where (a) is from (59) and (63), (b) is from (60), and (c) is from (63) and (62). Hence, for large t, we have that

$$\Delta \hat{w}_{s,k}(t) < \min_{(r^*, j^*) \in S_1^*} \{ \Delta \hat{w}_{r^*, j^*}(t) \}. \quad (64)$$

Also, from (61), we have that

$$\Delta \hat{w}_{\hat{r}, \hat{j}}(t) \leq \epsilon_1, \quad (65)$$

for all $(\hat{r}, \hat{j}) \in \mathcal{P} \backslash (S_1 \cup S_1^*)$. Since (65) holds for an arbitrarily small ϵ_1 and from (64), D-BP favors link-flow-pairs of S_1^* for all large t. Note that $\Delta \hat{w}_{s,k}(t)$ is bounded for $(s, k) \in S_1$ from (59), and $\Delta \hat{w}_{\hat{r}, \hat{j}}(t)$ is bounded for $(\hat{r}, \hat{j}) \in P \backslash (S_1 \cup S_1^*)$ from (65), and $\Delta \hat{w}_{r*, j*}(t)$ increases linearly on the order of t for $(r^*, j^*) \in S_1^*$ from (63). Hence, there exists a large T_2' such that for all $t > T_2'$, link-flow-pairs in S_1^* will be scheduled at all the time slots between $[\lfloor x_{n_j} T_2' \rfloor + 1, \lfloor x_{n_j} t \rfloor]$ under D-BP. Then, we can choose $T_2 > T_2'$ and have that

$$h_{S_1}(T_2) \geq T_2 - T_2' > K_1.$$

However, this contradicts with (57), which shows that, the assumption (56) is false, and there exists a large T_2 such that

$$\hat{f}_{\tilde{s}, \tilde{k}}(T_2) \geq f_{\tilde{s}}(\epsilon_1), \text{ for at least one } (\tilde{s}, \tilde{k}) \in \mathcal{P} \backslash S_1. \quad (66)$$

In fact, our choice of T_2 depends on the set S_1. However, since there are only a finite number of flows, we can always choose a large enough T_2 so that (66) holds for some $(\tilde{s}, \tilde{k}) \in \mathcal{P} \backslash S_1$. ■

Acknowledgment

The authors would like to thank the associate editor and the reviewers for their valuable comments.

References

[1] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, 1992.

[2] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.

[3] A. Warrier, S. Janakiraman, S. Ha, and I. Rhee, "DiffQ: Practical differential backlog congestion control for wireless networks," in *The 28th IEEE International Conference on Computer Communications (INFOCOM)*, 2009.

[4] A. Sridharan, S. Moeller, and B. Krishnamachari, "Implementing Backpressure-based Rate Control in Wireless Networks," in *Information Theory and Applications Workshop*, 2009.

[5] S. Moeller, A. Sridharan, B. Krishnamachari, and O. Gnawali, "Routing without routes: The backpressure collection protocol," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2010, pp. 279–290.

[6] P. van de Ven, S. Borst, and S. Shneer, "Instability of MaxWeight Scheduling Algorithms," in *The 28th IEEE International Conference on Computer Communications (INFOCOM)*, 2009, pp. 1701–1709.

[7] S. Liu, L. Ying, and R. Srikant, "Throughput-Optimal Opportunistic Scheduling in the Presence of Flow-Level Dynamics," in *The 29th IEEE International Conference on Computer Communications (INFOCOM)*, 2010.

[8] ——, "Scheduling in multichannel wireless networks with flow-level dynamics," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, pp. 191–202, 2010.

[9] A. Mekkittikul and N. McKeown, “A starvation-free algorithm for achieving 100% throughput in an input-queued switch,” in *Proc. of the IEEE International Conference on Communication Networks (ICCCN)*, 1996.

[10] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, “Providing quality of service over a shared wireless link,” *IEEE Communications magazine*, vol. 39, no. 2, pp. 150–154, 2001.

[11] S. Shakkottai and A. Stolyar, “Scheduling for multiple flows sharing a time-varying channel: The exponential rule,” *Translations of the American Mathematical Society-Series 2*, vol. 207, pp. 185–202, 2002.

[12] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting, *Scheduling in a queuing system with asynchronously varying service rates*. Cambridge Univ Press, 2004, vol. 18.

[13] B. Sadiq and G. de Veciana, “Throughput optimality of delay-driven MaxWeight scheduler for a wireless system with flow dynamics,” in *Proceedings of the 47th Annual Conference on Communication, Control and Computing (Allerton)*, 2009.

[14] M. Neely, “Delay-based network utility maximization,” in *The 29th IEEE International Conference on Computer Communications (INFOCOM)*, 2010.

[15] A. Eryilmaz, R. Srikant, and J. Perkins, “Stable scheduling policies for fading wireless channels,” *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, pp. 411–424, 2005.

[16] J. Liu, A. Stolyar, M. Chiang, and H. Poor, “Queue Back-Pressure Random Access in Multihop Wireless Networks: Optimality and Stability,” *IEEE Transactions on Information Theory*, vol. 55, no. 9, pp. 4087 – 4098, Sept. 2009.

[17] L. Bui, R. Srikant, and A. Stolyar, “A Novel Architecture for Reduction of Delay and Queueing Structure Complexity in the Back-Pressure Algorithm,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 6, pp. 1597–1609, 2011.

[18] G. Gupta and N. Shroff, “Delay analysis and optimality of scheduling policies for multihop wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 1, pp. 129–141, 2011.

[19] G. Sharma, R. R. Mazumdar, and N. B. Shroff, “On the complexity of scheduling in wireless networks,” in *Proceedings of the annual international conference on Mobile computing and networking (MobiCom)*. ACM New York, NY, USA, 2006, pp. 227–238.

[20] B. Hajek and G. Sasaki, “Link scheduling in polynomial time,” *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 910–917, 1988.

[21] C. Joo, X. Lin, and N. B. Shroff, “Greedy Maximal Matching: Performance Limits for Arbitrary Network Graphs Under the Node-exclusive Interference Model,” *IEEE Transactions on Automatic Control*, vol. 54, no. 12, pp. 2734–2744, 2009.

[22] C. Joo and N. B. Shroff, “Performance of random access scheduling schemes in multi-hop wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 5, pp. 1481–1493, 2009.

[23] X. Lin and N. B. Shroff, “The impact of imperfect scheduling on cross-Layer congestion control in wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 302–315, 2006.

[24] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar, “Throughput and Fairness Guarantees Through Maximal Scheduling in Wireless Networks,” *Information Theory, IEEE Transactions on*, vol. 54, no. 2, pp. 572–594, 2008.

[25] X. Wu, R. Srikant, and J. Perkins, “Scheduling Efficiency of Distributed Greedy Scheduling Algorithms in Wireless Networks,” *IEEE Transactions on Mobile Computing*, pp. 595–605, 2007.

[26] M. Leconte, J. Ni, and R. Srikant, “Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks,” in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*. New York, NY, USA: ACM, 2009, pp. 165–174.

[27] M. Bramson, “Stability of queueing networks,” *Probability Surveys*, vol. 5, no. 1, pp. 169–345, 2008.

[28] A. Rybko and A. Stolyar, “Ergodicity of stochastic processes describing the operation of open queueing networks,” *Problems of Information Transmission*, vol. 28, pp. 199–220, 1992.

[29] V. Malyshev and M. Menshikov, “Ergodicity, continuity and analyticity of countable Markov chains,” *Transactions of the Moscow Mathematical Society*, vol. 39, pp. 3–48, 1979.

[30] J. Dai, “On positive Harris recurrence of multiclass queueing networks: a unified approach via fluid limit models,” *The Annals of Applied Probability*, pp. 49–77, 1995.

[31] A. Stolyar, “On the stability of multiclass queueing networks: a relaxed sufficient condition via limiting fluid processes,” *Markov Processes and Related Fields*, vol. 1, no. 4, pp. 491–512, 1995.

[32] B. Ji, C. Joo, and N. B. Shroff, “Delay-Based Back-Pressure Scheduling in Multi-Hop Wireless Networks,” *Arxiv preprint arXiv:1011.5674*, November 2012. [Online]. Available: http://arxiv.org/abs/1011.5674

[33] A. Dimakis and J. Walrand, “Sufficient conditions for stability of longest-queue-first scheduling: second-order properties using fluid limits,” *Advances in Applied Probability*, vol. 38, no. 2, p. 505, 2006.

[34] C. Joo, X. Lin, and N. B. Shroff, “Understanding the capacity region of the greedy maximal scheduling algorithm in multihop wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1132–1145, 2009.

[35] J. Hoepman, “Simple distributed weighted matchings,” *Arxiv preprint cs/0410047*, 2004.

[36] C. Joo and N. B. Shroff, “Local Greedy Approximation for Scheduling in Multi-hop Wireless Networks,” *IEEE Transactions on Mobile Computing*, accepted for publication.

[37] A. Brzezinski, G. Zussman, and E. Modiano, “Local pooling conditions for joint routing and scheduling,” in *Information Theory and Applications Workshop, 2008*, 2008, pp. 499–506.

[38] G. Zussman, A. Brzezinski, and E. Modiano, “Multihop Local Pooling for Distributed Throughput Maximization in Wireless Networks,” in *The IEEE International Conference on Computer Communications (INFOCOM)*, April 2008, pp. 1139–1147.

Bo Ji received his B.E. and M.E. degrees in Information Science and Electronic Engineering from Zhejiang University, China in 2004 and 2006, respectively. He is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering at The Ohio State University. His research interests include resource allocation, optimization, cross-layer network control, and low-complexity scheduling algorithm design. He is a student member of the IEEE.

Changhee Joo received his Ph.D degree from Seoul National University, Korea, 2005. He was with the Center of Wireless Systems and Applications, Purdue University, USA, as an associated researcher, and worked at The Ohio State University, USA, as a research scientist. In 2009, he joined Korea University of Technology and Education, Korea, and from 2011, he has been with Ulsan National Institute of Science and Technology (UNIST), Korea. His research interests include resource allocation in wireless networks, cross-layer network optimization, wireless sensor networks, and congestion control. He has served several primary conferences as a committee member, including IEEE INFOCOM, ACM MobiHoc, and IEEE SECON. He is a member of the IEEE, and a recipient of the IEEE INFOCOM 2008 best paper award.

Ness B. Shroff Ness B. Shroff received his Ph.D. degree from Columbia University, NY in 1994 and joined Purdue university immediately thereafter as an Assistant Professor. At Purdue, he became Professor of the school of ECE in 2003 and director of CWSA in 2004, a university-wide center on wireless systems and applications. In July 2007, he joined the ECE and CSE departments at The Ohio State University, where he holds the Ohio Eminent Scholar Chaired Professorship of Networking and Communications. His research interest are in fundamental problems in communication, sensing, social, and cyberphysical networks. Dr. Shroff is a Fellow of the IEEE, an NSF CAREER awardee, and a number of his conference and journals papers have received best paper awards.