Joint TCP Congestion Control and CSMA Scheduling without Message Passing

Xin Wang, *Senior Member, IEEE*, Zhaoquan Li, and Jie Wu, *Fellow, IEEE*

***Abstract*—In this paper, we consider joint congestion control and wireless-link scheduling design for TCP applications over the Internet with ad-hoc wireless links. Differing from the existing methods, the main idea of our approach is to develop non-standard window-based implicit primal-dual solver for the intended optimization problem. Then queueing delays are employed to decompose this solver into local algorithms that can be deployed and operated asynchronously at different layers of network nodes. Capitalizing on this approach, we put forth the QUIC-TCP congestion control algorithm, and a new queueing-delay based carrier sense multiple access (CSMA) scheduling scheme. The proposed schemes can be implemented asynchronously without message passing among network nodes. Confined to the design space of TCP and CSMA, they are readily deployed for practical Internet applications. Moreover, global convergence of the proposed joint design to optimal network equilibrium can be established using the Lyapunov method in an idealized network fluid model. Simulation results are provided to evaluate the proposed schemes in practical networks.**

***Index Terms*—TCP congestion control, CSMA scheduling, convex analysis, network fluid model, Lyapunov function.**

I. Introduction

MOST of the practical network protocols were designed based on sound yet ad-hoc heuristics. Although they have performed reasonably well over the past decades, it is well acknowledged that current protocols need to be redesigned or optimized for emerging wireless Internet applications. In this paper, we consider re-engineering the congestion control and wireless-link scheduling schemes for TCP applications over the Internet with distributed multi-hop (i.e., ad-hoc) wireless links.

A. Related Works

Network protocols are critical, yet difficult to understand and optimize, since they consist of *local* algorithms that are distributed spatially and vertically, and operated asynchronously to accomplish a *global* goal. To this end, a network utility maximization (NUM) paradigm was developed to design and analyze network protocols and schemes via optimization tools [1], [2]. In the NUM framework, joint designs across network layers are developed as the decomposition parts of a global sub-gradient type solver for relevant network optimization problems. In the context of joint congestion control and scheduling schemes for wireless networks, the queue lengths of links usually play the role of Lagrange dual variables. Based on the queue lengths, a source-rate controller at the transport layer decides the amount of traffic injected into the network, and a "MaxWeight" scheduling at the MAC layer fully exploits the wireless-link capacities [3]–[5]. For multi-hop networks, finding the MaxWeight policy is, in general, NP-complete [6]. A number of low-complexity algorithms were proposed to approximate the MaxWeight scheduling [7]–[9]. Even for these suboptimal algorithms, distributed implementation is not trivial without the help of message passing. Recently, a distributed carrier sense multiple access (CSMA) algorithm was proposed for ad-hoc wireless links in [10]–[12]. In an idealized CSMA model, it was shown that this random access scheme can achieve maximal throughput *without* message passing when all packets traverse only one link (i.e., single-hop) before they leave the network. With queue-length exchanges among nodes, a joint design of congestion control and CSMA scheduling was also developed for multi-hop wireless networks in [10].

In the queue-length based joint designs in [10] and [3]–[5] under the classic NUM framework, congestion control is simplified as a source-rate controller. The carefully designed source-rate control and scheduling schemes follow a global gradient-type iteration to solve the NUM problem. Stability/convergence of the schemes can then be readily shown, by drawing from the standard optimization tools. Although this approach is theoretically appealing, the resultant source-rate controllers do not fit well into the TCP design space. Congestion control for almost all ($> 90\%$) Internet traffic is, in fact, performed by the TCP in a window-control manner. Mapping from the direct rate control to such a TCP window-control implementation is non-trivial, since source rate is never a simple, rational function of its window size, even in the simplified network fluid model. Moreover, message passing of queue lengths is required among the network nodes/routers to perform the proposed rate control. Message passing not only introduces overhead, but also undermines the scalability due to its request of network support. As a result, the theoretically appealing NUM schemes in [3]–[5], [10] could be difficult to deploy in practical Internet applications.

Manuscript submitted January 18, 2013; revised April 5, June 29, and August 30, 2013; accepted September 12, 2013. The associate editor coordinating the review of this paper and approving it for publication was D. Tarchi.

Work in this paper was supported by the China Recruitment Program of Global Young Experts, and a grant from FAU Foundation/Tecore Networks. Part of this paper has appeared in the *Proc. of Globecom Conf.*, Anaheim, CA, Dec. 3-7, 2012.

X. Wang is with the State Key Laboratory of ASIC and System, the Dept. of Communication Science and Engineering, Fudan University, 220 Han Dan Road, Shanghai, China, and with the Dept. of Computer & Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431 (e-mail: xwang11@fudan.edu.cn, xwang11@fau.edu).

Z. Li is with the Dept. of Computer & Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431 (e-mail: zli7@fau.edu).

J. Wu is with the Dept. of Computer & Information Sciences, Temple University, 1801 N Broad St., Philadelphia, PA 19122 (e-mail: jiewu@temple.edu).

Digital Object Identifier 10.1109/TWC.2013.100313.130116

Since changes to Internet infrastructure take time and are costly, network engineers and practitioners actually prefer designs that can be used with the current infrastructure. To optimize the TCP for wireless, new heuristics were put forth to improve its packet-loss based additive-increase-multiplicative-decrease (AIMD) window adjustment by increasing window size more aggressively, and/or decreasing it less drastically in HSTCP and STCP [13], [14]. A number of schemes were also developed to make TCP more resilient, by distinguishing between packet loss due to congestion and that due to wireless-link errors in ATCP, TCP-Veno, TCP-Westwood and TCP-Jersey [15]–[17]. Recently, it was increasingly recognized that using packet loss as a congestion measure is insufficient for TCP to perform a stable and efficient congestion control. This motivates the TCP-Vegas, TCP-Westwood, TCP-Veno, and TCP-Jersey that employ (in part) queueing-delay based solutions for congestion control [15], [17]–[21]. Simulations/experiments were used to evaluate these heuristic schemes; yet, they lack systematic design procedures and analytical performance guarantees.

Inspired by the delay based TCP schemes, the Mo-Walrand scheme and FAST-TCP leveraged the practical TCP congestion control to the NUM paradigm [22], [23]. In these schemes, queueing delays (instead of queue lengths) were used to play the role of congestion measures and Lagrange multipliers. Then it was shown that the proposed TCP window-control mechanisms amount to implicit updates of source rates as "primal variables" and queueing delays as "Lagrange dual variables" to solve the target NUM problems from an optimization-theoretic perspective. Stability/convergence of the proposed schemes was either analytically established using a Lyapunov method [22], or demonstrated by extensive simulations and experiments [23], for *wired* networks.

B. Contributions of the Paper

To overcome the limitations of existing approaches, we propose a novel *design-space oriented* cross-layer optimization paradigm. Different from the classic NUM paradigm [3]–[5], [10], we extend the Mo-Walrand approach [22] to develop non-standard *window-based implicit primal-dual solver* solvers to accommodate the TCP design space. Queueing delays are then employed to *decompose* these solvers into local algorithms that can be deployed and operated asynchronously at different layers of distributed network nodes. Unlike the classic gradient-type algorithms, there is no general method available for the design and analysis of these non-standard solvers in the optimization textbooks. Interestingly, we established the existence of such solvers in our recent work [24]. Generalizing the Mo-Walrand scheme, we constructed a class of window-based QUIC-TCP algorithms for congestion control. For the Internet with centralized wireless links, it was shown that QUIC-TCP and a queueing-delay based MaxWeight-type wireless-link scheduler can constitute implicit primal-dual solvers for NUM problems in the network fluid model.

The MaxWeight scheduling in [24] requires centralized implementation, and it is efficient only for cellular (e.g., WiMax or UMTS HSDPA) wireless networks. This limits the applicability of the proposed schemes, since many emerging mobile devices access the Internet through ad-hoc wireless links enabled by (for example) WiFi, ZigBee, or MiWi networks, where CSMA is employed to allow random access, for greater flexibility and freedom. To fill this need, the present paper generalizes our approach to joint optimization of TCP congestion control and distributed CSMA scheduling for Internet traffic over (IEEE 802.11, 802.15.4, or 1451) ad-hoc networks. Building on the window-based primal-dual solution concept, we adopt the QUIC-TCP for congestion control. For distributed scheduling of wireless links, we incorporate the elegant principles in [10] to propose a new CSMA algorithm, where each link-transmitter employs its queueing delay to properly control its back-off time to randomly access the shared wireless channel. Development of the proposed algorithms is beyond simple, heuristic generalization and combination of the existing approaches. In particular, we reveal that the proposed CSMA scheduling can be "glued" with the QUIC-TCP algorithm to constitute an implicit primal-dual solver for an intended optimization problem. Relying on the Lyapunov argument, global convergence of this joint design to the optimal network equilibrium can be proven for the idealized CSMA implementation in the network fluid model. The design and analysis of the proposed design-space oriented cross-layer optimization are challenging, and fundamentally different from the classic NUM approach. As a return, we show that the engineering heuristics building on the proposed designs can lead to practical schemes that can be implemented *asynchronously without message passing* among nodes. Confined to the design space of TCP and CSMA, they are *readily deployed* for practical networks.

The rest of this paper is organized as follows: Section II describes the network models, problem formulation, and classic NUM solutions. Relying on the proposed novel approach, development and analysis of joint TCP and CSMA designs are presented in Section III. Section IV contains simulation results to demonstrate the merits of the proposed schemes, followed by the conclusion.

II. Network Modeling, Problem Formulation, and Existing Solution

Consider the Internet with a wired backbone and a CSMA wireless network in Fig. 1(a). The wireless network can consist of several (separated) CSMA-based (IEEE 802.11 WiFi, 802.15.4 ZigBee, or 1451 sensor) sub-networks without loss of generality. A logical link is a transmitter-receiver pair. The set of links $L = L_f \cup L_w$ is composed of a wired set L_f and a wireless set L_w. Any wired link $l \in L_f$ is assumed to have a constant and independent capacity c_l. When interference is absent, a wireless link $l \in L_w$ has an ergodic capacity b_l. The interference among wireless links is modeled by a conflict graph $\mathcal{G}$, with vertices being all wireless links. Two links interfere if, and only if, there is an edge between them in $\mathcal{G}$; see an example in Fig. 1(b).

For a given $\mathcal{G}$, we can find a total of $I + 1$ independent sets (ISs) $i = 0, \ldots, I$, where each IS contains a set of non-interfering links that are allowed to be simultaneously active. Denote the ith IS by a boolean vector $\boldsymbol{r}^i$ where the lth element of $\boldsymbol{r}^i$, $r_l^i = 1$, if link l is in the set, and $r_l^i = 0$, otherwise.

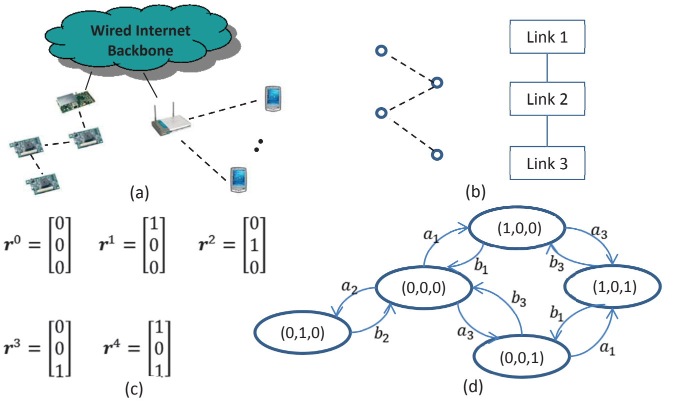

Wired Internet Backbone
(a)
Link 1
Link 2
Link 3
(b)
$r^0 = \begin{bmatrix}0\\0\\0\end{bmatrix}$
$r^1 = \begin{bmatrix}1\\0\\0\end{bmatrix}$
$r^2 = \begin{bmatrix}0\\1\\0\end{bmatrix}$
$r^3 = \begin{bmatrix}0\\0\\1\end{bmatrix}$
$r^4 = \begin{bmatrix}1\\0\\1\end{bmatrix}$
(c)
(1,0,0)
(0,0,0)
(1,0,1)
(0,1,0)
(0,0,1)
a_1
b_1
a_2
b_2
a_3
b_3
a_3
b_3
b_1
a_1
(d)

Fig. 1. Example: (a) Internet with a wired backbone and CSMA networks. (b) Conflict graph $\mathcal{G}$ for three wireless links. (c) IS vectors $\boldsymbol{r}^i$, $i = 0, 1, \ldots, 4$. (d) CSMA Markov chain.

As shown in Fig. 1(c), there always exists an 0th IS, $\boldsymbol{r}^0$ with $r_l^0 = 0$, $\forall l \in L_w$ (i.e., no link is active), and the $|L_w|$ ISs, $\boldsymbol{r}^i = \boldsymbol{e}^i$, where $\boldsymbol{e}^i$ denotes a standard basis vector with only the ith element set to 1, and all other elements equal to 0 (i.e., only link i is active).

A. Idealized CSMA Protocol

For completeness, we review an idealized model of CSMA due to [25], which was widely adopted by [10], [12], [26]. In this model, the carrier-sensing time is assumed to be negligible. If the transmitter of a link l senses the transmissions of any other links, it remains silent. When none of its interfering links are sensed to be active, the transmitter then independently backs off (i.e., waits) for a random period of time that is continuously distributed with mean $1/a_l$ before transmitting. Due to the negligible sensing time and continuous distribution of back-off times, the probability for two interfering links to transmit at the same time is zero; i.e., the collisions can be avoided. With such an idealized CSMA, the transmission status of wireless links form a continuous-time Markov chain, where each state can be described by an IS vector $\boldsymbol{r}^i$ (since simultaneous transmission of interfering links never occurs); see Fig. 1(d).

In the CSMA Markov chain, if link l in a state $\boldsymbol{r}^i$ is not active ($r_l^i = 0$), and all of its interfering links are not active either, then state $\boldsymbol{r}^i$ transits to a new state $\boldsymbol{r}^i + \boldsymbol{e}^l$ with rate a_l; i.e., its lth element changes from 0 to 1, while all other elements remain unchanged. Since link l has an ergodic capacity b_l (i.e., service time $1/b_l$), the state $\boldsymbol{r}^i + \boldsymbol{e}^l$ transits back to $\boldsymbol{r}^i$ with rate b_l. Define the "transmission aggressiveness" (TA) $\rho_l = \log(a_l/b_l)$. For a given positive vector $\boldsymbol{\rho} := \{\rho_l, \forall l \in L_w\}$, the stationary distribution of the feasible states $\boldsymbol{r}^i$, $i = 0, \ldots, I$, is given by [10], [26]:

$$\Pr(\boldsymbol{r}^i; \boldsymbol{\rho}) = \frac{\exp(\sum_{l \in L_w} \rho_l r_l^i)}{\sum_j \exp(\sum_{l \in L_w} \rho_l r_l^j)}. \tag{1}$$

In fact, it can be easily checked from (1) that:

$$\frac{\Pr(\boldsymbol{r}^i + \boldsymbol{e}^l; \boldsymbol{\rho})}{\Pr(\boldsymbol{r}^i; \boldsymbol{\rho})} = \exp(\rho_l) = \frac{a_l}{b_l},$$

which is exactly the balance equation between any two feasible states $\boldsymbol{r}^i$ and $\boldsymbol{r}^i + \boldsymbol{e}^l$. This confirms that (1) is invariant, and a steady distribution of the CSMA Markov chain.

With the stationary distribution (1), the probability for link l being active is clearly: $\sum_{i=0}^{I} [r_l^i \Pr(\boldsymbol{r}^i; \boldsymbol{\rho})]$; thus the "effective" capacity of link l is:

$$R_l = b_l \sum_{i=0}^{I} [r_l^i \Pr(\boldsymbol{r}^i; \boldsymbol{\rho})].$$

B. Problem Formulation

Suppose that the network is shared by a set of unicast flows, identified by their sources $s \in S$. With a given routing table, each flow s may go through multiple wired and wireless links. Let $L(s) \subseteq L$ be the set of links that source s uses, and $S(l) \subseteq S$ be he set of sources that use link l.

Let x_s denote the sending rate of source s. For congestion control, consider a weighted proportionally-fair utility function $p_s \log x_s$ for each flow, as in the TCP Vegas and FAST TCP [18], [23]. With a constant weight vector $\boldsymbol{p} := \{p_s, \forall s\}$, it was shown that the maximization of aggregate utility $\sum_{s \in S} p_s \log x_s$ can lead to an efficient and fair network equilibrium that does not penalize the flows with large propagation delays [22], [27]. Note that the proportionally-fair function is selected for elaboration purposes only; our approach can apply to general concave utility functions [24].

For the CSMA wireless network, the scheduling policy is to control the random back-off time of each link in order to attain high "effective link-capacities." In the idealized CSMA model, let $\boldsymbol{u} := \{u_i, i = 0, \ldots, I\}$ denote the stationary distribution of feasible states $\boldsymbol{r}^i$. It was shown that maximization of the (concave) information entropy of $\boldsymbol{u}$, i.e., $-\sum_{i=0}^{I} u_i \log u_i$, facilitates the development of "throughput optimal" distributed CSMA scheduling schemes [10]. For a joint design of congestion control and CSMA scheduling, we then consider the following network optimization subject to (s. t.) link capacity constraints:

$$\begin{aligned} \max_{\boldsymbol{x},\, \boldsymbol{u}} \quad & \sum_{s \in S} p_s \log x_s - \sum_{i=0}^{I} u_i \log u_i \\ \text{s. t.} \quad & \sum_{s \in S(l)} x_s \le c_l, \ \forall l \in L_f \\ & \sum_{s \in S(l)} x_s \le b_l \sum_{i=0}^{I} (u_i r_l^i), \ \forall l \in L_w \\ & x_s \ge 0, \ \forall s, \quad 0 \le u_i \le 1, \ \forall i, \ \sum_{i=0}^{I} u_i = 1. \end{aligned} \tag{2}$$

Compared to classic NUM formulation, the objective contains the entropy of $\boldsymbol{u}$ in addition to the aggregate utility of source rates. Since the entropy is bounded by $\log(I+1)$, the solution of (2) has, at most, a constant optimality gap $\log(I+1)$ with classic NUM solutions [10], [12]. This gap can be made (arbitrarily) relatively small when (very) large weights p_s are used for the utility functions.

Different from [10], here we consider a hybrid wired-wireless network, and assume that the interference-free capacity b_l can be different, since adaptive modulation-codings are employed in practical (e.g., in IEEE 802.11) CSMA networks to support different rates over links with different channel qualities. More importantly, our formulation (2) is a link-based one, as in [3], [5], [23], whereas [10] adopted a node-based formulation as with [4], [6]. Node-based formulation could facilitate the celebrated "queue-backpressure" type dynamic multi-path routing algorithm, at the expense of requiring each link transmitter to maintain separate queues per flow. The link-based one (2) only assumes that each transmitter has a single queue for all incoming traffic; it is more efficient from the implementation viewpoint, for the Internet is currently not supporting multi-path routing.

C. Classic Solution with the Help of Message Passing

With a node-based formulation, a queue-length based congestion control and "queue-backpressure" based scheduling scheme for CSMA wireless networks was developed in [10]. In a similar spirit, we now derive a queue-length based solution for the link-based formulation (2). It is easy to see that (2) is a convex optimization. Let $\boldsymbol{\lambda} := \{\lambda_l, \forall l\}$ denote the Lagrange multipliers for the link capacity constraints. The (partial) Lagrangian function of (2) is:

$$\begin{aligned} L(\boldsymbol{\lambda};\boldsymbol{x},\boldsymbol{u}) &= \sum_s p_s \log x_s - \sum_{l\in L_f} \lambda_l\big(\sum_{s\in S(l)} x_s - c_l\big) \\ &\quad - \sum_i u_i \log u_i - \sum_{l\in L_w} \lambda_l\big(\sum_{s\in S(l)} x_s - b_l \sum_i (u_i r_l^i)\big) \\ &= \sum_{l\in L_f} \lambda_l c_l + \sum_s (p_s \log x_s - \lambda^s x_s) \\ &\quad + \sum_i \big[-u_i \log u_i + u_i \sum_{l\in L_w} \big(\lambda_l b_l r_l^i\big)\big] \end{aligned}$$

where we define $\lambda^s := \sum_{l\in L(s)} \lambda_l$.

For convex program (2), its optimal solution $\{\boldsymbol{x}^*, \boldsymbol{u}^*\}$ and the optimal dual $\boldsymbol{\lambda}^*$ must satisfy the well-known Karush-Kuhn-Tucker (KKT) conditions [28]:

$$\frac{p_s}{x_s^*} = \lambda^{s*} := \sum_{l\in L(s)} \lambda_l^*, \quad \forall s \in S \tag{3}$$

$$\lambda_l^*\Bigg(\sum_{s\in S(l)} x_s^* - c_l\Bigg) = 0, \quad \sum_{s\in S(l)} x_s^* \le c_l, \quad \forall l \in L_f \tag{4}$$

$$\lambda_l^*\Bigg(\sum_{s\in S(l)} x_s^* - b_l \sum_i [u_i^* r_l^i]\Bigg) = 0, \quad \forall l \in L_w \tag{5}$$

$$\sum_{s\in S(l)} x_s^* \le b_l \sum_i [u_i^* r_l^i], \quad \forall l \in L_w \tag{6}$$

$$u_i^* = \frac{\exp(\sum_{l\in L_w}(\lambda_l^* b_l r_l^i))}{\sum_j \exp(\sum_{l\in L_w}(\lambda_l^* b_l r_l^j))}, \quad \forall i. \tag{7}$$

The optimal $\{\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{\lambda}^*\}$ satisfying (3)–(7) can be obtained by classic (sub-)gradient based dual iterations. For a given $\boldsymbol{\lambda}$, the dual-optimal source rate is clearly $x_s^*(\boldsymbol{\lambda}) = p_s/\lambda^s$, $\forall s$. It can also be shown from the optimality condition that, $\forall i$,

$$u_i^*(\boldsymbol{\lambda}) = \frac{\exp(\sum_{l\in L_w}(\lambda_l b_l r_l^i))}{\sum_j \exp(\sum_{l\in L_w}(\lambda_l b_l r_l^j))}. \tag{8}$$

Given $x_s^*(\boldsymbol{\lambda})$ and $u_i^*(\boldsymbol{\lambda})$, the dual problem of (2) can be solved by the sub-gradient descent iterations:

$$\begin{aligned} &\lambda_l[t+1] = \big[\lambda_l[t] + \epsilon\big(\sum_{s\in S(l)} x_s^*(\boldsymbol{\lambda}[t]) - c_l\big)\big]^+, \quad \forall l \in L_f, \\ &\lambda_l[t+1] = \big[\lambda_l[t] + \epsilon\big(\sum_{s\in S(l)} x_s^*(\boldsymbol{\lambda}[t]) - b_l \sum_i [u_i^*(\boldsymbol{\lambda}[t]) r_l^i]\big)\big]^+, \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall l \in L_w \end{aligned} \tag{9}$$

where t is the iteration index, ϵ is a stepsize, and $[x]^+ := \max(0, x)$. Convergence of (9) to the optimal $\boldsymbol{\lambda}^*$ is guaranteed from any initial $\boldsymbol{\lambda}[0]$ [28]. Due to the zero duality gap, the corresponding $x_s^*(\boldsymbol{\lambda}^*)$ and $u_i^*(\boldsymbol{\lambda}^*)$ yield the optimal solutions for the primal problem (2). Based on this sub-gradient method, a joint congestion control and CSMA scheduling can be devised as follows.

Comparing (8) with (1), we find that the distribution $u_i^*(\boldsymbol{\lambda})$ corresponds to the steady state distribution of the idealized CSMA network with the TA of each link set to $\rho_l \equiv \lambda_l b_l$. On the other hand, with the aggregate arrival rate $\sum_{s\in S(l)} x_s^*(\boldsymbol{\lambda}[t])$ and service rate c_l or $b_l \sum_i [u_i^*(\boldsymbol{\lambda}[t]) r_l^i]$ for each link, the sub-gradient iteration in (9) can be seen as an evolution of the scaled queue lengths ϵQ_l:

$$\lambda_l[t+1] \equiv \epsilon Q_l[t+1] =$$
$$\begin{cases} \epsilon\big[Q_l[t] + \big(\sum\limits_{s\in S(l)} x_s^*(\epsilon\boldsymbol{Q}[t]) - c_l\big)\big]^+, \quad \forall l \in L_f, \\ \epsilon\big[Q_l[t] + \big(\sum\limits_{s\in S(l)} x_s^*(\epsilon\boldsymbol{Q}[t]) - b_l \sum\limits_i [u_i^*(\epsilon\boldsymbol{Q}[t])\xi_l^i)\big)\big]^+, \forall l \in L_w \end{cases}$$

Based on the latter, we can have the following congestion control and CSMA scheduling schemes:

- *Congestion control*: In the update-period t per flow s, the source sets its rate $x_s[t] = \frac{p_s}{\epsilon \sum_{l\in L(s)} Q_l[t]}$.
- *CSMA scheduling*: In the update-period t per wireless link, the link-transmitter sets its TA as $\rho_l[t] = \epsilon b_l Q_l[t]$, or equivalently, sets the mean of its random back-off time as $e^{-\epsilon b_l Q_l[t]}/b_l$, such that the effective link capacity becomes $R_l = b_l \sum_i [u_i^*(\epsilon\boldsymbol{Q}[t]) r_l^i]$.

These schemes, together with the queue evolution $Q_l[t]$, then follow the sub-gradient approach (9) to solve (2). Note that the steady state distribution $u_i^*(\epsilon\boldsymbol{Q}[t])\xi_l^i)$ can be produced by the CSMA strategy based on local queue length, Q_l per link over a sufficiently long time, without centralized coordination and any message passing from neighbors [10], [12]; hence, the CSMA scheduling here is implemented in a truly distributed manner. However, the congestion controller at each source node requires message passing of the (aggregate) queue lengths $\sum_{l\in L(s)} Q_l$ along the flow route from intermediate nodes.

For both node-based solutions here and link-based schemes [10] under the classic NUM framework, it can be shown that one-hop message passing of queue lengths of all active links is required. Recall that, in order to perform the queue-backpressure schemes in link-based solutions, each node needs to maintain separate queues for each flow through it. Suppose that each active link is used by $N \ge 1$ flows on average. Then, the node-based schemes in [10] need to maintain N times more queues, and incur N times more message passing overheads, than the link-based solution here. Nevertheless, in both link- and node-based solutions, the congestion control is implemented by a source-rate controller (instead of TCP window-control), and explicit message passing from the network is required. As a result, they are difficult to operate over the current Internet infrastructure.

III. Joint Congestion Control and CSMA Scheduling without Message Passing

To overcome the limitations of classic NUM solutions, we next rely on a new design-spaced oriented approach to develop jointly optimal, yet readily deployable, TCP congestion control and CSMA scheduling for (2). In the proposed integral design and analysis process, we first consider an idealized CSMA protocol and a fluid model of network, where packets are infinitely divisible and small.

In this idealized network fluid model, let $\boldsymbol{w} := \{w_s, \forall s\}$ collect the window sizes for all sources $s \in S$, $\boldsymbol{q} := \{q_l, \forall l\}$ collect the round-trip queueing delays for all links $l \in L$, and $\boldsymbol{d} := \{d_s, \forall s\}$ collect the fixed round-trip propagation (plus processing) delays for all sources. For the CSMA wireless network, let $\boldsymbol{R} := \{R_l, \forall l \in L_w\}$ denote the effective capacity vector for wireless links under the given scheduling strategy. Upon defining the aggregate queueing delays $q^s := \sum_{l \in L(s)} q_l$ along the flow routes, we have the following relationships for x_s, w_s, and q_l [22]:

$$x_s(d_s + q^s) = w_s, \quad \forall s \in S \tag{10}$$

$$q_l\left(\sum_{s\in S(l)} x_s - c_l\right) = 0, \quad \sum_{s\in S(l)} x_s \le c_l, \quad \forall l \in L_f \tag{11}$$

$$q_l\left(\sum_{s\in S(l)} x_s - R_l\right) = 0, \quad \sum_{s\in S(l)} x_s \le R_l, \quad \forall l \in L_w \tag{12}$$

where (10) simply follows from that the source rate x_s is equal to the window size w_s divided by the total round-trip delay $d_s + q^s$, (11) and (12) are implied by the link capacity constraints, and the fact that if the aggregate rate through a link l is less than its capacity c_l or R_l, then the queueing delay at this link is equal to zero since packets are infinitely divisible and small.

A. Joint Design in Network Fluid Model

We next propose a joint TCP congestion control and CSMA scheduling design in the (simplified) fluid model. Comparing the fluid model identities (11)–(12) with the KKT conditions (4)–(5), we find that the queueing delay q_l in the fluid model can play a similar role of Lagrange multiplier λ_l associated with the link capacity constraints in (2). Then, implied by (7), we propose a CSMA scheduling strategy, where each link actually employs the queueing delay to set its TA as $\rho_l = b_l q_l$, such that the resultant steady state distribution is [cf. (1) and (8)]:

$$\Pr(\boldsymbol{r}^i) = \frac{\exp(\sum_{l\in L_w} b_l q_l r_l^i)}{\sum_j \exp(\sum_{l\in L_w} b_l q_l r_l^j)} = u_i^*(\boldsymbol{q}), \tag{13}$$

and the effective link capacities are:

$$R_l = b_l \sum_i [u_i^*(\boldsymbol{q}) r_l^i], \quad \forall l \in L_w. \tag{14}$$

On the other hand, in the TCP congestion control, each flow source adjusts its transmission window size (i.e., the maximum amount of outstanding packets that it can send to the network) to prevent network congestion according to a locally observable congestion measure. Using delay as the congestion measure, we consider adopting the QUIC-TCP algorithm [24] to adjust the window size per flow s. Denote the (local) total round-trip delay of flow s as $\bar{d}_s := d_s + q^s$. With $x_s = w_s/\bar{d}_s$ from (10), define $v_s := w_s - x_s d_s - p_s$. Parameterized by a constant $\rho \in [0, 1]$, QUIC-TCP entails a class of end-to-end algorithms with window adjustment following the ordinary differential equations (κ is a positive constant):

$$\frac{d}{dt} w_s(t) = -\kappa \frac{d_s}{\bar{d}_s} w_s^{-2\rho+1} v_s, \quad \forall s. \tag{15}$$

This class of algorithms specializes to the Mo-Walrand's scheme [22]: $\frac{d}{dt} w_s(t) = -\kappa \frac{d_s}{\bar{d}_s} \frac{v_s}{w_s}$ as $\rho = 1$. With $\rho = 1/2$, the proposed window update becomes $\frac{d}{dt} w_s(t) = -\kappa \frac{d_s}{\bar{d}_s} v_s$, which is similar to that in FAST-TCP [23]: $\frac{d}{dt} w_s(t) = -\kappa v_s$.

The proposed joint design adjusts the TCP window size w_s and the CSMA TA ρ_l, in order to control the source-rate $\boldsymbol{x}$, wireless-link capacity $\boldsymbol{R}$, and queueing delay $\boldsymbol{q}$. The goal of the proposed network control is to drive the network operating point towards a desired equilibrium that yields the optimal solution for (2).

B. Optimality and Convergence Analysis

Define $\boldsymbol{v} := \{v_s, \forall s\}$. In an equilibrium of window update (15), we clearly have $v_s = 0$, $\forall s$; i.e., $\boldsymbol{v} = \boldsymbol{0}$. Let $\boldsymbol{w}^*$ denote the window-size vector with this equilibrium, and let $\boldsymbol{x}^*$, $\boldsymbol{u}^*$, and $\boldsymbol{q}^*$ be the corresponding source-rate, CSMA stationary distribution, and queueing-delay vectors. By showing that $\boldsymbol{x}^*$, $\boldsymbol{u}^*$ and $\boldsymbol{q}^*$ satisfy the KKT conditions (3)–(7) under the proposed joint design, we can rigorously establish:

Theorem 1: *For the proposed joint congestion control and CSMA scheduling scheme, there is a unique window size vector $\boldsymbol{w}^*$ such that $\boldsymbol{v} = \boldsymbol{0}$, and the corresponding rate vector $\boldsymbol{x}^*$ and CSMA steady distribution $\boldsymbol{u}^*$ are the optimal ones for (2).*

Proof: See Appendix A. ■

Theorem 1 states that the unique equilibrium of the proposed joint window adjustment and CSMA scheduling scheme leads to the optimal $\boldsymbol{x}^*$ and $\boldsymbol{u}^*$ for (2). Based on local observations without message passing, the proposed scheme aims to entail an *implicit primal-dual* update of $\{\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{q}\}$ towards the optimal equilibrium that solves (2). Different from the prior queue-length based NUM solutions, the proposed scheme fits well into the design space of the Internet protocols. Also different from the heuristic schemes, we next show the global convergence of the proposed scheme to its optimal equilibrium from any initial state.

For convenience, rewrite the problem (2) as:

$$\begin{aligned} \max_{\boldsymbol{x},\,\boldsymbol{u}} \quad & \sum_s p_s \log x_s - \sum_i u_i \log u_i \\ \text{s. t.} \quad & \boldsymbol{A}_f \boldsymbol{x} \le \boldsymbol{c}, \quad \boldsymbol{A}_w \boldsymbol{x} \le \boldsymbol{R}, \\ & x_s \ge 0,\ \forall s, \quad 0 \le u_i \le 1,\ \forall i,\ \sum_i u_i = 1. \end{aligned} \tag{16}$$

where $\boldsymbol{c} := \{c_l, \forall l \in L_f\}$, $\boldsymbol{R} := \{R_l, \forall l \in L_w\}$ with $R_l = b_l \sum_i (u_i r_l^i)$, and the routing matrix $\boldsymbol{A} := [\boldsymbol{A}_f^T,\ \boldsymbol{A}_w^T]^T$ with its (l, s)th entry $A_{ls} = 1$ if $s \in S(l)$, and $A_{ls} = 0$ otherwise.

Consider the idealized network fluid model. In the proposed CSMA scheduling scheme, the queueing delays q_l of wireless links are employed for random access to result in the link capacities R_l in (14). Let $\boldsymbol{q}_w := \{q_l, \forall l \in L_w\}$, and $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w} := \{\frac{\partial R_l}{\partial q_n}, \forall l, n\}$ denotes the Jacobian matrix of vector $\boldsymbol{R}$ with respect to vector $\boldsymbol{q}_w$. Then we can establish the following lemma:

Lemma 1: *With the proposed CSMA scheduling strategy, the Jacobian matrix* $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w}$ *is positive definite for any queueing delay vector* $\boldsymbol{q}_w > \boldsymbol{0}$.

Proof: See Appendix B. ■

Lemma 1 shows that, when queueing delays are used to set the TA of links in an idealized CSMA, the wireless links are coupled in a "desired" manner such that $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w}$ is positive definite. Using this nice property, we can then show the following global stability result:

Theorem 2: *The unique equilibrium* $\boldsymbol{v} = 0$ *of the proposed joint QUIC-TCP congestion control and CSMA scheduling scheme is globally asymptotically stable for* $0 \le \rho \le 1$*; i.e., the proposed scheme globally converges to its unique equilibrium that yields the optimal solution for (2).*

Proof: See Appendix C. ■

The proof for Theorem 2 is based on the Lyapunov method. Relying on the positive definiteness of $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w}$ under the proposed CSMA scheduling per Lemma 1, we show that a quadratic function $Y(\boldsymbol{w}) = (1/2)\sum_{s \in S}(v_s/(w_s)^\rho)^2$ is a Lyapunov function for (15); consequently, global convergence of the proposed schemes to its unique equilibrium readily follows.

Remark 1: Theorem 2 is a generalization of [22, Theorem 5], which holds only for the QUIC-TCP with $\rho = 1$ (i.e. Mo-Walrand scheme) in *wired* networks. Here, the theorem holds for $0 \le \rho \le 1$, and it proves the existence of jointly optimal TCP congestion control and CSMA scheduling schemes, *without explicit message passing*. A key for the generalization is that queueing delays are employed to play the role of Lagrange multipliers to integrate the elegant CSMA approach [10] into our window-based cross-layer optimization framework. Using the queueing delays (instead of queue lengths in [10]) to set the random backoff, the proposed CSMA scheduling can be then "glued" with the QUIC-TCP algorithm to constitute an implicit primal-dual solver for the intended NUM problem. Specifically, we are able to show the novel positive definite $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w}$ result in Lemma 1. Relying on this result, convergence analysis in [22] for *wired* networks can be then generalized to prove global convergence of the proposed QUIC-TCP and queueing-delay based CSMA scheduling for Internet with wired and *ad-hoc wireless* links.

Remark 2: Note that our approach, in fact, hinges on the tight integration of theory and design, where analysis is actually done *at* the design time, *not after*. Specifically, we perform a Lyapunov function based stability analysis *during* the development of the non-standard window-based implicit solver in the network fluid model. With queueing delays playing the role of Lagrange multipliers, we rely on the KKT conditions to identify the desired equilibrium of the proposed solver and, subsequently, propose a quadratic candidate function $Y(\boldsymbol{w})$. By checking the first time-derivative $\frac{d}{dt}Y(\boldsymbol{w}(t))$, we derive the QUIC-TCP window-update (15) and queueing-delay based CSMA scheduling (13) as the sufficient and necessary conditions to render $Y(\boldsymbol{w})$ a Lyapunov function for the desired equilibrium. Global convergence of (15) and (13) to the optimal operating point is then established via the Lyapunov argument. This design and analysis process is fundamentally different from that in the classic NUM framework. The approach is an attempt to leverage the network optimization theory to practical Internet designs.

C. Development of Practical Schemes

Using the insights provided by the joint design in the network fluid model, we next develop the congestion control and CSMA scheduling schemes for practical TCP and IEEE 802.11 protocol.

Congestion control: The QUIC-TCP congestion control was specified in [24], which we briefly repeat here for completeness. Implementation of the QUIC-TCP window update (15) requires the estimations of the propagation (plus processing) delay d_s and total round-trip delay $\bar{d}_s$ per flow s. To this end, let each flow source s compute the round-trip-time (RTT), RTT_s, of the acknowledged packet, whenever an in-order acknowledge (ACK) is received. Using RTT_s, the source s updates a variable $BaseRTT_s$ as the minimum RTT observed so far, to approximate d_s [23]. Given the current RTT_s, the source s also updates the average total round-trip delay $AvgRTT_s$ via a low-pass filter (e.g., $AvgRTT_s \leftarrow \frac{255}{256} \times AvgRTT_s + \frac{1}{256} \times RTT_s$) to estimate $\bar{d}_s$.

Note that $v_s = w_s - x_s d_s - p_s = w_s - \frac{w_s}{\bar{d}_s} d_s - p_s$. With $BaseRTT_s$ and $AvgRTT_s$ playing the role of d_s and $\bar{d}_s$, respectively, flow source s then adjusts the window size w_s [cf. (15)]:

$$w_s \leftarrow w_s - \kappa \frac{BaseRTT_s}{AvgRTT_s} w_s^{-2\rho+1}\left(w_s - \frac{BaseRTT_s}{AvgRTT_s} w_s - p_s\right) \qquad (17)$$

where κ is a stepsize. When an in-order ACK is not received, the slow start and/or fast recovery schemes in standard TCP are preserved to deal with transmission time-outs or duplicate ACKs.

CSMA scheduling: In the proposed CSMA scheduling strategy (at link-layer), each wireless link transmitter employs the queueing delay to set its TA. To estimate its queueing delay, let each wireless transmitter node read its queue length $QueLEN_l[t]$, and calculate the average rate $AveR_l[t]$ using a similar low-pass filter at an update time t. Implied by Little's law, the average queueing delay is given by the ratio of average queue length and average rate. Based on the fluid limit argument [29], the "stochastic" delay $\frac{QueLEN_l[t]}{AveR_l[t]}$, i.e., current queue length $QueLEN_l[t]$ divided by the average rate $AveR_l[t]$, could play the role of average delay q_l in the deterministic fluid model. Supposing that the interference-free ergodic capacity b_l is known (e.g., informed by a bandwidth estimator at the physical layer), we then propose that the transmitter of link l sets its TA as:

$$\rho_l[t] = b_l \frac{QueLEN_l[t]}{AveR_l[t]}, \qquad (18)$$

or equivalently, it sets the mean of its backoff time as $\exp(-b_l \frac{QueLEN_l[t]}{AveR_l[t]})/b_l$, to randomly access the channel.

In the IEEE 802.11 CSMA protocol, after a successful transmission without collisions, the wireless transmitter resets its contention window size to a predetermined value $CW_{\min}$, and randomly selects an integer c from $\{0, 1, \ldots, CW_{\min} - 1\}$. If the channel is sensed idle, the transmitter then backs off a time of cT_{slot} before the next transmission, where T_{slot} is a predefined minslot duration. In this case, the mean backoff time is $\frac{CW_{\min}-1}{2} T_{slot}$. To incorporate the proposed strategy, we let the transmitter of link l set its own contention window size to $2\exp(-b_l \frac{QueLEN_l[t]}{AveR_l[t]})/(b_l T_{slot}) + 1$ (instead of $CW_{\min}$) after a successful transmission, and perform aforementioned random backoff such that the mean backoff time becomes $\exp(-b_l \frac{QueLEN_l[t]}{AveR_l[t]})/b_l$. Except for this modification, all other components of IEEE 802.11 CSMA, including the binary exponential backoff (BEB) scheme and maximum window size value $CW_{\max}$ upon packet collisions, are preserved.

Remark 3: Clearly, the proposed congestion control and scheduling schemes are confined to the design space of TCP and CSMA protocols. While the congestion controller (17) preserves the distributed end-to-end window-control mechanism of TCP, the CSMA random back-off strategy (18) can be performed in a truly distributed manner, without message passing. They can be implemented in an asynchronous manner, and can be readily deployed with the current (layered) Internet infrastructure. In addition, as they are designed based on the fluid-model counterparts that are provably optimal for (2), high performance can be expected for these schemes.

Note that, although the global convergence/optimaility result in Theorem 2 relies on the idealized CSMA model in [10], [12], [25], [26], the proposed schemes (17) and (18) can be readily operated over practical networks, where not only non-zero carrier-sensing time and discrete back-off periods are used for CSMA implementation (thus packet collisions exist), but stochastic network dynamics and feedback delays are also present.

IV. Simulation Results

In this section, we first use Matlab simulations to validate the proposed algorithms in the idealized network fluid model, and then rely on ns-2 simulations to test the proposed QUIC-TCP and CSMA schemes under a realistic network environment in the presence of stochastic dynamics, packet collisions, and feedback delays.

A. Simulation Network Topology

The simulation network consists of six fixed computer nodes (W_0–W_5), an access point (AP), and fifteen wireless nodes (N_1–N_{15}), as shown in Fig. 2. The fixed nodes constitute the wired Internet backbone. Wired links between fixed nodes have the same capacity, 1 MHz, and the same propagation delay, 5 ms. The wireless nodes and the AP are arranged in a grid, consisting of a multi-hop wireless local area network with a shared bandwidth of 1 MHz. The distance between two adjacent nodes (horizontally or vertically) is 150 m, and the propagation times are assumed to be negligible. The transmission range of each wireless node is 250 m. Four TCP connections are established in the network to carry four FTP flows. The source nodes for the flows are W_0, W_1, W_2, and W_3, while the corresponding destination (i.e., sink) nodes are N_{15}, N_{12}, N_3, and N_6, respectively. The routing between each source and its sink is determined by the default Dijkstra's algorithm-based static routing protocol for the wired backbone, and the destination-sequenced distance-vector (DSDV) protocol for ad-hoc wireless links.

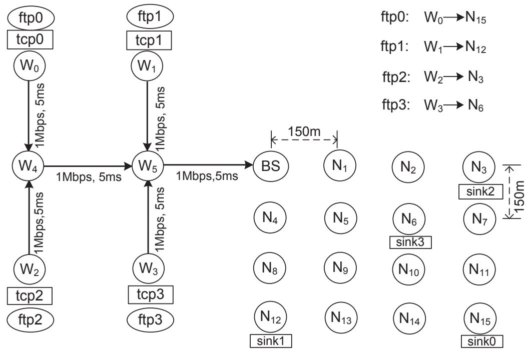


Fig. 2. Network topology.

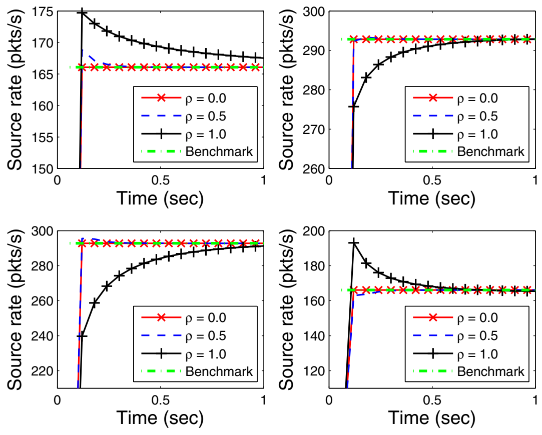


Fig. 3. Evolutions of the source rates (Matlab simulation).

B. Matlab Simulations

Consider the idealized network fluid model. The proposed window-control algorithm (15) is implemented at the source nodes, and the queueing-delay based CSMA scheduling (13) is performed at the wireless nodes. Fig. 3 shows the source rate evolutions of the four flows for the QUIC-TCP algorithms with $\rho = 0, 0.5, 1$. The optimal source rates obtained by solving (2) via standard sub-gradient iteration are also presented as the benchmark. It is clearly seen that all QUIC-TCP algorithms converge to the optimal equilibrium, showing the correctness of the proposed algorithms.

C. NS-2 Simulations

Consider again the network in Fig. 2 with four FTP flows. Relying on (17), we modify the ns-2 module of FAST-TCP to produce a QUIC-TCP agent. The standard IEEE 802.11b MAC module in the ns-2 is modified to incorporate the proposed strategy (18), and it is then used to simulate a CSMA network with a bandwidth of 20 MHz.

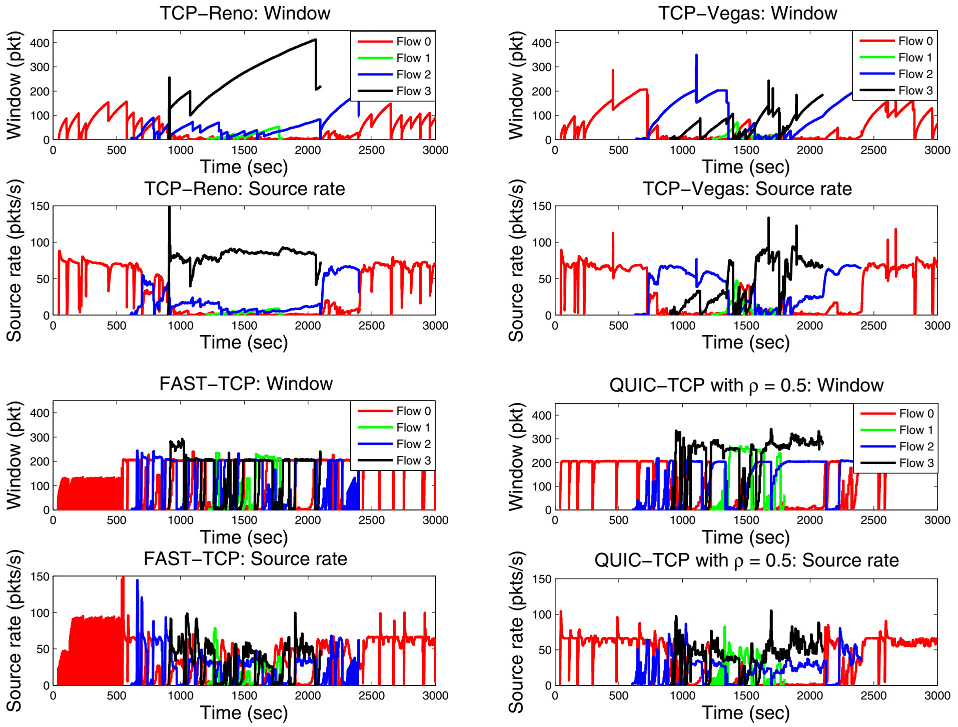


Fig. 4. Window sizes and source rates (ns-2 simulation).

All of the four FTP flows start at 0 sec, and end at 3000 sec. The packet size for FTP flows is 200 bytes, buffer sizes of fixed nodes, AP and wireless nodes are 2000 packets, and p_s of each flow is set to 200 packets. With the proposed CSMA algorithm performed at the wireless nodes, we test TCP-Reno, TCP-Vegas, FAST-TCP and the proposed QUIC-TCP with $\rho = 0.5$, *in the presence of feedback delays and packet collisions*. The stepsize for window updates in both FAST- and QUIC-TCP is set to $\kappa = 1.5$. Fig. 4 depicts the evolutions of window sizes and source rates of the TCP schemes. It is shown that the proposed QUIC-TCP converges to the equilibrium point quickly and stably. This is because the congestion window in QUIC-TCP is adjusted aggressively when it is far away from the optimal point, and the pace of updating is slowed down around the equilibrium; see (17). As a result, the average throughputs for flows ftp0, ftp1, ftp2, and ftp3 are 2.22, 6.24, 3.21, and 5.15 packets/ms, respectively. The FAST-TCP also converges well, although its global convergence was not proven, and it behaves similarly to the QUIC-TCP with $\rho = 0.5$, as expected. The average throughputs for the flows are 2.21, 6.24, 3.21, and 5.13 packets/ms, slightly smaller than those with QUIC-TCP.[1] The TCP-Vegas approaches the equilibrium points slowly. With the lower and upper bounds set to 200 and 203 packets, TCP-Vegas compares the current queue size with these two thresholds, and increases or decreases the congestion window accordingly. The adjustment is made linearly, even if the equilibrium is far away; thus the convergence is slow. Due to its slow convergence, the average flow throughputs become 2.27, 5.99, 2.81, and 3.66 packets/ms, which amounts to a 12.4% loss in aggregate throughput when compared to QUIC-TCP. Different from other TCPs, TCP-Reno uses the packet loss as a congestion measure, and implements an AIMD window adjustment. This algorithm increases the window size linearly to probe for available bandwidth until packet loss occurs. Upon packet loss, it halves the window size to relieve the congestion. Due to this AIMD scheme, TCP-Reno exhibits “sawtooth-like” oscillations in window adjustments and resultant source rates. In addition to the oscillations, the average flow throughputs are 2.66, 6.22, 1.91, and 2.75 packets/ms, a 19.5% loss in total throughput, as compared to QUIC-TCP. Also note that TCP-Reno penalizes the flows ftp2 and ftp3 with large propagation delays, leading to an unfair resource allocation among the flows.

We next evaluate the TCP performance under different CSMA schedulers. Besides the proposed queueing-delay based one (18), we consider two other schedulers: the original CSMA scheduling algorithm in the IEEE 802.11 protocol and the queueing-length based scheduling algorithm, which sets the TA of link l as $\rho_l[t] = QueLEN_l[t]$. Supposing that all of the flows are active from 0 sec to 3000 sec, Fig. 5 compares the performance of the TCP-Reno, TCP-Vegas, FAST-TCP and QUIC-TCP (with $\rho = 0.5$) under different CSMA schedulers. Besides aggregate throughput $\sum_s x_s$, we adopt the well-known Jain's index to measure the fairness [30]: $F = \frac{(\sum_s x_s)^2}{|S| \sum_s (x_s)^2}$, where $|S|$ denotes the number of flows.[2] In addition, we present the resultant $\sum_s p_s \log x_s$ (i.e., objective of (2)) for different schemes. Per Theorem 1, the QUIC-TCP produces the optimal (i.e., $\boldsymbol{p}$-weighted proportionally fair) $\boldsymbol{x}^*$ for (2), only under the proposed scheduler (18). This is corrob-

[1] Note that here the ns-2 module of QUIC-TCP is simply modified from that of FAST-TCP. A re-engineering of QUIC-TCP based on its specifications may yield a better performance.

[2] As we have $|S| = 4$, this index ranges from 0.25 (most unfair) to 1 (most fair).

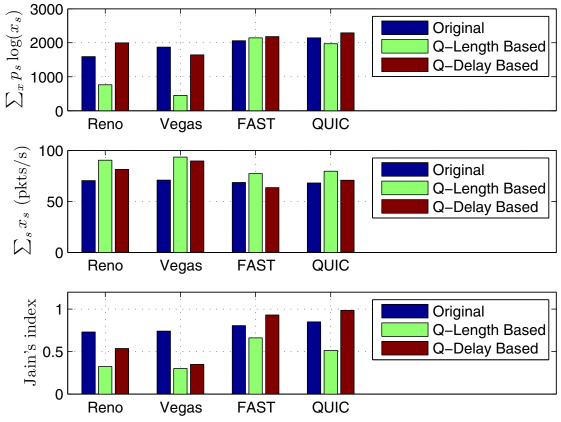


Fig. 5. Comparison for TCP performance under different schedulers (ns-2 simulation).

orated by Fig. 5 (top) where QUIC-TCP, with this scheduler, yields the highest $\sum_s p_s \log x_s$. This leads to a good balance between the aggregate throughput $\sum_s x_s$ and the fairness among flows. Compare it to the results under the original and queue-length based CSMA schedulers. It is found in Fig. 5 (middle) that the aggregate throughput is 70 packets/ms for QUIC-TCP with the proposed scheduler, which is larger than that (67 packets/ms) under the original scheduler, and smaller than that (75 packets/s) under queue-length based schedulers. On the other hand, the Jain's fairness index is $F = 0.99$ under the queueing-delay based scheduler, which improves significantly over those (0.85 and 0.50) under original and queue-length based schedulers; see Fig. 5 (bottom). FAST-TCP has a similar performance as that of QUIC-TCP under different schedulers. The aggregate throughputs of TCP-Vegas and TCP-Reno may be slightly larger than that of QUIC-TCP; however, their Jain's fairness is clearly smaller. For instance, under the queueing-delay based scheduler, the Jain's indices for TCP-Vegas and TCP-Reno are 0.53 and 0.35, respectively.

V. Conclusion

We developed joint TCP congestion control and CSMA scheduling schemes for Internet traffic over distributed multi-hop wireless links without the need of explicit message passing. The proposed schemes are readily deployed for practical networks. Global convergence/stability of these schemes to optimal network equilibrium was proven using the Lyapunov method in the idealized network fluid model. Performance of the proposed schemes in practical networks was evaluated by simulations.

Appendix

A. Proof of Theorem 1

At the equilibrium $\boldsymbol{v} = 0$ with the proposed QUIC-TCP, we have $w_s^* - x_s^* d_s - p_s = 0$; on the other hand, it follows from the fluid model identity (10) that $w_s^* = x_s^*(d_s + q^{s*})$. Hence, we readily have $p_s = x_s^* q^{s*}$, $\forall s$. This is exactly the KKT condition (3), if we let the optimal dual vector $\boldsymbol{\lambda}^* \equiv \boldsymbol{q}^*$. With this equivalence mapping, the fluid model identities (11)–(12), together with (13)–(14) under the proposed CSMA scheduling, also become the KKT conditions (4)–(7). Since $\boldsymbol{x}^*$, $\boldsymbol{u}^*$ and $\boldsymbol{q}^*$ satisfy the sufficient and necessary KKT optimality conditions, it implies that they are the optimal solutions of (2) and the corresponding optimal dual variables. Furthermore, we can follow the similar lines of [24] to show the uniqueness of mapping from $\boldsymbol{w}^*$ to $\boldsymbol{x}^*$ under (11)–(14). Due to this uniqueness, as well as the existence and uniqueness of $\boldsymbol{x}^*$ for (2), it then follows that the optimal window equilibrium $\boldsymbol{w}^*$ always exists, and it is unique.

B. Proof of Lemma 1

From (13), we find the partial derivative: $\forall n \in L_w$,

$$\begin{aligned}\frac{\partial u_i(\boldsymbol{q})}{\partial q_n} &= \frac{b_n r_n^i \exp(\sum_l [q_l b_l r_l^i])}{\sum_j \exp(\sum_l [q_l b_l r_l^j])} \\ &\quad - \frac{\exp(\sum_l [q_l b_l r_l^i]) \sum_j [b_n r_n^j \exp(\sum_l q_l b_l r_l^j)]}{[\sum_j \exp(\sum_l [q_l b_l r_l^j])]^2} \\ &= b_n u_i(\boldsymbol{q}) r_n^i - u_i(\boldsymbol{q}) \Big(\frac{b_n \sum_j [r_n^j \exp(\sum_l [q_l b_l r_l^j)]}{\sum_{j'} \exp(\sum_l [q_l b_l r_l^{j'}])} \Big) \\ &= u_i(\boldsymbol{q})(b_n r_n^i - R_n).\end{aligned}$$

It in turn follows from (14) that: $\forall l,\ n \in L_w$,

$$\begin{aligned}\frac{\partial R_l}{\partial q_n} &= b_l \sum_i \Big(\frac{\partial u_i(\boldsymbol{q})}{q_n} r_l^i \Big) = b_l \sum_i [u_i(\boldsymbol{q})(b_n r_n^i - R_n) r_l^i] \\ &= b_l b_n \sum_i [u_i(\boldsymbol{q}) r_l^i r_n^i] - b_l \Big(\sum_i [u_i(\boldsymbol{q}) r_l^i] \Big) R_n \\ &= b_l b_n \Big[\sum_i [u_i(\boldsymbol{q}) r_l^i r_n^i] - \Big(\sum_i [u_i(\boldsymbol{q}) r_l^i] \Big) \Big(\sum_i [u_i(\boldsymbol{q}) r_n^i] \Big) \Big].\end{aligned} \tag{19}$$

Suppose that there are $I + 1$ ISs for the given conflict graph $\mathcal{G}$ of the wireless links. Recall that there is always an 0th IS, $\boldsymbol{r}^0 = [0\ 0\ \ldots\ 0]^T$. This implies that

$$\sum_{i=0}^{I} [u_i(\boldsymbol{q}) r_l^i r_n^i] = \sum_{i=1}^{I} [u_i(\boldsymbol{q}) r_l^i r_n^i],$$

$$\sum_{i=0}^{I} [u_i(\boldsymbol{q}) r_l^i] = \sum_{i=1}^{I} [u_i(\boldsymbol{q}) r_l^i];$$

i.e., we can omit $i = 0$ for the summations in (19).

Define a vector $\boldsymbol{\mu} := [u_1(\boldsymbol{q})\ \ldots\ u_I(\boldsymbol{q})]^T$, a diagonal matrix $\boldsymbol{U} := \text{diag}(\boldsymbol{\mu})$, and let matrix $\boldsymbol{\Upsilon} := [\boldsymbol{r}^1\ \ldots\ \boldsymbol{r}^I]^T$ collect the IS vectors for the given conflict graph of wireless links. Note that $u_0(\boldsymbol{q})$ and $\boldsymbol{r}^0$ are not included in the vector and matrices. Furthermore, let $\boldsymbol{b} := \{b_l, \forall l \in L_w\}$ and $\boldsymbol{B} := \text{diag}(\boldsymbol{b})$. Using $\boldsymbol{\mu}$, $\boldsymbol{\Upsilon}$, and $\boldsymbol{B}$, it then follows from (19) that the Jacobian matrix $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w} = \{\frac{\partial R_l}{\partial q_n}, \forall l, n\}$ is given by:

$$\begin{aligned}\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w} &= \boldsymbol{B}^T \boldsymbol{\Upsilon}^T \boldsymbol{U} \boldsymbol{\Upsilon} \boldsymbol{B} - \boldsymbol{B}^T \boldsymbol{\Upsilon}^T \boldsymbol{\mu} \boldsymbol{\mu}^T \boldsymbol{\Upsilon} \boldsymbol{B} \\ &= \boldsymbol{B}^T \boldsymbol{\Upsilon}^T (\boldsymbol{U} - \boldsymbol{\mu}\boldsymbol{\mu}^T) \boldsymbol{\Upsilon} \boldsymbol{B}.\end{aligned} \tag{20}$$

By the definitions of $\boldsymbol{\mu}$ and $\boldsymbol{U}$, we have:

$$\boldsymbol{U} - \boldsymbol{\mu}\boldsymbol{\mu}^T = \begin{bmatrix} u_1(1-u_1) & -u_1 u_2 & \cdots & -u_1 u_I \\ -u_2 u_1 & u_2(1-u_2) & \cdots & -u_2 u_I \\ \vdots & \vdots & \ddots & \vdots \\ -u_I u_1 & -u_I u_2 & \cdots & u_I(1-u_I) \end{bmatrix}$$

Now check the diagonal dominance of matrix $\boldsymbol{U}-\boldsymbol{\mu}\boldsymbol{\mu}^T$: For each row i,

$$|u_i(1-u_i)|-\sum_{j\neq i,j=1}^{I}|-u_iu_j|=u_i(1-u_i)-\sum_{j\neq i,j=1}^{I}u_iu_j$$

$$=u_i(1-\sum_{j=1}^{I}u_j)=u_1u_0>0, \qquad (21)$$

since $u_0=\frac{1}{\sum_j e^{\sum_l [q_l b_l r_l^j]}}>0$ and $u_i=\frac{e^{\sum_l [q_l b_l r_l^i]}}{\sum_j e^{\sum_l [q_l b_l r_l^j]}}>0$.

Since the symmetric $\boldsymbol{U}-\boldsymbol{\mu}\boldsymbol{\mu}^T$ is strictly diagonally dominant with positive diagonal entries, it is positive definite. Recall that the $I\times|L_w|$ matrix $\boldsymbol{\Upsilon}$ must have full column rank, i.e., rank$(\boldsymbol{\Upsilon})=|L_w|$, since it always contains the $|L_w|$ standard basis vectors $\boldsymbol{r}^i=\boldsymbol{e}^i$, $i=1,\ldots,|L_w|$. Because $\boldsymbol{B}$ is a $|L_w|\times|L_w|$ diagonal matrix with positive diagonals, matrix $\boldsymbol{\Upsilon B}$ has full column rank. By (20), the positive definiteness of $\boldsymbol{U}-\boldsymbol{\mu}\boldsymbol{\mu}^T$ then readily implies that of $\boldsymbol{J}_{\boldsymbol{R}|\boldsymbol{q}_w}$.

C. Proof of Theorem 2

We say a link is a "bottleneck" if the aggregate rate through it is equal to its capacity. For a given $\boldsymbol{w}$, let $\mathcal{B}$ denote the set of bottleneck links. Denote $\boldsymbol{A}_B$ as the sub-matrix of $\boldsymbol{A}$ obtained by keeping only the rows that correspond to bottleneck links, and $\boldsymbol{q}_B$, $\boldsymbol{c}_B$, and $\boldsymbol{R}_B$ are the corresponding sub-vectors of $\boldsymbol{q}$, $\boldsymbol{c}$, and $\boldsymbol{R}$ for bottleneck links. Define the diagonal matrices $\boldsymbol{X}:=$ diag$(\boldsymbol{x})$, $\boldsymbol{W}:=$ diag$(\boldsymbol{w})$, $\boldsymbol{D}:=$ diag$(\boldsymbol{d})$, and $\bar{\boldsymbol{D}}:=$ diag$(\bar{\boldsymbol{d}})$ where $\bar{\boldsymbol{d}}:=\{\bar{d}_s,\forall s\}$. Recalling that all non-bottleneck links have zero queueing delays, we can rewrite (10) in the matrix form:

$$\boldsymbol{X}(\boldsymbol{A}_B^T\boldsymbol{q}_B+\boldsymbol{d})=\boldsymbol{w}. \qquad (22)$$

As with [22], suppose first that $\boldsymbol{w}$ is an "interior" point, where the set $\mathcal{B}$ remains unchanged within a small neighborhood such that the mapping $F:\boldsymbol{w}\to(\boldsymbol{x},\boldsymbol{q})$ is differentiable. Note that $\boldsymbol{A}_B^T\boldsymbol{q}_B+\boldsymbol{d}=\bar{\boldsymbol{d}}$. Differentiating both sides of (22) with respect to $\boldsymbol{w}$ yields:

$$\bar{\boldsymbol{D}}\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}+\boldsymbol{X}\boldsymbol{A}_B^T\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}=\boldsymbol{I} \qquad (23)$$

where $\boldsymbol{I}$ denotes identity matrix, and $\boldsymbol{J}$ denotes the Jacobian matrix (of $\boldsymbol{x}$ or $\boldsymbol{q}_B$ over $\boldsymbol{w}$). Multiplying both sides of (23) by $\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}$, we further have:

$$\boldsymbol{A}_B\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}=\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}. \qquad (24)$$

Partition the bottleneck-only routing matrix $\boldsymbol{A}_B$ and queueing delay vector $\boldsymbol{q}_B$ into two parts:

$$\boldsymbol{A}_B=\begin{bmatrix}\boldsymbol{A}_B^f\\ \boldsymbol{A}_B^w\end{bmatrix}, \qquad \boldsymbol{q}_B=\begin{bmatrix}\boldsymbol{q}_B^f\\ \boldsymbol{q}_B^w\end{bmatrix}$$

where ^f_B and ^w_B denote the parts related to the wired and wireless bottleneck links, respectively. For the wired bottleneck links, it holds $\boldsymbol{A}_B^f\boldsymbol{x}=\boldsymbol{c}_B$; thus $\boldsymbol{A}_B^f\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=0$. For wireless bottleneck links, $\boldsymbol{A}_B^w\boldsymbol{x}=\boldsymbol{R}_B$ implies:

$$\boldsymbol{A}_B^w\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=\boldsymbol{J}_{\boldsymbol{R}_B|\boldsymbol{w}}=[\boldsymbol{0}\ \ \boldsymbol{J}_{\boldsymbol{R}_B|\boldsymbol{q}_B^w}]\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}.$$

Then overall we have:

$$\boldsymbol{A}_B\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=\begin{bmatrix}\boldsymbol{A}_B^f\\ \boldsymbol{A}_B^w\end{bmatrix}\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=\begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{J}_{\boldsymbol{R}_B|\boldsymbol{q}_B^w}\end{bmatrix}\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}$$
$$:=\boldsymbol{N}\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}$$

It then follows from (24) that:

$$(\boldsymbol{N}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T)\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}=\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}.$$

Since $\boldsymbol{J}_{\boldsymbol{R}_B|\boldsymbol{q}_B^w}$ is positive definite from Lemma 1 (or it is $\boldsymbol{0}$ if there are no wireless bottlenecks), the matrix $\boldsymbol{N}$ is clearly positive semi-definite. As $\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T$ is positive definite; so is $\boldsymbol{N}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T$. It follows that:

$$\boldsymbol{J}_{\boldsymbol{q}_B|\boldsymbol{w}}=(\boldsymbol{N}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T)^{-1}\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}.$$

Substituting the latter into (23), we further obtain:

$$\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=\bar{\boldsymbol{D}}^{-1}(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{A}_B^T(\boldsymbol{N}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T)^{-1}\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}).$$

Use the convenient notation:

$$\boldsymbol{M}:=\boldsymbol{A}_B^T(\boldsymbol{N}+\boldsymbol{A}_B\bar{\boldsymbol{D}}^{-1}\boldsymbol{X}\boldsymbol{A}_B^T)^{-1}\boldsymbol{A}_B.$$

It is clear that the matrix $\boldsymbol{M}$ is positive semi-definite and $\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}=\bar{\boldsymbol{D}}^{-1}(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{M}\bar{\boldsymbol{D}}^{-1})$.

For the QUIC-TCP algorithms (15), consider a function:

$$Y(\boldsymbol{w})=\frac{1}{2}\sum_{s\in S}\left(\frac{v_s}{(w_s)^\rho}\right)^2. \qquad (25)$$

Define diagonal matrices $\boldsymbol{V}:=$ diag$(\boldsymbol{v})$ and $\boldsymbol{P}:=$ diag$(\boldsymbol{p})$. At an interior point $\boldsymbol{w}$, we have:

$$\begin{aligned}
\frac{d}{dt}Y(\boldsymbol{w}(t)) &= \sum_s\left(\frac{\partial Y}{\partial w_s}\frac{dw_s(t)}{dt}\right)\\
&= \sum_s\left[\left(\sum_{s'}\left(\frac{v_{s'}}{w_{s'}^\rho}\frac{\partial(v_{s'}/w_{s'}^\rho)}{\partial w_s}\right)\right)\frac{dw_s(t)}{dt}\right]\\
&= -\kappa\boldsymbol{v}^T\boldsymbol{W}^{-\rho}[\boldsymbol{W}^{-\rho}\boldsymbol{J}_{\boldsymbol{v}|\boldsymbol{w}}-\rho\boldsymbol{W}^{-\rho-1}\boldsymbol{V}]\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}^{-2\rho+1}\boldsymbol{v}\\
&= -\kappa\boldsymbol{v}^T[\boldsymbol{W}^{-2\rho}(\boldsymbol{I}-\boldsymbol{D}\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}})-\rho\boldsymbol{W}^{-2\rho-1}\\
&\qquad\times(\boldsymbol{W}-\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}-\boldsymbol{P})]\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}^{-2\rho+1}\boldsymbol{v}\\
&= -\kappa\boldsymbol{v}^T[\boldsymbol{W}^{-2\rho}(\boldsymbol{I}-\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{M}\bar{\boldsymbol{D}}^{-1}))-\rho\boldsymbol{W}^{-2\rho-1}\\
&\qquad\times(\boldsymbol{W}-\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}-\boldsymbol{P})]\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}^{-2\rho+1}\boldsymbol{v}\\
&= -\kappa\boldsymbol{v}^T[(1-\rho)\boldsymbol{W}^{-2\rho}(\boldsymbol{I}-\boldsymbol{D}\bar{\boldsymbol{D}}^{-1})\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}^{-2\rho+1}\\
&\qquad+\rho\boldsymbol{W}^{-2\rho-1}\boldsymbol{P}\boldsymbol{D}\bar{\boldsymbol{D}}^{-1}\boldsymbol{W}^{-2\rho+1}\\
&\qquad+\boldsymbol{W}^{-2\rho+1}\boldsymbol{D}\bar{\boldsymbol{D}}^{-2}\boldsymbol{M}\bar{\boldsymbol{D}}^{-2}\boldsymbol{D}\boldsymbol{W}^{-2\rho+1}]\boldsymbol{v}. \qquad (26)
\end{aligned}$$

Since $(\boldsymbol{I}-\boldsymbol{D}\bar{\boldsymbol{D}}^{-1})$ is a diagonal matrix with nonnegative entries, $\boldsymbol{M}$ is positive semi-definite, and all $\boldsymbol{W}$, $\boldsymbol{D}$, $\bar{\boldsymbol{D}}$ and $\boldsymbol{P}$ are diagonal matrices with positive diagonal entries, it is easy to see that the whole matrix inside the square bracket of (26) is positive definite for $0\leq\rho\leq 1$. This implies that $dY(\boldsymbol{w}(t))/dt<0$, i.e., $Y(\boldsymbol{w}(t))$ is strictly decreasing in t, at all interior points, unless $\boldsymbol{v}=0$.

For the "boundary" (i.e., non-interior) points, we can extend the definition of $\boldsymbol{J}_{\boldsymbol{x}|\boldsymbol{w}}$ as a function of direction d, since the right-hand directional derivative of $\boldsymbol{x}(\boldsymbol{w})$ is always well-defined for an arbitrary direction d. With this extension, we can follow the similar lines in the proof of [22, Theorem 5] to argue that $Y(\boldsymbol{w}(t))$ is strictly decreasing in t at boundary points, unless $\boldsymbol{v} = 0$.

We have shown that $Y(\boldsymbol{w}(t))$ is a nonnegative Lyapunov function with a globally negative time derivative. The unique equilibrium $\boldsymbol{v} = 0$ of the system (15) is thus globally asymptotically stable for $0 \leq \rho \leq 1$.

REFERENCES

[1] M. Chiang, S. Low, A. Calderbank, and J. Doyle, "Layering as optimization decomposition," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[2] Y. Yi and M. Chiang, "Stochastic network utility maximization: a tribute to Kelly's paper published in this journal a decade ago," *Eur. Trans. Telecommun.*, vol. 19, no. 4, pp. 421–442, 2008.

[3] X. Lin and N. Shroff, "The impact of imperfect scheduling on cross-layer rate control in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 302–315, Apr. 2006.

[4] M. Neely, E. Modiano, and C. Li, "Fairness and optimal stochastic control for heterogenous networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 396–409, Apr. 2008.

[5] Y. Yu and G. Giannakis, "Joint congestion control and OFDMA scheduling for hybrid wireline-wireless networks," in *Proc. 2007 INFOCOM*, pp. 973–981.

[6] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 36, no. 12, pp. 1936–1948, Dec. 1992.

[7] C. Joo, X. Lin, and N. Shroff, "Understanding the capacity region of the greedy maximal scheudling algorithm in multi-hop wireless networks," in *Proc. 2008 INFOCOM*, pp. 1103–1111.

[8] M. Leconte, J. Ni, and R. Srikant, "Improved bounds on the throughput efficiency of greedy maximal scheudling in wireless networks," in *Proc. 2009 MobiHoc*, pp. 165–174.

[9] X. Wu and R. Srikant, "Scheduling efficiency of distributed greedy scheduling algorithms in wirleess networks," in *Proc. 2006 INFOCOM*, pp. 1–12.

[10] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 3, pp. 960–972, June 2010.

[11] L. Jiang, D. Shah, J. Shin, and J. Walrand, "Distributed random access algorithm: scheduling and congestion control," *IEEE Trans. Inf. Theory*, vol. 56, no. 12, pp. 6182–6207, Dec. 2010.

[12] J. Liu, Y. Yi, A. Proutiere, M. Chiang, and H. Poor, "Towards utility-optimal random access without message passing," *Wireless Commun. Mobile Comput.*, DOI:10.1002/wcm.000, 2009.

[13] S. Floyd, "High speed TCP for large congestion windows," RFC 3649, Dec. 2003.

[14] T. Kelly, "Scalable TCP: improving performance in highspeed wide area networks," *ACM SIGCOMM Comput. Commun. Review*, vol. 33 no. 2, Apr. 2003.

[15] C. Fu and S. Liew, "TCP Veno: TCP enhancement for transmission over wireless access networks," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 2, pp. 216–228, Feb. 2004.

[16] S. Mascolo, C. Casetti, M. Gerla, M. Sanadidi, and R. Wang, "TCP Westwood: bandwidth estimation for enhanced transport over wireless links," in *Proc. 2001 Mobicom*, pp. 287–297.

[17] K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless IP communications," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 4, pp. 747–756, May 2004.

[18] L. Brakmo and L. Peterson, "TCP Vegas: end-to-end congestion avoidance on a global Internet," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 8, pp. 1465–1480, Oct. 1995.

[19] S. Hagag and A. El-Sayed, "Enhanced TCP westwood congestion avoidance mechanism (TCP WestwoodNew)," *Intl. J. Comput. Applications*, vol. 45, no. 5, pp. 21–29, May 2012.

[20] M. Podlesny and C. Williamson, "Improving TCP performance in residential broadband networks: a simple and deployable approach," *ACM SIGCOMM Comput. Commun. Review*, vol. 42, no. 1, pp. 61–68, Jan. 2012.

[21] F. Ren and C. Lin, "Modeling and improving TCP performance over cellular link with variable bandwidth," *IEEE Trans. Mobile Comput.*, vol. 10, no. 8, pp. 1057–1070, Aug. 2011.

[22] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[23] D. Wei, C. Jin, S. Low, and S. Hedge, "FAST TCP: motivation, architecture, algorithms, performance," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1246–1259, Dec. 2006.

[24] X. Wang, Z. Li, and N. Gao, "Joint congestion control and wireless-link scheduling for mobile TCP applications," in *Proc. 2011 Globecom*.

[25] F. Kelly, "Stochastic models of computer communication systems," *J. Royal Stat. Soci.*, vol. 47, no. 3, pp. 379–395, 1985.

[26] X. Wang and K. Kar, "Throughput modelling and fairness issues in CSMA/CA based ad-hoc networks," in *Proc. 2005 INFOCOM*, vol. 1, pp. 23–34.

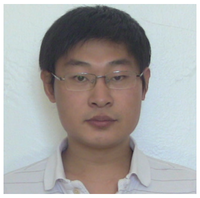
[27] F. Kelly and A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Opl. Res. Soci.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Uiversity Press, 2004.

[29] A. Stolyar, "Maximizing queueing network utility subject to stability: greedy primal-dual algorithm," *Que. Syst.*, vol. 50, pp. 401–457, 2005.

[30] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling*. Wiley, 1991.

Xin Wang (SM'09) received the B.Sc. degree and the M.Sc. degree from Fudan University, Shanghai, China, in 1997 and 2000, respectively, and the Ph.D. degree from Auburn University, Auburn, AL, in 2004, all in electrical engineering. From September 2004 to August 2006, he was a Postdoctoral Research Associate with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis. In August 2006, he joined the Department of Computer & Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, as an Assistant Professor, and then an Associate Professor from August 2010. He is now a Professor with the Department of Communication Science and Engineering, Fudan University, China. His research interests include stochastic network optimization, energy-efficient communications, cross-layer design, and signal processing for communications.



Zhaoquan Li received the B.Sc. degree from Shandong University, Jinan, Shandong, China, in 2001, and the M.Sc. degree from Shanghai Academy of Spaceflight Technology, Shanghai, China, in 2004, respectively, both in electrical engineering. From December 2004 to October 2008, he was a DSP engineer in Huawei Technologies Co. Ltd. He is currently working toward the Ph.D. degree in the Department of Computer & Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL. His research interests include stochastic resource allocation and wireless networks.

Jie Wu (F'09) is the chair and a Laura H. Carnell Professor in the Department of Computer and Information Sciences at Temple University. Prior to joining Temple University, he was a program director at the National Science Foundation and Distinguished Professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly published in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON SERVICE COMPUTING, and the *Journal of Parallel and Distributed Computing*. Dr. Wu was general co-chair/chair for IEEE MASS 2006 and IEEE IPDPS 2008 and program co-chair for IEEE INFOCOM 2011. Currently, he is serving as general chair for IEEE ICDCS 2013 and ACM MobiHoc 2014, and program chair for CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a CCF Distinguished Speaker and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.