

Design and Analysis of a Flow Control Scheme over Wireless Networks[†]

Alessandro Abate^{1,*}Minghua Chen² Yue Wang³ Avideh Zakhor⁴ Shankar Sastry⁴

¹ *Delft Center for Systems and Control, TU Delft - Delft University of Technology, Delft, The Netherlands*
a.abate@tudelft.nl

² *Department of Information Engineering, The Chinese University of Hong Kong, PR China*
minghua@ie.cuhk.edu.hk

³ *School of Information, Central University of Finance and Economics, PR China*
yue.wang@cufe.edu.cn

⁴ *Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, USA*
{avz,sastry}@eecs.berkeley.edu

SUMMARY

This paper studies the problem of congestion control on wireless networks. A dynamical model for the end-to-end network flow control is proposed, which exploits the differentiation between congestion loss and physical channel error loss. The introduction of a specific wireless model is motivated by the distinctive presence of channel errors, which are often not known exactly. We assume that each wireless link is associated with an additional error function that depends on the current flow along the link and which accounts for the packet loss rate caused by the physical channel. This leads to a new dynamic flow control scheme that naturally extends a known mathematical model for the fluid flow approximation of the Transmission Control Protocol for wireline networks. The main objective of this work is to study the dynamical properties of the new model: we analyze its nonlinear dynamics, derive its stability properties, and study its robustness to delays. We also present and discuss some ns-2 simulations of its dynamics. This work additionally looks at the actual implementation of the proposed scheme: by requiring only modifications to the application layer rather than the transport one, no alterations to the network infrastructure or transport protocols are needed. The article argues that the new scheme appears to be not only theoretically meaningful but also practically relevant for an application layer implementation.

Copyright © 2011 John Wiley & Sons, Ltd.

KEY WORDS: Nonlinear systems, stability, robustness, systems with delays, transport protocols, wireless networks.

*Correspondence to: Faculty of 3mE, Mekelweg 2, 2628CD, Delft, The Netherlands.

[†]This work has been supported by the NSF grants CCR-0225610 and ANI-9905799; by the AFOSR contract F49620-00-1-0327; the European Commission under the MoVeS project, FP7-ICT-2009-257005; by the European Commission under Marie Curie grant MANTRAS 249295; by NWO under VENI grant 016.103.020; by the Discipline Construction Foundation of Central University of Finance and Economics, China; by Competitive Earmarked Research Grants (Project 411008, 411209, and 411010) and an Area of Excellence Grant (Project AoE/E-02/08), all established under the University Grant Committee of the Hong Kong, China; and by a Direct Grant (Project 2050397) of The Chinese University of Hong Kong.

1. Introduction

Transmission Control Protocol (TCP) has been considerably successful on the wireline Internet since its first implementation by Jacobson [25] in 1988. TCP Reno, the most widely used TCP version today, is a window-based scheme that adapts itself according to the measured conditions of the network. Specifically, in the congestion avoidance stage, it increases its window size by a constant if no packet is lost in the previous round trip time, while it halves it if there is any packet loss during the previous round trip time. This latter case indicates the presence of information overflow on the network, since the key assumption TCP relies on is that packet loss is a sign of congestion. In wireless networks however, packet loss can also be caused by physical channel errors, due to message corruption, presence of noise or interference. Thus, the assumption holding for the wireline case breaks down, resulting in underutilization of the wireless bandwidth [24]. Similar observations hold for TCP-friendly schemes, e.g. TCP-Friendly Rate Control (TFRC) [12,21] or other modifications of the Reno protocol (for instance NewReno [22]), since they share the same key assumption as TCP.

There have been a number of efforts to improve the performance of TCP (or TFRC) over wireless, which exploit information from the network [4,5,8,10,16–18,24,35,47,48,52,54,60]. For instance, Snoop [5] is a TCP-AWARE link layer approach which suppresses acknowledgement packets (ACK) from the TCP receiver, and does local retransmissions when a packet is corrupted by wireless channel errors. More generally, local retransmissions are widely used in today's wireless system design [19]. However, even with local retransmissions, there may still be packet losses caused by wireless error: in either wireless local area network (WLAN) or cellular systems the number of local retransmissions is usually small and cannot be arbitrarily increased to make the wireless link free of packet-losses because it would incur a large end-to-end delay variation. This would cause TCP in the transport layer to suffer from timeout, which affects the throughput much more severely than wireless packet loss. Explicit Loss Notification (ELN) [4] can be applied to notify the TCP sender when a packet loss is caused by the physical channel rather than by the presence of congestion. Furthermore, end-to-end statistics can also be used to detect congestion when a packet is lost [8,35,48,52,54]. For instance, by examining trends in the one-way delay variation, one could interpret loss as a sign of congestion if the one-way delay is increasing, and a sign of wireless channel error otherwise [8,48]. From another perspective, TCP Westwood is a technique that tries to counteract wireless lossy links using end-to-end bandwidth estimation [41]. Vegas [9] and, more recently, FAST [43,56] rely on queuing delays as the congestion signal. The principle is nicely tailored to wireless networks, however it can suffer from underutilization due to high levels of noise in the delay measurement [37]. Additionally, [40] investigates pricing strategies coming from optimization problems in a wireless setting. A survey of the cited end-to-end approaches is contained in [10]. All these methods either hide end-hosts from packet loss caused by wireless channel error, or provide end-hosts with the ability to distinguish between packet loss caused by congestion, and that caused by wireless channel error. The disadvantage of these schemes is that they either require support from the network infrastructure or require modifications to the transport layer protocols. As such they are less likely to be widely deployed.

A recent approach called MULTFRC [12], which is based on modifications of TFRC only to the application layer, puts forward a new method to improve the performance in wireless networks, and is also applicable to TCP. MULTFRC opens an appropriate number of connections depending on the observed round trip time, so as to fully utilize the wireless

bandwidth, while keeping the packet loss rate and the round trip time at a minimum. An important and challenging issue to be addressed in the MULTFRC framework is the study of its dynamical properties. In particular, how should the problem of flow control over wireless network be systematically redefined? Are there any end-to-end stable and robust application layer schemes that address the problem, without changing the network infrastructure or the protocol stack?

Recently there has been a great deal of research activities on algorithm synthesis and analysis for decentralized, end-to-end network flow control. A widely recognized setting for the wireline case has been introduced by Kelly *et al.* in the seminal work [29], and is based on a fluid-flow approximation of the TCP Reno in its congestion avoidance stage. It associates a utility function to each flow, a cost function to each resource, and maximizes the aggregate net system utility function. Under this framework, flow control schemes can be viewed as algorithms to compute the optimal solution to this maximization problem. The work proposes two complementary flow control algorithms, the *primal* and the *dual*. In primal algorithms, the users adapt their sending rates dynamically based on the costs that incur along the path through the network, while the routers determine their prices directly from the arrival rates at the links according to a static law [2, 3, 26, 29, 32, 55]. In dual algorithms, on the other hand, the routers adapt the prices dynamically based on the link rates, and the users select a static law to determine the source rates directly from the prices along the path and the source parameters [29, 39, 43, 57]. These two approaches can be also combined [28, 39, 43, 53].

The major difference between wireline and wireless networks is that in the latter the physical layer is a major source of packet loss. The TCP schemes devised for the wireline case are documented to underperform on wireless networks [24]. In this paper, we shall analyze a new scheme for the wireless scenario that mitigates this underutilization issue. The new model is related to that in [29] for the wireline case, and by explicitly including the physical error loss aims at improving the TCP scheme over the network utilization. More precisely, in the model we assume that each wireless link, which is associated with a fixed capacity, is enhanced with an additional error function as an additive term that depends on the current flow along the link. This error function accounts for the packet loss rate caused by the physical channel. We then introduce a new dynamic flow control scheme that is based on feedback from the channel, and prove that the new system will perform as optimally as the corresponding one for the wireline case. The main objective of the work is to study its dynamical properties and to offer a theoretical support for its actual application, which has been investigated in [14, 15]. By exploiting the heterogeneous time scales of the system, we prove global stability of the new model. Furthermore, we derive some conditions for local stability of the scheme in the presence of delays. We provide some numerical results in ns-2 to validate our analysis.

We then interpret the proposed modification of the fluid-flow model from a practical perspective and justify the claim that no actual change in the corresponding TCP scheme for the wireline case or in the underlying infrastructure is required to implement it. We in fact suggest that the new scheme can be practically implemented by opening parallel connections by the user, each of which employs the original TCP protocol. We remark that the possibility of relying on the existing TCP protocol sets this work apart from the adjacent literature. In a related work [14], an actual implementation of a protocol derived from the ideas here presented is shown. This approach also connects back to the techniques proposed in [12], and revisits them within a systematic framework. The presence of channel errors is the cause of imperfect feedback from the network to the users. These errors prevent the exact measurement

of the congestion status on the network. This issue is overcome by exploiting other available information as feedback coming from the network. This suggests a practical way of turning our proposed schemes into actual protocols. A number of schemes following similar ideas have been proposed recently [13, 14], which further argues in favor of the implementability of the model introduced and studied in this work.

The paper introduces the problem of congestion control for wireline networks and the extension to the wireless case in Section 2. In Section 3 the proposed modification is introduced and described. The main theoretical contributions of the article are contained in Section 4, where the dynamical properties of the model are analyzed. A discussion on its practical implementability is contained in Section 5. Finally, simulations in ns-2 are presented in Section 6. Future research directions conclude the contribution.

2. Problem Formulation

In this section we describe the dynamical model of the well known general flow control problem first introduced in [29]. In doing so, we adhere to the notation in [29]. Starting from the wireline scenario, we motivate and build up the extensions for the more challenging wireless case.

2.1. A fluid-flow Model as a primal algorithm for Congestion Control of Wireline Networks

A communication network is described via a set J of *resources*, i.e. links, and a set R of *users*, i.e. sender-receiver pairs, or routes through the network. Each $j \in J$ has a finite capacity $C_j < \infty$. The network interconnections are described via a constant 0-1 routing matrix $A = (a_{jr}, j \in J, r \in R)$, where $a_{jr} = 1$ if $j \in r$ and is equal to zero otherwise.

A fluid-flow, continuous-time model for the TCP scheme has been proposed in order to facilitate the analysis of the properties of the protocol [29]. To each user r a sending rate $x_r \geq 0$ and a utility function $U_r(x_r)$ are associated. Users are supposed to behave independently. $U_r(x_r)$ is assumed to be increasing, strictly concave and continuously differentiable over the range $x_r \geq 0$, according to the hypothesis of elastic traffic in [50]. Assume the utilities are additive, so that the aggregate benefit for the entire system is $\sum_{r \in R} U_r(x_r)$. Define the cost incurred at link j as $P_j(\cdot)$. The exchange of information between users over the links can be interpreted as a concave maximization problem [38, 39] depending on the aggregate utility functions for the rates and on the costs at the links:[†]

$$\max \sum_{r \in R} U_r(x_r) - \sum_{j \in J} P_j \left(\sum_{s: j \in s} x_s \right), \quad (1)$$

where the cost functions $P_j(\cdot)$, similarly to the one introduced in [32], are defined as

$$P_j(y) = \int_0^y p_j(z) dz. \quad (2)$$

The terms $p_j(y)$ can be interpreted as “prices” at the link and are assumed to be non-negative, continuous and increasing functions; they represent a congestion measure and, as can be

[†]The abuse in notations on the indices of the sum in the argument of the prices adheres to [29].

inferred from their structure, they have a local dependence on the aggregate rate passing through the link. With these assumptions on $p_j(z)$, the function $P_j(y)$ is strictly convex. As in [28], in this paper we choose the following “packet loss rate”:

$$p_j(y) = \frac{(y - C_j)^+}{y}. \quad (3)$$

Fig. 1(left) shows an example of $p_j(\cdot)$ as packet loss rate on a link with capacity of 1 kilobits-per-second (kbps). The end-to-end packet loss rate for user r is $1 - \prod_{j \in r} (1 - p_j(\sum_{s:j \in s} x_s))$, which is approximately $\sum_{j \in r} p_j(\sum_{s:j \in s} x_s)$ when $p_j(\sum_{s:j \in s} x_s)$ is small for each j . In the rest of this paper we assume the price function to be small enough so that $\sum_{j \in r} p_j(\sum_{s:j \in s} x_s)$ is the end-to-end packet loss rate for any user r .

Flow control can also be examined from a dynamical perspective, studying its evolution around an equilibrium that is the solution of (1). For the logarithmic utility function $U_r(x_r) = w_r^o \log x_r$, where w_r^o is a weight and can be understood as pay per unit time, Kelly *et al.* [29] have shown that both primal and dual algorithms can be used to solve the optimization problem (1). In the rest of this paper, we focus on the primal algorithm only, since primal algorithms result in an end-to-end based TCP-like behavior. The rate control scheme has the following shape:

$$\frac{d}{dt} x_r(t) = k_r \left(w_r^o - x_r(t) \sum_{j \in r} p_j \left(\sum_{s:j \in s} x_s(t) \right) \right), r \in R, \quad (4)$$

with k_r a positive scale factor affecting the adaptation rate. The constant w_r^o , rather than being thought of as another adaptation rate, can be physically interpreted as the number of connections that the user establishes within the network. To see that, assume w_r^o is set to w_r^i in (4) to control the rate of connection when user r opens one (the i^{th}) connection. Then, opening $N - 1$ additional connections can be thought of adding $N - 1$ additional users along the same path as user r , and associating the same source parameter w_r^i to control their rate. Thus, the aggregate rate of these N connections, x_r , is the sum of the rates of these N users, and is adjusted in a way shown in (4) but with $w_r^o = N \cdot w_r^i$. Hence the source parameter w_r^o is proportional to the number of connections opened by user r , and changing the number of connections is equivalent to adjusting w_r proportionally. Later we shall further exploit this interpretation. As already discussed, the congestion signal, i.e. the packet loss rate, depends on the sum of the prices along all the links that are crossed by the user. Therefore, the system exploits only aggregate, end-to-end information from the network. The system in (4) corresponds to the TCP-like additive increase and multiplicative decrease algorithm. A wealth of research on the dynamical and robustness properties of the scheme in (4) is summarized in the contributions [26, 29, 55].

If the end users and the links in the network deploy the algorithm in (4), the unique, globally asymptotically stable rates of the entire network, denoted vectorially by $\mathbf{x}^o = (x_r^o, r \in R)$, are given by

$$x_r^o = \frac{w_r^o}{\sum_{j \in r} p_j \left(\sum_{s:j \in s} x_s^o \right)}, \quad r \in R. \quad (5)$$

This unique solution is also optimal in the sense that the network bottlenecks are fully utilized, the total net utility is maximized, and the users are proportionally fair to each other [29].

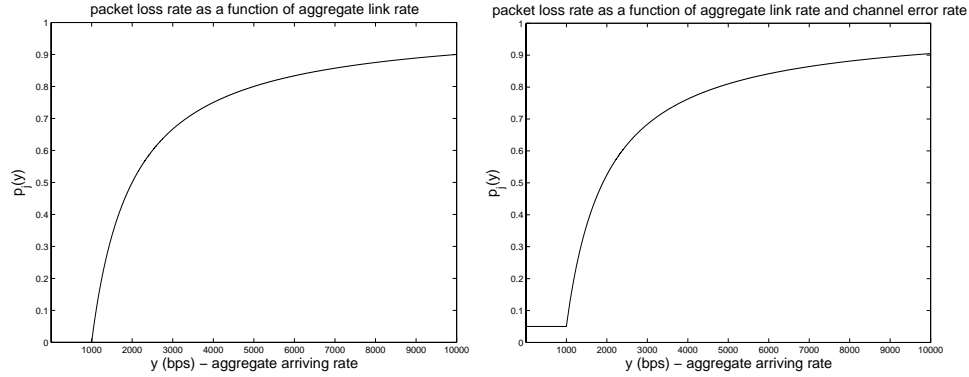


Figure 1. Examples of the price function as packet loss rate on a link with capacity 1 kbps. The error rate is taken to be a constant, while in general it depends on the aggregate flow rate, as in Eqn. (6).

2.2. Extending the Model to the Wireless Scenario

Wireless channels are affected by errors, due to interference and noise. In a TCP-like setting this directly influences the end-to-end packet loss rate at each link, i.e. the price sensing by end-hosts via end-to-end measurements. We encompass this fact within a new price function for each link $j \in J$:

$$q_j \left(\sum_{s:j \in s} x_s(t) \right) = p_j \left(\sum_{s:j \in s} x_s(t) \right) + \epsilon_j \left(\sum_{s:j \in s} x_s(t) \right) \geq p_j \left(\sum_{s:j \in s} x_s(t) \right). \quad (6)$$

This function accounts for both the congestion measure, i.e. the dependence on the term $p_j(\cdot)$, as well as the channel error $\epsilon_j(\cdot)$. The function $\epsilon_j(\cdot)$ is assumed to be monotonically increasing with respect to its variable, and accounts for the probability of errors on link j .

Another feature of wireless channels is their time-varying nature. The introduced model inherits the assumption of constant capacity for each link from the framework in [29], which was developed for wireline networks. This point is key in technically proving fundamental results in this paper and its extension to the time-varying case continues to be an open problem. Our model also deals with a constant routing matrix A . These hypotheses are reductive in general, especially for some networks such as ad-hoc or sensor networks: this is because of the actual time-varying nature of wireless channels (e.g., due to the fading characteristic of their structure), as well as because of the observed dependence of the capacity on the current information flow along each link. However, in practice the assumptions can be verified in the study of certain classes of static wireless access cellular networks, such as WLAN, CDMA or WiMax. Furthermore, the constant capacity assumption can practically hold within the relatively short time scales we are interested in. This premise has also been adopted by other practical work that stem from similar ideas, and the positive performance over both 802.11b access networks and commercial cellular networks again hints to its validity [14, 15, 58]. We thus realistically expect that the performance of the schemes will also be reasonable in cases where such assumptions might not strictly hold.

Fig. 1(right) shows an example of $q_j(\cdot)$ as packet loss rate on a link with capacity of 1 kbps, a fixed $\epsilon_j = 0.05$, while $p_j(\cdot)$ is shown in Fig. 1(left). When the wireless link is not congested,

$q_j(\cdot)$ becomes equal to $\epsilon_j(\cdot)$ since all the packet loss is caused by channel error. When the link is congested and packets are dropped at the router, $q_j(\cdot)$ gradually increases, similar to $p_j(\cdot)$ in Fig. 1(left). Notice that in general the quantity $q_j(\cdot)$ is monotonically increasing as a function of its variable, because of a similar assumption on the two terms that compose it.

The TCP model (4) then depends on this new function $q_j(\cdot)$. It is straightforward to calculate the equilibrium of this new dynamic relation, $\mathbf{x}^* = (x_r^*, r \in R)$:

$$x_r^* = \frac{w_r^o}{\sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s^* \right)}, \quad r \in R. \quad (7)$$

Comparing the stable point \mathbf{x}^* in (7) with \mathbf{x}^o in (5), which is the optimal solution for the optimization problem shown in Eqn. (1), we have $x_r^* < x_r^o$ if $\sum_{j \in r} \epsilon_j(\cdot) > 0$. Since the optimal solution for problem in (1) is unique, the equilibrium point \mathbf{x}^* is then a suboptimal solution. Interpreting this fact through an underlying optimization problem, as in (1), shows that the new equilibrium will be suboptimal, as long as there is nonzero channel error, and hence the network bottleneck could be underutilized. This motivates and defines the flow control problem over wireless, which we address in this paper by the introduction of an enhancement to the wireline scheme, as described in the following section.

3. A New Flow Control Scheme for Wireless Networks

In this section we introduce a new scheme as an enhancement to the TCP protocol model, aimed at compensating the suboptimality of the equilibrium point of the rates when the wireline scheme in (4) is employed on wireless links.

As argued later, an important advantage is that it works at the application layer level, thus not requiring any modifications of the infrastructure or the transport layer protocols of the network. As anticipated on page 5, the underlying rationale for the described approach is that the parameter w_r for each user r can be directly modified.

Let $w_r(t)$ denote the source parameter for user r , which is assumed to be a function of time. It is desirable to adjust $w_r(t)$ slowly so that (a) the number of connections is also adjusted slowly, and (b) the sensitivity to measurement error in packet loss rate is decreased. We consider the case where $w_r(t)$ is dynamically updated as follows: $\forall r \in R$,

$$\frac{d}{dt} w_r(t) = c_r \left(w_r^o - w_r(t) \frac{\sum_{j \in r} p_j \left(\sum_{s: j \in s} x_s(t) \right)}{\sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t) \right)} \right), \quad (8)$$

where $p_j(\cdot)$ and $q_j(\cdot)$ are defined in (3) and (6) respectively. The source rate for user r then is given by:

$$\frac{d}{dt} x_r(t) = k_r \left(w_r(t) - x_r(t) \sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t) \right) \right). \quad (9)$$

The intuition behind this approach is that when noise is significant and $q_j(t)$ is large, $w_r(t)$ in (8) is increased so as to compensate for the $x_r(t) \sum_{j \in r} q_j(t)$ term in (9), and hence to properly control $x_r(t)$. It is easy to see that the unique equilibrium point referring to the rates for the system (8)-(9) is \mathbf{x}^o , as given by Eqn. (5). It is also the solution for the optimal problem in

Eqn. (1), studied by Kelly *et al.* for wired networks. In the following, we shall denote the equilibrium referring to the connection variables as \mathbf{w}^* .

The composite system built upon Eqn. (9) for the rates x_r and Eqn. (8) for w_r , is a nonlinear, coupled, and multivariable one. Also, note that the two equations (8)-(9) are not exactly symmetric. In Section 4 we analyze in detail stability and delay properties of this system.

As already mentioned in Section 1, existing approaches [4,5,8,10,16–18,24,35,47,48,52,54,60] have been translated into a number of practical schemes to provide user r with the correct price $\sum_{j \in r} p_j(t)$, or its estimated value. End users can then modify the source rate control law to react to $\sum_{j \in r} p_j(t)$ rather than to the noisy $\sum_{j \in r} q_j(t)$. Hence, in most existing techniques the problem of flow control over wireless is reduced to the problem of flow control over wireline networks by directly synthesizing a communication protocol. Our approach, on the other hand, adjusts w_r to solve the problem. The intuition is clearly seen from Eqn. (7), in that if the numerator w_r^o is changed to a larger value, then it is possible to preserve $x_r^* = x_r^o$. It is important to notice that this scheme can be easily implemented by interpreting the quantity w_r as the number of connections which an application opens in an actual network: from a fluid-flow perspective, the new scheme suggests a way to adjust this term. Therefore, it is an *application layer*-based approach and is thus easy to deploy, since it does not require changes to the infrastructure of the network or to its transport layer protocols. From a performance standpoint, it achieves the optimal total net utility. We stress that in this approach we assume the end user r to have access to both the correct price function $\sum_{j \in r} p_j(t)$ and the noisy one $\sum_{j \in r} q_j(t)$ – we will revisit this important assumption in the light of the proposed model in Section 5.

4. Analysis of the New Scheme

In this Section, some dynamical properties of the scheme introduced in Section 3 are studied. The focus is initially on global stability. We then provide proper conditions to retain local stability in the presence of delays. Throughout the paper, given a vector $\mathbf{v} = [v_1, \dots, v_n]^T$, $\text{diag}(v_i)$ (or $\text{diag}(\mathbf{v})$) denotes the $n \times n$ diagonal matrix with elements taken in order from vector \mathbf{v} .

4.1. Global Stability

Global stability ensures that in an arbitrarily large network with any number of users and at any condition, the rates $x_r, r \in R$ converge to the optimal finite point x_r^o . This means the rates are appropriately controlled according to the network congestion conditions such that bottlenecks are fully utilized and, at the same time, congestion collapse [25, 53] is avoided. Motivated by the work in [29], we prove here the global stability of the interconnections by Lyapunov arguments. While in this manuscript we have adopted a specific form for the price function, i.e. Eqn. (3), we in fact do not make any assumptions in the stability proofs, other than the basic ones for general price functions in Sec. 2.1. Hence our stability results apply to more general models.

The Two Time Scales Assumption

In this subsection we will show that the dynamic update system, explicitly defined by (8)-(9), is endowed with some stability properties. To begin with, we make the following key observation: *the dynamics corresponding to (8) and (9) evolve in two different time scales: the first in a faster one, while the second in a slower one.* This assumption reflects the fact that in practice $w_r(t)$, interpreted as the number of connections of a user within the network, should be adjusted much more slowly than the source rate $x_r(t)$. We will further discuss this assumption in Section 5. After vectorizing (8)-(9), we can then rewrite the coupled equations as the pair:

$$\eta \dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}, \mathbf{w}); \quad (10)$$

$$\dot{\mathbf{w}}(t) = \mathbf{G}(\mathbf{x}, \mathbf{w}). \quad (11)$$

The term η is a very small number which we introduce to reframe our current setting within the classical *small-perturbations* framework [30, 49]; this change can be practically obtained through rescaling of the constant vector parameters $k_r, c_r, r \in R$. Intuitively, since \mathbf{w} changes much more slowly than \mathbf{x} , it can be considered a fixed parameter within equation (10). On the other hand, (11) may be thought of evolving once (10) has reached its equilibrium configuration. More rigorously, by formally applying the two time scales decomposition, we can decouple the entire system into two subsystems: namely, $\forall r \in R$ at fast time scale, a *boundary layer system*, which is obtained through a rescaling of time, $\tau = t/\eta$ and letting $\eta \rightarrow 0$:

$$\begin{cases} \dot{x}_r(t) = k_r \left(w_r(t) - x_r(t) \sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t) \right) \right), \\ w_r(t) = \text{constant}; \end{cases} \quad (12)$$

furthermore, $\forall r \in R$ at slow time scale, a *reduced order system*, which is obtained letting $\eta \rightarrow 0$:

$$\begin{cases} x_r(t) = \frac{w_r(t)}{\sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t) \right)}, \\ \dot{w}_r(t) = c_r \left(w_r^o - w_r(t) \frac{\sum_{j \in r} p_j \left(\sum_{s: j \in s} x_s(t) \right)}{\sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t) \right)} \right). \end{cases} \quad (13)$$

We leverage the notion of semi-global exponential stability (as in [49, Chapter 5]) to describe the asymptotic properties of the equilibrium of the overall system (10)-(11) as follows:

Theorem 1. *For the singularly perturbed system (10)-(11) the following holds:*

- Eqn. (10) has a region of equilibrium points $\mathbf{F}(\mathbf{x}, \mathbf{w}) = \mathbf{0}$ that identifies a manifold $\mathbf{w} = \mathbf{h}(\mathbf{x})$;
- Within this manifold, Eqn. (11) has a unique global equilibrium which is the solution of $\mathbf{G}(\mathbf{x}, \mathbf{h}(\mathbf{x})) = \mathbf{0}$. This composition is the reduced order system, described in (13);
- The equilibrium manifold for the boundary layer system in (12) is semi-globally exponentially stable;
- For the reduced order system in (13), $\dot{\mathbf{w}} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \dot{\mathbf{x}} = \mathbf{G}(\mathbf{x}, \mathbf{h}(\mathbf{x}))$ is also semi-globally exponentially stable.

Hence, by the stability properties of singular perturbation non-linear systems, there exists an $\eta^* > 0$ such that, $\forall \eta \leq \eta^*$, the global equilibrium point of the composite system is semi-globally exponentially stable.

Proof: Refer to Appendix I.1.

The main point of the theorem is that, under the assumption that the source rates $x_r(t)$ are adjusted much faster than the connection terms $w_r(t)$ (practically implied by the condition $c_r \ll k_r, \forall r \in R$), and assuming that both entities are bounded in value, the network exponentially converges to its equilibrium point regardless of its initial condition and the size of the system. We remark that it can be shown that the unique equilibrium of the proposed scheme solves the optimal problem in Eqn. (1), studied by Kelly *et al.* for wired networks.

4.2. Stability in the presence of Delays

It is also important to investigate whether our proposed control system is stable when there is bidirectional delay (sender-receiver) between the entities that generate the feedback signals from the network, i.e. $p_j(t)$ and $q_j(t)$, and the users that receive the feedback signals. We investigate the delay stability locally, i.e. in a region around equilibrium point for the rates \mathbf{x}^o —this will enable us to make use of arguments in the frequency domain. Focusing for the sake of simplicity on the dynamics of the wired system (4), delays can be modeled as follows [55]:

$$\frac{d}{dt}x_r(t) = k_r \left(w_r^o - x_r(t - T_r) \sum_{j \in r} \mu_j(t - d_2(j, r)) \right), \quad (14)$$

where

$$\mu_j(t) = p_j \left(\sum_{s: j \in s} x_s(t - d_1(j, s)) \right), \quad \forall j \in J,$$

and

$$d_1(j, r) + d_2(j, r) = T_r, \quad \forall r \in R.$$

Here $d_1(j, r)$ is the forward delay from the sender of route r to link j , and $d_2(j, r)$ is the backward delay from link j to the receiver r . T_r is thus the *round trip time* on route r . Here we assume that T_r is fixed: this is quite a common assumption, at least for the time scales we are interested in, as also argued in [26, 29]. The following theorem provides a sufficient condition on each end user for the entire network to be robust to *heterogeneous delay*, that is, different T_r values for different users r .

Theorem 2. *The system (8)-(9), affected by heterogeneous delay $T_r, r \in R$, is locally stable if the following two conditions hold, $\forall r \in R$:*

$$k_r \left(\sum_{j \in r} q_j + \sum_{j \in r} q'_j \sum_{s: j \in s} x_s^o \right) < \frac{\pi}{2T_r}; \quad (15)$$

$$c_r \frac{\left(\sum_{j \in r} p_j + \sum_{j \in r} p'_j \sum_{s: j \in s} x_s^o \right)}{\min_{r \in R} \sum_{j \in r} q_j} < \frac{\pi}{2T_r}, \quad (16)$$

where p_j, q_j, p'_j, q'_j denote respectively the values of $p_j(\cdot), q_j(\cdot)$ and the derivatives of $p_j(\cdot), q_j(\cdot)$, all evaluated at the corresponding equilibrium point.

Proof: Refer to Appendix I.2.

Interpreting k_r as the speed at which user r adapts its rate x_r , similar to the results in [26, 55], Theorem 2 suggests that user r should change its rate within a time interval that depends on the round trip time T_r . This is because k_r and c_r are bounded by numbers inversely proportional to T_r . This confirms the general intuition, already pointed out in [25, 26, 55], that end users in a network should change their rate at the time scale at least on the order of their round trip time.

It is interesting to compare the delay stability condition for the wireless scenario, in particular Eqn. (15), to the one for the wireline case: $k_r \left(\sum_{j \in r} p_j + \sum_{j \in r} p'_j \sum_{s: j \in s} x_s^o \right) < \frac{\pi}{2T_r}$, $\forall r \in R$. As seen, the conditions for user r in the wireline and the wireless case only differ by a route-dependent constant, which is larger than one when $p_j(\cdot)$ and $q_j(\cdot)$ take the forms of, respectively, (3) and (6). Hence the upper bound on k_r in the wireless case, given by (15), is smaller than that in the wireline case. Intuitively this can be explained as follows: since in the wireless case both $w_r(t)$ and $x_r(t)$ are adjusted based on the feedback prices, the system is more sensitive to old prices caused by delays in the system; hence the system should get adjusted more cautiously, eventually resulting in a smaller upper bound on the step size k_r . As a side comment, the reader should notice that, in a worst case scenario, the second bound may not be too tight.

For the sake of completeness we mention here that recent results extended the results on delay stability to the global case, either by exploiting a Lyapunov-Razumikhin approach [61], or by invariance-based global stability arguments for nonlinear ordinary differential equations [46]. These interesting results cannot however be applied to the current setup because, unlike [55], they derive assumptions that are based on the specific structures of the utility functions and the prices associated with the models, and which unfortunately do not encompass those used in this work.

5. Discussion on the Practical Implementation of the Dynamic Update Scheme

From the structure of Eqn. (8) we observe that the implementation of the control law on w_r depends on the *precise* measurement of the ratio $\frac{\sum_{j \in r} p_j (\sum_{s: j \in s} x_s(t))}{\sum_{j \in r} q_j (\sum_{s: j \in s} x_s(t))}$, which is the portion of the end-to-end packet loss rate for user r that is exclusively caused by congestion. From an end-to-end point of view, a user can infer that a packet is lost and determine the missing one by merely observing a discontinuity in the sequence number that is carried by every packet.[‡] Hence, the total end-to-end packet loss rate, i.e. $\sum_{j \in r} q_j (\sum_{s: j \in s} x_s(t))$, can be precisely measured by the user. However, the actual cause of the loss (congestion or channel error) would not be directly assessable. Therefore, in practice the user needs to estimate the quantity caused exclusively by congestion, $\sum_{j \in r} p_j (\sum_{s: j \in s} x_s(t))$, in order to implement the control law for w_r . It is in general not advisable to require a direct modification of the existing TCP for this estimate. Our approach robustly achieves this estimation with no need of direct modification of the control protocol.

The literature has witnessed a considerable amount of work on the issue of practical

[‡]In practice, the sender waits for three duplicate acknowledgements while asking for the retransmission of the missing packet before it asserts that the packet is lost.

estimation of the packet loss rate caused by congestion [6–8, 10, 31, 35, 36, 44, 48, 51, 52, 54, 59]. Essentially, the idea is to estimate whether the network route is congested whenever a packet loss happens, by looking at the information delay around the instant when a packet is lost. For example, [8, 10, 48] argue that a packet loss is caused by congestion if the two-way delays are increasing at the time of the loss, while it is caused by wireless error otherwise. More precisely, [48] argues how, on detecting a packet loss, if the measured RTT is less than a calculated threshold, then the packet loss can be considered to be due to link noise, otherwise the loss is attributed to congestion. This way, users can have the knowledge of the terms $\sum_{j \in r} p_j(\sum_{s: j \in s} x_s(t))$.

One way to implement the scheme proposed in this work is the following. Users open multiple TCP connections to transmit data, with TCP controlling the sending rates of each individual connection. Moreover, users open an additional user datagram protocol (UDP) connection to send small probing packets along the same network path, in order to estimate $\sum_{j \in r} p_j(\sum_{s: j \in s} x_s(t))$ and $\sum_{j \in r} q_j(\sum_{s: j \in s} x_s(t))$ according to the techniques in [8, 48]. Since the probing connection packets can be assumed to travel along the same path as the data connection, the two packet loss rates measured by the probing connection are likely to be the same. It is also reasonable to assume that their respective loss rates are comparable. The use of dedicated UDP connections on the application layer is motivated because: 1. the functionality of estimating $\sum_{j \in r} p_j(\sum_{s: j \in s} x_s(t))$ is not available in TCP; and 2. it is not desirable to modify TCP to add such a functionality. Upon estimating the ratio, user r can then control w_r according to equation (8) by changing the number of connections. The number of probing packets sent per second could be selected in the same way as the number of data packets, which can be directly measured by sender. The packet size of the probing packets should be selected to be small enough so that the probing overhead is only a small portion of the sending rate. Notice that the robustness results that we have shown in this work suggest that the imprecision that will come with this estimation procedure will not be detrimental to the performance of the overall scheme.

In Section 1 we mentioned other existing schemes in the literature that attempt to enhance the performance of flow control in wireless networks by applying end-to-end statistics to differentiate packet loss due to congestion from that of channel error *within* TCP [6–8, 10, 31, 35, 36, 44, 48, 51, 52, 54, 59]. The difference between our suggested techniques in this paper and those schemes is that they implement those methodologies inside the TCP protocol, and hence require all end user operating systems to implement a new TCP protocol. On the contrary, our method performs the differentiation between channel error and congestion at the application layer and hence requires no modifications to the transport layer protocols such as TCP, nor the networking infrastructure.

It might appear to be a concern to approximate the dynamics of w_r , which takes continuous values, by finitely adjusting the number of connections, which is by itself an integer entity. In particular, it may be argued at first that this discretization of the dynamics may disrupt its properties. However, often the user opens a number of connections that is “large enough” to justify the fluid model, similar to the case of the packet size dynamics. Furthermore, it should be noticed that we can dynamically change the packet size of each connection to implement the small dynamics of w_r . For instance, if a change of w_r corresponds to opening one connection with packet size S , then it is also corresponds to opening two connections with packet size $S/2$, and so on—these different implementations would result in the same change in aggregate sending rate, which is at the end the net objective of the dynamic adjustment of w_r .

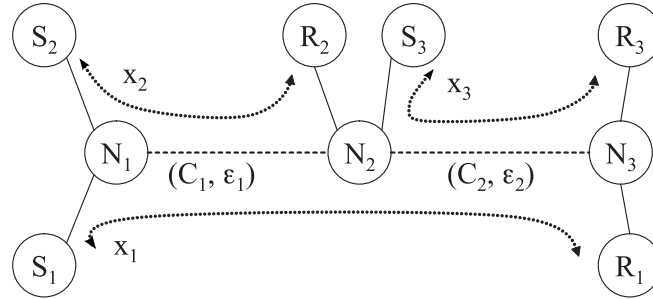


Figure 2. Network topology used for the simulations.

Since the data stream in our proposed scheme is transmitted using multiple connections, the receiver could potentially receive packets that are out of order. However, reordering application packets from multiple connections using a receiving buffer is a rather mature technology, which is widely used in peer-to-peer applications, such as BitTorrent file sharing (e.g. Kazza [27]), and peer-to-peer streaming (e.g. PPLive [45]). Hence, it is fair to delegate to these technology the solution of this issue at the application layer.

6. Simulations

In this subsection, we perform simulations to qualitatively verify the results of our analysis. We have implemented on ns-2 the schemes formally discussed in this work. The topology of the network under study is known and used in the literature and shown in Fig. 2. It is made up of three users in a nine-node network with two wireless links. A wireless link is represented by a dashed line and is characterized by two parameters: a finite capacity C and a packet loss rate ϵ that is caused by channel error. In this study, for the sake of simplicity we keep the channel error fixed. The values of the parameters for the two links are $C_1 = 1.2$ Mbps, $\epsilon_1 = 0.01$, and $C_2 = 2.0$ Mbps, $\epsilon_2 = 0.02$.

Each couple of end-users r apply Eq. (9) to control their source rate x_r , and Eq. (8) to control the source parameter w_r . Since ns-2 is a packet-level simulator, the unit of x_r is packets/sec, which is in the text reported as Mbps. The packet size is taken to be equal to 500B. To meet the two time scales assumption made in Theorem 1, we set the update frequency of w_r to be 20 times slower than the one of x_r . This can be obtained by setting c_r 20 times smaller than k_r . The parameters k_r , $r = 1, 2, 3$ are all set to be equal to 20. Additionally, in order to differentiate between the convergence of the three users, w_1^0 , w_2^0 and w_3^0 are set to 3, 4, 5 respectively. The initial conditions for the rates are $x_1 = x_2 = x_3 = 0.2$ Mbps. In order to set a benchmark to compare our ns-2 simulations against, let us calculate the optimal rate allocation for the scenario: $x_1 = 0.38$ Mbps, $x_2 = 0.82$ Mbps and $x_3 = 1.62$ Mbps. Time is discretized, with an update period for both x_r and w_r of 0.1 second. After each update of w_r , we round it to an integer and feed this integer into the rate adaptation algorithm of Eq. (8). This is necessary in practice since w_r is interpreted as the number of connections of each source. Each simulation has a time horizon of 1000 seconds, which allows to observe its long-term behavior. We set the size of a FIFO buffer in each wireless node N_i , $i = 1, 2, 3$, to be

equal to 100 packets. In our simulations each packet has a sequence number, and a destination feedbacks an acknowledgment to its source whenever it successfully receives a packet. Thus a source calculates the packet loss rate based on the number of un-received packets. Reverse traffic is a potential problem in ACK-based congestion control protocols and can lead to utilization losses [42]. It is desirable to solve this issue at the router level—for instance, a router can provide priority queues, enabling the placement of delay-sensitive traffic on the outbound link before delay-insensitive traffic. If we place ACKs in the highest-priority traffic class while we place data packets in the lowest-priority traffic class, ACKs are always serviced before data packets and thus we avoid ACK to be congested by reverse traffic. [34] discusses models of ow-size based priorities in Internet routers. If a router does not support priority queues, end-to-end solutions are needed. For example in [20], whenever the sender detects an ACK loss, it notifies the receiver to decrease the sending rate of ACKs. The use of selective ACKs and delayed ACKs can also mitigate the effect of ACK losses caused by reverse traffic. A second option to mitigate the reverse traffic problem is the following: since packets are sent in sequence, a receiver can count how many packets circulate and notify the real packet loss rate to its sender. In the present case study reverse traffic does not appear to be an issue. Arriving packets are dropped when the buffer overflows, which thus denotes congestion loss. Besides, packets sometimes cannot be transmitted successfully on a wireless link due to channel error, which is related to channel loss. We will consider two approaches to differentiate congestion losses from channel losses.

The ns-2 simulation is first run with *perfect loss differentiation*, which can be done as follows: we mark (instead of dropping) a packet when it causes the buffer to overflow. The destination feedbacks the congestion information using an acknowledgment when it receives a marked packet. Thus, the sources can completely differentiate congestion losses from channel losses. This mechanism is equivalent to explicit congestion notification (ECN) [32,33]. Fig. 3 plots the dynamics of the process $x_r(t)$, $w_r(t)$, as well as channel loss rate and congestion loss rate as estimated by sources. We draw a number of observations. Plot (a) shows that, in steady-state, on average $x_1 = 0.39\text{Mbps}$, $x_2 = 0.80\text{Mbps}$ and $x_3 = 1.60\text{Mbps}$. These results are nearly the same as the benchmark figures. Notice that the capacity of each link is saturated. Plot (b) shows that, in steady-state, $w_1 = 5$ or 6 , $w_2 = 6$, and $w_3 = 13$ or 14 (the persistent oscillatory behavior of w_3 is due to its integer value, and results in some oscillation also on the source rate x_3). Plot (c) shows channel loss rate as estimated by each source. Notice that, in steady-state, $\epsilon_1 + \epsilon_2 = 0.029$ (source 1), whereas $\epsilon_1 = 0.010$ (source 2) and $\epsilon_2 = 0.020$ (source 3). Thus the channel loss estimations are nearly the same as our predefined settings. Finally, plot (d) shows the congestion loss rate estimated by each source: $p_1 + p_2 = 0.033$ (source 1), $p_1 = 0.019$ (source 2) and $p_2 = 0.012$ (source 3).

As the second batch of ns-2 simulations, based on *imperfect loss differentiation*, we implemented the scheme discussed in [8], which allows differentiating channel losses from congestion losses based on end-host estimations. Fig. 4 represents the same entities as Fig. 3. Let us draw a number of observations. Plot (a) shows that, in steady-state, on average $x_1 = 0.52\text{Mbps}$, $x_2 = 0.70\text{Mbps}$, and $x_3 = 1.51\text{Mbps}$. In spite of imperfect loss differentiation, the result is acceptable when compared with the benchmark. Notice that the capacity of each link is saturated. Plot (b) shows that, in steady-state, $w_1 = 8$ or 9 , $w_2 = 5$ or 6 , and $w_3 = 14$ or 15 . Plot (c) shows the channel loss rate as estimated by each source. Notice that, in steady-state, $\epsilon_1 + \epsilon_2 = 0.042$ (source 1), $\epsilon_1 = 0.010$ (source 2) and $\epsilon_2 = 0.025$ (source 3). Again the channel loss estimation is not far from our predefined settings. Plot (d) displays the congestion loss

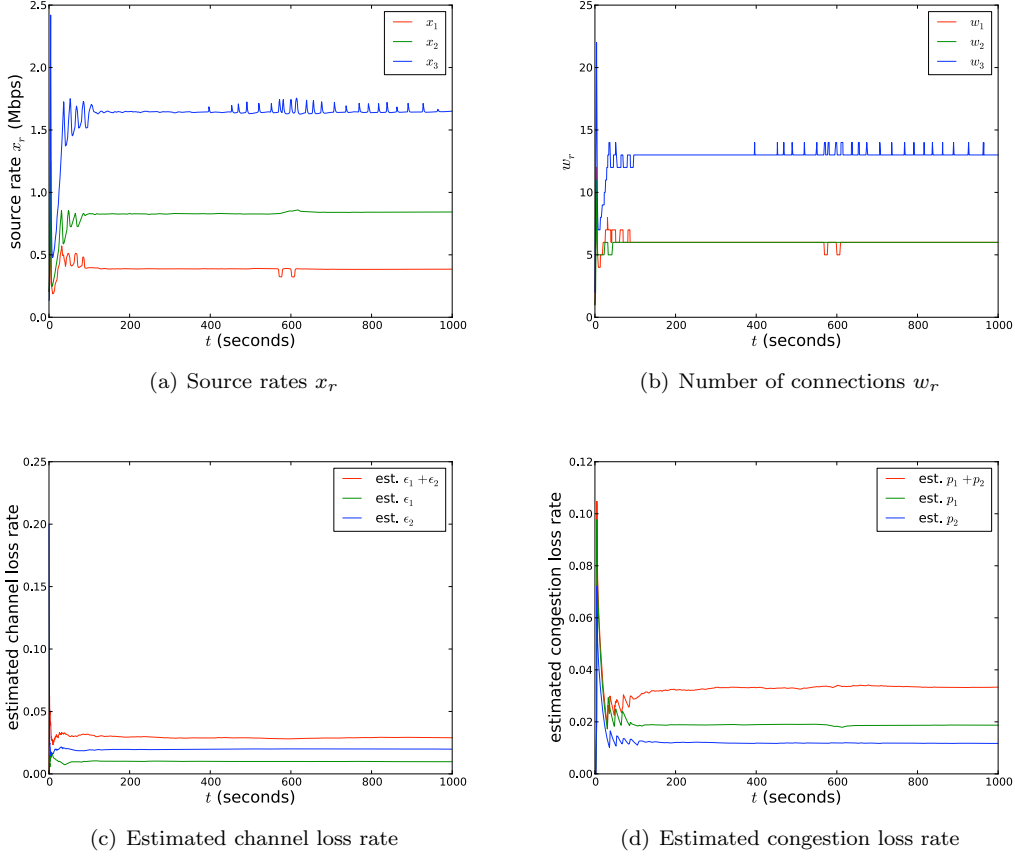


Figure 3. ns-2 outcomes, based on perfect loss differentiation.

rate estimated by each source: in the long run $p_1 + p_2 = 0.023$ (source 1), $p_1 = 0.023$ (source 2) and $p_2 = 0.013$ (source 3). Note that source 1 underestimates the congestion loss rate because it overestimates the channel loss rate. Thus, x_1 (0.52Mbps) is larger than its optimal rate (0.38Mbps).

TCP NewReno improves retransmission during the fast recovery phase of TCP Reno [22], however it is not designed for the wireless scenario with channel loss since it specifically assumes that packet loss is a sign of congestion. Fig. 5 compares two ns-2 simulations run under the same conditions of the previous experiments: the first simulation (left plot) implements the NewReno scheme under no channel losses, whereas the second (right plot) is run over channels characterized by the loss rates used in the previous simulations. One can observe that in steady-state the first output yields a performance for the source rates that can be put in relationship with that of Figure 3, as expected. The outcomes for the lossy channels instead display a performance degradation, which is particularly evident over the two-hop source x_1 . On average, the source rates are $x_1 = 0.30$ Mbps, $x_2 = 0.80$ Mbps, and $x_3 = 1.50$ Mbps, which

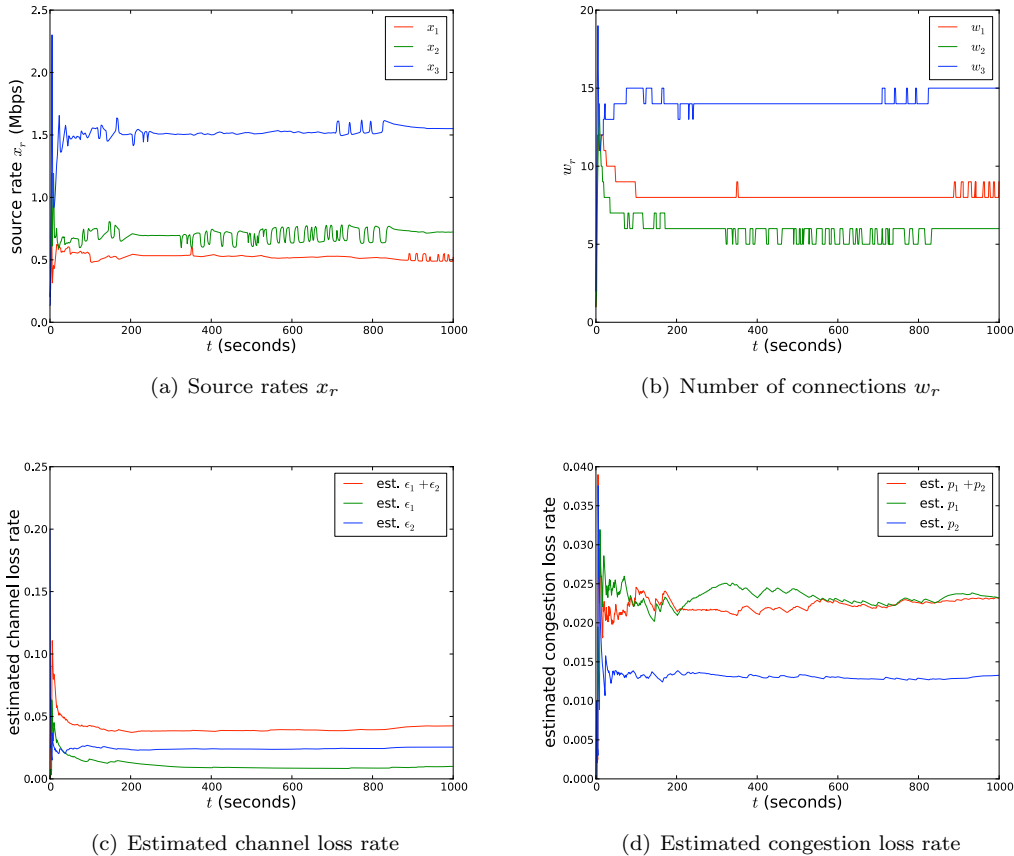


Figure 4. ns-2 outcomes, based on imperfect loss differentiation.

leads to the observation that both link capacities are underutilized.

7. Conclusions

Motivated by the increasing importance of wireless networks and by their applications, this work has studied a modified model of the classical TCP protocol and analyzed some of its dynamical properties. After evaluating the possible limitations of the introduced mathematical model, we have argued about its practical applicability, and outlined a comparison with other schemes proposed in the literature. In adjacent research, we have performed further simulations and actual experiments of an actual protocol derived from the one presented here. Further extensions and improvements of the dynamical scheme, both from a design and analysis standpoint [53], are currently under study. For instance, it would be interesting to extend the global delay stability results in [46, 61] to the current framework. Furthermore, exploiting

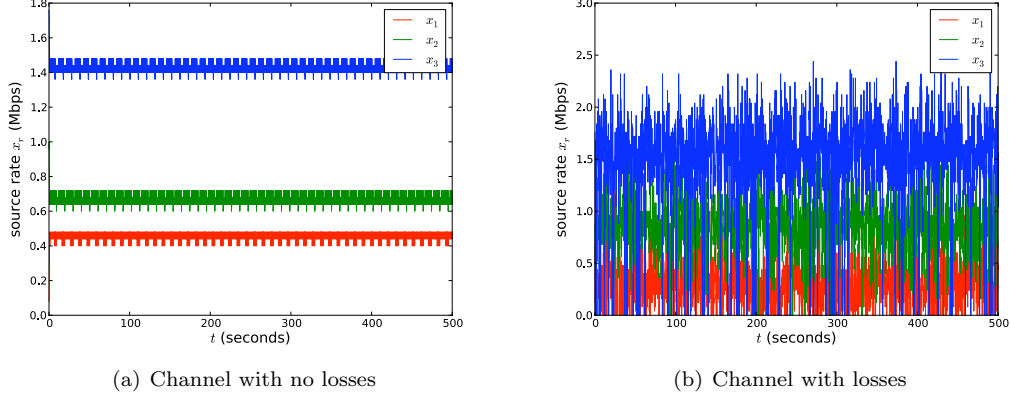


Figure 5. ns-2 outcomes based on NewReno protocol.

non-standard techniques from Systems Theory may suggest ways to extend the current results to the case of time-varying link capacities and round trip times.

Additional Material

The ns-2 software used for the simulation can be found at the following URL:

<http://www.dsc.tudelft.nl/~aabate/ijrnc>

APPENDIX

In this Section we adhere to the notations we used throughout the paper. In particular, let us recall that the quantities $p_j, p'_j, q_j, q'_j, \epsilon_j, \epsilon'_j, j \in J$ are the values of the respective functions evaluated at the equilibrium points.

I.1. Proof of Theorem 1

The first lemma elucidates the notion of semi-global exponential stability and its relationship with local exponential stability and global asymptotical stability.

Lemma 1. *Consider a system $\dot{\xi} = \varphi(\xi, t)$ satisfying the following assumptions:*

- $\varphi(\xi, t)$ is continuous w.r.t. time and Lipschitz continuous w.r.t. variable ξ ;
- it has a unique equilibrium at $\xi = 0$ that is locally exponentially stable and globally asymptotically stable.

Then the equilibrium of the system is semi-globally exponentially stable [49, Chapter 5].

Proof: Denoting with $\xi(t, \xi_0)$ the unique solution of the vector field φ starting at an arbitrary ξ_0 at time $t = 0$, by definition of local exponential stability there exist positive constants $r < 1, K > 1$, and γ s.t. $\|\xi(t, \xi_0)\| \leq K\|\xi_0\|e^{-\gamma t}, \forall \xi_0 : \|\xi_0\| \leq r, \forall t \geq 0$. Define $T_r(\xi_0) = \inf\{t \geq 0 : \|\xi(t, \xi_0)\| \leq r\}$. By the assumption on the global asymptotical stability of the system, $T_r(\xi_0) < \infty$. Since the solution $\xi(t, \xi_0)$ is continuous with respect to t , let $M_r(\xi_0) = \max_{0 \leq t \leq T_r(\xi_0)} \{K, \|\xi(t, \xi_0)\|\} < \infty$. Define $L = M_r(\xi_0)e^{\gamma T_r(\xi_0)}/r$. We claim that

$$\|\xi(t, \xi_0)\| \leq L\|\xi_0\|e^{-\gamma t}. \quad (17)$$

If $\|\xi_0\| \leq r$, then the claim holds trivially. If instead $\|\xi_0\| > r$, the following holds in the interval $t \in [0, T_r(\xi_0)]$: $L\|\xi_0\|e^{-\gamma t} = M_r(\xi_0)\frac{\|\xi_0\|}{r}e^{\gamma T_r(\xi_0)}e^{-\gamma t} \geq M_r(\xi_0) \geq \|\xi(t, \xi_0)\|$. For $t \geq T_r(\xi_0)$, let $t' = t - T_r$; we then have $\|\xi(t', \xi_0)\| \leq r$, which leads to the first case. Equation (17) implies semi-global exponential stability of the equilibrium [49, Definition 5.10]. (The semi-global part comes from the dependency of L on the initial condition ξ_0 .) ■

In the following we state two lemmas that are needed for proving the stability of the complete model. The first lemma explores an interesting structure of the vector fields of both the boundary layer system and the reduced order system.

Lemma 2. *Consider the variables $w_r(t), r \in R$, in the reduced order system (13). There exists a compact set, denoted by $\Omega_1 \subseteq \mathbb{R}^{\text{card}(R)}$, such that any set $\lambda_1\Omega_1, \lambda_1 > 1$, is positive invariant. Similarly, consider the variables $x_r(t)$ in the boundary layer system (12). There exists a corresponding compact set $\Omega_2(\mathbf{w}) \subseteq \mathbb{R}^{\text{card}(R)}$, defined as a function of \mathbf{w} , such that any set $\lambda_2\Omega_2(\mathbf{w}), \lambda_2 > 1$, is positive invariant.*

Proof: To construct the compact set $\Omega_1 \subseteq \mathbb{R}^{\text{card}(R)}$ for the variables $w_r, r \in R$ in the reduced order system, first note that there always exists a large enough $w_r^{\text{max}} > 0$ that makes route r be congested regardless of the cross traffic (which happens if only user r is active and the others are inactive). Furthermore, $w_r^{\text{max}} > 0$ can be chosen so that $\dot{w}_r^{\text{max}} \leq 0$. Cross traffic only contributes to congestion and results in a larger value for $\sum_{j \in r} p_j (\sum_{s: j \in s} x_s(t))$. Consequently, cross traffic increases the value of $\frac{\sum_{j \in r} p_j (\sum_{s: j \in s} x_s(t))}{\sum_{j \in r} q_j (\sum_{s: j \in s} x_s(t))}$: hence, \dot{w}_r^{max} remains negative in the presence of cross traffic. By a similar argument, any $w_r > w_r^{\text{max}}$ will result in $\dot{w}_r < 0$. The set Ω_1 can be thus defined as $\Omega_1 \doteq [0, w_1^{\text{max}}] \times [0, w_2^{\text{max}}] \times \dots \times [0, w_{\text{card}(R)}^{\text{max}}]$. It is easy to check that the flow of the vector field satisfies the following: if $w_r(t) \leq 0$, then $\dot{w}_r(t) > 0$ according to (8) or (13); if $w_r(t) \geq w_r^{\text{max}}$, then $\dot{w}_r(t) \leq 0$ by the above arguments. Therefore, on the boundary of any compact set $\lambda_1\Omega_1 \doteq [0, \lambda_1 w_1^{\text{max}}] \times [0, \lambda_1 w_2^{\text{max}}] \times \dots \times [0, \lambda_1 w_{\text{card}(R)}^{\text{max}}], \lambda_1 > 1$,[§] the vector field for w_r defined in (13) points inward. Hence by definition, the compact set Ω_1 is positive invariant.

Similarly, a compact set $\Omega_2(\mathbf{w})$ can be defined for the variables $x_r(t)$ in the boundary layer system (12) as $\Omega_2(\mathbf{w}) \doteq [0, x_1^{\text{max}}(w_1)] \times [0, x_2^{\text{max}}(w_2)] \times \dots \times [0, x_{\text{card}(R)}^{\text{max}}(w_{\text{card}(R)})]$, where $x_r^{\text{max}}(w_r) = \frac{w_r}{\sum_{j \in r} (q_j (\sum_{s: j \in r} x_s))}$ satisfies the following, given a w_r and regardless of cross traffics

[§]In practice, from a networking point of view, “containing Ω_1 ” means that the number of connections of each user should be allowed to take large enough values to fully utilize his bottleneck, even if only that user is active. Similarly, in the following, “containing $\Omega_2(\mathbf{w})$ ” means that the sending rate of each user can be sufficiently large to achieve its equilibrium. These requirements are trivial to meet in practice.

along route r : if $x_r(t) \leq 0$, then $\dot{x}_r(t) > 0$ according to (9) or (12); if $x_r(t) \geq x_r^{max}(w_r)$, then $\dot{x}_r(t) \leq 0$. ■

The upcoming lemma investigates the global asymptotical stability of the reduced order system. From a technical standpoint, we shall make use of a La Salle's argument, as the particular non-linear shape of the vector field for $\dot{w}_r(t)$ in (13) makes the problem of finding a suitable Lyapunov function apparently insurmountable – specifically, none of the classical shapes [11, 29] work in this instance. We believe that our solution may provide to be a general choice for La Salle functions in similar problems.

Lemma 3. *The unique equilibrium of reduced order system in (13) is globally asymptotically stable.*

Proof: We apply a La Salle's argument to prove the global asymptotical stability for the equilibrium of the reduced order system (13). According to Lemma 1 in [1], for the reduced order system in (13) it is possible to write $\dot{\mathbf{w}} = \text{diag}(\mathbf{x})D\dot{\mathbf{x}}$, where

$$D = \left(\text{diag} \left(\sum_{j \in r} \frac{q_j(\sum_{s: j \in s} x_s)}{x_r} \right) + A^T \text{diag} \left(q'_j \left(\sum_{s: j \in s} x_s \right) \right) A \right).$$

Notice that $D \succeq 0$. Define the quantity $E \triangleq \text{diag} \left(\frac{x_r^2}{w_r} \right) - D^{-1}$. From direct observation (see also Appendix B in [1]) we see that $E \succeq 0$. Introducing the new variable $z_r \triangleq \frac{w_r}{x_r}$, we have that $\sum_{j \in r} p_j(\sum_{s: j \in s} x_s) = z_r - \sum_{j \in r} \epsilon_j(\sum_{s: j \in s} x_s)$, $\forall r \in R$, and that the following holds:

$$\begin{aligned} \dot{\mathbf{z}} &= \text{diag} \left(\frac{1}{x_r} \right) \dot{\mathbf{w}} - \text{diag} \left(\frac{w_r}{x_r^2} \right) D^{-1} \text{diag} \left(\frac{1}{x_r} \right) \dot{\mathbf{w}} \\ &= \text{diag}(z_r) \text{diag} \left(\frac{1}{x_r} \right) E \text{diag} \left(\frac{1}{x_r} \right) \dot{\mathbf{w}}. \end{aligned}$$

Define the monotonically increasing functions $f_r(\lambda) = \lambda / (\lambda + \sum_{j \in r} \epsilon_j(\sum_{s: j \in s} x_s))$, $\forall r \in R$; then $f_r(z_r - \sum_{j \in r} \epsilon_j(\sum_{s: j \in s} x_s(t))) = \frac{\sum_{j \in r} p_j(\sum_{s: j \in s} x_s(t))}{\sum_{j \in r} q_j(\sum_{s: j \in s} x_s(t))}$. Given a vector $\mathbf{w}(0)$ of initial conditions, let Σ be the minimal connected compact set containing $\mathbf{w}(0)$ and the invariant set Ω_1 defined in Lemma 2. Introduce the monotonically increasing functions $\phi_r(\lambda)$, $\forall r \in R$ as follows:

$$\phi_r(\lambda) = \int_{(\sum_{j \in r} \epsilon_j(\cdot))}^{(f_r^{-1}(\lambda) + \sum_{j \in r} \epsilon_j(\cdot))} \frac{f_r(y - \sum_{j \in r} \epsilon_j(\cdot))}{y} dy.$$

Define a continuous function over Σ as follows:

$$V(\mathbf{z}, \mathbf{w}) = - \sum_{r \in R} c_r \left[w_r^0 \ln(z_r \sum_{j \in r} \epsilon_j(\cdot)) + \int_{\sum_{j \in r} \epsilon_j(\cdot)}^{z_r} \frac{w_r}{y} f_r \left(y - \sum_{j \in r} \epsilon_j(\cdot) \right) dy + \int_{\frac{w_r}{w_r^0}}^{\infty} \phi_r \left(\frac{1}{\lambda} \right) d\lambda \right].$$

The time derivative of the above quantity is

$$\begin{aligned} \dot{V}(\mathbf{z}, \mathbf{w}) = & - \sum_{r \in R} c_r \left(\frac{w_r^o}{z_r} - \frac{w_r}{z_r} f_r \left(z_r - \sum_{j \in r} \epsilon_j(\cdot) \right) \right) \dot{z}_r \\ & - \sum_{r \in R} c_r \left(\phi_r \left(\frac{w_r^o}{w_r} \right) - \phi_r \left(f_r \left(z_r - \sum_{j \in r} \epsilon_j(\cdot) \right) \right) \right) \dot{w}_r \\ & - \sum_{r \in R} c_r \left(\frac{w_r^o}{\sum_{j \in r} \epsilon_j(\cdot)} + w_r \left(\frac{1}{z_r} - \frac{1}{\sum_{j \in r} \epsilon_j(\cdot)} \right) \right) \frac{\partial(\sum_{j \in r} \epsilon_j(\cdot))}{\partial x_r} \dot{x}_r. \end{aligned} \quad (18)$$

The first term in the above expression can be written as $\dot{\mathbf{w}}^T \text{diag} \left(\frac{1}{x_r} \right) E \text{diag} \left(\frac{1}{x_r} \right) \dot{\mathbf{w}}$. It can be noted that $\dot{V}(\mathbf{z}, \mathbf{w}) \leq 0$. Exploiting the monotonicity assumption for the functions $\phi_r(\cdot)$, a few steps of algebra show that also the second term is a (negative) quadratic function of $\dot{\mathbf{w}}$. Finally, the third term, which comes from the time derivative of the error functions for each link, can be also related to the dynamics of w_r by noticing that it can be expressed as $-\sum_{r \in R} \frac{\dot{w}_r}{\sum_{j \in r} \epsilon_j(\cdot)} \frac{\partial(\sum_{j \in r} \epsilon_j(\cdot))}{\partial x_r} \dot{x}_r$. By using the relationship between the time derivatives of \mathbf{x} and \mathbf{w} , this term can be reordered and compactly written as

$$-\dot{\mathbf{w}}^T \text{diag} \left(\frac{1}{\sum_{j \in r} \epsilon_j(\cdot)} \right) A^T \text{diag}(\epsilon'_j(\cdot)) A D^{-1} \text{diag}(x_r^{-1}) \dot{\mathbf{w}}.$$

Let us denote this quantity $-\dot{\mathbf{w}}^T \Delta \dot{\mathbf{w}}$. This expression is again a negative quadratic function of $\dot{\mathbf{w}}$: the inverse Δ^{-1} of the kernel matrix can in fact be rewritten as

$$\text{diag}(x_r) \left(\text{diag} \left(\frac{\sum_{j \in r} q_j(\cdot)}{x_r} \right) A^{-1} \text{diag} \left(\frac{1}{\epsilon'_j(\cdot)} \right) (A^T)^{-1} + A^T \text{diag} \left(\frac{q'_j(\cdot)}{\epsilon'_j(\cdot)} \right) A \right) \text{diag} \left(\sum_{j \in r} \epsilon_j(\cdot) \right),$$

and be shown to be positive definite by looking at its symmetric part, $\frac{1}{2} (\Delta^{-1} + (\Delta^{-1})^T)$, and deduce its positive definiteness. Hence, relation (18) is negative, and equality is achieved if and only if $\dot{\mathbf{w}} = 0$, i.e. at the equilibrium point.

From the above consideration it follows that we can apply the La Salle principle (see for instance [49]) to conclude that the trajectory starting from $\mathbf{w}(0)$ converges asymptotically to the equilibrium. As $\mathbf{w}(0)$ is arbitrary, we conclude that the equilibrium is globally asymptotically stable. Notice that the theorem holds in particular if the error quantities $\sum_{j \in r} \epsilon_j(\cdot)$ are constant functions of their variables. \blacksquare

Proof of Theorem 1 We have shown that the reduced order system in (13) has a unique equilibrium that is globally asymptotically stable (Lemma 3). Furthermore, its local exponential stability can be shown by one of the following two arguments: by exploiting the frequency analysis of Theorem (2) and choosing $T_r = 0, \forall r \in R$; equivalently, by linearizing Eqn. (8) around its equilibrium and applying similar arguments to those shown in Lemma 2, [1]. Invoking Lemma 1 we conclude that the unique equilibrium of the reduced order system is semi-globally exponentially stable.

A similar argument is also true for the boundary layer system in (12). By leveraging the above approach and slight variation of theorems in [29], it has a unique equilibrium that is globally

asymptotically stable and locally exponentially stable. Hence, again by Lemma 1, we conclude that the unique equilibrium of the boundary layer system is semi-globally exponentially stable.

It is necessary to define a compact set for the composition. As in Lemma 2, define a set Ω_1 , where $\bar{w} = \max\{w_r : w_r \in R\}$ and $w_r^{max} = \bar{w}, \forall r \in R$; then Ω_1 and any positive rescaling $\Sigma_1 \doteq \lambda_1 \Omega_1$ are positive invariant, by the arguments developed for Lemma 2. We shall then constrain $\mathbf{w}(t)$ to the compact set Σ_1 . Similarly, let us constrain $\mathbf{x}(t)$ into a compact set, denoted by $\Sigma_2 \doteq \lambda_2 \Omega_2(\mathbf{w}^{max})$, as defined for Lemma 2. On $\Sigma_1 \times \Sigma_2$, the equilibrium of the reduced order system in (13) is exponentially stable, and that of the boundary layer system in (12) is exponentially stable, uniformly in w ; the verification of this claim follows similarly to the statement in Appendix II.F in [33]. Together with the fact that $\Sigma_1 \times \Sigma_2$ is a positive invariant set, by the application of Theorem 11.4 in [30], we conclude that the singularly perturbed system in (10)-(11) has a unique equilibrium in $\Sigma_1 \times \Sigma_2$ that is exponentially stable. Hence, the equilibrium is semi-globally exponentially stable. ■

I.2. Proof of Theorem 2

Let us start with a technical lemma:

Lemma 4. Consider the matrices $P = P^* \succ 0$, $Q = Q^* \succ 0$, $L = \text{diag}(l_i)$, $l_i \in \mathbb{C}$, $\forall i$. P, Q, L are assumed to have the same size. Then

$$\sigma(Q^{-1}LP) \subset \rho(Q^{-1}P)(Co(0 \cup \{l_i\}) \cup Co(0 \cup \{-l_i\})),$$

where σ is the spectrum of a matrix, ρ its spectral radius, while the symbol Co denotes a function giving the convex hull of its arguments.

Proof: Let v be a normalized right eigenvector of $Q^{-1}LP$, corresponding to the eigenvalue λ : $Q^{-1}LPv = \lambda v$. Then $LPv = \lambda Qv \Rightarrow (Pv)^*L(Pv) = \lambda(Pv)^*Qv = \lambda v^*P^*(QP^{-1})Pv$. Therefore, naming $k = \rho(Q^{-1}P)((Pv)^*QP^{-1}(Pv))$ and observing that $|k| \geq 1$, $\lambda = \frac{(Pv)^*L(Pv)}{v^*P^*(QP^{-1})Pv} = \rho(Q^{-1}P)(\sum_i \frac{|Pv_i|^2}{k}(\pm l_i) + 0)$. The conclusion follows. ■

Proof of Theorem 2 As in the proof of global stability for the dynamic update case, we shall exploit the idea of the two time scales. To begin with, the condition in inequality (15) refers to the boundary layer system (12) and is obtained as follows. By setting $x_r(t) = x_r^o + y_r(t)$, $r \in R$, and linearizing the system equations around the equilibrium $x_r^o = w_r^o / \sum_{j \in r} q_j$:

$$\frac{d}{dt}y_r(t) = -k_r y_r(t - T_r) \sum_{j \in r} q_j - k_r x_r^o \sum_{j \in r} \sum_{s: j \in s} y_s(t - d_1(j, s) - d_2(j, r)) q'_j.$$

Taking the Laplace transform and simplifying by introducing matrix terms, we have

$$sY(s) = -\text{diag}(e^{-sT_r}) \text{diag}(x_r^o) \left(\text{diag}((x_r^o)^{-1}) \text{diag}\left(\sum_{j \in r} q_j\right) + M(s) \right) Y(s), \quad (19)$$

where $Y(s) = (Y_1(s), Y_2(s), \dots, Y_{\text{card}(R)}(s))^T$ is a $\text{card}(R) \times 1$ vector made up of the Laplace transforms $Y_i(s)$ of the elements $y_i(t)$, and $M(s) = (m_{rq}(s))_{\text{card}(R) \times \text{card}(R)}$ is a matrix with elements $m_{rq} = \sum_{j \in q \cap r} q'_j \cdot e^{-s(d_1(j, q) - d_1(j, r))}$. Comparing Eqn. (19) to (6) in [55], we can see

they only differ by the feedback functions from the links. Thus the local stability investigated here is exactly the same as the one in [55]. The linearization of the relations for the reduced order system (13) around its equilibrium, comprehensive of the delays, are, $\forall r \in R$:

$$\begin{aligned}\dot{\omega}_r(t) &= c_r \omega_r^o - c_r x_r(t - T_r) \sum_{j \in r} p_j \left(\sum_{s: j \in s} x_s(t - d_1(j, s) - d_2(j, r)) \right); \\ \omega_r(t - T_r) &= x_r(t - T_r) \sum_{j \in r} q_j \left(\sum_{s: j \in s} x_s(t - d_1(j, s) - d_2(j, r)) \right).\end{aligned}$$

Taking the derivative of the second term, after a proper shift in time, we obtain:

$$\dot{\omega}_r(t) = \dot{x}_r(t) \sum_{j \in r} q_j \left(\sum_{s: j \in s} \tilde{x}_s(j, r) \right) + x_r(t) \sum_{j \in r} q'_j \left(\sum_{s: j \in s} \tilde{x}_s(j, r) \right) \sum_{s: j \in s} \dot{\tilde{x}}_s(j, r),$$

where $\tilde{x}_s(j, r) = x_s(t - d_1(j, s) - d_2(j, r) + T_r)$. Substituting this last relation within the first one, letting $x_r(t) = x_r^o + y_r(t)$, $\dot{x}_r(t) = 0 + \dot{y}_r(t)$, and linearizing around these equilibrium points, yields:

$$\begin{aligned}\sum_{j \in r} q_j \dot{y}_r(t) &= -c_r x_r^o \sum_{j \in r} p'_j \sum_{s: j \in s} y_s(t - d_1(j, s) - d_2(j, r)) - c_r y_r(t - T_r) \sum_{j \in r} p_j \\ &\quad - x_r^o \sum_{j \in r} q'_j \sum_{s: j \in s} \dot{y}_s(t - d_1(j, s) - d_2(j, r) + T_r).\end{aligned}\quad (20)$$

Taking the Laplace transform and simplifying out common matrix terms, we obtain:

$$\begin{aligned}\left[\text{diag} \left(\frac{1}{x_r^o} \right) \text{diag} \left(\sum_{j \in r} q_j \right) + N(s) \right] sY(s) &= \\ - \text{diag}(c_r) \text{diag}(e^{-sT_r}) \left[\text{diag} \left(\frac{1}{x_r^o} \right) \text{diag} \left(\sum_{j \in r} p_j \right) + M(s) \right] Y(s),\end{aligned}\quad (21)$$

where $N(s)$ and $M(s)$ are made up of the following terms:

$$\begin{aligned}N_{rq}(s) &= \sum_{j \in r \cap q} q'_j \exp(-s(d_1(j, q) - d_2(j, r))) \\ M_{rq}(s) &= \sum_{j \in r \cap q} p'_j \exp(-s(d_1(j, q) - d_2(j, r))).\end{aligned}$$

Introducing as intuitive the quantities $\bar{Q}(s)$ and $\bar{P}(s)$, Eqn. (21) can be rewritten as

$$sY(s) = -\bar{Q}(s)^{-1} \text{diag}(c_r) \text{diag}(e^{-sT_r}) \bar{P}(s) Y(s).$$

We are interested in checking for the stability of this interconnection, and in case pose conditions on the c_r terms to enforce it. Name $L = \text{diag} \left(c_r \frac{\pi}{2} \frac{\exp(-j\omega T_r)}{j\omega T_r} \right)$, $P = \text{diag}(\sqrt{x_r}) \bar{P}(j\omega) \text{diag}(\sqrt{x_r})$, $Q = \text{diag}(\sqrt{x_r}) \bar{Q}(j\omega) \text{diag}(\sqrt{x_r})$. Observing that the matrices $\bar{Q}(s)^{-1} \text{diag}(c_r) \text{diag}(e^{-sT_r}) \bar{P}(s)$ and $Q^{-1}LP$ are *similar* [23], we then obtain

$$\sigma(Q^{-1}LP) \subset \rho(Q^{-1}P) \left(Co \left(0 \cup \left(c_r \frac{\pi}{2} \frac{\exp(-j\omega T_r)}{j\omega T_r} \right) \right) \cup Co \left(0 \cup \left(-c_r \frac{\pi}{2} \frac{\exp(-j\omega T_r)}{j\omega T_r} \right) \right) \right).$$

The necessary introduction of the additional negative terms in the convex hull does not change its structural property of exclusion of the point -1 in the complex plane. The problem then boils down to posing conditions on the term $\rho(Q^{-1}P)$; we know that $\rho(Q^{-1}P) \leq \rho(Q^{-1})\rho(P) \leq \frac{\rho(P)}{\min_{\lambda} \sigma(Q)}$. Notice that $N(s) = A^T(-s)\text{diag}\left(\sum_{j \in s} q'_j\right)A(s)$ and that $N(s) = N(-s)$, $N(j\omega) \succ 0, \forall \omega$. From a linear algebra fact [23], given two matrices $A = A^* \succeq 0$ and $B = B^*$, then the eigenvalues of their sum, ranked increasingly, are correspondingly lower bounded by those of B . Therefore, focusing on the structure of matrix $Q = \text{diag}\left(\frac{1}{x_r^2}\right)\text{diag}\left(\sum_{j \in r} q_j\right) + N(s)$, we claim that $\min_r \{\sum_{j \in r} q_j\} \leq \min_{\lambda} \sigma(Q)$. This translates into the condition that, $\forall r \in R$:

$$\rho(Q^{-1}P) < c_r \frac{\left(\sum_{j \in r} p_j + \sum_{j \in r} p'_j \sum_{s:j \in s} x_s^o\right)}{\min_{r \in R} \sum_{j \in r} q_j} < \frac{\pi}{2T_r}.$$

■

REFERENCES

1. A. Abate, M. Chen, and S. Sastry. Analysis of an implementable application layer scheme for flow control over wireless networks. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, Kyoto, JP, 2006.
2. T. Alpcan and T. Basar. A game-theoretic framework for congestion control in general topology networks. In *Proc. IEEE Conference on Decision and Control*, pages 1218–1224, Las Vegas, NV, December 2002.
3. T. Alpcan and T. Basar. Global stability analysis of end-to-end congestion control schemes for general topology networks with delay. In *Proc. IEEE Conference on Decision and Control*, pages 1092–1097, Maui, Hawaii, December 2003.
4. H. Balakrishnan and R. Katz. Explicit loss notification and wireless web performance. In *Proc. of IEEE Globecom Internet Mini-Conference*, November 1998.
5. H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz. A comparison of mechanisms for improving TCP performance over wireless links. *IEEE/ACM Trans. Networking*, 5(6):756–769, 1997.
6. D. Barman and I Matta. Effectiveness of loss labeling in improving TCP performance in wired/wireless networks. In *Proc. of the ICNP*, pages 2–11, Washington, DC, USA, 2002.
7. S. Biaz and N. H. Vaidya. Distinguishing congestion losses from wireless transmission losses: a negative result. In *Proc. of the Seventh International Conference on Computer Communications and Networks (IC3N)*, pages 722 – 731, New Orleans, USA, October 1998.
8. S. Biaz and N. H. Vaidya. Discriminating congestion loss from wireless losses using inter-arrival times at the receiver. In *Proc. of IEEE Symposium on Application-specific System and Software Engr. and Techn.*, pages 10–17, Richardson, TX, USA, March 1999.
9. L.S. Brakmo and L.L. Peterson. TCP Vegas: end-to-end congestion avoidance on a global internet. *IEEE J. Select. Areas Commun.*, 13(8):1465–1480, October 1995.
10. S. Cen, P. Cosman, and G. Voelker. End-to-end differentiation of congestion and wireless losses. *IEEE/ACM Trans. Networking*, 11(5):703–717, 2003.
11. M. Chen, A. Abate, and S. Sastry. New congestion control schemes over wireless networks: Stability analysis. In *Proceedings of the 16th IFAC World Congress*, Prague, CZ, 2005.
12. M. Chen and A. Zakhor. Rate control for streaming video over wireless. In *Proc. IEEE INFOCOM*, Hong Kong, China, March 2004.
13. M. Chen and A. Zakhor. AIO-TFRC: A light-weighted rate control scheme for streaming over wireless. In *Proc. of IEEE WirelessCom Symposium on Multimedia over Wireless 2005*, June 2005.
14. M. Chen and A. Zakhor. Flow control over wireless network and application layer implementation. In *Proc. IEEE INFOCOM*, pages 1–12, Barcelona, April 2006.
15. M. Chen and A. Zakhor. Multiple TFRC connections based rate control for wireless networks. *IEEE Trans. Multimedia*, 8(5):1045–1062, Oct. 2006.
16. C.F. Chiasserini and M. Meo. Improving TCP over wireless through adaptive link layer setting. In *Proc. of IEEE Global Telecommunications Conference*, pages 1766–1770, Piscataway, NJ, USA, 2001.

17. J.-H. Choi, S.-H. Yoo, and C. Yoo. A flow control scheme based on buffer state for wireless TCP. In *Proc. of the 4th International Workshop on Mobile and Wireless Communications Network*, pages 592–596, Piscataway, NJ, USA, 2002.
18. W. Ding and A. Jamalipour. A new explicit loss notification with acknowledgment for wireless TCP. In *Proc. of 12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pages B–65–9, Piscataway, NJ, USA, 2001.
19. D.A. Eckhardt and P. Steenkiste. Improving wireless LAN performance via adaptive local error control. In *ICPN*, pages 327–338, 1998.
20. S. Floyd, A. Arcia, D. Ros, and J. Iyengar. Adding acknowledgement congestion control to TCP - rfc 5690, February 2010.
21. S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-based congestion control for unicast applications. In *Proc. ACM SIGCOMM*, pages 43–56, Stockholm, Sweden, August 2000.
22. S. Floyd, T. Henderson, and A. Gurtov. The NewReno modification to TCP's fast recovery algorithm - rfc 3782, April 2004.
23. R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 2001.
24. C. Ja and P. Agrawal. Congestion or corruption? A strategy for efficient wireless TCP sessions. In *Proc. of IEEE Symposium on Computers and Communications*, pages 262–268, Los Alamitos, CA, USA, 1995.
25. V Jacobson. Congestion avoidance and control. In *Proc. ACM SIGCOMM*, pages 314–329, Stanford, CA, August 1988.
26. R. Johari and D. Tan. End-to-end congestion control for the internet: delays and stability. *IEEE/ACM Trans. Networking*, 9(6):818–832, December 2001.
27. Kazza. <http://www.kazza.com>.
28. F. P. Kelly. Fairness and stability of end-to-end congestion control. *European Journal of Control*, 9:159–176, 2003.
29. F. P. Kelly, A. Maulloo, and D. Tan. Rate control for communication networks: shadow prices, proportional fairness, and stability. *Journal of the Operational Research Society*, 49:237–252, 1998.
30. H. Khalil. *Nonlinear Systems (3rd edition)*. Prentice Hall, 2001.
31. T. Kim, S. Lu, and V. Bharghavan. Improving congestion control performance through loss differentiation. In *Proc. ICPP Workshop*, pages 140–145, 1999.
32. S. Kunniyur and R. Srikant. End-to-end congestion control: utility functions, random losses and ECN marks. In *Proc. IEEE INFOCOM*, pages 1323–1332, Tel-Aviv, Israel, March 2000.
33. S. Kunniyur and R. Srikant. A time scale decomposition approach to adaptive ECN marking. *IEEE Trans. Automat. Contr.*, 47(6):882–894, June 2002.
34. A. Lakshmikantha, R. Srikant, and C. L. Beck. Differential equations models of flow-size based priorities in internet routers. *International Journal of Systems, Control and Communications*, 2:179–196, 2010.
35. J. Joon Lee, Fang Liu, and C. C.J. Kuo. End-to-end wireless TCP with non-congestion packet loss detection and handling. In *Proc. of the SPIE*, pages 104–113, San Jose, USA, January 2003.
36. J. Liu, I. Matta, and M. Crovella. End-to-end inference of loss nature in a hybrid wired/wireless environment. In *Proc. WiOpt*, 2003.
37. S. Liu, T. Basar, and R. Srikant. TCP-Illinois: A loss and delay-based congestion control algorithm for high-speed networks. In *Proc. First International Conference on Performance Evaluation Methodologies and Tools (VALUETOOLS)*, October 2006.
38. S. H. Low. A duality model of TCP and queue management algorithms. *IEEE/ACM Trans. Networking*, 11:525–536, 2002.
39. S.H. Low and D.E. Lapsley. Optimization flow control, I: Basic algorithm and convergence. *IEEE/ACM Trans. Networking*, 7(6):861–875, December 1999.
40. P. Marbach and R. Berry. Downlink resource allocation and pricing for wireless networks. In *Proc. IEEE INFOCOM*, New York, USA, June 2002.
41. S. Mascolo, C. Casetti, M. Gerla, M.Y. Sanadidi, and R. Wang. TCP Westwood: Bandwidth estimation for enhanced transport over wireless links. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 287–297. ACM New York, NY, USA, 2001.
42. S. Mascolo and F. Vacirca. The effect of reverse traffic on TCP congestion control algorithms. In *Proc. of Protocols for Fast Long-distance Networks*, February 2006.
43. F. Paganini, Z. Wang, J. Doyle, and S. Low. A new TCP/AQM for stable operation in fast networks. In *Proc. IEEE INFOCOM*, pages 96–105, San Francisco, CA, March 2003.
44. C. Parsa and J.J. Garcia-Luna-Aceves. Improving TCP congestion control over internet with heterogeneous media. In *Proc. of the ICNP*, pages 213–221, 1999.
45. PPLive – Internet Peer-to-peer video streaming. <http://www.pplive.com>.
46. P. Ranjan, R. La, and E. Abed. Global stability conditions for rate control with arbitrary communication delays. *IEEE/ACM Trans. Networking*, 14(1):94–107, 2006.
47. K. Ratnam and I. Matta. WTCP: an efficient mechanism for improving wireless access to TCP services.

- International Journal of Communication Systems*, 16(1):47–62, February 2003.
48. N. Samaraweera. Non-congestion packet loss detection for TCP error recovery using wireless links. *IEE Proceedings of Communications*, 146(4):222–230, August 1999.
 49. S. Sastry. *Nonlinear Systems, Analysis, Stability and Control*. Springer Verlag, New York, NY, 1999.
 50. S. Shenker. Fundamental design issues for the future internet. *IEEE J. Select. Areas Commun.*, 13:1176–1188, 1995.
 51. P. Sinha, T. Nandagopal, N. Venkitaraman, R. Sivakumar, and V. Bharghavan. A wireless transmission control protocol for cdpd. In *Proc. of IEEE Wireless Communications and Networking Conference*, pages 953–957, Piscataway, NJ, USA, January 1999.
 52. P. Sinha, T. Nandagopal, N. Venkitaraman, R. Sivakumar, and V. Bharghavan. WTCP: a reliable transport protocol for wireless wide-area networks. *Wireless Networks*, 8(2-3):301–316, 2002.
 53. R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, Boston, 2004.
 54. Y. Tobe, Y. Tamura, A. Molano, S. Ghosh, and H. Tokuda. Achieving moderate fairness for udp flows by pathstatus classification. In *Proc. of 25th Annual IEEE Conf. on Local Computer Networks*, pages 252–261, Tampa, FL, USA, November 2000.
 55. G. Vinnicombe. On the stability of end-to-end congestion control for the internet. Technical Report CUED/F-INFENG/TR.398, University of Cambridge, Cambridge, UK, 2001.
 56. D.X. Wei, C. Jin, S.H. Low, and S. Hegde. FAST TCP: motivation, architecture, algorithms, performance. *IEEE/ACM Trans. on Networking*, 14(6):1246–1259, 2006.
 57. H. Yaiche, R. R. Mazumdar, and C. Rosenberg. A game-theoretic framework for bandwidth allocation and pricing in broadband network. *IEEE/ACM Trans. Networking*, pages 667–678, October 2000.
 58. G. Yang, L.-J. Chen, T. Sun, M. Gerla, and M. Y. Sanadidi. Smooth and efficient real-time video transport in presence of wireless errors. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2:109–126, 2006.
 59. G. Yang, M. Gerla, and M. Y. Sanadidi. Adaptive video streaming in presence of wireless errors. In *Proc. ACM MMNS*, San Diego, USA, January 2004.
 60. Y. Yang, H. Zhang, and R. Kravets. Channel quality based adaptation of TCP with loss discrimination. In *Proc. of IEEE Global Telecommunications Conference*, pages 2026–2030, Piscataway, NJ, USA, 2001.
 61. L. Ying, G. Dullerud, and R. Srikant. Global stability of internet congestion controllers with heterogeneous delays. *IEEE/ACM Trans. Networking*, 14(3):579–591, 2006.