Distributed Rate Control of Prioritized Flows with End-to-End Delay and Rate Constraints

Zvi Rosberg[1] and Fariza Sabrina[1]

Abstract

To address end-to-end quality of service (QoS) requirements, we derive a novel distributed combined rate and end-to-end delay control in a network serving multi-class flows with priority packet scheduling. We show that the control is globally asymptotically stable without information time lags. The stable flows attain the end-to-end delay requirements and have no packet loss. We also show that by enhancing the network with bandwidth reservation and admission control, minimum rate is also guaranteed. The stability with very long time lags of a discrete time version control with non-greedy flows and random packet arrivals is studied numerically by an NS2 packet-based simulation of the Australian Academic and Research Network.

Index Terms

Rate control, End-to-end delay, Multi-service flows, Bandwidth guarantee, Stability, Internet, QoS.

I. Introduction

Rate control of flows in communication networks has been studied mainly in the context of transfer control protocol (TCP). A rigorous analytical foundation was paved in [8], where *proportional fair* rates have been advocated and rate control is modeled as an optimization problem subject to link capacity constraints using a fluid model. The authors derived two distributed control algorithms; one is based on the gradient search in the primal problem and the other on the solution of the dual problem. The global asymptotic stability without information time lags of those algorithm was also established by the authors. It is believed that fluid models are adequate for large networks such as the Internet [19] and can explain stability properties and provide insight for new control schemes, e.g., FAST TCP [24]. Our own unpublished prototype implementation in network processors of an enhanced dual algorithm also demonstrates very close performance matching between a theoretical fluid network and a real network comprising real routers and 100 Mb/s links.

CSIRO ICT Centre, PO Box 76, Epping, NSW 1710, Sydney, Australia; Email: {zvi.rosberg,fariza.sabrina}@csiro.au.

Following [8], fluid models have been subsequently used to characterize fairness and stability and to derive rate-based and window-based controls [11] [12] [13] [15]. Other end-to-end window-based congestion controls using packet round-trip-time (RTT) information have been also introduced [3] [5] [6] [9] [26]. See [24] for an extensive list of references and discussion.

Extended proportional fairness which includes proportional fairness and max-min fairness [4], was introduced in [15] by adding a fairness level parameter. The authors derived a fair end-to-end window-based control and proved its global asymptotic stability without time lags. Unlike the algorithms of [8] [13], their algorithm uses RTT and does not require feedback information from the routers, hence is suitable for TCP congestion control. The stability and performance of another fair end-to-end window-based control using RTT, was analyzed in [16]. The local asymptotic stability of its discrete-time version has been subsequently established in [23].

The window-based controls above (and others) assume that at any time t, the rate of every flow n, $x_n(t)$, its window size, $w_n(t)$, and its RTT, $RTT_n(t)$, are related by $x_n(t) = w_n(t)/RTT_n(t)$. This relation is a deterministic version of Little's theorem for ergodic queueing systems and will be referred to as the *delay-window assumption*. Recently, this relation was relaxed in [22].

A control combining the explicit congestion notification (ECN) marking scheme with adaptive virtual queues was used in [10] [11]. The stability of the primal and dual controls under arbitrary time lags have been studied in [7] [14] [27] and references there. Notable is the most general sufficient condition for global stability derived in [27].

Although window-based controls use packet delay information, i.e., RTT, the delays are incorporated into the model only through the *delay-window assumption*, which is an average law. More explicit incorporations of link delays were proposed and analyzed in [1] [16] [21] [25]. The model of [21] extends [8] by representing each link delay as a function of its total load. Unlike [21], where the actual link delay trajectories are not modeled, the authors of [1] [16] and [25] incorporated them by using differential equations to specify their dynamic and studied the global asymptotic stability of primal, dual and combined primal-dual controls, respectively.

All the control schemes above assume FIFO packet scheduling and aim at stable fair rates subject only to the link capacity constraints. Such controls are suitable for TCP flows requiring only *best-effort* service. However, multimedia application flows, which usually use the user-datagram-protocol (UDP), require also quality of service (QoS) guarantee, e.g., maximum end-to-end delay, minimum bandwidth and maximum packet loss.

Motivated by the QoS needs, we consider the most explicit fluid delay model of [1] [16] [21] [25] and extend it (for the first time) to flows with priority packet scheduling and end-to-end packet delay constraints. The system model is given in Section II and the combined rate and end-to-end delay control with priority scheduling is derived in Section III. Extensions and implementation issues are also discussed in Section III. In Section IV, we studied the stability of a discrete time version of our control in the presence of very long time lags. The study uses an NS2 packet-based simulation of the Australian Academic and Research Network with non-greedy flows and random packet arrivals.

II. System Model

Consider a fluid network model comprising a set of N flows and L links with link capacities of $\boldsymbol{c} = (c_1, \ldots, c_L)^T$. Assume a fixed routing matrix, $\boldsymbol{A} = [A_{n,l}]$, where $A_{n,l} = 1$, if flow n traverses link l, and 0 otherwise. Each link has a finite buffer sufficiently large to absorb a maximum end-to-end queueing delay, which is also subject to our control.

Each flow, n, is associated with a static priority index, $p_n \in \{1, \ldots, P\}$, determining its scheduling preference in the output links. By convention, priority index 1 is the highest. To prevent starvation of low priority flows, we upper bound by $0 \leq u(p) \leq 1$ the utilizations of every link allocated to flows with priorities $q,\ q \leq p$.

Let $\boldsymbol{x}(t) = \big(x_1(t), \ldots, x_N(t)\big)^T$ be the flow rates at time t; $R(n)$ be the set of links comprising the route of flow n; and $T(l)$ be the set of flows traversing through link l. For every p and t, let $y_l(p,t)$ be total rates of flows with priorities $q,\ q \leq p$, traversing link l at time t, i.e.,

$$y_l(p,t) = \sum_{\{n \in T(l): p_n \leq p\}} x_n(t).$$

For a combined rate and end-to-end delay control, each flow is associated with a rate utility function, U_n, and a route delay penalty function, L_n. We assume that $U_n = U_n(x_n)$ is a differentiable, strictly increasing and strictly concave function; and $L_n = x_n D_n(p_n, t)$, where $D_n(p_n, t)$ is the end-to-end queueing time of flow n at time t, assuming a preemptive priority packet scheduling regime. Note that we assume that the route delay penalty function of flow n is proportional to its rate, x_n, which stems from the rational that each bit of flow n is experiencing a delay of $D_n(p_n, t)$. Also note that $D_n(p_n, t)$ is determined by the trajectories $\{\boldsymbol{x}(s), s < t\}$

(excluding $s = t$) and is given by

$$D_n(p_n, t) = \sum_{l \in R(n)} d_l(p_n, t), \tag{1}$$

where $d_l(p_n, t)$ is the time to clear the fluid backlog of flows with priorities $q, q \leq p_n$, residing in the buffer of link l at time t. The time derivative of $d_l(p, t)$ is given by

$$\dot{d}_l(p,t) = \begin{cases} \frac{y_l(p,t)}{u(p)c_l} - 1, & \text{if } d_l(p,t) > 0; \\ \left[\frac{y_l(p,t)}{u(p)c_l} - 1\right]^+, & \text{if } d_l(p,t) = 0. \end{cases} \tag{2}$$

Remark 1: Queueing delays and buffer occupancies are related by $b_l(p, t) = d_l(p, t)u(p)c_l$, $\forall\ q \leq p$. Thus, delay and buffer occupancy controls are equivalent. ■

To combine rate utilities and delay penalties into a single objective function, we define positive delay prices $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)^T$ and an objective function given by

$$J_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \sum_{n=1}^{N} \big[U_n(x_n) - \alpha_n D_n(p_n) x_n\big]. \tag{3}$$

First, we derive a globally asymptotically stable distributed rate control with fixed route delay prices. Then, we extend it by adapting the route delay prices to obtain a globally asymptotically stable combined control of both, the rates and the end-to-end delays. The stable variables will be superscripted with a "$*$".

III. A Combined Rate and End-to-End Delay Control

A. Rate Control with Fixed Prices with FIFO Scheduling

In this subsection, we analyze the single priority case and omit p from our notation. The multiple priority case is analyzed in Subsection III-C by reducing it into a single priority case.

Consider the following rate control for $J_{\boldsymbol{\alpha}}(\boldsymbol{x})$ used in [1]:

$$\dot{x}_n(t) = U_n'\big(x_n(t)\big) - \alpha_n(t) D_n(t), \tag{4}$$

where $\alpha_n(t) = \alpha_n$ and $U_n'(x_n(t)) := \frac{dU_n(x)}{dx}\Big|_{x=x_n(t)}$.

Equations (1), (2) and (4) determine the time trajectories of $\big(\boldsymbol{x}(t), \boldsymbol{d}(t), \boldsymbol{D}(t)\big)$.

Remark 2: The control of (4) cannot be specified as the following primal algorithm in the resource allocation formalization context of [20, Sec. 3.1]:

$$\dot{x}_n(t) = k_n(x_n)\Big(U_n'\big(x_n(t)\big) - \sum_{l \in R(n)} f_l(y_l(t))\Big),$$

where $f_l(\cdot)$ is non-decreasing and continuous whose integral up to y approaches infinity as y approaches infinity.

The reason is that the link queueing time, $d_l(t)$, is a function of $\{y_l(s);\ s \leq t\}$ and cannot be expressed only by $y_l(t)$. Likewise, (4) cannot be specified as the *adaptive virtual queue* (AVQ) algorithm of [20, Sec. 3.3] since AVQ is a also a function $y_l(t)$ and another external control variable. Therefore, the stability proofs of [20] cannot be applied to (4). ■

Remark 3: We cannot apply the derivation of [1] as well since it linearizes the delay dynamic by $\dot{d}_l(t) = y_l/c_l - 1$ and assumes unique equilibrium bottleneck links. ■

The advantage of using $D_n(t)$ over the route penalty $\sum_{l \in R(n)} f_l(y_l(t))$ of Remark 2 or the AVQ algorithm is that it reflects the real delay of the model and does not require a signaling protocol for delivering the route penalty. Additionally, as the AVQ algorithm, we show in Remark 6 below that the control of (4) maximizes $\sum_n U_n(x_n)/\alpha_n$ subject to the link capacity constraints.

Firstly, we assume that link buffers are sufficiently large to absorb a given maximum end-to-end queueing delay. In the next step, where an upper bound, $\overline{D}_n$, is enforced on every D_n^* and a combined rate and end-to-end delay control is used, the buffer sizes can be set to $b_l = c_l \times \max_{n \in T(l)} \overline{D}_n$ (see Remark 1).

Let $\boldsymbol{A}_b$ denote the matrix comprising the columns of $\boldsymbol{A}$ corresponding to the equilibrium bottleneck links. The uniqueness of $\boldsymbol{A}_b$ was established in [15].

Lemma 1: For every fixed $\boldsymbol{\alpha} > 0$, the rate control of (4) with delay dynamic of (2) has a unique stable point $(\boldsymbol{x}^*, \boldsymbol{D}^*)$. If $\boldsymbol{A}_b$ is a full rank matrix, then $\boldsymbol{d}^*$ is also unique.

Proof: We show that the equilibrium point of the problem of [15] and that of our control given by (1), (2) and (4) are the same.

Given $\boldsymbol{\alpha} > 0$ and $P = 1$, a stable state, $(\boldsymbol{x}^*, \boldsymbol{d}^*)$, of (2) and (4) is characterized by two conditions: (i) $\dot{d}_l(t) = 0,\ \forall\ l$; and (ii) $\dot{x}_n(t) = 0,\ \forall\ n$. By (2), condition (i) implies

$$\boldsymbol{A}^T \boldsymbol{x} \leq \boldsymbol{c} \quad ; \quad \operatorname{diag}(\boldsymbol{d})\big(\boldsymbol{A}^T \boldsymbol{x} - \boldsymbol{c}\big) = 0. \tag{5}$$

Noting that $\boldsymbol{A}\boldsymbol{d} = \boldsymbol{D}$, condition (ii) and (4) imply

$$\operatorname{diag}\left(\frac{1}{\boldsymbol{U}'(\boldsymbol{x})}\right) \boldsymbol{A}\boldsymbol{d} = 1/\boldsymbol{\alpha}. \tag{6}$$

Assuming a utility function of the form $U_n(x) = p_n \ln(x)$, then for any given $(\boldsymbol{A}, \boldsymbol{c}, \boldsymbol{\alpha})$, the equilibrium state equations (5) and (6) is a special case of the equations of [15, Eqs. (1)- (3)]

with propagation delays set to zero. Thus, by [15], there is a solution, $(\boldsymbol{x}^*, \boldsymbol{d}^*)$, for (5) – (6) and $\boldsymbol{x}^*$ is unique. The uniqueness of $\boldsymbol{D}^*$ follows by (6), which implies $D_n^* = U_n'(x_n^*)/\alpha_n$.

It is further shown in [15], that if $\boldsymbol{A}_b$ is a full rank matrix, then $\boldsymbol{d}^*$ is unique as well. It can further be verified that the proofs of [15] hold true for every set of strictly increasing and strictly concave functions $\{U_n(x)\}$. ■

Remark 4: It worth noting that if $U_n(x) = \beta_n \ln(x)$, the TCP window size variable, w_n, used in [15], equals β_n/α_n. ■

Remark 5: The uniqueness of $\boldsymbol{D}^*$, which holds true regardless of the rank of $\boldsymbol{A}_b$, is essential for the general applicability of our end-to-end delay control. ■

The rate control of (4) with link delay dynamics of (2) is covered by the combined primal/dual control of [25, Subsection III-C]. Using a passivity framework for network flow control, the authors have shown that if the routing matrix $\boldsymbol{A}$ is a full rank, the control of (4) without information time lags is globally asymptotically stable.

The following Theorem now follows from Lemma 1, Remark 5 the the global asymptotic stability proof of [25].

Theorem 1: For every fixed $\boldsymbol{\alpha} > 0$, the rate control of (4) is globally asymptotically stable, i.e., it converges to a unique equilibrium, $(\boldsymbol{x}^*, \boldsymbol{D}^*)$, from any initial state. It also minimizes $J_{\boldsymbol{\alpha}}(\boldsymbol{x}, t)$ with no bit loss. If $\boldsymbol{A}_b$ is a full rank matrix, then $\boldsymbol{d}^*$ is also unique. ■

Remark 6: Observe that for any given $\boldsymbol{\alpha} > 0$, (6) can be written as $U_n'(x_n)/\alpha_n = \sum_{l \in R(n)} d_l$, $\forall\ n$. Thus, Eqs. (5) and (6) are the Karush-Kuhn-Tucker conditions [2] for the optimal solution of $\max_{\boldsymbol{x} \geq 0} \sum_{n=1}^{N} U_n(x_n)/\alpha_n$ subject to $\boldsymbol{A}^T \boldsymbol{x} \leq \boldsymbol{c}$, where $\boldsymbol{d}$ is the Lagrangian multiplier vector. Thus, as the primal and the AVQ controls [20, Sec. 3], the control of (4) also solves the constrained optimization problem above.

Remark 7: Another optimization property of the control of (4) follows from Remark 4 showing that the delay prices, $\boldsymbol{\alpha}$, and the windows sizes of [15] are playing the same control role. Therefore, by [15, Theorems 3,4], the stable rates for each $\boldsymbol{\alpha}$ equal to one of the extended proportional fair rates proposed in [15]. ■

B. Combined Rate and Delay Control with FIFO Scheduling

From (6), end-to-end delays decrease with the delay prices, $\boldsymbol{\alpha}$. Hence, can be made arbitrary small by a proper adaption of $\boldsymbol{\alpha}$. Since we are interested in proportional fair rates and a distributed

control, we set $U_n(x) = \beta \ln(x)$ and adapt each α_n based only on $D_n(t)$.

Let $\boldsymbol{\alpha}(t)$ be the delay price vector used at time t, and refer to the event of $D_n(t) > \overline{D}_n$ as an *excess delay event*. Define the following adaptation of $\boldsymbol{\alpha}(t)$.

Whenever an *excess delay event* occurs at the source of flow n, the event is disseminated by an *excess delay signal* to every flow source k such that $R(k) \cap R(n) \cap \{l, b_l > 0\} \neq \phi$. That is, to every source of flow k sharing a common congested link with flow n.

Since such protocol is not part of the standard protocols implemented by current Internet Protocol (IP) routers, we propose the following dissemination protocol.

> *For every flow n, if $D_n(t) > \overline{D}_n$, the source of flow n sends an "excess delay event" signal along its route $R(n)$. Every router of a congested link $l \in R(n)$, i.e, with $b_l > 0$, then disseminates the signal to every flow source $k \in T(l)$.*

In Subsection III-F, we describe a generic implementation of the *delay dissemination signalling (DDS)* protocol which has been developed in our laboratory using network processor units (NPUs) (final system testing).

Given that a DDS protocol is in place, we amend the rate control of (4) by

$$\dot{\alpha}_n(t) = \begin{cases} \frac{1}{\alpha_n(t)}, & \text{excess delay signal is received,} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

By (7), every flow k traversing a congested link which is in the path of one of the flows, say n, with $D_n(t) > \overline{D}_n$, adapts its $\alpha_k(t)$ by a logarithmic increase. The $\boldsymbol{\alpha}(t)$ of other flows are unchanged.

Theorem 2: Given a vector of delay bounds $\overline{\boldsymbol{D}}$ and a vector of initial delay prices $\boldsymbol{\alpha}(0)$, the combined rate and delay control of (4) and (7) is globally asymptotically stable. That is, from any initial state it converges to a unique equilibrium $(\boldsymbol{\alpha}^*, \boldsymbol{x}^*, \boldsymbol{D}^*)$. It also minimizes $J_{\boldsymbol{\alpha}^*}(\boldsymbol{x}, t)$ with $\boldsymbol{D}^* \leq \overline{\boldsymbol{D}}$ and no bit loss.

Proof: By remark 4, β_n/α_n equals the TCP window size, w_n, of [15], which is the stable backlog of flow n in the network. Since some of the backlog is in transit and the rest is distributed amongst the link buffers, it follows that for every link l,

$$b_l^* = \sum_{k \in T(l)} \rho_{k,l} \; w_k = \sum_{k \in T(l)} \rho_{k,l} \; \frac{\beta_k}{\alpha_k}$$

for some $0 \leq \rho_{k,l} \leq 1$.

Thus, by also invoking Remark 1 we have

$$D_n^* = \sum_{l \in R(n)} \frac{b_l^*}{c_l} = \sum_{l \in R(n)} \frac{1}{c_l} \sum_{k \in T(l)} \rho_{k,l}\ w_k = \sum_{l \in R(n)} \frac{1}{c_l} \sum_{k \in T(l)} \rho_{k,l}\ \frac{\beta_k}{\alpha_k}. \tag{8}$$

By (8), each D_n^* decreases as the vector $\boldsymbol{\alpha}$ increases. Therefore, there is an $\overline{\boldsymbol{\alpha}}$ such that $\boldsymbol{D}^* \leq \overline{\boldsymbol{D}}$ for every $\boldsymbol{\alpha}$, $||\boldsymbol{\alpha}|| > ||\overline{\boldsymbol{\alpha}}||$. Consequently, by the monotonic adaptation of (7) there is a t^* such that $\boldsymbol{D}(t) \leq \overline{\boldsymbol{D}}$, for every $t > t^*$.

Let t_k be the time of the k^{th} excess delay event. By the monotonic bounding argument above, $\boldsymbol{\alpha}(t_k)$ is an increasing bounded sequence converging to a finite $\boldsymbol{\alpha}^*$. The global asymptotic stability and the minimization of $J_{\boldsymbol{\alpha}^*}(\boldsymbol{x}, t)$ now follows from the results derived for the fixed prices case with $\boldsymbol{\alpha} = \boldsymbol{\alpha}^*$. ■

Remark 8: The stable state, $(\boldsymbol{\alpha}^*, \boldsymbol{x}^*, \boldsymbol{d}^*)$, may depended on the initial values $\boldsymbol{\alpha}(0)$. Indeed, for the single link and single flow case, any (x_1, d_1) satisfying $x_1 = c_1$ and $d_1 \leq \overline{D}_1$, is a stable state that can attract the process $\big(\boldsymbol{\alpha}(t), \boldsymbol{x}(t), \boldsymbol{d}(t)\big)$. ■

Remark 9: If α_n is adapted based on local excess delays and are are not disseminated to other sources, then the delay constraints may not be met. Indeed, consider a single link and two flows with $\overline{D}_1 < \overline{D}_2$. Clearly, for any stable control, we have $D_1^* = D_2^* = d_1^*$. However, in states where $\overline{D}_1 < d_l(t) < \overline{D}_2$, only flow one decreases its rate while flow two takes over the released rate. Furthermore, flow two has no incentive to decrease its rate so as to reduce its delay below $\overline{D}_2$. Consequently, $d_1(t)$ will converge to $d_1^* > \overline{D}_1$. ■

C. Combined Rate and Delay Control with Priority Scheduling

With multiple priorities, the delay trajectories are given by (2) and the rate control is the same as for the single priority case, i.e., given by (4). Recall however, that $D_n(t)$ in (4) is now determined by the priority of flow n. Also, with priority scheduling, the *excess delay signals* have to be amended with the priority of the flow which has experienced the *excess delay event*. Consequently, the delay price of flow n with priority p_n is adapted by

$$\dot{\alpha}_n(t) = \begin{cases} \frac{1}{\alpha_n(t)}, & \text{an excess delay signal with} \\ & \text{priority } q \geq p_n \text{ is received,} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

We will show that the priority case can be reduced into a single priority control system. To this end, we extend the routing matrix, $\boldsymbol{A}$, into an N by $P \cdot L$ routing matrix, $\tilde{\boldsymbol{A}}$, defined as

follows. Every link column, l, is extended to a block of P consecutive columns labeled by $g_i(l), i = 1, \ldots, P$. The elements of $\tilde{\boldsymbol{A}}$, $\tilde{A}_{n,g_i(l)}$, are defined by

$$\tilde{A}_{n,g_i(l)} = \begin{cases} 1, & \text{if } A_{n,l} = 1 \text{ and } i \geq p_n, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

For every l, the link capacity of $g_i(l)$ is set to $u(i)c_l$. Note that $\tilde{\boldsymbol{A}}$ represents a routing matrix by which the rate of each flow n using link l in $\boldsymbol{A}$, also uses links $g_i(l),\ i \geq p_n$. Hence, $\tilde{\boldsymbol{A}}$ reflects the backlogs generated by flows with priority packet scheduling. The number of rows of $\tilde{\boldsymbol{A}}$ and $\boldsymbol{A}$ are the same. Hence, the flow rate vector, $\boldsymbol{x}$, is the same for both representations.

With the new representation of the priority control system, each link delay, $d_l(i,t)$, corresponds to $d_{g_i(l)}(t)$ and the end-to-end delays, $\boldsymbol{D}(i,t)$, correspond to $\tilde{\boldsymbol{D}}(t)$ defined by

$$\tilde{\boldsymbol{D}}(t) = \tilde{\boldsymbol{A}}\tilde{\boldsymbol{d}},$$

where $\tilde{\boldsymbol{d}}$ is the column vector of $\{d_{g_i(l)}(t)\}$.

The uniqueness of $(\boldsymbol{x}^*, \boldsymbol{D}^*)$ and the global asymptotic stability of the combined rate and delay control with priority scheduling follow from Subsections III-A and III-B.

D. Control with Minimum Rate Constraints

Suppose that the flows also have minimum required rates denoted by $\overline{\boldsymbol{r}} = (\overline{r}_1, \ldots, \overline{r}_N) \geq 0$. Denote by $\bar{\boldsymbol{x}}(t) = \boldsymbol{x}(t) - \overline{\boldsymbol{r}}$ the residual rates at time t and assume a bandwidth reservation protocol under which routers know their total reserved capacity. That is, each router knows $\bar{r}(l) = \sum_{n \in T(l)} \overline{r}_n$ of every link l attached to it.

By further applying the necessary admission control, we can assume that the residual capacity of each link l, $\bar{c}_l = c_l - \bar{r}(l)$, is not negative. Let $\bar{U}_n(x_n)$ be a residual utility function defined for $x_n > \overline{r}_n$.

Since a capacity of $\bar{r}(l)$ is used for transmitting the reserved flow rates on each link l, its delay trajectories, $d_l(t)$, is determined by its residual rate, $\bar{y}_l(t)$, and its residual capacity, $\bar{c}_l$. That is, they are given by (2) after replacing $\{y_l(t)\}$ and $\{c_l\}$ with $\{\bar{y}_l(t)\}$ and $\{\bar{c}_l\}$, respectively.

The combined rate-delay control with minimum bandwidth is obtained by replacing (4) with

$$\dot{x}_n(t) = \bar{U}_n'\big(x_n(t) - \overline{r}_n\big) - \alpha_n(t) D_n(t), \tag{11}$$

where $\bar{U}_n'(0)$ is arbitrarily large so as to force $x_n(t) \geq \overline{r}_n$ for every t.

The adaption of the delay prices given by (7) is unchanged. All stability properties of Sections III-A, III-B and III-C hold true for the transformed system.

E. Stability Under Delayed Information

We showed global asymptotic stability without information time lags. That is, the trajectories at time t are determined based on the immediate state information, $\boldsymbol{x}(t)$ and $\boldsymbol{D}(t)$. In practice, the states available to the flow sources at time t have time lags depending on the propagation times between the flow sources and the router ports of the various links.

Using a standard time lag model (e.g., [27]), we can examine the control stability with information time lags by numerically computing the rate and delay trajectories of our control for specific networks. We examine two types of trajectories, "microscopic" (at time granularity of 1 millisecond) and "macroscopic" (at time granularity of 1 second). For each view, a trajectory point represents a running average within a time window of 1 millisecond or second, respectively.

Figure 2 depicts the end-to-end "microscopic" and "macroscopic" delay trajectories of 3 flows in the Australian Academic and Research Network (AARNet3) (Fig. 1) described in Section IV below. The trajectories are depicted after numerical stabilization and illustrate oscillations in the "microscopic" view. However, the oscillations are bounded within the delay requirements. In the "macroscopic" view, the trajectories exhibit significantly less oscillations. For practical purposes, such "mean-stability" is satisfactory. The same behavior is observed for the rate trajectories as well as in the NS2 packet simulation described in Section IV below.

F. DDS Protocol Implementation

A prototype implementation of the excess delay dissemination signalling (DDS) protocol proposed in Subsection III-B is in its final stages of system testing in our laboratory. The implementation is done in EZchip network processor units (NPUs) attached to CISCO routers. The NPUs intercept all crossing packets and execute the DDS protocol on behalf of the routers. Payload packet processing is done by a microcode program running in a specialized network processor at a line speed of 1 Gb/s. The DDS signalling packets are processed by the Pentium III processor of the NPU.

Beside dissemination, the signalling packets also probe the RTT along each flow path, perform bandwidth reservation, admission control, flow classification, flow marking and rate policing. The packet scheduling is done by the routers based on the NPU packet marking.

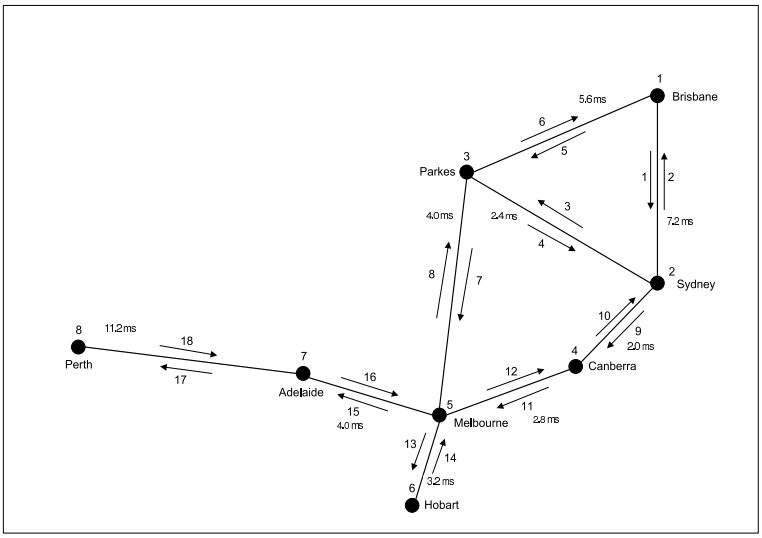

Fig. 1. AARNet3 optical network topology, Australia.

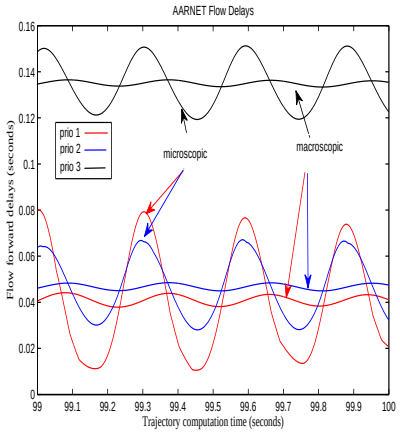

Fig. 2. End-to-end queueing time with 3 priorities.

The end-to-end queueing time, $D_n(t)$, used for the policed rate of flow n, is extracted from RTT by $D_n(t) = RTT(t) - baseRTT(t)$, where $baseRTT(t) = \min_{s \le t} RTT(s)$ (as in TCP Vegas). If the end-to-end delay target is RTT, they are readily available by time stamping the DDS probes at the sources. If the delay target is the forward delay, they are available by a time synchronization algorithm along with time stamping of the DDS probes at the sources and at the destinations. Note that current technology can achieve a universal time clock (UTC) with a precision of 20 nanoseconds [17]. Additionally, in the implementation we use an estimator of $D_n(t)$ based on filtered values and damped historical values.

A detailed description of the architecture for a combined rate and delay control is reported in [18]. We further note that our unpublished prototype implementation with NPUs of Kelly's dual algorithm extended to end-to-end delay control demonstrates performance and stability that match the theoretical fluid model of six routers and 100 Mb/s links.

IV. An NS2 Simulation Case Study

To test our control in a realistic environment we developed a packet-based simulator using the NS2 network simulation package and simulated the Australian Academic and Research Network (AARNet3) shown in Fig. 1. The fundamental differences between the theoretical fluid model and the simulated system are the following.

- Transmission units are packets of size 1500 bytes rather than fluid.
- The greedy uniform influx of fluid is replaced by a non-greedy stationary stochastic packet arrival process. Specifically, for each flow n, the packet inter-arrival times are independent and uniformly distributed in an interval $[0.8\mu_n, 1.2\mu_n]$, where $\{\mu_n\}$ are set in a range resulting flow bit rate of 48 - 480 Mb/s (4 - 40 K packets per second).
- The continuous time rate policing is replaced by a standard token bucket policer with adaptive token generation rate.
- The large buffer assumption is replaced with a buffer that can store 500 packets per outgoing link.
- The theoretical strict priority scheduling with link utilization upper bounds $\{u_l(p)\}$ is implemented by using transmission frames each comprising at most 30 packet transmissions. Within each frame of link l, a packet with priority p is scheduled before a packet with priority $q,\ q > p$, however no more than a proportion of $u_l(p)$ of the frame size. If packets of priority p are exhausted before the bound $u_l(p)$ is reached, the next priority packets are scheduled. If the buffer is completely exhausted, a new frame starts before a total of 30 packets are transmitted within the current frame.

The simulated network comprises eight nodes (numbered 1 to 8) and nine full-duplex core links of capacity 0.6 Gb/s each, i.e., 18 unidirectional links numbered 1 to 18. The propagation delay of each core link is a function of the link distance and is shown in the figure in milliseconds.

The flows comprise 56 one-way client-server connections between any pair of core nodes using the shortest-hop-count route. Flows require different end-to-end delays: 30% of the flows require at most 100 milliseconds one-way queueing time (denoted by class 1); another 30% of them require at most 150 milliseconds (denoted by class 2); and the rest (40%) require at most 200 milliseconds (denoted by class 3).

We compared between a single and three priority cases, where class i flows are assigned priority i. With the prioritized scheme, we set the link bandwidth utilization to $u(1) = 0.5$, $u(2) = 0.8$ and $u(1) = 1.0$, for flows with priorities 1, 1-2 and 1-3, respectively. The values of $\boldsymbol{u}$ are selected heuristically. For proportional fairness rates, the utility function of each flow is set to $U_n(x) = \ln(x)$ and flow rates are updated every millisecond.

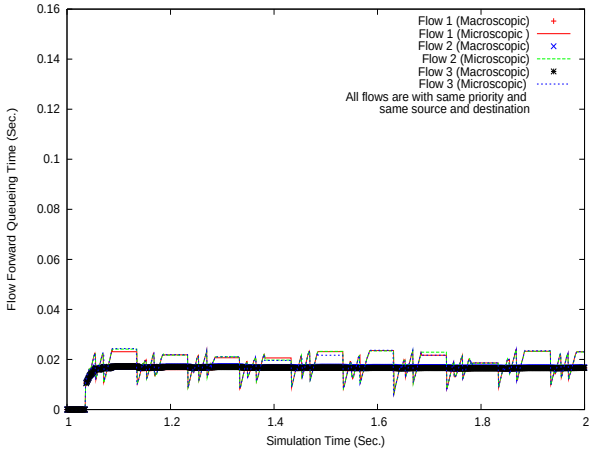

Fig. 3. End-to-end queueing time without priorities.

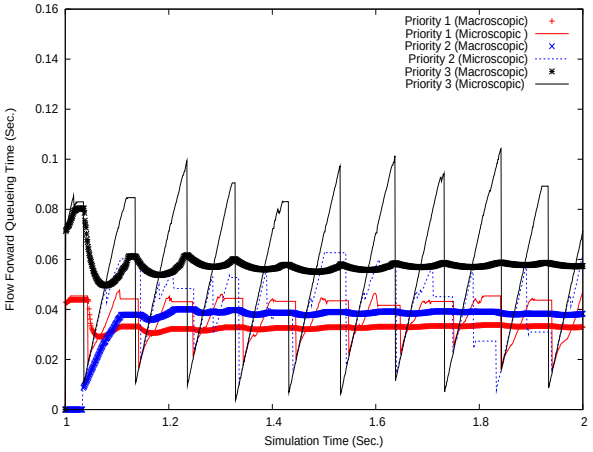

Fig. 4. End-to-end queueing time with priorities.

The microscopic and macroscopic end-to-end delays of three flows (one from each class) from Brisbane to Perth are depicted in Figures 3 and 4. The graphs present the delay trajectories in the first second of the simulation.

Note that without priorities (Fig. 3), the trajectories of the three flows using the same path are identical, and with three priorities (Fig. 4) they are different. The trajectories are depicted from the beginning of the simulation and are converging to a stable oscillation pattern after less than 1.5 seconds. The trajectories of the individual packet delays (microscopic view) not stable in the strict sense. However, the oscillations are bounded within the delay requirements and the trajectories in the macroscopic view exhibits a pattern which is sufficiently stable for practical purposes. The illustrated pattern holds true also for other flows.

The rate trajectories are converging to a stable oscillation pattern after about 2 seconds. For illustration, the microscopic and macroscopic rate trajectories of three flows using the same path from Brisbane to Perth and FIFO scheduling are depicted in Fig. 5.

The performance comparison between FIFO and and three priority scheduling is given in Table I. We also simulated the network with other update frequencies, propagation delays and end-to-end delay requirement. The main conclusions of our simulation case study are:

- The trajectory fluctuation is reduced with the update frequency; likewise, its convergence rate is improved.
- The trajectory fluctuation is reduced when the end-to-end delay requirements is increased; likewise, its convergence rate is improved. That is, stability is sensitive to α.

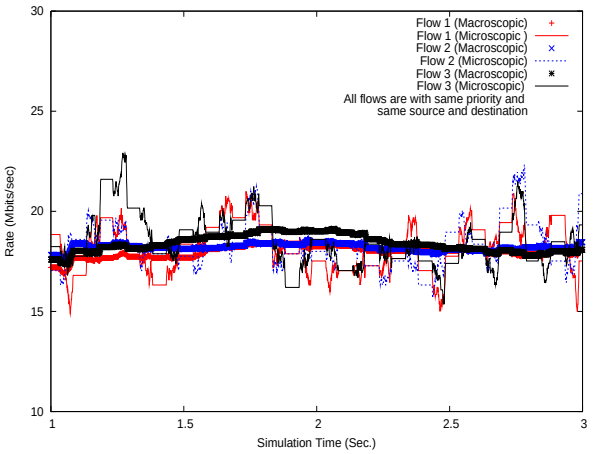

Fig. 5. "Microscopic" and "macroscopic" rates without priorities.

Measure	1 priority	3 priorities
Avg Rate - class 1	69 Mb/s	72 Mb/s
Avg Rate - class 2	72 Mb/s	84 Mb/s
Avg Rate - class 3	75 Mb/s	63 Mb/s
Overall Avg Rate	72 Mb/s	72.9 Mb/s
Avg Delay - class 1	0.008 sec	0.0151 sec
Avg Delay - class 2	0.0079 sec	0.0159 sec
Avg Delay - class 3	0.0076 sec	0.0217 sec
Overall Avg Delay	0.0080 sec	0.018 sec
Max Delay	0.03 sec	0.08 sec
Avg link queue	104 Packets	100 Packets
Max link queue	488 KBytes	300 KBytes
Convergence time	1.5 sec	1.5 sec

Table I. Performance with 1 and 3 priorities.

- The proportions of bandwidth allocated to different priorities, $\boldsymbol{u}$, affect the trajectory fluctuation. The control can be unstable with one $\boldsymbol{u}$ and stable with another.
- Three priority scheduling decreases the buffer occupancies compared to FIFO scheduling.
- Compared to FIFO scheduling, with three priority scheduling, the average rates of flows from class 1 and 2 (preferred flows) are increased and that of class 3 is decreased.
- With three priority scheduling, the average delay across priority classes are different and decreases with the priority level. With FIFO scheduling, the average delays are uniform across all classes.

REFERENCES

[1] T. Alpcan and T. Basar, "A Globally Stable Adaptive Congestion Control Scheme for Internet-Style Networks With Delay," *IEEE/ACM TON*, vol. 13, no. 6, pp. 1261–1274, Dec. 2005.

[2] D. P. Bertsekas, *Nonlinear Programming: Second Edition*. Athena Scientific, Belmont, MA, USA, 1999.

[3] L. S. Brakmo and L. L. Peterson, "TCP Vegas: end-to-end congestion avoidance on a global internet," *IEEE JSAC*, vol. 13, no. 8, pp. 1465–1480, Oct. 1995.

[4] A. Charny, "An algorithm for rate allocation in a packet-switching network with feedback," M.A. thesis, MIT, Cambridge, MA, 1994.

[5] R. Jain,, "A delay-based approach for congestion avoidance in interconnected heterogeneous computer networks," ACM Comput. Commun. Rev., vol. 19, no. 5, pp. 56–71, Oct. 1989.

[6] S. Jin, L. Guo, I. Matta, and A. Bestavros, "A spectrum of TCP-friendly window-based congestion control algorithms," *IEEE/ACM TON*, vol. 11, no. 3, pp. 341-355, Jun. 2003.

[7] R. Johari and D. Tan, "End-to-end congestion control for the Internet: Delays and stability," *IEEE/ACM TON*, vol. 9, no. 6, pp. 818–832, Dec. 2001.

[8] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication networks: Shadow price proportional fairness and stability,” *J. Oper. Res. Soc.*, vol. 49, pp. 237-252, 1998.

[9] A. Kuzmanovic and E. Knightly, “TCP-LP: a distributed algorithm for low priority data transfer,” in Proc. IEEE INFOCOM, 2003, pp. 1691-1701, 2003.

[10] S. Kunniyur and R. Srikant, “A time-scale decomposition approach to adaptive explicit congestion notification (ECN) marking,” *IEEE Trans. Automat. Contr.*, vol. 47, no. 6, pp. 882-894, Jun. 2002.

[11] S. Kunniyur and R. Srikant, “End-to-end congestion control schemes: Utility Functions, Random Losses and ECN Marks,” *IEEE/ACM TON*, vol. 11, no. 5, pp. 689–702, Oct. 2003.

[12] C. M. Lagoa, H. Che, and B. A. Movsichoff, “Adaptive Control Algorithms for Decentralized Optimal Traffic Engineering in the Internet,” *IEEE/ACM TON*, vol. 12, no. 3, pp. 415–428, June 2004.

[13] S. H. Low and D. E. Lapsley, “Optimization flow control, I: basic algorithm and convergence,” *IEEE/ACM TON*, vol. 7, pp. 861-875, Dec. 1999.

[14] L. Massoulie, “Stability of distributed congestion control with heterogeneous feedback delays,” *IEEE Trans. Automat. Contr.*, vol. 47, no. 6, pp. 895-902, Jun. 2002.

[15] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM TON*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[16] F. Paganini, Z. Wang, S. H. Low and J. C. Doyle, “A new TCP/AQM for stability and performance in fast networks,” in Proc. of IEEE INFOCOM, April 2003.

[17] T. E. Parker, D. Matsakis, Time and Frequency Dissemination: Advances in GPS Transfer Techniques, *GPS World*, pp. 32-38, Nov. 2004.

[18] Z. Rosberg and M. Zukerman, “Rate Allocation for a Multi-service Internet,” Proc. of Globecom’07, Washington DC., USA, pp. 2678-2683, Nov. 2007.

[19] S. Shakkottai and R. Srikant, “How Good are Deterministic Fluid Models of Internet Congestion Control?,” in Proc. IEEE INFOCOM, 2002, pp. 497–505, 2002.

[20] R. Srikant, *The Mathematics of Internet Congestion Control*, Birkhauser, Boston, 2003.

[21] S. Stidham, “Pricing and Congestion Management in a Network With Heterogeneous Users,” *IEEE Trans. Auto. Contr.*, vol. 49, no. 6, pp. 976-981, Jun. 2004.

[22] A. Tang., L. L. H. Andrew, K. Jacobsson, K. H. Johansson, S. H. Low and H. Hjalmarsson, “Window Flow Control: Macroscopic Properties from Microscopic Factors,” The Proceedings of IEEE INFOCOM 2008, Pheonix, AZ, April 2008.

[23] J. Wang, A. Tang and S. H. Low, “Local stability of FAST TCP,” in Proc. IEEE Conf. Decision and Control, pp. 1023-1028, Dec. 2004.

[24] D. X. Wei, C. Jin and S. H. Low, “FAST TCP: Motivation, architecture, algorithms, performance,” *IEEE/ACM TON*, vol. 14, no. 6, pp. 1246-1259, Dec. 2006.

[25] J. T. Wen and M. Arcak, “A Unifying Passivity Framework for Network Flow Control,” *Trans. Auto. Contr.*, vol. 49, no. 2, pp. 162–174, Feb 2004.

[26] L. Xu, K. Harfoush and I. Rhee, “Binary increase congestion control (BIC) for fast long-distance networks,” in Proc. IEEE INFOCOM 2004, pp. 2514-2524, 2004.

[27] L. Ying, G. E. Dullerud and R. Srikant, “Global Stability of Internet Congestion Controllers With Heterogeneous Delays,” *IEEE/ACM TON*, vol. 14, no. 3, pp. 579-591, Jun. 2006.