Proceedings of the 45th IEEE Conference on Decision & Control
Manchester Grand Hyatt Hotel
San Diego, CA, USA, December 13-15, 2006

Fairness and Convergence Results for *Additive-Increase Multiplicative-Decrease* Multiple-Bottleneck Networks

Richard H. Middleton, Christopher M. Kellett, and Robert N. Shorten

***Abstract*— We examine the behavior of the Additive-Increase Multiplicative-Decrease (AIMD) congestion control algorithm. We present a variant of a recently proposed matrix model that allows us to obtain previous results for competition via a single bottleneck link. We then extend these results to the case of multiple bottleneck links paying particular attention to some aspects of fairness and convergence properties for multiple bottleneck systems. We examine both the synchronous (deterministic) and asynchronous (stochastic) cases. A simple simulation example illustrates the results.**

I. Introduction

Traffic generated by the *Transmission Control Protocol* (TCP) accounts for 85% to 95% of all traffic in today's Internet [1]. TCP, in congestion avoidance mode, is based primarily on Chiu and Jain's [2] *Additive-Increase Multiplicative-Decrease* (AIMD) paradigm for decentralized allocation of a shared resource (e.g., bandwidth) among competing users. The AIMD paradigm is based upon a network of users who independently compete for the available resource by using two basic strategies; each user probes for its share of the available resource by gradually utilizing more and more of the resource (the additive increase stage), and then instantaneously down-scales its utilization-rate in a multiplicative fashion when notified that capacity has been reached (the multiplicative decrease stage). With some minor modifications, the AIMD algorithm has served the networking community well over the past two decades and it continues to provide the basic building block upon which today's internet communication is built.

The dynamics of communication networks in which the AIMD algorithm is deployed have been studied extensively from an empirical viewpoint in the networking and computer science community, and more recently from a mathematical perspective in the mathematics literature; for example, see [3], [4], [5], [6], [7], [8], [9], [10] and references therein. In these papers, some fundamental properties of networks that utilize the AIMD algorithm have been established. For networks where the resource constraint is a bound on the sum of the resource shares of the users, basic stability and convergence properties have been determined, both in a deterministic and in a stochastic setting. In particular, it has been shown that (with a fixed number of users) such networks possess unique stable equilibria to which the system converges geometrically from all starting points.

However, the original AIMD algorithm, as proposed by Chiu and Jain, was based upon a number of assumptions that are generally not valid in real network scenarios. In particular, these include the notion that all sources compete for bandwidth in a single bottleneck scenario. Recently, a number of authors have reported that in such circumstances, AIMD dynamics can lead to network oscillations. In this context our primary interest here is to derive results that describe the behavior of AIMD networks in a quantifiable manner in multiple-bottleneck networks. We note that a number of network models have already been proposed in the literature that purport to capture the essential dynamics of AIMD networks. Roughly speaking, two modeling approaches can be discerned; (i) models based upon fluid approximations of network behavior [8], [11]; and (ii) linear algebraic models that take into account the multi-modal behavior of AIMD networks [9], [12]. While both of these approaches successfully accommodate single bottleneck scenarios, extensions to networks with multiple congested routers have not been straightforward. In particular, the linear algebraic models proposed in the literature have failed to deliver results in multiple-bottleneck scenarios. The main contribution of this note is to present a variant of a recently proposed matrix model that allows us to derive results which predict a degree of fairness in resource allocation between flows that compete directly with each other; even in the presence of network oscillations.

II. Mathematical Framework

The general problem setup that we consider follows that of [9]. Throughout this paper, we consider a set of n_s AIMD sources, flowing through a network with multiple bottlenecks. We assume that each AIMD source has (effectively) an infinite amount of data, and therefore will always alternate between an additive increase, followed by the source detecting congestion, resulting in a multiplicative decrease phase. Let $\mathbb{Z}_{\geq 0}$ denote the nonnegative integers. We denote the (ordered) set of times at which congestion occurs at any node by t_k, $k \in \mathbb{Z}_{\geq 0}$. We assume that the sequence t_k does not contain any accumulation points[1]. We shall return to this assumption later to give conditions sufficient to guarantee that this is the case.

We denote by $x_i(k) \in \mathbb{R}^+$ the flow rate of the i^{th} source at the k^{th} congestion event. Denote the additive increase

R.H. Middleton is with the ARC Centre for Complex Dynamic Systems and Control, The University of Newcastle, Callaghan NSW, 2308, Australia. E-mail: Rick.Middleton@newcastle.edu.au.

C. M. Kellett and R. N. Shorten are with the Hamilton Institute, National University of Ireland, Maynooth, Co. Kildare, Ireland, and are supported by Science Foundation Ireland Grant 04/IN3/I460. E-mails: chris.kellett@nuim.ie and robert.shorten@nuim.ie.

[1] In other words, we wish to rule out the possibility that an infinite number of congestion events occur in an arbitrarily short time.

and multiplicative decrease parameters of the i^{th} source by $\alpha_i \in (0, \infty)$ and $\beta_i \in (0, 1)$ respectively. We assume that the additive increase parameter is small enough compared to the total number of packets in the network that the additive increase phase is effectively continuous. We ignore network and queueing delays in this model framework.

We assume that the network consists of n_n nodes, labeled $j = 1, 2, \ldots, n_n$, and that each flow originates with a source, $i \in \{1, 2, \ldots, n_s\}$ and passes through a set of nodes $N_i \subset \{1, 2, \ldots, n_n\}$. Also, denote by $\Omega_j \subset \{1, 2, \ldots, n_s\}$ the set of flows that pass through node j. We assume that each node has a total capacity, $B_j > 0$, and that the node capacity constraints can be expressed as:

$$B_j \geq \sum_{i \in \Omega_j} x_i(k); \forall j \in \{1, 2, \ldots, n_n\},\ k \in \mathbb{Z}_{\geq 0}. \quad (1)$$

Suppose that (1) is satisfied at the k^{th} congestion event. Then for all times after the previous congestion event, leading up to the current event (that is for $t \in (t_{k-1}, t_k)$), all flows will be in their additive increase phase. It therefore follows that their maximum over this time interval is at t_k, and therefore if the constraints are satisfied at all congestion events, then they must also be satisfied at all intervening times.

To simplify the notation, we denote the stacked vector of source flows by $X^T(k) := \begin{bmatrix} x_1(k) & x_2(k) & \cdots & x_{n_s}(k) \end{bmatrix}$ then note that the constraints (1) can be expressed in vector form as:

$$B_j \geq L_j^T X(k); \forall j \in \{1, 2, \ldots, n_n\}, k \in \mathbb{Z}_{\geq 0} \quad (2)$$

where L_j is a vector with i^{th} element unity if the i^{th} flow includes node j, and zero otherwise; that is, $(L_j)_i = \mathbb{I}_{j \in N_i}$. We make the following assumption on the flows:

Assumption 1: All flows include at least one node. That is:

$$N_i \neq \emptyset : \forall i \in \{1, 2, \ldots, n_s\}. \quad (3)$$

Note that Assumption 1 implies that the constraints (2) form a compact set. More specifically, under Assumption 1 there exists[2] an $X_{max} \in (0, \infty)$ such that for all $X(k)$ in the positive orthant satisfying (2), we have:

$$\|X(k)\|_2 \leq X_{max}. \quad (4)$$

At the k^{th} congestion event, we assume that at least one node is congested. We denote by $J(k) \subset \{1, 2, \ldots, n_n\}$ the set of nodes congested at t_k, that is; $J(k) = \{j : B_j = L_j^T X(k)\}$.

A. Synchronous Traffic

In the synchronous traffic case, we assume that all flows that include a congested node experience congestion. In practice, this would mean that all flows through a congested link would have a packet dropped. This may be a reasonable assumption in some cases such as: where the flows are at least somewhat fair (that is, no flows experience a much smaller share of the available bandwidth than the average); or, the total number of packets dropped by routers at each congestion event is large compared to the number of flows through the router (for example, if the product of the packet data rate and round trip time is large compared to the number of flows).

Note that in this case, using our model framework, it is straightforward to show that a congestion event at time t_k in node j causes a drop in flow in node j from $B_j = \sum_{i \in \Omega_j} x_i(k)$ to $\sum_{i \in \Omega_j} \beta_i x_i(k)$. Thus the total decrease in flow in node j is $\sum_{i \in \Omega_j} (1 - \beta_i) x_i(k) \geq \left(1 - \max_{i \in \Omega_j} \{\beta_i\}\right) B_j$. Since the rate of increase of flow through node j is at most[3] $\sum_{i \in \Omega_j} \alpha_i$, it follows that the minimum time between congestion events for node j is at least $B_j \left(\frac{1 - \max_{i \in \Omega_j} \{\beta_i\}}{\sum_{i \in \Omega_j} \{\alpha_i\}} \right)$. Furthermore, since there are a finite number of nodes (each of which has a nontrivial lower bound on the time between successive congestion events), there cannot be any accumulation points in congestion times.

For the case of synchronous traffic, the model we adopt for the recursion is given by:

$$X(k+1) = A_{j(k)} X(k) + UT(k) \quad (5)$$

where U is a stacked vector of the additive increase parameters $U = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_{n_s} \end{bmatrix}^T$; $T(k) = t_{k+1} - t_k$ is the time between congestion events given by:

$$T(k) = \max_{T : B_m \geq L_m \left(A_{j(k)} X(k) + UT\right); m = 1, 2, \ldots n_n} \{T\}; \quad (6)$$

and A_j denotes a diagonal matrix with i^{th} element given by:

$$(A_j)_{ii} = \left\{ \begin{array}{c} \beta_i : i \in \Omega_j \\ 1 : otherwise \end{array} \right\}. \quad (7)$$

In other words, when node j experiences congestion, all flows which transit node j (i.e., all flows in Ω_j) reduce their flow rate as $\beta_i x_i(k)$, while flows not crossing node j are unaffected and continue to increase their rates.

Note that in this case, it is straightforward to show that the time between congestion events is bounded. In particular, if we let $\underline{\alpha} = min_i\{\alpha_i\}$ and take any $\bar{X} > X_{max}$, then using (5), $T(k) > \bar{X}/\underline{\alpha}$ implies $\|X(k+1)\| > \bar{X}$ which contradicts (4).

B. Asynchronous Flows

In this case, we no longer assume that when a node is congested, that all flows experience congestion (i.e., not all flows are notified that the node is congested). Rather, at random, one or more flows will experience congestion. In this case the model of (5) becomes more complex, since the appropriate A matrix is no longer a deterministic function of the congested node, $j(k)$. For simplicity, we follow the model framework of [9] wherein the probabilities associated with whether or not source i experiences congestion is independent[4] of other sources and is given by λ_i. More

[2] For example, it suffices to take $X_{max} = \sqrt{n_s} \max_j \{B_j\}$.

[3] Since a congestion event at a node other than node j may cause some flows through node j to decrease.

[4] Here the independence is both serially in time and between different flows.

complex models that take into account the dependence of the probability of detecting congestion on the relative fraction of the total flow can be developed. A detailed analysis of such models is substantially more complex due to the nonlinear nature of the models. However, in the case of independent drop probabilities, the equivalent model to (5) becomes:

$$X(k+1) = A_k X(k) + UT(k) \tag{8}$$

where A_k is a diagonal random matrix with elements given by

$$A_k(ii) = \left\{ \begin{array}{ll} \beta_i & \text{w.p. } \lambda_i \text{ for } i \in \Omega_{j(k)} \\ 1 & otherwise \end{array} \right\}.$$

Under the assumption above on independence of the probabilities in the elements of A_k, from (8) we can show that the expected value of $X(k)$ (denoted $\mathcal{E}\{X(k)\}$) satisfies the recursion:

$$\mathcal{E}\{X(k+1)\} = A'_k \mathcal{E}\{X(k)\} + U\mathcal{E}\{T(k)\} \tag{9}$$

where A'_k is a constant diagonal matrix with elements $\beta'_i = 1 - \lambda_i + \lambda_i \beta_i$ for $i \in \Omega_{j(k)}$.

III. Single Bottleneck Analysis

Before addressing the multiple bottleneck case, we first review some existing results (see for example [9]) on the single bottleneck case. If there is a single bottleneck, it must restrict all flows, that is, L is a vector of all unity elements. Otherwise, any flows not restricted would grow without bound.

A. Synchronous Single Bottleneck

For synchronous flows with a single bottleneck, the model (5) simplifies to:

$$X(k+1) = AX(k) + UT(k) \tag{10}$$

where A is diagonal with i^{th} element $A_{ii} = \beta_i$. $T(k)$ can be computed as the duration required to achieve the capacity constraint $L^T X(k+1) = B$, that is, $T(k) := \frac{B - L^T AX(k)}{L^T U}$. Define[5]

$$T^* := \frac{B}{L^T(I-A)^{-1}U} \quad \text{and} \quad X^* := (I-A)^{-1}UT^*.$$

Further, define error coordinates $E(k) := X(k) - X^*$ and $\Delta(k) := T(k) - T^*$. Then we can re-write (10) as:

$$E(k+1) = AE(k) + U\Delta(k) \tag{11}$$

Now $\Delta(k) = -\frac{L^T A}{L^T U}E(k)$ and therefore, we can re-write (11) as:

$$E(k+1) = \left(A - \frac{UL^T A}{L^T U}\right)E(k) \tag{12}$$

We now perform a diagonal state transformation, with $D = diag\{\sqrt{\alpha_i/\beta_i}\}$. Then if we define $V^T := \left[\begin{array}{cccc} \sqrt{\alpha_1} & \sqrt{\alpha_2} & \ldots & \sqrt{\alpha_n} \end{array}\right]$, it can be shown that we have $D^{-1}U = A^{\frac{1}{2}}V$; $L^T AD = V^T A^{\frac{1}{2}}$ and $L^T U = V^T V$. Therefore, with $F(k) := DE(k)$ we have:

$$F(k+1) = A^{\frac{1}{2}}\left(I - \frac{VV^T}{V^T V}\right)A^{\frac{1}{2}}F(k) \tag{13}$$

It then follows from (13) that[6]

$$\|F(k+1)\| \leq \|A\|\|F(k)\| = max\{\beta_i\}\|F(k)\|.$$

As $\beta_i \in (0,1)$ for all i, this implies that $F(k)$ converges exponentially to the origin. Consequently, $X(k)$ converges to X^*.

B. Asynchronous Single Bottleneck

Similarly, if we consider the asynchronous flows case, then A'_k in (9) becomes the constant matrix $A' = diag\{\beta'_i\}$, and (9) becomes:

$$\mathcal{E}\{X(k+1)\} = A'\mathcal{E}\{X(k)\} + U\mathcal{E}\{T(k)\}. \tag{14}$$

Then by direct extension of the analysis in Section III-A (or from [9]), if we define

$$T^{*\prime} = \frac{B}{L^T(I-A')^{-1}U} \quad \text{and} \quad X^{*\prime} = (I-A')^{-1}UT^{*\prime},$$

then $\mathcal{E}\{T(k)\}$ converges exponentially fast to $T^{*\prime}$ and $\mathcal{E}\{X(k)\}$ converges exponentially fast to $X^{*\prime}$.

IV. Parallel Flows

In this section, we wish to consider aspects of the behavior of more general network configurations. One particular aspect of more general, multiple bottleneck networks, is to consider the behavior of 'parallel' flows. Here we define parallel flows as flows that pass through an identical set of congested nodes. In other words, flows $i_1, i_2, \ldots, i_p$ are parallel if and only if:

$$N_{i_1} = N_{i_2} = \cdots = N_{i_p} =: N_p \tag{15}$$

Recall that N_i is the set of nodes through which flow i passes.

We would argue that some measure of fairness between parallel flows is necessary (though not sufficient) for overall network 'fairness'. In particular, users might at least expect relative fairness with other users who share identical routes, even if it is not possible to simply quantify or ensure 'fairness' in relation to users having different paths and destinations.

A. Synchronous Parallel Flows: Time Averages

Consider first the simple case where we have synchronous flows modeled by (5) and (6), where some of the flows $i_1, i_2, \ldots, i_p$ are parallel in the sense of (15). In this case, define a selection vector $E_p \in \mathbb{R}^{p \times n_s}$ as:

$$E_p = \left[\begin{array}{c} e_{i_1}^T \\ e_{i_2}^T \\ \vdots \\ e_{i_p}^T \end{array}\right] \tag{16}$$

[5] X^* here is closely related with the Perron eigenvector given in [13, Theorem 2.1].

[6] This is slightly weaker than is possible, but suffices for the main properties we wish to show.

where e_i denotes the i^{th} elementary vector. Define $X_p(k) = E_p X(k)$ as the subflows at the k^{th} congestion event. We now consider the evolution of $X_p(k)$. For this subsection, without loss of generality, we assume that for all k, congestion occurs on one of the nodes in the parallel flow[7]. In this case, using (5) and (16), the recursion for the subflows becomes:

$$X_p(k+1) = A_p X_p(k) + U_p T(k) \tag{17}$$

where $U_p := E_p U$ and $A_p := E_p A_{i_p} E_p^T$ are independent of which node within N_p is congested.

Note that we cannot apply the same analysis as in the previous section since, due to the presence of flows outside the parallel grouping, we cannot calculate the time between congestion events. However, we can make the following claim:

Claim 2: Consider any set of parallel flows. Take any $U_p^\perp$ orthogonal to U_p; that is, $U_p^\perp U_p = 0$. Suppose that either of the following conditions hold:

$$\beta_{i_1} = \beta_{i_2} = \cdots = \beta_{i_p} =: \beta_p \tag{18}$$

or

$$\lim_{k\to\infty} T(k) = T_\infty \tag{19}$$

then

$$\lim_{k\to\infty} \left(U_p^\perp (I - A_p) X_p(k) \right) = 0. \tag{20}$$

Proof: Note that in either case, from (17) that:

$$X_p(k) = A_p^k X_p(0) + \sum_{\ell=0}^{k-1} A_p^{k-1-\ell} U_p T(\ell). \tag{21}$$

The first term in (21) decays exponentially fast to zero, so it remains to evaluate properties of the remaining term.

First, suppose that (18) holds. Then it follows that $A_p = \beta_p I$, $A_p^\ell = \beta_p^\ell I$, and $U_p^\perp (I - A_p)^{-1} = \frac{1}{(1-\beta_p)} U_p^\perp$. Using these facts along with (21) and ignoring initial conditions gives:

$$\begin{aligned} \left(U_p^\perp (I - A_p) X_p(k) \right) &= (1-\beta_p) U_p^\perp X_p(k) \\ &= (1-\beta_p) U_p^\perp U_p \sum_{\ell=0}^{k-1} \beta_p^{k-1-\ell} T(\ell) \\ &= 0 \end{aligned} \tag{22}$$

Alternatively, suppose that (18) is not assumed to hold, but that instead (19) holds. In this case we note first that as $k \to \infty$, again ignoring initial conditions:

$$\begin{aligned} X_p(k) &= \sum_{\ell=0}^{k-1} A_p^{k-1-\ell} U_p T(\ell) \\ &= \sum_{\ell=0}^{k-1} A_p^{k-1-\ell} U_p T_\infty + \sum_{\ell=0}^{k-1} A_p^{k-1-\ell} U_p (T(\ell) - T_\infty) \\ &\to (I - A_p)^{-1} U_p T_\infty \end{aligned}$$

[7] Note that if this is not the case, then none of the subflows are congested, and as far as the subflows are concerned, we simply ignore this congestion event. This cannot happen indefinitely since otherwise the subflows would become unbounded.

and the result follows immediately. ■

In other words, under the conditions stated in Claim 2, the states converge to a one dimensional subspace aligned with $(I - A_p)^{-1} U_p$ (the Perron eigenvector in [9, Theorem 2.1]) which has i^{th} element $(1-\beta_i)^{-1}\alpha_i$. Note however, that this does not apply in general (i.e., in the absence of (18) or (19)). However, the following is true:

Claim 3: Consider any set of parallel synchronized flows, (17), subject to Assumption 1, then

$$\lim_{K\to\infty} \left(\frac{1}{K} \sum_{k=k_0}^{k_0+K} U_p^\perp (I - A_p) X_p(k) \right) = 0. \tag{23}$$

Proof: First, take summations of (17) to give:

$$\frac{1}{K} \sum_{k=k_0}^{k_0+K} X_p(k+1) = A_p \frac{1}{K} \sum_{k=k_0}^{k_0+K} X_p(k) + U_p \frac{1}{K} \sum_{k=k_0}^{k_0+K} T(k). \tag{24}$$

Then we can rearrange (24) to the form:

$$(I - A_p) \frac{1}{K} \sum_{k=k_0}^{k_0+K} X_p(k) = \frac{1}{K} \left(X_p(k_0) - X_p(k_0+K+1) \right) + U_p \frac{1}{K} \sum_{k=k_0}^{k_0+K} T(k). \tag{25}$$

Multiplying (25) from the left by $U_p^\perp$ and taking the limit as $K \to \infty$ gives the desired result in view of the boundedness of $X_p(k)$. ■

Remark 4: By operating in terms of expected values, the extension to stochastic networks defined in Section II-B is immediate. We defer detailed discussion of this to Section IV-C after we have given some interpretations of Claim 3.

We first note that in the case where the states do converge, Claim 3 immediately gives the following corollary.

Corollary 5: Consider any set of parallel synchronized flows, (17), subject to Assumption 1 and suppose that $X_p(k)$ converges to a limit denoted by $X_p(\infty)$. Then

$$U_p^\perp (I - A_p)^{-1} X_p(\infty) = 0. \tag{26}$$

Remark 6: Since Claim 3 holds for any set of synchronized flows, including any pair of flows, it represents a kind of average inter-flow fairness. The time average of the peak flows represented in $X_p(k)$ lies on a given ray from the origin. Moreover, for any flows ℓ and m that are parallel, take $U_p^\perp$ as a vector with all elements zero, except the ℓ^{th} element $1/\alpha_\ell$ and the m^{th} element $-1/\alpha_m$. We then have the long term time average (which we denote with an overbar; i.e., $\overline{f_k} := \lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K} f_k$):

$$\overline{\left(\frac{1-\beta_\ell}{\alpha_\ell} \right) (X(k))_\ell - \left(\frac{1-\beta_m}{\alpha_m} \right) (X(k))_m} = 0 \tag{27}$$

and therefore, provided the appropriate time averages exist, (27) implies

$$\left(\frac{1-\beta_\ell}{\alpha_\ell} \right) \overline{(X(k))_\ell} = \left(\frac{1-\beta_m}{\alpha_m} \right) \overline{(X(k))_m}. \tag{28}$$

As an aside, we note that the basic result of Claim 3 can be generalized to models that allow for a nonlinear decrease but maintain an additive increase as follows:

Corollary 7: Consider an Additive Increase, Nonlinear Decrease (AIND) process with parallel flows that obey the recursion:

$$X_p(k+1) = f(X_p(k)) + U_p T(k) \tag{29}$$

where $X_p(k)$ satisfies (4). Then for any $U_p^{\perp}$ orthogonal to U_p:

$$\lim_{K\to\infty}\left(\frac{1}{K}\sum_{k=k_0}^{k_0+K} U_p^{\perp}\left(X_p(k) - f(X_p(k))\right)\right) = 0. \tag{30}$$

Proof: (29) can be rewritten as:

$$X_p(k) - f(X_p(k)) = X_p(k) - X_p(k+1) + U_p T(k). \tag{31}$$

Summing (31) gives:

$$\sum_{k=k_0}^{k_0+K}\left(X_p(k) - f(X_p(k))\right) =$$

$$X_p(k_0) - X_p(k_0+K+1) + U_p \sum_{k=k_0}^{k_0+K} T(k). \tag{32}$$

Multiplying (32) from the left by $U_p^{\perp}$, dividing by K, and taking limits gives the desired result. ■

B. Synchronous Parallel Flows: Ensemble Averages

Because the rest of the network can influence the detailed behavior of a set of parallel flows, even in the synchronous case, it is not possible to guarantee that the parallel flows converge. The results in Section IV-A give time average results that apply in this case. Here we give some results for ensemble averages for the synchronous parallel flow case.

To facilitate this analysis, we note that in the previous approach, the capacity constraint (1) can be thought of as a "router view" of congestion. An alternate approach is to consider what bandwidth constraint a group of parallel flows will see at congestion. We observe that this bandwidth will vary depending on both which node is congested as well as how much capacity is being used by other flow groups. As such, the capacity constraint seen by any individual flow group will be random and time-varying. Using $\mathcal{I}_p$ to denote the p^{th} parallel flow group, we can write the capacity constraint at congestion as

$$B_p(k) = \sum_{i\in\mathcal{I}_p} x_i(k);\ \forall k \in \mathbb{Z}_{\geq 0}. \tag{33}$$

Note that $B_p(k)$ is necessarily bounded by the minimum capacity link traversed by the flow group $\mathcal{I}_p$.

We make the following assumption:

Assumption 8: The process $B_p(k)$ is a stationary random process; i.e., there exists a finite real number $\bar{B}_p > 0$ such that $\mathcal{E}\{B_p(k)\} = \bar{B}_p$.

Using the previous vector notation, at congestion we have

$$L_p^T X_p(k+1) = B_p(k+1), \tag{34}$$

where L_p is a vector of dimension $|\mathcal{I}_p|$ consisting of all ones. Taking expectations on both sides, and using the evolution equation (17) we obtain

$$L_p^T\left(A_p\mathcal{E}\{X_p(k)\} + U_p\mathcal{E}\{T(k)\}\right) = \bar{B}_p. \tag{35}$$

Similar to the analysis in the single bottleneck case (Section III), we can compute the expected time between congestion events as $\mathcal{E}\{T(k)\} = \bar{T}^* = \frac{\bar{B}_p}{L_p^T(I-A_p)^{-1}U_p}$ and the expected flow rate $\mathcal{E}\{X_p(k)\}$ converges exponentially (via the same arguments in Section III-A) to:

$$\bar{X}_p^* = (I - A_p)^{-1}U_p\bar{T}^*. \tag{36}$$

Remark 9: It is important to note here that we not only characterize the asymptote but also the dynamics of the process. Convergence to the equilibrium state is exponential and bounds on the rate of convergence can be derived. It is also important to note that the dynamics of the second moment can be expressed in a similar manner to the above analysis.

From a practical viewpoint, it may well be that we do not know the expected value of the bandwidth; i.e., we will not know $\bar{B}_p$. However, the above analysis does indicate how parallel flows will share available bandwidth within the parallel group. For example, if all flows in the group have the same increase and decrease parameters, the (unknown) bandwidth will be shared equally on average.

C. Asynchronous Parallel Flows: Ensemble Averages

We now consider the more general model framework of Section II-B, that allows randomness in determining which flows experience lost packets at a congestion event. In this case, by the same arguments as in Claim 3, applied to (9), we obtain:

$$\lim_{K\to\infty}\frac{1}{K}\sum_{k=k_0}^{k_0+K} U_p^{\perp}(I - A_p)\mathcal{E}\left\{X_p(k)\right\} = 0. \tag{37}$$

Furthermore, if the process is ergodic, then (37) simplifies to:

$$U_p^{\perp}(I - A_p)\mathcal{E}\left\{X_p(k)\right\} = 0. \tag{38}$$

V. Examples

Example 1: Consider the network topology depicted in Figure 1. Here, $C_1 = 2.5$ units and $C_2 = 5$ units. This arrangement gives rise to a constraint surface that is polyhedral in nature. Drops at congestion are generated so that all nodes contributing to congestion are informed of congestion; namely, x_1 and x_2 are informed every time a node is congested, whereas x_3 is informed only when node N_2 is congested. The network was simulated for 2000 seconds.

Here $\alpha_1 = \alpha_2 = \alpha_3 = 1$ and $\beta_1 = 0.5$, $\beta_2 = 0.75$, and $\beta_3 = 0.9$. It follows from Claim 3 that $\bar{X}(k)_1 = 2\ \bar{X}(k)_2$. This is immediately confirmed from the simulation results depicted in Figure 2.

Example 2: Consider again the network topology depicted in Figure 1. Drops at congestion are generated at each node

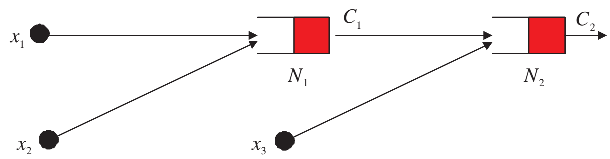

Fig. 1. Two node, thee flow scenario

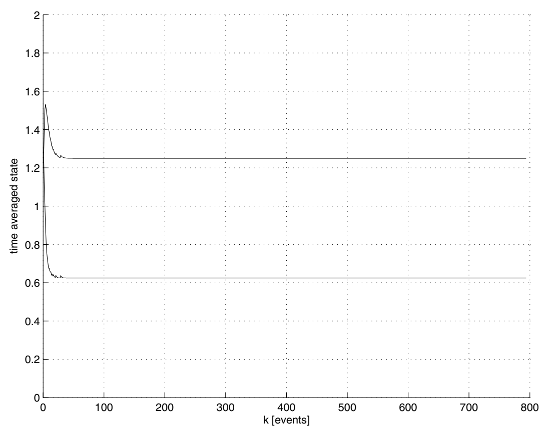

Fig. 2. Predictions of Claim 3: Synchronized network

according to fixed probabilities (uniform for every source utilizing the constraint). Again, the flows $x_1(k)$ and $x_2(k)$ are parallel flows in this scenario. In this simulation we observe $\lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K} X_i(k)$, $i \in 1, 2$ over the first 1600 congestion events.

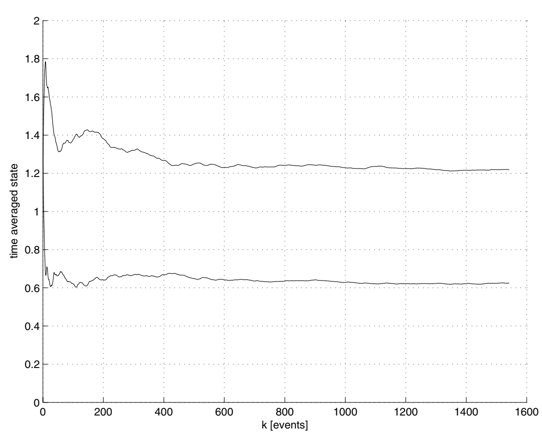

Fig. 3. Predictions of Claim 3: Stochastic network

We see from Figure 3 that

$$\lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K} X_1(k) \approx 2\left(\lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K} X_2(k)\right),$$

which is in complete agreement with Claim 3 and the associated remarks.

VI. Conclusions

In this paper we have extended work on the analysis of some AIMD multiple bottleneck systems, based on fairness and convergence analysis such as in [9] for the single bottleneck case. For the multiple bottleneck case, it is known that convergence does not hold in general. However, by introducing the concept of parallel flows for flows that experience an identical set of bottleneck nodes, we are able to establish results for *average* fairness amongst parallel flows. These results imply directly that when the flows do converge, parallel flows must satisfy a form of fairness directly analogous to those of [9]. The results have been pursued in both the synchronous and asynchronous cases, and are illustrated by a simple simulation study.

References

[1] Z. Hhoa, S. Darbha, and A. Reddy, "A method for estimating the proportion of nonresponsive traffic at a router", *IEEE/ACM Transactions on Networking*, v. 12, pp. 708-718, 2004.

[2] D. Chiu and R. Jain, "Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks", *Journal of Computer Networks* v. 17, pp. 1-14, 1989.

[3] A. Berman, R.N. Shorten, and D.J. Leith, "Positive matrices associated with synchronised communication networks", *Linear Algebra and its Applications*, v. 393, pp. 47-54, 2004.

[4] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability", *Journal of the Operational Research Society*, v. 49, pp. 237-252, 1998.

[5] J. Hespanha, S. Bohacek, K. Obrarzka, and J. Lee, "Hybrid modeling of TCP congestion control", *Hybrid Systems: Computation and Control*, pp. 291-304, 2001.

[6] A. Leizarowitz, R. Stanojevic and R.N. Shorten, "Towards an analysis and design framework for communication networks with Markovian dynamics", *IEE Proceedings on Control Theory*, Accepted for publication, 2006.

[7] U.G. Rothblum and R.N. Shorten, "Convergence results for high-speed TCP", Submitted to *SIAM Journal of Control and Optimization*, 2005.

[8] R. Srikant, "*The Mathematics of Internet Congestion Control*", Birkhauser, 2003.

[9] R.N. Shorten, F. Wirth and D.J. Leith, "A Positive Systems Model of TCP-like congestion control: asymptotic results", *IEEE/ACM Transactions on Networking*, v. 14, pp. 616-629, 2006

[10] F. Wirth, R. Stanojevic, R.N. Shorten, and D.J. Leith, "Stochastic equilibria of AIMD communication networks", *SIAM Journal of Matrix Analysis*, Accepted for publication, 2006.

[11] S.H. Low, F. Paganini, and J.C. Doyle, "Internet Congestion Control", *IEEE Control Systems Magazine*, pp. 28-43, February 2002

[12] F. Baccelli and D. Hong, "Interaction of TCP flows as billiards", in *Proceedings of INFOCOM*, 2003.

[13] R.N. Shorten, D.J. Leith, J. Foy, and R. Kilduff, "Analysis and design of AIMD congestion control algorithms in communication networks", *Automatica*, v. 41, pp. 725–730, 2005.