Washington University in St. Louis

Washington University Open Scholarship

All Computer Science and Engineering Research

Computer Science and Engineering

Report Number: WUCS-90-06

1992-12-01

The Effect of Delayed Feedback Information on Network Performance

Andreas D. Bovopoulos and Aurel A. Lazar

The performance of a network subject to either state dependent or state independent flow control is investigated. In the state dependent case, the flow control policy is a function of the total number of packets for which the controller has not yet received an acknowledgement. In this case it is shown that the optimal flow control is a sliding window mechanism. The effect of the delayed feedback on the network performance as well as the size of the window are studied. The state independent optimal rate is also derived. The performance of the state dependent and state independent flow... Read complete abstract on page 2.

Follow this and additional works at: https://openscholarship.wustl.edu/cse_research

Recommended Citation

Department of Computer Science & Engineering - Washington University in St. Louis
Campus Box 1045 - St. Louis, MO - 63130 - ph: (314) 935-6160.

The Effect of Delayed Feedback Information on Network Performance

Andreas D. Bovopoulos and Aurel A. Lazar

Complete Abstract:

The performance of a network subject to either state dependent or state independent flow control is investigated. In the state dependent case, the flow control policy is a function of the total number of packets for which the controller has not yet received an acknowledgement. In this case it is shown that the optimal flow control is a sliding window mechanism. The effect of the delayed feedback on the network performance as well as the size of the window are studied. The state independent optimal rate is also derived. The performance of the state dependent and state independent flow control policies are compared. Conditions for employing one of the two types of flow control policies for superior end-to-end network performance are discussed. All the results obtained are demonstrated using simple examples.

The Effect of Delayed Feedback Information on
Network Performance

Andreas D. Bovopoulos and Aurel A. Lazar

WUCS-90-06

December 1992 (revised)

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
St. Louis MO 63130-4899

The effect of delayed feedback information on network performance

Andreas D. Bovopoulos

Department of Computer Science and Computer and Communications Research Center, Washington University, St. Louis, MO 63130-4899, USA

Aurel A. Lazar

Department of Electrical Engineering and Center for Telecommunications Research, Columbia University, New York, NY 10027-6699, USA

The performance of a network subject to either state dependent or state independent flow control is investigated. In the state dependent case, the flow control policy is a function of the total number of packets for which the controller has not yet received an acknowledgment. In this case it is shown that the optimal flow control is a sliding window mechanism. The effect of the delayed feedback on the network performance as well as the size of the window are studied. The state independent optimal rate is also derived. The performance of the state dependent and state independent flow control policies are compared. Conditions for employing one of the two types of flow control policies for superior end-to-end network performance are discussed. All the results obtained are demonstrated using simple examples.

1. Introduction

One of the most challenging problems in the design of communication protocols is the derivation of the most appropriate flow control mechanism. In the X.25 User-Network interface recommendation, and in the OSI layered architecture, the flow control used is the sliding window (bang-bang control). Such a flow control mechanism can be applied either between a data terminal equipment (DTE) and a data circuit-terminating equipment (DCE) or between two DTEs [14].

One of the versions of the sliding window mechanism requires that, for each of the delivered packets, an acknowledgment packet (token) is sent back to its source. The sliding window flow control mechanism is a reliable mechanism for flow control because the receiver (a DCE or a DTE) is in control of the flow of packets in the network.

A different flow control mechanism proposed in the literature is rate flow control (see, e.g., [1]). With this mechanism, the rate with which the source sends packets into the network is controlled. This mechanism has recently received support from researchers in the internetworking community [7,19,20]. If we compare these two different flow control mechanisms from their implementation point of view, we see that the window flow control mechanism can be easily implemented in the form of a sliding window. Reliable rate flow control mechanisms require very large time windows [1].

Probably the most fundamental difference between these two flow control strategies is the amount of information that they are based upon. Window flow control is based on feedback information about the state of the network whereas the rate flow control does not take into account the state of the network. In most of the studies of flow control it was assumed that acknowledgment packets are instantaneously received by the source [2,11]. In reality, the acknowledgment packets travel from destination back to source and as such they are also subject to time delay.

In this paper the performance of a network that operates under the above mentioned flow control strategies is investigated. First, the optimal state dependent flow control with *delayed feedback information* about the network state is derived. Second, the results are compared with the optimal state independent (rate) flow control. In order to make the analysis tractable, the communication resources are modeled as a Jackson network with a forward as well as an acknowledgment network. Furthermore, it is assumed that the maximum rate with which the source sends packets into the network is bounded. The optimal flow control mechanism which maximizes the average throughput under the condition that the expected packet time delay in the forward network does not exceed an upper bound is derived. Related work in the area of state dependent resource allocation is presented in [2–4,6,13].

This paper is organized as follows. In section 2, the formulation of the problem is introduced. In section 3, it is shown that the optimal state dependent flow control with delayed feedback information under the previous optimization criterion is a window flow control mechanism. Furthermore, the effect of the delayed feedback information on the network performance is studied. An example that sheds light on the derived theoretical results is given. In section 4, a number of conclusions are drawn and a number of issues that need further study are discussed.

2. The statement of the problem

A user wishes to optimally utilize the resources of a Jackson network consisting of a forward and an acknowledgment network (see fig. 1).
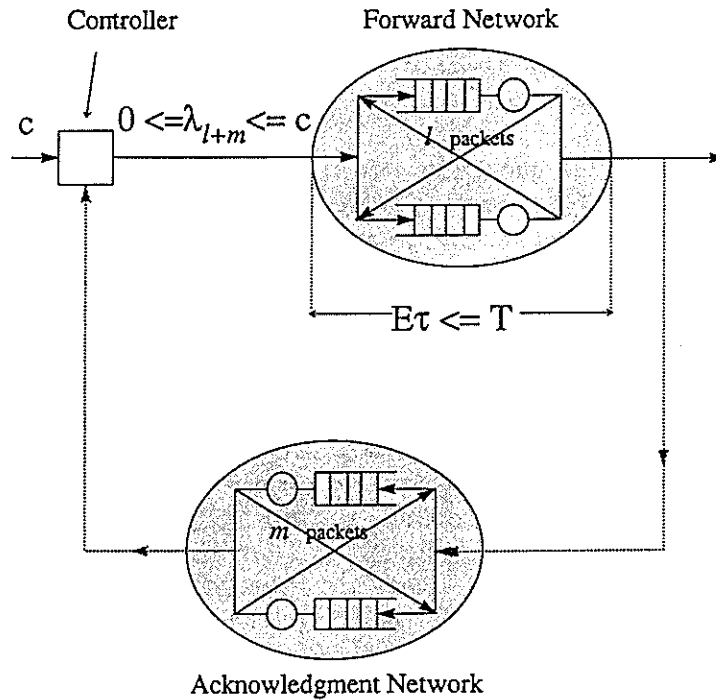
Fig. 1. A Jackson network with acknowledgment delays.

Each of the I processors of the forward network has an infinite number of buffers and serves packets at each node with an exponential service rate. There are J processors in the acknowledgment network. Let μ^i be the service rate of the ith processor, $1 \leqslant i \leqslant M$, with $M = I + J$. Let $\mathbf{R} = [r^{ij}]$ be the $(M + 1) \times (M + 1)$ routing matrix $(0 \leqslant i \leqslant M, 0 \leqslant j \leqslant M)$. Using this notation, packets join the network at node i with probability r^{0i}. Upon completion of service at node i, packets leave the network with probability r^{i0} or are routed from node i to node j with probability r^{ij}. It is assumed that the topology of the network does not change with time *and that, at the time a packet reaches its destination, an acknowledgment packet is sent to the source controller through the acknowledgment network.*

Our goal in this paper is to highlight the effect to transmission delays on network performance. The queueing network defined above models a single source–destination pair. Transmission delays are modeled as lags between the time when a knowledge structure about the system is available at the source and the time the same structure is available at the destination node. Networks with multiple source–destination pairs are not covered here. Such networks are modeled and flow control problems formulated in [1,9,10]. However, these papers do not address the issues associated with acknowledgment delays. It is also assumed here that there is no interference between the forward and the

acknowledgment network. In a communication network, acknowledgment packets do actually interfere with packets that travel forward for a given connection. This interference appears, however, to be negligible. To the best of our knowledge, no experimental data is available to support or refute the latter assumption.

The evolution of the queueing network is described by the stochastic vector $Q_t = (Q_t^1, \cdots, Q_t^{M-1}, Q_t^M)$, where Q_t^i refers to the number of packets at node i, $1 \leqslant i \leqslant M$. Let the partial knowledge about the state of the network be given by $X_t = Q_t^1 + \cdots + Q_t^{M-1} + Q_t^M$, where Q_t^i refers to the total number of packets at node i, $1 \leqslant i \leqslant M$. The user's packets enter the network at rate λ_{X_t}. $\lambda = \lambda_{X_t}$ denotes the *flow control policy* that is enforced by the controller, and it is a function of the number of packets for which the controller has not yet received an acknowledgment. Let c be the maximum rate of the packets entering the network. Any flow control policy must satisfy the peak constraint $0 \leqslant \lambda_{X_t} \leqslant c$, which defines the class of admissible flow control policies. The expected time that the packets spent in the forward network is denoted by $\mathscr{E}\tau$ and it should not exceed a given upper bound T. Let $\mathscr{E}\gamma$ denote the expected number of packets per unit of time (the expected throughput from now on) that the controller sends into the network. The flow control policy which the controller implements is said to be optimum over the class of admissible control policies for a given upper bound of the average time delay of the packets in the network T, if the maximum average throughput is achieved, that is, the policy that achieves the

$$\max_{\mathscr{E}\tau \leqslant T} \mathscr{E}\gamma. \tag{2.1}$$

3. The effect of delayed feedback information on network performance

3.1. OPTIMAL STATE DEPENDENT FLOW CONTROL

In the sequel, the flow control problem of a Jackson network with nonzero acknowledgment delays is studied.

3.1.1. An equivalent formulation of the problem

If at most N packets are permitted to enter the network, the original Jackson network (depicted in fig. 1) is identical to the network depicted in fig. 2, which in turn can be transformed into a first order equivalent Jackson network (shown in fig. 3) by using Norton's theorem [8,17].

$\mathscr{E}\gamma_N$ and $\mathscr{E}\tau_N$ are the expected forward throughput and expected time delay, respectively, given that at any given moment no more than N packets can be in the network.
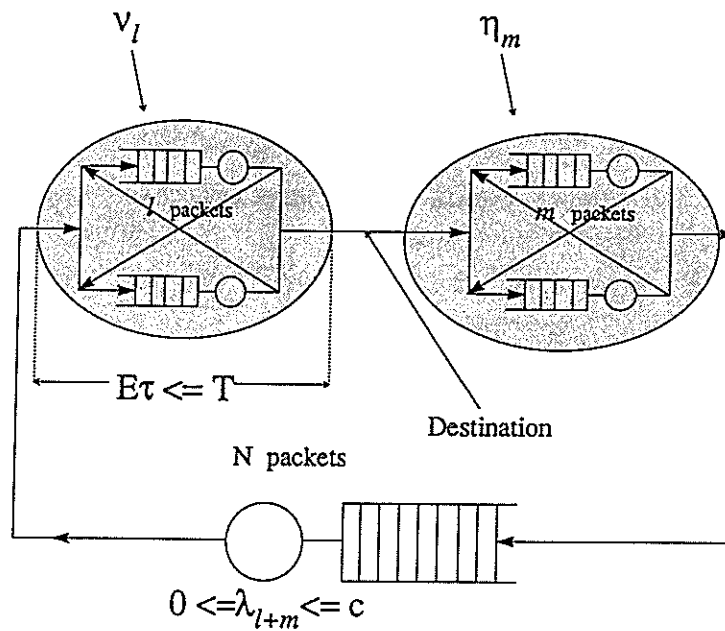
Fig. 2. A Jackson network with acknowledgment delays with at most N packets.

Let the $1 \times (M+1)$ matrix $\Theta \stackrel{\mathrm{def}}{=} [\theta^0 \theta^1 \cdots \theta^M]$ be the solution of the traffic flow equations

$$\Theta = \Theta R,$$

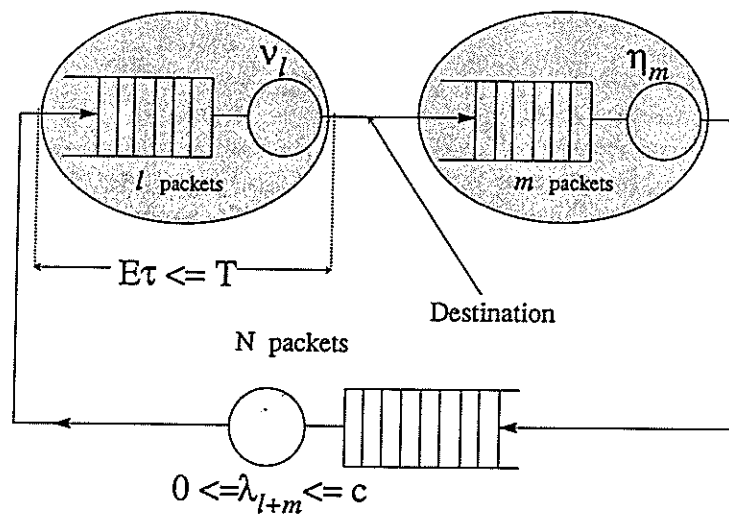Fig. 3. The Jackson network after the aggregation of the forward and the acknowledgment networks.

where $\theta^0 = 1$. Let

$$g_l \stackrel{\text{def}}{=} \sum_{k_1+k_2+\cdots+k_I=l} \prod_{j=1}^{I} \left(\frac{\theta^j}{\mu^j} \right)^{k_j}, \tag{3.1}$$

for all l, $1 \leqslant l \leqslant N$, where $0 \leqslant k_i$, for $i = 1, \cdots, I$.

If l is the total number of packets in processors $1, 2, \cdots, I$, then the Norton equivalent, symbolized by ν_l, is given by

$$\nu_l \stackrel{\text{def}}{=} \frac{g_{l-1}}{g_l}. \tag{3.2}$$

If, in addition, m is the total number of packets in processors $I + 1, I + 2, \cdots, M$, then the Norton equivalent of the processors serving the acknowledgment packets is symbolized by η_m and given by the equation

$$\eta_m = \frac{h_{m-1}}{h_m}, \tag{3.3}$$

where

$$h_m = \sum_{k_{I+1}+\cdots+k_M=m} \prod_{j=I+1}^{M} \left(\frac{\theta^j}{\mu^j} \right)^{k_j}. \tag{3.4}$$

An intuitive explanation of the Norton equivalent of the forward network is, that ν_l is the throughput of the short-circuited forward network given that l packets circulate in it. The mathematical explanation of the Norton equivalent is that ν_l is the conditional expectation of the throughput of the forward network given that there are l packets in it [8]. Similar interpretations can be given for the Norton equivalent of the acknowledgment network.

The mapping F defined by:

$$F(T) = \max_{\mathcal{O}\tau \leqslant T} \mathcal{O}\gamma \tag{3.5}$$

is called the throughput time delay function.

If we assume that the acknowledgment packets instantaneously reach the controller $\eta_m = \infty$, for all $m = 1, 2, \cdots, N$ (see also fig. 3). In [11] it was proven that with concave increasing service rates of the Norton's equivalent (as in the case of Jackson networks [15–17]) the optimal flow control is a window type mechanism.

PROPOSITION 3.1

If ν_k is a concave increasing function with respect to k, the optimal flow control is a window type mechanism with a random point, if it exists, corresponding to the last packet in the window, i.e.,

$$\lambda_k = \begin{cases} c & \text{if } 0 \leqslant k \leqslant L - 2, \\ 0 < \lambda_{L-1} \leqslant c & \text{if } k = L - 1, \\ 0 & \text{if } L \leqslant k. \end{cases} \tag{3.6}$$

Proof

The proof of this proposition is in [11]. Algorithm 3.4 can be used for the computation of the optimal flow control. □

Except for at most one packet, the previous proposition indicates that all packets should be injected into the network at the maximum rate c. The difference rate of the last packet in the window would increase the difficulty in implementation. This difficulty can be easily avoided in practice by only considering a discrete number of admissible delays $T = T_{\max}^{(L)}$, where L will be the window size. Here $T_{\max}^{(L)}$ is the maximum delay that can be achieved with at most L packets in the network. Let

$$\rho_k = \prod_{l=0}^{k-1} \frac{c}{\nu_{l+1}},$$

for all $k \geq 1$. The tuple $(T_{\max}^{(L)}, F(T_{\max}^{(L)}))$ for all $L \geq 1$ is called an operating point.

COROLLARY 3.2

The set of operating points is given by:

$$\left(T_{\max}^{(L)}, F(T_{\max}^{(L)}) \right) = \left(\frac{\sum\limits_{k=1}^{L} k\rho_k}{\sum\limits_{k=1}^{L} \nu_k \rho_k}, \frac{\sum\limits_{k=1}^{L} \nu_k \rho_k}{1 + \sum\limits_{k=1}^{L} \rho_k} \right), \tag{3.7}$$

for all $L \geq 1$.

3.1.2. Optimum control with acknowledgment delay

Let us assume that at any given moment, at most N packets can be unacknowledged (see fig. 4). The *equivalent controller* that describes the behavior of the original controller together with the acknowledgment network is a state dependent processor that corresponds to their Norton equivalent. Let δ_l be the equivalent state dependent arrival rate into the forward network when there are l packets in the forward network. From fig. 3 we see that δ_l is the Norton equivalent of the acknowledgment and the controller, given that N packets circulate in the whole closed network and l packets are in the forward network. If m packets are in the acknowledgment network, its service rate is η_m, and the service rate of the controller is λ_{l+m}. δ_l can be easily computed as the throughput of a closed network with $N - l$ packets circulating in it. Then,

$$\delta_l \overset{\text{def}}{=} \frac{\sum\limits_{n=1}^{N-l} \prod\limits_{k=1}^{n} \left(\frac{\lambda_{l+k-1}}{\eta_k} \right) \eta_n}{1 + \sum\limits_{n=1}^{N-l} \prod\limits_{k=1}^{n} \left(\frac{\lambda_{l+k-1}}{\eta_k} \right)}, \tag{3.8}$$
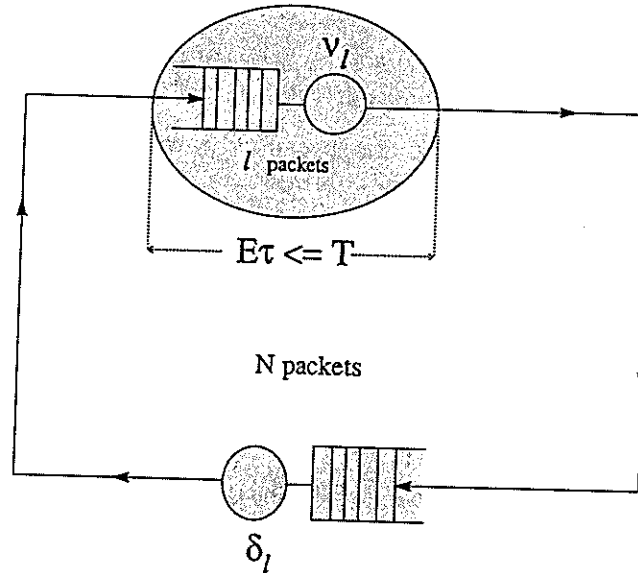
for all $l \leq N$.

Fig. 4. The equivalent forward network with the equivalent controller.

From eq. (3.8), we see that δ_l is a function of $\lambda_l, \lambda_{l+1}, \ldots, \lambda_{N-1}$, for all $l \leq N$. In particular δ_{N-1} is only a function of λ_{N-1}.

Notice that δ_l is the average throughput of a closed network with $N - l$ packets. This network consists of two queueing systems with rates η_m and λ_{l+m}, respectively. Because the service rate η_m is concave increasing, the maximum throughput of this closed network is achieved when the arrival process follows a window flow control policy (see proposition 3.1).

PROPOSITION 3.3

The optimal flow control policy of a Jackson network with acknowledgment delay is of the form

$$\lambda_l = \begin{cases} c & \text{if } 0 \leq l < L - 1, \\ 0 < \lambda_{L-1} \leq c & \text{if } l = L - 1, \\ 0 & \text{if } L \leq l. \end{cases} \tag{3.9}$$

L and λ_{L-1} take the highest feasible values that do not violate the time delay constraint $\mathscr{E}Q - T\mathscr{E}\gamma \leq 0$.

Proof

If l is the total number of packets inside the forward network, then its service rate ν_l is a concave increasing function with respect to l, for $l \geq 0$. From proposition 3.1 we conclude that the optimal flow control is a window flow control with respect to the equivalent arrival rates δ_k, for $k \geq 0$. Therefore, under the time delay constraint $\mathscr{E}\tau \leq T$, each variable δ_k, for $k \geq 0$, must take

the highest possible value. From the discussion that follows eq. (3.8), we conclude that δ_k takes the highest possible value when the variables λ_k, $\lambda_{k+1},\ldots$, take their highest possible values, for $k \geq 0$.

Therefore, we first assign to λ_0 its highest possible value; we next assign to λ_1 its highest possible value; and so on for λ_2, $\lambda_3,\ldots$. The only condition that must be maintained is $\mathscr{E}\tau \leq T$. Therefore the optimal control is a window mechanism with at most one random point at the end of the window. □

In what follows an iterative algorithm for the computation of the optimal flow control is given.

ALGORITHM 3.4

Step 0: $L := 1$. Set $\lambda_0 := c$ and $\lambda_k := 0$ for all k, $k \geq 1$. Check to see whether $\mathscr{E}\tau \leq T$. If yes, continue to step 1. Otherwise stop; no packets can enter into the network.

Step 1: $L := L + 1$. Set $\lambda_k := c$ for all k, $0 \leq k \leq L - 1$. Check whether $\mathscr{E}\tau \leq T$. If yes, repeat step 1. Else, find the exact value of λ_{L-1} (which is between 0 and c), with which the last packet should be accepted and which results in $\mathscr{E}\tau = T$; the resulting flow control is the optimal flow control; stop.

Let

$$\epsilon_k = \prod_{l=0}^{k-1} \frac{\delta_l}{\nu_{l+1}},$$

for all $k \geq 1$. We have the following:

COROLLARY 3.5

The set of operating points of the forward networks is given by:

$$\left(T_{\max}^{(L)}, F(T_{\max}^{(L)})\right) = \left(\frac{\displaystyle\sum_{k=1}^{L} k\epsilon_k}{\displaystyle\sum_{k=1}^{L} \nu_k\epsilon_k}, \frac{\displaystyle\sum_{k=1}^{L} \nu_k\epsilon_k}{1 + \displaystyle\sum_{k=1}^{L} \epsilon_k} \right), \tag{3.10}$$

for all $L \geq 1$.

3.1.3. The effect of feedback information on the performance of the optimal control

In this section we study the effect of the delayed feedback information on the optimal window flow control.

LEMMA 3.6

For a given optimal flow control policy, the state dependent rate of the aggregated controller δ_l for all l, $0 \leqslant l \leqslant L - 1$, is

(i)   an increasing function with respect to the controller rates λ_k for all k, $l \leqslant k \leqslant L - 1$,

(ii) an increasing function with respect to the service rates of the processors of the acknowledgment network,

(iii) a concave decreasing function with respect to l, $0 \leqslant l \leqslant L - 1$.

Proof

When l packets are in the forward network δ_l is the Norton equivalent of a network consisting of the original controller and the Norton equivalent of the acknowledgment network (see figs. 3, 4 and eq. (3.8)). The result follows from proposition 6.1(i) (see the appendix). To prove the second statement note that the Norton equivalent of the acknowledgment network is an increasing function with respect to the service rate of any of its constituent service processors [15]. The result then follows from proposition 6.2(i). The fact that δ_l is decreasing is a direct conclusion of proposition 6.1(i). Concavity can be shown based on [15]. □

Lemma 3.7 below demonstrates the way the window size is affected due to variations in the service rates of the processors of the acknowledgment network.

LEMMA 3.7

The window size of the optimal flow control is

(i)   decreasing with respect to the maximum controller rate c,

(ii) increasing with respect to the service rate of any of the processors of the forward network,

(iii) decreasing with respect to the service rate of any of the processors of the acknowledgment network.

Proof

(i) From proposition 3.3 the optimal flow control is of the form $\lambda_l = c$ for $0 \leqslant l \leqslant L - 1$, and $0 < \lambda_{L-1} \leqslant c$. Let $c^* > c$ and let us change the values of the optimal arrival rates to $\lambda_l^* = c^*$ for $0 \leqslant l \leqslant L - 1$, $\lambda_{L-1}^* = \lambda_{L-1}$, and $\lambda_l^* = 0$ for $l \geqslant L$. From lemma 3.6(i) we conclude that the service rates of the aggregated controller δ_l, $0 \leqslant l \leqslant L - 1$, increase, and from proposition 6.1(v) the expected time delay of the packets in the forward network increases. But under the previous optimal policy $\mathscr{E}\tau = T$. Therefore under the new control policy the time delay constraint is violated. As a result for the optimal flow control $\lambda_{L-1}^* < \lambda_{L-1}$, and the result follows. The proof of (iii) is based on lemma 3.6(ii) and arguments similar to the ones used above.

(ii) The network operates under the optimal flow control. Without affecting the flow control policy we increase the service rate of any of the processors in the forward network. The service rate of equivalent controller δ_l, $0 \leqslant l \leqslant L - 1$, does not change. From lemma 3.6(iii) and proposition 6.2(iv) we conclude that the expected time delay of the packets in the forward network decreases. The result then follows from proposition 3.3.   □

LEMMA 3.8

For a given upper bound on the average time delay T of the forward network, the optimum average throughput $\mathcal{E}\gamma$ is

(i)   an increasing function with respect to the maximum capacity c of the original controller,

(ii) an increasing function with respect to the service rates of the processors in the forward network,

(iii) an increasing function with respect to the service rates of the processors of the acknowledgment network.

Proof

(i) Let $c^* > c$. Then the set of the control policies that can be implemented when $0 \leqslant \lambda_l \leqslant c^*$, for $l \geqslant 0$, is a superset of the control policies that can be implemented when $0 \leqslant \lambda_l \leqslant c$ for $l \geqslant 0$. The result then follows.

(ii) The network operates under the optimal flow control. Without affecting the flow control policy we increase the service rate of any of the processors in the forward network. The service rate of the equivalent controller δ_l, $0 \leqslant l \leqslant L - 1$, does not change. From lemma 3.6(iii), proposition 6.2(i), and proposition 6.2(iv) we conclude that the network throughput increases and that the expected time delay of the packets in the forward network decreases. The result then follows from proposition 3.3.

(iii) From lemma 3.6(ii) we know that the state dependent service rate of the aggregate controller is an increasing function with respect to the service rate of any of the processors in the acknowledgment network. From proposition 3.3, we know that higher maximum arrival rates δ_l^*, $0 \leqslant l \leqslant L - 1$, result in greater network throughput for a given T. The result then follows.   □

The above results show that as the congestion in the acknowledgment network builds up, the effective arrival rate of packets in the forward network decreases. Therefore, the network throughput decreases whereas the size of the optimal window increases. Notice that if the service rate of the processors of the acknowledgment network drops below a critical value, the acknowledgment network becomes the bottleneck of the flow control protocol. In the limiting situation in which the service rates of the processors of the acknowledgment network tend to zero, the network throughput tends to zero whereas the size of the optimal window tends to infinity. Recall that a different limiting case was

studied in section 3.1.2. There, the maximum obtainable average throughput for the network was obtained.

Another limiting case highlights the effect of the acknowledgment network on the performance of the optimal window flow control. This case arises when $c \to \infty$, i.e., the acknowledgment network becomes the effective network controller. Thus, the limitations on the network performance imposed by the acknowledgment network can be inferred. The actions of the controller are instantaneous in this case. They amount to a specification of the actual window size.

Let

$$\beta_k = \prod_{l=0}^{k-1} \frac{\eta_l}{\nu_{l+1}},$$

for all $k \geqslant 1$. We have the following:

COROLLARY 3.9

The upper bound on the set of operating points of the forward network is given by:

$$\left(T_{\max}^{(L)}, F\left(T_{\max}^{(L)} \right) \right) = \left(\frac{\sum_{k=1}^{L} k\beta_k}{\sum_{k=1}^{L} \nu_k \beta_k}, \frac{\sum_{k=1}^{L} \nu_k \beta_k}{1 + \sum_{k=1}^{L} \beta_k} \right), \tag{3.11}$$

for all $L \geqslant 1$.

Proof

When $c \to \infty$ the closed network of fig. 3 consists of the Norton equivalents of the forward and acknowledgment networks. The result is readily obtained by observing that the network contains L packets. □

3.2. OPTIMAL STATE INDEPENDENT FLOW CONTROL OF A JACKSON NETWORK

By abuse of notation, let the $1 \times I$ matrix $\Theta = [\theta^1 \cdots \theta^I]$ be the solution of the traffic flow equations of the forward network, i.e.,

$$\Theta = \Lambda + \Theta R,$$

or

$$\Theta = \Lambda (I - R)^{-1}. \tag{3.12}$$

Here Λ denotes the load vector of the input traffic flows

$$\Lambda = \lambda \left[r^{\cdot 1} \cdots r^{\cdot I} \right].$$

With

$$\alpha = \left[\alpha^1 \cdots \alpha^I\right] \stackrel{\text{def}}{=} \left[r^{\cdot 1} \cdots r^{\cdot I}\right](\mathbf{I} - \mathbf{R})^{-1},$$

we obtain

$$\theta^j = \alpha^j \lambda,$$

for all j, $j = 1, 2, \cdots, I$. The time delay amounts to

$$\mathscr{E}_T = \frac{\mathscr{E}Q}{\mathscr{E}\gamma} = \frac{\mathscr{E}Q^1 + \cdots + \mathscr{E}Q^I}{\lambda}.$$

Equivalently,

$$\mathscr{E}_T = \sum_{j=1}^{I} \frac{\alpha^j}{\mu^j - \alpha^j \lambda}. \tag{3.13}$$

The expected time delay is an increasing function of the external arrival rate. Therefore, the maximum feasible arrival rate λ that achieves the upper bound constraint T

$$\sum_{j=1}^{I} \frac{\alpha^j}{\mu^j - \alpha^j \lambda} \leqslant T \tag{3.14}$$

is optimal. We have, thus, shown the following:

LEMMA 3.10

The maximum feasible rate λ is given by:

$$\sum_{j=1}^{I} \frac{\alpha^j}{\mu^j - \alpha^j \lambda} = T. \tag{3.15}$$

The interested reader could refer to [1] for an extensive study of rate based flow control.

3.3. A COMPARISON BETWEEN THE TWO FLOW CONTROL STRATEGIES

PROPOSITION 3.11

Given a maximum average time delay T for the forward network,

(i)   given an acknowledgment network and a maximum capacity c for the controller, and a given maximum expected delay T, there exists a surface in the space of service rates of the forward network such that any forward network having rates below this surface will have a window flow control policy as optimal;

(ii)   for a given capacity of the controller c and a given forward network, and a given maximum expected delay T, there is a surface in the space of service
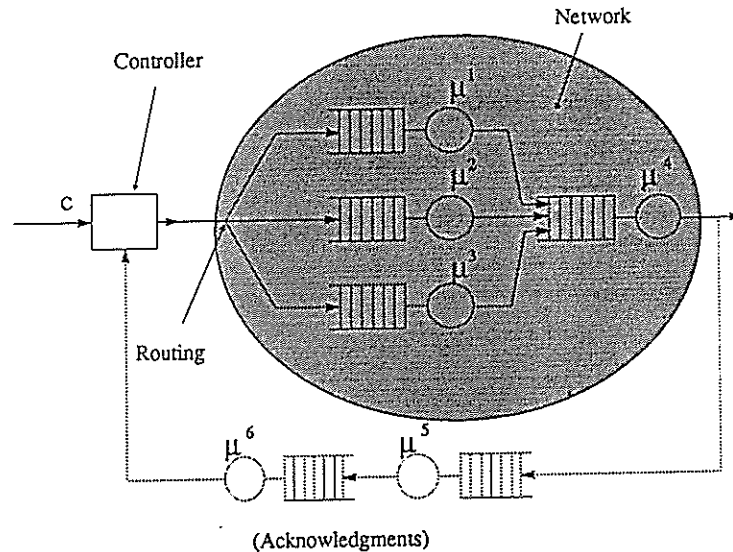
Fig. 5. A network of processors with acknowledgment delays subject to state dependent flow control.

rates of the acknowledgment network such that any acknowledgment network that has the rates above this surface will have a window flow control as the optimal policy for controlling the forward network;

(iii) given the forward and the acknowledgment network, and a given maximum expected delay T, there is a c^*, such that for $c \leqslant c^*$ the rate control leads to higher performance. For $c \geqslant c^*$, the window flow control leads to higher performance.

Proof

This proposition is based on the results presented in the previous subsection.

Example

In the sequel the effect of delayed feedback information on network performance is examined. Consider the network of processors depicted in fig. 5. Packets carrying information and acknowledgments reach their destination in two hops. The service rates of the processors are $\mu^1 = 2.0$ packets/sec, $\mu^2 = 1.0$ packet/sec, $\mu^3 = 0.5$ packet/sec, and $\mu^4 = 3.0$ packets/sec. For the case in which the network operates under state dependent flow control, acknowledgment packets are sent back to the source. In this example the specific choice of the acknowledgment network is not important. What is important is that the Norton equivalent of the acknowledgment network is a concave increasing function with respect to the number of packets in it. Our choice of acknowledgment network captures this effect. In our example the acknowledgment network is represented by two processors in tandem with service rates μ^5 packet/sec and
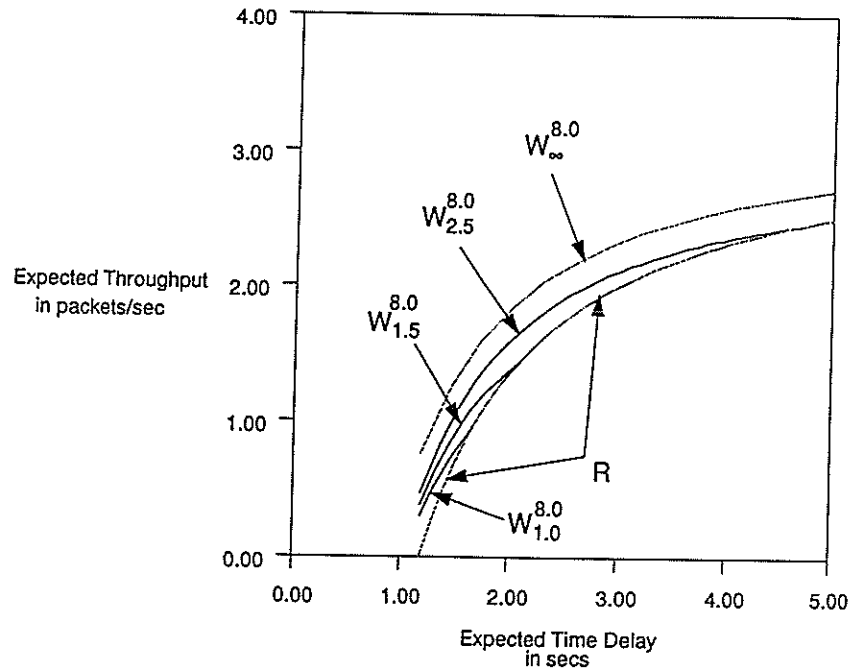
Fig. 6. The effect of the acknowledgment network on the average throughput.

μ^6 packet/sec. Packets arrive into the network with state dependent arrival rate λ_k, where 0.0 packets/sec $\leq \lambda_k \leq c$ packets/sec. The packets are routed to server μ^1 with probability 4.0/7.0, to server μ^2 with probability 2.0/7.0 and to server μ^3 with probability 1.0/7.0.

Network performance results are labeled with the letter W or R, indicating whether the results were obtained using window control or rate control, respectively. The results show the dependence of the average throughput of the forward network and the window size on two parameters, namely the maximum controller capacity c, and the service rate of the processors in the acknowledgment network. Although the average throughput of the forward network and window size also depend on the service rate of the processors in the forward network, we have not included the corresponding curves because of space constraints.

In fig. 6, the curve W_k^x depicts the network performance under state dependent flow control and state independent routing when $\mu^5 = \mu^6 = k$ packets/sec and $c = x$ packets/sec. Curve R shows the network performance when the network is subject to state independent flow control. In fig. 7, the optimal window size is depicted as a function of the upper bound of the time delay constraint T. The sets of curves shown in figs. 6 and 7 illustrate three important points: (1) Network performance deteriorates as the acknowledgment delays increase (see lemma 3.7(iii) and lemma 3.8(iii)). (2) The window size increases as
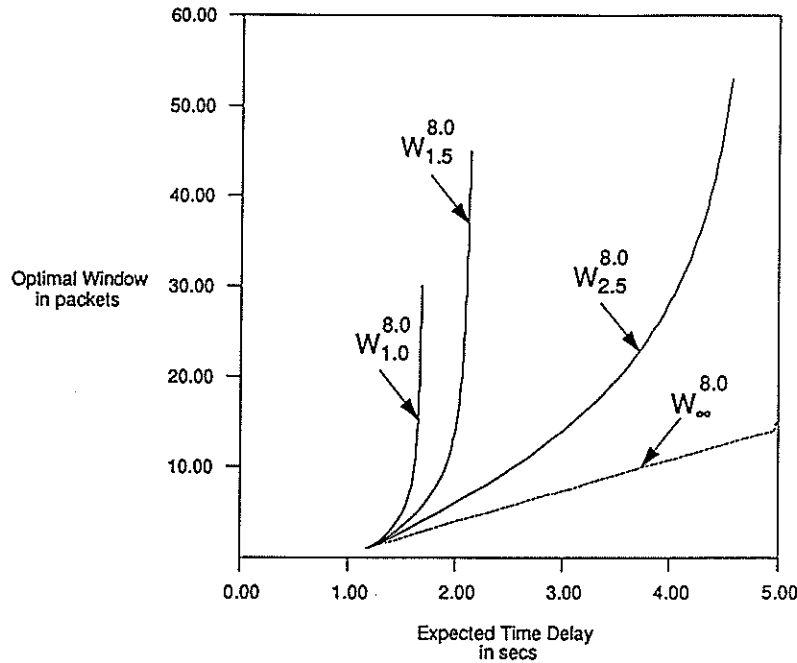
Fig. 7. The effect of the acknowledgment network on the optimal window size.

acknowledgment delays increase. (3) *If* a given end-to-end time delay is achievable using both window flow control and state independent flow control, then network throughput is greater using window flow control. However, using a window flow control, a given end-to-end time delay may not be achievable. For example, as we see in fig. 7, when $\mu^5 = \mu^6 = 1.5$ packets/sec, the maximum achievable end-to-end time delay is roughly 2.0 sec. We see from fig. 6 that if an acceptable end-to-end time delay is 3.0 sec, then the achievable network throughput using rate flow control exceeds that which can be achieved using window flow control. This is an example of a situation in which *the delayed feedback information represents the effective bottleneck of the flow control mechanism. As a result, the state independent flow control becomes a more effective alternative to the state dependent flow control.*

In fig. 8, curve $W_{1.0}^k$ depicts the network performance under state dependent flow control and state independent routing when $\mu_5 = \mu_6 = 1.0$ packets/sec, and $c = k$ packets/sec. In fig. 9, the optimal window size is depicted as a function of the upper bound of the time delay constraint T. The sets of curves shown in figs. 8 and 9 show that as the controller capacity increases, the network throughput increases and the optimal window size decreases (see lemma 3.8(i) and lemma 3.7(i)).

In fig. 10, we demonstrate the way in which network performance is affected when the service rate of a processor in the acknowledgment network is varied and when the end-to-end expected time delay is held fixed at 1.7 sec.
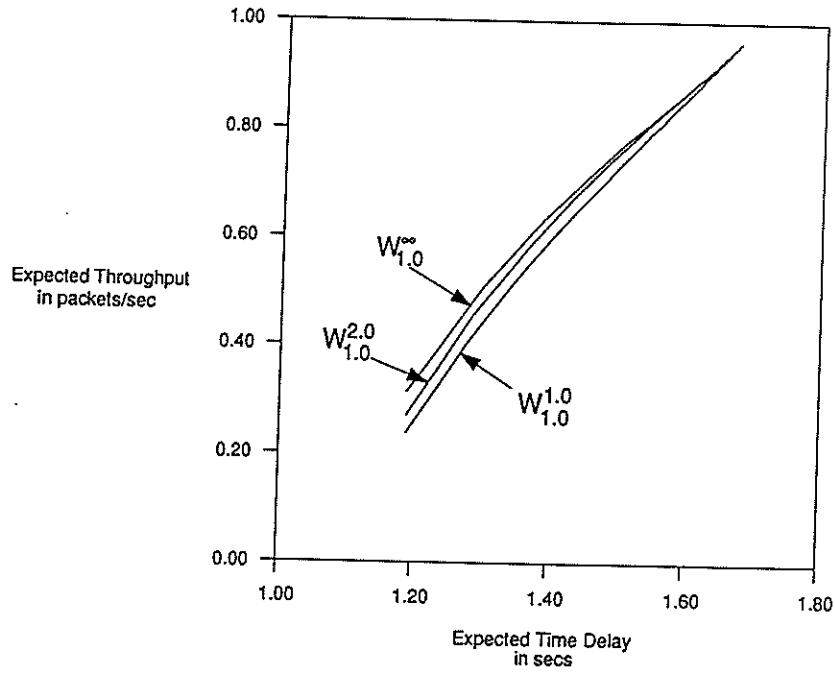
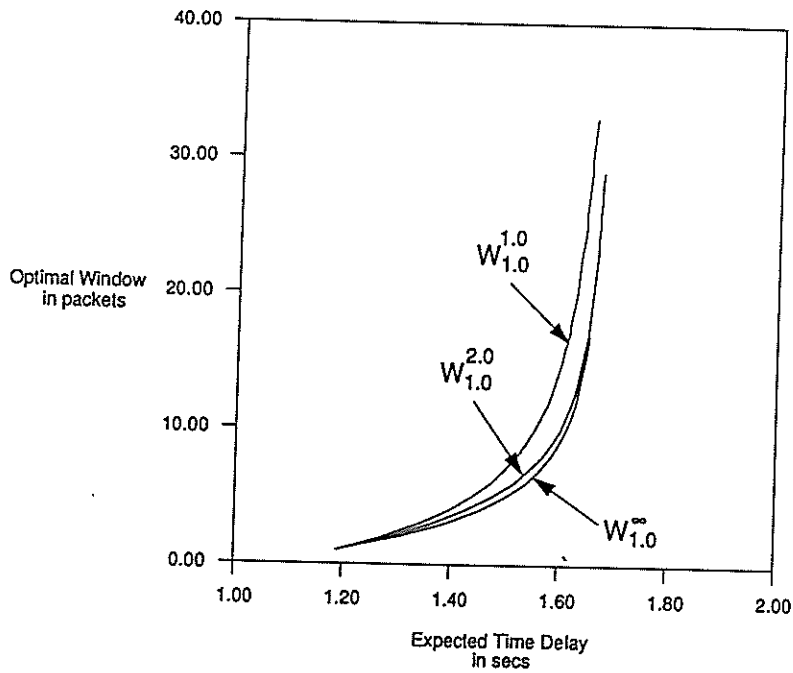Fig. 8. The effect of the capacity constraint on the average throughput.

Fig. 9. The effect of the capacity constraint on the optimal window size.

c = μ_5 = 6.0 packets/sec
Eτ = 1.7 secs

Optimal Window Size
in packets

Service rate μ_6 in packets/sec

c = μ_5 = 6.0 packets/sec
Eτ = 1.7 secs

Expected Throughput
in packets/sec
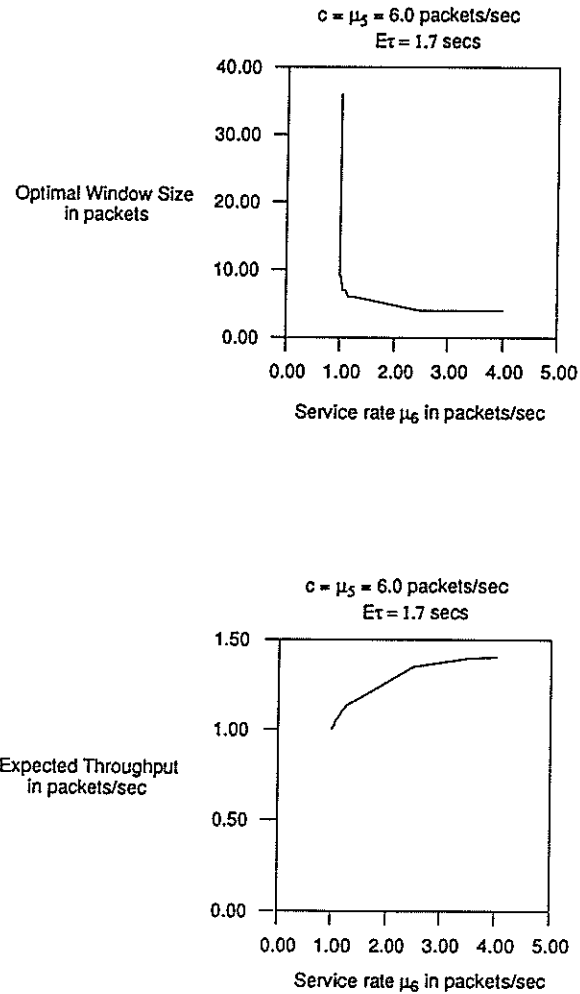
Service rate μ_6 in packets/sec

Fig. 10. The effect of the acknowledgment network on network performance.

Summarizing the results presented thus far, we have compared the performance of networks operating under either window or rate flow control and have studied the dependence of the optimal window flow control on different network parameters.

In practical situations we would like to have some idea of the size of the optimal flow control window as quickly as possible. In the sequel we introduce two ways of approximating the optimal window size. Both approximations are computationally less complex than algorithm 3.4. We study the accuracy of these approximations through examples. Our motivation for trying to assess rapidly the optimal window size arises from our recognition that the size of the window flow control cannot exceed the size of the available buffer utilized for its
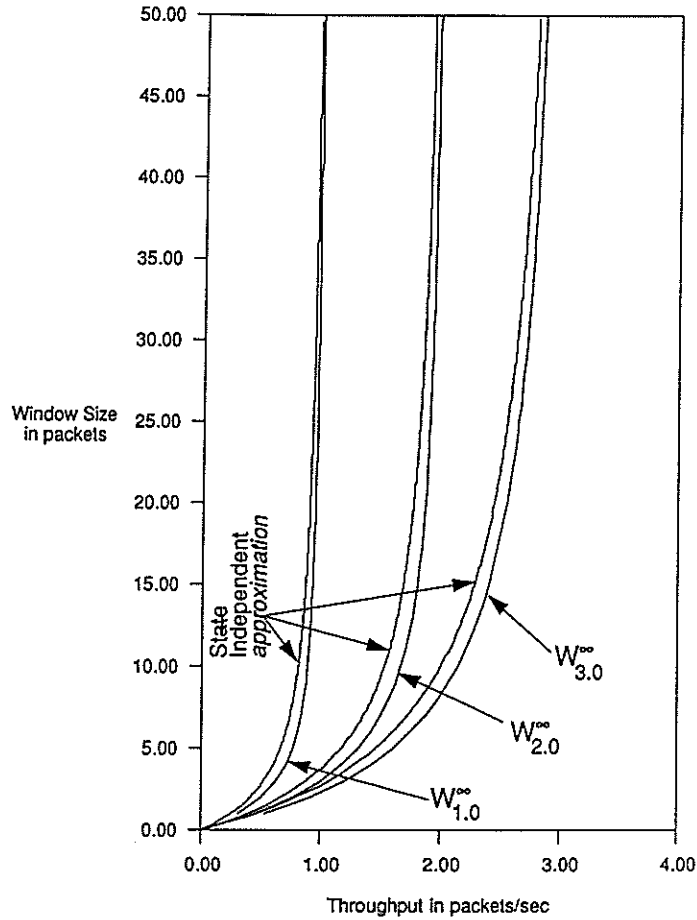
Fig. 11. Approximate window size.

implementation. Thus we would like to decide as quickly as possible whether to use a window or rate flow control given a maximum available buffer size.

We first approximate the optimal window size by the optimal window size corresponding to a network operating with $c = \infty$. We note that when $c = \infty$, we can directly apply corollary 3.9. The $c = \infty$ approximation gives a lower bound on the size of the optimal window. See fig. 9. We next approximate the optimal window size by the expected number of packets in both the forward and acknowledgment networks operating under state independent flow control and under the constraint that the expected time delay of the packets in the forward network does not exceed a given upper bound T. We see from figs. 11 and 12 that this approximation is quite accurate and thus can be used in practical applications. From numerous examples, we observe that the approximation appears to provide an upper bound for the required window size.
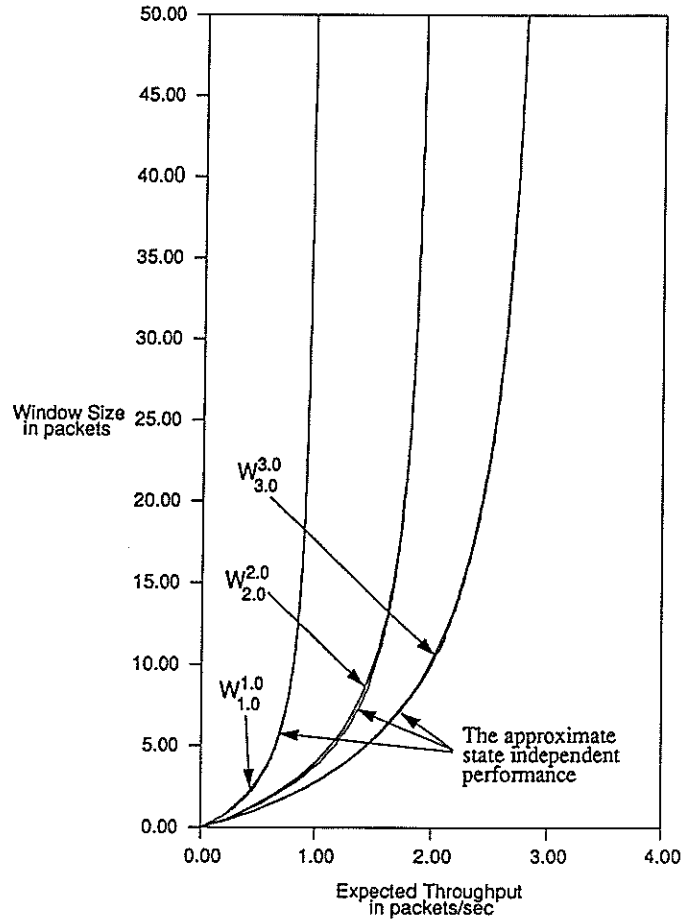
Fig. 12. Approximate window size (cont.).

Concluding, we believe that the examples studied in this section reveal realistic and computationally effective approximations that can be used in practice for the determination of the effectiveness of rate-based flow control as well as window-based flow control. We notice that the congestion of the acknowledgment network and its effect on the optimal window size can be effectively approximated by either a network operating with $c = \infty$ or an open network (consisting of both the forward and the acknowledgment networks) operating under state independent flow control.

When the original network was approximated by an open network, we noticed that the optimal window was approximately equal to the expected number of packets in the forward and acknowledgment networks. Notice that at server j of the acknowledgment network, the expected number of packets is

$$EQ^j = \frac{\alpha^j \lambda}{\mu^j - \alpha^j \lambda}.$$

Therefore if congestion builds up in server j of the acknowledgment network, in order to keep the throughput at the same level, the user should increase its window size. Simple calculations reveal that if the service rate of the server j is changed from μ^j_{old} to μ^j_{new}, the user can keep its throughput at the same level if the network user changes its window size according to the equation

$$EQ^j_{new} - EQ^j_{old} = \frac{\mu^j_{old} - \mu^j_{new}}{\left(\mu^j_{old} - \alpha^j\lambda\right) \times \left(\mu^j_{new} - \alpha^j\lambda\right)}.$$

The previous equation also reveals that if the service rate decreases, the user should increase the window size, whereas if the service rate increases, the user should decrease the window size. Obviously this approximation does not take into account the effect of changes in the end-to-end expected time delay. It can be used, however, as a first approximation of the effect of congestion on the user's window size.

If, on the other hand, the user wishes to keep its window size fixed and at the same time maintain its throughput, the user should reduce its effective rate through that particular resource. This can be done by reducing the acknowledgment rate through the utilization of selective acknowledgment or by the introduction of additional resources into the network.

We believe that these examples also suggest that in a real network, combinations of window and rate based flow control, could result in a rich and effective family of flow and congestion control schemes. A scheme along these lines is explored in [5].

4. Conclusions

In the present paper the effect of acknowledgment delays was studied in detail. We first showed that when congestion arises in the forward network, the size of the optimal window of state dependent flow control that maximizes the performance of the network *decreases*. On the other hand, when congestion appears in the acknowledgment network, the size of the optimal window *increases*. We further proved that under heavy congestion in the acknowledgment network, the state independent flow control avoids the acknowledgment network congestion and thus gives better performance. Whereas the congestion that arises in the forward network affects both window and rate flow control, the congestion that appears in the acknowledgment network affects only the state dependent (window) flow control.

The problem of flow control remains a subtle issue in computer communications. A number of questions require further study. Is it really advantageous to have an end-to-end control all the time? The previous analysis suggests that it would be advantageous to have an end-to-end window flow control as long as there is no congestion and as long as the optimal window size is acceptable.
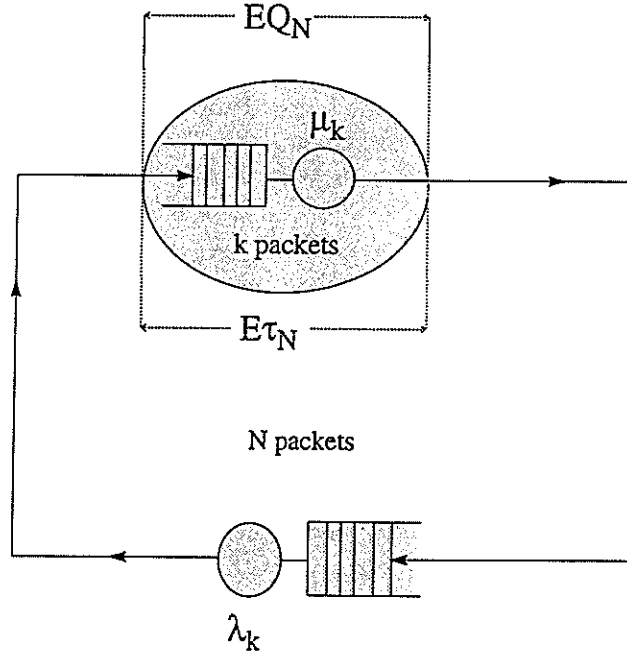
Fig. 13. A single class network with at most N packets subject to state dependent flow control.

Whenever local congestion affects the acknowledgment packets, local control procedures are needed to attempt to eliminate or by-pass the congestion. One such mechanism is scheduling. By using scheduling, the detrimental effects of congestion in the acknowledgment network would not affect the performance of the whole network. In addition, combinations of window and rate based flow control, could result in a rich and effective family of flow and congestion control schemes.

Appendix (monotonicity properties for a controlled finite birth–death process)

In this appendix we analyze the behavior of the closed network depicted in fig. 13. If k is the total number of packets in the upper processor, the upper and the lower processors serve packets with state dependent rates μ_k and λ_k, respectively. Let $\mathscr{E}\gamma_N$, $\mathscr{E}Q_N^c$ and $\mathscr{E}\tau_N^c$ be the expected throughput, the expected number and the expected time delay of the packets in the lower processor of the network in fig. 13, respectively. Similarly, let $\mathscr{E}Q_N$ and $\mathscr{E}\tau_N$ be the expected number and the expected time delay of the packets in the upper processor of the network in fig. 13. Observe that

$$\mathscr{E}\gamma_N = \frac{\mathscr{E}Q_N}{\mathscr{E}\tau_N} = \frac{\mathscr{E}Q_N^c}{\mathscr{E}\tau_N^c} = \frac{N}{E\tau_N + \mathscr{E}\tau_N^c}.$$

The results presented here follow an approach due to [18].

PROPOSITION 6.1

(i)   If μ_k is an increasing function of k, the expected throughput $\mathscr{E}\gamma_N$ is increasing in λ_k for $k = 0, 1, \ldots, N-1$.

(ii)   The expected number of packets $\mathscr{E}Q_N$ is increasing in λ_k for $k = 0, 1, \ldots, N-1$.

(iii) The expected number of packets $\mathscr{E}Q_N^c$ is decreasing in λ_k for $k = 0, 1, \ldots, N-1$.

(iv)   If μ_k is an increasing function of k, the expected time delay of the packets in the lower processor $\mathscr{E}\tau_N^c$ is decreasing in λ_k for $k = 0, 1, \ldots, N-1$.

(v)   If k/μ_k is an increasing function of k, the expected time delay $\mathscr{E}\tau_N$ is increasing in λ_k for $k = 0, 1, \ldots, N-1$.

Proof

Let $p^* = (p_0^*, \ldots, p_N^*)$ correspond to the control $\lambda^* \overset{\text{def}}{=} (\lambda_0^*, \ldots, \lambda_N^*)$ and let $p = (p_0, \ldots, p_N)$ correspond to the control $\lambda \overset{\text{def}}{=} (\lambda_0, \ldots, \lambda_N)$. Let us assume that $\lambda_i \geq \lambda_i^*$ for all $i \leq N-1$. Then

$$\frac{p_k^*}{p_{k-1}^*} = \frac{\lambda_{k-1}^*}{\mu_k} \leq \frac{\lambda_{k-1}}{\mu_k} = \frac{p_k}{p_{k-1}},$$

for $k = 1, \ldots, N-1$, from which it follows that

$$\sum_{i=k}^{N} p_i^* \leq \sum_{i=k}^{N} p_i.$$

Since the μ_k's are increasing,

$$\sum_{k=1}^{N} \mu_k p_k^* \leq \sum_{k=1}^{N} \mu_k p_k,$$

which completes the proof of (i).

(ii) $\mathscr{E}Q_N = \sum_{k=1}^{N} k p_k^*$. The arguments of (i) hold here if μ_k is substituted for k.

(iii) $\mathscr{E}Q_N^c = N - \mathscr{E}Q_N$. The statement is then true because of (ii).

(iv) This statement holds because $\mathscr{E}\tau_N^c = \mathscr{E}Q_N^c / \mathscr{E}\gamma_N$.

(v) $\mathscr{E}\tau_N$ is a weighted average of k/μ_k, for $k = 1, \ldots, N$, with weights $q_k^* = \mu_k p_k^* / \sum_{i=1}^{N} \mu_i p_i^*$. The arguments of (i) hold if q_k^* is substituted for p_k^*, and the statement follows. $\square$

In a similar way we can prove the following relations.

PROPOSITION 6.2

(i)   If λ_k is a decreasing function of k, the expected throughput $\mathscr{E}\gamma_N$ is increasing in μ_k for $k = 0, 1, \ldots, N-1$.

(ii)   The expected number of packets $\mathscr{E}Q_N^c$ is increasing in μ_k for $k = 0, 1, \ldots, N-1$.

(iii) The expected number of packets $\mathscr{E}Q_N$ is decreasing in μ_k for $k = 0, 1, \ldots, N-1$.

(iv) If λ_k is a decreasing function of k, the expected time delay of the packets in the network $\mathscr{E}\tau_N$ is decreasing in μ_k for $k = 0, 1, \ldots, N-1$.

(v) If $(N-k)/\lambda_k$ is a decreasing function of k, the expected time delay $\mathscr{E}\tau_N^c$ is increasing in μ_k for $k = 0, 1, \ldots, N-1$.

References

[1] D. Bertsekas and R. Gallager, *Data Networks* (Prentice Hall, Englewood Cliffs, NJ, 1987).

[2] A.D. Bovopoulos and A.A. Lazar, Optimal resource allocation for Markovian queueing networks: The complete information case, Stochastic Models 7 (1991).

[3] A.D. Bovopoulos and A.A. Lazar, Load balancing algorithms for Jacksonian networks with acknowledgment delays, *Proc. IEEE INFOCOM'89 Conf.*, Ottawa, Canada (1989) pp. 749–757.

[4] A.D. Bovopoulos, Resource allocation for Markovian queueing networks: the partial information case, Technical Report WUCS-89-22, Department of Computer Science, Washington University (1989).

[5] A.D. Bovopoulos, Performance evaluation of a traffic control module for ATM networks, Technical Report WUCS-91-22, Department of Computer Science, Washington University (1991).

[6] D. Chiu and R. Jain, Analysis of the increase and decrease algorithms for congestion avoidance in computer networks, Computer Networks and ISDN Systems 17 (1989) 1–14.

[7] D. Comer and R. Yavatkar, FLOWS: Performance guarantees in best effort delivery systems, *Proc. IEEE INFOCOM'89 Conf.*, Ottawa, Canada (1989) pp. 100–109.

[8] M.-T. Hsiao and A.A. Lazar, An extension to Norton's equivalent, Queueing Systems 5 (1989) 401–411.

[9] M.-T. Hsiao and A.A. Lazar, Optimal flow control of multi-class queueing networks with partial information, IEEE Trans. Auto. Control AC-35 (1990) 855–860.

[10] M.-T. Hsiao and A.A. Lazar, Optimal decentralized flow control of Markovian queueing networks with multiple controllers, Performance Evaluation (1991) to appear.

[11] A.A. Lazar, Optimal flow control of a class of queueing networks in equilibrium, IEEE Trans. Auto. Control AC-28 (1983) 1001–1007.

[12] D.G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. (Addison–Wesley, 1984).

[13] K.K. Ramakrishnan and R. Jain, A binary feedback scheme for congestion avoidance in computer networks, ACM Trans. Computer Systems (1990) 158–181.

[14] M. Schwartz, *Telecommunications Networks, Protocols, Modeling and Analysis* (Addison–Wesley, 1987).

[15] J.G. Shanthikumar and D. Yao, Second-order properties of the throughput of a closed queueing network, Math. Oper. Res. 13 (1988) 524–534.

[16] J.G. Shanthikumar and D. Yao, Second-order stochastic properties in queueing systems, Proc. IEEE 77 (1989) 162–170.

[17] J. Walrand, *An Introduction to Queueing Networks* (Prentice Hall, New Jersey, 1988).

[18] G. Weiss, private communication (1982).

[19] L. Zhang, Designing a new architecture for packet switching communication networks, IEEE Commun. Mag. 25 (1987) 5–12.

[20] L. Zhang, Some thoughts on the packet network architecture, ACM Comput. Commun. Rev. 17 (1987).