A FILTERED-BASED FLOW CONTROL SCHEME

By

Chia-Feng Lee

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
AT
NATIONAL CHIAO TUNG UNIVERSITY
1001, DA-HSUEH ROAD, HSIN CHU, TAIWAN 30050, R.O.C.
June 1, 2000

NATIONAL CHIAO TUNG UNIVERSITY
DEPARTMENT OF
COMMUNICATION ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "**A filtered-based flow control scheme**" by **Chia-Feng Lee** in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE**.

Dated: June 1, 2000

Supervisor: _____
Po-Ning Chen

Readers: _____
Cheng-shang Chang

_____
Maria C. Yuang

_____
Jung-Tsung Tsai

Table of Contents

List of Tables

List of Figures

Abstract

In this thesis, we propose a new flow control scheme based on filter theory, where the conventional $(+, \times)$ algebra is replaced by the $(\min, +)$ algebra.

An essential part of the filter flow control is to determine the filter formula for the measured arrival and departure so that we can use "convolution" of all the intermediate node filters to estimate the end-to-end congenstionness.

We first show that it is unlikely to determine the exact filter expression for given input arrival and output departure in terms of spectrum analysis or transformation technique. We then prove in terms of time-domain analysis that there are situations where the filter formula (for given arrival and departure processes) does not exist. We therefore turn to an absolute-error approximation of the filter expression for which the output due to the measured input and the estimated filter yield minimum absolute-error with the true departure. For ease of convolution calculation, we only take first-order filter approximate $\hat{h}(t)$ whose formula can be expressed as $c \cdot t + d$, where t is the time scale.

Since $h(t)$, by filter theory, behaves as an upper bound to the departure process, we then employ the sample of $x(t) - h(t)$ at the end of each measured window as a degree of crowdedness over the network, where $x(t)$ is the non-decreasing input arrival process. In order to enhance the performance (throughput), we also propose to include both the feedback and feedforward regulation rules in our flow control scheme. The feedback regulation aims at letting the previous node to regulate the departure rate once crowdedness is sensed. The feedforward regulation focuses on letting the node to re-decide the outgoing rate once crowdedness of its subsequent

nodes are sensed. Furthermore, as aforementioned, we also convolve the filter (i.e., c and d) of the subsequent nodes to estimate the degree of congestionness, which we named history information. As expected, each node will periodically pass its estimate of c and d to its previous node.

Experiment results over a four-node network will be performed on conventional rate-based and credit-based, as well as our newly proposed filter-based flow control schemes. Simulations show that the filter-based flow control can reach almost the same throughput and packet loss rate as the conventional rate-based and credit-based flow control schemes; however, it results in a much smaller variance of throughput than the other two. As a result, the filter-based flow control is more robust to variability of traffic patterns. When a system involved with more than four network nodes is considered, the superiority of filter-based flow control in its robustness becomes more certain.

Acknowledgements

I would like to thank Dr. Po-Ning Chen for his encouragement and guidance. This work would not have been possible without his advice and commitment. I would also like to thank all the people who have helped me, especially, the members of Network Technology Laboratory at Department of Communication Engineering, National Chiao Tung University.

Chapter 1

Introduction

With the increasing demand on network bandwidth due to the growth of network users, control on the smoothness of the traffic flows becomes significant in recent years. In short, flow control is to find an end-to-end control scheme operated on each intermediate network node such that the end-to-end traffic can flow smoothly.

In the literature, flow control schemes are categorized into two classes: rate-based flow control, which regulates traffic based on the buffer length, and credit-based flow control, which permits traffic according to the credits given by the subsequent nodes [8, 9]. In this thesis, we propose a new flow control scheme under the recently developed filtering theory [1, 2, 3]. The merit of employing the filtering-model is that by exchanging the two coefficients of the linear first-order filters on each node, any network node is able to estimate the degree of congestion in all its subsequent nodes by means of convoluting all its subsequent filters. Experiment results shows that the filter-based flow control can yield a more stable throughput than the conventional rate- or credit-based flow control (i.e., the variance of throughput is smaller).

The thesis is organized in the following fashion. In Chapter 2, we will introduce some theoretical thinking regarding to the filter-based flow control. In Chapter 3, the

model of the filter-based flow control will be introduced. We will then describe the simulation parameters in Chapter 4, and present the simulation results in Chapter 5. Remarks and conclusions will be given in Chapter 6.

Next, we give a brief introduction to flow control and filter theory.

1.1 Flow control

The primary reason for doing flow control is to reduce the flow of excess traffic [5]. There are several techniques that can be used to serve this purpose. For example, to use feedback mechanism to reduce the flow of excess traffic, to employ routing techniques to re-direct the flow of excess traffic, or to drop the packets on the floor to clean the buffer from excess traffic.

One can also classify the flow control scheme according to the basic architecture/topoligy of the control mechanism (cf. Figure 1.1). For example, source-to-destination flow control is known as end-to-end flow control, and is often governed by Transport Protocol window flow control. Node-to-node and/or switch flow control is known as hop-by-hop control, and is a layer-two responsibility. Entry-to-exit flow is layer-three flow control; it polices certain edges of the network from incoming traffic. The layers of control are interchangeable, e.g., the end-to-end flow control at layer-two or below of asynchronous transfer mode(ATM) can assume the end-to-end flow control responsibilities of layer-four.

Each layer controls different fragments in size. The comparison between controlling a few large segments and many small fragments is like comparing a big truck on a fast lane with a small and efficient car on a highway. Sequencing and keeping track of all the small fragments are often more difficult. Small fragments tend to be

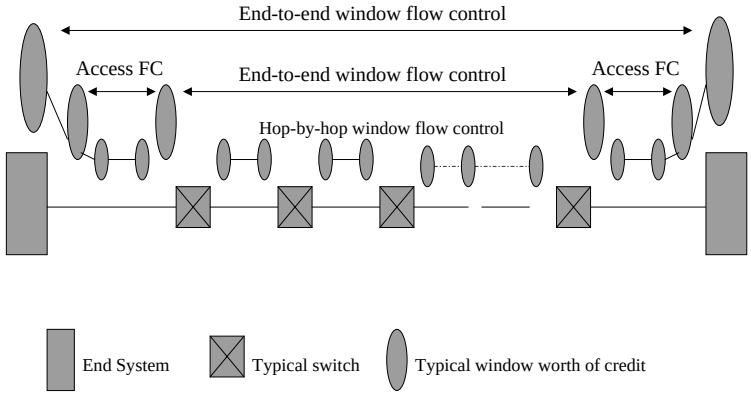


Figure 1.1: The model of flow control.

switched faster than large packets; thus, their inter-arrival time at the destination is shorter.

1.2 Filter theory on networks

The filtering theory is developed under the (min,+)-algebra [1, 2, 3], where the former replaces the usual addition operator, and the latter replaces the multiplication operator. For ease of understanding, the min and + operators are usually denoted by $\oplus$ and $\otimes$, respectively. As in the classical linear system theory, the filtering theory treats an arrival process $A(t)$ (resp. a departure process $B(t)$) as a signal, and a network as a system. In the filtering theory, a signal $f \equiv \{f(t), t = 0, 1, 2, ...\}$ is a nonnegative and increasing function. It is said to be larger than another, if all its values at any time t are larger. Two operations under linear systems will be used in this thesis:

1. *(Sum)*

$$(f \oplus g)(t) = \min[f(t), g(t)].$$

It now becomes the pointwise minimum of two signals.

2. *(Convolution)* The convolution of two signals is now:

$$(f \star g)(t) = min_{0 \leq s \leq t}[f(s) + g(t-s)].$$

These two operations are associative, commutative and distributive.

Chapter 2

The filter formula for known input and output

2.1 To find the filter expression under given input and output signals: frequency-domain approach

In the conventional linear time-invariant filtering theorem in signal processing, the filter expression for given input and output signals can be obtained through transformation techniques. Specifically, for input $x(t)$ and output $y(t)$, the filter $h(t)$ satisfies

$$y(t) = x(t) \star h(t) = \int_0^t x(s) \cdot h(t-s) ds \tag{2.1.1}$$

By taking the Fourier transform of both sides, we obtain

$$Y(\omega) = X(\omega) \cdot H(\omega) \tag{2.1.2}$$

Hence, we can take the inverse Fourier transform of $H(\omega) = Y(\omega)/X(\omega)$, and yield the time-domain expression for the filter.

In the basic theory of our flow control scheme, we assume that the accumulated arrival (input) and the accumulated output in each node is known; and we want

to find the best filter that can describe the system based on the $(\min, +)$ filtering theory. From the previous paragraph, it seems that we could also use *Transformation* technique in this problem.

We begin this research from an assumption and a simple lemma.

Definition 2.1.1. The accumulated input arrival $x(t)$ and the accumulated output departure $y(t)$ are non-decreasing, and equal zero for $t < 0$. The filter process $h(t)$ is zero for $t \leq 0$, and non-negative for $t > 0$. For convenience, we will use indicator function $\mathbf{1}\{\cdot\}$ in the definitions of $x(t)$, $y(t)$ and $h(t)$.

Lemma 2.1.1. *In the* $(\min, +)$ *algebra, we can express the output*

$$y(t) = \min_{0 \leq s \leq t} \{x(s) + h(t-s)\}$$

due to input $x(t)$ *and filter* $h(t)$ *as:*

$$y(t) = \min_{s \geq 0} \{x(s) + h(t-s)\}.$$

Proof.

$$\begin{aligned}
& \min_{s \geq 0} \{x(s) + h(t-s)\} \\
= \; & \min \left[\min_{0 \leq s \leq t} \{x(s) + h(t-s)\}, \min_{s > t} \{x(s) + h(t-s)\} \right] \\
= \; & \min \left[\min_{0 \leq s \leq t} \{x(s) + h(t-s)\}, \min_{s > t} \{x(s)\} \right], \; \textit{since } h(t-s) = 0 \; \forall \; t < s \\
= \; & \min \left[\min_{0 \leq s \leq t} \{x(s) + h(t-s)\}, x(t+1) \right], \; \textit{since } x(t) \textit{ is nondecreasing} \\
= \; & \min_{0 \leq s \leq t} \{x(s) + h(t-s)\},
\end{aligned}$$

where the last step follows since

$$\min_{0 \leq s \leq t} \{x(s) + h(t-s)\} \quad \leq \quad \{x(t) + h(0)\} \leq x(t+1).$$

□

Next, parallel to transformation used in $(+, \times)$ algebra, we define a transformation[1] under $(\min, +)$ algebra as:

$$F(\omega) = \min_{t \geq 0} [f(t) + \phi_1(\omega, t)] \Longleftrightarrow f(t) = \min_{\omega} [F(\omega) + \phi_2(\omega, t)]. \tag{2.1.3}$$

We then show in the next theorem that a transformation that results in $Y(\omega) = X(\omega) + H(\omega)$ does not exist.

Theorem 2.1.2. *$\phi_1(\omega, t)$ and $\phi_2(\omega, t)$ that satisfy (2.1.3) and*

$$y(t) = x(t) \star h(t) \Longleftrightarrow Y(\omega) = X(\omega) + H(\omega)$$

does not exist.

Proof. Suppose such transformation exist. Then

$$\begin{aligned} Y(\omega) &= \min_{t\geq 0}[y(t) + \phi_1(\omega, t)] \\ &= \min_{t\geq 0}\left(\min_{s\geq 0}[x(s) + h(t-s)] + \phi_1(\omega, t)\right) \\ &= \min_{t\geq 0}\left(\min_{s\geq 0}[x(s) + h(t-s) + \phi_1(\omega, t)]\right) \\ &= \min_{s\geq 0}\left(x(s) + \min_{t\geq 0}[h(t-s) + \phi_1(\omega, t)]\right) \end{aligned}$$

is equal to

$$X(\omega) + H(\omega) = \min_{s\geq 0}[x(s) + \phi_1(\omega, s)] + \min_{u\geq 0}[h(u) + \phi_1(\omega, u)].$$

The above equivalence hold for any legal $x(t)$ and $h(t)$. Therefore, we can take $x(t) = h(t) = 0$, and obtain

$$\min_{s\geq 0} \phi_1(\omega, s) = 0.$$

Now take $x(t) = 1 \cdot \mathbf{1}\{t \geq 0\}$ and $h(t) = (t+1) \cdot \mathbf{1}\{t > 0\}$. Then

$$\begin{aligned} & \min_{s\geq 0}[x(s) + \phi_1(\omega, s)] + \min_{u\geq 0}[h(u) + \phi_1(\omega, u)] \\ = \; & \min_{s\geq 0}[1 + \phi_1(\omega, s)] + \min\left\{\phi_1(\omega, 0), \min_{u\geq 1}[u + 1 + \phi_1(\omega, u)]\right\} \\ = \; & 1 + \min\left\{\phi_1(\omega, 0), \min_{u\geq 1}[u + 1 + \phi_1(\omega, u)]\right\} \\ \geq \; & 1 + \min\left\{\phi_1(\omega, 0), 1\right\} \end{aligned}$$

---

[1] Conventional transformation formula often has the form:

$$F(\omega) = \sum_t [f(t)\phi_1(\omega, t)] \Longleftrightarrow f(t) = \sum_\omega [F(\omega)\phi_2(\omega, t)].$$

Here we replace summation by minimum operation; and product by plus operation.

and

$$\begin{aligned} & \min_{s\geq 0}\left(x(s)+\min_{t\geq 0}[h(t-s)+\phi_1(\omega,t)]\right) \\ = & \min_{s\geq 0}\left(1+\min_{t\geq 0}[h(t-s)+\phi_1(\omega,t)]\right) \\ = & 1+\min_{s\geq 0}\min_{t\geq 0}[h(t-s)+\phi_1(\omega,t)] \\ = & 1+\min_{s\geq 0}\phi_1(\omega,s)=1. \end{aligned}$$

Therefore, $\phi_1(\omega,0)=0$.

Now $\phi_1(\omega,0)=0$ and $\min_{s\geq 0}\phi_1(\omega,s)=0$, together with the non-decreasing property of $x(t)$, imply that

$$X(\omega) = \min_{s\geq 0}[x(s)+\phi_1(\omega,s)]=x(0)$$

and

$$Y(\omega) = \min_{s\geq 0}\left(x(s)+\min_{t\geq 0}[h(t-s)+\phi_1(\omega,t)]\right)=x(0).$$

Thus, for any legal $h(t)$, $H(\omega)=Y(\omega)-X(\omega)=0$. Accordingly, no $\phi_2(\omega,t)$ can reconstruct $h(t)$ based on the degenerate $H(\omega)=Y(\omega)-X(\omega)=0$, which holds for any legal $h(t)$. □

Remark: In the conventional $(+,\times)$ algebra, both operators possess "inverse" operations; however, in the $(\min,+)$ algebra, the min operator does not have inverse operation. This may be the key reason that transformation techniques fail to apply in such case.

Due to the failure of transformation technique, it seems that we can only resolve $h(t)$ for given input $x(t)$ and output $y(t)$ directly from the time-domain technique. Some discussion along this line will be given in the next section.

2.2 To find the filter expression under given input and output signals: time-domain approach

2.2.1 Sufficient condition for the nonexistence of filter formula for CBR arrivals

Again, the output departure $y(t)$ and input arrival $x(t)$ satisfy

$$y(t) = \min_{0 \le s \le t} \{x(s) + h(t-s)\}. \tag{2.2.1}$$

The previous section hints that the possibility of finding the best $h(t)$ in terms of transformation technique is remote. However, to determine the best $h(t)$ based on known $x(t)$ and $y(t)$ may not be possible for general $x(t)$ and $y(t)$. In order to gain some insight on this problem, we here assume that $x(t)$ is a constant-bit-rate(CBR) process. Based on the assumption, we will show that $h(t)$ exists only when the service rate $y(t)$ is smaller than arrival rate $x(t)$ for any t.

Theorem 2.2.1. *Assume $x(t)$ is CBR. If for some t, the service rate of $y(t)$ is greater than the arrival rate of $x(t)$, i.e.,*

$$(\exists\ t)\ y(t) - y(t-1) > x(t) - x(t-1),$$

then there is no $h(t)$ satisfying (2.2.1).

Proof. By definition,

$$y(t-1) = \min_{0 \le s \le t-1} [x(s) + h((t-1) - s)].$$

Hence, there exists $\hat{s} \in [0, t-1]$ such that

$$y(t-1) = x(\hat{s}) + h((t-1) - \hat{s}).$$

Hence,

$$\begin{aligned} y(t) &\le x(\hat{s}+1) + h(t - (\hat{s}+1)) \\ &= x(\hat{s}+1) - x(\hat{s}) + y(t-1), \end{aligned}$$

which implies

$$y(t) - y(t-1) \le x(\hat{s}+1) - x(\hat{s}) = x(t) - x(t-1).$$

□

Corollary 2.2.2. *Assume $x(t)$ is CBR. If*

$$(\exists\ t\ and\ s)\ \frac{y(t)-y(t-s)}{s} > \frac{x(t)-x(t-s)}{s},$$

then there is no $h(t)$ satisfying (2.2.1).

Proof. This is a straightforward extension of the previous theorem. □

Remark:

- It is easy to verify that if the arrival rate of $x(t)$ is non-decreasing, the above theorem holds as well. Yet, the case of increasing-arrival-rate input process may be of secondary interest in practice.

- If the input curve $x(t)$ is not CBR, the above theorem is not necessarily valid. An example is provided below. Let $x(t)$, $y(t)$ and $h(t)$ be as shown in the next table.

t	0	1	2	3	4	5	6
$x(t)$	0	3	3	3	6	6	6
$y(t)$	0	1	2	3	4	5	6
$h(t)$	0	1	2	3	4	5	6

  Then we notice that $y(3)-y(2) > x(3)-x(2)$; however, $h(t)$ exists.

- A good example for which $y(t) \le x(t)$ for all t, but the service rate is transiently larger than the arrival rate of $x(t)$, is a non-FCFS system. Actually, we doubt that the filtering theory can be applied for such system.

  Another example that can be encountered in practice is a system with double buffers, in which the new arrivals feed into one buffer at the time the server depletes the other buffer.

2.2.2 Nonuniqueness of $h(t)$ for known $x(t)$ and $y(t)$

The previous subsection gives a sufficient condition under which $h(t)$ does not exist. In fact, even if $h(t)$ exists, it may not be *unique*. An example is given below. Take $x(t) = (2t-1)\cdot \mathbf{1}\{t \geq 1\}$, and $h(t) = t^2 \cdot \mathbf{1}\{t \geq 0\}$. Then

$$\begin{aligned}
y(t) &= x(t) \star h(t) \\
&= \min_{0\leq s\leq t}\{(2s-1)\cdot \mathbf{1}\{s\geq 1\} + (t-s)^2 \cdot \mathbf{1}\{s \leq t\}\} \\
&= \min\left[t^2, \min_{1\leq s\leq t}\{2s-1+(t-s)^2\}\right] \\
&= \min\left[t^2, \min_{1\leq s\leq t}\{2s-1+t^2-2st+s^2\}\right] \\
&= \min\left[t^2, \min_{1\leq s\leq t}\{[s-(t-1)]^2+2t-2\}\right] \\
&= \mathbf{1}\{t\geq 1\} + (2t-3)\cdot \mathbf{1}\{t\geq 2\}.
\end{aligned}$$

However, when $h(t)$ is replaced by $(2t-1)\cdot \mathbf{1}\{t \geq 1\}$, we can obtain the same output, i.e.,

$$\begin{aligned}
y(t) &= x(t) \star h(t) \\
&= \min_{0\leq s\leq t}\{(2s-1)\cdot \mathbf{1}\{s\geq 1\} + [2(t-s)-1] \cdot \mathbf{1}\{s \leq t-1\}\} \\
&= \mathbf{1}\{t\geq 1\} + (2t-3)\cdot \mathbf{1}\{t\geq 2\}.
\end{aligned}$$

Hence, different $h(t)$ can induce the same $y(t)$ for the same $x(t)$. Note that the result in this subsection is already mentioned in [3].

2.2.3 Condition under which $h(t)$ has unique solution

In this subsection, we distinguish the cases of solution $h(t)$ in terms of the relations between $x(t)$ and $y(t)$.

Lemma 2.2.3. $y(t) = \min_{0\le s\le t}[x(s) + h(t-s)] \le \min\{x(t), x(0) + h(t)\}$.

Proof. This follows directly from its definition. □

Theorem 2.2.4. *Assume that* $x(0) = y(0) = 0$.

1. *If* $y(t) - y(t-k) < x(k)$ *for all* $t \ge 0$ *and all* $0 < k \le t$, *then* $h(t)$ *exist, and is unique. Moreover,* $h(t)$ *is equal to* $y(t)$ *for every* t.

2. *If* $y(t)-y(t-k) < x(k)$ *for all* $0 \le t \ne \hat{t}$ *and all* $0 < k \le t$, *but* $y(\hat{t})-y(\hat{t}-\hat{k}) = x(\hat{k})$ *for some* $0 < \hat{k} \le \hat{t}$, *then* $h(t)$ *exists, but is not unique.*

3. *If* $y(t)-y(t-k) < x(k)$ *for all* $0 \le t \ne \hat{t}$ *and all* $0 < k \le t$, *but* $y(\hat{t})-y(\hat{t}-\hat{k}) > x(\hat{k})$ *for some* $0 < \hat{k} \le \hat{t}$, *then* $h(t)$ *does not exist, provided that* $\hat{t} > 1$.

Proof. **1.** A legal filter satisfies that $h(0) = 0$, which validate the relation $y(0) = \min_{0\le s\le 0}[x(s) + h(t-s)] = x(0) + h(0) = x(0)$. Next,

$$\begin{aligned} y(1) &= \min\{x(0) + h(1), x(1) + h(0)\} \\ &= \min\{x(0) + h(1), x(1)\} \\ &= \min\{h(1), x(1)\} = h(1), \end{aligned}$$

where the last step follows from $y(1) - y(0) < x(1)$. Hence, $h(1) = y(1)$. Similarly,

$$\begin{aligned} y(2) &= \min\{x(0) + h(2), x(1) + h(1), x(2) + h(0)\} \\ &= \min\{h(2), x(1) + y(1), x(2)\} = h(2), \end{aligned}$$

which gives $h(2) = y(2)$. (You may already notice that $y(t) - y(t-k) < x(k)$ for $0 < k \le t$ and $h(u) = y(u)$ for $0 \le u < t$ imply that $h(t) = y(t)$.)

Now, we finish the proof by induction. Suppose $h(u) = y(u)$ for $0 \le u < t$. Then

$$y(t) = \min\{h(t), x(1) + h(t-1), x(2) + h(t-2), \ldots, x(t) + h(0)\} = h(t).$$

So the solution $h(t)$ exists, and is uniquely equal to $y(t)$.

2. Following the proof of 1), we have $h(u) = y(u)$ for $0 \le u < \hat{t}$. Now $y(\hat{t})-y(\hat{t}-\hat{k}) = x(\hat{k})$ implies

$$\begin{aligned} y(\hat{t}) &= y(\hat{t} - \hat{k}) + x(\hat{k}) \\ &= h(\hat{t} - \hat{k}) + x(\hat{k}) \le h(\hat{t}). \end{aligned}$$

Hence, we can let $h(\hat{t})$ be equal to $y(\hat{t}) + m$ for $m > 0$ (to satisfy the relation $y(\hat{t}) = \min_{0 \le s \le \hat{t}}[x(s) + h(\hat{t} - s)]$).

Now for $t = \hat{t} + 1$, we have

$$\begin{aligned} y(\hat{t}+1) &= \min\{h(\hat{t}+1), x(1) + h(\hat{t}), x(2) + h(\hat{t}-1), \ldots, x(\hat{t}+1)\} \\ &= \min\{h(\hat{t}+1), x(1) + y(\hat{t}) + m, x(2) + y(\hat{t}-1), \ldots, x(\hat{t}+1)\} \\ &= h(\hat{t}+1). \end{aligned}$$

Again, we can continue the above derivation to obtain $h(u) = y(u)$ for $u > \hat{t}$.

3. Following the proof of 1), we have $h(u) = y(u)$ for $0 \le u < \hat{t}$. It is obvious that $y(\hat{t}) - y(\hat{t} - \hat{k}) > x(\hat{k})$ contradicts to

$$\begin{aligned} y(\hat{t}) &= \min\{h(\hat{t}), x(1) + h(\hat{t}-1), x(2) + h(\hat{t}-2), \ldots, x(\hat{t})\} \\ &= \min\{h(\hat{t}), x(1) + y(\hat{t}-1), x(2) + y(\hat{t}-2), \ldots, x(\hat{t})\} \\ &\le x(\hat{k}) + y(\hat{t} - \hat{k}). \end{aligned}$$

□

Remark: The condition that $\hat{t} > 1$ is essential, which can be confirmed by the next example for which $y(t) - y(t-k) > x(k)$ for $t = k = 1$ but $h(t)$ exists.

t	0	1	2	3	4	5	6
$x(t)$	0	1	4	5	7	8	8
$y(t)$	0	1	2	4	6	8	8
$h(t)$	0	1	3	5	7	8	8

We close the chapter by remarking that the discussion in this chapter indicates that to find the exact solution of $h(t)$ for known $x(t)$ and $y(t)$ may not be feasible. Hence, in the next chapter, we will turn to determine the "best" $h(t)$ that can best described the output $y(t)$ in term of absolute-error criterion. The "best" $h(t)$ will be used as the basis of our filter-based flow control.

Chapter 3

Filter-based flow control

3.1 Preliminaries

To simplify the flow control framework, we adopt the simplest form of a filter $h(t)$, i.e., $h(t) = c \cdot t + d$. Our work is to determine the best c and d such that the convolved departure $\hat{y}(t)$ is very close to the true (measured) departure $y(t)$. We will then use the estimated c and d for each node to form the estimated overall filter representation of the end-to-end flow by a simple first-order convolution. Based on this, we can somehow characterize the degree of smoothness of the entire traffic flow, and take necessary action when congestion happens.

3.2 Approximation of $h(t)$

In Figure 3.1, we show the diagram for determining the best $h(t)$. The distance measure $\|\hat{y}(t) - y(t)\|$ that we choose is the absolute-error criterion; however, the exact solution for c and d that minimize $\|\hat{y}(t) - y(t)\|$ may not be easy to obtain. Instead, we minimize an upper bound of $\|\hat{y}(t) - y(t)\|$.

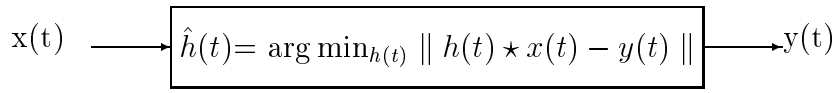


Figure 3.1: Filter estimate model.

Let the decision window be denoted by N. Then

$$\begin{aligned}\sum_{t=1}^{N} |\hat{y}(t) - y(t)| &= \sum_{t=1}^{N} \left| \min_{0\le s\le t} [x(s) + c(t-s) + d] - y(t) \right| \\ &= \sum_{t=1}^{N} \left| \min_{0\le s\le t} [x(s) - cs] + ct + d - y(t) \right| \\ &= |p_1(c) + d| + |p_2(c) + d| + \cdots + |p_N(c) + d|,\end{aligned}$$

where $p_i(c) = \min_{0\le s\le i}[x(s) - cs] + ci - y(i)$. The optimal d is therefore the median[1] of the sequence $-p_1(c)$, $-p_2(c)$, ..., $-p_N(c)$.

The algorithm to find the best c and d is summarized below.

1. Pick up a c within a reasonable region (i.e., between the maximum rate and the minimum rate).

2. Substitute c into $\{p_i(c)\}_{i=1}^{N}$.

3. The best d is the median of $\{p_i(c)\}_{i=1}^{N}$.

[1]For a sorted sequence $q_1 \le q_2 \le \ldots \le q_{2k}$,

$$|q_1 - d| + |q_2 - d| + \cdots + |q_{2k} - d|$$

is minimized by $d \in [q_k, q_{k+1}]$. This can be proved as follows. For any $\hat{d} \in [q_i, q_{i+1}]$ with $i < k$,

$$\begin{aligned}&(|q_1 - d| + |q_2 - d| + \cdots + |q_{2k} - d|) - \left(|q_1 - \hat{d}| + |q_2 - \hat{d}| + \cdots + |q_{2k} - \hat{d}|\right) \\ <\ & i(d - \hat{d}) + (k - i)(d - \hat{d}) - k(d - \hat{d}) = 0.\end{aligned}$$

The above relation holds also for any $\hat{d} \in [q_i, q_{i+1}]$ with $i > k$. Consequently, the selected d minimizes the targeted quantity.

4. Calculate the resultant absolute error, and compare it with the current minimum value obtained so far. If the new absolute error is smaller, replace the current minimum value by the new value, and record the current c and d.

5. Repeat steps 2–4 until all the possible choices are exhausted.

3.3 Regulation criterion

For a network node, an almost-full buffer usually implies a crowded traffic. Question is how we know the "crowdedness" by means of the estimated c and d.

¿From the filtering theory, $y(t)$ is actually upper-bounded by $h(t)$. If $h(t)$ is a good approximate of $y(t)$, then $x(N) - cN - d$ can be used to approximate the buffer length or crowdedness for the node at the end of each measure window.

Now if the network node j also knows the estimated c_u and d_u $(j < u < J)$ of its subsequent nodes, it can also estimated the crowdedness of the entire subsequent traffic flow by convolving the filters, i.e., $c_{j-J} = \min\{c_j, c_{j+1}, \ldots, c_J\}$ and $d_{j-J} = d_j + d_{j+1} + \cdots + d_J$. We can then use the difference between $x(N)$ (the input arrival of node j at the end of the measure window) and $c_{j-J} \cdot N + d_{j-J}$ as an estimate of the crowdedness of the entire subsequent network flow.

The basic regulate criterion is: if $x(N) - cN - d$ is small, the network is not crowed; so we can increase the departure rate of the previous node. On the other hand, if $x(N) - cN - d$ is large, which indicates the network is crowed, then the departure rate of the previous node must decrease. The threshold for characterizing $x(N) - cN - d$, as well as the step size for departure rate increment/decrement, will be decided through simulations.

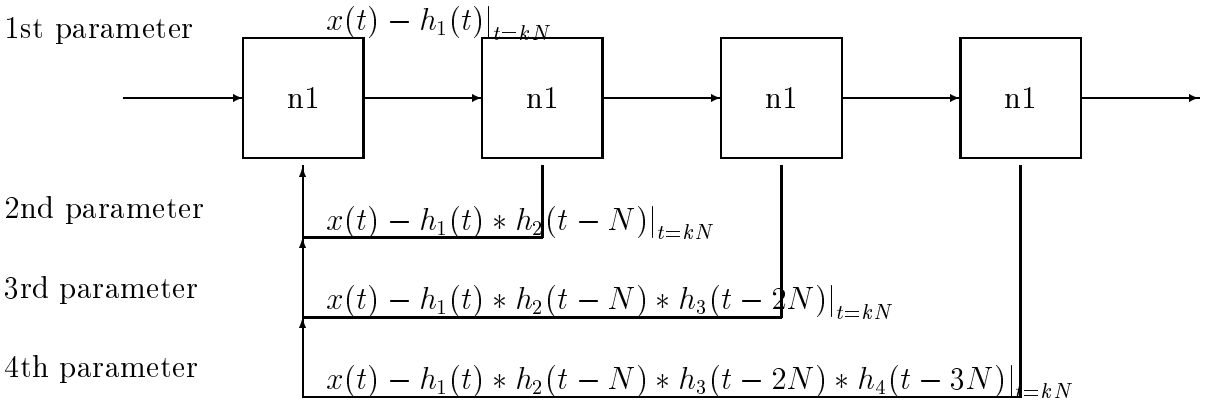


Figure 3.2: The current and subsequent filter estimates.

3.3.1 Reliability of the subsequent filter estimates

The estimate filter coefficient pair (i.e., c and d) is exchanged between adjacent network node periodically. The period is exactly the measure window N. As shown in Figure 3.2, after three measure window period, the first node will have four estimate filter coefficient pair. However, some of them are obtained some moment ago; and hence, may not exactly characterize the current status of the subsequence nodes. Thus, we set a weight on their degree of "trustness" as shown in Table 3.1. The decision of the first node is therefore based on the weighted sum of the four parameters in Figure 3.2. Same scenario applies to nodes 2, 3 and 4.

Table 3.1: Weight of parameters.

weight	1st para.	2nd para.	3rd para.	4th para.
Node1	0.5	0.25	0.15	0.1
Node2	N/A	0.5	0.3	0.2
Node3	N/A	N/A	0.6	0.4
Node4	N/A	N/A	N/A	1

3.3.2 Feedback control and feedforward control

In this subsection, we introduce how we regulate the traffic flow based on the weighted decision quantity (described in the previous subsection).

First of all, we will set an appropriate threshold, and then, use the threshold to characterize the status of the current traffic flow as shown in Table 3.2. There are four status categories: *smooth*, *normal*, *crowded* and *very crowded*. The algorithm to regulate the speed is as follows.

1. We first determine if the network node is in a *feedforward regulating* condition: i.e.,

   - its arrival subtracting its estimated filter evaluated at the end of the measured window is categorized as *very crowded*;
   - its departure subtracting its convolved subsequent filter, however, is categorized as *smooth*.

   If the answer is affirmative, the network node will increase its departure rate by $4 \times stepsize$.

2. On the other hand, if the node is in *smooth* status, and its subsequent nodes are in *very crowed* status, then the network node will perform another feedforward regulation by decreasing its departure rate by $4 \times stepsize$.

3. If no feedforward regulation is performed, the network node will proceed to do the feedback regulation (i.e, only one of the feedforward and feedback regulation will be performed).

Table 3.2: The status categories and feedback regulation control.

Network status	threshold region	relative change
smooth	$-\infty \sim 0$	$+1 \times stepsize$
normal	$0 \sim 1 \times threshold$	no change
crowded	$1 \times threshold \sim 2 \times threshold$	$-1 \times stepsize$
very crowded	$2 \times threshold \sim \infty$	$-2 \times stepsize$

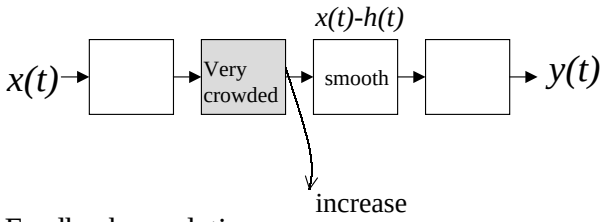
The network node will ask its previous node to increase/decrease departure rate according to its status and Table 3.2.

For ease of understanding, examples of feedback and feedforward regulation are given in Figure 3.3.

3.3.3 History information

For convenience, we name the filter estimates of the subsequence nodes *the history information*, since they are the estimate some time ago. In order to realize the effect of the history information on system performance, two kinds of experiments will be performed for filter-based flow control: one is with history information as described above, and the other one is without history information where only feedback regulation is operated.

– Feedforward regulation



– Feedback regulation

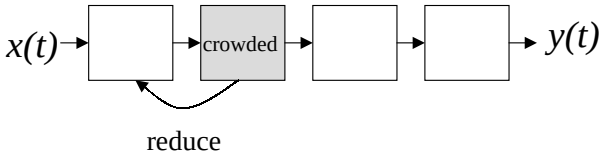


Figure 3.3: Feedback and feedforward model.

Chapter 4

Simulations

4.1 System model

We will give the detail of our simulation model in this section. As shown in Figure 4.1, there are four network nodes. Each node consists of a FCFS queue (buffer) with size 200 packets. All packets are assumed to have fixed size.

There are two input arrivals for each node. The first one represents the controlled traffic flow with initial rate 5 packet per unit time (PPUT). Its rate is ranged between 0.5 PPUT and 10 PPUT, and is subject to change due to the flow control mechanism. We will measure its throughput and packet loss rate as the performance index of the selected flow control scheme.

The second arrival represents the aggregated arrivals of all other traffics. It is modeled as a two-state bursty Poisson process (cf. Figure 4.2) with mean rate on each state being 40 PPUT and 50 PPUT respectively. The overall mean rate of the bursty Poisson is 45 PPUT.

As shown in Figure 4.2, the two-state bursty Poisson process is one with two states, where transition between states perform like a first-order Markov. We define the *bursty period* as the time period for which the two-state bursty Poisson might

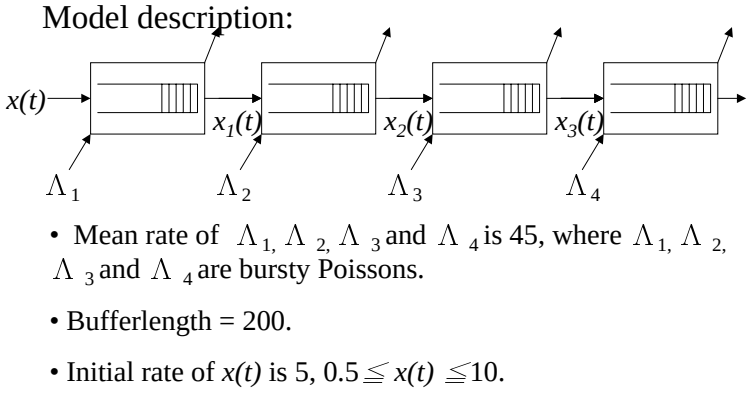


Figure 4.1: The system model.

change states. As a result, a small bursty period hints a fast changing exotic traffic, while a large bursty period indicates a slow changing one.

There are several factors that can influence the system performance of our proposed filter-based flow control scheme. The first one is the threshold which our decision is based on. The second one is the regulated step size of the departure rate when the network node decides to regulate the rate. The final factor is the regulation period, which is exactly the size of the measure window N.

We will simulate the system performance based on different values of the ratio, *regulation period/bursty period*.

Our simulations will be performed in two steps. In the first step, we will find the best threshold and step size that results in the largest throughput for each flow control schemes, including rate-based, credit-based and filter-based flow controls, subject to a fixed upper bound on the packet loss rate. Secondly, we will use these thresholds and step sizes to evaluate their performance by varying system parameters, such as *regulation period/bursty period* and mean rate of the exotic traffic.

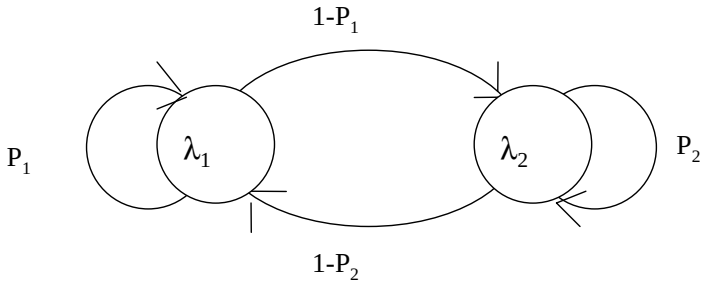


Figure 4.2: The bursty Poisson.

4.2 Determination of the thresholds and step sizes

Fix the upper bound on the packet loss rate at 2%. We then simulate to find the thresholds and step sizes that can yield the largest throughput.

As shown in Figures 4.3 and 4.4 (also Tables 4.1 and 4.2), the threshold and step size, which yield the maximum throughput subject to the pack loss rate bound, 2%, are 8 and 1, respectively.

For the rate-based flow control, there are four parameters to be decided: high threshold, low threshold, increasing ratio, and decreasing ratio. It is not easy to find the optimal values of these four parameters at the same time. So we first fix the high threshold and low threshold at 190 and 150 respectively, and find the best step sizes that result in the largest throughput under 2% packet loss rate. Then use the resultant step sizes to determine the best thresholds that yield the maximum throughput under 2% packet loss rate.

As shown in Figures 4.5 and 4.6 (also Tables 4.3 and 4.4), the acceptable increasing ratio and decreasing ratio are 0.1 and 0.2 respectively. Note that for rate-based flow control scheme, the rate increases linearly but decreases exponentially. For details,

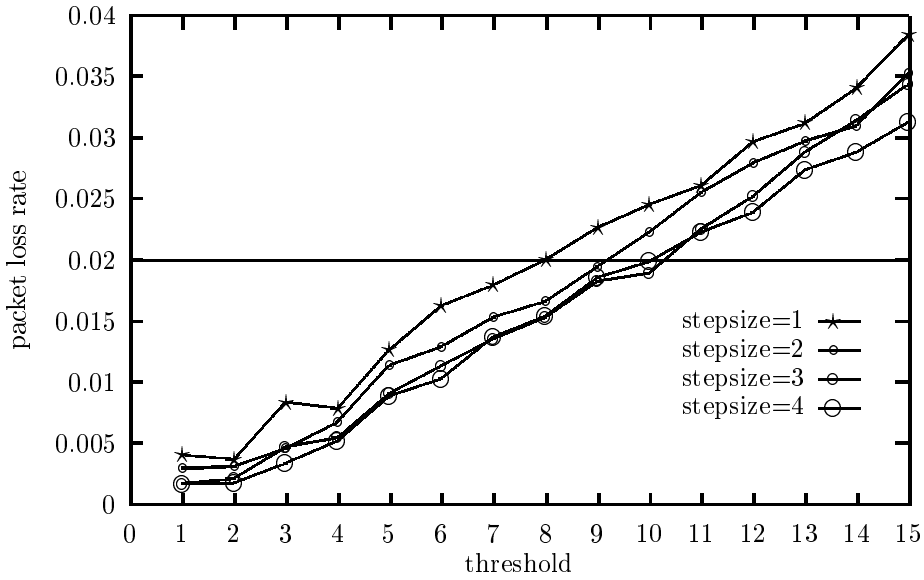


Figure 4.3: Packet loss rate of filter-based flow control for various thresholds and step sizes.

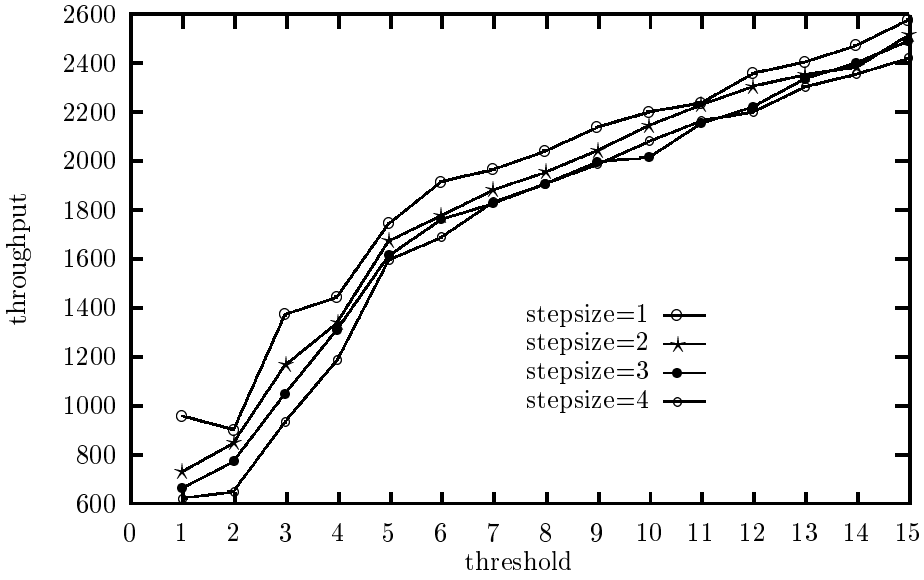


Figure 4.4: Throughput of filter-based flow control for various thresholds and step sizes.

Table 4.1: Packet loss rate of filter-based flow control for various thresholds and step sizes. The column indexes the step size, which varies from 1 to 4; and the row indexes the thresholds that ranges from 1 to 15.

	1	2	3	4
1	0.0041	0.0030	0.0017	0.0017
2	0.0037	0.0032	0.0022	0.0018
3	0.0084	0.0046	0.0047	0.0034
4	0.0079	0.0067	0.0055	0.0052
5	0.0127	0.0114	0.0091	0.0089
6	0.0163	0.0129	0.0114	0.0103
7	0.0179	0.0153	0.0135	0.0137
8	0.0200	0.0167	0.0154	0.0154
9	0.0227	0.0195	0.0182	0.0186
10	0.0245	0.0223	0.0190	0.0199
11	0.0261	0.0256	0.0226	0.0223
12	0.0296	0.0279	0.0252	0.0239
13	0.0312	0.0297	0.0289	0.0274
14	0.0341	0.0309	0.0314	0.0288
15	0.0384	0.0352	0.0343	0.0313

Table 4.2: Throughput of filter-based flow control for various thresholds and step sizes. The column indexes the step size, which varies from 1 to 4; and the row indexes the thresholds that ranges from 1 to 15.

	1	2	3	4
1	958	730	662	621
2	902	848	773	650
3	1374	1169	1050	935
4	1445	1342	1312	1186
5	1747	1675	1615	1598
6	1917	1779	1763	1689
7	1967	1883	1828	1834
8	2042	1956	1909	1910
9	2138	2045	1997	1987
10	2201	2147	2017	2081
11	2239	2231	2154	2165
12	2360	2308	2223	2201
13	2406	2353	2337	2305
14	2475	2385	2404	2357
15	2580	2515	2492	2421

reader can refer to Appendix A. Based on the selected ratios, we found that the best high and low thresholds are 182 and 154 respectively according to Figures 4.7 and 4.8 (also Tables 4.5 and 4.6).

For the credit-based flow control, there is only one factor to determine, which is the multiplicative coefficient (cf. Appendix B). The number of packets that allow to enter is equal to the multiplicative coefficient times the vacant buffer length. When the coefficient is large, there will have more packets allowed to enter. From Figures 4.9 and 4.10 (also Table 4.7), we obtain that 1.2 is the best multiplicative coefficient that achieves the maximum throughput under 2% packet loss rate.

4.3 Comparison among rate-, credit- and filter-based flow controls

Now we can use the parameters obtained in the previous section to perform simulations on the three flow control schemes. Comparisons among the three will be based on our simulation results.

200 experiments for each selected regulation period and bursty period will be performed. Based on the 200 experiment results, three performance indices will be given: packet loss rate, throughput, and variance of throughput. Note that a small throughput variance hints that the flow control scheme is more robust against variability of traffics.

At first, we let the regulation period and the bursty period be 10 and 20 respectively. The overall observatory period is 400. The exotic traffic mean rate ranges from 41 to 49.

Some observations are made from Tables 4.8, 4.9, 4.10 and Figures 4.11, 4.12,

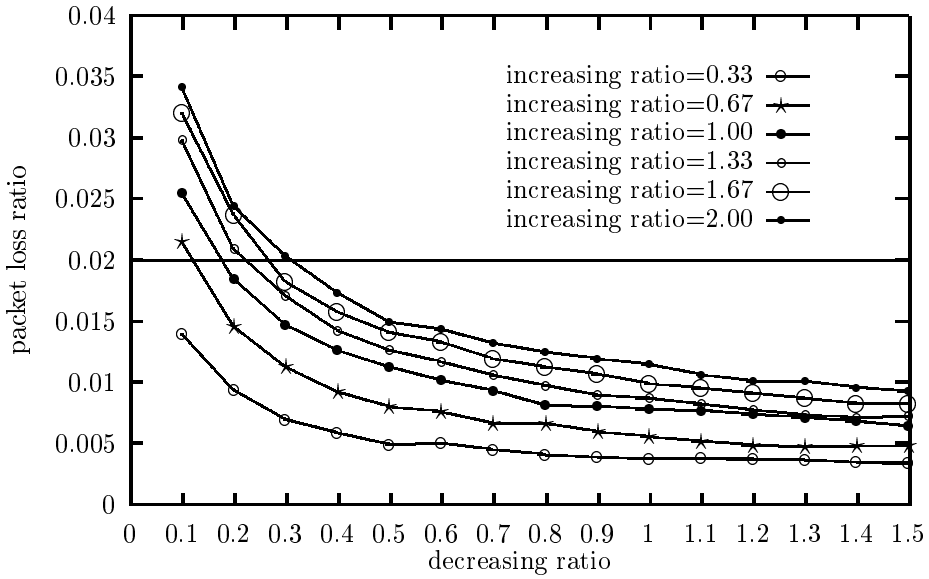


Figure 4.5: Packet loss rate of rate-based flow control under fixed thresholds and variable step sizes.

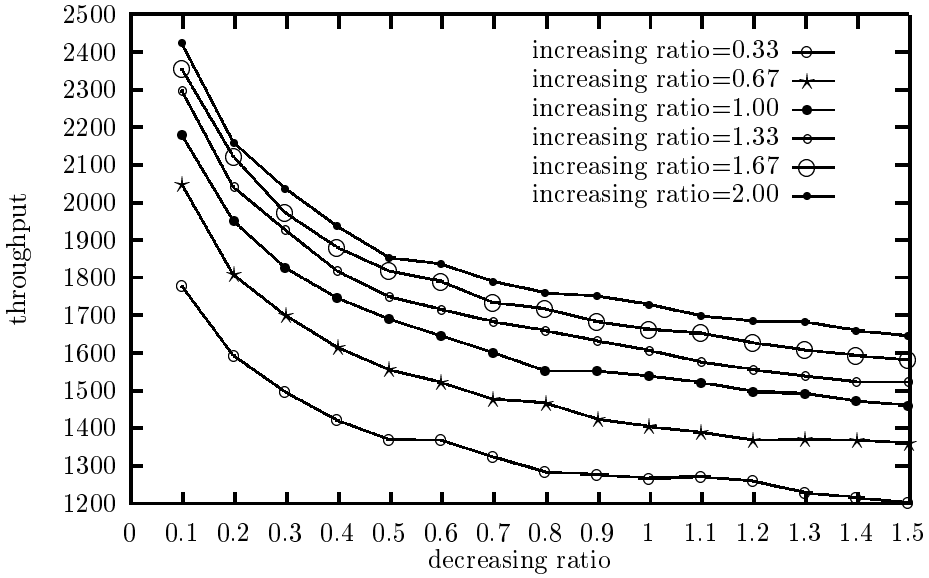


Figure 4.6: Throughput of rate-based flow control under fixed thresholds and variable step sizes.

Table 4.3: Packet loss rate of rate-based flow control under fixed thresholds and variable step sizes. The column indexes the increasing ratio, and the row indexes the decreasing ratio.

	0.33	0.67	1.00	1.33	1.67	2.00
0.1	0.0140	0.0215	0.0255	0.0298	0.0320	0.0341
0.2	0.0094	0.0146	0.0184	0.0209	0.0237	0.0244
0.3	0.0070	0.0113	0.0147	0.0171	0.0182	0.0203
0.4	0.0059	0.0092	0.0126	0.0143	0.0158	0.0173
0.5	0.0049	0.0080	0.0112	0.0126	0.0141	0.0149
0.6	0.0050	0.0076	0.0102	0.0117	0.0133	0.0143
0.7	0.0045	0.0066	0.0093	0.0106	0.0119	0.0132
0.8	0.0041	0.0066	0.0081	0.0098	0.0113	0.0125
0.9	0.0039	0.0059	0.0081	0.0090	0.0107	0.0119
1	0.0038	0.0055	0.0078	0.0087	0.0099	0.0115
1.1	0.0038	0.0052	0.0077	0.0082	0.0095	0.0106
1.2	0.0037	0.0049	0.0074	0.0077	0.0091	0.0101
1.3	0.0037	0.0048	0.0071	0.0074	0.0087	0.0101
1.4	0.0035	0.0048	0.0068	0.0071	0.0083	0.0096
1.5	0.0034	0.0048	0.0065	0.0072	0.0083	0.0093

Table 4.4: Throughput of rate-based flow control under fixed thresholds and variable step sizes. The column indexes the increasing ratio, and the row indexes the decreasing ratio.

	0.33	0.67	1.00	1.33	1.67	2.00
0.1	1778	2049	2180	2298	2356	2424
0.2	1593	1807	1950	2040	2122	2159
0.3	1496	1701	1827	1926	1973	2037
0.4	1422	1616	1746	1817	1880	1937
0.5	1370	1556	1690	1750	1817	1853
0.6	1368	1522	1645	1716	1791	1836
0.7	1324	1477	1601	1683	1735	1791
0.8	1283	1467	1552	1659	1717	1760
0.9	1276	1424	1552	1632	1683	1751
1	1266	1405	1539	1607	1663	1730
1.1	1271	1389	1521	1576	1653	1699
1.2	1259	1368	1497	1556	1627	1685
1.3	1229	1370	1492	1538	1609	1684
1.4	1215	1369	1473	1524	1593	1660
1.5	1201	1360	1460	1524	1581	1645

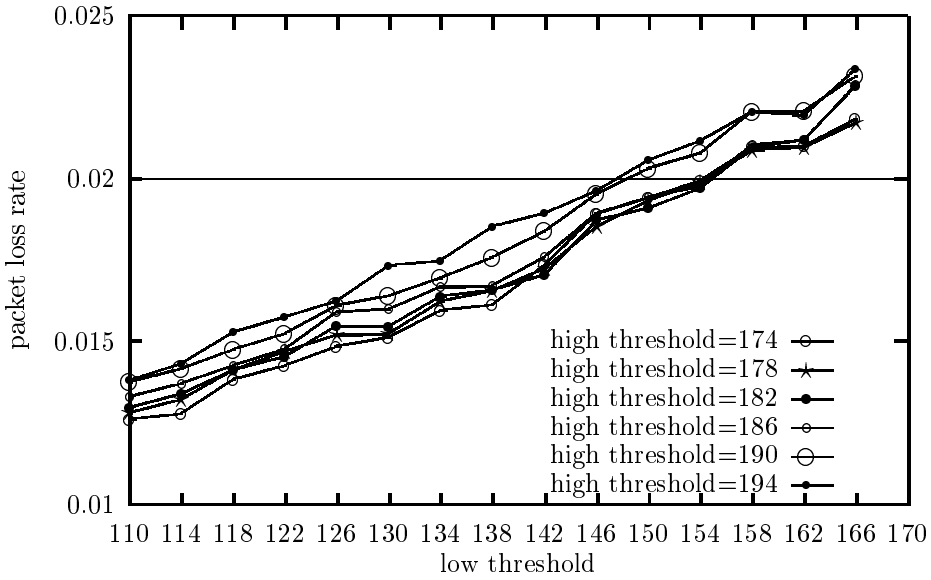


Figure 4.7: Packet loss rate of rate-based flow control under fixed ratios and various thresholds.

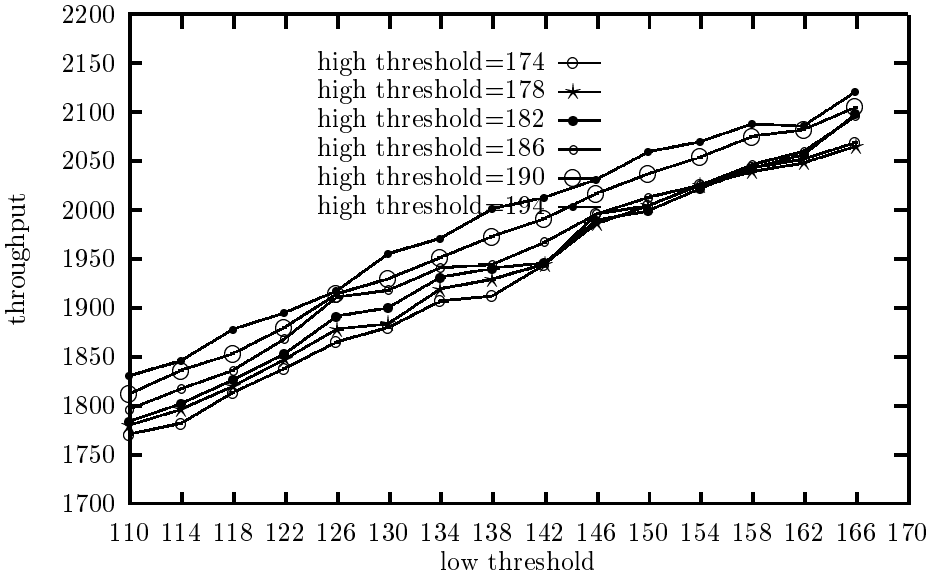


Figure 4.8: Throughput of rate-based flow control under fixed ratios and various thresholds.

Table 4.5: Packet loss rate of rate-based flow control under fixed ratios and various thresholds. The column indexes the low threshold, and the row indexes the high threshold.

	174	178	182	186	190	194
166	0.0218	0.0217	0.0228	0.0229	0.0231	0.0234
162	0.0210	0.0210	0.0212	0.0212	0.0221	0.0219
158	0.0210	0.0209	0.0209	0.0210	0.0221	0.0220
154	0.0199	0.0199	0.0197	0.0198	0.0208	0.0211
150	0.0194	0.0193	0.0191	0.0194	0.0203	0.0206
146	0.0189	0.0185	0.0187	0.0189	0.0195	0.0196
142	0.0173	0.0172	0.0170	0.0176	0.0184	0.0189
138	0.0161	0.0166	0.0166	0.0167	0.0176	0.0185
134	0.0160	0.0162	0.0164	0.0167	0.0169	0.0175
130	0.0151	0.0152	0.0155	0.0160	0.0164	0.0173
126	0.0149	0.0152	0.0155	0.0159	0.0161	0.0162
122	0.0143	0.0147	0.0145	0.0148	0.0152	0.0158
118	0.0138	0.0141	0.0141	0.0143	0.0148	0.0153
114	0.0128	0.0132	0.0134	0.0137	0.0142	0.0143
110	0.0126	0.0128	0.0130	0.0133	0.0138	0.0138

Table 4.6: Throughput of rate-based flow control under fixed ratios and various thresholds. The column indexes the low threshold, and the row indexes the high threshold.

	174	178	182	186	190	194
166	2069	2065	2098	2096	2105	2121
162	2052	2048	2057	2060	2082	2086
158	2043	2039	2042	2046	2075	2088
154	2026	2025	2022	2025	2054	2070
150	2004	2004	1999	2013	2037	2059
146	1996	1986	1989	1996	2017	2031
142	1944	1945	1946	1967	1991	2013
138	1912	1929	1940	1944	1973	2001
134	1907	1919	1931	1941	1951	1971
130	1880	1884	1900	1918	1930	1955
126	1865	1878	1891	1911	1914	1917
122	1838	1847	1853	1868	1880	1895
118	1813	1820	1826	1836	1853	1878
114	1782	1796	1802	1817	1836	1846
110	1771	1780	1784	1796	1812	1830

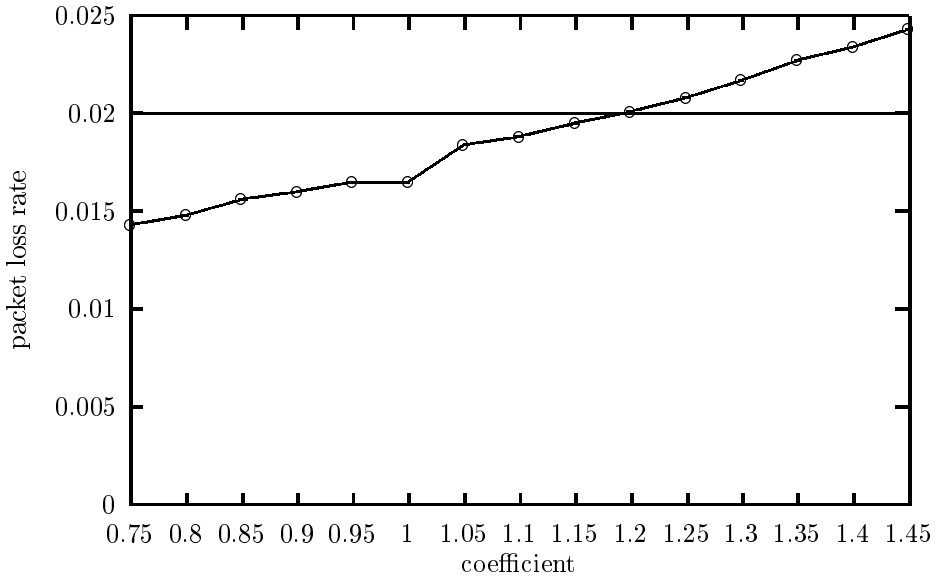


Figure 4.9: Packet loss rate of credit-based flow control.

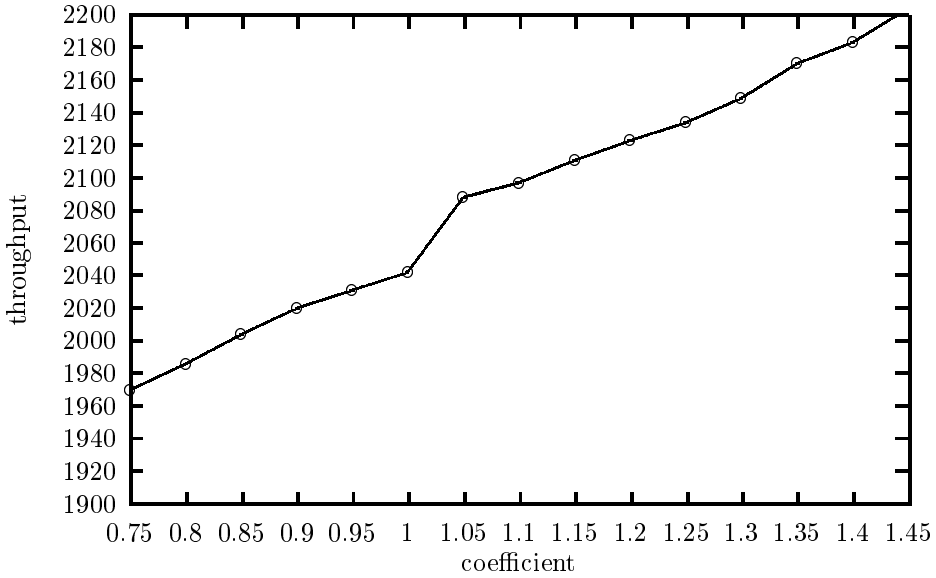


Figure 4.10: Throughput of credit-based flow control.

Table 4.7: Packet loss rate and throughput of credit-based flow control.

multiplicative coefficient	packet loss rate	throughput
0.75	0.0143	1970
0.80	0.0149	1986
0.85	0.0156	2004
0.90	0.0161	2020
0.95	0.0165	2031
1.00	0.0166	2042
1.05	0.0185	2088
1.10	0.0188	2097
1.15	0.0196	2111
1.20	0.0201	2123
1.25	0.0209	2134
1.30	0.0218	2149
1.35	0.0228	2170
1.40	0.0234	2183
1.45	0.0243	2204

4.13.

- The thresholds and step sizes are determined under 2% packet loss rate and 45 mean rate of exotic traffic. The results do reflect the faithfulness of our determination.

- We then use the same thresholds and step sizes for other mean rate of exotic traffics. We see that the packet loss rate of the credit-based flow control perform a little better than the others when the mean rate of exotic traffic equals 45. However, when the exotic traffic is lighter, the packet loss rate of filter-based flow control is lower than the credit-based and rate-based flow controls. As anticipated, the throughput of the filter-based flow control, at light exotic traffic, is smaller than the other two, since it is more conservative in rate adjustment.

- When the exotic traffic increases, we observe that the packet loss rate of filter-based flow control grows faster than the other twos. This indicates that the

packet loss rate of the filter-based flow control scheme is more sensitive to the traffic load variability.

- We then compare the two filter-based flow control schemes (with and without history). We conclude that history information does help in improving the performance.

- Finally, we note that the variance of throughput of the filter-based flow control scheme is much smaller than the other twos. Notice that when the mean rate of the exotic traffic is 45, the variance of throughput of the filter-based flow control is about one-fourth of the the other twos. This hints that the filter-based flow control is more robust in traffic changes in its throughput.

In Figures 4.15–4.18, we collect all the throughput corresponding to specified range of packet loss rate at each mean rate of exotic traffic. When the specified packet loss rate ranges from 0% to 0.5%, the credit-based flow control cannot survive, i.e., the credit-based flow control cannot support such mean rate with such low packet loss rate. However, the filter-based flow control sustains well in this low packet loss rate, in spite of its low throughput.

Our next experiment focuses on the performance of the three flow control schemes under various bursty period/regulation period ratio. Now the mean rate of exotic traffic is fixed at 45. For each regulation period/bursty period chosen, we again use the maximum-throughput thresholds and step sizes corresponding to 2% packet loss rate. Such choices of thresholds and step sizes are confirmed by Table 4.11.

¿From Tables 4.12, 4.13 and Figures 4.19, 4.20, 4.21, we find that the throughput decay fast when (bursty period)/(regulation period) is beyond 1. We also note that

Table 4.8: The packet loss rate under bursty period = 20 and regulation period = 10.

mean rate of exotic traffic	filter-based w/h	filter-based-wo/h	credit-based	rate-based
41	0.0022	0.0019	0.0076	0.0055
42	0.0053	0.0052	0.0107	0.0105
43	0.0092	0.0089	0.0134	0.0145
44	0.0137	0.0138	0.0162	0.0178
45	0.0192	0.0195	0.0192	0.0191
46	0.0238	0.0234	0.0223	0.0212
47	0.0296	0.0288	0.0252	0.0213
48	0.0355	0.0350	0.0286	0.0208
49	0.0441	0.0425	0.0336	0.0188

Table 4.9: The throughput under bursty period = 20 and regulation period = 10.

mean rate of exotic traffic	filter-based w/h	filter-based wo/h	credit-based	rate-based
41	3013.55	2963.84	3644.73	3489.73
42	2703.64	2681.05	3159.04	3119.98
43	2490.06	2460.71	2787.61	2811.49
44	2264.63	2246.22	2439.21	2462.99
45	2050.77	2048.20	2095.67	2050.98
46	1870.48	1842.63	1809.44	1797.25
47	1690.16	1660.73	1525.28	1455.66
48	1511.86	1492.13	1285.76	1139.19
49	1388.99	1351.23	1080.21	813.87

Table 4.10: The variance of throughput under bursty period = 20 and regulation period = 10.

mean rate of exotic traffic	filter-based w/h	filter-based wo/h	credit-based	rate-based
41	37861.2	28414.0	92151.3	17135.7
42	38821.6	39245.5	80391.8	46935.3
43	47811.5	53561.6	102513.1	70189.4
44	49172.7	49000.4	118296.1	113257.4
45	42973.8	49755.4	154958.1	149774.3
46	31281.5	33328.6	99737.1	100647.1
47	24675.5	27023.7	71449.5	89205.0
48	20344.8	17340.0	49837.2	60394.5
49	13265.1	11567.4	24953.7	47763.7

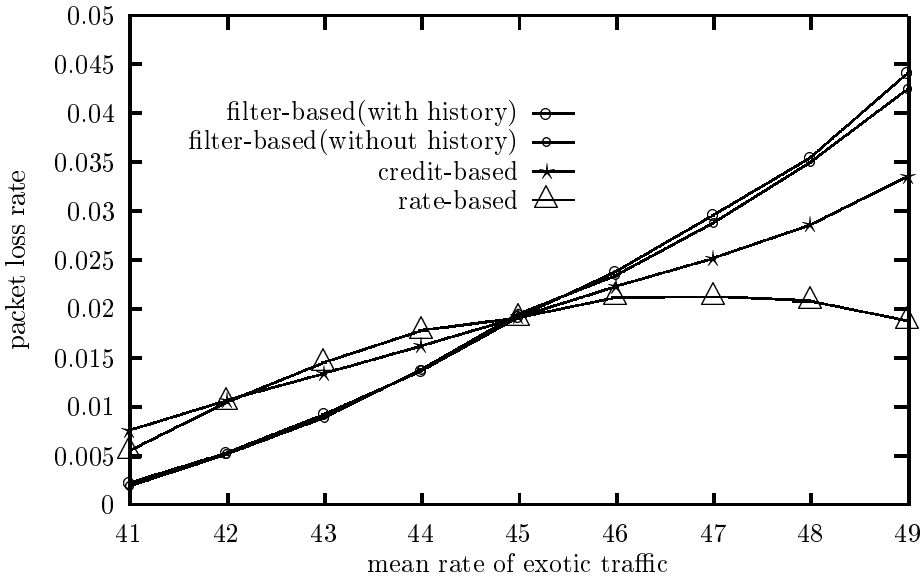


Figure 4.11: The packet loss rate under bursty period = 20 and regulation period = 10.

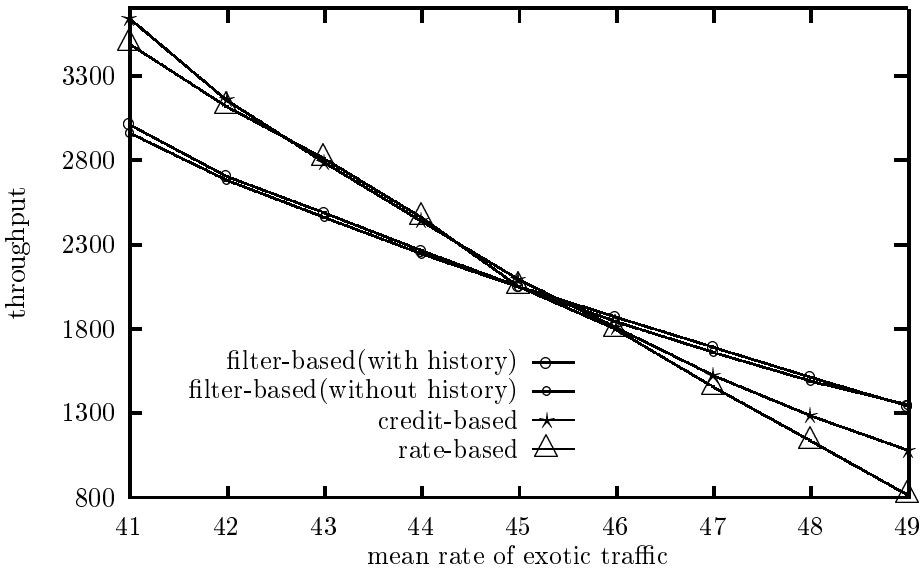


Figure 4.12: The throughput under bursty period = 20 and regulation period = 10.

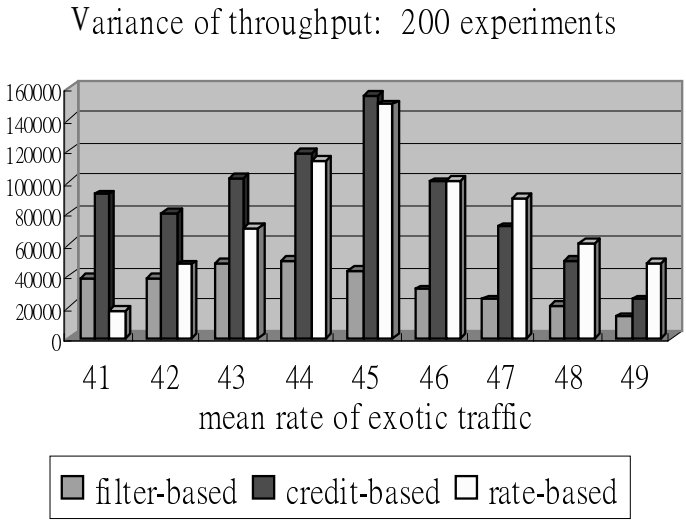


Figure 4.13: The variance of throughput under bursty period = 20 and regulation period = 10.

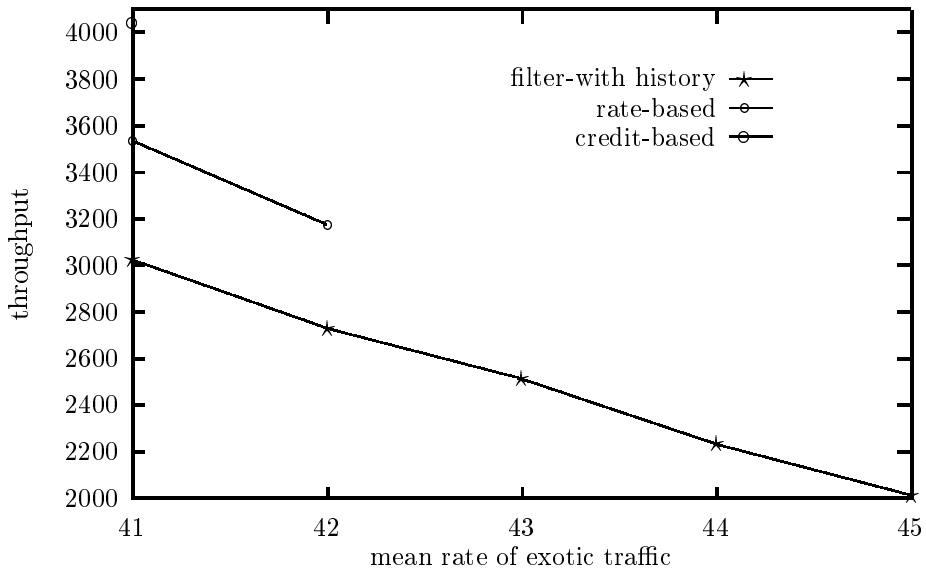


Figure 4.14: Packet loss rate: 0% - 0.5%

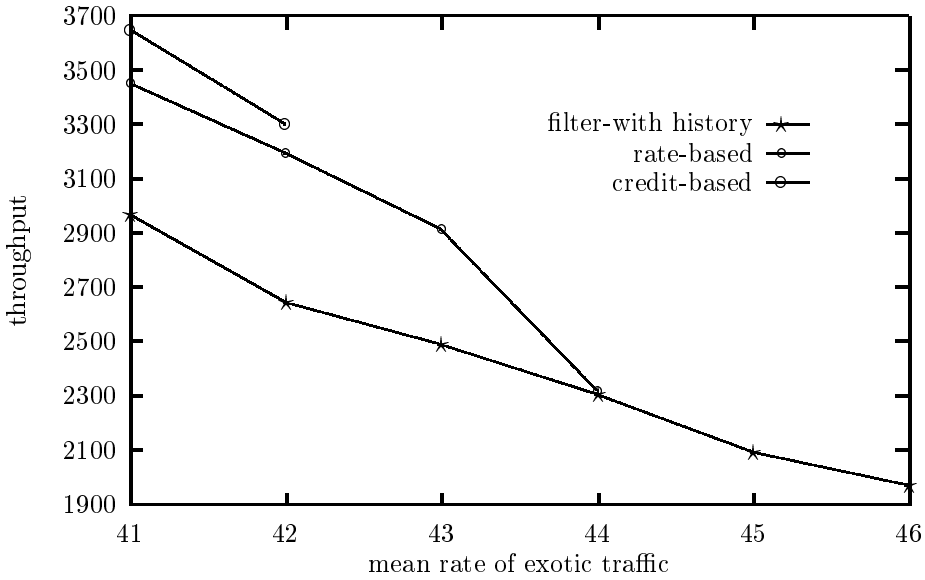


Figure 4.15: Packet loss rate is between 0.5% - 1 %

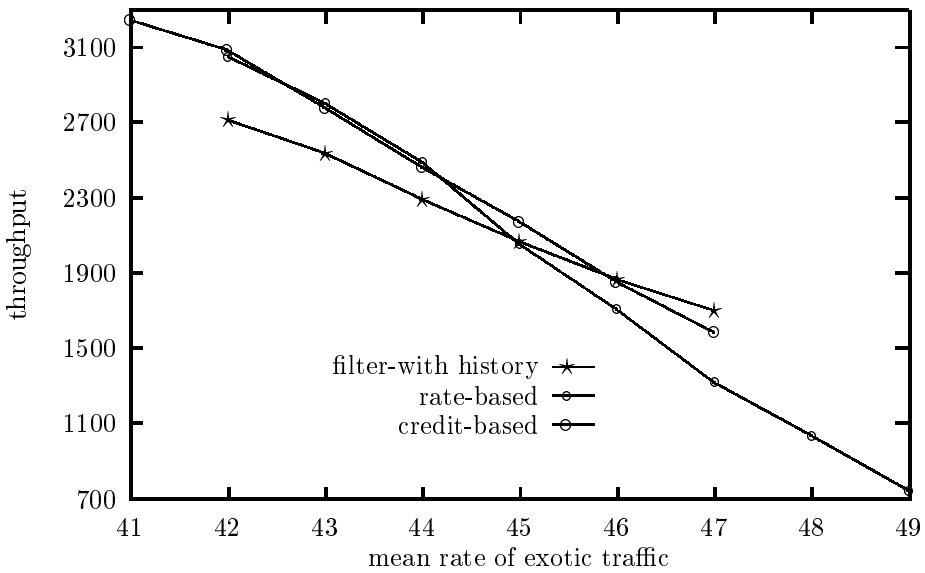


Figure 4.16: Packet loss rate is between 1% - 2%

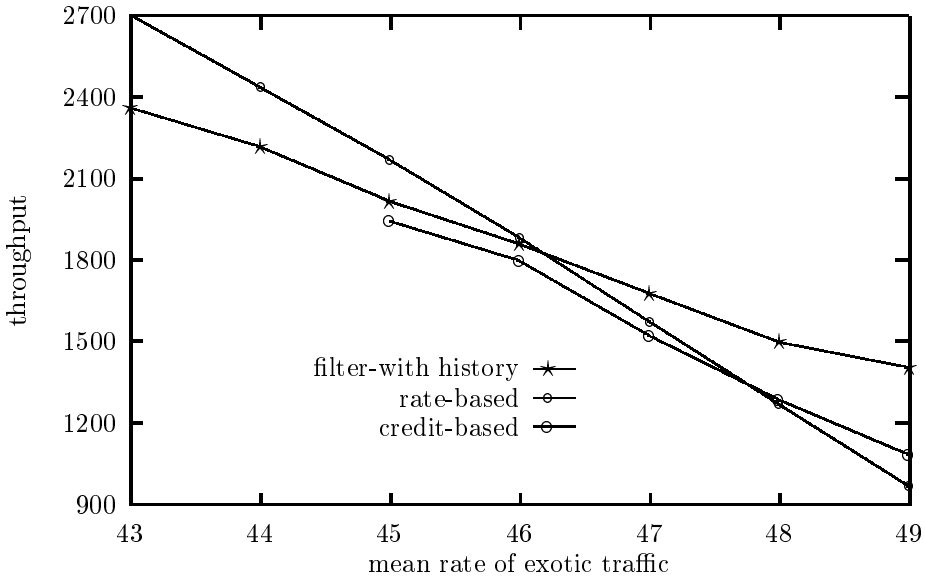


Figure 4.17: Packet loss rate is between 2% - 4%

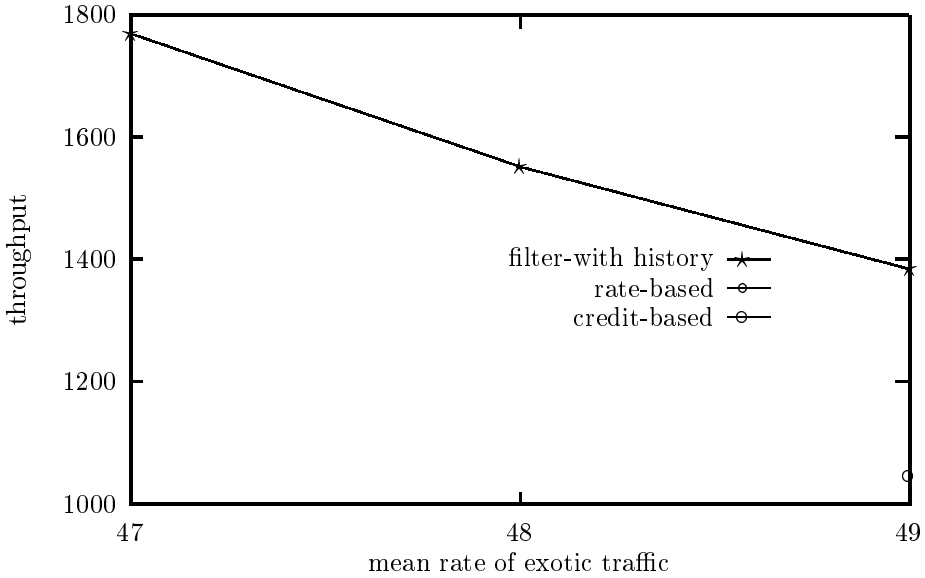


Figure 4.18: Packet loss rate is between 4% - 6%

the throughput of credit-based flow control is much smaller than the others when the (bursty period)/(regulation period) is low, while the throughputs of filter-based and rate-based flow controls are almost the same at each (bursty period)/(regulation period). However, when comparing the variance of throughput, we again find that the filter-based is better than the other two, especially when (bursty period)/(regulation period) is large.

The final experiment concentrates on a system with 8 and 12 network nodes. From Tables 4.14 and 4.15, we can see that the performance of the filter-base flow control is better than the other twos when the network node number increases.

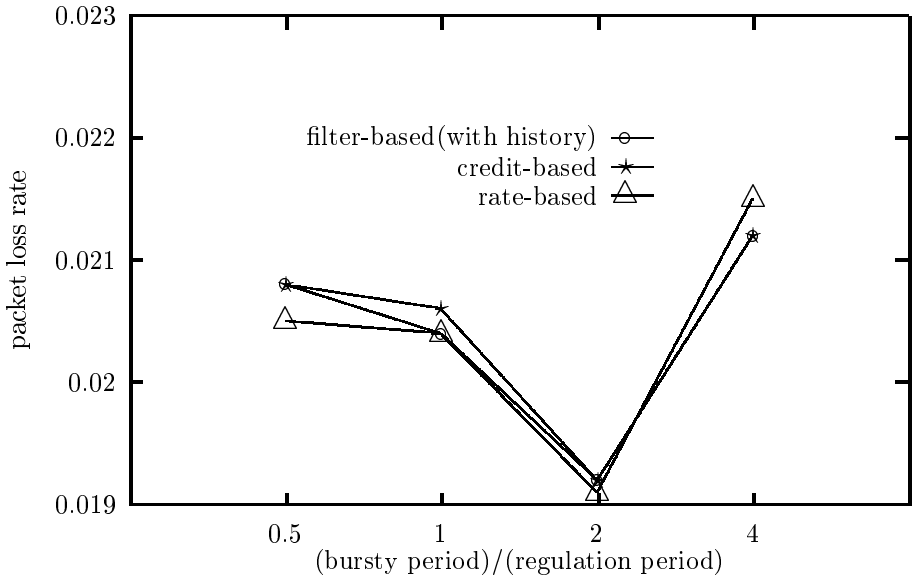


Figure 4.19: The packet loss rate under fixed mean rate of exotic traffic = 45 and various bursty period/regulation period.

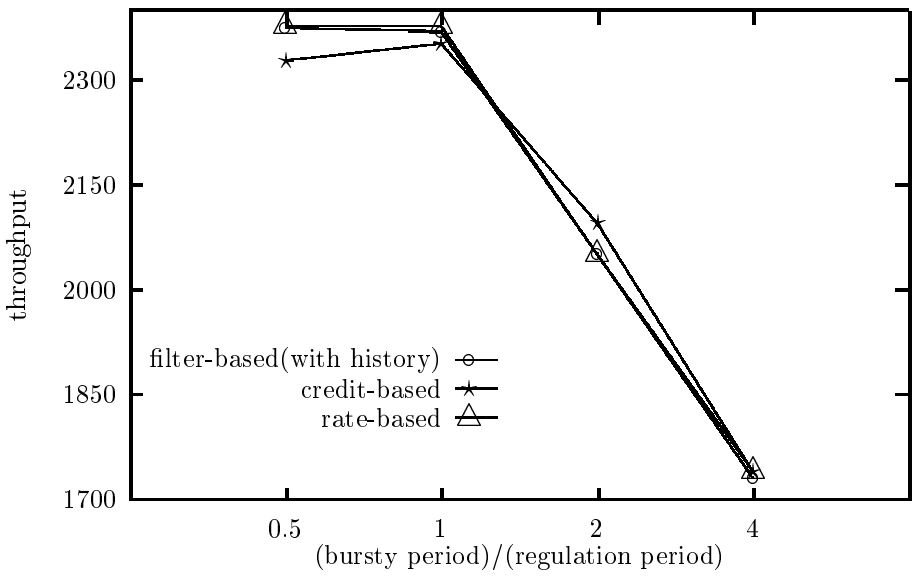


Figure 4.20: The throughput under fixed mean rate of exotic traffic = 45 and variout bursty period/regulation period.

Table 4.11: The packet loss rate under fixed mean rate of exotic traffic = 45 and variout bursty period/regulation period.

(bursty period)/(regulation period)	0.5	1.0	2.0	4.0
filter-based	0.0208	0.0204	0.0192	0.0212
credit-based	0.0208	0.0206	0.0192	0.0212
rate-based	0.0205	0.0204	0.0191	0.0215

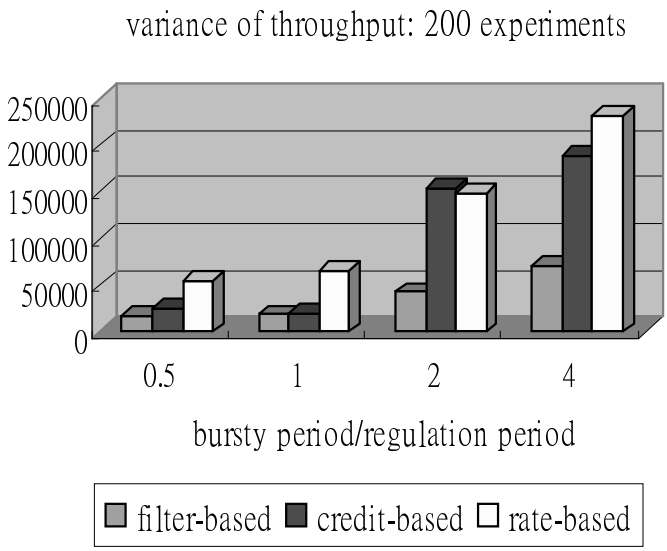


Figure 4.21: The variance of throughtput under fixed mean rate of exotic traffic = 45 and variout bursty period/regulation period.

Table 4.12: The throughput under fixed mean rate of exotic traffic = 45 and variout bursty period/regulation period.

(bursty period)/(regulation period)	0.5	1.0	2.0	4.0
filter-based	2374.86	2369.87	2050.77	1730.94
credit-based	2328.67	2353.39	2095.67	1739.98
rate-based	2377.72	2377.76	2050.98	1741.66

Table 4.13: The variance of throughput under fixed mean rate of exotic traffic = 45 and variout bursty period/regulation period.

(bursty period)/(regulation period)	0.5	1.0	2.0	4.0
filter-based	17912.6	19643.8	42973.8	71594.4
credit-based	25724.9	20626.3	154958.1	189603.4
filter-based	54914.8	65945.3	149774.3	232888.6

Table 4.14: System with eight network nodes.

	packet loss rate	throughput
filter-based	0.0142	2214.63
credit-based	0.0145	1997.40
filter-based	0.0146	2182.24

Table 4.15: System with twelve network nodes.

	packet loss rate	throughput
filter-based	0.0122	2127.24
credit-based	0.0124	1848.28
rate-based	0.0122	2084.53

Chapter 5

Conclusions

5.1 Discussion

1. We use a first-order filter $h(t)$ to approximate the traffic pattern of the network node, and $x(t) - h(t)$ to estimate its crowdedness. From our experiment, we notice that $x(t) - h(t)$ does not really reflect the recent buffer length. Hence, sometime the network is crowded, but the filter-based flow control still sense smooth. As expected, the filter-based flow control gets low packet loss rate when the traffic is not so heavy.

2. By employing a simple convolution operation, the filter-based flow control can not only know the traffic load of each node, but also realize the congestionness of its subsequent nodes. This is different from the conventional rate-based and credit-based which make adjustment only based on the measure quantity in one node. This explains the reason why the filter-based flow control is more robust (smaller variance in throughput) than the other twos.

3. History information does help improving the performance of filter-based flow control. But the improvement is only limited. This may be due to our discrimination (by weightedness) on the history information. The current weighting is given base on simple intuition. Further elaborate design on the weighting may provide more useful usage of history information.

4. For systems with 8 and 12 network nodes, the superiority of the filter-based flow control in throughput variance is more obvious. Also, in such cases, the difference in throughput among filter-based, rate-based and credit-based flow controls become significant.

5.2 Future Work

1. In our proposed filter-based flow control, we use identical thresholds and step sizes for each node. Since each node may experience different congenstionness, a possible improvement could be resided on dynamically using different thresholds and step sizes for each node.

2. Also, in each node, the threshold and step size is set fixed in spite of the varies of exotic traffic. It is also possible to dynamically update the thresholds and step sizes at the node to cope with different exotic traffic.

Appendix A

The rate-based flow control

The rate-based flow control is an end-to-end control scheme using a single-bit feedback from the network[5][8][9][7]. When a cell is generated at one end, its EFCI state is set to "congestion not experienced." As the cell traversed through the network, the EFCI state of the cell is changed to "congestion experienced" if an intermediate node is in a congestion state. Hence, a cell arriving at the other end with the EFCI state set to "congestion experienced" indicates that at least one internodal link along the path is congested. This "Congestion in the forward path" is sent back by using a Resources Management (RM) cell to the source end. The source end then adjusts its rate using additive increase and multiplicative decrease algorithm.

The congestion condition is determined by monitoring the queue length. There are two thresholds used: a high threshold and a low threshold. The link is said to be in a congested state if the queue length exceeds the high threshold, and the EFCI state of the passing cell is updated to "congestion experienced." When the queue length drops below the low threshold, the link is said to be in a normal state, and no EFCI state update is necessary. Sometimes, the rate adjustment of the source end is modeled by linear increase and exponentially decrease in a continuous domain for

the analytical purpose. Usually, the following parameters are defined for rate-based flow control:

β - Multiplicative decrease factor for the rate reduction.

α - Additive rate increase slope.

$\triangle$ - Interval determine how fast the rate reduction is.

ICR - Initial cell rate.

PCR - Peak cell rate.

Q_h - High threshold in the queue.

Q_l - Low threshold in the queue.

The rate increase and decrease algorithm is expressed as

$$R(t) = R(t_0) + \alpha(t - t_0),$$

$$R(t) = R(t_0)e^{-(1-\beta)\frac{t-t0}{\triangle}}$$

where t is the current time and t_0 is the time corresponding to the last rate change.

Appendix B

The credit-based flow control

In a credit-based flow control scheme, each link consists of sender node (either a source end-system or a switch) and a receiver node (either a destination end-system or a switch)[6][10]. Each node maintains a separate queue for each VC. The sender can transmit as many cells as allowed by the credit. If there is only one active VC, the credit must be large enough to allow the whole bandwidth to be utilized at all times and also match the smallest link on the path. The link cell rate can be computed by dividing the link bandwidth by the cell size.

In the later version, the credit-based approach was enhanced to give adaptive credit where each VC only gets a fraction of the round-trip delay's buffer allocation. The fraction depends on the rate at which the VC uses the credit. For highly active VCs, the fraction is larger; for less active VCs, the fraction is smaller.

In our simulation, credits can be calculated by the total buffer length minus the recent buffer length, and multiply it by a multiplicative coefficient. Hence, we only need to determine one factor.

Bibliography

[1] Cheng-Shang Chang,"A filtering theory for deterministic traffic regulation," *IEEE INFOCOM'97* , vol. 2, pp. 436-443, 1997.

[2] ————————-, "On deterministic traffic regulation and service guarantees : A systematic approach by filtering," *IEEE Trans. Info. Theory*, vol. IT-44, pp. 1097 - 1110, May 1998.

[3] ————————-, *Performance Guarantees in Communication Network*, draft, August 1998.

[4] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation,"*IEEE Tran. Inform. Theory* , vol. 37, pp. 114-131, 1991.

[5] Cambyse Guy Omidyar and Guy Pujolle, "Introduction to flow and congestion control," *IEEE Comm. Magazine*, pp. 30-32, November 1996.

[6] R. Schoenen, G. Post and A. Muller, "Analysis and dimensioning of credit-based flow control for the ABR service in ATM networks," *IEEE GLOBECOM 1998*, vol. 4, pp. 2399-2404, 1998.

[7] Hiroyuki Ohsaki and Masayuki Murata, "Analysis of rate-based congestion control algorithms for ATM networks-Part 1: Steady state analysis," *IEEE INFOCOM '95* , vol. 1, pp. 296-303, 1995.

[8] Nanying Yin, "Analysis of a rate-based traffic management mechanism for ABR service," *IEEE GROBECOM 1995*, vol. 2, pp. 1076-1082, 1995.

[9] Nanying Yin, G. Michael and Hluchyj, "On closed-loop rate control for ATM cell relay networks," *IEEE INFOCOM '94* , vol. 1, pp. 99-108, 1994.

[10] Xi Zhang, K. G. Shin and Qin Zheng, "Integrated rate and credit feedback control for ABR service in ATM networks," *INFOCOM '97*, vol. 3, pp. 1295-1303, 1997.