GENERALISED MINIMUM VARIANCE CONTROL OF QUEUES IN PACKET SWITCHING NETWORKS[*]

HARSHA SIRISENA[†] AND MAHBUB HASSAN[‡]

Abstract. The design of rate allocation and queue length control in computer networks is treated as a stochastic optimal control problem. The performance index is chosen to achieve a trade-off between minimising the queue length fluctuations and reducing fluctuations in the sending rate while fully utilising the available bandwidth. Simple, practically realisable optimal control schemes comprising queue length feedback and available bandwidth feedforward are obtained for both LANs and WANs. Also an adaptive scheme is proposed where the auto-regressive parameters of the traffic, needed for gain calculations, are estimated by an LMS algorithm. Discrete event simulations are carried out to verify the fluid-flow models used in developing the controllers, to compare their performance against PI controllers proposed previously, and to study the effect of self-similar traffic. Two key results are obtained. First, queue-length variance, and hence potentially packet losses, are much smaller for the optimal feedforward controller than for the PI controller. Second, in contrast to uncontrolled queues, the queue length variance decreases with increasing Hurst parameter for self-similar traffic.

1. Introduction. In packet switching networks, the incoming traffic load on certain links in the routers and switches can momentarily exceed the link capacity. In such situations, packets are temporarily queued in buffers. Network queues can be controlled by dynamically adjusting the transmission rates of the sources feeding traffic into these queues. The source rates should be decreased when the queue grows too much and increased when the queue recedes. Such dynamic control of source rates leads to the typical oscillating queueing behaviour. Large fluctuations cause the following undesirable effects: increased buffer overflow (packet loss), buffer underflow (waste of available link bandwidth) and network jitter or delay variance. It is therefore important to minimise queue fluctuations (or equivalently queue length variance) in computer networks to minimise the above undesirable effects.

Optimal control of queues aims to minimise queue length variance by optimally computing the source rates. TCP/IP networks currently do not implement optimal control. The queues in IP routers are loosely controlled by TCP's adaptive congestion control algorithms [9] which mainly focus on preventing congestion collapse and maintaining a stable network. TCP achieves this by reacting to packet loss (buffer overflow) with a severe cut to packet flow and slowly increasing the rate as it starts

---

[†]Department of Electrical and Electronic Engineering, University of Canterbury, Christchurch, New Zealand. E-mail: Harsha.Sirisena@canterbury.ac.nz

[‡] School of Computer Science and Engineering, The University of New South Wales, Sydney 2052, Australia. E-mail: mahbub@cse.unsw.edu.au

to receive acknowledgements from the receiver. Large queue oscillations in TCP/IP networks are evidenced by high packet loss rate and large jitter (delay variance) in the Internet.

Although theoretical analyses of source rate allocations and queue length dynamics in TCP/IP networks are rare, there has been great interest in modelling the rate control problem as a feedback control system in the context of ATM networks [1, 5, 11, 12]. In an early deterministic treatment, Benmohamed and Meerkov [5] essentially proposed a PI (*proportional* and *integral*) feedback controller without any disturbance feedforward. In their so called PD scheme, the time derivative of the admission rate is proportional to the queue length and its derivative. Such PI controllers are known to provide queue control with zero offset (the mean queue-length converges to the target value), but are inherently less stable and slower acting than feedforward control. Benmohamed [5] did not provide an optimal solution to the queue control problem. There have been several other control-theoretic solutions proposed [1, 11, 12] to design stable and robust rate control mechanisms, but none of them address the issue of optimality in terms of minimising queue length variance.

Altman et al. [2] use stochastic optimal control theory to devise a rate control scheme with the twin objectives of minimising queue variance and variance of the rates allocated to the sources. For their optimal solution to work adaptively, it would require the solution of a complex non-linear discrete Riccati equation at every time step. This is too complex for practical implementation and so they propose a suboptimal solution. Implementation complexity is a crucial factor in the adoption of practical solutions. However, the widely used ATM ABR algorithms, such as ERICA [10], are simple to implement, but do not provide optimal queue control.

In the present paper, we design a rate controller based on generalised minimum variance control [6] instead of the LQR approach of [2]. An explicit expression is thereby obtained for the optimal rate controller. The performance index employed is a weighted sum of the variances of the queue length and the rate fluctuations. By varying the relative weighting, the effects of undue rate fluctuations on higher layer protocols, e.g. TCP, may be kept in check. This generalises the minimum variance controller we presented in [7].

Our approach yields a simple, practically realisable optimal control scheme that meets the two objectives stated above. We minimise the generalised variance at the earliest time affected by the current control signal, and in so doing achieve the second objective of fully utilising the mean available bandwidth. We find that the optimal control comprises simple proportional control, without the integral term, together with feedforward of the disturbance (background traffic). By using disturbance feedforward our controller is *proactive* to traffic changes (in addition to being *reactive* to queue length changes).

We derive optimal controllers for both LANs (no delay) and WANs (homogeneous

delays; all sources are at the same distance from the bottleneck queue). We also propose an adaptive algorithm which adaptively estimates the background traffic level and its correlation parameter without requiring any prior knowledge of the dynamics of the background traffic. We perform discrete event simulations with both exponential and self-similar background traffic and compare the performance of our optimal controller with that of typical PI controllers to assess the efficacy of our controller.

The rest of the paper is organised as follows. Section 2 presents the design and analysis of our generalised minimum variance controller for both LANs and WANs. Section 3 describes our simulation model. Simulation results are presented in Section 4 followed by our conclusion in Section 5.

2. Optimal Controller Analysis and Design.

2.1. LAN Scenario. The first scenario considered is of a LAN host, having a connection across a WAN, that has its rate controlled by the WAN gateway. In this case the latency of the feedback loop is negligible in comparison to packet transmission times.

Let $B(k)$, $R(k)$ and $Q(k)$ denote the k-th samples of, respectively, the bandwidth available to the connection, the controlled source rate and the queue length at the buffer in the gateway. $B(k)$ is the capacity of the outgoing link from the gateway less the (uncontrolled) background traffic rate, and is modelled as the stochastic autoregressive (AR) process

$$(1) \qquad B(k) = \alpha B(k-1) + \beta W(k) + (1-\alpha)\bar{B}$$

where $W(k)$ is unit white noise and $\bar{B}$ is the mean value of $B(k)$. The parameter α captures the time correlation of $B(k)$. The parameter β depends on the variance of $B(k)$, which can be derived from (1) as

$$(2) \qquad Var\{B(k)\} = \frac{\beta^2}{(1-\alpha^2)}.$$

Note that Equation (1) models short-range dependency, and is exact for on-off sources with exponentially distributed on and off times. Also note that our methodology is readily extended to the case where (1) is replaced by a higher-order AR process. We argue that the exponential distribution is actually a worst-case in the sense that future values of $B(k)$ are less predictable than when $B(k)$ exhibits long-range dependency. This is borne out by the results presented in Section 4 which show decreased queue length variance with Pareto-distributed on and off times.

Denoting the sampling period by T, the queue length $Q(k)$ obeys the following equation

$$(3) \qquad Q(k+1) = Q(k) + T[R(k) - B(k)].$$

Our goal now is to find the optimal rate $R(k)$ that minimises the packet loss rate due to buffer overflow while keeping the mean queue length equal to a reference value Q_r. To this end, we first determine the variance of $Q(k+1)$ and then choose $R(k)$ to minimise it, keeping in mind that because of the averaging delay only a measurement of $B(k-1)$ is available. This is an optimal minimum-variance control problem [4].

Substituting (1) in (3) gives:

$$(4) \qquad Q(k+1) = Q(k) + T[R(k) - \alpha B(k-1) - \beta W(k) - (1-\alpha)\bar{B}].$$

As the mean queue length is required to be Q_r,

$$Var\{Q(k+1)\} = E\{Q(k+1) - Q_r\}^2$$
$$(5) \qquad\qquad = E\{Q(k) - Q_r + T[R(k) - \alpha B(k-1) - (1-\alpha)\bar{B}] - T\beta W(k)\}^2.$$

$T\beta W(k)$ is the only stochastic term on the RHS, so $R(k)$ that minimises $Var\{Q(k+1)\}$ must make the deterministic terms vanish. This gives the optimal minimum variance (MV) control law:

$$(6) \qquad\qquad R(k) = \bar{B} + (1/T)[Q_r - Q(k)] + \alpha[B(k-1) - \bar{B}].$$

And then the minimum variance of the queue length equals the variance of the stochastic term. Using (2) to substitute for β, this is

$$(7) \qquad\qquad Var\{Q(k+1)\} = T^2\beta^2 = T^2(1-\alpha^2)Var\{B(k)\}.$$

Notice that the optimal control law (6) is the sum of three terms, the first being the *mean* available bandwidth $\bar{B}$. This term ensures that the available bandwidth is fully utilised; it arises from the requirement implicit in Eqn. (5) that $E\{Q(k+1)\} = Q_r$. The other two terms in (6) are a *feedback* gain $(= 1/T)$ times the difference between the current and reference queue lengths, and a *feedforward* gain $(= \alpha)$ times the deviation of the most recent available bandwidth $B(k-1)$ from $\bar{B}$.

In order to implement (6), $B(k-1)$ is estimated from the buffer's drain rate, and $\bar{B}$ is then obtained by low-pass filtering these estimates. The correlation parameter α may be estimated in real time by an LMS (or WLS) algorithm. Details are given in Section 2.3.

A well-known issue with minimum variance control laws such as (6) is that the optimal control signal has large fluctuations. In the present context, large fluctuations in the ABR service rate $R(k)$ could conceivably cause problems to higher layer protocols, e.g. timeouts in TCP and thus a reduction in throughput. Hence Generalised Minimum Variance (GMV) control [6] could be more suitable. That is, we minimise the quadratic index

$$J\{R(k)\} = Var\{Q(k+1)\} + \gamma \times Var\{R(k)\}$$
$$(8) \qquad\qquad = Var\{Q(k+1)\} + \gamma \times E\{(R(k) - \bar{B})^2\}$$

because $\bar{B}$ is the expected value of $R(k)$. Performing the minimisation, we get the optimal GMV control law

$$(9) \qquad R(k) = \bar{B} + \frac{T}{T^2 + \gamma}[Q_r - Q(k)] + \frac{\alpha T^2}{T^2 + \gamma}[B(k-1) - \bar{B}].$$

Note that setting $\gamma = 0$ in (9) gives the MV control law (6).

Finally, and most importantly, stability of our controller is assured by the standard result [3] that a linear quadratic optimal control law for a controllable system (which Eq. (3) is) is stabilising.

2.2. WAN Scenario. Next we turn to the scenario where source rate control is exerted by a bottleneck node on a WAN. Now there is a control loop latency equal to the round-trip time. The buffer equation (3), modified to reflect the latency is:

$$(10) \qquad Q(k+1) = Q(k) + T[R(k-1) - B(k)],$$

with the sampling period chosen to be one round-trip time. This choice makes sense because changes in control action should only be made after the effect of previous actions is known to the controller (e.g. as in TCP).

Clearly, the first queue length that can be affected by the controlled rate $R(k)$ is $Q(k+2)$, which by recursive use of (10) and (1) is obtained as

$$Q(k+2) = Q(k) + T[R(k-1) - \alpha B(k-1) - \beta W(k) - $$
$$(1-\alpha)\bar{B}] + T[R(k) - \alpha^2 B(k-1) - \alpha\beta W(k)$$
$$(11) \qquad\qquad -\alpha(1-\alpha)\bar{B} - \beta W(k+1) - (1-\alpha)\bar{B}].$$

Similarly as before, the variance of $Q(k+2)$, i.e. $E\{Q(k+2) - Q_r\}^2$, is minimised when the deterministic terms on the RHS of (11) sum to Q_r. This gives the MV control law:

$$(12) \qquad R(k) = 2\bar{B} - R(k-1) + (1/T)[Q_r - Q(k)] + (\alpha + \alpha^2)[B(k-1) - \bar{B}].$$

This corresponds to a minimum queue length variance of

$$Var\{Q(k+2)\} = T^2\beta^2[1 + (1+\alpha)^2]$$
$$(13) \qquad\qquad = T^2(1-\alpha^2)[1 + (1+\alpha)^2]Var\{B(k)\}.$$

As in the MV rate control law (6) for the LAN scenario, the control law (12) requires feedback of the queue length deviation and feedforward of the available bandwidth (but with a gain of $\alpha + \alpha^2$ rather than α). But the term $\bar{B}$ is replaced by $2\bar{B} - R(k-1)$. The latter may be viewed as the addition to $\bar{B}$ of the deviation from the mean available bandwidth of the controlled rate during the previous sampling

period. As in the LAN scenario, the term $\bar{B}$ ensures that the available bandwidth is fully utilised in the mean.

For the WAN scenario too, we can minimise an augmented quadratic index

$$(14) \qquad J\{R(k)\} = Var\{Q(k+2)\} + \gamma \times E\{(R(k) - \bar{B})^2\}$$

to derive the GMV control law:

$$R(k) = \frac{2T^2 + \gamma}{T^2 + \gamma}\bar{B} - \frac{T^2}{T^2 + \gamma}R(k-1)$$

$$(15) \qquad + \frac{T}{T^2 + \gamma}[Q_r - Q(k)] + \frac{T^2}{T^2 + \gamma}(\alpha + \alpha^2)[B(k-1) - \bar{B}].$$

2.3. Parameter Estimation. Implementation of the two GMV control laws (9) and (15) requires real-time estimates of α as well as of $\bar{B}$. The latter is obtained by low-pass filtering (EWMA) the measurements of $B(k)$,

$$(16) \qquad \bar{B}(k) = \lambda \bar{B}(k-1) + (1 - \lambda)B(k).$$

We used $\lambda = 0.95$ in the simulations. Then α can be estimated by applying the normalised Least Mean Square (LMS) algorithm [8] to equation (1):

$$(17) \quad \alpha(k) = \alpha(k-1) + \frac{\delta[B(k) - \alpha(k-1)B(k-1) - (1 - \alpha(k-1))\bar{B}(k-1)]}{[\varepsilon + (B(k) - \bar{B}(k))^2]}.$$

We set the adaptation gain δ to 0.01. The quantity ε, introduced to avoid division by zero, was set to 0.1 times $Var\{B(k)\}$, estimated by low-pass filtering $[B(k) - \bar{B}(k)]^2$ using an equation similar to (16).

2.4. Adaptive Rate Allocation Algorithm. At each sampling interval, the total optimal rate for all sources is computed in three steps as follows:

Step 1: Estimate mean available bandwidth using Eq. (1).

Step 2: Estimate α, the correlation parameter of the background traffic, using Eq. (17).

Step 3: Compute the optimal rate using Eq. (9) for a LAN and Eq. (15) for a WAN.

Once the total optimal rate is computed, any "fair" allocation algorithm can be used (e.g., max-min) to fairly distribute the total rate among the competing sources. In our simulations presented in the following section, we divide the total admission rate equally among all the competing sources.

3. Network Simulation. The network queue model simulated is shown in Figure 1. There are three "greedy" non-real time sources trying to send traffic through a packet switching router as fast as possible. The router link is also shared by a number of higher priority delay-sensitive ON-OFF sources. The link scheduler serves packets from the greedy sources only when there is no traffic from ON-OFF sources. In our simulation, ON-OFF sources do not explicitly send any packets and hence no packet

queue is explicitly simulated at the scheduler for these high priority sources. Instead, whenever an ON-OFF source is turned on, the scheduler reduces the link capacity by the peak rate and adds the same to the link capacity whenever it is turned off.
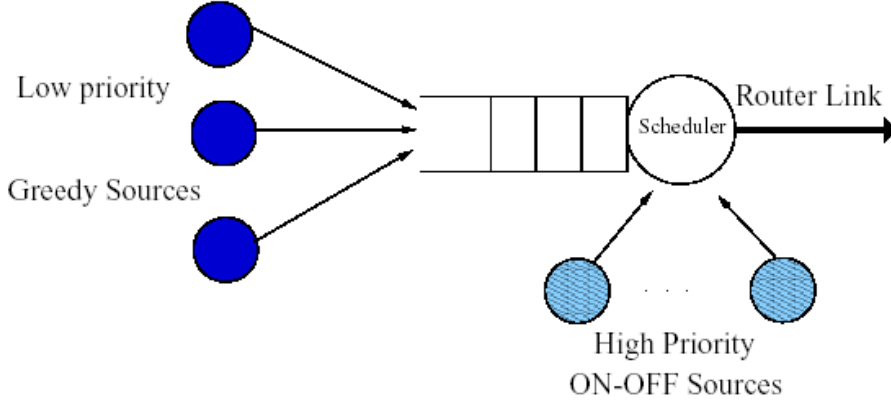

FIG. 1. *Network queue configuration for the simulation model*

Traffic from the greedy sources is queued at the router link. This queue is the subject of control and the behaviour of this queue is evaluated in this simulation. The background traffic from ON-OFF sources act as a disturbance to the system, as the available link rate to serve the packets from the queue varies dynamically due to these ON-OFF sources.

The sending rates of the greedy sources are adjusted dynamically according to the rate feedback from the queue controller. Each source gets one third of the optimal rate computed by the controller. The objective of the controller is to keep the queue at a pre-determined target level by adjusting the aggregate admission rate to the queue.

An ON-OFF source transmits traffic at a constant rate during the ON period and does not send any traffic during the OFF period. For exponential background traffic, the ON and OFF periods are exponentially distributed. To simulate self similar background traffic, the lengths of ON and OFF periods are drawn from Pareto distribution as suggested in [14]. To simulate self-similar traffic with a specific Hurst parameter H, we use the following algorithm: (1) generate an uniformly distributed random variable x between 0 and 1, (2) determine the length of an ON or OFF period as $K(x^{-1/a} - 1)$ where $a = 3 - 2H$ and $K = m(a - 1)$ with m being the mean length of an ON or OFF period.

In our simulations we chose the target queue length Q_r such that the mean queueing delay would be a small fraction (about 0.2) of the mean round-trip time. We recorded the variance of the queue length around this target value as well as the variance of the sending rate R. The values of the simulation parameters are summarised in Table 1.

Parameter	Value
Number of ON-OFF priority sources	90
Transmission rate during ON period for VBR source	50 pk/sec
Mean length of ON and OFF period for VBR source	0.4 sec
Capacity of router link	4500 pk/sec
Sampling interval T	0.1 sec
Target queue-length Q_r	100 packets

The lengths of all simulation runs were dynamically controlled until the values of the estimated variables (e.g., the queue variance) reached steady state, and 95% confidence intervals were achieved with 5% relative precisions. With such adaptive length simulations, the number of packets simulated ranged from millions to billions of packets depending on the simulation parameters. Results obtained from the simulations are presented and compared with the analytical results in the following section.

We simulated both our GMV proportional plus feedforward controller and a *tuned* PI controller. Note that although the plant is an integrator, integral control action is needed to avoid a steady state error due to changing background traffic levels. Benmohamed [5] only showed the existence of a stabilising PI controller, but actually an infinite number of stabilising PI controllers exist. Hence, for simulation purposes, we chose tuning parameters given by Ziegler-Nichols' continuous cycling method [13] which are known to give a good disturbance response.

The discrete PI controller has the transfer function $K_c[1 + (\frac{T}{T_i})/(z-1)]$. For RTT = 0 (LAN scenario), the settings are found to be $K_c = 0.9/T$; $T_i = 1.67T$ and for RTT = T (WAN scenario), they are $K_c = 0.45/T$; $T_i = 5T$. In our simulations, the PI control law was coded in velocity form [4] in order to avoid integrator windup.

4. Results. In this section, we present results obtained from our simulations, with both exponential and self-similar background traffic. Although our optimal control analysis is based on exponential background traffic, it is of interest to study the effect of relaxing the exponential assumption.

4.1. Exponential Background. To verify the accuracy of our optimal control analysis of the queue variance, we compare the theoretical values for MV control with the ones obtained by simulation for the ratio $\frac{Var\{Q\}}{Var\{B\}}$. The ratio is compared, instead of the actual queue variance, to avoid the need to compute the variance of the available link capacity. Table 2 compares this ratio for exponential on and off periods.

For ON-OFF sources with exponential on and off periods with identical means of 0.4 seconds, the theoretical value of the correlation parameter α of the background

traffic is obtained as 0.7266. This value of α is used to compute the values in column 2 (Analysis) of Table 2. From Table 2 we see that the simulation results for $\frac{Var\{Q\}}{Var\{B\}}$ are only slightly different than the ones predicted by our analysis. This may be attributed to errors in the estimation of α in the simulation using the LMS algorithm (Eqn. (17)).

TABLE 2

Analysis and Simulation results for $\frac{Var\{Q\}}{Var\{B\}}$, for GMV controllers ($\gamma = 0$) with exponential background.

Scenario	Analysis	Simulation
LAN	0.118	0.116
WAN	0.470	0.541

Now we compare the performance of MV controller against the PI controller in terms of queue variance. For this purpose, we set $\gamma = 0$ in the GMV controller equation. To observe queue length fluctuations, we recorded queue lengths at discrete intervals of T seconds. Figure 2 shows the queue lengths for the WAN scenario. From the figures it is evident that although both controllers successfully maintain the mean queue length around the target value of 100, the PI controller performs worse than the MV controller in terms of the deviation of the queue length from the target value.

Next, we investigate the effectiveness of the GMV controller in trading off queue variance against rate variance. Figures 3 and 4 show queue lengths and rates as a function of time for different settings of the weighting parameter γ. It is clearly seen that by increasing γ we can reduce rate fluctuations at the expense of higher queue variance.

For a quantitative evaluation, we run simulations until the queue and rate variances converge (many millions of sampling intervals are simulated). For the LAN and WAN scenarios, steady state queue and rate variances along with the mean queue lengths are shown in Tables 3 and 4, respectively. A glance through these tables confirm three important results:

1. In the steady state, both GMV and PI controllers can successfully maintain the mean queue length at the target value (100 in this case).
2. The MV controller (GMV controller with the setting $\gamma = 0$) achieves much lower queue variances (30% lower in LAN and 60% lower in WAN in our simulated network) than the conventional PI controller.
3. An appropriate balance between queue and rate variance can be achieved with GMV control by adjusting the parameter γ. For example, in our simulated WAN scenario, the rate variance is reduced by a factor of eight at the expense of a three-fold increase in queue variance, by increasing γ from zero to $4T^2 = 0.04$.
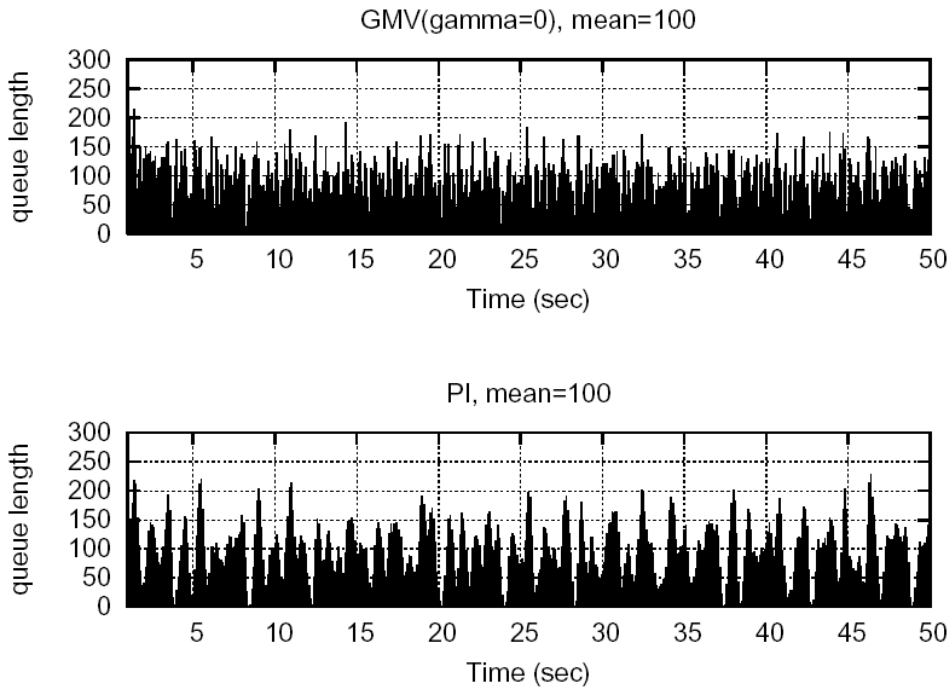
GMV(gamma=0), mean=100

PI, mean=100

Fig. 2. *Queue length comparison of GMV and PI controllers (WAN case).*

TABLE 3
Steady state results for LAN scenario.

Type of Control	mean qlength	qlength variance	rate variance
Tuned PI	100	224	59278
GMV $(\gamma = 0)$	100	156	77298
GMV $(\gamma = T^2)$	100	351	32394
GMV $(\gamma = 2T^2)$	100	726	22672
GMV $(\gamma = 4T^2)$	100	1433	14615

TABLE 4
Steady state results for WAN scenario.

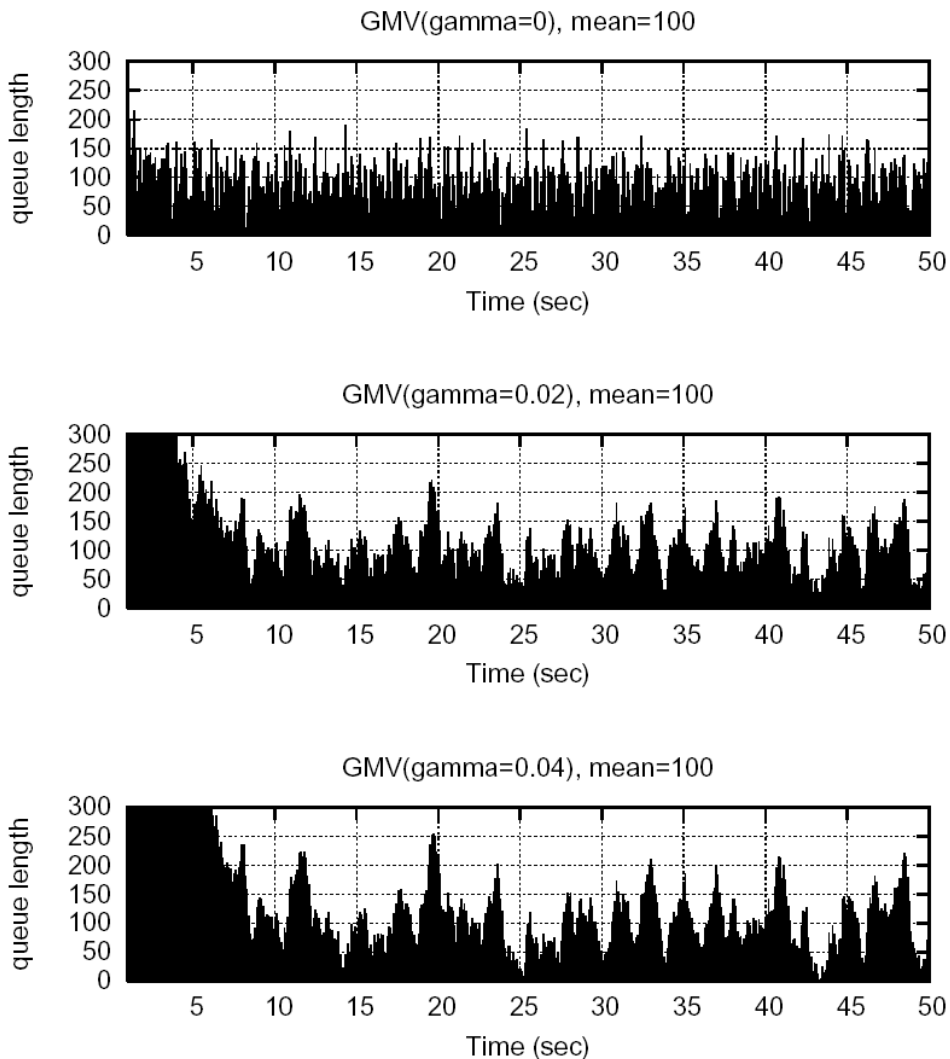| Type of Control	mean qlength	qlength variance	rate variance
Tuned PI	100	2102	68010
GMV $(\gamma = 0)$	100	751	145410
GMV $(\gamma = T^2)$	100	1046	46378
GMV $(\gamma = 2T^2)$	100	1704	29301
GMV $(\gamma = 4T^2)$	100	2692	18140

Fig. 3. *Queue length comparison for different γ settings in GMV controller (WAN case).*

4.2. Self-Similar Background. Figure 5 shows queue and rate variances with Pareto ON-OFF sources with increasing Hurst parameter values. We see that the MV controller ($\gamma = 0$) continues to outperform the PI controller even with self-similar background. The more interesting result here is that irrespective of the control law (MV or PI), a lower queue variance is observed with self-similar background traffic than with exponential ON-OFF traffic. The higher the Hurst parameter of the self-similar traffic, the lower the queue variance. This result suggests that our analysis, which is based on exponential traffic, actually provides results for a worst case scenario.
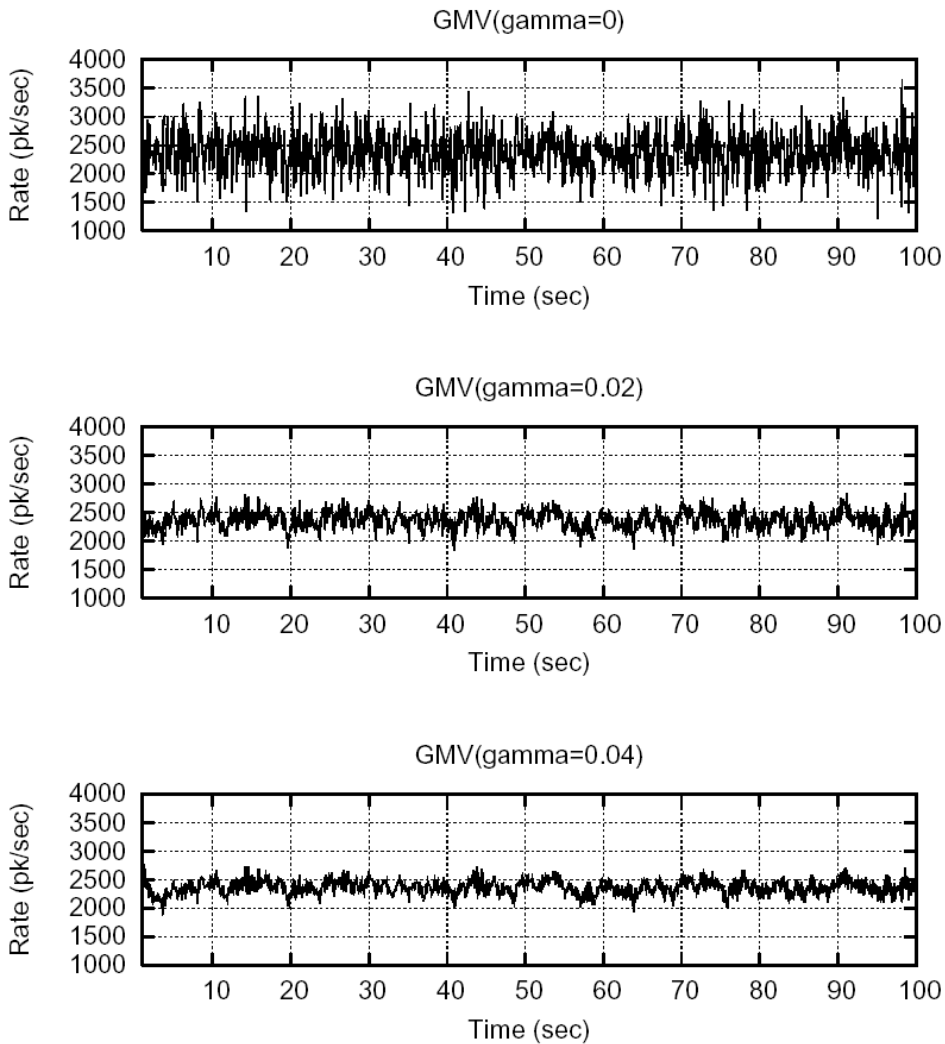
FIG. 4. *Rate fluctuation for different γ settings in GMV controller (WAN case).*

5. Conclusion. We have designed simple and practically realisable adaptive rate allocation algorithms for generalised minimum variance control of queues in both LANs and WANs. Using our algorithms, queue variance can be minimised to reduce packet loss and network jitter, and to increase bandwidth utilisation. No such simple, explicit optimal solutions have been proposed before. Discrete event simulations were carried out to verify the fluid-flow models used in developing our algorithms and to compare their performance against non-optimal PI controllers proposed previously. The results showed that queue variance is much smaller for our optimal algorithm than for a PI controller. We studied the tradeoff between queue variance and rate variance possible by varying the weight parameter in the performance index.
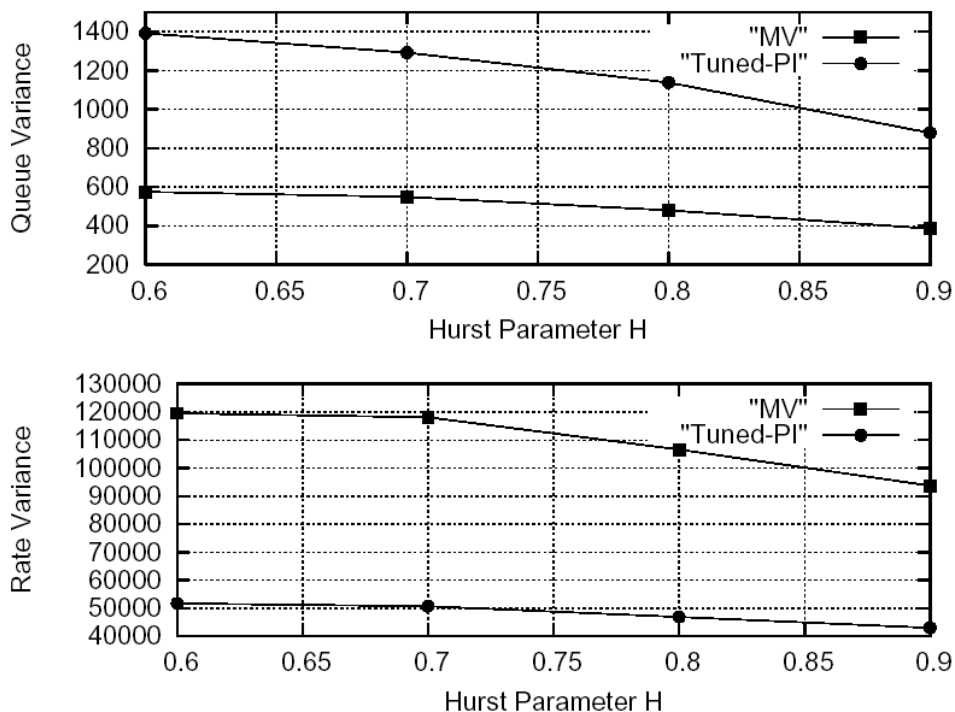
FIG. 5. *Effect of self-similar background traffic on queue and rate variance.*

We also studied the effect of self-similar traffic on both types of controller by performing simulations with aggregated ON-OFF sources having Pareto-distributed on and off times. It was found that the queue length variance *decreases* with increasing Hurst parameter for self-similar traffic. This result implies that if a queue control system was designed assuming exponential background traffic, the system performance will not deteriorate if the actual traffic is of a self-similar nature. This is in stark contrast with the well known problem of *uncontrolled* queues, where self-similar traffic has a deleterious effect on the queue length.

REFERENCES

[1] E. ALTMAN, T. BASAR, AND R. SRIKANT, *Robust Rate Control for ABR Sources*, in: IEEE INFOCOM, pp. 166–173, 1998.

[2] E. ALTMAN, T. BASAR, AND R. SRIKANT, *Congestion Control as a Stochastic Control Problem with Action Delays*, Automatica, 35:12(1999), pp. 1937–1950.

[3] ANDERSON AND MOORE, *Optimal Control*, Prentice Hall, 2nd edition, 1989.

[4] K.J. ASTROM AND B. WITTENMARK, *Computer Controlled Systems*, Prentice Hall, second edition, 1990.

[5] L. BENMOHAMED AND S.M. MEERKOV, *Feedback Control of Congestion in Packet Switching Networks: The Case of a Single Congested Node*, IEEE/ACM Transaction on Networking, 1:6(1993), pp. 693–708.

[6] D. W. Clarke and P. J. Gawthrop, *Self Tuning Controller*, Proceedings of IEE, 122:9(1975), pp. 929–934.

[7] M. Hassan and H. Sirisena, *Optimal Control of Queues in Computer Networks*, in: IEEE ICC, pp. 637–641, 2001.

[8] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 3rd edition, 1998.

[9] V. Jacobson, *Congestion Avoidance and Control*, ACM Computer Communication Review, 18(1998), pp. 314–329.

[10] S. Kalyanaraman, R.Jain, S. Fahmy, R Goyal, and B. Vandalore, *The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks*, IEEE/ACM Transactions on Networking, 2000.

[11] A. Kolarov and G. Ramamurthy, *A Control-Theoretic Approach to the Design of an Explicit Rate Controller for ABR Service*, IEEE/ACM Transactions on Networking, 7:5(1999), pp. 741–753.

[12] C. E. Rohrs, *A Control Engineers Look at ATM Congestion Avoidance*, in: IEEE GLOBE-COM, pp. 1089–1094, 1995.

[13] F. G. Shinskey, *Process Control*, McGraw-Hill, 2nd edition, 1979.

[14] W. Willinger, M. S. Taqqu, R. Sherman, and D.V. Wilson, *Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level*, IEEE/ACM Transactions on Networking, 5:1(1997), pp. 71–86.