AQM GENERALIZED NYQUIST STABILITY IN MULTIPLE BOTTLENECK NETWORKS [⋆]

D. Bauso [∗∗] D. Di Bernardo [∗] L. Giarré [∗] G. Neglia [∗∗∗]

[∗] *Dipartimento di Ingegneria dell'Automazione e dei Sistemi - Università di Palermo, Palermo, Italy, giarre@unipa.it*
[∗∗] *Dipartimento di Ingegneria Informatica,- Università di Palermo, Palermo, Italy, dariobauso@libero.it*
[∗∗∗] *Dipartimento di Ingegneria Elettronica,- Università di Palermo, Palermo, Italy, giovanni.neglia@tti.unipa.it*

Abstract: The influence of multiple bottlenecks on the stability of Active Queue Management (AQM) controllers, usually configured on a single bottleneck basis is discussed. We consider a network scenario where RED is configured at each router according to previously developed control theoretic techniques. These configuration rules assure stability in a single bottleneck scenario. We show that instability may arise when two links become congested. We justify this result through a multiple bottleneck model using the Generalized Nyquist stability criterion *Copyright[©] 2005 IFAC.*

Keywords: Stability analysis, Communication Networks, Control Theory

1. INTRODUCTION

AQM algorithms usually rely on some heuristics and their performances appear to be highly dependent on the considered network scenario (see, e.g., (May *et al.*, 2000; Firoiu and Borden, 2000), as regards the RED algorithm).

Our research has been motivated by the consideration that the distributed fashion of TCP flows control across the network has not been explicitly considered up to now. As a matter of fact TCP flows may turn to be controlled at the same time by two or more nodes acting independently according to their AQM settings. According to our opinion, this can hardly affect AQM algorithms performance. In particular, we propose a counterexample to show that RED controllers, configured according to (Hollot *et al.*, 2001), do not prevent from instability if two or more nodes face congestion at the same time (this is referred to as *multiple bottleneck scenario*). In our papers (Neglia *et al.*, 2004) instability in multiple bottleneck has been tackled by considering a distributed Multi-Input Multi-Output model, whose stability has been studied considering the poles of the rational Linear Time Invariant model obtained through linearization and Padé approximation for time delays. Here the MIMO linearized model is analyzed via the Generalized Nyquist stability criterion.

2. SINGLE BOTTLENECK MODEL

The starting point in (Hollot *et al.*, 2001) is the model described by the following coupled, nonlinear differential equations:

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))}p(t-R(t)) \quad (1)$$

$$\dot{q}(t) = \frac{W(t)}{R(t)}N(t) - 1_{q(t)}C$$

where $1_q = 1$ if $q > 0$, $1_q = 0$ otherwise. Symbols used in the model above are summarized in the following table.

W	expected TCP window size (packets);
q	expected queue length (packets);
R	round-trip time;
C	link capacity (packets/sec);
R_0	propagation delay (secs);
N	load factor (number of TCP sessions);
p	probability of packet drop;

The first equation represents the TCP window, that increases by one every round trip time, and halves when a packet loss occurs. Packet loss rate is computed as the dropping probability times the number of packets sent per time unit. The round trip time is related to the propagation delay and the queue occupancy by the following relation: $R = R_0 + \frac{q}{C}$. The second equation represents the variation of queue occupancy as the difference between the input traffic and the link capacity.

AQM schemes determine the relation between the dropping probability and the nodes congestion status.

Here we considered RED as AQM scheme. RED configuration is specified through four parameters: the minimum and the maximum threshold (THR_{min}, THR_{max}), the maximum dropping probability in the region of random discard P_{max}, and the memory coefficient w_q. RED can be modelled by the following equations:

$$\dot{x}(t) = -kx(t) + kq(t) \quad (2)$$

$$p(x) = \begin{cases} 0, & 0 \leq x < THR_{min} \\ \dfrac{(x-THR_{min})P_{max}}{THR_{max}-THR_{min}}, & THR_{min} \leq x < THR_{max} \\ 1, & THR_{max} \leq x, \end{cases}$$

where $k = -\ln(1-\alpha)/\delta$ and δ is the time between two queue samples. The time interval δ can be assumed to be equal to $1/C$ for a congested node.

The linearized system (TCP sources, congested node queue and AQM controller) can be represented by the block diagram of Fig. 1. In the block diagram $L_{RED} = P_{max}/(THR_{max} - THR_{min})$.

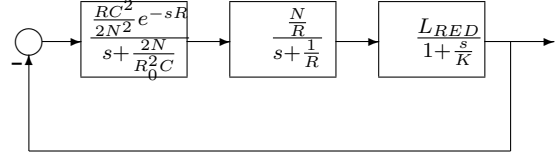The open-loop transfer function of the system in Fig. 1 is:

Fig. 1. Linearized control system for a single bottleneck scenario

$$H(s) = \frac{L_{RED}\frac{(RC)^3}{(2N)^2}e^{-sR}}{\left(1+\frac{s}{k}\right)\left(1+\frac{s}{\frac{2N}{R^2C}}\right)\left(1+\frac{s}{\frac{1}{R}}\right)} \quad (3)$$

In (Hollot *et al.*, 2001) the authors present RED configuration rules, that guarantee the stability of the linear feedback control system in Fig. 1 for $N \geq N^-$ and $R_0 \leq R^+$.
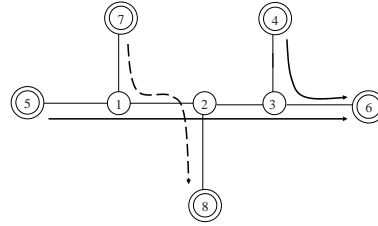

Fig. 2. Network topology

3. AN INSTABILITY EXAMPLE

In this section we show the example of a network where each router has been configured so that stability is guaranteed when congestion occurs at a single link, but instability arises when there are two bottlenecks at the same time. We consider a

Table 1. Network Parameters

Link	Capacity (Mbps)	Propagation Delay (ms)
5-1	20	2.1
1-2	10	10
2-3	20	2.1
4-3	20	19.1
3-6	10	10
7-1	20	9.4
2-8	20	9.4

parking lot network whose topology is depicted in Fig. 2. The capacity and the propagation delay of each link are reported in Table 1. Packet size is 1500 bytes. Links between nodes 3 and 6 and between nodes 1 and 2 will play the role of bottlenecks. The RED algorithm is deployed at nodes 3 and 1, respectively to manage the output queues for the link $3-6$ and $1-2$. In what follows we refer to these buffers simply as node 3 buffer and node 1 buffer, without specifying the link. We consider TCP flows aggregates from node 5 to node 6, from node 4 to node 6 and from node 7 to node 8. We indicate the number of flows of these aggregates respectively N_5, N_4 and N_7. Our RED configuration relies on the control theoretic

analysis of RED presented in (Hollot *et al.*, 2001). Nevertheless, we do not adopt exactly the configuration rules proposed there, since their high stability margins do not allow simple counterexample. Then, we verify RED-configuration stability directly through the Nyquist plot of the open loop transfer function. We recall that the Nyquist criterion allows one to study the stability of the closed loop system through the polar plot of the open loop transfer function $H(j\omega)$. For the functions we are interested in, the closed loop system is stable if and only if the plot does not encircle the point $(-1,0)$. We choose $THR_{min} = 2$, $THR_{max} = 20$, $P_{max} = 9\%$, and $w_q = 0.0017$. The analysis in (Hollot *et al.*, 2001) shows that stability increases as the number of flows increases and the RTT decreases. Given a network scenario these quantities are not independent because the queuing delay, and hence the RTT, depends on the number of flows. In particular the higher the number of flows the bigger the queue. In fact in order to evaluate the equilibrium point, we can impose the derivatives in equation system (1) equal to zero, obtaining

$$pW^2 = 2, \quad \frac{W}{R}N = C, \quad R = R_0 + \frac{q}{C},$$

and from equation system (2) considering that $q > THR_{min}$:

$$p = L_{RED}(q - THR_{min}).$$

It follows that the equilibrium queue value has to satisfy the following equation

$$(q - THR_{min})^2(q + CR_0)^2 = 2\frac{N^2}{L_{RED}^2}. \quad (4)$$

It is easy to see that as N increases, q increases. The previous equation shows also that the most critical condition is the lowest number of flows. Indeed q is approximately proportional to $\sqrt{N}$, hence the gain of $H(s)$ is proportional to $1/\sqrt{N}$. In conclusion we expect that if stability is guaranteed for a value N^- with a certain gain margin, then the gain margin is greater for $N > N^-$. Given the RED configuration, if there are $N_5^- = N_4^- = 3$ flows (and $N_7^- = 0$) link $3-6$ is congested, the average queue is $q = 7$ packets. The maximum RTT is equal to 69 ms and the open loop transfer function $H(j\omega)$ does not encircle the point $(-1, 0)$. Hence the system is stable, in particular the gain margin is equal to 1.35. This gain margin assures stability even if flows number reduces to 5 and if the average queue length increases to 12 packets. As it is expected, if $N_5 > N_5^-$ and $N_6 > N_6^-$ the gain margin increases. For example it is equal to 2.5 when $N_5 = N_4 = 5$. Link bandwidths and propagation delays are such that the path from 4 to 6 and that from 7 to 8 have the same characteristics. Hence the same numerical

results hold if only the link $1 - 2$ is congested due to aggregate 5 and aggregate 7, while $N_4 = 0$.

Let us assume that both links are congested and $N_5 = N_5^-$, $N_4 = N_4^-$ and $N_7 = N_7^-$. If we evaluate the new equilibrium point according to following equations, it holds $q_4 = q_2 \approx 5.48$. In comparison to the above situation, the RTT of aggregate 5 increases due to queueing delays at both node 3 and node 1 buffers, but the maximum RTT is always lower than 69 ms. Hence the *local* stability conditions are satisfied and we would expect the network to be stable and the gain margin should be even greater if $N_5 \geq N_5^-$, $N_4 \geq N_4^-$ and $N_7 \geq N_7^-$. Conversely we are going to show that a refined two-bottleneck model predicts instability.

3.1 Two Bottleneck Model

We extend the single bottleneck congestion model to the case of two congested nodes. With reference to the network topology depicted in Fig. 2, according to notations introduced in (Han *et al.*, 2004) we obtain:

$$\begin{cases} \dot{W}_4 = \frac{1}{R_4} - \frac{W_4^2}{2R_4}p_3(t - \overleftarrow{R}_{34}) \\ \dot{W}_5 = \frac{1}{R_5} - \frac{W_5^2}{2R_5}(p_3(t - \overleftarrow{R}_{35}) + p_1(t - \overleftarrow{R}_{15})) \\ \dot{W}_7 = \frac{1}{R_7} - \frac{W_7^2}{2R_7}p_1(t - \overleftarrow{R}_{17}) \\ \dot{q}_3 = \\ -C_3 I_{q>0} + \frac{N_4}{R_4}W_4(t - \overrightarrow{R}_{34}) + \frac{N_5}{R_5}W_5(t - \overrightarrow{R}_{35}) \\ \dot{q}_1 = \\ -C_1 I_{q>0} + \frac{N_5}{R_5}W_5(t - \overrightarrow{R}_{15}) + \frac{N_7}{R_7}W_7(t - \overrightarrow{R}_{17}) \end{cases} \quad (5)$$

where W_j is the average window of the flows originating at node j, p_i is the dropping probability at node i buffer, $\overleftarrow{R}_{ij}$ represents the backward delay from node i buffer to source j (including queuing delay) and $\overrightarrow{R}_{ij}$ the forward delay from source j to node i buffer and $I_{q>0}$ represents a logical function that is equal to 1 if the queue $q > 0$ and zero otherwise. For sake of simplicity in equation system (5), the time dependence is indicated only for delayed function values. Now, we linearize the Model 5 around the equilibrium point $(\hat{W}_4, \hat{W}_7, \hat{W}_5, \hat{p}_3, \hat{p}_1)$ that is the solution of the following system:

$$\begin{cases} W_4^2 p_3 = 2 \\ W_5^2(p_1 + p_3) = 2 \\ W_7^2 p_1 = 2 \\ \frac{W_4}{R_4}N_4 + \frac{W_5}{R_5}N_5 = C_3 \\ \frac{W_5}{R_5}N_5 + \frac{W_7}{R_7}N_7 = C_1 \\ p_3 = L_{RED3}(q_3 - THR_{min}) \\ p_1 = L_{RED1}(q_1 - THR_{min}) \\ R_4 = R_{40} + \frac{q_3}{C_3} \\ R_7 = R_{70} + \frac{q_1}{C_1} \\ R_5 = R_{50} + \frac{q_1}{C_1} + \frac{q_3}{C_3} \end{cases} \quad (6)$$
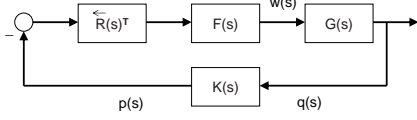
Fig. 3. Linearized control system for the multiple
bottleneck scenario

where R_{k0} is the round trip time experienced from
source k when no congestion is present, and we
omitted the chapeau to simplify notation. In the
Laplace domain we obtain the following relations:

$$w(s) = -F(s)\overleftarrow{R}(s)^T p(s) \qquad (7)$$

$$q(s) = (sI + \Omega)^{-1}\overrightarrow{R}(s)NT^{-1}w(s) = G(s)w(s) \quad (8)$$

where $N = diag\{N_j\}$ and $T = diag\{R_j\}$ for
$j = 4, 7, 5$ and

$$F(s) = \begin{bmatrix} \dfrac{W_4^2}{s + \frac{2}{W_4 R_4}} & 0 & 0 \\ 0 & \dfrac{W_7^2}{s + \frac{2}{W_7 R_7}} & 0 \\ 0 & 0 & \dfrac{W_5^2}{s + \frac{2}{W_5 R_5}} \end{bmatrix} \quad (9)$$

$$\Omega = \begin{bmatrix} \dfrac{N_5 W_5}{R_5^2 C_3} + \dfrac{N_4 W_4}{R_4^2 C_3} & \dfrac{N_5 W_5}{R_5^2 C_1} \\ \dfrac{N_5 W_5}{R_5^2 C_3} & \dfrac{N_5 W_5}{R_5^2 C_1} + \dfrac{N_7 W_7}{R_7^2 C_1} \end{bmatrix}$$

In the delay matrices $\overleftarrow{R}(s)$ and $\overrightarrow{R}(s)$ the columns
correspond to sources 4, 7 and 5, and the rows to
buffer 3 and 1. In particular

$$\overleftarrow{R}(s) = \begin{bmatrix} e^{-s\overleftarrow{R}_{34}} & 0 & e^{-s\overleftarrow{R}_{35}} \\ 0 & e^{-s\overleftarrow{R}_{17}} & e^{-s\overleftarrow{R}_{15}} \end{bmatrix}$$

$$\overrightarrow{R}(s) = \begin{bmatrix} e^{-s\overrightarrow{R}_{34}} & 0 & e^{-s\overrightarrow{R}_{35}} \\ 0 & e^{-s\overrightarrow{R}_{17}} & e^{-s\overrightarrow{R}_{15}} \end{bmatrix}$$

Note that $\overrightarrow{R}(s = 0) = \overleftarrow{R}(s = 0)$ is the so called
routing matrix of our scenario.

$$R(0) = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad (10)$$

The RED AQM control law that computes the
packet marking probability p as a function of
measured queue length q is $p(s) = K(s)q(s)$,
where

$$K(s) = \begin{bmatrix} \dfrac{k_3 L_{RED3}}{s + k_3} & 0 \\ 0 & \dfrac{k_1 L_{RED1}}{s + k_1} \end{bmatrix} \qquad (11)$$

The overall feedback loop is depicted in Fig. 3.

3.2 Stability Analysis

Many methods have been developed in the past
fifty years to analyze the stability of time-delay
system, (i.e. see (Gu and Niculescu, 2003) for a
detailed survey), but for our purpose, the clas-
sical result of the Generalized Nyquist stability
criterion (Desoer and Wang, 1980) is sufficient
and gives us an easy analytical tool. We recall it
hereafter:

Theorem 1. (**Generalized Nyquist Criterion**)
If the open loop matrix $L(s)$ has P_0 unstable
poles, then the closed-loop system with return ra-
tio $-kL(s)$ is stable if and only if the characteristic
loci of $kL(s)$, taken together, encircle the point
$(-1, 0)$, P_0 times anticlockwise.

As proved in (Desoer and Wang, 1980), the above
theorem is valid not only for the *lumped* case when
$L(s)$ is a square rational transfer matrix $L(s)$, but
it has been extended also to the *distributed* case.
In the lumped case, $L(s)$ is factorized in two part,
and P_0 are the poles of the rational part. In our
case, the open loop matrix is given by

$$L(s) = K(s)(sI + \Omega)^{-1}\overrightarrow{R}(s)NT^{-1}F(s)\overleftarrow{R}(s)^T$$

Applying the Generalized Nyquist criterion, we
analyze the characteristic loci of $L(s)$ in the case
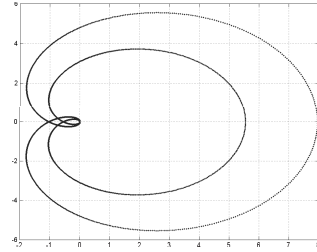of $N_4 = N_5 = N_7 = 3$, reported in Fig. 4. Since

Fig. 4. $L(s)$ characteristic loci in the case of $N_4 = N_5 = N_7 = 3$

the open loop is stable, and the characteristic
loci of $L(j\omega)$ encircle the point $(-1, 0)$, two times
anticlockwise, as shown in Fig. 5, the closed loop
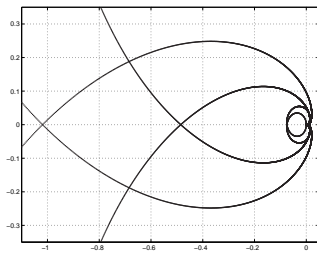system is unstable. This example shows the limits

Fig. 5. Particular of the $L(s)$ characteristic loci in
the case of $N_4 = N_5 = N_7 = 3$

of local AQM configuration ignoring the distributed nature of TCP flows control in a multiple bottleneck scenario. If we consider the configuration rules given in (Hollot *et al.*, 2001), instability probably does not arise in such a simple example, but there is a reduction of stability margins. This modifies the system dynamic response and reduces the system robustness with respect to the flows number and the round trip time variation.

4. SIMULATION RESULTS

In this section we investigate through simulations the previous analytical result. Firstly we are going to discuss what instability implies as regards network performance in the single bottleneck scenario. Secondly we show that the two bottleneck scenario with $N_5 = N_4 = N_7 = 3$ (unstable according to the analysis in Sec. 3.2) exhibits performance similar to those identified as unstable in the single bottleneck scenario. Finally we observe similar results even in other scenarios, which are stable according to the two-bottleneck model. Simulations were conducted through ns v2.1b9a (network simuator, n.d.). We used TCP Reno implementation.

4.1 Single Bottleneck

In order to analytically show how instability of the linear model concretely affects the network performance, we first present some results regarding the single bottleneck scenario. When dealing with simulation tests a first issue regards the choice of a metric able to catch potential instability phenomena. For example (Low *et al.*, 2002) shows the oscillations of the TCP window and a deterministic limit cycles in the average window, averaged over all the flows of the same aggregate, and (P. *et al.*, 2002) shows nonlinear phenomena, such as bifurcations, using Liapunov exponents as a measure. Following the same line as in (Hollot *et al.*, 2001) we will look at the oscillating nature of the queue length to distinguish between stable or unstable behaviors. Differently from those papers, our results suggest that the amplitude of queue oscillations is not significant by itself when RED is considered. As we are going to show it can be more appropriate to consider the amplitude of queue oscillations in relation to the average queue value.

Let us consider two aggregates, each one of five TCP flows ($N = 10$), entering the network through node 4 and node 5 with destination node 6 (solid lines in Fig. 2). The link between nodes 3 and 6 is congested. Fig. 6 shows the instantaneous queue occupancy time-plot for the buffer at node 3. RED should be able to keep the queue occupancy within the two thresholds (dotted lines).
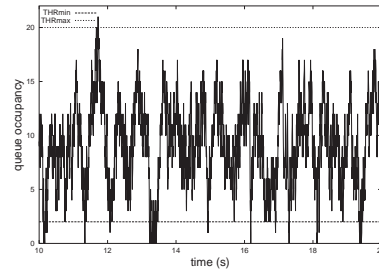

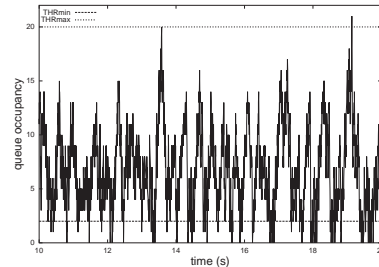Fig. 6. Instantaneous buffer occupancy with number of flows $N = 10$

Fig. 7. Instantaneous buffer occupancy with $N = 6$

Numerical results for the throughput and the queue are shown in Table 2. In particular the average queue size is in column "queue occupancy", while the standard deviation is in column "queue oscillation". The value in parentheses is the normalized queue oscillation, i.e. the ratio between the standard deviation and the average queue value.

Let us reduce the number of flows through the network and see if instability occurs as claimed in (Hollot *et al.*, 2001). In Fig. 7 the buffer occupancy is shown to revisit with a higher frequency the regions associated to buffer underload (out of RED thresholds). As the total flow number decreases from 10 to 6 we note that i) the throughput over the link $3 - 6$ ($Thr_5 + Thr_4$) reduces from 9.91 Mbps to 9.74, ii) both the average queue occupancy and the oscillation amplitude decrease, respectively from 9.06 to 6.71 and from 4.72 to 4.09, and iii) the normalized standard deviation, i.e. the ratio between standard deviation and mean, increases from 0.52 to 0.61.

If we reduce drastically the number of flows to 2, the above RED configuration, turns to be too aggressive, which is evidenced by higher frequencies of buffer occupancy oscillations and further reduction of the throughput. Even longer periods, where buffer is underloaded results from Fig. 8. In general, experimental results show that instability predicted by the model in (Hollot *et al.*, 2001) leads to reduced link utilization and queue oscillation and higher normalized oscillations (higher jitter in percentage). Hence, while one could expect larger queue oscillations when the number of flows decreases and the stability margins decrease, this
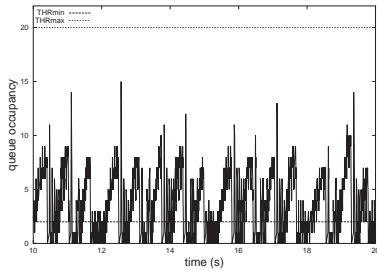
Fig. 8. Instantaneous buffer occupancy with $N = 2$

intuition is not confirmed by simulative results. The explanation is straightforward: RED couples queue length and loss probability, in particular a lower number of flows needs a lower global dropping probability, hence a lower average queue (from a control theoretic point of view one says that the RED controller has steady state regulation errors). As the average queue size decreases, the physical constraint of positive queue values can determine smaller oscillations. On the contrary the normalized standard deviation looks to reflect system change from stability to instability. So in what follows we focus on it to analyze instability phenomena.

4.2 Two Bottlenecks

Now we consider an additional aggregate entering the network from node 7, with destination node 8 (dotted line in Fig. 2). In Fig. 9 node 3 buffer appears to be empty more often than in Fig. 7.
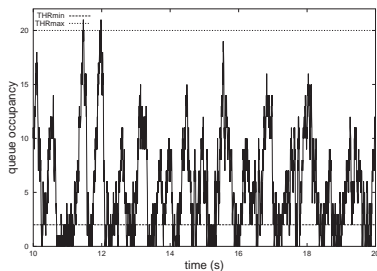

Fig. 9. Instantaneous node 4 buffer occupancy in a two bottleneck scenario

The numerical values stored in Table 2 support quantitatively our claims rising from Fig. 9.

In particular the normalized oscillation values of queue 3 is 0.76, equal to the value stored in the fifth row, corresponding to a single bottleneck high instability scenario due to a low number of flows ($N_5 + N_4 = 2$). Note that, though the number of flows at each node and the flow round trip time should assure stable operation, instability arises due to the traffic aggregate from 5 to 6, which traverses both the congested links. Moreover results in Table 2 indicate that even stable scenarios (according to the two bottleneck model in Sec. 3) like that corresponding to $N_5 = N_4 = N_7 = 5$ cannot be easily distinguished by an unstable scenario. This remark suggests that the study through linear models could provide not significative insight on the stability of network with TCP traffic.

REFERENCES

Desoer, C. A. and Y.T. Wang (1980). On the Generalized Nyquist Stability Criterion. *IEEE Trans. on Automatic Control* **25**(2), 187–195.

Firoiu, V. and M. Borden (2000). A study of Active Queue Management for Congestion Control. *Proc. of IEEE Infocom.*

Gu, K. and S. Niculescu (2003). Survey on Recent Results in the Stability and Control of Time-Delay Systems. *ASME J. of Dyn. Syst., Meas., and Control* **125**, 158–165.

Han, H., C. V. Hollot, Y. Chait and V. Misra (2004). Tcp Networks Stabilized by Buffer-Based AQMs. *Proc. of IEEE Infocom.*

Hollot, C. V., V. Misra, D. Towsley and W. Gong (2001). A Control Theoretic Analysis of RED. *Proc. of IEEE Infocom.*

Low, S.H, F. Paganini, J. Wang, S. Adlakha and J.C. Doyle (2002). Dynamics of TCP/RED and a scalable control. *Proc. of IEEE Infocom* **1**, 239 – 248.

May, M., T. Bonald and J. C. Bolot (2000). Analytic Evaluation of RED Performance. *Proc. of IEEE Infocom.*

Neglia, G., D. Bauso and L. Giarré (2004). About the Stability of Active Queue Management Mechanism. *Proc. of ACC.*

network simuator (n.d.). The network simulator, ns-2. *http://www.imashi.edu/nsnam/ns/.*

P., Ranjan, E. H. Abed and R. J. La (2002). Nonlinear instabilities in TCP-RED. *Proc. of IEEE Infocom* **1**, 249– 258.

Table 2. Numerical Results

N_5	N_5	N_7	Thr_5	Thr_4	Thr_7	$queue_3$ occupancy	$queue_3$ oscillation	$queue_1$ occupancy	$queue_1$ oscillation
5	5	0	5.12	4.79	-	9.06	4.72 (0.52)	0.95	0.066 (0.27)
4	4	0	5.09	4.75	-	7.93	4.48 (0.56)	0.95	0.057 (0.25)
3	3	0	5.12	4.62	-	6.71	4.09 (0.61)	0.95	0.067 (0.27)
2	2	0	5.06	4.49	-	5.32	3.57 (0.67)	0.97	0.075 (0.28)
1	1	0	4.80	4.47	-	3.49	2.64 (0.76)	0.97	0.085 (0.30)
5	5	5	3.40	6.41	6.33	7.76	5.29 (0.68)	7.83	5.47 (0.70)
4	4	4	3.43	6.28	6.20	6.85	4.98 (0.73)	6.93	5.13 (0.74)
3	3	3	3.62	5.99	5.91	5.76	4.38 (0.76)	5.94	4.59 (0.77)
2	2	2	3.80	5.60	5.52	4.72	4.01 (0.85)	4.75	4.06 (0.85)