Performance Evaluation of Robust Active Queue Management Schemes in IP Networks

Sai S. Oruganti\(^+\) and Michael Devetsikiotis\(^++\)
\(^+\)Operations Research Program and \(^++\)ECE Dept
North Carolina State University
Raleigh, NC 27695-7913

May 14, 2003

*IEEE International Conference on Communications, Anchorage, Alaska*
Outline of the Presentation

- Objective and Motivation - Our Contribution
- Characteristics of Robust AQM techniques
- Characteristics and Prediction of TCP Traffic
- Definition and Quantification of Robustness
- Logical Modules of an AQM scheme - PCC - DTMW - wDTMW - PL-EFAQL - LMSFQO

- Test Network Configuration - Network Topology - Network Traffic

- Results (Robustness - Delay jitter - Goodput) and Conclusions
Objective and Motivation

- Objectives of an Active Queue Management (AQM) scheme are
  - Maintain small queue size at the router - to ensure small queueing delays
  - Maintain high link utilization - to ensure high goodput
  - Pro-actively notify connections to reduce window size - to avoid global synchronization

- Motivation: Network traffic shows variation over timescales. To maintain efficiency need to regularly fine tune the AQM parameters. Prefer robust AQM schemes over efficient ones
Our Contribution

• Analyze AQM schemes based on their logical operations

• Exploit long-range dependence of network traffic to predict the future arrival rates

• Study AQM schemes for their robustness and tight delay bounds in the presence of long-range dependence.

• Monitor the goodput to make sure that better robustness and delay bounds are not at the expense of goodput

• Compare performance with the traditional schemes
Characteristics of Robust AQM Techniques

• An AQM scheme that does not display *sustained loss of performance* under varying traffic environments can be considered as robust

• Robust AQM schemes should work efficiently under the following traffic environments
  *
  • Small number of high-bandwidth flows
  • Large number of low-bandwidth flows
  • Heterogeneous round trip times
  • Heterogeneous bandwidths due to upstream constraints (broadband vs. modem access rate constraints)
  • Heterogeneous flow durations
  • Non-stationary flow duration distribution
Characteristics of Robust AQM Techniques (contd.)

- Graph of Goodput vs. Average queueing delay for a DropTail scheme and two hypothetical AQM schemes, AQM1 and AQM2
Characteristics and Prediction of TCP Traffic

- TCP traffic is adaptive, bursty and chaotic
  - Adaptive nature exhibited by window scaling phenomenon
  - Cycles of packet streams followed by wait periods results in bursty behavior
  - Exponential back-off due to packet losses results in change of periodicity causing chaotic nature
- Traffic by multiple TCP sources is pseudo self-similar in nature
- Exploit this correlation to predict the future
- Prediction imparts robustness to AQM schemes
**Definition and Quantification of Robustness**

- Robustness is the insensitivity of output against small deviations in the assumptions.

- Robustness of AQM defined as the ability of a router to maintain its performance even under harsh changes in its environment.

- Our robustness index is defined as \( R_g = \frac{g_2 - g_1}{g_2} \)
  - \( g_1, g_2 \) are goodputs of the same AQM scheme under two different network traffic conditions.
  - \( R_g \) is a relative robustness index for the same average queueing delay.
  - Lower values of \( R_g \) mean better robustness.
Logical Modules of a Predictive AQM scheme

- Based on their operation AQM schemes can be dissected into two logical modules
  - Measurement module or *predictor* - Measures traffic intensity at regular intervals and captures future trend
  - Control module or *controller* - Makes packet marking decision based on input from predictor

- Study of robustness of AQM schemes as a combination of
  - 4 Predictors
  - 2 Controllers
Predictive Congestion Control (PCC)

- Divides the traffic arrival rates into 8 intervals based on its mean ($\mu$) and standard deviation ($\sigma$) values.

- Based on these intervals builds an $8 \times 8$ conditional probability matrix.

- Prediction is exploited after the training period.

- The conditional probability matrix is updated at the end of every update interval.
Double Threshold Moving Window (DTMW)

- A type of low-pass but nonlinear finite impulse response (FIR) filter
- Consists of a moving window of size $N$ where each slot in the window records the traffic rate for that slot

Oruganti & Devetsikiotis
Weighted Double Threshold Moving Window (wDTMW)

- Same operation as DTMW except that the observations are weighted. Higher priority is assigned to the more recent observations and lower priority to the older ones.

- The packet marking decision is taken based on $P$ and $T2$

\[
P = \sum_{i=N}^{1} (2^{i-1} \times X_i) = (2^{N-1} \times X_N) + (2^{N-2} \times X_{N-1}) + \ldots + (2^{0} \times X_1)
\]

(1)

where $X_n$ is bit status of slot $n$ and is defined as:

\[
X_n = \begin{cases} 
1 & \text{if arrival rate at slot } n \geq T1 \\
0 & \text{otherwise}
\end{cases}
\]
Phase Lag (PL)

- Representation of ideal predictor with zero prediction error due to lookahead into simulation model
- Traffic generated by end nodes enters into the simulation system after \( h \) slots. Future arrivals in next \( h \) slots are known with zero error
- Used as benchmark to compare the performance of other predictors

Oruganti & Devetsikiotis
Estimated Future Average Queue Length (EFAQL)

- Its control action is similar to EWMA

- The estimated future average queue length is calculated as

\[
Q_{Avg}(t = n) = w_1 * Q_{Avg}(t = n - 1) + \\
w_2 * Q_{Avg}(t = n + M) + \\
(1 - w_1 - w_2) * Q_{Inst}(t = n)
\]  \hspace{1cm} (2)

\(w_2\) is the weight attached to the future average queue length
Estimated Future Average Queue Length (EFAQL)

- \( Q_{avg}(t = n + M) \) is calculated recursively as:

\[
Q_{Avg}(t = n + M) = (1 - u1) \cdot Q_{Avg}(t = n + M - 1) + \\
u1 \cdot Q_{Inst}(t = n + M)
\] (3)

- Future instantaneous queue length is,

\[
Q_{Inst}(t = n + M) = Q_{Inst}(t = n + M - 1) \\
+ \widehat{Ar}(t = n + M) - Se(t = n + M) \\
- Dr(t = n + M)
\] (4)
Least Mean Square Fixed Queue Occupancy (LMSFQO)

- This controller is represented as

$$\inf J = \sum_{j=1}^{M} (Q(t = n+j) - Q_{fixed})^2$$

(5)

$Q(t = n+j)$ is actual queue occupancy

$Q_{fixed}$ is target fraction of the maximum queue size
Least Mean Square Fixed Queue Occupancy (LMSFQO)

The control action for the next sampling interval is:

\[
p = \begin{cases} 
0 & Q < Se + Q_{fixed} - \hat{Ar} \\
\frac{Q + \hat{Ar} - Se - Q_{fixed}}{\hat{Ar}} & Se + Q_{fixed} - \hat{Ar} < Q < Se + Q_{fixed} \\
1 & Q > Se + Q_{fixed}
\end{cases}
\]

- \( p \) - drop probability for the next interval, \( k + 1 \)
- \( Ar \) - Arrival rate at current interval, \( k \)
- \( Se \) - Service rate at current interval, \( k \)
- \( \hat{Ar} \) - Estimated arrival rate of next interval, \( k + 1 \).
**Network Topology**

- **Network Configuration**
  - Link capacity - 20Mbps, Buffer capacity - 100 packets
  - OFF period $\sim$ *Pareto*(1.21, 2)*sec, File size $\sim$ *Pareto*(1.21, 2)*$10,000$ Bytes
  - RTT $\sim$ *$U$(80, 120)* msec, Sampling time - 10msec
Network Traffic

- Modelled traffic is http traffic, the dominant network traffic
- TCP Reno with delayed ACK with assumption that ACKs are never lost

Data Collection and Analysis

- Several simulation runs were conducted to calculate the 95th percentile confidence intervals of goodput
- Steady-state period of buffer occupancy was determined by moving average method
- Delay jitter was calculated during the steady-state period to remove bias during the warm-up period
Results for Robustness

<table>
<thead>
<tr>
<th>(Robustness)</th>
<th>EFAQL</th>
<th>LMSFQO</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCC</td>
<td>1.16 (83)</td>
<td>18.62 (140)</td>
</tr>
<tr>
<td>DTMW</td>
<td>5.32 (2.86)</td>
<td>3.27 (2.56)</td>
</tr>
<tr>
<td>mDTMW</td>
<td>5.32 (9.12)</td>
<td>3.23 (4.1)</td>
</tr>
<tr>
<td>PL</td>
<td>5.75 (0.388)</td>
<td>6.67 (30.3)</td>
</tr>
</tbody>
</table>

Table 1: Robustness in $10^{-3}$

- Robustness for a fixed RTT setup (in parenthesis for variable RTT setup)
- Small number of high-bandwidth flows: 30 sources, $W_{max}$ is 128
- Large number of low-bandwidth flows: 100 sources, $W_{max}$ is 16
- Value of robustness is for an average queueing delay of 0.2 seconds

Oruganti & Devetsikiotis
Results for Robustness

- Robustness of DropTail scheme is $4 \times 10^{-3}$ ($8 \times 10^{-3}$).

- Real traffic scenarios (variable RTT setup) are less robust than their ideal counterparts (fixed RTT setup) because of a high degree of variability inherent in the real traffic.

- AQM schemes with stochastic based traffic prediction (PCC) are usually more robust when the control is based on EWMA of queue length. Similarly, AQM schemes with traffic prediction based on finite impulse response filters (DTMW) are more robust when the control is based on fixed queue occupancy.

- Robustness of Predictive AQM schemes is correlated with goodput performance. More robust schemes have higher goodput.
- Graph of Goodput vs. Average queueing delay for a fixed RTT setup
- Similar to the graph comparing two AQM schemes
## Results for Delay Jitter

<table>
<thead>
<tr>
<th>(Delay jitter)</th>
<th>EFAQL</th>
<th>LMSFQO</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCC</td>
<td>$21.83 \pm 13.37 \times 10^{-6}$</td>
<td>$56.4 \pm 286.18 \times 10^{-6}$</td>
</tr>
<tr>
<td>DTMW</td>
<td>$41.51 \pm 548.08 \times 10^{-6}$</td>
<td>$36.9 \pm 1606.27 \times 10^{-6}$</td>
</tr>
<tr>
<td>wDTMW</td>
<td>$41.63 \pm 1207.6 \times 10^{-6}$</td>
<td>$57.5 \pm 675.04 \times 10^{-6}$</td>
</tr>
<tr>
<td>PL</td>
<td>$53.76 \pm 783.13 \times 10^{-6}$</td>
<td>$95.0 \pm 3349.7 \times 10^{-6}$</td>
</tr>
</tbody>
</table>

Table 2: Delay jitter in units of $10^{-6}$ seconds

- The average queue occupancy with EFAQL controller is between 5 and 15 packets and with LMSFQO controller is 90 packets.

- With a proper selection of the weight of future observations Predictive AQM schemes achieve better delay bounds than RED ($85.3 \pm 9548.5 \times 10^{-6}$) or DropTail ($138.1 \pm 7934.8 \times 10^{-6}$).
Results for Goodput

<table>
<thead>
<tr>
<th></th>
<th>RED ($w^2 = 0$)</th>
<th>EFAQL</th>
<th>LMSFQO</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCC</td>
<td>123367.4 ± 223</td>
<td>123543.5 ± 257</td>
<td>119230.2 ± 542</td>
</tr>
<tr>
<td>DTMW</td>
<td>123892.4 ± 82</td>
<td>123912.8 ± 111</td>
<td>123825.4 ± 109</td>
</tr>
<tr>
<td>wDTMW</td>
<td>123818.3 ± 117</td>
<td>123884.2 ± 90</td>
<td>123811.5 ± 121</td>
</tr>
<tr>
<td>PL</td>
<td>122855.4 ± 319</td>
<td>122946 ± 260</td>
<td>123114.9 ± 218</td>
</tr>
</tbody>
</table>

Table 3: Goodput in units of 1000 bytes

- Buffer size is 100 packets. RED parameters as suggested by Floyd et. al.
- Predictive AQM schemes, for most cases, display mild improvement in goodput values over the traditional ones
- Goodput results only used to show that robust AQM schemes do not achieve drastically worse goodput performance
• Graph of Goodput vs. $w_2$ for a fixed RTT setup

• Goodput increases with prediction. Excessive dependence worsens the performance
Conclusions

• Analysis of AQM schemes
  ★ Analyzed AQM schemes as combination of a measurement module and a control module
  ★ Allows us to observe carefully the effect of each module on the performance of the AQM scheme
  ★ Implemented different predictors and controllers and compared the performance of an AQM scheme for various combinations

• Study of robustness of AQM schemes
  ★ Robust schemes will show little sensitivity in performance to changes in network traffic
Conclusions

• Our Contribution
  ✷ Careful control decisions of predictive AQM schemes achieve better delay bounds
  ✷ AQM schemes with stochastic based traffic prediction are usually more robust when the control is based on EWMA of queue length. Similarly, AQM schemes with traffic prediction based on finite impulse response filters are more robust when the control is based on fixed queue occupancy
  ✷ More robust AQM schemes usually have higher goodput for comparable average queueing delay values
  ✷ Goodput performance is worse than RED when current packet marking decisions are too heavily relied on future prediction
Acknowledgments

This research is supported by the Center for Advanced Computing and Communication - North Carolina State University, as a Core project. The authors thank Steve Blake and Zsolt Haraszti from Ericsson IPI, Raleigh, for their comments and assistance.