Adaptive REM: random exponential marking with improved robustness

L. Tan, G. Peng and S. Chan

To improve the robustness of the random exponential marking (REM) scheme, it is proposed to apply an update function to the dynamics of the price in REM. By simulations, the performance improvement over REM and another approach of making REM adaptive is demonstrated.

Introduction: As a representative active queue management (AQM) scheme, random exponential marking (REM) [1] faces the challenge of dynamically tuning its parameters to satisfy the stability requirement under various network conditions adaptive to network conditions [2, 3]. In [4], Deng et al. proposed an adaptive control mechanism for the proportional and integral (PI) controller based AQM (PI-AQM) developed in [3]. The mechanism improves system stability and performance under changing network conditions by applying an update function to the packet dropping probability. Unfortunately, although REM also belongs to the family of PI controllers [4], due to the fact that the proportional control gain \(K_p\) of REM is a decreasing function of the equilibrium point of marking probability, the mechanism proposed in [4] is not suitable to enable REM to adapt its control parameters to the changing of network conditions.

To improve the robustness of REM, in this Letter we propose to apply an update function to the dynamics of the price in REM. The enhanced REM, termed as ‘adaptive REM’, can adapt its control parameters to various network conditions.

Model and analysis: Consider a single link of capacity \(C\) with \(N\) TCP flows traversing it. Let \(R\) be the round-trip delay of each flow. The TCP/REM dynamics can be modelled as the following nonlinear delay differential equation

\[
\begin{align*}
\dot{y}(t) &= \frac{N}{C} - \beta \frac{y(t)(t - R)}{N} p(t - R) \\
q(t) &= y(t) - C
\end{align*}
\]

where \(y(t)\) is the aggregate arrival rate at the link, \(q(t)\) is the queue length, and \(p(t)\) is the marking function which gives the probability of packets being marked at the link, at time \(t\) respectively, and \(\beta = 2/3\). The price function \(m(t)\) and marking function of REM are described by

\[
\begin{align*}
\dot{p}(t) &= \gamma (q(t) - q_0) + \dot{y}(t) - C \\
p(t) &= 1 - \phi^{-m(t)} \quad \text{where} \quad \gamma > 0, \quad 0 < \phi < 1, \quad q_0 > 0, \quad \text{and} \quad \Phi = q(0) < 1
\end{align*}
\]

where \(\gamma > 0\) and \(\phi > 0\) are small constants, \(\phi\) is an arbitrary constant greater than one and \(q_0\) is the target queue length. The equilibrium points of the above model are given by

\[
y_0 = C, \quad p_0 = \frac{N^2}{\beta(RC)}, \quad \mu_0 = \log_\phi \frac{\beta C^2 R^2}{\beta C^2 R^2 - N^2}
\]

We have the transfer function of REM [5] as

\[
G_c(s) = \gamma (1 - p_0) \log_\phi + \frac{\gamma (1 - p_0) \log_\phi}{T_s}
\]

\[
= K_p + \frac{K_p}{s}
\]

From [4], one observes that the TCP/REM system has better stability with larger \(N\), smaller \(R\) and \(C\); and/or smaller proportional gain \(K_p\), i.e. decreasing \(N\) or increasing \(R\) would make the system unstable. Even worse, decreasing \(N\) also decreases \(p_0\) and hence increases \(K_p\) because it can be seen from (3) that \(K_p\) monotonically decreases with \(p_0\). Therefore, the system will be made even more unstable. To maintain system stability as the network and traffic condition change dynamically, \(K_p\) should increase with \(N\) or \(1/R\). To this end, Deng et al. proposed an update function applied to the marking probability \(p\) in PI-AQM [4]. Unfortunately, since \(K_p\) monotonically decreases with \(p_0\) in REM, deploying the same mechanism as in [4] for REM still cannot eliminate the negative influence of decreasing \(p_0\) on the system stability. In adaptive REM, we introduce an update function applied to the dynamics of the price in REM. As shown below, it can eliminate the negative influence of decreasing \(p_0\) and improve the robustness of the TCP/REM system.

Adaptive REM: The relationship between the network parameters and performance objectives of REM has opened the ability to tune the algorithms for adaptive control of the network performance. We introduce an update function \(m(t)\) and \(m(x) = 1/(kx)\), where \(k\) is defined as a monotonically increasing function and \(m(x)\) is the derivative of \(m(x)\).

First as in [4], we apply an update function to the dropping probability in REM. We refer to this modified REM as \(p\)-REM. Then, based on (2), we have

\[
m(p(t)) = 1 - \phi^{-m(t)}
\]

where \(\phi\) is a constant. The transfer function of \(p\)-REM is given by

\[
G_c'(s) = \frac{\gamma (1 - p_0) \log_\phi + \frac{\gamma (1 - p_0) \log_\phi}{T_s}}{K_p + \frac{K_p}{s}}
\]

With \(p\)-REM, the controller gains are \(K_p = \gamma (1 - p_0) \log_\phi\) and \(K_p = \frac{\gamma (1 - p_0) \log_\phi}{T_s}\). A discrete implementation of (4) is as follows

\[
m(p(k)) = 1 - \phi^{-m(x)}
\]

where \(\mu(k)\) and \(p(k)\) are the sampled price and dropping probability, respectively, at \(t = kT\).

Now, for adaptive REM, the update function is applied to the dynamics of the price \(\mu\). Based on the price function of REM given by (2) and the relationship \(q(t) = y(t) - C\), we have

\[
m(\mu(t)) = \gamma q(t) + \frac{\gamma (1 - p_0) \log_\phi}{T_s}
\]

where \(\gamma, x\) and \(T\) are constants. The transfer function of adaptive REM is given by

\[
G_c'(s) = \frac{\gamma (1 - p_0) \log_\phi + \frac{\gamma (1 - p_0) \log_\phi}{T_s}}{K_p + \frac{K_p}{s}}
\]

With adaptive REM, the controller gains are \(K_p = \gamma (1 - p_0) \log_\phi\) and \(K_p = \frac{\gamma (1 - p_0) \log_\phi}{T_s}\). We can see that \(K_p\) is an increasing function of \(p_0\) that can reduce the negative influence of \(p_0\) on the system stability when the traffic loads and other conditions change.

A discrete implementation of (7) is as follows

\[
m(\mu(k + 1)) - m(\mu(k)) = \gamma \mu(k) + \gamma (k + 1)
\]

where \(\mu(k)\) and \(q(k)\) are the sampled price and queue length, respectively, at \(t = kT\) and \(\epsilon(k) = q(k) - q_0\).

Fig. 1 Changes of proportional gain according to various traffic loads, adaptive REM vs. \(p\)-REM and REM

- Adaptive REM, \(K_p = 2\sqrt{(\gamma \mu(1 - p_0) \log_\phi)}\)
- \(p\)-REM, \(K_p = 2\sqrt{(\gamma \mu(1 - p_0) \log_\phi)}\)
- REM, \(K_p = \gamma \log_\phi (1 - p_0)\)

The choice of the function \(I\) determines how rapidly \(K_p\) and \(K_p\) adapt to the system parameters. We set \(\frac{I(x)}{x} = 2\sqrt{x}\), so \(m(x) = \sqrt{x}\). We compare \(K_p\), \(K_p\) and \(K_p\) under changing network traffic loads \(N\) in

ELECTRONICS LETTERS 18th January 2007 Vol. 43 No. 2
and assuming $p_0$ is not greater than 0.5. As shown in Fig. 1, we can see that $K_p$ monotonically decreases with $N$, while $K_p'$ only slightly increases with $N$. However, $K_p'$ increases with $N$ more significantly than $K_p$. More specifically, for the considered range of $N$, $K_p$ changes in the range of $[10^{-4}, 1.2 \times 10^{-3}]$, while $K_p'$ changes in the range of $[0.1 \times 10^{-3}, 3.5 \times 10^{-3}]$. This suggests that adaptive REM has stronger capability of maintaining system stability under varying network conditions than p-REM.

Finally according to the function $m$, we can determine that the marking update equation in p-REM is

$$p(k) = (1 - \phi^{m(k)})^2$$

(10)

Also, the price update equation in adaptive REM is

$$\mu(k + 1) = (\sqrt{\mu(k)} + \gamma(\sigma(q(k) - q_0) + y(k) - C))^2$$

(11)

Simulation results: Simulations are conducted in the ns-2 simulator by using a commonly used dumb-bell topology to evaluate the performance of REM, p-REM and adaptive REM. The capacity of the bottleneck link is 3 Mbit/s. The capacity of other links is 10 Mbit/s. The propagation delays for the flows range uniformly between 50 and 240 ms, with average packet size being 1000 bytes. The buffer size of the router is 200 packets. For the three schemes, we set $\gamma$ as 0.001, $\sigma$ as 0.1 and $\phi$ as 1.001, and $q_0$ as 50 packets. The simulation time is 200 s. In each simulation there are initially 50 ftp (greedy) flows. At time $t = 50$ s, 80 new ftp flows join and then leave at time $t = 100$ s. Subsequently, at $t = 150$ s, we add other 40 ftp flows in the network again. The simulation results are shown in Fig. 2 and Table 1.

![Image](image-url)

Fig. 2 Instantaneous queue length of adaptive REM against p-REM and REM

Table 1: Comparison of mean and standard deviation of queue length, throughput and loss rate among three schemes (target queue length = 50)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Queue length (mean)</th>
<th>Queue length (SD)</th>
<th>Throughput (mean)</th>
<th>Loss rate (mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td>REM</td>
<td>89.9</td>
<td>57.9</td>
<td>97.4%</td>
<td>1.1%</td>
</tr>
<tr>
<td>p-REM</td>
<td>125.1</td>
<td>64.5</td>
<td>99.8%</td>
<td>3.8%</td>
</tr>
<tr>
<td>Adaptive REM</td>
<td>52.5</td>
<td>20.6</td>
<td>99.5%</td>
<td>0%</td>
</tr>
</tbody>
</table>

As evident in Fig. 2, adaptive REM shows much better response and stability. Its settling time is much less than REM and p-REM, and it also responds much more quickly to load variations and round trip delay changes. REM and p-REM, on the other hand, are quite sluggish in responding to changes in the load level, and their queues have more oscillation. We also compare their steady-state performances in Table 1. It can be seen that the mean queue length of adaptive REM is the closest to the target queue length (recall that the target is 50), and the standard deviation (SD) of its queue length is the smallest. It can also be seen that adaptive REM outperforms REM and p-REM in terms of throughput and packet loss ratio. This confirms that adaptive REM is more robust and stable, and can provide better QoS.

Conclusion: We propose a new refined REM scheme, adaptive REM, which is based on control theory fundamentals and the new price update function. We have demonstrated that adaptive REM performs better than REM and p-REM by both theoretical analyses and simulations.

Acknowledgments: The research of L. Tan and G. Peng has been supported by the National Natural Science Foundation of China under grant numbers 60573188, 60473085, and the research of S. Chan by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 110705).

© The Institution of Engineering and Technology 2007
23 September 2006
Electronics Letters online no: 20072966
doi: 10.1049/el:20072966
L. Tan and G. Peng (Department of Computer Science, Central China Normal University, Wuhan 430079, People’s Republic of China)
S. Chan (Department of Electronic Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong, People’s Republic of China)
E-mail: eeschan@cityu.edu.hk

References