

네트워크 부하감지를 기반으로 한 확장 RED

김 동 춘* · 김 은 범* · 안 기 중**

Extended RED based on Network Load Variations

Dong-Choon Kim*, Eun-Bum Kim* and Khi-jung Ahn**

ABSTRACT

This paper addresses the extended RED to be adapted in various network conditions for congestion control. RED has some problems in various network conditions due to the fixed operation of parameters, P_{max} and TH_{min} . That is, if the traffic load is relatively small the link utility can be too low and if the load is too large the management of queue can result in various difficulties. In this paper we propose the long-term queue average method in which those parameters of P_{max} and TH_{min} can be automatically adjusted based on the variations of network load. The experimental result shows that the proposed method can overcome the problems caused by RED.

Key Words : RED, Congestion Control, long-term queue average

1. 서 론

인터넷에서 체증을 해결하는 방법은 송신 호스트의 윈도우 크기(window size)와 재전송시간(RTO: retransmission time out)을 이용하여 체증을 조절하는 방법과 망 중간 노드인 라우터에서의 버퍼관리를 통해, 체증을 미리 예견하여 라우터의 체증정도에 따라 라우터에 들어오는 패킷을 계산된 확률에 의해 폐기하는 RED(random early detection)방법이 있다[1,2].

RED 방법은 중간 노드인 라우터에서 체증이

발생하기 전에 능동적인 버퍼관리를 통하여 global synchronization에 의해 순간적으로 링크의 이용률이 감소하는 것을 방지하고 라우터 내의 큐 길이를 가능한 작은 크기로 유지케하는 방법이며, 망 중간 노드에서의 큐잉지연이 전체 패킷 처리율(throughput)에 매우 큰 영향을 미치기 때문에 중간 노드의 큐 길이를 작게 유지하는 것도 체증 제어의 중요한 목적의 하나로 인식되고 있다[3].

RED는 가중 평균 큐 길이에 따라 망의 체증정도를 결정하여 라우터에 들어오는 패킷을 폐기(drop)하는데 적용되는 확률이 결정되며, 이 RED 방식을 사용할 경우 Drop Tail 방식과 비교하여 링크의 사용 효율을 높일 수 있다는 것은 이미 여러 문헌에서 보여주고 있다[3,4,5,6].

그러나 RED에서 사용하는 파라미터들은 망의 상

* 제주대학교 대학원

The Graduate of Cheju Nat'l Univ.

** 제주대학교 통신·컴퓨터공학부, 첨단기술연구소

Faculty of Telecommunication and Computer Engineering, Res. Inst. Adv. Tech., Cheju Nat' Univ.

황과 상관없이 동일한 값으로 설정되기 때문에 다양한 트래픽 상황에 적용하게 되면 여러 가지 문제점이 발생한다.

일반적으로 버스트한 데이터의 특성을 갖는 망에서는 최소 큐 한계값(TH_{min})을 크게 하면 링크의 사용효율을 높일 수 있으며, 지속적으로 망에 과부하가 가해지고 있을 때는 최대확률(P_{max})값을 높게 설정하여야 효율적인 체증관리가 가능하다고 알려져 있다. 현재의 망의 특성은 버스트한 데이터의 특성과 지속적인 과부하의 망 특성을 동시에 갖고 있어 동일한 파라미터의 값을 설정하면 상이한 두 환경에서 잘 적용되기란 불가능하다.

본 논문에서는 망의 전체적인 부하를 감지하는 방법을 제안하고 이 방법을 이용하여 망의 상태에 따라 각각의 파라미터의 값을 자동적으로 조절하여 다양한 망 환경에서도 잘 적용될 수 있는 확장 RED를 제안하고자 한다.

II. RED 파라미터에 대한 동작특성

RED에서 정의되어 있는 4개의 파라미터는 가중치(w_q), 최소 큐 한계값(TH_{min}), 최대 큐 한계값(TH_{max}), 최대 확률(P_{max})이 있으며 이 파라미터들의 특성은 아래와 같다.

2.1 큐 가중치(w_q)와 평균 큐길이(Q_{avg})

평균 큐길이(Q_{avg})를 구할 때 실제 큐 길이가 Q_{avg} 가 변하는데 미치는 영향을 나타내는 값이 큐가중치(w_q)이다. w_q 가 작으면 작을수록 현재의 큐의 실제 길이가 Q_{avg} 가 변하는데 미치는 영향이 작게 된다. 이런 경우에는 짧은 시간 동안 큐의 길이가 증가되더라도 Q_{avg} 가 증가하는데 미치는 영향이 작기 때문에 일시적인 트래픽의 증가로 인한 오류를 피할 수 있다.

w_q 는 0.001이상을 쓰도록 권하고 있고 일반적으로 0.002를 사용하고 있다.

2.2. 최대 확률(P_{max})

P_{max} 의 값을 조절할 경우 가장 영향을 받는 부분은 Q_{avg} 이다. 이 P_{max} 가 커질 경우에 패킷 폐기 확률이 커져 폐기하는 횟수가 많아지므로 실제 큐길이나 Q_{avg} 가 전체적으로 줄어들게 된다. 망의 부하가 증가하면 P_{max} 값을 높여 주어야 한다.

2.3. 최소 큐 한계값(TH_{min}), 최대 큐 한계값(TH_{max})

실제 망에 RED를 적용했을 때 TH_{min} 과 TH_{max} 값을 설정하는 것은 상당히 중요한 일이지만, 이것은 수학적으로나 시뮬레이션만으로 정할 수 있는 파라미터는 아니다. [4]에서 이 파라미터의 범위를 다음과 같이 정하고 있다.

트래픽이 버스트한 특성을 가질수록 TH_{min} 은 크게 주어 링크의 사용효율을 유지할 수 있도록 해준다. 그러나 정상적인 환경에서는 TH_{min} 은 가급적 작게 주는 것이 유리하다. TH_{max} 은 Drop Tail의 최대 버퍼크기와 비슷하게 동작하므로 이 파라미터의 값을 클 경우는 큐 길이가 전체적으로 높게 유지되기 때문에 큐잉 지연시간(queueing delay time)이 길어진다. TH_{min} 과 TH_{max} 의 차이는 왕복시간(roundtrip-time)동안에 계산되는 평균 큐길이의 추정치보다 커야 RED 라우터는 효율적으로 동작할 수 있다. 대략 TH_{max} 은 최소한 TH_{min} 의 두배 이상으로 할 것을 권고하고 있다.

2.4. RED 파라미터의 적용의 문제점

위에서 설명한 바와 같이 RED의 4가지 파라미터들은 그 값을 어떻게 정하느냐에 따라 RED의 성능에 증대한 영향을 미친다. 하지만 다양한 트래픽 상황에서도 범용적으로 적용할 수 있는 파라미터를 결정하는 것은 매우 어렵다.

이 파라미터의 최적 값은 망의 트래픽 상황에 따라 달라지기 때문에 결국은 트래픽 상황에 변화에 따라 그 값을 변화 할 수 있어야 할 것이다. RED의 파라미터를 고정시켜놓고 호스트의 수를 증가시키면

어느 순간에는 효율이 극히 나빠지는 상황이 발생할 수 있으며 그 이유는 트래픽의 버스트 기간이 증가되므로써 심한 체증 현상을 초래하게 되며 이러한 트래픽 상황에 대해서 고정적으로 정한 RED 파라미터가 원래의 의도대로 동작하지 못하기 때문이다.

또한 이러한 상황에 맞도록 파라미터의 값을 정하게 되면 트래픽부하가 적을 경우에 throughput에 영향을 주어 성능이 나빠진다.

위의 내용을 간추리면 RED에서는 4개의 파라미터 등을 고정된 값으로 사용하기 때문에 지속적으로 트래픽의 버스트한 상황일 때는 RED가 제대로 기능을 발휘하지 못하는 상황이 발생할 수 있으며 이를 보완하기 위해서는 파라미터의 값을 망의 상태에 따라 조절하여 적응적으로 반응할 수 있도록 하여야 한다.

III. 제안 알고리즘

3.1. RED의 문제점

RED의 고정된 파라미터들은 적정한 트래픽부하일 경우에 잘 적용되도록 정해지므로 위에서 지적한 버스트한 트래픽부하가 지속적으로 가해졌을 때도 문제가 발생하지만 낮은 경우에도 개선의 여지가 있다. 4개의 파라미터중 TH_{min} 의 값은 트래픽부하가 적을 경우 링크 이용률을 높게 유지하기 위해 사용되어지며 높게 할수록 작은 트래픽 상황일 경우에 좋은 링크의 이용률을 유지한다.

그러나 이러한 TH_{min} 을 적정한 트래픽부하일 때에도 동일하게 적용된다면 링크 이용률에는 향상이 없고 불필요한 평균 큐의 크기만 증가시키게 되므로 이 경우에는 TH_{min} 은 작게 유지되어야 한다.

3.2. 제안 알고리즘

위에서 기술한 RED의 문제점들은 망의 지속적인 트래픽 상태를 감지할 수 없다는 알고리즘의 특성 때문에 야기되는 문제이다. 본 논문에서는 지속적인 트래픽 상태를 감지하는 방법을 제시하고 이를 이용하여 최소 큐 한계값(TH_{min})과 최대 확률(P_{max})을

망 상태에 따라 적응적으로 반응하는 확장된 RED 알고리즘을 제안한다.

3.2.1 트래픽상태 감지방법제시

일반적으로 사용되는 $w_q=0.002$ 를 1/20이하로 감소시켜 식 (1)처럼 가중평균(Q_{avg})을 계산하여 Q_L 라 하고 망 트래픽 상태의 척도로 사용한다.

$$Q_L \leftarrow (1 - W_L) \times Q_L + W_L \times Q_{size} \quad (1)$$

단. $W_L < \frac{W_q}{20}$

이 값이 TH_{min}/β 일 부근에 있을 경우를 정상적인 상태, 클 경우를 과부하상태, 작을 경우는 낮은 부하상태라 판단한다. 여기서 β 는 정상상태의 기준을 결정하는 파라미터로 망의 특성에 따라 달라질 수 있다. 작은 큐잉지연을 요구되면 크게 하고, 큐잉 지연보다 좋은 링크 이용률을 원한다면 작은 값을 선택할 수 있다. 본 논문에서는 $\beta = 2$ 로 설정하여 모의실험을 하였다.

3.2.2 최소 큐 한계값(TH_{min}) 결정

최소 큐 한계값은 낮은 망 부하 상태일 때는 높게, 높은 망 부하상태일 때는 낮게 설정할 수 있도록 식 2와 같이 결정한다.

$$TH_{min} = \alpha \times \left(\frac{TH_{max}}{\beta} - Q_L \right) + \gamma \quad (2)$$

여기서 α 와 γ 값은 최소 큐 한계값을 결정하는 파라미터로 본 논문에서는

$\alpha = 5, \gamma = 5$ 로 설정하여 모의실험을 하였다.

3.2.3 최대확률(P_{max}) 결정

망의 트래픽상태에 따라 P_{max} 값을 자동적으로 아래의 식을 이용하여 결정한다. 망의 부하가 적을 때는 낮게 하고 많을 때는 높게 하여 다양한 망의 환경에 효율적으로 적용할 수 있도록 결정한다.

$$Par \leftarrow \frac{Q_L}{TH_{max} - Q_L \times (\beta - 1)} \quad (3)$$

$$P_{max} \leftarrow P_{mini} \times (\delta^{Par} - \delta + 1)$$

P_{mini} : P_{max} 의 초기 값으로 주어지는 값

IV. 모의실험 결과 및 분석

본 논문에서는 $\delta = 5$ 로 설정하여 모의실험을 하였다.

3.2.4 제안 확장 RED 알고리즘

Saved Variables

- Q_{avg} : average queue size
- Q_L : long term average queue size
- q_time : start of the queue idle time
- count : packets since last marked packet
- TH_{min} : minimum threshold for queue
- β : parameter to control P_{max}

Fixed parameter

- w_1 : queue weight
- w_2 : long term queue weight

Others

- P_s : current packet-marking probability
- Q_{cur} : current queue size
- time : current time
- $f(x)$: a linear function of the time

Initialization:

$$Q_{avg} \leftarrow 0$$

$$LONG_Q_{avg} \leftarrow \frac{TH_{min}}{\beta}$$

$$count \leftarrow -1$$

for each packet arrival

- if the queue is nonempty
 - $Q_{avg} \leftarrow (1 - W_1) \times Q_{cur} + W_1 \times Q_{avg}$
 - $Q_L \leftarrow (1 - W_2) \times Q_L + W_2 \times Q_{cur}$
- else
 - $m \leftarrow f(\text{time} - q_time)$
 - $Q_{avg} \leftarrow (1 - W_1)^m \times Q_{cur}$
 - $Q_L \leftarrow (1 - W_2)^m \times Q_L$

calculate new each average and TH_{min}

$$TH_{min} = \alpha \times \left(\frac{TH_{max}}{\beta} - Q_L + \tau \right)$$

if $Q_{cur} > TH_{min}$

if $TH_{min} < Q_{cur} < TH_{max}$

increment count

calculate P_s

$$P_s \leftarrow \frac{Q_L}{TH_{min} - Q_L(\beta - 1)}$$

$$P_{max} \leftarrow P_{max} \times (\beta^{count} - \beta + 1)$$

$$P_s \leftarrow P_{max} \times \left(\frac{Q_{cur} - TH_{min}}{TH_{max} - TH_{min}} \right)$$

$$P_s \leftarrow \frac{P_s}{1 - count \times P_s}$$

mark the arriving packet with probability P_s .

count $\leftarrow 0$

else if $Q_{cur} > TH_{max}$

mark the arriving packet

count $\leftarrow 0$

else $Q_{cur} \leq TH_{min}$

Queue Packet

else $Q_{cur} \leq TH_{max}$

Queue Packet

when queue become empty

$q_time \leftarrow time$

4.1. 모의실험 환경

본 논문에서 사용한 Network 환경은 Fig.1 과 같이 구성하였으며 다음과 같은 전제조건을 두었으며, 일반적인 사항은 고려하지 않았다.

- ① 모든 링크의 대역폭은 100Mbps
- ② 링크와 gateway 사이의 delay time은 1. 2. 3. 5. 4. 4 ... msec으로 반복적용
- ③ Sink 노드와 Router 사이의 delay time : 2msec

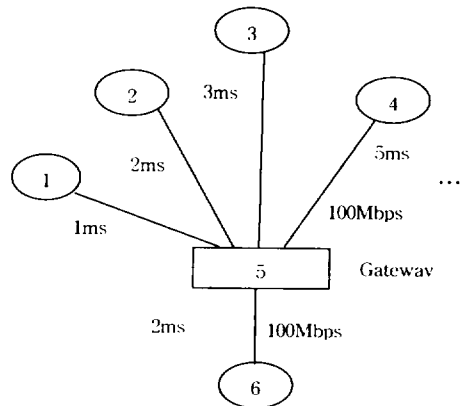


Fig. 1. simulation network.

모의실험은 링크 수를 늘여가면서 Gateway 형태를 RED와 제안 확장 RED의 방식을 비교하였으며, 모의 실험 시간은 10초, Gateway 큐 길이는 100패킷, 1패킷은 1000byte로 하였고 Ack packet은 40byte로 하였다.

체증인지 방법은 gateway가 패킷을 폐기하였다는 것을 링크 송신노드가 10msec 지나서 알 수 있도록 하였다. 각 호스트들은 자신이 전송한 패킷이 체증의 영향으로 폐기되었을 경우 TCP의 체증제어 방식인 slow start와 congestion avoidance를 행하게 된다. 그리고 송신 호스트에서 수신 호스트로 단방향 전송만을 가정하였다.

RED의 매개변수는 기존 RED알고리즘에 사용했던 파라미터는 RED인 경우 P_{max} 는 0.02, w_q 는 0.002, TH_{max} 는 60, TH_{min} 는 9, 패킷 크기는 1000.

윈도우 크기는 1024를 사용하였다.

제안 알고리즘의 w_o 는 0.002, w_L 는 0.00004, P_{max} 는 0.02, m 는 0.1, α 는 5, β 는 2, γ 는 5, δ 는 5로 설정하고, TH_{max} 값이 50, 60, 70에 대하여 모의 실험을 하였다.

제안 확장 RED 알고리즘과 기존 RED 알고리즘의 성능을 비교하기 위해 Microsoft Visual C++ 6.0을 사용하여 시뮬레이션을 실행하였다.

4.2. 모의실험결과 및 분석

링크 이용률은 전송 가능한 최대 전송 패킷 수와 실제 전송한 패킷 수와 비율이고, 평균 큐 길이는 패킷이 도착할 때의 큐 길이의 평균으로 나타내었다.

Fig. 2는 제안 RED 알고리즘이 부하가 적을 경우에는 기존 RED 알고리즘보다 링크 이용률면에서 우수하다는 것을 보여주고 있으며, Fig. 3은 부하가 많은 경우에는 링크이용률의 감소 없이 평균 큐의 길이를 감소시킬 수 있음을 보여주고 있다.

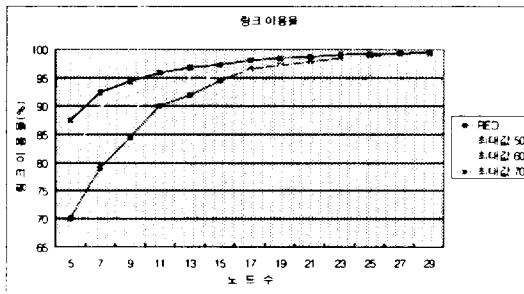


Fig. 2. Link Utility.

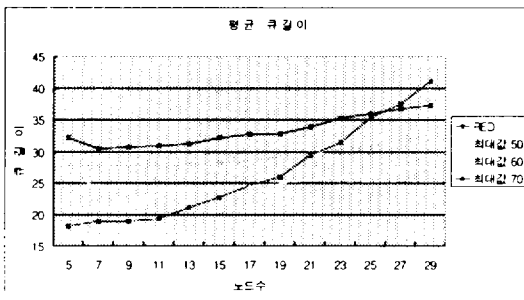


Fig. 3. Average Queue Length.

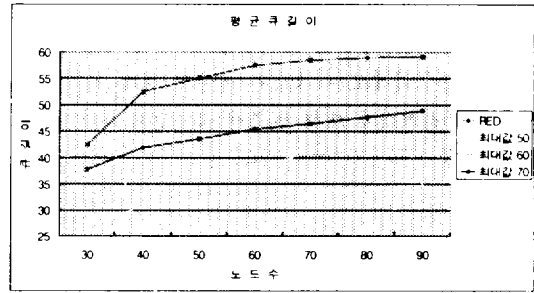


Fig. 4. Average Queue Length in Overload.

Fig. 4는 망 상태가 과부하로 되는 40개 노드부터는 RED는 더 이상 큐를 관리하지 못해 평균 큐의 길이가 최대값 근처에 있음을 보여주는 반면 제안 확장 RED는 적절하게 동작하고 있음을 보여준다. 특히 Fig. 3에서는 제안 알고리즘은 평균 큐의 길이가 TH_{min}/β 근처에 있음을 보여주고 있어 기존 RED와는 달리 과부하일 경우에도 평균 큐의 제어가 가능함을 보여주고 있다.

VI. 결론

본 논문에서는 RED 알고리즘의 문제점을 제시하고 문제점을 보완하고 확장할 수 있는 방안을 제시하였다.

RED 알고리즘의 문제점은 부하가 적을 때는 링크이용률이 낮으며, 부하가 아주 많을 때에는 큐 관리를 못하는 것들이다.

본 논문에서는 장기적인 가중평균을 이용하여 지속적인 망상태를 검색하고, 이러한 가중평균을 최소 큐 한계값과 최대 폐기확률 결정에 활용함으로써 부하의 변동이 심한 다양한 인터넷 환경에 잘 부합될 수 있는 확장 RED를 제안하였다.

제안 확장 RED의 설계결과는 기존 RED와 비교하여 볼 때, 부하가 적은 망 상태일 경우에는 링크 이용률을 증가시켰으며 부하가 많을 때에는 링크이용률의 저하없이 평균 큐의 길이를 감소시켰고, 특히 부하가 아주 많은 경우에, 큐의 관리가 기존 RED에서는 불가능하였으나 제안 확장 RED는 적절한 관리가 가능함을 보여주었다. 또한 평균 큐의 관리도 가

능하여 다양한 환경에서 라우터에서의 지연시간도 적절히 제어할 수 있음을 보여주었다.

향후 연구과제로는 새로이 제안된 확장 RED에서 사용된 파라미터에 대한 연구가 다양하게 이루어져야 할 것이다.

참고문헌

- 1) V. Jacobson. August 1988. Congestion Avoidance and Control. Computer Communication Review 18(4): 314~329.
- 2) W. Steven. April 1997. TCP Slow start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC 2309.
- 3) B. Braden et al.. April 1998. Recommendations on Queue Management and Congestion Avoidance in the Internet. RFC2309.
- 4) S. Floyd and V. Jacobson. 1993. Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Transaction on Networking. 1(4): 397~413.
- 5) M. Gaynor. Proactive Packet Dropping Methods for TCP Gateway. <http://www.eecs.harvard.edu/~gaynor/fina.ps>.
- 6) 유영석, 홍석원. 1997년 11월. 체증제어를 위한 Random Early Detection(RED) 알고리즘의 파라미터 분석. 통신학회 추계학술발표 논문집. pp. 41~44.
- 7) D. Lin and R. Moris. September 1997. Dynamics of Random Early Detection. Proc. of ACM SIGCOMM.
- 8) W. Feng et al.. Technique for Eliminating Packet Loss Congested TCP/IP Network. U. Michigan CSE-TR-349-97. <http://www.eecs.umich.edu/~wuchang/ared>.
- 9) Advanced QoS Services for the Intelligent Cisco white paper. http://www.cisco.com/warp/public/732/net_enable/qos_wp.htm#HDR7.
- 10) Zhang.L. August. 1989. A New Architecture for Packet Switching Network Protocols. MIT LCSTR-455. Laboratory for Computer Science. Massachusetts Institute of Technology.
- 11) W.Feng, D.Kandlur, D.Saha, K.Shin. November. 1997. Techniques for Eliminating Packet Loss in Congested TCP/IP Networks. UM-CSE-TR-349-97.
- 12) W.Richard Stevens. 1998. TCP/IP Illustrated, Volume 1.