

두 개의 병목지점 ECN/RED 게이트웨이의 분석 *

이계민¹⁾ 강영경²⁾ 전중우³⁾

요약

대다수의 TCP 사용자가 접근하여 병목현상을 보이는 두 개의 ECN/RED 게이트웨이에 대한 마코프 모형을 제시하고 분석하였다. 사용자의 수가 증가함에 따라, 각 게이트웨이의 버퍼를 점유하는 사용자당 패킷수는 일정한 상수들로 수렴한다는 것을 보이고, 이 상수값을 구하는 알고리즘을 제시하였다.

주요용어: TCP 혼잡제어, 실시간 버퍼관리기법(ECN/RED), 전파시간, 대기시간

1. 서론

지난 10여년간 인터넷은 폭발적인 인기와 거대한 상업적인 성공을 거두었다. 이로 인하여 인터넷망의 자원을 효율적으로 이용하기 위한 인터넷 트래픽의 변화양상에 대한 정확한 트래픽 모형화와 트래픽의 속도 조절방법에 대한 연구가 절실하게 되었다. 최근 통계에 따르면, 인터넷 트래픽의 95%가 TCP 혼잡제어 매카니즘을 이용하고 있으며, 종단 대 종단간 통신수에서 보면 80%에 이른다(Jacobson(1998), Casetti and Meo(2000)). 따라서 인터넷 트래픽을 정확하게 모형화하기 위해서는 TCP 혼잡제어에 대해 이해하여야 할 뿐만 아니라 TCP 혼잡제어와 네트워크 자원간에 일어나는 상호작용에 대해서도 이해하는 것이 중요하게 되었다.

Padhye et. al.(2000)는 한 명의 TCP 사용자가 매우 큰 파일을 전송하려고 하고 네트워크에서 발생하는 패킷의 손실확률이 고정된 상황에서 상호작용에 대해 분석하였다. 그러나 보다 현실적인 상황, 즉 다수의 사용자가 하나의 네트워크 자원을 이용하기 위해 경쟁하고 있는 상황에서는 다음 세 가지의 요인들로 인해 이러한 상호작용에 대한 분석은 매우 복잡하게 된다:

첫째, 인터넷 트래픽은 다수의 서로 다른 특징을 갖는 TCP 사용자가 발생하는 트래픽들의 집합체이다. 즉, 작은 파일을 전송하려는 사용자, 큰 파일을 전송하려는 사용자, 짧은 전파시간을 갖는 사용자, 긴 전파시간을 갖는 사용자 등이 발생한 트래픽들이 인터넷망에 들어온다. 이러한 다양한 특징을 갖는 TCP 사용자들의 매우 많은 상황을 직접적인 마

* 이 연구는 서울대학교 복잡계통계연구센터의 지원으로 이루어졌다.

1) (660-701) 진주시 경상대학교 통계정보학과, 부교수

Email:gyemin@gnsu.ac.kr

2) (447-791) 오산시 한신대학교 정보시스템공학과, 부교수

Email: ykang@hanshin.ac.kr

3) (151-742) 서울시 서울대학교 통계학과, 복잡계연구센터, 교수

Email: jwjeon@plaza.snu.ac.kr

코프모형을 통해 분석할 경우, 사용자들과 네트워크 상황을 설명하기 위해 필요한 변수들의 수가 폭발적으로 증가하여 필요한 계산량을 통제할 수 없게 된다.

둘째, 최근에 게이트웨이와 같은 네트워크 자원의 사용율을 높이기 위해 실시간 버퍼관리 기법(Active Queue Management : AQM)이 소개되었다. 전통적인 Tail-Drop 게이트웨이에서는 혼잡상태가 발생할 때 다수 사용자들의 패킷이 동시에 손실되고, 그로 인해 일시적으로 네트워크 자원의 이용률이 저하되는 현상이 반복적으로 나타난다는 사실이 관측과 실험을 통해 확인됨으로써 이러한 AQM이 제안되고 개발되어졌다. 소위 Explicit Congestion Notification(ECN)나 Random Early Detection(RED)와 같은 AQM은 혼잡상태가 일어나기 전부터, 게이트웨이의 버퍼에 있는 패킷수에 대한 지수 평활 평균값에 기초하여 버퍼에 들어오는 패킷을 랜덤하게 손실시키거나 마킹하는 방법을 통해 이러한 문제를 약화시킨다(Floyd (1994), Floyd and Jacobson(1993)). 이러한 기법은 개별 TCP 사용자들에게, 게이트웨이 혼잡이 일어나기 시작하는 초기에 패킷의 손실이나 마킹정보에 기초하여 송신 패킷수를 조절하게 함으로써 심각한 혼잡상태를 피하는 것을 가능하게 한다. 하지만 이러한 AQM의 도입은 TCP 혼잡제어과 게이트웨이의 상호작용을 이해하는 것을 더욱 어렵게 만든다.

세째, AQM이 운용 가능한 게이트웨이와 다양한 전파시간을 갖는 TCP의 상호작용을 모형화할 경우, AQM에서 개별 TCP에게 보내지는 피드백 정보가 전달되는 속도가 다르다는데서 문제는 더욱 복잡해진다.

이러한 세 가지의 요인들은 인터넷 트래픽의 변화과정의 중요한 측면을 반영할 수 있는 모형을 이끌어 내는데 상당한 어려움을 던진다. AQM 중의 하나인 ECN/RED가 운용 가능한 게이트웨이와 다수의 TCP의 상호작용에 대한 모형화에 대한 연구는 현재까지 많이 이루어져 있지 않다. 몇 가지의 초기 연구들은 Low et. al.(2002)에서 제시되어 있다. 그러나 이러한 연구들은 많아야 백 명 이하의 TCP 사용자들을 전제로 하고 있을 뿐이다.

Tinnakornsriruphap et. al.(2002)는 동일한 전파시간을 갖는 다수의 TCP 사용자들과 ECN/RED 게이트웨이에 대한 모형화에 새로운 접근법을 제시하였다. 그들의 접근법은 다수의 사용자들과 게이트웨이의 상호작용뿐만 아니라, 사용자들이 전송하려는 다양한 크기의 파일에 대해 모형을 제시하고, 적절한 극한정리를 체계적으로 적용하여, TCP 사용자수가 커짐에 따라 (게이트웨이의 버퍼에 있는 패킷수)/(사용자수)의 비율이 deterministic process가 된다는 것을 보였다. 그리고 Tinnakornsriruphap and Makowski(2003)는 TCP 사용자수가 커짐에 따라 (게이트웨이의 버퍼에 있는 패킷수)/(사용자수)의 비율이 시간이 충분히 흐른 평형상태에서 다시 일정한 상수로 수렴한다는 사실을 확인시키는 모의실험결과와 그 상수에 대한 근사값을 구하는 공식을 발표하였다. Tinnakornsriruphap and La(2003)는 TCP 사용자가 다양한 전파시간을 가질 수 있도록, Tinnakornsriruphap et. al.(2002)의 연구 결과를 확장하였다.

그러나 앞의 모형들은 패킷이 게이트웨이의 버퍼에서 서비스를 받기 위해 머무르는 시간인 대기시간(queueing delay time)을 반영하지는 못했다. 대기시간은 게이트웨이의 성능에 영향을 미치는 중요한 요소 중의 하나이다. Lee et. al.(2003)은 대기시간을 반영할 수 있도록 Tinnakornsriruphap et. al.(2002)의 연구 결과를 확장하였고, (게이트웨이의 버퍼에

있는 패킷수)/(사용자수)의 비율이 평형상태에서 수렴하는 상수값을 정확하게 구하는 방법론을 제시하였다. 본 논문에서, 우리는 Lee et. al.(2003)의 연구결과를 확장하여, 두 개의 ECN/RED 게이트웨이와 다수의 TCP 사용자가 있는 경우에 대해 분석하였다. 여기서도 TCP 사용자의 수가 커짐에 따라 각각의 게이트웨이에서 (게이트웨이의 버퍼에 있는 패킷수)/(사용자수)의 비율이 평형상태에서 일정한 상수들로 수렴한다는 사실을 증명하고, 그 상수들을 정확하게 구하는 방법을 유도하였다.

본 논문의 구성은 다음과 같다. 먼저 2절에서는 TCP 혼잡제어와 ECN/RED의 원리에 대해 살펴보고, 3절에서는 본 논문에서 사용하는 가정들과 기호들을 정리하였다. 4절에서는 분석 모형을 제시하고, 성질들을 살펴본 후, (게이트웨이의 버퍼에 있는 패킷수)/(사용자수)의 비율의 수렴값을 구하는 알고리즘에 대해 설명하였다. 끝으로 5절에서는 앞에서 제시한 모형의 분석과 공식의 정확성을 알아보기 위한 모의실험결과를 제시하였다

2. TCP 혼잡제어와 ECN/RED의 원리

우리는 아래의 그림 2.1과 같이 2개의 ECN/RED 게이트웨이와 각각 N_1 명과 N_2 명의 사용자가 있는 TCP 사용자 그룹을 가정하자. 그리고 시간은 적당한 간격의 slot들로 나누어져 있다고 가정한다. 1번 게이트웨이는 slot당 N_1C_1 개의 패킷, 2번 게이트웨이는 slot당 $(N_1 + N_2)C_2$ 개의 패킷을 처리할 수 있으며, 각 게이트웨이는 무한 버퍼를 가지고 있어 오버플로우로 인한 패킷손실은 없다고 가정한다.

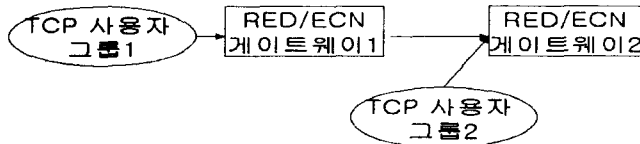


그림 2.1: 2개의 RED/ECN 게이트웨이와 TCP 사용자 그룹

게이트웨이의 버퍼에 많은 패킷이 차는 것을 방지하기 위해, ECN/RED 방식은 다음 그림 2.2와 같이 정의되는 마킹확률함수 (marking probability function) $f(\cdot)$ 를 갖는다.

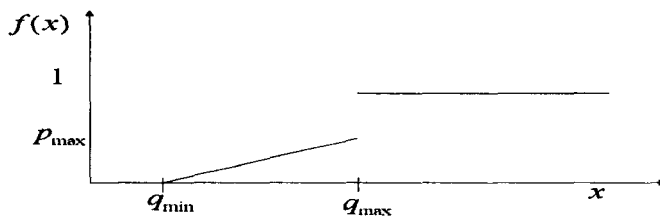


그림 2.2: 마킹확률함수

여기서 q_{min} , q_{max} , p_{max} 는 ECN/RED에서 사용하는 모수이다.

인터넷 사용자, 사용자의 TCP, ECN/RED 게이트웨이의 상호작용은 다음 네 가지의 단계로 이루어진다:

첫째, 개별 사용자는 Active 또는 Idle, 두 가지의 상태로만 행동한다. 다시 말해, 사용자가 새로운 idle 상태에 들어가면, 랜덤한 시간동안 아무런 행동을 하지 않고 기다린다. 반면에 새로운 Active 상태에 들어가면, 랜덤한 크기의 파일을 자신의 TCP에 넘긴다. 그러면 TCP는 이 파일을 패킷단위로 쪼개고, 자신의 혼잡제어가 정해주는 크기만큼의 패킷들을 게이트웨이에 전송한다.

둘째, 일반적으로 TCP 혼잡제어는 round 단위로 설명되는데, 하나의 round는 w 개의 패킷을 차례로 게이트웨이에 전송하는 것으로 시작한다. 이 때, w 를 congestion window size라고 한다. 목적지에서는 w 개의 패킷들을 하나씩 수신할 때마다, 이를 수신했다는 응답신호(Acknowledgement : ACK)를 보낸다. w 개의 패킷들에서 어느 하나에 대해서라도 수신했다는 ACK가 들어오면, 이전 round는 끝이 나고 새로운 round가 시작된다.

셋째, 게이트웨이의 ECN/RED는 버퍼에 들어오는 개별 패킷들에 대해 랜덤하게 마킹을 한다. 이 때, 어떤 패킷이 마킹될 확률은 버퍼에 있는 패킷수에 대한 지수 평활 평균값과 그림 2에서 정의된 마킹확률함수 $f(\cdot)$ 에 의해 결정된다. 그리고 나서 게이트웨이는 자신의 버퍼에 있는 패킷들을 해당 목적지로 전송한다. 각 목적지는 패킷을 받는 순간 그 패킷에 마킹이 되어 있는지를 확인하여, 그 패킷의 전송지로 marked ACK 또는 unmarked ACK을 되돌려 준다.

넷째, TCP 혼잡제어방식은 RED/ECN 게이트웨이와의 상호작용을 통하여, 2가지 종류의 작동모드: slow start (SS) 또는 congestion avoidance (CA)에서 작동하게 된다. 사용자가 새로운 Active 상태에 들어갈 때, 그의 TCP의 작동모드는 SS이고 congestion window size는 1로 세팅된다. SS모드에서는 매 round마다 congestion window size가 두 배씩 증가된다. 만일 하나 또는 그 이상의 marked ACK를 받게 되면, 작동모드는 CA로 전환되고 congestion window size는 현재 congestion window size의 절반으로 줄어든다. CA모드에서는 매 round마다, congestion window size가 1씩 증가한다. 만일 CA모드에서 하나 또는 그 이상의 marked ACK를 받게 되면, 작동모드는 그대로 CA모드를 유지하고, congestion window size는 현재 congestion window size의 절반으로 줄어든다.

3. 가정들과 기호들

다수의 TCP 사용자들을 갖는 ECN/RED 게이트웨이에 대해 분석 가능한 모형을 유도하기 위해, 다음과 같은 가정을 한다.

- (A1) 개별 TCP는 자신의 congestion window 안에 있는 모든 패킷을 크기에 관계없이 동시에 전송하고, 전송된 패킷들은 게이트웨이에 한꺼번에 도착한다.
- (A2) 서로 다른 TCP에서 전송된 패킷들이 게이트웨이에 동시에 도착할 경우, 하나의 TCP에서 보내진 패킷들은 한꺼번에 게이트웨이에 들어간다. 이 때 어떤 TCP의 패킷들을 먼저 처리할 것인가는 랜덤하게 결정된다.

(A3) 개별 TCP의 congestion window size는 W_{max} 를 넘지 않는다. 보통 W_{max} 는 receiver advertised window size에 의해 결정된다.

(A4) marked ACK 또는 unmarked ACK가 목적지에서 전송 TCP에게 보내지는데 걸리는 시간은 0으로 정한다.

각 그룹의 사용자, 사용자의 TCP 혼잡제어, 게이트웨이의 운동과정은 다음 그림 3.1과 그림 3.2와 같이 표현될 수 있다.

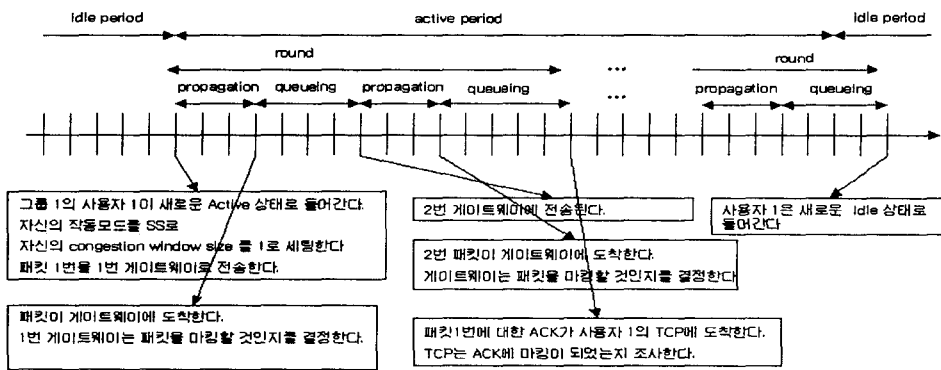


그림 3.1: 그룹 1의 TCP 사용자 1의 운동과정

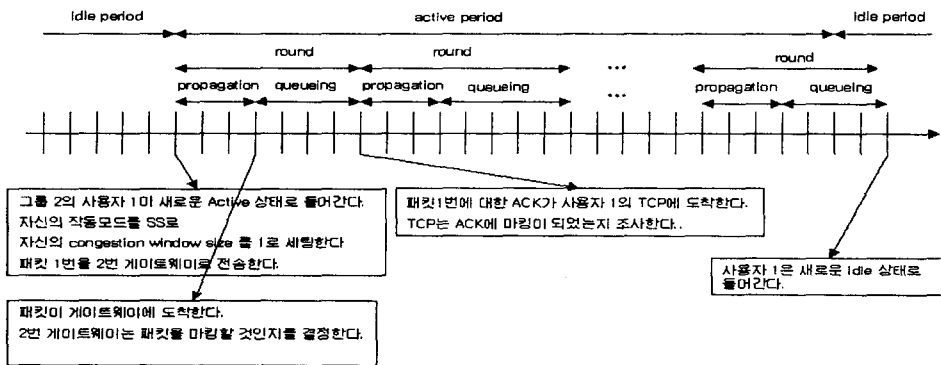


그림 3.2: 그룹 2의 TCP 사용자 1의 운동과정

그림 3.1과 그림 3.2를 보면, 사용자 그룹 1에서의 하나의 round의 길이는 두 개의 전파시간(propagation time)과 두 개의 대기시간(queueing delay time)으로 이루어지며, 사용자 그룹 2에서의 하나의 round의 길이는 하나의 전파시간과 하나의 대기시간으로 구성되어 있다. 그러나 Tinnakornsriruphap and La(2003)는 전파시간과 대기시간을 구별하지 않고 단지 round 단위로 모형화하였다. 그 결과, 게이트웨이의 버퍼에 있는 패킷수에 따라 대

기시간의 변화로 인한 게이트웨이의 성능의 변화를 고려하지 못하였다. 이러한 이유로 우리는 Lee et. al.(2003)에서는 아래의 그림 3.3과 그림 3.4에서와 같은 action point와 delay period라는 개념을 도입하였다.

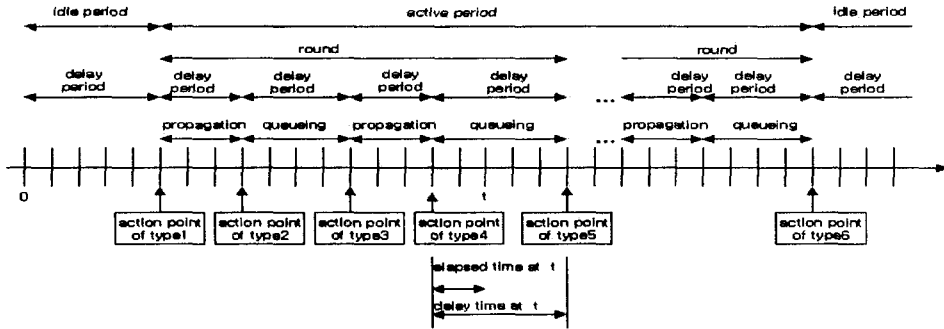


그림 3.3: 그룹 1의 TCP 사용자에게 대한 action point와 delay period

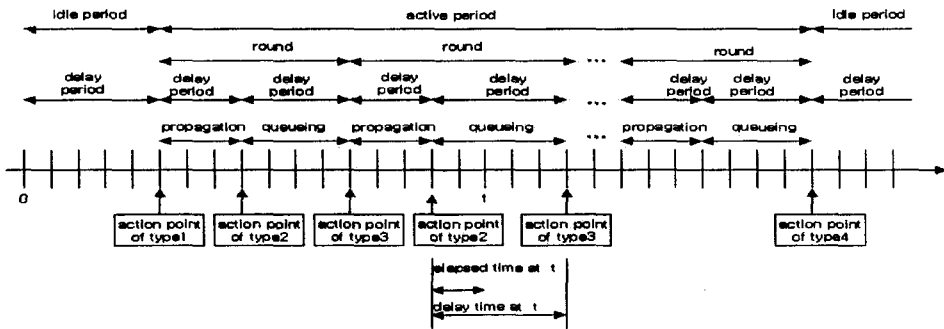


그림 3.4: 그룹 2의 TCP 사용자에게 대한 action point와 delay period

실용적인 모형을 유도하기 위해, TCP 사용자들의 상태, TCP 혼잡제어, 게이트웨이에 대해 추가적인 가정을 하자. 그룹 1에 있는 사용자들에 대해서는 첨자 $i(1 \leq i \leq N_1)$ 와 (1)를, 그룹 2에 있는 사용자들에 대해서 첨자 $j(1 \leq j \leq N_2)$ 와 (2)를 사용하기로 하며, 그룹 2 사용자에게 대한 기호는 생략한다.

(A5) 그룹 1에 있는 TCP 사용자 i 가 시점 t 에서 새로운 Active 상태로 들어가서, 자신의 TCP에게 넘겨주는 파일의 크기를 패킷단위로 환산할 때, 그 값은 $F_{t,i}^{(1)}$ 로 표시한다. 이 때, $F_{t,i}^{(1)}$ 은 확률분포 $\{p_{F^{(1)}}(1), p_{F^{(1)}}(2), \dots\}$ 을 따른다.

(A6) TCP 사용자 i 가 시점 t 에서 새로운 Idle 상태로 들어갈 때, Idle 상태에 머무르는 시간은 $I_{t,i}^{(1)}$ 로 표시한다. 이 때, $I_{t,i}^{(1)}$ 은 확률분포 $\{p_{I^{(1)}}(1), p_{I^{(1)}}(2), \dots\}$ 을 따른다.

(A7) TCP 사용자 i 가 시점 t 에서 새로운 Active 상태로 들어갈 때, 이후 사용자 i 의 TCP에 의해 보내지는 개별 패킷들이 1번 게이트웨이에 도달하는데 걸리는 전파시간은 $P_{i,i}^{(1)}$ 로 표시하자. 이 때, $P_{i,i}^{(1)}$ 는 확률분포 $\{p_{P^{(1)}}(1), p_{P^{(1)}}(2), \dots\}$ 를 따른다.

(A8) 1번 게이트웨이를 빠져 나온 패킷들이 2번 게이트웨이에 도달하는데 걸리는 시간은 일정하며, 그 값은 $P^{(1)}$ 로 표시한다.

이제, 시점 t 에서의 TCP 사용자, 사용자의 TCP 혼잡제어, 게이트웨이의 상태를 표시하기 위해 상태변수들을 정의하자.

- $b_i^{(1)}(t)$: 사용자 i 가 Idle 상태인지 Active 상태인지를 나타내는 변수
 사용자 i 가 active 상태이면 $b_i^{(1)}(t) = 1$ 이고, 아니면 $b_i^{(1)}(t) = 0$ 이다.
- $x_i^{(1)}(t)$: 사용자 i 가 보유하고 있는 미전송 또는 전송중인 패킷수
- $o_i^{(1)}(t)$: 사용자 i 의 TCP의 작동모드
 SS이면 $o_i^{(1)}(t) = 0$ 이고, CA이면 $o_i^{(1)}(t) = 1$ 이다.
- $w_i^{(1)}(t)$: 사용자 i 의 TCP의 congestion window size
- $a_i^{(1)}(t)$: 사용자 i 의 TCP에 의해 게이트웨이로 전송된 패킷수
- $m_i^{(1)}(t)$: 사용자 i 의 TCP에 의해, 시점 t 를 포함한 round 시작점에서 보내진 패킷들 중에서 하나라도 마킹됐는지 여부를 나타내는 변수
 마킹된 패킷이 있으면 $m_i^{(1)}(t) = 1$ 이고, 없으면 $m_i^{(1)}(t) = 0$ 이다.
- $d_i^{(1)}(t)$: 사용자 i 의 TCP의 delay time
- $e_i^{(1)}(t)$: 사용자 i 의 TCP의 elapsed time
- $action_i^{(1)}(t)$: 사용자 i TCP의 시점 t 에서 action의 종류
- $Q_1(t)$ (또는 $Q_2(t)$) : 1번 (또는 2번) 게이트웨이의 버퍼에 있는 패킷수
- $\hat{Q}_1(t)$ (또는 $\hat{Q}_2(t)$) : $Q_1(t)$ (또는 $Q_2(t)$)의 지수 평활 평균값
- $\mathbf{x}_i^{(1)} = (b_i^{(1)}(t), x_i^{(1)}(t), o_i^{(1)}(t), w_i^{(1)}(t), a_i^{(1)}(t), m_i^{(1)}(t), d_i^{(1)}(t), e_i^{(1)}(t))$

끝으로 우리는 $a \wedge b$ 를 a 와 b 의 최소값, $[x]$ 를 x 를 넘는 가장 작은 정수, $I(A)$ 는 사건 A 가 일어났는지를 나타내는 지시변수로 정의한다.

4. 모형 제안과 분석

이 절에서는, 먼저 앞에서 가정된 가정들과 기호를 이용해, TCP 사용자, 그의 TCP 혼잡제어, 게이트웨이의 운동과정에 대해 모형화한다. 여기서도 논문의 분량을 줄이고 이해력을 높이기 위해, TCP 사용자 그룹 1에 대해서만 설명한다. 그룹 2에 대해서는 그룹 1과 비슷하게 설명될 수 있으나, 통과하는 게이트웨이가 하나뿐이라는 사실만 고려하면 된다.

(TCP1) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 1에 있다면, 즉 사용자 i 가 새로운 Active 상태를 시작한다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= 1, \quad x_i^{(1)}(t+1) = F_{t+1,i}^{(1)}, \quad o_i^{(1)}(t+1) = 0, \quad w_i^{(1)}(t+1) = 1, \\ a_i^{(1)}(t+1) &= x_i^{(1)}(t+1) \wedge w_i^{(1)}(t+1), \quad m_i^{(1)}(t+1) = 0, \\ d_i^{(1)}(t+1) &= P_{t+1,i}^{(1)}, \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

(TCP2) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 2에 있다면, 즉 사용자 i 의 TCP가 보낸 패킷들이 1번 게이트웨이에 도착했다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= b_i^{(1)}(t), \quad x_i^{(1)}(t+1) = x_i^{(1)}(t), \quad o_i^{(1)}(t+1) = o_i^{(1)}(t), \\ w_i^{(1)}(t+1) &= w_i^{(1)}(t), \quad a_i^{(1)}(t+1) = a_i^{(1)}(t), \\ m_i^{(1)}(t+1) &= I\left(U_{t+1,i}^{(1,1)} > (1 - f_1(\hat{Q}_1(t)/N_1))^{a_i^{(1)}(t+1)}\right), \\ Q_1(t+1) &= Q_1(t+1) + a_i^{(1)}(t+1), \quad d_i^{(1)}(t+1) = \lceil \frac{Q_1(t+1)}{N_1 C_1} \rceil, \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

여기서, $U_{t+1,i}^{(1,1)}$ 은 1번 게이트웨이에서 사용자 i 의 TCP가 보낸 패킷의 마킹여부를 결정하기 위해 (0,1)사이에서 뽑은 난수이다. 그리고 $f_1(\cdot)$ 은 1번 게이트웨이에서 사용하는 마킹확률 함수로 $q_{min}^{(1)}, q_{max}^{(1)}, p_{min}^{(1)}$ 에 의해 정의된다.

(TCP3) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 3에 있다면, 즉 사용자 i 의 TCP가 보낸 패킷들이 1번 게이트웨이를 빠져나갔다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= b_i^{(1)}(t), \quad x_i^{(1)}(t+1) = x_i^{(1)}(t), \quad o_i^{(1)}(t+1) = o_i^{(1)}(t), \quad w_i^{(1)}(t+1) = w_i^{(1)}(t), \\ a_i^{(1)}(t+1) &= a_i^{(1)}(t), \quad m_i^{(1)}(t+1) = m_i^{(1)}(t), \quad d_i^{(1)}(t+1) = P^{(1)}, \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

(TCP4) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 4에 있다면, 즉 사용자 i 의 TCP가 보낸 패킷들이 2번 게이트웨이에 도착했다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= b_i^{(1)}(t), \quad x_i^{(1)}(t+1) = x_i^{(1)}(t), \quad o_i^{(1)}(t+1) = o_i^{(1)}(t), \\ w_i^{(1)}(t+1) &= w_i^{(1)}(t), \quad a_i^{(1)}(t+1) = a_i^{(1)}(t), \\ m_i^{(1)}(t+1) &= 1 - (1 - m_i^{(1)}(t)) \left(1 - I(U_{t+1,i}^{(1,2)} > (1 - f_2(\hat{Q}_2(t)/(N_1 + N_2))^{a_i^{(1)}(t+1)})\right) \\ Q_2(t+1) &= Q_2(t+1) + a_i^{(1)}(t+1), \\ d_i^{(1)}(t+1) &= \lceil \frac{Q_2(t+1)}{(N_1 + N_2)C_2} \rceil, \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

여기서 $U_i^{(1,2)}$ 은 2번 게이트웨이에서 그룹 1의 사용자 i 의 TCP가 보낸 패킷의 마킹여부를 결정하기 위해 (0,1)사이에서 뽑은 난수이다. 그리고 $f_2(\cdot)$ 은 2번 게이트웨이에서 사용하는 마킹확률함수로 $q_{min}^{(2)}, q_{max}^{(2)}, p_{max}^{(2)}$ 에 의해 정의된다.

(TCP5) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 5에 있다면, 즉 사용자 i 의 TCP가 보낸 패킷들에 대해 목적지에서 보낸 ACK가 도착했고, 사용자 i 의 TCP에게 아직 미전송된 패킷이 남아 있다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= b_i^{(1)}(t), \quad x_i^{(1)}(t+1) = x_i^{(1)}(t) - a_i^{(1)}(t), \\ o_i^{(1)}(t+1) &= (1 - o_i^{(1)}(t))(1 - m_i^{(1)}(t)), \\ w_i^{(1)}(t+1) &= \begin{cases} \min(2w_i^{(1)}(t), W_{max}) & \text{if } m_i^{(1)}(t) = 0, o_i^{(1)}(t+1) = 0 \\ \min(w_i^{(1)}(t) + 1, W_{max}) & \text{if } m_i^{(1)}(t) = 0, o_i^{(1)}(t+1) = 1 \\ \lceil \frac{w_i^{(1)}(t)}{2} \rceil & \text{if } m_i^{(1)}(t) = 1 \end{cases} \\ a_i^{(1)}(t+1) &= x_i^{(1)}(t+1) \wedge w_i^{(1)}(t+1), \quad m_i^{(1)}(t+1) = 0 \\ d_i^{(1)}(t+1) &= d_i^{(1)}(t), \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

(TCP6) 시점 $t + 1$ 에서 사용자 i 의 TCP가 action point of type 6에 있다면, 즉 사용자 i 의 TCP가 보낸 패킷들에 대해 목적지에서 보낸 ACK가 도착했고, 사용자 i 의 TCP에게 미전송된 패킷이 없다면, 다음 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= 0, \quad x_i^{(1)}(t+1) = 0, \quad o_i^{(1)}(t+1) = 0, \quad w_i^{(1)}(t+1) = 0, \\ a_i^{(1)}(t+1) &= 0, \quad d_i^{(1)}(t+1) = I_{t+1,i}^{(1)}, \quad e_i^{(1)}(t+1) = 0. \end{aligned}$$

(TCP7) 시점 $t + 1$ 에서 사용자 i 의 TCP가 어떤 종류의 action point에도 있지 않다면, 다음과 같은 변동이 일어난다.

$$\begin{aligned} b_i^{(1)}(t+1) &= b_i^{(1)}(t), \quad x_i^{(1)}(t+1) = x_i^{(1)}(t), \quad o_i^{(1)}(t+1) = o_i^{(1)}(t), \\ w_i^{(1)}(t+1) &= w_i^{(1)}(t), \quad a_i^{(1)}(t+1) = a_i^{(1)}(t), \quad m_i^{(1)}(t+1) = m_i^{(1)}(t), \\ d_i^{(1)}(t+1) &= d_i^{(1)}(t), \quad e_i^{(1)}(t+1) = e_i^{(1)}(t) + 1. \end{aligned}$$

위에서 설명한 (TCP1)~(TCP8)까지의 모형을 이용하여, 아래에서 설명하는 (M1) (M3)을 매 시점마다 차례대로 수행한다. 단 시점 0에서는 (M0)에서 설명한 작업을 수행한다.

(M0) 시점 0에서는 모든 TCP 사용자들이 새로운 idle period를 시작하고, 각 게이트웨이의 버퍼는 비어 있는 것으로 가정한다. 이러한 가정은 모의실험의 기술적 편리성을 위해 만들어진 것이다.

$$\begin{aligned} b_i^{(1)}(0) &= 0, \quad x_i^{(1)}(0) = 0, \quad o_i^{(1)}(0) = 0, \quad w_i^{(1)}(0) = 0, \\ a_i^{(1)}(0) &= 0, \quad m_i^{(1)}(0) = 0, \quad d_i^{(1)}(0) = I_{0,i}^{(1)}, \quad e_i^{(1)}(0) = 0 \\ Q_1(0) &= 0, \quad \hat{Q}_1(0) = 0, \quad Q_2(0) = 0, \quad \hat{Q}_2(0) = 0, \end{aligned}$$

(M1) 매 시점을 시작할 때, 각 게이트웨이의 버퍼에 있는 패킷수에 대한 정보를 갱신한다.

$$Q_1(t+1) = Q_1(t), Q_2(t+1) = Q_2(t)$$

(M2) 그룹 1과 그룹 2에 포함되는 모든 TCP 사용자들에 대해 처리순서를 랜덤하게 결정하며, 그 순서에 따라 개별 TCP별로 (TCP1)~(TCP8) 중에서 해당되는 것을 선택하여 수행한다. 단, 그룹 2에 속하는 사용자에 대해선 앞에서 정의한 (TCP1)~(TCP8)를 수정하여 적용해야 한다.

(M3) 게이트웨이의 버퍼에 있는 패킷수에 대한 평활지수 평균값은 다음과 같이 변한다.

$$\begin{aligned} Q_1(t+1) &= \max(Q_1(t+1) - N_1 C_1, 0), \hat{Q}_1(t+1) = (1 - \alpha_1)\hat{Q}_1(t) + \alpha_1 Q_1(t+1) \\ Q_2(t+1) &= \max(Q_2(t+1) - N_2 C_2, 0), \hat{Q}_2(t+1) = (1 - \alpha_2)\hat{Q}_2(t) + \alpha_2 Q_2(t+1) \end{aligned}$$

여기서, α_1 은 1번 게이트웨이의 ECN/RED에서 사용하는 모수이고, α_2 는 2번 게이트웨이의 ECN/RED에서 사용하는 모수이다.

위의 (M0)~(M3)에 의해 정의되는 모형하에서 $X(t) = (\mathbf{x}_1^{(1)}(t), \dots, \mathbf{x}_{N_1}^{(1)}(t), \mathbf{x}_1^{(2)}(t), \dots, \mathbf{x}_{N_2}^{(2)}(t), Q_1(t), \hat{Q}_1(t), Q_2(t), \hat{Q}_2(t))$ 는 하나의 다차원 이산시간형 마코프체인이 된다는 것을 쉽게 알 수 있다. 그리고 개별 TCP의 혼잡제어와 게이트웨이의 ECN/RED 기능 때문에, 각 게이트웨이의 버퍼에 있는 패킷수는 무한히 증가하지 못한다. 따라서 어떤 조건에서도 확률과정 $X(t)$ 는 극한분포(equilibrium distribution)를 가지게 된다. 그 결과, $\mathbf{x}_i^{(1)}(t)$, ($1 \leq i \leq N_1$)는 동일한 극한분포를 가지며, $\mathbf{x}_j^{(2)}(t)$, ($1 \leq j \leq N_2$) 역시 동일한 극한분포를 가진다.

확률과정 $\mathbf{x}_i^{(1)}(t)$, ($1 \leq i \leq N_1$)와 $\mathbf{x}_j^{(2)}(t)$, ($1 \leq j \leq N_2$)는 $(Q_1(t), \hat{Q}_1(t), Q_2(t), \hat{Q}_2(t))$ 를 통해서도 연관되어 있다. 개별 TCP가 특정 시점에 최대 W_{max} 개의 패킷을 게이트웨이에 보낼 수 있기 때문에, TCP 사용자의 수가 많아지면 개별 TCP 사용자가 $(Q_1(t), \hat{Q}_1(t), Q_2(t), \hat{Q}_2(t))$ 에 미치는 효과는 무시할 수 있게 된다. 따라서 사용자의 수가 많아지면, $\mathbf{x}_i^{(1)}(t)$, ($1 \leq i \leq N_1$)와 $\mathbf{x}_j^{(2)}(t)$, ($1 \leq j \leq N_2$)들은 점차 독립이 된다. 그리고 강대수법칙(strong law of large numbers)을 적용하면, (N_1, N_2, t) 가 커짐에 따라, 다음과 같은 수렴식을 얻을 수 있다:

$$\begin{aligned} \frac{1}{N_1} \sum_{i=1}^{N_1} a_i^{(1)}(t) I(\text{action}_i^{(1)}(t) = 2) &\rightarrow a_1^* \\ \frac{1}{N_2} \sum_{j=1}^{N_2} a_j^{(2)}(t) I(\text{action}_j^{(2)}(t) = 4) &\rightarrow a_2^* \end{aligned}$$

여기서, $I(\text{action}_i^{(1)}(t) = 2)$ 는 그룹 1의 사용자 i 가 보낸 패킷이 1번 게이트웨이에 들어가고 있다는 의미이고, $I(\text{action}_j^{(2)}(t) = 4)$ 는 그룹 2의 사용자 j 가 보낸 패킷이 2번 게이트웨이에 들어가고 있다는 것을 의미이다. a_1^* 는 그룹 1에 속하는 개별 TCP 사용자에 의해 1번 게이트웨이에 보내지는 평균 패킷수를 의미하며, a_2^* 는 그룹 2에 속하는 개별 TCP 사용자

에 2번 게이트웨이에 보내지는 평균 패킷수를 의미한다. 또한 각 게이트웨이의 버퍼에 있는 패킷수의 변화과정을

$$Q_1(t+1) = \max \left(Q_1(t) + \sum_{i=1}^{N_1} a_i^{(1)}(t+1) I(\text{action}_i^{(1)}(t+1) = 2) - N_1 C_1, 0 \right)$$

$$Q_2(t+1) = \max \left(Q_2(t) + \sum_{i=1}^{N_1} a_i^{(1)}(t+1) I(\text{action}_i^{(1)}(t+1) = 4) \right. \\ \left. + \sum_{j=1}^{N_2} a_j^{(2)}(t+1) I(\text{action}_j^{(2)}(t+1) = 4) - N_2 C_2, 0 \right)$$

로 표현할 수 있으므로, (N_1, N_2, t) 가 커짐에 따라, 우리는 다음과 같은 식을 얻을 수 있다:

$$\frac{Q_1(t)}{N_1} \rightarrow q_1^*, \quad \frac{\hat{Q}_1(t)}{N_1} \rightarrow q_1^*, \quad \frac{Q_2(t)}{N_1 + N_2} \rightarrow q_2^*, \quad \frac{\hat{Q}_2(t)}{N_1 + N_2} \rightarrow q_2^*$$

단, $N_1/(N_1 + N_2) = r$ 로 일정하다. 여기서, q_1^* 는 시간이 충분히 흐른 평형상태에서 (1번 게이트웨이의 버퍼에 있는 패킷수)/(그룹 1의 사용자수)의 비율, q_2^* 는 (2번 게이트웨이의 버퍼에 있는 패킷수)/(그룹 1과 그룹 2의 사용자수)의 비율로 해석할 수 있다.

위의 논의를 정리하면, (N_1, N_2, t) 가 충분히 큰 경우, 강대수법칙에 의해 $Q_1(t)/N_1$ 은 입력속도가 a_1^* , 처리속도가 C_1 인 deterministic 대기행렬체계와 확률적으로 동일하다는 사실을 알 수 있다. 비슷하게 2번 게이트웨이에 적용하면, $Q_2(t)/(N_1 + N_2)$ 는 입력속도가 $a_1^* N_1/(N_1 + N_2) + a_2^* N_2/(N_1 + N_2)$, 처리속도가 C_2 인 deterministic 대기행렬체계와 확률적으로 동일하다는 사실을 알 수 있다. 따라서 다음과 같은 결론을 내릴 수 있다:

첫째, 그룹 1의 사용자가 하나의 round 동안에 보낸 패킷들이 1번 게이트웨이를 통과하는데 걸리는 대기시간은 congestion window size에 관계없이 항상 $q_1^*/C_1 + 1$ 이 된다.

둘째, 그룹 1의 사용자 또는 그룹 2의 사용자가 하나의 round 동안에 보낸 패킷들이 1번 게이트웨이를 통과하는데 걸리는 대기시간은 congestion window size에 관계없이 항상 $q_2^*/C_2 + 1$ 이 된다.

셋째, 1번 게이트웨이에 들어가는 개별 패킷이 마킹될 확률은 $f_1(q_1^*)$ 이고, 2번 게이트웨이에 들어가는 개별 패킷이 마킹될 확률은 $f_2(q_2^*)$ 이다.

그러므로 (N_1, N_2, t) 가 충분히 큰 경우, 개별 TCP 사용자들과 그의 TCP들은 마치 독립적으로 동작하는 것처럼 취급해도 되며, $q_1^*/C_1 + 1$, $q_2^*/C_2 + 1$, $f_1(q_1^*)$, $f_2(q_2^*)$ 를 알고 있다면 네트워크의 상황에 전혀 영향을 받지 않는 것으로 해석할 수 있다. 따라서 q_1^* , C_1 , $q_{min}^{(1)}$, $q_{max}^{(1)}$, $p_{max}^{(1)}$, q_2^* , C_2 , $q_{min}^{(2)}$, $q_{max}^{(2)}$, $p_{max}^{(2)}$ 를 알고 있다면, $(b_1^{(1)}(t), x_1^{(1)}(t), o_1^{(1)}(t), w_1^{(1)}(t))$ 과 $(b_1^{(2)}(t), x_1^{(2)}(t), o_1^{(2)}(t), w_1^{(2)}(t))$ 는 각각 하나의 Semi-Markov process이 된다. 그리고 Lee et al.(2003)에서 사용한 논리를 따라가면, 우리는

$$a_1^* = \lim_{t \rightarrow \infty} E \left[\min(w_1^{(1)}(t), x_1^{(1)}(t)) \right], \quad a_2^* = \lim_{t \rightarrow \infty} E \left[\min(w_1^{(2)}(t), x_1^{(2)}(t)) \right],$$

라는 사실을 증명할 수 있으며, q_1^* , C_1 , $q_{min}^{(1)}$, $q_{max}^{(1)}$, $p_{max}^{(1)}$, q_2^* , C_2 , $q_{min}^{(2)}$, $q_{max}^{(2)}$, $p_{max}^{(2)}$ 이 주어졌을 때, (a_1^*, a_2^*) 를 계산하는 방법을 얻을 수 있다.

이제, q_1^* 과 q_2^* 를 구하는 방법에 대해 생각해 보자. 그리고 TCP의 혼잡제어 기능과 ECN/RED 기능으로 인해,

$$\bar{a}_1^* \equiv a_1^* \leq C_1, \bar{a}_1^* \equiv \frac{a_1^* N_1}{N_1 + N_2} + \frac{a_2^* N_2}{N_1 + N_2} \leq C_2$$

이라는 사실을 알 수 있다. 그리고 $\bar{a}_1 < C_1$ (또는 $\bar{a}_2 < C_2$)이면 $q_1^* = 0$ (또는 $q_1^* = 0$)이고, $\bar{a}_1 = C_1$ (또는 $\bar{a}_2 = C_2$)이면 $q_1^* > 0$ (또는 $q_2^* > 0$)이 된다는 사실을 알 수 있다. 이러한 사실에 기초해서 다음과 같은 반복적인 알고리즘을 세울 수 있다.

표 4.1: q_1^* 과 q_2^* 을 구하는 알고리즘

```

 $l_1 = q_{max}^{(1)}, u_1 = -q_{max}^{(1)}, a_1^* = 0$ 
do while ( $l_1 > 0 \& |a_1^* - C_1| > 0.00001$ )
   $q_1^* = \frac{1}{2}(l_1 + u_1)$ 
   $l_2 = q_{max}^{(2)}, u_2 = -q_{max}^{(2)}, a_2^* = 0$ 
  do while ( $l_1 > 0 \& |a_1^* - C_1| > 0.00001$ )
     $q_1^* = \frac{1}{2}(l_1 + u_1)$ 
     $a_1^* = \lim_{t \rightarrow \infty} E \left[ \min(w_1^{(1)}(t), x_1^{(1)}(t)) \right]$ 을 계산
     $a_2^* = \lim_{t \rightarrow \infty} E \left[ \min(w_1^{(2)}(t), x_1^{(2)}(t)) \right]$ 을 계산
     $\bar{a}_1$ 과  $\bar{a}_2$ 를 계산
     $\bar{a}_2 > C_2$ 이면  $u_2 = q_2^*$ 이고,  $\bar{a}_2 \leq C_2$ 이면  $l_2 = q_2^*$ 으로 치환
  end
   $\bar{a}_1 > C_1$ 이면  $u_1 = q_1^*$ 이고,  $\bar{a}_1 \leq C_1$ 이면  $l_1 = q_1^*$ 으로 치환
end

```

5. 모의실험과 결론

이 절에서는 하나의 모의실험을 통해, 4절에서 제시한 모형에 대한 분석방법론의 정확성을 살펴본다. 간단한 모의실험을 위해, $F_{t,i}^{(1)}, F_{t,i}^{(2)}, I_{t,i}^{(1)}, I_{t,i}^{(2)}$ 은 각각 모수가 $p_{file}^{(1)}, p_{file}^{(2)}, p_{idle}^{(1)}, p_{idle}^{(2)}$ 인 기하분포를 따르고 가정하자. 이러한 가정하에서 표 4.1에 있는 알고리즘 실행을 위한 핵심사항에 해당하는 (a_1^*, a_2^*) 를 구하는 방법은 Lee et. al.(2003)에 나와 있다.

예제) 각 TCP 사용자 그룹과 게이트웨이에 대하여 다음과 같은 모수를 사용한다.

사용자그룹 1 : $p_{file}^{(1)} = 1/1000, p_{idle}^{(1)} = 1/20, p_p^{(1)}(k) = 1/6, k = 1, 2, \dots, 6, P^{(1)} = 1$

사용자그룹 2 : $p_{file}^{(2)} = 1/1000, p_{idle}^{(2)} = 1/20, p_p^{(2)}(k) = 1/6, k = 1, 2, \dots, 6$

TCP : $W_{max} = 64$
 게이트웨이 : $C_1 = 1, C_2 = 1$
 ECN/RED : $q_{min}^{(1)} = 2, q_{max}^{(1)} = 20, p_{max}^{(1)} = 0.20, \alpha_1 = 0.01$
 $q_{min}^{(2)} = 2, q_{max}^{(2)} = 20, p_{max}^{(2)} = 0.20, \alpha_2 = 0.01$

아래의 그림 5.1~그림 5.3은 TCP 사용자의 수를 $N_1 = N_2 = 100, 1000, 10000$ 으로 바꾸어 가며, 4절의 (M0)~(M3)에서 제시한 방법에 따라 모의실험을 수행하여 (1번 게이트웨이의 버퍼에 있는 패킷수)/(그룹 1의 사용자수)의 비율과 (2번 게이트웨이의 버퍼에 있는 패킷수)/(그룹 1과 그룹 2의 사용자수)의 비율을 구한 결과이다. 이 그림들을 보면, 사용자의 수가 증감함에 따라 변동폭이 감소하고, 시간이 지남에 일정한 값으로 수렴한다는 것을 확인할 수 있다.

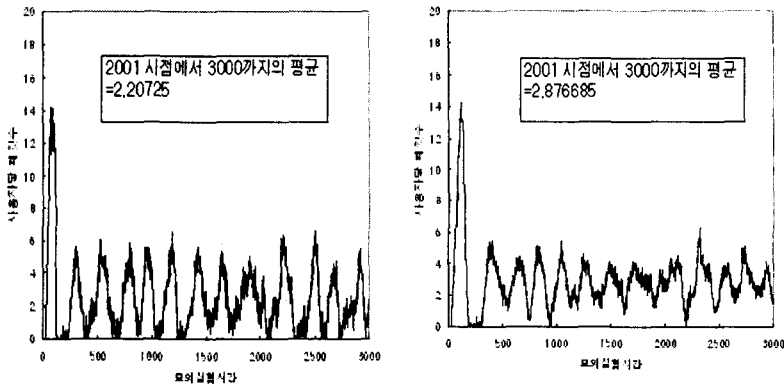


그림 5.1: $N_1 = N_2 = 100$ 일 때, 1번 게이트웨이와 2번 게이트웨이

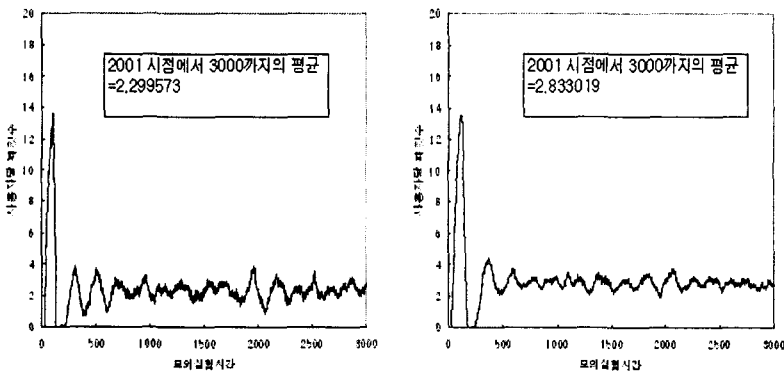


그림 5.2: $N_1 = N_2 = 1000$ 일 때, 1번 게이트웨이와 2번 게이트웨이

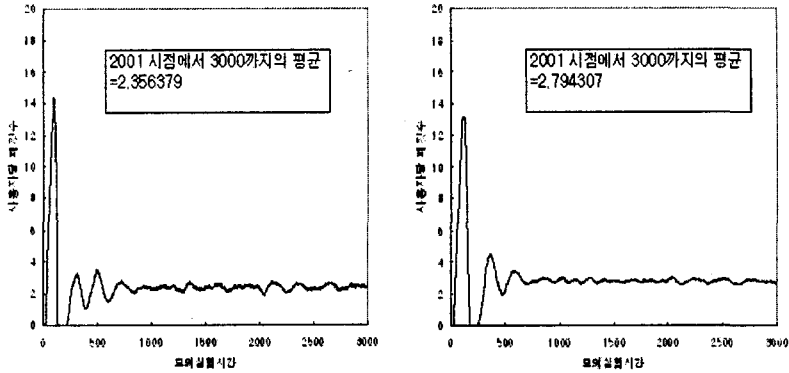


그림 5.3: $N_1 = N_2 = 10000$ 일 때, 1번 게이트웨이와 2번 게이트웨이

그리고 아래의 그림 5.4는 표 5.1에서 제시한 알고리즘에서 바깥쪽에 있는 do문의 반복수에 따라 q_1^* 와 q_2^* 의 값의 변화과정을 도시한 결과이다. 앞의 그림 5.1~그림 5.3까지의 결과와 비교하면, 우리가 본 논문에서 제시한 분석 방법이 정확하다는 것을 확인할 수 있다.

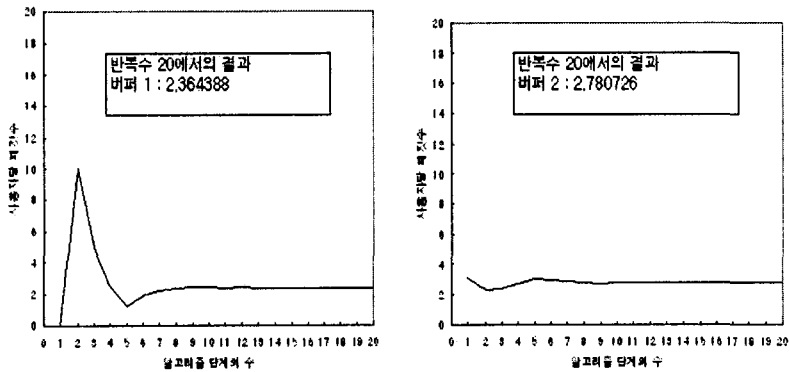


그림 5.4: 알고리즘에 의해 구한 1번 게이트웨이와 2번 게이트웨이의 극한값

위 모의실험 결과를 통해 TCP 사용자가 충분히 많으면, 제시된 알고리즘이 q_1^* 와 q_2^* 를 정확하게 구해 준다는 것을 확인할 수 있다. 따라서 제시된 알고리즘을 통해, 1번 게이트웨이와 2번 게이트웨이에서 하나의 round 안에 있는 패킷들이 처리되는 시간 $q_1^*/C_1 + 1$ 과 $q_2^*/C_2 + 1$, 1번 게이트웨이와 2번 게이트웨이에서 하나의 패킷이 marking 될 확률 $f_1(q_1^*)$ 와 $f_2(q_2^*)$ 을 정확하게 구할 수 있게 된다. 결론적으로 이상의 결과를 적절히 활용하면, 주어진 크기의 파일을 주어진 처리속도의 네트워크를 통해 전송하는데 걸리는 시간의 확률분포를 정확하게 예측할 수 있을 것이다.

참고문헌

- Casetti, C. and Meo, M.(2000). A new approach to model the stationary behavior of TCP connections, *In Proceedings of IEEE INFOCOM*, 367 -375.
- Floyd, S. and Jacobson, V.(1993). Random early detection gateways for congestion avoidance, *IEEE/ACM Transactions on Networking*, 1, 397-413.
- Floyd, S.(1994). TCP and explicit congestion notification, *ACM Computer Communication Review*, 24, 10-23.
- Lee, G. M., Kang, Y. G., and Jeon, J. W.(2003). Analysis of two bottleneck ECN/RED gateways in many user TCP networks, *Journ. of the Kor. Data Analysis society*, 5, 719-735.
- Jacobson, V.(1998). Congestion avoidance and control, *In Proceedings of SIGCOMM'88 Symposium*, 314-332.
- Low, S. H., Paganini, F., Wang, J., Adlakha, S., and Doyle, J. C.(2002). Dynamics of TCP/RED and a scalable control, *In Proceedings of IEEE INFOCOM*, 23-27.
- Padhye, J.,Froiu, V., Towsley, D., and Kurose, J.(2000). Modeling TCP throughput: a simple model and its empirical validation, *IEEE/ACM Transactions on Networking*, 8, 133-145.
- Tinnakornsrisuphap, P., La R. J., and Makowski, A. M.(2002). Characterization of general TCP traffic under a large number of flows regime, Technical report, Institute for Systems Research, University of Maryland.
- Tinnakornsrisuphap, P. and Makowski, A. M.(2003). Limit behavior of ECN/RED gateways under a large number of TCP flows, *In Proceedings of IEEE INFOCOM*, 873 -883.
- Tinnakornsrisuphap, P. and La, R. J.(2003). Limit model of ECN/RED under a large number of heterogeneous TCP flows, Technical report, Institute for Systems Research, University of Maryland.

[2003년 12월 접수, 2004년 2월 채택]

Analysis of Two Bottleneck ECN/RED Gateways in Many User TCP Networks *

Gyemin Lee ¹⁾ Yung-Gyung Kang ²⁾ Jongwoo Jeon ³⁾

ABSTRACT

We propose a Markov model of two bottleneck ECN/RED gateways experiencing a large number of TCP users. We show that, as the number of users becomes large, the queue sizes per user of the both gateways converge at steady-state to individual fixed points. Also, we derive a computational algorithm to exactly obtain the fixed points and present simulation example showing the limiting behavior of the gateway traffic.

Keywords: TCP congestion control; ECN/RED scheme; Propagation time; Queueing delay time

* This research was supported by Statistical Research Center for Complex Systems at Seoul National University.

1) Associate Professor, Dept of Statistics, Gyeongsang National University, Jinju, 660-701, Korea
Email: gyemin@gsnu.ac.kr

2) Associate Professor, Dept of Information System Engineering, Hanshin Universtiy, Osan, 447-791, Korea
Email: ykang@hanshin.ac.kr

3) Professor, Dept of Statistics, Statistical Research Center for Complex Systems, Seoul National University, Seoul, 151-742, Korea
Email: jwjeon@plaza.snu.ac.kr