

# TCP 트래픽에 대한 QoS를 향상시키기 위한 폭주제어

양진영(梁眞榮), \*이팔진(李八鎭), 김종화(金種和)  
목포대학교 컴퓨터공학과, \*초당대학교 컴퓨터과학과  
전화 : (061) 450-1227 / 팩스 : (061) 453-3944

## Congestion Control to Improve QoS with TCP Traffic

Jin Young Yang, \*Pal Jin Lee, Jong hwa Kim  
Dept. of Computer Engineering Mokpo National University,  
\*Dept. of Computer Science Chodang University  
E-mail : jyayang@chodang.ac.kr

### Abstract

End-to-end congestion control mechanism have been critical to the robustness and stability of the Internet. Most of today's Internet traffic is TCP, and we expect this to remain so in the future.

TCP/IP is the intermediate transport layer candidate for today's applications. TCP uses an adaptive window-based flow control. The congestion avoidance and control algorithms deployed by TCP aims at using the available network bandwidth. This paper compares different congestion control policies, and proposes the new design mechanism for future public networks

### I. 서론

최근 인터넷은 오디오 및 비디오와 같은 응용 서비스의 증가로 인하여 breadth와 depth측면에서 폭발적인 성장을 가져왔다. 그러한 응용 서비스들의 서로 다른 특징에 따라 서로 다른 QoS를 요구하고 있다. 이러한 급속한 성장에 따라 인터넷은 글로벌 통신망 하부 구조의 필수적인 부분이 되었다. 컴퓨터 트래픽의 예측할 수 없고 버스티한 성질은 질적인 보장을 할 수 없을 뿐만 아니라 폭주의 문제를 생성한다.

종단간(End-to-End) 폭주제어 기법은 인터넷에서 강인함(robustness)과 안정성(stability)에 대한 필수적인 부분이다. 인터넷의 대부분의 트래픽은 TCP-friendly 트래픽으로 이루어져 있으나 인터넷의 확장 및 사용자 수의 증가, 그리고 새로운 응용서비스 등장에 따라 misbehaving 트래픽이 급속도로 증가하여 종단간 폭주제어에 많은 문제를 낳게되었다.

게다가 오늘날의 인터넷은 종단간 지연(end-to-end delay)에 대한 상한값(upper bound) 또는 이용가능한 전송대역에 대한 하한값(lower bound)를 보장할 수 없어 새로운 응용에 전달된 QoS는 제어할 수 없을 뿐만 아니라 예측할 수 없는 실정이다.

본 논문에서는 지금까지 연구되고 있는 폭주제어 기법들을 비교 분석할 것이며, 이를 토대로 차세대 인터넷에 적합한 폭주제어 기법을 제안한다.

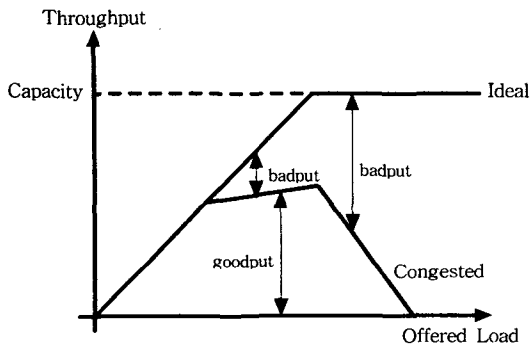
본 논문의 구성은 다음과 같다. 2장에서는 기존의 폭주제어 기법에 대하여 비교 분석을 할 것이며, 3장에서는 2장에서 비교 분석된 결과를 토대로 차세대 인터넷상에서 적용할 수 있는 새로운 폭주제어 기법을 제안한다. 마지막으로 4장에서는 본 논문의 결론을 맺는다.

### II. Congestion Control

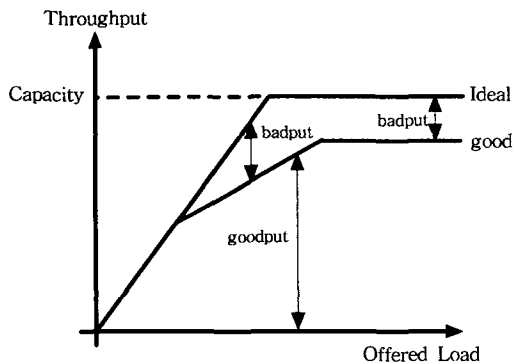
#### 2.1 개요

인터넷상에서 사용중인 대부분 전송 프로토콜은 중간 폭주제어를 제공하는 TCP(Transmission Control Protocol)이다. TCP의 동작은 패킷을 잃어버리지 않는 한 패킷의 전송률을 증가시키는 것이다.

망 폭주는 버퍼 넘침(buffer overflow)으로 인해 패킷이 손실되는데, 이럴 경우 망은 손실된 패킷에 대해 재전송을 요구한다. 패킷이 재전송 되면 throughput은 현저히 감소하게 된다. 폭주를 제어하지 못하면 goodput이라는 throughput은 감소되며, 이에 따라 모든 대역은 그림1의 (a)에 보는 바와 같이 재전송을 위해 소모된다. 이와 같은 현상을 congestion collapse라 한다. 따라서 폭주제어의 역할은 의미없는 재전송(badput)을 줄이는 것이며, 그림1의 (b)에 보여진 바와 같이 적정 수준에서의 goodput을 유지하는 것이다.[14]



(a) Network behaviour under congestion



(b) Network behaviour under controlled congestion

그림1. 폭주에 따른 망의 상태

Fig.1 Network Behavior due to congestion

2.2 기법들의 비교 분석

대부분의 패킷 교환망에서는 미리 확보된 자원없이

망으로 패킷을 전송하며, 망은 그 패킷을 처리하기 위해 최선(best effort)을 다한다. Best-Effort 서비스의 QoS는 망의 행위(network action) 뿐만 아니라 제공된 부하(offered load)에 따라 달라진다. Best-effort 서비스에서 망은 가능한 빨리 패킷을 전송하기 위해 노력하지만 전송된 QoS에 대해서 정량적인 보장(quantitative assurance)은 할 수 없다. 즉, 컴퓨터 트래픽은 예측할 수 없고, 버스티한 성질로 인하여 질적인 보장을 할 수 없다.

폭주제어 알고리즘은 open loop와 closed loop로 구분된다. Open loop는 국부적 링크의 전송대역, 시스템에 있는 이용 가능한 대역과 같은 국부적인 노드의 지식을 기반으로 제어를 수행한다. 제어 지점은 source와 destination에서 이루어질 수 있으나 robust하지 않으며, 모든 트래픽 패턴에 대해 네트워크를 보호할 수 없다. Closed loop는 source에 대한 feedback 정보를 이용하여 제어가 이루어지며, feedback은 global 또는 local일 수 있다. 이들 알고리즘은 네트워크 성능을 능동적으로 감시할 수 있으며, 이들 feedback에 포함된 제어 형태는 implicit 또는 explicit 일 수 있다. Explicit 제어는 piggyback이라고 하는 메시지로 명확하게 보내질 수 있으며, 반면에 implicit는 확인신호의 time delay 또는 패킷의 도착률을 이용하여 제어할 수 있다.

표1 패킷 탈락기법 분류

Table 1 Classification of Packet Dropping Policies

Criteria	Environment		Congestion		Threshold		stat info		Behavior		
	ATM	IP	회피	제어	없음	global	연결정보	global	연결정보	종역	동역
DF	x	x		x	x			x		x	
PPD	x			x	x			x		x	
EPD	x		x			x		x		x	
SD	x		x			x			x	x	
EPDFBA	x		x			x			x		x
ESPD	x		x			x		x		x	
EPD Hys	x		x			x		x		x	
Fair Hys	x		x			x			x	x	
APPDHEPD	x		x			x			x	x	
APPDPPD	x			x	x				x	x	
pPPD	x			x	x				x	x	
DFF	x			x	x				x		x
RED		x	x			x		x			x
FRD		x	x				x		x		x
RIO		x	x			x		x			x

표1은 서로 다른 특성을 갖는 다양한 알고리즘을 정리한 것이다. 이들에 대한 비교 평가는 다음과 같은 요인에 의해 이루어진다.[2,10,17]

- Increase Network Utilization and Application Throughput
- Fairness
- Simplicity of Operation
- Global Synchronization
- Scalability

TCP는 네트워크에 폭주가 발생하면 많은 연결들이 가용자원에 대해 경쟁한다. TCP는 네트워크 링크를 공유하는 개별 연결들의 전송률을 조절하는데, 이를 위해 윈도우 기반(window-based)의 프로토콜을 이용한다. 여기에는 Slow-Start, 폭주회피(congestion avoidance), Fast Retransmit과 Fast Recovery가 있다. 이들 TCP 알고리즘들이 현재 네트워크상에서 갖는 문제는 큐 오버플로우에 의해 패킷이 손실될 때 전송률을 줄이는 것인데, 이는 라우터에서의 패킷드롭과 소스에서 패킷손실 탐지사이의 상당한 시간이 흘러 많은 패킷이 드롭된다는 것이다.[17]

버퍼 오버플로우에 의한 패킷 손실을 막기 위한 방법으로서 적극적인 큐 관리 기법이 제안되었다. 이 기법의 목적인 초기 폭주를 탐지하여 이를 End-host에 통보하는 것인데, 이는 큐 오버플로우나 패킷 손실전에 패킷 전송률을 줄이는 것이다. 이 중 대표적인 기법은 IETF에서 제안된 RED(Random Early Detection)이다.

폭주제어가 게이트웨이에서 이루어지는 요인은 다음과 같다.

첫째, 게이트웨이는 전송지연(propagation delay)과 persistent 큐잉 지연 사이를 구분할 수 있다. 둘째, 시간에 따른 큐잉 특성(queueing behavior)을 갖고, 개별 연결(connection)들의 관계가 패킷 도착 분포 패턴에 의해 제한되기 때문이다. 셋째, gateway는 다양한 RTT, Tolerance of delay, throughput requirement를 갖고, 이들을 많은 연결에 의해 공유되기 때문이다.

RED의 기초가 되는 기법 중의 한 가지는 버퍼가 full 일 때 자동적으로 패킷 드롭이 이루어진다. 큐가 full 일 때의 default 알고리즘은 임의의 새로운 패킷을 드롭시키는 것인데, 이를 Drop-Tail이라 한다. 또 다른 알고리즘은 큐가 full 일 때 새로운 패킷이 도착하면 하나의 패킷은 큐로부터 임의로 선택되어 드롭되는 이를 Random Drop이라 한다. Drop Tail과 Random Drop 게이트웨이의 단점은 많은 연결로부터 패킷이 드롭된다는 것과 동시에 throughput의 감소로 윈도우가

감소한다는 것이다.

Early Random Drop 게이트웨이는 Tail Drop과 Random Drop을 조금 더 향상시킨 것인데 이는 큐 사이즈가 어떤 임계치를 초과할 때마다 고정된 확률을 가지고 들어오는 패킷을 드롭 시킨다는 것이다.

Flyod는 이들 중 어떤 것도 misbehaving 연결을 잘 다룰 수 없다고 지적하고 있다.[12]

또한 Drop-Tail은 다수 연결들 사이에 폭주가 발생한 링크의 이용률이 낮은 global 동기화 문제를 야기한다.[18]

Flyod는 RED(Random Early Detection)라고 하는 active queue management를 제안하고 있다. 이것은 큐 크기가 일시적으로 변함과 동시에 평균 큐잉지연을 제어하는 것이다. 이 방법에서 average queue가 임계치보다 높으면 queue size와 관련된 어떤 확률을 갖는 패킷이 드롭된다. Average queue size를 계산하기 위한 exponentially weighted moving average는 (1)과 같다.[1,11,12]

$$avg = (1 - w_q)avg + w_q * Queue\_Size \quad (1)$$

Average queue size를 계산하기 위해 사용된 파라미터는 표2와 같다.

표2 RED 파라미터

Table 2 RED parameters

Parameter	Function
min <sub>th</sub>	minimum threshold
max <sub>th</sub>	maximum threshold
max <sub>p</sub>	maximum marking probability
w <sub>q</sub>	weight for updating average queue length

그러나 이 기법은 다른 기법에 비해 많은 장점을 지니고 있지만 각 연결이 수신한 대역의 공정성을 확신할 수 없고, 또한 misbehaving 사용자를 명확하게 제어할 수 없는 문제점이 있다. 그러므로 적응적인 트래픽의 비용으로 비적응적인 트래픽에 대한 공정성 및 global synchronization을 막기 위해서는 추가의 기법이 고려되어야 한다.[19]

### III. 새로운 폭주제어 기법

통신산업이 하부구조로 ATM을 사용함에 따라

best-effort like한 서비스는 ABR(Available Bit Rate)로 통합하려는 움직임이 있으며, best-effort 트래픽에 대한 다양한 서비스 모델이 구성되고 있다. 이 중 특히 고려해야 할 사항으로는 best-effort 서비스를 사용하는 응용의 요구사항을 수용하는 것인데, 지연요구는 가능한 작아야 하고(ASAP : as soon as possible), 상용하는 대역 요구는 가능한 많아야(AMAP : as much as possible) 한다는 것이다.[19]

이와 같이 광대역 컴퓨터 망(wide-area computer network)으로 이동함에 따른 인터넷의 특징은 다음과 같다.

① 더 많은 시스템들이 상호 이질적으로 구성되어 개발된 표준에 따른다면 그 표준을 갱신한다는 것은 대단히 어려울 것이다.

② 인터넷상의 다양한 응용이 꾸준히 증가하고 있다.

③ 상업적인 망으로 접어들에 따라 상업적인 서비스의 급속한 변이가 이루어지고 있고, ATM과 같은 다른 네트워크 서비스가 국제적인 통신서비스로 설계되고 있다.

현재 적용중인 RED는 best-effort like한 서비스 중심의 폭주제어기법을 다루고 있다. 이와 같은 환경에 적용하기 위해서는 우선 RED에서 사용된 파라미터의 수정이 필요하다. 따라서 큐 오버플로우로 인한 패킷 손실을 없앨 수 있으며, bottleneck 큐가 empty되더라도 throughput은 그대로 유지가 가능하다.

따라서, Best-effort 서비스에 대한 폭주제어의 기본적인 패러다임은 새로운 상업적 통신망에 적합하도록 재구성되어야 할 것이다.

#### IV. 결론

네트워크 트래픽을 제어하기 위한 폭주제어는 미래의 네트워크 통신을 개발함에 있어 매우 중요한 기술이다. 지금까지 많은 제어기법들이 제안되어 왔고, 그 중 몇몇 기법은 SNA, DNA, Internet 등과 같은 실제 네트워크에 구현되어 이용되고 있다. 그러나 많은 연구 노력에도 불구하고 망 폭주 제어의 문제는 향후 네트워크의 성능이나 규모 측면에서 아직까지 중요한 문제로 남아있다.

본 논문은 폭주제어에 대한 기존의 기법들을 비교 분석하고, 미래 통신망에 적합한 폭주 제어 설계 기법

을 제안하였다. 앞으로 본 연구는 현재의 분석결과를 토대로 시뮬레이션을 수행하여 그 결과를 검토할 것이다.

#### Reference

- [1] S. Flyod, et al, "Recommendations on Queue Management and Congestion Avoidance in the Internet," RFC2309, April, 1998
- [2] \_\_\_\_\_, "Congestion Control Schemes for TCP/IP Networks," \_\_\_\_\_
- [3] Shrikishna Karadiker, Shivkumar Kalyanaraman, Prasad Bagal, "TCP Rate Control,"
- [4] Shiv Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, Fang Lu, Saragur Srinidhi, "Performance of TCP/IP over ABR Services on ATM Networks," Submitted to IEEE/ACM Transactions on Networking, Nov. 1997.
- [5] Mark Gaynor, "Proactive Packet Dropping Methods for TCP Gateways," <http://www.eecs.harvard.edu/~gaynor/final.ps>
- [6] Vincent Rosolen, Olivier Bonaventure and Guy Leduc, "A RED discard strategy for ATM networks and its performance evaluation with TCP/IP traffic,"
- [7] Boming Feng, "Impact of ATM ABR Control on the Performance of TCP-Tahoe and TCP-Reno," IEEE, 1997
- [8] Farooq M. Anjum and Leandros Tassiulas, "Fair Bandwidth Sharing among Adaptive and Non-Adaptive Flows in the Internet, IEEE, 1999
- [9] Sally Flyod, "A Report on Some Recent Developments in TCP Congestion Control," June 5, 2000
- [10] Farooq M. Ankum and Leandros Tassiulas, "Fair Bandwidth Sharing among Adaptive and Non-Adaptive Flows in the Internet," IEEE, 1999.
- [11] Sally Flyod and Kevin Fall, "Router Mechanism to Support End-to-End Congestion Control," \_\_\_\_\_, Feb. 15, 1997
- [12] Sally Flyod and Van Jacobson, "Random Early Detection Gateways for Congestion Avoidance," To appear in the August 1993 IEEE/ACM Transactions on Networking
- [13] S.Keshav and R.Sharama, "Achieving Quality of Service through Network Performance Management," <http://www.cs.cornell.edu/cnrg>
- [14] Moshe Zukerman and Sammy Chan, "Congestion Control by Maintaing Fairness in High Speed Data Networks," IEEE GLOBECOM'94, Nov.28 to Dec. 2, 1994
- [15] Bobby Vandalore, Sonia Fahmy, Raj Jain, Rohit Goyal, Mukul Goyal, "A Definition of General Weighted Fainess and its Support in Explicit Rate Switch Algorithms," Proceedings of ICNP'98
- [16] Migul A. Labrador and Sujata Banerjee, "Packet Dropping Policies for ATM and IP Networks," IEEE Communications Surveys, <http://www.comsoc.org/pubs/surveys>, Vol.2, No.3, 1999
- [17] Wu-chang Feng, Dilip D.Kandlur, Debanjan Saha, Kang G. Shin, "A Self-Configuring RED Gateway," IEEE, 1999
- [18] Sally Floyd, "A Report on Some Developmens in TCP Congestion Control," <http://>
- [19] Bryan Lyles and Scott Shenker, "Congestion Control for Best-Effort Service: Why We Need a New Paradigm," IEEE Network, Vol.10, No.1, Jan. 996.