

# MSS 조절을 통한 내부 네트워크의 성능 향상

오세민\*, 권경희\*

\*단국대학교 전자계산학과

e-mail:oosem@cs.dankook.ac.kr, khkwon@dku.edu

## Improving a internal network performance by controlling MSS

Sae-Min Oh\*, Kyung-Hee Kwon\*

\*Dept of Computer Science, Dan-Kook University

### 요 약

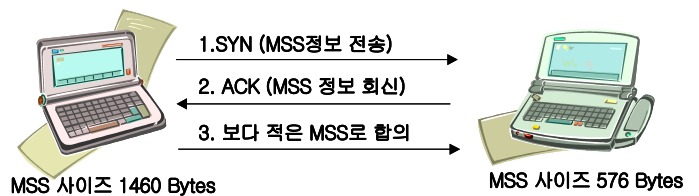
본 논문에서는 내부 LAN(Local area network)에서 MSS(Maximum segment size)의 변화가 네트워크 성능에 미치는 영향에 대하여 분석한다. 내부 LAN에 웹 클라이언트와 웹 서버를 두고 이더넷(Ethernet)이 허용하는 사이즈 내에서 MSS 값의 변화가 네트워크의 성능에 미치는 영향을 분석한다. 처리량(Throughput), 이용률(Utilization), 분실율, 수신율, RTT(Round Trip Time)를 사용하여 성능을 평가하였다. 실험은 네트워크 시뮬레이터 중의 사실상 표준 도구인 NS-2로 한다.

### 1. 서론

인터넷의 보급이 급속히 확대됨에 따라 네트워크의 사용량 또한 크게 많아졌다. 네트워크 사용량의 급속한 증가는 성능(Performance)의 저하를 가져오게 된다. 네트워크의 성능을 향상시키는 방법 중에 많은 비용을 필요로 하지 않는 것이 TCP의 파라미터(Parameter)의 값을 최적화 시키는 것이다. 네트워크 구조가 매우 다양함에도 불구하고 TCP의 여러 파라미터들은 운영체제에 의해 확립적으로 적용되고 있다. 패킷(Packet)의 최대 전송단위를 결정하는 MTU(Maximum Transmission Unit) 사이즈도 그런 예 중에 하나이다. MTU 사이즈를 결정하는 MSS(Maximum Segment Size)의 Default값이 적용되기 때문이다. MSS는 네트워크 성능에 큰 영향을 미치는 TCP 파라미터 중의 하나이다. 특히 제공되는 문서 크기의 변화가 그리 크지 않는 웹의 경우에는 MSS가 네트워크 성능에 많은 영향을 미친다. 웹 클라이언트와 웹 서버는 TCP 연결을 하는데 이때 클라이언트와 서버는 MSS를 합의하는 절차를 갖게 되고 또한 합의된 MSS를 설정하게 된다. 인터넷에서 관습적으로 사용되는 MSS는 536 바이트나 윈

도우와 같은 운영체제에서는 1460 바이트를 사용한다.

일반적으로 패킷이 인터넷을 통해 전달되면서 여러 개의 라우터를 거치게 되며 이들 라우터에는 서로 다른 MSS가 설정되어있을 수 있다. 패킷의 크기가 라우터의 MSS보다 크면 단편화(Fragmentation)가 일어나게 되며 이는 네트워크의 성능을 크게 저하시킨다. 그러나 패킷의 크기를 너무 작게 하면 패킷의 개수가 많아지게 되고 이렇게 되면 헤더의 개수도 많아지게 되는데 이로 인한 오버헤드(Overhead)가 네트워크의 성능을 저하시킨다. 따라서 네트워크를 효율적으로 이용하기 위해서는 적절한 MSS가 필요하다.



<그림 1> MSS 합의 과정

최대의 효율을 내는 MSS의 최대 크기를 결정하는 것은 네트워크의 효율에 큰 영향을 미치는 중요

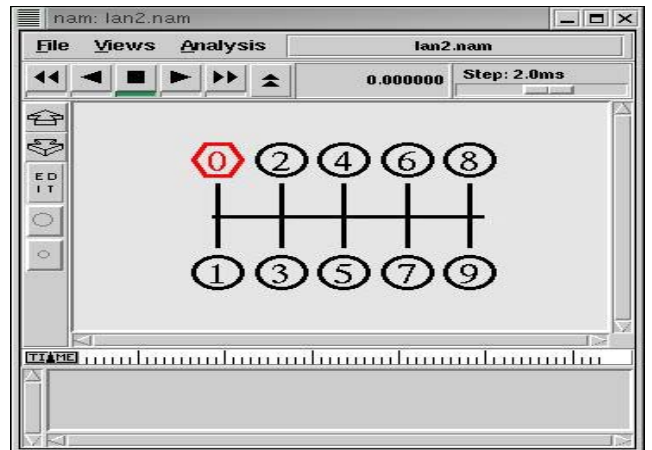
한 일이다. 이런 MSS의 결정은 <그림 1>처럼 이루어진다. 최초의 연결 설정시 송신측은 SYN 패킷과 함께 자신의 MSS 정보를 보내고 수신측 또한 ACK 패킷을 보내며 자신의 MSS정보를 보낸다. 이 중 보다 작은 MSS로 두 노드간의 MSS는 합의되는 것이다. 이런 과정을 통해 단편화의 발생 등을 방지하며 적절한 MSS를 결정 할 수 있다.

최근에는 많은 기업들이 인트라넷(Intranet)과 같은 내부 네트워크를 구성해 회사 업무를 처리하는 경우가 매우 많다. 이런 인트라넷의 경우, 웹 서버와 웹 클라이언트가 존재하게 되고 웹 서버는 내부 클라이언트의 요청에만 응답하는 지역적 제한성이 있는 웹 서버이다. 서버와 클라이언트에 적용되는 MSS는 일반적으로 여러 라우터를 거쳐 웹 서버에 도착하게 되는 WAN(Wide Area Network)에서의 MSS와는 다른 값을 설정할 수 있을 것이다. 여러 라우터를 거치면서 합의과정을 거쳐야 하는 MSS와는 다르게 내부 LAN에서의 최대 효율을 발생시키는 MSS를 찾아내는 것이다. 라우터의 존재 여부와 상관없이 네트워크의 구조가 분명하게 틀린 내부 네트워크에서의 MSS 변화는 네트워크의 성능에 많은 영향을 미칠 것이다. 그래서 본 연구에서의 실험환경은 내부 LAN이다. 내부 LAN에서 MSS가 네트워크의 효율에 미치는 영향을 구체적인 수치로 산출해낸다. 한 개의 웹 서버와 여러 클라이언트로 LAN을 구성하였고 웹 서버와 클라이언트의 위치는 동일한 LAN에 있다. 내부 LAN은 이더넷(Ethernet)환경이며 이더넷의 최대 MTU사이즈 1500(Byte)을 넘어서는 범위는 실험하지 않았다. 시뮬레이션 도구로는 NS-2 (Network Simulator-2)를 사용 하였으며 처리량(Throughput), 이용률(Utilization), 분실율, 수신율, RTT(Round Trip Time)을 조사하였다. 처리량은 네트워크로 유입된 데이터의 양을 말하며 처리량이 커지면 대역폭의 사용 효율이 높아진다는 의미가 되겠지만 지나치게 커지면 트래픽의 혼잡으로 인해 오히려 역효과가 난다. 분실율은 전송된 패킷이 분실되는 비율로 재전송 하게 함으로 패킷 분실율이 커지면 네트워크 성능은 떨어진다. RTT는 응답시간을 나타내며 이것은 분실율과 관계가 된다. 분실율이 많아지면 재전송이 많아지므로 응답시간은 필연적으로 길어지게 되는 것이다. 수신율은 네트워크로 유입된 총 데이터양과 서버와 클라이언트에 의해 수신된 데이터 의 총량과의 비율이다. 이런 여러 가지 수치를 통해 네트워크의 전반에 걸친 성능을 알아볼

수 있다.

## 2. 네트워크의 구성

본 연구에서는 <그림 2>와 같은 토폴로지의 이더넷(Ethernet)을 구성하였다. 이더넷이라는 용어는 IEEE 802.3을 포함하여 Xerox사에서 개발된 Ethernet Specification을 만족하는 모든 LAN을 지칭하는 용어로 사용된다. Ethernet과 IEEE 802.3은 모두 CSMA/CD 방식을 사용하지만 약간의 차이가 있다. IEEE 802.3은 데이터 링크 계층 중 MAC(Medium Access Control)계층만을 정의하고 LLC(Logical Link Control) 계층은 정의하지 않고 기존 WAN의 데이터 링크 계층의 프로토콜을 사용할 수 있게 한다.



<그림 2> 네트워크 토폴로지

LAN의 세부 환경은 <표 1>과 같은 형태로 설정하였다. Mac Layer의 Mac 802.3은 IEEE 802.3과 꼭 같지는 않지만 CSMA/CD를 사용하는 이더넷이며 NS-2에서 사용하는 이더넷 표준이다. 네트워크 토폴로지에서도 보이는 0번 노드는 내부 웹 서버이며 다른 노드는 모두 클라이언트들이다. 여기서 발생하는 단일 패킷의 크기는 일률적으로 MSS 값과 동일하게 설정된다. 하지만, 패킷의 개수는 랜덤하게 설정되어 전체 크기를 랜덤하게 정해준다. 또한 트래픽은 일정시간 간격으로 시뮬레이션이 끝나는 시간까지 무한정으로 발생된다.

Link	Configuration
Bandwidth	10Mbps
Delay	10ms
Link Layer	LL Type, DropTail
Mac Layer	Mac 802.3
Physical Layer	Channel

<표 1> LAN의 세부 환경

세부 환경에서 적용된 값들과 시뮬레이션을 할 때 적용된 웹 서버의 페이지 사이즈는 내부 네트워크의 대표적인 형태인 캠퍼스 네트워크(Campus Network)의 실제 데이터 통계를 바탕으로 설정하였다. 데이터 통계를 참고한 캠퍼스 네트워크는 서울과 천안에 소재한 3곳의 학내 웹 서버 통계 자료들을 바탕으로 하였다. 웹 서버의 페이지 사이즈는 가장 많이 요청한 파일의 크기와 전송된 데이터 중에는 가장 높은 비율을 차지하는 파일의 크기를 고려하여 평균치를 사용했다.

### 3. 시뮬레이션 및 성능분석

#### 3.1 시뮬레이션

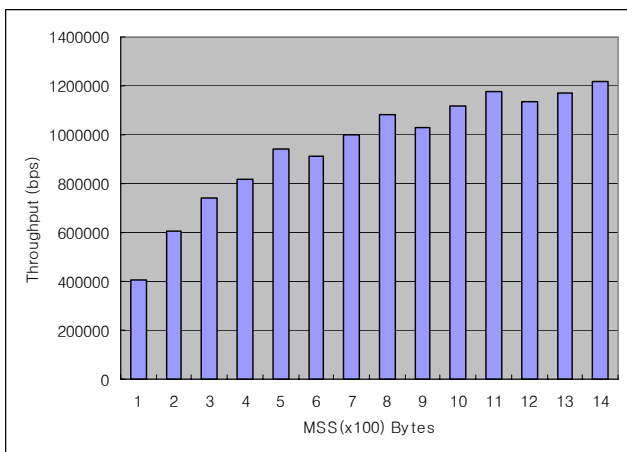
본 연구에서는 다음을 가정하였다.

첫째, TCP의 동작과정에서 송수신되어지는 응답 신호(ACK:ACKnowledge)의 에러 및 분실율은 0%이다.

둘째, 내부 네트워크의 노드들은 반이중 통신(半二重)에 의해 연결되어 있다.

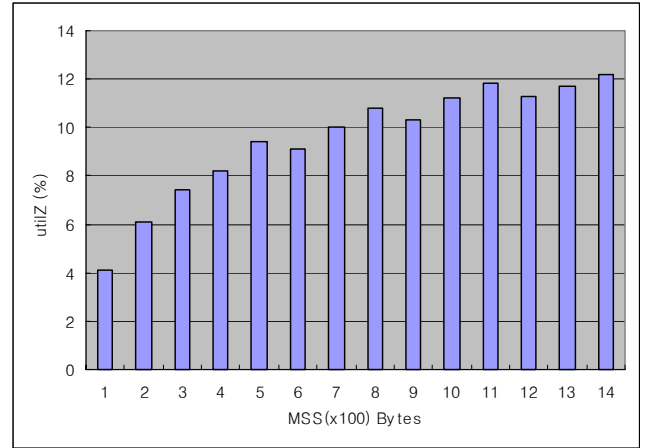
서두에서도 언급했듯이 내부 LAN에서의 클라이언트들이 내부 웹 서버에 접속하는 경우에 대한 시뮬레이션이다. 여기서 MTU 사이즈를 결정하는 MSS 값을 100부터 1400까지 변경해 가며 내부 네트워크의 성능에 미치는 영향에 대하여 조사했다. 네트워크의 성능은 처리량과 이용률, RTT, 수신율과 분실율을 기준으로 조사하였다. 변경하기전 NS-2의 기본값으로 설정되어 있는 MSS값은 576이다.

#### 3.2 네트워크 성능분석



<그림 3> 처리량의 변화

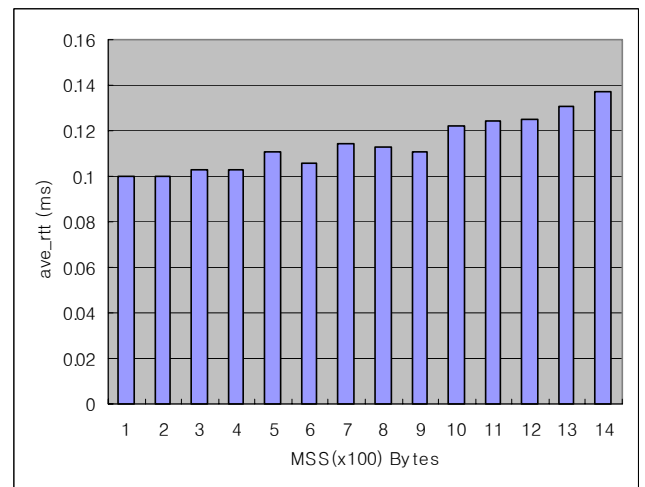
<그림 3>에서 볼 수 있듯이 웹 서버의 MSS값이 1400 바이트일 때 가장 높은 처리량을 보여준다. 최대처리량이 발생하는 지점은 소폭의 득락이 있지만 대체적으로 큰 MSS값을 지정한 쪽이 보다 높은 효율을 보인다.



<그림 4> 이용률의 변화

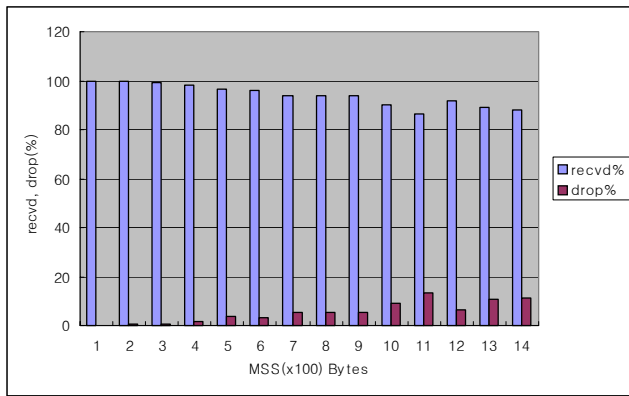
<그림 4>는 <그림 3>과 동일한 패턴을 보여주고 있는 Utilization의 변화에 관한 그림이다. 이용률(Utilization) 또한 웹 서버의 MSS 값이 클수록 좋은 효율을 보여준다.

<그림 5>는 MSS값의 변화에 따른 RTT의 변화를 보여준다. 내부 네트워크를 고려한 시뮬레이션이기 때문에 MSS 값이 RTT에 큰 영향을 미치지 않지만, MSS값이 커지면서 분실 패킷의 발생량이 늘어나는 만큼 재전송이 발생하므로 근소한 폭의 증가를 볼수 있다.



<그림 5> RTT 값의 변화

다음 <그림 6>은 MSS 값의 변화에 따른 수신율



<그림 6> 분실율과 수신율의 변화

(Received-Rate)과 분실율(Drop-Rate)에 관한 변화를 보여준다. 수신율과 분실율의 측정은 모든 클라이언트 노드에서 보낸 데이터의 총량과 서버노드에 도착한 데이터의 총량 사이에 오차를 말한다. 서버의 수신율은 MSS값이 커질수록 소폭으로 감소하고 분실율은 증가함을 볼 수 있다. MSS값이 커질수록 처리량과 이용률은 증가하고 분실율 또한 커진다는 것은 분실율이 증가하는 비율보다 전체 처리량이 증가되는 비율이 더 크다는 것을 의미한다.

#### 4. 결론

데이터 링크 계층에서 TCP 파라미터의 변화를 통해 네트워크의 성능향상을 가져올 수 있다는 사실은 이미 여러 연구에서 확인된 것이다. 본 연구에서는 내부 LAN에서 웹서버와 클라이언트 사이의 MSS 변화가 가져오는 네트워크 성능향상에 대해 연구하였다.

MSS값의 증가에 따라 별도의 경제적인 추가 비용 없이 처리량과 이용률이 보다 좋은 성능을 낸다는 사실은 매우 괄목할 만한 검증이다. 내부 네트워크에서 일반적으로 적용되는 576 바이트의 MSS 값보다 큰 값을 설정해주는 것은 다소 높아지는 분실율을 감안하더라도 향상된 처리량을 가져오게 한다. MSS값의 증가로 처리량과 이용률이 향상되는 것은 분명하지만 분실율의 증가와 수신율이 감소하는 것도 또한 중요한 사실이다. 이를 바탕으로 내부 웹 서버가 제공하는 서비스의 성격에 따라 적절한 MSS값을 설정해야 할 것이다. 큰 MSS값이 무조건 좋은 네트워크 성능을 보장해 주는 것은 아니라는 것을 기억해야 할 것이다.

향후 연구방향은 시뮬레이터 NS-2의 구현상 한계로 인해 현실 세계와 모든 조건을 똑같이 설정할

수 없었던 부분을 실제 운영체제의 TCP 파라미터를 변경하고 조절가능양의 트래픽을 발생시켜 실험함으로써 보다 정확한 결과를 도출하고 분석하는 것이다.

#### 참고문헌

- [1] Larry L.Peterson/Bruce S.Davie "Computer Networks : A Systems Approach", 1999
- [2] 김진희, 권경희 "재전송 타임아웃 간격의 범위 조절에 의한 Web 서버의 성능 향상", 정보처리학회 춘계학술발표 논문집, 제 9권 제 2호, 2002
- [3] 전철완, 권경희 "윈도우 사이즈 조절을 통한 내부 네트워크 성능향상", 정보처리학회 춘계학술대회, VOL.10 NO.01, pp 1189~1192, 2002
- [4] "The Network Simulator ns-2", <http://www.isi.edu/nsnam/ns/>
- [5] "NS by Example", <http://nile.wpi.edu/NS/>
- [6] 권경희, "Web Engineering" 배움터, 2002
- [7] W.Richard Stevens, "TCP/ IP Illustrated, Volume 1", Addison Westey, pp 236~238, pp 340~344, 1994
- [8] 박도용, "무선망에서의 TCP 성능향상 기법에 관한 연구", 건국대학교대학원 석사학위논문, 2002
- [9] Kevin Fall, Kannan Varadhan, "The ns Manual", <http://www.isi.edu/nsnam/ns/doc/index.html>
- [10] L.Wood, "Introducing ns", <http://www.ee.surrey.ac.uk/Personal/L.Wood/ns/>
- [11] Marc Greis, "ns tutorial", <http://www.isi.edu/nsnam/ns/tutorial>
- [12] Chadi Barakat, Eitan Altman, and Walid Dabbous, "On TCP Performance in a Heterogeneous Network", IEEE Communications Magazine, pp 40~46, Jan, 2002
- [13] Konstantinos Michael Pentikousis, "Error Modeling for TCP Performance Evaluation", Master Thesis, State University of New York at Stony Brook, 2000
- [14] RFC 879 The TCP Maximum Segment Size and Related Topics