

Differentiated Service에서 공정한 자원 공유를 위한 Traffic Conditioner 설계

^{*)}이진호 ^{*)}김용수 ^{*)}장정성 ^{*)}김병기
⁰전남대학교 정보통신연구소, ⁰⁰초당대학교 정보통신공학과
jinho@superse.chonnam.ac.kr

A Design Traffic Conditioner for Fairness Resource share in Differentiated Service

^{*)}Jin-Ho Lee ^{*)}Young-Su Kim ^{*)}Kyung-Sung Jang ^{*)}Byung-Ki Kim
⁰Dept. of Computer Science, Chonnam National University

요 약

명시적인 자원 할당과 스케줄링에 의해 QoS를 제공 할 수 있는 방안으로 RSVP가 제안되었으나 내부 구조 변경이나 확장성의 문제를 내포하고 있다. 그래서 IETF에서는 차등 서비스(Differentiated Service)를 이용하여 서비스 영역을 확장하려는 연구가 활발히 진행중이다. DiffServ는 확장성을 고려하여 RSVP와 같은 개별 flow에 대한 QoS 제공이 아닌 flow들의 집합(aggregate)으로 차등적인 서비스를 제공하는 방법이다. 본 논문에서는 aggregated flow로 서비스를 제공할 때 불특정 flow에 대한 QoS 손실을 막기 위해 Token Bucket Meter 처리 전에 사전 모니터링에 의해 자원공유의 공정성과 처리 속도를 향상시킬 수 있는 TC(Traffic Conditioner)를 설계, 구현하여 다수 subnetwork에서의 flow가 자원을 공정하게 공유함을 결과로 보였다.

1. 서론

현재 초고속 패킷처리를 위한 노력의 두 가지 큰 흐름은 현재의 인터넷 라우터 기반의 망을 고속화하는 것과 다른 하나는 링크계층의 스위칭 기술을 사용하여 패킷전달의 고속화와 QoS 제공 능력확장을 위한 연구라 할 수 있다[9]. 본 논문의 배경은 전자인 현재의 인터넷 라우터 기반의 망을 고속화하기 위한 아키텍처인 DiffServ(Differentiated Service)이다. 비록 DiffServ 아키텍처의 핵심이 PHB(Per Hop Behavior)라 할지라도 이는 특정 PHB를 위해 트래픽을 마킹(marking)하는 결과로 얻을 수 있는데 서비스 제공자는 트래픽조정자(Traffic Conditioner)의 구현으로 PHB의 기능을 제공할 수 있으므로 트래픽 조정자의 기능은 매우 중요하다. 즉, 트래픽조정자는 PHB를 충족시킬 수 있는 조정 기능을 하는 것이다.

본 논문에서는 Diffserv 영역으로 향하는 subnetwork들의 flow가 aggregate되는 시점에서 Token Bucket에 미칠 영향을 사전에 모니터링 하여 공정한 자원 공유를 할 수 있도록 조정하고 shaping되거나 dropping될 패킷을 사전에 감지하여 Token Bucket의 처리를 기치지 않게 하여 수행 시간을 단축할 수 있는 TC(Traffic Conditioner)를 제시하였다. 또한 이에 사용될 효율적인 알고리즘을 제시하고 구현 및 결과를 보인다.

먼저 2장에서는 DiffServ와 TC 컴포넌트들에 대해 간략히 소개하고, 3장에서는 TC내의 Meter의 세 가지 타입에 대한 소개와 IETF의 IS(Integrated Services Internet Model)가 근거를 두고 있는 Token Bucket 모

델에 기반한 기존 제안된 모델들을 소개한다. 4장에서는 제안하는 TC모델을 제시하고 이에 사용될 효율적인 알고리즘을 제시한다. 5장에서는 제안하는 모델과 알고리즘에 대한 시뮬레이션을 보이고 결론을 맺는다.

2. 관련연구

2.1 DiffServ(Differentiated Service)

DiffServ는 network provider가 그들 customer에게 provider network내에서 받아들여 질 수 있는 레벨의 서비스를 제공하는 메커니즘이다. 망으로 들어오는 traffic을 망 경계(network boundary)에서 분류하고, PHB 가능하게 조정하여 차등 BA(behavior aggregate)을 할당하고, 각 BA는 단일 DS codepoint에 의해 식별되어 망 내부에서는 단지 DS codepoint에 관련된 PHB에 따라 전송하는 단순한 모델을 기반으로 한다. 장점으로는 복잡한 자원 할당 절차가 불필요, 우선 순위 지정 IP header의 일부를 그대로 사용, 각 호스트 및 라우터의 트래픽제어 모듈이 쉽게 구성 가능, PHB 사용으로 망 내부는 각 application 또는 session당 flow나 전송상태 등을 유지할 필요가 없다는 것 등이 있다.

DS 패러다임은 망의 경계노드(edge node)에서 IP 헤더내의 특정영역을 설정(DS codepoint)하고, 특정영역의 값에 따라 입력된 패킷을 어떻게 전송시킬 것인지 결정(PHB)하고, 각 서비스 규격에 맞추어 요구 사항에 일치하는 방법으로 처리(Traffic Conditioning)하는 것 등을 들 수 있다.

2.2 TC(Traffic Conditioner) 구성 컴포넌트

우선 순위 핸들링은 원하는 서비스 level을 맞추기 위해 네트워크의 edge에 있는 지능적인 전송제어 메커니즘을 이용해서 패킷을 마크하고 우선적으로 처리하고자 하는데 이 지능적인 전송제어 메커니즘이 TC이다.

TC의 구성요소들은 TC의 다른 컴포넌트에서 처리하기 이전에 패킷 헤더의 특정영역을 참조하여 트래픽 스트림에서 특정 패킷을 분류하는 Classifier, Classifier에 의해 분류된 패킷들(traffic stream)의 트래픽 특성을 TCA의 traffic profile을 기반으로 측정하여 in-profile/out-of-profile을 판별하는 Meter, 패킷 헤더의 DS field 영역에 특정 codepoint값을 지정하는 Marker, 트래픽 스트림을 profile에 일치시키기 위해 하나 이상의 패킷을 지연시키거나 폐기시키는 Shaper, Dropper등의 component를 가진다[1].

3. Token Bucket 기반의 기존 제안된 모델

3.1. Meter의 세 가지 타입

첫째, Average Rate Meter는 단순한 meter의 예로 패킷이 들어오는 평균 전송율을 측정하는 방법으로 profile을 고려하여 측정하는데 일치/불일치 중 하나의 level에 상응하는 패킷임을 판단한다.

둘째, Exponential Weighted Moving Average Meter는 다음 계산식과 구현으로 형식화된다.

$$\begin{aligned}
 &avg(n+1) = (1-Gain) * avg(n) + Gain * actual(n+1) \\
 &t(n+1) = t(n) + \Delta \\
 &----- \\
 &if (avg(m) > Arrival Rate) \\
 &\quad non-conforming \\
 &else \\
 &\quad conforming
 \end{aligned}$$

[표1]. EWMA의 계산식과 처리 방법

셋째, Token Bucket Meter는 보다 더 정교한 방법으로 세 개의 파라미터 average rate, packet rate, burst size를 갖는다. 기본적으로 average rate와 패킷의 arrival rate를 비교하는 방법을 사용하며 Token Bucket profile의 일치여부로 측정하는 방법이다[2].

3.2 기존 제안된 모델

srTCM(A Single Rate Three Color Marker), trTCM(A Two Rate Three Color Marker), GTC(A Generic Traffic Conditioner), FM(A Fair Marker)등이 있으며, Average Rate Meter의 방법을 사용하는 TSWTCM(A Time Sliding Window Three Colour Marker)등이 있다.

srTCM은 두 모드(Color-Blind mode, Color-Aware mode)가 있으며 세 개의 traffic parameter-CIR (Committed Information Rate), CBS(Committed Burst Size), EBS(Excess Burst Size)-를 사용하여 traffic stream을 측정하여 green, yellow, red중 하나로 패킷을 마크하는 방법이다[3].

trTCM은 srTCM과 비슷하나 두 개의 rate(PIR, CIR)와 burst size를 기반으로 세 개의 color중 하나로 마크한다. 이 방법은 패킷이 PIR을 초과하면 red로 마크하고, 그렇지 않으면 CIR를 초과하는지 여부에 따라 yellow,

green으로 마크된다[4].

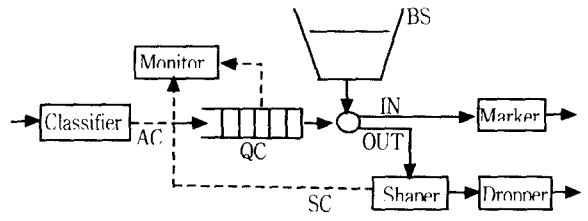
GTC는 [1]에서 명시하는 기능들을 대부분 이행하며 구성되는 컴포넌트와 세 개의 traffic parameter(Token Rate(TR), Token Bucket Size(TBS), Shaping Buffer Size(SBS))에 의해 구성된다[5].

FM은 동일 subscriber network에서 기원이 되는 flow들이 그들간의 공정성을 갖도록 하기 위해 token bucket으로부터 토큰 분배를 제어한다. PER-FLOW CONTROL 컴포넌트 내의 fair 할당 알고리즘을 통해 flow가 다른 flow에 비해 부적절한 token 양을 사용하는지 시험한다[6].

TSWTCM은 트래픽 스트림을 측정하고 두 개의 명세된 rate(CTR-Committed Target Rate, PTR-Peak Target rate)에 의해 측정된 상대적인 throughput에 기반 하여 green, yellow, red중 하나로 패킷을 마크한다[7].

4. 제안 모델 및 알고리즘

3.2절에 열거한 TC내의 Marker들 모두 개별 flow가 아닌 aggregate flow에 대해 시간적인 특성을 측정하기 때문에, burst가 발생되는 시점에 도착하는 패킷은 항상 out-of-profile로 판명이 되어 예기치 않은 손실을 야기한다는 문제를 가지고 있다. 극단적인 경우 두 서브네트워크 망에서 진입하는 flow들간에 경합이 발생할 때 둘 중 하나의 서브네트워크 망에서 오는 모든 flow가 손실을 입을 수도 있다는 점은 심각한 문제라 할 수 있다. 제안하는 모델은 classifier를 통해 각 서비스별로 분류된 aggregate flow에 대해 Monitor라는 컴포넌트를 추가하여 버킷의 burst를 사전에 감지하여 burst를 일으키는 aggregate flow중에서 가장 많은 토큰 비율을 사용하는 망의 패킷을 식별하여 토큰 버킷에서의 측정 없이 바로 shaper로 넘겨지게 하는 방안을 제시하였다.



[그림1] 제안하는 TC 모델

[그림1]에 사용된 파라미터는 다음과 같다.

NAME	설 명
AC	Classifier에 의해 분류된 aggregated flow양
SC	remetering을 위해 되돌려질 flow양
QC	TB의 처리를 대기하는 flow양
IN	측정후 in-profile로 판명된 packet
OUT	측정후 out-of-profile로 판명된 packet
BS	bucket의 size

[표2] 제안 모델에 사용되는 파라미터

위의 제안모델을 수행할 알고리즘은 아래와 같다.

```

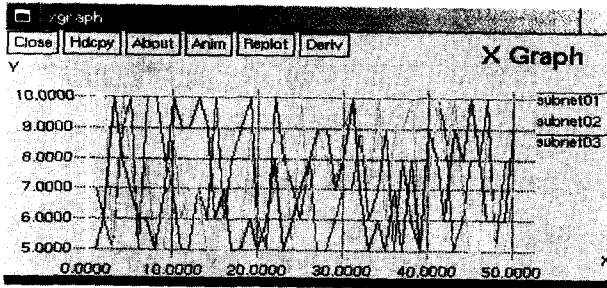
AC = (Classifier를 거친 Aggregated flow양)
QC = AC // 초기 대기 큐에 삽입
while(TRUE) {
    Token Bucket과 Shaper에서의 처리

    // == 다음단계 상황 Monitoring ==
    AC = (Classifier를 거친 Aggregated flow양)
    // 다음단계에 사용될 총 flow양 계산
    TC = AC + SC + QC
    if(TC > BS) { // token bucket burst 발생
        // burst가 SC에서 기인한다면
        if(TC-SC > BS) {
            TC -= SC
            SC flow를 dropper로 전달
        } else {
            TC -= (TC - BS)
            Over된 SC flow를 dropper로 전달
        }
        // SC량 drop후에도 burst가 발생한 경우
        while(TC > BS) {
            aggregated flow중 가장 많은 토큰-
            비율을 사용하는 망의 패킷 식별
            TC --
            SC ++ // shaper로 바로 전송
        }
    } else {
        SC flow를 Remetering을 위해 Feedback
    }
    TC = 0
}
    
```

[표3] 제안 모델의 주요 알고리즘

5. 시뮬레이션

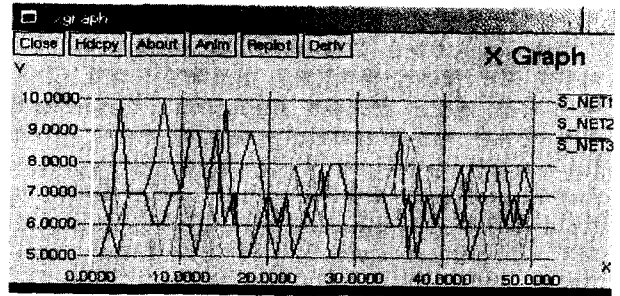
다음은 제안모델의 시뮬레이션 결과이다.



[그림2] 알고리즘 적용전 세 flow의 자원 사용량

[그림2]는 알고리즘 적용전의 세 개의 flow로 구성된 한 개의 aggregated flow중 각각 사용하는 token 사용량 즉, 자원 사용량을 보이고 있다. 이 시뮬레이션에서 지원하는 서비스 class는 aggregated flow에 대해 21Mbps를 지원(세 flow의 합이 21Mbps를 초과하면 burst로 간

주)하도록 구성하여 잦은 burst가 발생하도록 함으로서 자원 공유의 공정함을 보인다.



[그림3] 알고리즘 적용후 세 flow의 자원 사용량

[그림3]에서 알 수 있는 것은 세 flow에 의해 burst가 발생할 경우 [그림2]에서의 하한(5 Mbps 부근)에서의 사용량이 [그림3]에서 그대로 유지되지만 공정성을 어긴 flow들의 사용량이 다른 flow와 비슷하게 자원이 사용됨을 알 수 있다.

6. 결론 및 향후 연구 방향

본 논문에서는 Token Bucket 방식의 단점을 보완하기 위하여 사전 모니터링을 통해 burst를 감지하여 한 서비스 class를 이루는 subnetwork들에서 기인하는 flow들에 대해 공정한 자원 공유를 할 수 있는 TC와 그에 대한 알고리즘을 적용하여 공정한 자원 공유의 결과를 보였다. 향후 연구 방향으로는 TC의 향상된 조정기능에 가장 잘 적용하는 PHB(Per-Hop Behavior, Queuing, Scheduling)에 관한 연구를 깊이 하고자 한다.

참고 문헌

- [1] D. Black, . "An Architecture for Differentiated Services", RFC 2475
- [2] Y. Bernet, A. Smith, S. Black, "A Conceptual Model for Diffserv Routers", Internet Draft, December 1999
- [3] J. Heinanen and R. Guerin, "A Single Rate Three Color Marker," RFC2697, March 1999.
- [4] J. Heinanen and R. Guerin, "A Two Rate Three Color Marker," RFC2698, March 1999.
- [5] Fang-Ching Ou, "A Generic Traffic Conditioner" Internet Draft, October 1999
- [6] Hyogon Kim, "A Fair Marker", INTERNET-DRAFT, October 1999
- [7] Longsong Lin, "A Generic Traffic Conditioner", INTERNET-DRAFT, April 1999
- [8] Peter. F., "Integrated Services Over Differentiated Services", Internet Draft, March 1998
- [9]. 한국전자통신연구원, "ATM상의 인터넷 서비스 기술개론", 진한도서, 1999