

차별화 서비스 망에서 low-delay 멀티캐스트 서비스 제공 방안

A Scheme to Support a Low-delay Multicast Service on Differentiated Services Network

문 영 성* 조 승 윤** 정 병 철***
Young-Song Mun Seoung-Yun Cho Byung-Chul Chung

요 약

차별화 서비스(Differentiated Services) 망에서 실시간 멀티캐스트를 지원하기 위해서 EF(Expedite Forwarding) PHB(Per Hop Behavior)를 사용하여 전송한다면 EF PHB와 멀티캐스트 플로우의 특성 때문에 EF 멀티캐스트 플로우가 증가하게 될 경우 망의 관리나 자원 관리가 어렵다. 또한 LBE(Lower Than Best-effort) PHB를 이용한 멀티캐스트 전송은 전송지연을 보장할 수 없다는 단점이 있다.

이러한 문제를 해결하기 위하여 본 논문에서는 AF(Assured Forwarding) PHB를 이용하여, 허용 가능한 홉 당 지연시간 이내에서 처리를 해줄 수 있는 저지연 멀티캐스트(Low-delay Multicast)라는 새로운 PHB를 제안하고, 이를 위한 차별화 서비스 망에서의 핵심요소인 첫 홉 라우터, 내부 라우터, 경계 라우터들의 큐 구조를 설계하고 각 라우터들에서의 큐 관리 및 스케줄링 기법을 설계한다.

Abstract

To support real-time multicast, the existing DS network may use the EF PHB for transmission. However, because of the dynamic characteristics of the multicast flow and the features of the EF PHB, it is very difficult to make a provision of resources or to manage the network when EF multicast flows increase. Also, the existing LBE PHB can not guarantee the transmission delay bounds required for the real-time multicast traffic. In this paper, a new concept of PHB, namely, low-delay multicast PHB, that may guarantee approximately the per-hop delay is proposed. For the low-delay multicast PHB, router architectures are devised. The queue management algorithm on a low-delay multicast queue and a scheduling method are also devised. This new architecture of the LDM AF class is simple to be implemented.

1. 서 론

인터넷이 급성장하면서 통신회선의 용량 또한 급격히 증가하고 있다. 그러나 통신회선의 용량은 가변적이지 않기 때문에 가입자 수의 증가 또는 시시각각으로 변하는 통신량의 변화에 따라서 적용할 수는 없다. 그래서 가입자는 자신이 최대한으로

얼마만큼의 서비스를 받을 수 있는지에 대한 정보만 있을 뿐, 실제로 어느 정도의 서비스를 받고 있는지에 대한 확신이 없다. 또한 갑자기 통신량이 증가하면 라우터에서는 혼잡 제어가 행해지게 되는데, 이때 각각의 가입자의 플로우에 대해 고려하지 않기 때문에 특정 가입자에게만 피해를 입힐 소지가 있다. 이와 더불어 인터넷 상에서 새로 생겨나는 많은 서비스들이 실시간성을 가지게 됨에 따라서 많은 응용들이 QoS(Quality of Service)의 지원을 필요로 하고 있다. 인터넷 사용자들 또한 비용을 더 들여서라도 질 좋은 서비스를 받고자 하는 욕구가 생겨나고 있다.

* 종신회원 : 숭실대학교 컴퓨터학부 교수
mun@computing.soongsil.ac.kr

** 준 회원 : 숭실대학교 대학원 컴퓨터학부
orchid@sunny.soongsil.ac.kr,

*** 비 회원 : 숭실대학교 대학원 컴퓨터학부
dolcom02@sunny.soongsil.ac.kr

이러한 추세와 요구로 인해 차별화 서비스(DS: Differentiated Services)[1]에 대한 연구가 진행되고 있다. DS는 기존의 시그널링 기반의 RSVP(Resource ReSerVation Protocol)[2] 등의 프로토콜들과는 달리 각각의 플로우에 대한 자원예약 시그널링의 처리가 필요없기 때문에 확장성이 좋은 특성이 있으므로 인터넷에서 QoS를 지원하기 위한 좋은 방안으로 제시되고 있다. DS에서는 간단히 각 가입자에게 첫 홉이 되는 라우터(first hop router)에서 사용자와 계약한 서비스 수준(SLA: Service Level Agreement)에 맞게 사용자의 모든 패킷마다 특정 DSCP(Differentiated Service Code Point)를 설정하여 이 패킷의 DSCP 값을 보고 망 내부에서 자원을 효율적으로 분배하도록 하고 있으며, 혼잡 제어 시에는 각각의 가입자에게 비교적 공평한 패킷의 폐기를 제공하고 있다.

하지만 이러한 DS에서 멀티캐스트[3]를 지원하는 것은 어려운 문제가 되고 있다. 멀티캐스트 패킷 한 개는 일반적인 유니캐스트 패킷 한 개가 필요로 하는 자원보다 많기 때문에 멀티캐스트 패킷과 유니캐스트 패킷 간에 공평성 문제가 생길 뿐만 아니라, 내부 망에서 멀티캐스트를 지원하는 동시에 다른 사용자들의 서비스 수준을 지키는 것이 어렵기 때문이다. 이러한 문제가 발생하게 되는 근본적인 이유는, DS는 구조의 단순성을 위해서 첫 홉 라우터에서 입력 패킷에 특정 DSCP를 마킹한 후에는 내부 라우터들에서는 무조건 그 정해진 DSCP만을 근거로 모든 트래픽 처리를 하기 때문이다. 또한 멀티캐스트는 화상 회의, 인터넷 방송 등 대부분의 실시간성 서비스에 사용될 것으로 예상되기 때문에, 기존의 DS망에서 이러한 실시간 멀티캐스팅을 지원하기 위해서는 저지연 및 저지터를 지향하는 최상위 PHB인 EF[4]를 사용해서 전달해야 했다. 하지만 멀티캐스트의 전달 경로는 수신자의 임의적인 가입/탈퇴에 따라 동적으로 변하는 특성이 있기 때문에, 새로운 경로에 예비 자원이 미리 충분히 할당되지 않거나 적절히 동적으로 자원이 준비[5]되지

않다면, 이 망에서 EF PHB를 이용하여 멀티캐스트 패킷을 전송하는 다른 가입자에게 할당된 자원 뿐만 아니라 다른 PHB의 자원을 침해할 소지가 있다[6].

본 논문에서는 멀티캐스트를 EF PHB보다 코스트가 낮은 AF[7] PHB를 사용하여 지원하는 방안을 고려하였다. 또한 AF에는 폐기순위(drop precedence)와 큐 관리 알고리즘이 사용되기 때문에, AF를 사용하면 멀티캐스트 플로우의 증가에 따른 충격을 완화시킬 수 있다. 특히 AF PHB로 멀티캐스트를 지원할 때 가장 문제가 되는 전송 지연과 전송 지연 변화를 해결하기 위하여 AF의 각 클래스에서 선별적으로 저지연 멀티캐스트를 지원할 수 있도록 한다. 저지연 멀티캐스트를 지원하는 클래스는 기존의 큐와는 별도의 큐를 두어서 멀티캐스트 패킷의 홉 당 최대 지연을 근사적으로 보장하면서 각 클래스의 가장 낮은 폐기 순위 패킷들과 평균적으로 같은 비율의 패킷 폐기율을 보장하여, 멀티캐스트 패킷이 마치 가장 낮은 폐기 순위의 패킷이 전송 지연을 보장받는 것과 같이 전파되도록 한다. AF PHB에서 저지연 멀티캐스트를 지원하는 방안으로 이러한 모델을 제시하여, QoS를 위하여 멀티캐스트를 EF PHB로만 구현할 때 생길 수 있는 문제점들을 개선할 수 있다.

2. DS 망에서 멀티캐스트 지원의 문제점

본 장에서는 DS 망에서 멀티캐스트를 구현할 때 발생할 수 있는 문제점들을 살펴보고, DS에서 멀티캐스트를 지원하기 위한 기존의 제안과 문제점, 본 논문의 접근 방법을 살펴보기로 한다.

2.1 멀티캐스트 플로우와 유니캐스트 플로우의 차이점

망 관리의 입장에서 볼 때 멀티캐스트 플로우와 일반 유니캐스트 플로우의 가장 큰 차이점은

하나의 멀티캐스트 패킷으로 인해 망 내에서 얼마나 많은 포워딩 자원이 필요할지 알 수 없다는 것이다. 즉, 멀티캐스트는 송신자의 의도와는 무관하게 수신자들이 그 그룹에 동적으로 참여하게 되기 때문에 멀티캐스트 전송 경로가 더욱 복잡해지며, 따라서 더 많은 망 내의 자원을 필요로 하게 된다.

또 멀티캐스트 플로우는 유니캐스트 플로우보다 전송 경로를 예측하기 힘든 특성이 있다. 즉, 유니캐스트 마이크로 플로우(micro flow)는 보통 한번 발생한 후부터 그 마이크로 플로우가 종료될 때까지 같은 목적지를 향하게 되지만, 멀티캐스트 마이크로 플로우는 송신자의 의지와는 상관없이 가입자의 참여/탈퇴나 멀티캐스트 라우팅 프로토콜의 트리 구성 방법에 따라서 경로와 목적지가 동적으로 변하게 된다.

2.2 멀티캐스트를 DS 망에서 지원할 때의 문제점

DS에서 멀티캐스트를 지원함에 있어서 가장 큰 문제는 자원의 예비(provisioning) 문제라고 할 수 있다. 즉, 같은 양의 대역폭을 사용하는 유니캐스트 플로우 2개가 망 내에 유입되면, 두 경우에 유발되는 망 부하는 거의 같다고 볼 수 있다. 이러한 사항을 근거로, DS에서는 첫 홉에서 망에 유입되는 양을 조절하거나(예:EF PHB), 특정량을 초과할 경우에 다른 마킹을 하고 이에 따른 취급으로(예: AF PHB) 망 부하를 조절할 수 있다. 하지만 멀티캐스트의 경우 같은 양의 대역폭을 사용하는 멀티캐스트 플로우가 유입되더라도 각각의 경우마다 수신자 상태에 따라서 망 내에서 유발될 망 부하가 크게 틀려질 수 있기 때문에 망 관리 입장에서는 어려운 문제가 된다.

또한 경계 라우터에서의 협상에도 문제가 있다. 만약 멀티캐스트 플로우는 트래픽 양이 망에서 차지하는 비중이 크다면, 멀티캐스트 트래픽의 전송 경로가 동적으로 변하는 특성 때문에 각 경계 라우터에서 이웃한 도메인으로 전송되는 양이

동적으로 심하게 변하게 된다. 이러한 문제를 해결하기 위해서, 트래픽의 변화량에 따라 동적으로 협상을 하는 방법 또는 대역폭을 충분히 여유 있게 협상하는 방법을 사용하거나 서비스의 저하를 감수해야 하는데 이 세 가지 경우 모두 문제점을 가지고 있다. 물론 이러한 트래픽 양의 변화는 내부 라우터에서도 같은 문제를 발생하게 된다.

2.3 멀티캐스트를 EF PHB로 지원할 때의 문제점

EF PHB는 QoS 측면에서 보면, 실시간성의 멀티캐스트 플로우에 적합한 특성을 가지고 있다. 하지만 2.1절에서 언급되었던 멀티캐스트 플로우는 EF PHB에 적합하지 않다. 그 이유는 우선 EF PHB는 최상위 PHB라는 점에 있다. EF PHB를 구현하는 방법은 여러가지가 있겠지만 일반적으로 간단하게 구현할 수 있는 단순 우선 순위로 구현할 경우 하위 PHB들의 서비스가 침해받을 가능성이 높다. 또한 이를 우선 순위가 아닌 스케줄링으로 구현하게 되면 다른 EF 플로우들을 침해할 가능성이 높다. 이 EF PHB는 일반적으로 고가의 서비스를 위한 PHB로 패킷의 입력량이 최대 가능 출력량보다 작은 경우를 가정하고 있기 때문에 최대 가능 출력량을 넘어서는 경우에 대한 제어방법이 없다. 즉, 패킷 폐기의 공평성 제공을 위한 큐 관리 알고리즘이 없으며 가입자들의 대역폭을 보장해주기 힘들다. 물론 동적인 자원예약[5]이 충분히 적절히 동작해 준다면 가능하겠지만, 이 또한 각 노드당 예비 자원이 풍부하고 멀티캐스트 플로우 수준이 이를 넘지 않는 경우에만 해당되며, 멀티캐스트의 전송 경로에 새로운 경로(sub tree)가 생겨날 때마다 대역폭 브로커 등의 관리 유닛의 시그널링이 필요하기 때문에 확장성이 좋은 DS의 장점을 해치게 된다. 이러한 현상은 특히 멀티캐스트 플로우는 비중이 크다면 심각한 문제를 발생시킬 수 있다. 또한 EF PHB는 그 특성상 고급의 서비스를 구현하기 위한

PHB이며, 망의 상태에 무관하게 낮은 지연과 손실을 제공하는 포워딩이 가능해야 하기 때문에 이러한 멀티캐스트의 특성을 수용하기 힘들다.

2.4 멀티캐스트를 지원하기 위한 기존의 제안

멀티캐스트를 DS 망에서 지원하기 위한 제안은 이미 R. Bless와 K. Wehrle에 의해 IETF 드래프트로 제출된 바 있다[6]. 이 드래프트에서는 멀티캐스트 플로우로 야기될 수 있는 서비스의 저하를 지적하면서, 이미 설정된 멀티캐스트 전송 경로에 새로운 경로가 추가될 때 이 새로운 경로로 전달될 복사된 패킷의 DSCP를 원본 패킷의 DSCP로 복사해 넣는 것이 아니라 LBE[8]라는 최하위 PHB로 마킹하고, 적절한 검증절차(예: admission test)를 거친 새로운 경로에 대해서만 대역폭 브로커 등의 관리 유닛이 새로운 경로가 시작되는 노드에 메시지를 보내서 적절한 DSCP로 마킹할 것을 지시하게 하는 방법을 제시하고 있다.

이 방법은 확실히 멀티캐스트로 인해 유발될 수 있는 망 자원의 고갈이나 서비스 저하를 막아 줄 수 있지만, LBE로 서비스되는 경로에 대해서는 실시간성이나 QoS를 전혀 보장할 수 없는 단점이 있다. 또한, 새로운 경로가 추가될 때마다 시그널링이 필요하고 여전히 멀티캐스트 플로우를 발생시키는 송신자들이 유니캐스트 플로우를 발생시키는 송신자들과 구분이 되지 않으며, 망 내에서도 멀티캐스트 플로우를 별도의 DSCP로 구분하지 않기 때문에 추가적인 제어가 불가능하다. 이 제안은 패킷이 경로에 따라서는 송신자의 의도대로, 즉, 원하는 수준의 서비스로 전달되지 않으므로 멀티캐스트를 쓰려는 의도(예: 실시간 서비스)나 기존 DS의 개념과는 거리가 있다.

2.5 본 논문의 접근방법

본 논문에서는 2.3절과 같은 문제점에 착안하

여 AF PHB를 이용하여 멀티캐스트 서비스를 지원하는 방법을 고려하였다. AF는 망 내에서 차지하는 대역폭의 비중이 상대적으로 클 뿐 아니라 폐기 순위가 정해져 있기 때문에 멀티캐스트 플로우로 인한 영향을 EF보다 상대적으로 적게 할 수 있다. 즉, 예를 들어 두 번째 이상의 폐기 순위 패킷들에 대해서 보장하는 폐기율을 탄력적으로 적용한다면, 멀티캐스트 플로우로 인한 충격을 많이 줄일 수 있다. 하지만 AF PHB를 그대로 사용하면, 전송 지연을 보장할 수 없기 때문에 추가적으로 AF 클래스마다 새로운 개념의 PHB인 저지연 멀티캐스트라는 방법을 선택할 수 있도록 했다. 또한 멀티캐스트 플로우를 허가된 가입자들만 발생할 수 있도록 하여, 멀티캐스트 플로우에 대해 망 관리 차원에서 허가 관리가 가능하도록 하였다.

EF PHB를 통하여 망에서 준비된 자원의 상태를 넘어가지 않는 수준의 멀티캐스트가 발생한 경우에는 EF PHB가 멀티캐스트를 지원할 수 있으나, 망에서 준비된 자원의 상태를 넘어가지 않는 수준의 멀티캐스트가 발생하는 경우에는 EF PHB로 멀티캐스트를 지원 시 2.3절에서 지적한 것과 같은 문제점들이 발생한다. 이러한 경우에는 본 논문에서 제시하는 AF 저지연 멀티캐스트 PHB로 멀티캐스트를 지원하여 EF PHB로 지원시 발생하는 문제점을 해결할 수 있다.

다음 장에서는 AF PHB를 이용하여 멀티캐스팅을 지원하기 위한 방안을 제안하고 이를 위해 DS 망에서 필요로 하는 첫 홉 라우터, 내부 라우터, 경계 라우터 등의 핵심 요소들을 설계한다. 각 라우터들 내에서의 큐 관리 및 스케줄링 기법도 설계한다.

3. 제안 방안: 저지연 멀티캐스트 PHB

우선 멀티캐스트 지원을 위한 새로운 서비스부터 정의한다.

3.1 서비스 정의

① AF 멀티캐스트 서비스

AF로 구현되는 서비스에 추가적인 서비스로서 가입자가 멀티캐스트 패킷을 보낼 수 있게 해준다. 이 추가 서비스를 신청하지 않은 가입자가 발생시킨 멀티캐스트 패킷은 첫 홉에서 버려지게 된다. 이 추가 서비스를 신청한 사용자의 멀티캐스트 패킷들은 일반 유니캐스트 패킷과 동일하게 취급되어 사용자가 동의한 프로필에 맞게 DSCP가 설정된다. 이러한 과정을 통해서 DS 도메인의 관리자는 발생할 수 있는 멀티캐스트 패킷의 양을 제한시킬 수 있다. 이러한 멀티캐스트 패킷들은 일반 유니캐스트 패킷처럼 전송 지연을 보장받을 수 없다.

② AF 저지연 멀티캐스트 서비스

AF로 구현되는 서비스에 추가적인 서비스로서 가입자가 특정 멀티캐스트 주소로 전송 지연을 근사적으로 보장받으면서 멀티캐스트 패킷을 전송할 수 있게 해준다. 이 추가 서비스는 복수의 주소로도 신청이 가능하며, 가입자의 프로필에 각각의 항목이 기록되게 된다. 이 항목들은 사전에 협의된 만큼의 대역폭을 공유하게 된다. 즉, 여러 항목을 신청하여 동시에 사용하더라도 이 각각의 항목이 대역폭을 점유하는 것이 아니라, 현재 사용하고 있는 항목들은 같이 대역폭을 공유하게 되는 것이다. 또한 이 대역폭은 사용자의 가장 낮은 폐기 순위(예: green)의 대역폭을 초과할 수 없다. 이렇게 설정된 대역폭 만큼의 멀티캐스트 패킷들은 기존의 폐기 순위를 나타내는 DSCP와는 구분되는 별도의 DSCP값이 설정된다. 또한 DS 망 내에서 가장 낮은 폐기 순위와 비슷한 폐기 확률을 가지며, 홉 당 최대 지연을 근사적으로 보장받게 된다. DS 망 관리자는 자신의 DS 도메인 내에서 AF 클래스 중 어떤 클래스가 이러한 추가적인 서비스를 지원할 것인지 결정할 수 있으며, 일반적으로 AF 클래스 중 상대적으로 예비된 자원이 풍부한 상위

클래스에서 지원할 것을 권고한다.

③ 두 개의 추가 서비스 간의 관계

두 서비스를 같이 신청할 수 있으며, 이러한 경우 AF 저지연 멀티캐스트 서비스가 우선한다. 즉, 멀티캐스트 패킷 중 AF 저지연 멀티캐스트 항목이 있는 목적지의 경우는 AF 저지연 멀티캐스트 서비스가 적용되며, 그렇지 않을 경우는 AF 멀티캐스트 서비스가 적용된다. AF 저지연 멀티캐스트 서비스만 신청한 경우에는 신청된 목록에 있는 주소로만 저지연 멀티캐스트 패킷을 보낼 수 있으며, 그 이외의 멀티캐스트 주소로 목적지 주소가 설정된 패킷들은 폐기된다. 또 AF 멀티캐스트 서비스만 신청한 가입자의 경우는 모든 멀티캐스트 패킷들은 일반 유니캐스트 패킷과 같은 취급을 받으며 전송된다.

3.2 제안하는 PHB 구현을 위한 DS 망 구조

① 가입자 프로필 및 이웃한 도메인 간의 SLA

기존 DS의 가입자의 프로필과 이웃한 도메인 간의 SLA에는 멀티캐스트 관련 필드가 없다. 따라서 본 논문에서는 멀티캐스트 관련 필드들을 추가하였다. 가입자 프로필에 필수적으로 추가되어야 할 것은 AF 멀티캐스트 서비스 가입 여부 필드, 저지연 멀티캐스트 지원 목적지 리스트 필드와 저지연 멀티캐스트에 할당된 대역폭 필드등 세가지이다. 여기서 마지막 필드의 값은 그 가입자의 가장 낮은 폐기 순위의 대역폭보다 작아야 한다. 그 외에도 DS 도메인 관리자의 필요에 따라서 필드를 추가할 수 있다. 이러한 필드들의 내용은 기존의 다른 필드와 마찬가지로 DS에서 정의하는 방식, 즉, DS 관리자에 의하거나 대역폭 브로커에 의해서 동적으로 처리되는 방식, 두 가지를 다 지원해야 한다. 이웃한 도메인 간의 SLA의 경우에는 AF 멀티캐스트 서비스 허락 여부 필드와 저지연 멀티캐스트에 할당된 대역폭 필드 두 가지가 추가적으로 필요하다. DS 도메인 관리

자의 필요에 따라 필드가 추가될 수 있으며, 이 내용 역시 DS에서 정의하는 방식 즉, 정적인 협약과 동적인 협약에 의해서 재협상 될 수 있다.

② 첫 홉 라우터

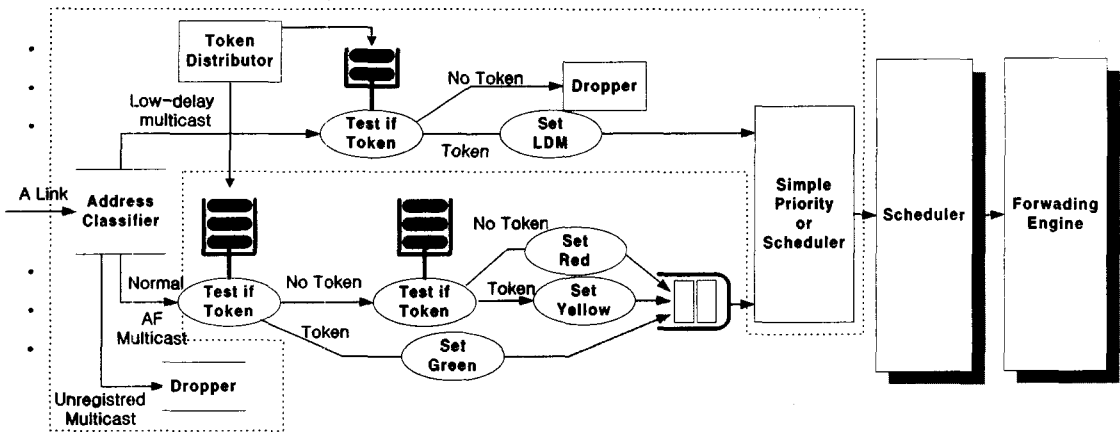
첫 홉 라우터에서는 기존 DS에서 사용되지 않던 몇 가지 구성 성분이 더 필요하며, 본 논문에서 제안하는 첫 홉 라우터의 AF 부분 구성도는 그림 1과 같다. 이 그림에서 점선 안의 부분이 저지연 멀티 캐스트를 처리하기 위해 새로 추가되는 부분에 해당되며, 기존의 첫 홉 라우터의 AF 부분 구성도에 이 부분이 추가된 형태이다. 이 새로운 처리 경로를 따라서 저지연 멀티캐스트 패킷들이 처리된다. 즉, 사용자 프로필에 설정된 저지연 멀티캐스트 대역폭에 맞도록 토큰 분배기에 의해서 분배된 토큰이 생성되며, 패킷이 마커에 도착했을 때 토큰이 존재하면(실제로는 패킷의 크기와 토큰을 비교해서 토큰이 더 큰지를 비교하는 방식으로 구현이 가능하다), 저지연 멀티캐스트에 해당되는 별도의 DSCP로 마킹되어 포워딩 된다. 점선 밖 부분은 일반적인 AF 패킷이나 AF 멀티캐스트 패킷들이 처리되는 경로이며, 음영이 있는 구성성분은 다른 PHB에 의해서도 공유되는 부분이다. AF의 마커는 일반적인 3컬러 마커(Three Color Maker) [9]를 따랐다.

• 주소 분류기(address classifier)

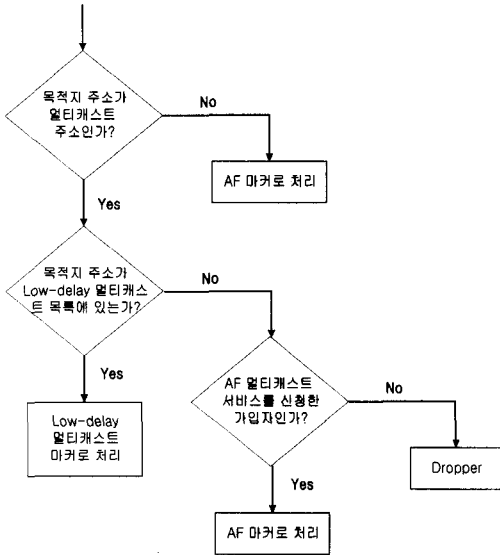
주소 분류기는 가입자 패킷의 목적지 주소를 보고 패킷을 분류한다. 여기서는 그림 2와 같은 결정 방법에 따라서 패킷을 분류한다. 이렇게 분류된 패킷은 그림 1에서 볼 수 있듯이, 저지연 멀티캐스트에 등록이 된 패킷들은 다른 패킷들과는 다른 처리 과정을 거치며 AF 멀티캐스트 추가 서비스에 등록된 멀티캐스트 패킷들은 일반 유니캐스트 패킷들과 같은 처리과정을 거침을 알 수 있다. 여기서 저지연 멀티캐스트 추가 서비스나 AF 멀티캐스트 추가 서비스에 등록이 안된 멀티캐스트 패킷들은 폐기된다.

• 토큰 분배기(token distributor)

토큰 분배기의 가장 중요한 역할은 사용자 프로필 중 가장 낮은 폐기 순위의 대역폭과 저지연 멀티캐스트 대역폭이 공유될 수 있도록 하는 것이다. 즉, 저지연 멀티캐스트가 사용되지 않을 때는 원래 사용자가 신청했던 가장 낮은 폐기 순위의 대역폭을 전부 다 사용하고, 저지연 멀티캐스트가 사용될 때는 가장 낮은 폐기 순위의 대역폭은 그만큼 줄어들게 되는 것이다. 이것은 저지연 멀티캐스트 버킷이 넘칠 시에는 green 토큰을 증가시켜주는 방법으로 구현할 수 있다.



(그림 1) 제안하는 DS 첫 홉 라우터의 AF 부분 구성도



(그림 2) 첫 홉 라우터의 주소 분류기 논리

③ 내부 라우터

기존의 DS 망의 구성 성분에 네 가지 구성 성분이 더 필요하며, 이 네 가지가 첨가된 내부 라우터의 구조는 그림 3과 같다. 여기서도 점선 안의 부분이 기존의 DS 내부 라우터에 새로 추가되어야 할 부분이다. A, B 카운터는 각각 Green과 저지연 멀티캐스트 패킷들의 입력량과 출력량

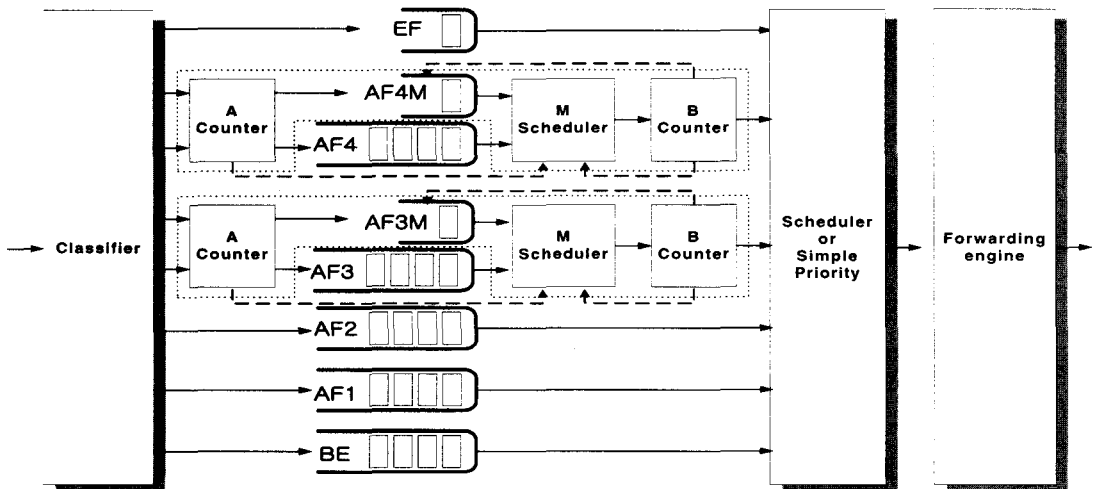
을 측정하는 역할을 하며, M 스케줄러는 한 클래스 내에 속하는 멀티캐스트 큐와 일반 AF 큐의 스케줄링을 담당하게 된다. 그림에서는 상위 2개의 클래스에서 저지연 멀티캐스트를 지원하는 것을 가정하였다. 패킷이 들어오면 주소 분류기에 의해서 각 DSCP에 해당되는 큐로 진행하게 되는데, 물론 이때 다른 클래스의 저지연 멀티캐스트 패킷은 다른 DSCP를 갖기 때문에 구별할 수 있다.

• A 카운터(counter)

내부 라우터는 분류기(classifier)에서 DSCP값만을 보고 패킷을 분류한다. A 카운터는 저지연 멀티캐스트를 지원하는 AFx 클래스의 가장 낮은 폐기 순위 패킷과 AFx 클래스의 저지연 멀티캐스트 패킷의 특정 시간당 입력량을 측정해서 이를 그 특정 시간 T마다 주기적으로 M 스케줄러에게 알리게 된다. 이 정보를 토대로 M 스케줄러는 AFxM과 AFx 큐의 스케줄을 조정하게 된다.

• B 카운터

B 카운터의 동작은 A 카운터와 같은 주기(T)로 동작한다. 저지연 멀티캐스트를 지원하는 AFx 클래스의 가장 낮은 폐기 순위 패킷과 AFx 클래스



(그림 3) 제안하는 DS 내부 라우터 구성도

스의 저지연 멀티캐스트 패킷의 양을 측정하는 것은 비슷하지만, B 카운터는 AFx 큐와 AFxM 큐에서 포워딩 엔진으로 출력되는 패킷의 출력량을 측정하고, 그 측정 결과를 M 큐의 관리유닛과 M 스케줄러 두 군데로 보낸다. 이 정보를 토대로 M 스케줄러는 AFxM과 AFx 큐의 스케줄을 조정하게 되며, M 큐에서는 예상 대기 시간을 계산하게 된다.

• M 큐

M 큐에서는 저지연 멀티캐스트 패킷들이 스케줄링을 대기한다. M 큐의 관리는 관리 유닛이 하게 되며, B 카운터에서 오는 정보를 이용하여 매 주기(T)마다 미리 설정된 홉 당 최대 지연을 초과할 것으로 예상되는 패킷을 폐기한다.

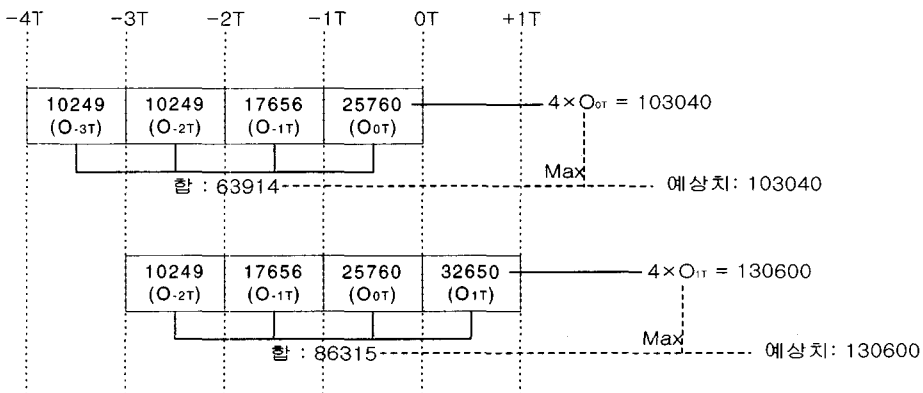
• M 스케줄러(Scheduler)

M 스케줄러는 AFxM 큐와 AFx 큐의 스케줄링을 담당한다. M 스케줄러의 중요한 역할은 저지연 멀티캐스트 패킷과 가장 낮은 폐기 순위(예:green) 패킷의 폐기율을 근사적으로 같게 만들어 주는 것이다. 즉, 혼잡이 발생하여 순간적으로 AFx 큐의 입력률이 출력률보다 높아지는 경우 큐에서 대기하는 패킷의 수가 증가하게 되고, 이렇게 되면 DS에서는 큐 관리 정책에 따라서 폐기 순위가 높은 패킷을 우선적으로 폐기하는데 이때 가

장 낮은 폐기 순위의 패킷도 얼마 간의 확률로 폐기 되도록 하는 것이다. M 큐에서도 마찬가지로 입력률이 출력률보다 높아질 경우 큐에서의 대기 시간을 근사적으로 예측하여 이 예측치가 홉 당 허용된 최대치를 넘을 경우 혹은 순간적으로 큐의 최대 길이를 초과할 만큼 입력 패킷이 폭주하는 경우에는 패킷이 폐기되는데, 이러한 AFx와 AFxM 큐에서의 폐기율이 같아지도록 스케줄링 한다. 이것은 A 카운터와 B 카운터에서 주기적으로 오는 정보를 스케줄링에 반영함으로써 이루어진다.

• T의 설정법

T는 A 카운터와 B 카운터가 M 큐와 M 스케줄러에 정보를 제공하는 시간 단위이며, 각 홉 당 정해진 최대 지연을 2ⁿ으로 나눈 값을 사용한다. 이는 M 큐와 M 스케줄러에서의 계산을 비트 쉬프트(bit shift)로 간략화하기 위함이다. T값이 너무 작으면 A 카운터와 B 카운터의 동기화 문제가 발생할 수 있으며, M 큐와 M 스케줄러의 단위 시간당 계산량이 결과적으로 증가하게 된다. T값이 커지게 되면, M 큐에서는 큐에서의 예상 지연과 실제 지연의 차이가 커질 수 있으며, M 스케줄러에서는 부정확한 스케줄링이 행해질 수 있다. 또한 이 T값이 커지더라도 홉 당 최대 지연을 초과해서는 안된다. T 값은 라우터의 성능



(그림 4) 시간 슬라이딩 윈도우를 이용한 예상 대기시간 계산의 예

에 따라 적절히 조정하는 것이 바람직하다.

• M 큐의 최대 지연 계산법 예

M 큐에는 B 카운터에서 온 정보를 이용하여 T마다 주기적으로 홉 당 최대 지연 시간 내에 처리할 수 있는 패킷의 용량을 예측하여 이를 초과하는 패킷을 폐기한다. 본 논문에서는 시간 슬라이딩 윈도우(Time sliding window)를 사용하는 방법을 제시한다. 여기서 O_x 를 x시간에 B 카운터로부터 받은 저지연 멀티캐스트 패킷의 처리량이라고 정의하고 시간 슬라이딩 윈도우의 크기를 4로 가정하면, 그림 4에서 보는 바와 같이 이번에 B 카운터로부터 받은 T시간 동안 처리된 저지연 패킷의 출력량과 그 전 3T 동안 B 카운터로부터 받은 출력량을 이용하여 홉 당 최대 지연시간 내의 예상 처리 용량을 예측한다. 예를 들어, OT일 시점에서의 계산이라면 $Max((O_{-3T} + O_{-2T} + O_{-1T} + O_{0T}), 4 \times O_{0T})$ (식 1)과 같은 식을 이용한다(여기서 4는 윈도우 수이다). 이 계산 결과로 측정된 용량을 초과하는 위치에 있는 대기 패킷들은 폐기된다.

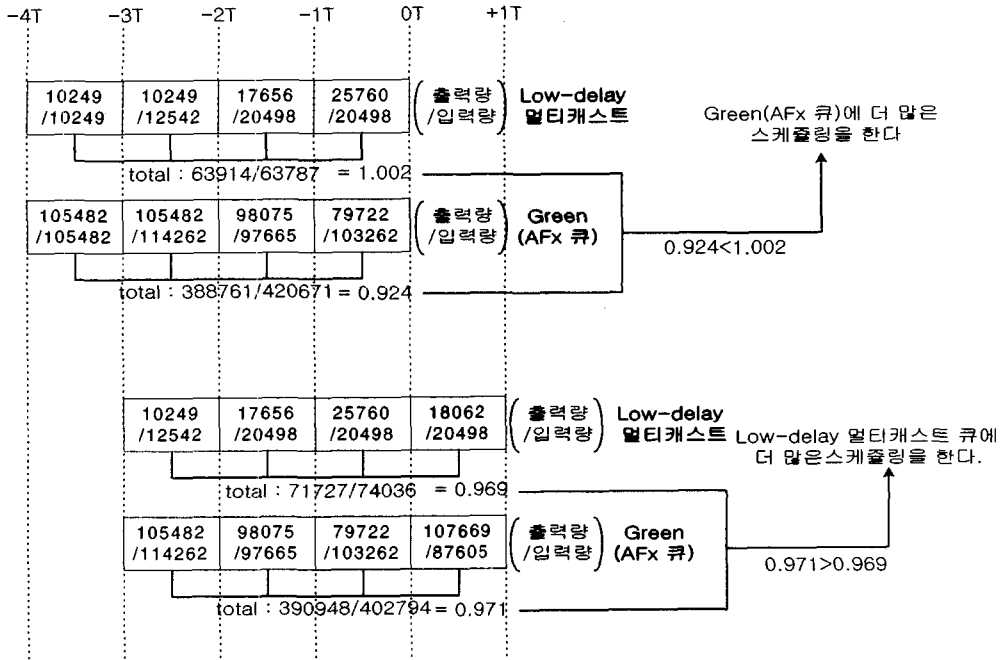
이때 이 식에서 $(O_{-3T} + O_{-2T} + O_{-1T} + O_{0T})$ 와 $4 \times O_{0T}$ 의 두 가지 예측치 중 큰 값을 선택한다. 첫 번째 예측치는 입력되는 패킷의 양이 동적으로 변할 때, 패킷의 입력이 적은 주기(T)동안의 값만으로 예측하게 되면, 실제로는 허용된 홉 당 지연 내에 처리 가능한 패킷도 폐기될 수 있기 때문에 이를 방지하기 위해 몇 개의 주기동안의 평균을 고려하기 위한 것이며, 두 번째 예측치는 저지연 멀티캐스트 플로우가 신규로 발생해서 이 패킷들이 급격히 증가하는 경우, 이러한 플로우의 증가에 더 빨리 대처하기 위한 것이다(실제로는 두 번째 예측치를 사용하지 않고 첫 번째 값만으로 예측하더라도 결국 입력된 저지연 멀티캐스트 패킷들의 양은 많은데 처리된 양이 적다면 이것이 M 스케줄러에 반영되어 점점 저지연 멀티캐스트 패킷들의 처리량이 늘어갈 것이다. 하

지만 이렇게 증가되는 것이 첫 번째 예측치에는 빨리 반영되지 않기 때문에, 이 과정에서 실제적으로 허용된 홉 당 지연 내에 처리할 수 있는 패킷을 폐기할 수 있기 때문이다). 이렇게 두 가지 예측치 중 큰 값을 선택하는 방식은 허용된 홉 당 지연 내에 처리할 수 있는 패킷을 폐기하는 것을 방지해 준다.

그림 4에서 보는 바와 같이 이번에 B 카운터로부터 받은 T시간 동안 처리된 저지연 패킷의 출력량과 그 전 3T 동안 B 카운터로부터 받은 출력량을 이용하여 식 1과 같은 계산으로부터 지금부터 홉 당 최대 지연 시간 내의 예상 처리 용량을 예측한다. 이 연산에서는 비트 쉬프트 한 번과 더하기 연산 세 번이 소요된다. 이 계산 결과로 측정된 용량을 초과하는 위치에 있는 대기 패킷들은 폐기된다.

• M 스케줄링 예

M 스케줄러는 A 카운터와 B 카운터에서 오는 정보를 이용하여 가장 낮은 폐기 순위 패킷과 저지연 멀티캐스트 패킷의 폐기율을 근사적으로 같게 만들어 주는 역할을 한다. 여기서도 시간 슬라이딩 윈도우 기법을 사용할 수 있는데, 이는 그림 5에서 보는 바와 같이 최근 4T 동안의 입력량과 출력량에 대한 상대적인 비율을 스케줄러에 반영함으로써 구현할 수 있다. 이렇게 주기 T마다 최근 4T 동안의 두 종류 패킷들의 입력량 대 출력량의 비를 비교하여, 더 낮은 쪽에 스케줄링 비중을 더 줌으로써 두 종류 패킷들의 폐기율을 비슷하게 맞추어 나가는 것이다. A 카운터는 한 주기(T) 동안의 입력량을, B 카운터는 한 주기(T) 동안의 출력량을 측정한다. 경우에 따라서는 B 카운터에서 보고된 출력량이 A 카운터에서 보고된 입력량보다 많을 수 있는데, 이는 A 카운터에서 보고된 양만큼의 패킷 중 얼마나 처리되었는지를 B 카운터가 보고하는 것이 아니라, 각각 독립적으로 한 주기(T)동안 입력/출력된 양을 주기 T마다 보고하기 때문에, 그림 5와 같은 스케줄링의



(그림 5) Time sliding window를 이용한 M 스케줄링의 예

결과로 입력량보다 많은 처리를 받을 수 있으며, 이러한 경우 큐의 길이 또한 짧아지게 된다.

④ 경계 라우터

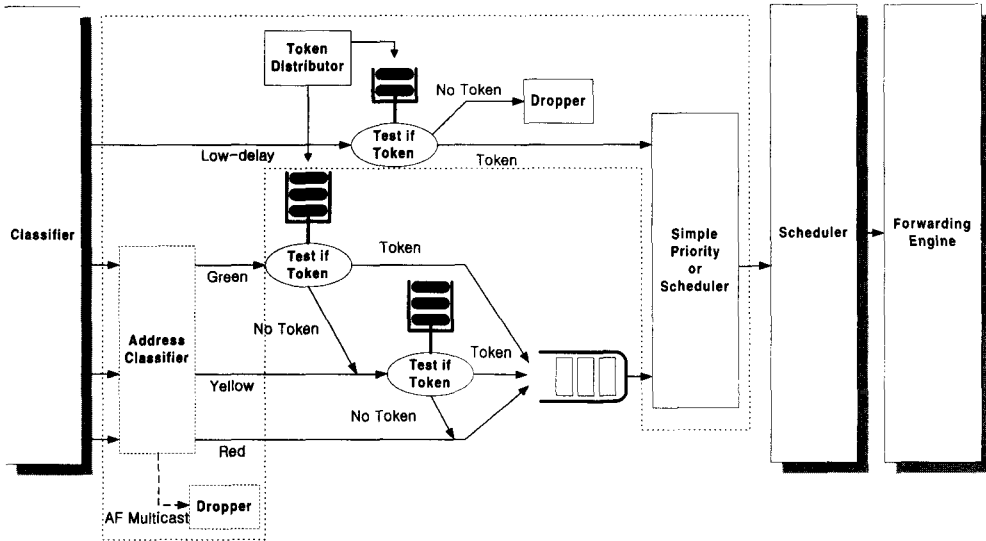
본 논문에서 제안하는 저지연 멀티캐스트를 지원하기 위한 경계 라우터의 AF부분 구조는 그림 6과 같다. 첫 홉 라우터와 마찬가지로 AF부분만을 표시하였으며, 역시 음영이 있는 구성 성분은 다른 PHB에도 공용되는 부분이다. 점선 안의 부분은 기존의 DS 경계 라우터에 새로이 추가될 부분이며, 토큰 분배기와 주소 분류기가 추가적으로 필요하게 된다. 첫 홉 라우터에서와 마찬가지로 저지연 멀티캐스트 패킷은 그림 6 상에서 위쪽에 해당되는 처리경로를 거치며, 일반 AF 패킷과 AF 멀티캐스트 패킷은 아래 쪽의 처리경로를 거쳐서 처리된다. 또한 입구 라우터와 출구 라우터는 그 기본적인 구조가 같다.

- 토큰 분배기

토큰 분배기의 주요기능은 첫 홉 라우터 부분에서 언급하였듯이 가장 낮은 폐기 순위 패킷과 저지연 멀티캐스트 패킷이 대역폭을 공유하도록 하는 것이다. 그리고 동작 방법도 첫 홉 라우터와 같다. 하지만 첫 홉 라우터와 다른 점은 첫 홉 라우터에서는 사용자의 프로필에 따라 토큰의 분배량이 정해지지만, 경계 라우터에서는 이웃한 도메인과의 협약(SLA)에 따라서 정해진다는 것이다. 물론, 이 대역폭의 양은 첫 홉 라우터의 경우와 같이 해당 클래스의 가장 낮은 폐기순위의 대역폭보다 적어야 한다.

- 주소 분류기

저지연 멀티캐스트 지원 경계 라우터에서 주소 분류기는 항상 필요한 구성 성분은 아니다. 이 주소 분류기는 일반 AF 플로우 중 AF 멀티캐스트 패킷만을 분류하는 역할을 한다. 그래서 이웃한 도메인과의 협약의 내용상 이웃한 도메인에서 AF 멀티캐스트를 허락하지 않을 경우에만 필요하게



(그림 6) 제안하는 DS 경계 라우터의 AF 부분 구조

된다. 즉, 일반적인 경우에는 필요하지 않다. 이런 경우에 AF 멀티캐스트 패킷은 이웃한 도메인으로 전파되지 않는다.

3.3 제안하는 서비스의 동작 방법

① 저지연 멀티캐스트 서비스 동작 방법

저지연 멀티캐스트 서비스는 아래와 같은 흐름을 따라서 동작하게 된다.

- 가입자는 최초에 혹은 사용 중에 저지연 멀티캐스트 서비스를 신청할 수 있으며 멀티캐스트 주소와 사용할 대역폭을 정적/동적으로 신청하게 된다.
- DS 도메인 관리자는 가입자의 신청을 받은 후, 가입자의 저지연 멀티캐스트 트래픽을 수용할 수 있는지 판단하여 수용할 수 있다고 판단되면, 가입자의 첫 홉 라우터와 내부 라우터, 경계 라우터에 그 클래스에 해당되는 자원을 예약하거나 이웃한 도메인과의 재협상을 하는 등의 적절한 조치를 취한다.(이는 동적으로 대역폭 브로커를 통해서 행해질 수 있다.) 만약 수용할 수 없다고 판단되면

서비스를 거부하거나, 수용할 수 있을 때까지 유보한다.

- 신청이 허가된 가입자의 패킷 중 저지연 멀티캐스트 서비스에 신청된 주소에 해당되는 목적지를 가진 패킷이 도착하면, 첫 홉 라우터에서는 협상된 대역폭 만큼은 저지연 멀티캐스트 DSCP를 설정하게 된다.
- 내부 라우터에서는 저지연 멀티캐스트 패킷을 해당 클래스의 M 큐로 보내며, 이 M 큐에서 대기하는 동안 홉 당 최대 지연을 넘어설 것으로 예측되는 패킷들은 폐기된다. 이렇게 패킷이 폐기되는 비율이 그 클래스의 가장 낮은 폐기순위 패킷들의 폐기율과 비교되어서 폐기율이 더 높은 쪽이 좀 더 스케줄링 가중치를 받게되며, 이것이 주기적으로 행해져서 결국은 근사적으로 유사한 폐기율을 유지하게 된다.
- 경계 라우터에서는 이웃한 도메인과 협상된 대역폭 만큼의 저지연 멀티캐스트 패킷을 이웃한 도메인으로 보낼 수 있으며, 이를 넘어서는 저지연 패킷들은 폐기된다. DS 도메인 관리자는 이러한 현상이 되도록 생기지 않도록

록 적절한 조치를 취해야 한다.

② AF 멀티캐스트 서비스 동작 방법

AF 멀티캐스트 서비스는 아래와 같은 흐름을 따라서 동작하게 된다.

- 가입자는 최초에 혹은 사용 중에 AF 멀티캐스트 서비스를 신청할 수 있다.
- DS 도메인 관리자는 가입자의 신청을 받아서 가입자의 AF 멀티캐스트 트래픽을 수용할 수 있는지 판단하여, 수용할 수 있다고 판단되면, 가입자의 첫 홉 라우터와 내부 라우터, 경계 라우터에 그 클래스에 해당되는 자원을 예약하거나, 이웃한 도메인과의 재협상을 하는 등의 적절한 조치를 취한다.(이는 동적으로 대역폭 브로커를 통해서 행해질 수 있다.) 만약 수용할 수 없다고 판단되면 서비스를 거부하거나, 수용할 수 있을 때까지 유보한다.
- 신청이 허가된 가입자의 패킷 중 멀티캐스트에 해당되는 패킷은 첫 홉 라우터에서 일반 AF 패킷들과 같이 섞여서 3컬러 마커 등을 통하여 DSCP를 설정하게 된다.
- 내부 라우터에서는 AF 멀티캐스트 패킷들을 일반 AF 패킷과 같은 방법으로 처리한다.
- 경계 라우터에서는 이웃한 도메인과 AF 멀티캐스트 서비스가 허락되었다면, 이 AF 멀티캐스트 패킷들을 일반 AF 패킷들과 같이 처리하며, AF 멀티캐스트 서비스가 허락되지 않았다면, 이 AF 멀티캐스트 패킷들을 폐기하게 된다.

4. 성능 평가 계획

향후 본 논문에서 제안된 방안의 운용 상 발생할 수 있는 문제점을 개선하며 성능 평가를 통해서 제안한 방안의 우수성을 입증한다. 성능 평가

에는 Markov 분석 모델을 이용할 계획이며, 본 논문에서 제시한 Low-delay 멀티캐스트를 지원하는 DS 망에서 Low-delay 멀티캐스트 서비스를 이용하여 멀티캐스트 패킷이 전송될 때와 기존의 DS 망에서 AF PHB를 이용하여 전송될 때의 성능을 분석하여 비교하기로 한다.

5. 결 론

DS 망에서 멀티캐스트를 지원하기 위하여 EF PHB를 이용하거나 LBE PHB를 이용하는 방안이 제안되어 있으나 두 방법 모두 상당한 문제점을 내포하고 있다. 특히, EF PHB를 이용한다면 멀티캐스트 트래픽 증가 시 타 트래픽에 심각한 부정적 영향을 미치게 되며, LBE PHB 이용 시에는 홉당 최대 지연 시간을 만족시킬 수 없는 단점이 있다. 따라서 본 논문에서는 AF PHB에서도 저지연 멀티캐스트를 지원할 수 있는 새로운 PHB를 제안하며 이를 위한 DS 망 내부의 핵심 요소인 라우터들의 구조를 설계한다. 특히 라우터 내에서의 큐 관리 기법 및 스케줄링 기법도 설계한다.

이 새로운 PHB는 AF PHB와 독립적이지 않고, AF 클래스와 AF PHB의 구현 하에 이를 이용하여 동작하도록 하고 있다. 이는 AF PHB가 멀티캐스트 플로우의 동적인 특성에 대처하기 쉬운 특성을 이용하기 위한 것이며, 이로 인해 AF 서비스 가입자가 실시간 멀티캐스트 서비스를 받고자 할 때, EF 서비스에 가입해야 하고 이로 인한 EF 멀티캐스트 플로우의 증가의 영향을 고려해야 하는 기존 DS 망에서 보다 망 자원의 관리 측면에서도 유리할 것으로 예측된다.

본 논문에서 제안하는 구조를 가지게 되는 DS 망은 AF 클래스에서도 홉 당 지연을 보장받을 수 있는 PHB를 가지게 되어 DS 망에서 멀티캐스트 플로우를 수용하기 힘든 특성들을 개선할 수 있게 된다.

참고문헌

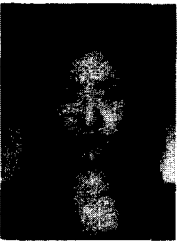
- [1] Y. Bernet, J. Blinder, S. Blake, and M. Carlson, "A Framework for Differentiated Services," *Internet Draft* draft-ietf-diffserv-framework-02.txt, Feb. 1999.
- [2] R. Braden, *et al.*, "RSVP: A New Resource ReSerVation Protocol," *IEEE Network*, Sept. 1993.
- [3] R. Wittman, and M. Zitterbart, "AMnet: active multicasting network," In Proc. ICC'98, June 1998.
- [4] V. Jacobson and K. Poduri, "An Expedited Forwarding PHB," *Internet RFC* 2598, June, 1999.
- [5] C-Y Lee, "Provisioning Resources for multicast Traffic in a Differentiated Services Network," *Internet draft* draft-leecy-multicast-provisioning, Dec. 1998.
- [6] R. Bless and K. Wehrle, "IP Multicast in Differentiated Services Networks," *Internet draft* draft-bless-diffserv-multicast-00.txt, Oct. 1996.
- [7] J. Heinanen, F. Baker and J. Wroclawski, "Assured Forwarding PHB Group," *Internet RFC* 2597, June 1999.
- [8] R. Bless and K. Wehrle, "A Lower Than Best-Effort Per-Hop Behavior," *Internet draft* draft-bless-diffserv-lbe-phb-00.txt, Sept. 1999.
- [9] J. Heinanen, T. Finland and R. Guerin, "A Three Color Marker," *Internet draft* draft-heinanen-diffserv-tcm-01.txt, Feb. 1999.
- [10] S. Blake, D. Black, M. Carlson, and E. Davies, "An Architecture for Differentiated Services," *Internet RFC* 2597, Dec. 1998.
- [11] K. Nichos, V. Jacobson and L. Zhang, "A Two-Bit Differentiated Services Architecture for the Internet," *Internet RFC* 2638, July 1999.
- [12] K. Nichols, "Definition of the Differentiated Services Field(DS Field) in the IPv4 and IPv6 Headers," *Internet RFC* 2474, Dec. 1998.
- [13] S.M. Ross, *Introduction to Probability Models*, ACADEMIC PRESS, 1997.

◎ 저 자 소개 ◎



문 영 성

1983년 연세대학교 전자공학과 졸업(학사)
1986년 Univ. of Alberta 전자공학과 졸업(석사)
1993년 Univ. of Texas, Arlington 전산학과 졸업(박사)
1994~현재 숭실대학교 컴퓨터학부 교수
관심분야 : Mobile IP, IPv6, Security
E-mail : mun@computing.soongsil.ac.kr



조 승 윤

2000년 숭실대학교 컴퓨터학부 졸업(학사)
2000년~현재 숭실대학교 대학원 컴퓨터학부 석사과정
관심분야 : Mobile IP, IPv6, QoS
E-mail : orchid@sunny.soongsil.ac.kr

정 병 철

1997년 한국외국어대학교 전산학과 졸업(학사)
2000년 숭실대학교 대학원 컴퓨터학부 졸업(석사)
관심분야 : QoS, Multicast
E-mail : dolcom02@sunny.soongsil.ac.kr