

# 무선 이동망에서의 이동 호스트를 지원하기 위한 자원 예약 프로토콜

(A Resource Reservation Protocol for Mobile Hosts in  
Wireless Mobile Networks)

김민선<sup>†</sup> 서영주<sup>\*\*</sup> 안성욱<sup>\*\*\*</sup>

(Min-Sun Kim) (Young-Joo Suh) (Syungog An)

**요약** 실시간 서비스를 제공하는데 있어서 호스트의 이동성은 통신경로의 변경으로 인해 QoS에 심각한 영향을 끼치게 된다. 현재 QoS를 위한 시그널링 프로토콜인 자원 예약 프로토콜(Resource ReSerVation Protocol : RSVP)은 고정 호스트만을 가정하고 있으므로 이동 컴퓨팅 환경에서 그대로 사용하기에는 적합하지 않다. 본 논문에서는 이동 호스트를 지원하는 무선 모바일 네트워크에서 RSVP 에이전트를 사용하여 Mobile IP와 연동하는 새로운 자원예약 시그널링 프로토콜인 RSVP-RA(RSVP by RSVP Agent)를 제안한다. 제안된 RSVP-RA 프로토콜은 RSVP 에이전트를 사용하여 차후에 방문이 예상되는 주위의 셀들에 미리 자원을 예약해 줌으로써 이동호스트의 이동성에 따른 유연한 QoS 제공과 종단간 자원예약의 필요성을 제거하여 네트워크의 시그널링 오버헤드를 줄여주는 장점을 갖는다. 본 논문에서는 제안하는 RSVP-RA의 기본 구조 및 이동성, 자원예약 지원기능 등을 설명하고, 기존의 관련연구에서 제안했던 프로토콜과 RSVP-RA 프로토콜을 시그널링 오버헤드 및 패킷지연 등의 성능지표를 시뮬레이션을 통해 비교 평가하였다.

**키워드** : 무선이동네트워크, 예약 프로토콜, QoS(Quality of Service), QoS 시그널링

**Abstract** Providing a mobile host with its required QoS is highly influenced by its mobility. The resource ReSerVation Protocol(RSVP) establishes and maintains a reservation state to ensure a given QoS level along the path from the sender to the receiver. However, RSVP is designed for use in fixed networks and thus it is inadequate in the mobile networking environment where a host changes its point of attachment. In this paper, we propose a new resource reservation protocol, RSVP-RA(RSVP by RSVP Agent) for mobile hosts. Our protocol assumes IETF Mobile IP as a mobility support mechanism. The proposed protocol introduce a new protocol entity - RSVP agent - to manage reservations in a mobile host's current visiting network. RSVP Agent is located in a local network and makes resource reservations in neighboring cells that the mobile host is expected to visit in the future. Thus, the proposed protocol can provide a seamless QoS to the mobile host and significantly improve the scalability problem of RSVP by reducing the end-to-end signalling messages acrossing the backbone networks. The proposed protocols reduce packet delay, bandwidth overhead and the number of RSVP messages to maintain reservation states. We compared the performance of our proposed protocol with other proposed protocols in terms of signalling overhead, packet delay by simulation.

**Key words** : Wireless mobile network, reservation protocol, QoS, QoS signalling

· 본 연구는 과학재단 특장기초, 정보통신부 ITRC 사업의 지원에 의하여 일부 수행되었음.

† 비회원 : 한국루슨트테크놀로지스 연구원  
minsun@lucent.com

\*\* 종신회원 : 포항공과대학교 컴퓨터공학과 교수  
yjsuh@postech.ac.kr

\*\*\* 종신회원 : 배재대학교 컴퓨터공학과 교수  
sungohk@mail.paichai.ac.kr

논문접수 : 2001년 10월 31일  
심사완료 : 2002년 2월 26일

## 1. 서론

모바일 네트워크에서 이동 호스트에 QoS를 제공하는 것은 제한된 무선링크의 자원과 호스트의 이동성 때문에 고정 네트워크에서보다 훨씬 어렵다고 할 수 있다. 이동 호스트가 핸드오프(handoff)를 하게 되면 다양한 QoS의 변화를 겪을 수 있는데, 이는 현재 이동 호스트가 위치해 있는 곳으로 자원을 예약하여 놓아도 이동 호스트가 다른 지역으로 위치를 옮기게 되면 현재의 예약은 더 이상 필요가 없어지고 새로운 경로의 예약을 필요로 하기 때문이다. 따라서 위치의 변화로 인해서 현재 예약되어 있는 자원이 쓸모 없게 되므로 다양한 QoS의 변화를 경험할 수 있게 된다. 그러므로 끊임이 없는 QoS를 제공하기 위해서는 이동 호스트가 이동하리라고 예상되는 모든 지역으로 자원을 예약해 놓아야 한다는 생각을 할 수 있다. 그러나 이러한 방식은 빈번한 자원예약을 위해 백본 네트워크에 상당한 오버헤드를 가져오는 방식이고, 이동 호스트가 이동해 갈 모든 지역을 예측한다는 것조차 어려운 일이라고 할 수 있다. 현재의 IETF RSVP[1]는 송신자와 수신자 사이에 주어질 QoS수준을 보장해 주기 위하여 자원을 예약하고 예약된 자원의 상태를 유지하여 주는 프로토콜이다. RSVP는 고정 네트워크를 가정하여 설계된 프로토콜이라서 모바일 네트워킹 환경에서는 빈번한 이동성에 의한 자원예약 시그널링 오버헤드 및 지연 등으로 그대로 사용하기에는 적합하지 못하다.

본 논문에서는 이동성과 QoS를 동시에 제공하면서 백본 네트워크에서의 시그널링 오버헤드를 개선한, 모바일 네트워크에서의 새로운 자원예약 프로토콜을 제안하고자 한다. 제2장에서는 기존의 연구, 그리고 그 문제점에 대하여 살펴보도록 한다. 제3장에서는 본 논문에서 제안하는 RSVP-RA(RSVP by RSVP Agent) 프로토콜에 대해 상세히 설명한다. 제4장에서는 다양한 시뮬레이션을 통하여 제안된 RSVP-RA와 기존 프로토콜들과 성능을 비교 평가한다. 마지막으로 제5장에서 요약 및 향후 연구로 결론을 맺는다.

## 2. 기존 연구 및 문제점 분석

Talukdar는 [2]에서 MRSVP라고 불리는 시그널링 프로토콜을 제안하였다. MRSVP에서는 이동 호스트의 이동성이 예측 가능하다는 가정하에서 MSPEC(*mobility specification*)을 생성한다. MSPEC은 이동 호스트가 가지는 플로우의 라이프 타임(life time) 동안에 이동이 예상되어 지는 셀(cell)들로 구성이 되어있다. 송신자는

이동호스트가 제공하는 MSPEC을 기반으로 이동 호스트가 이동해 갈 모든 지역으로 예약을 하고 있기 때문에 대역폭의 낭비가 심하고, 예약된 모든 경로로 주기적인 갱신(refresh) 메시지가 계속해서 전송되어야 하기 때문에 백본 네트워크에서 자원예약 상태를 유지하기 위한 비용이 크고, 이동 호스트의 이동성이 예측 가능하다는 가정은 좀 무리가 있다고 할 수 있다.

[3][4]에서는 QoS 도메인과 라우팅 도메인이라는 개념을 도입한 프로토콜을 제안하였다. 이 프로토콜은 계층적인 구조의 라우팅과 QoS의 지원으로 핸드오프 시의 지연을 없애고, 수동 형태의 자원예약 양을 줄인 효과적인 프로토콜이다. 그러나 이 프로토콜은 캠퍼스 네트워크의 크기만 가정하고 있고 다른 AS(autonomous system) 번호를 가지는 네트워크간의 핸드오프는 지원하지 않는다. 또 3계층 구조(3-Layer Architecture)로 네트워크 형태를 제한하고 있어서 일반적인 네트워크 구조에는 적용할 수 없다는 단점을 가지고 있다.

Terzis는 [5]에서 간단한 QoS 시그널링 프로토콜(Simple-QoS : Simple QoS Signaling Protocol)을 제안하였다. 이 프로토콜은 기존의 RSVP에서 쉽게 확장이 가능한 형태의 시그널링 프로토콜이다. 그리고 mobile IP[6]를 약간 수정하여 쉽게 연동이 가능하다는 장점을 가지고 있다. 그러나 이동 호스트가 새로운 지역으로 핸드오프 한 후에 새로운 데이터 플로우(flow) 경로를 따라서 다시 예약을 해야 하는데, 이때 충분한 자원이 없을 경우 원하는 QoS를 보장받지 못할 수도 있다는 문제와 새로운 경로상에 RSVP터널을 형성하기까지 발생하는 지연 문제를 해결하지 못하였다. 또한 이 프로토콜은 mobile IP를 기본으로 하고 있기 때문에 트라이앵글(triangle) 라우팅 문제를 가지고 있다.

[7]의 연구에서 Chen과 Huang은 이동 호스트의 자원을 예약하기 위한 시그널링 프로토콜을 제안하였는데, 이 프로토콜은 멀티캐스트 경로를 이용하도록 하였다. 따라서 최적의 경로를 따른다고 할 수 있고, 패킷 지연이 거의 없다고 할 수 있다. 그러나 송신자가 잘 알려진 멀티캐스트 주소를 가지는 고정 호스트로 가정을 하였기 때문에 이동호스트와의 유니캐스트에는 적용하기가 적합하지 못하다는 문제가 있다. 그리고 자원 예약을 무선 구간에서만 하는 방식의 연구[8][9]도 제안된 바 있다. 그러나 종단간의 자원 예약이 아니기 때문에 송신자와 이동 호스트 사이의 통신 경로가 보장되지 않으므로 지속적인 QoS 서비스의 요구를 충족시켜 줄 수 없게 된다.

### 3. RSVP-FA 프로토콜

RSVP-FA 프로토콜은 모바일 네트워킹 환경에서 RSVP 시그널링 오버헤드를 줄이기 위한 목적을 가지고 설계되었다. 즉, 예약되어야 하는 대역폭과 자원예약 상태를 유지하기 위한 RSVP 제어 메시지의 오버헤드 관점에서 네트워크의 효율을 높이고자 하는 목적을 가지고 있다. 또한 핸드오프에 의해서 유발되는 패킷 지연을 최소화 할 수 있는 방법도 같이 연구되었다. 이를 위해 기존의 mobile IP[6]와 RSVP[1]를 기반으로 하였으며, RSVP 에이전트라는 새로운 시그널링 엔티티(entity)를 도입하였다.

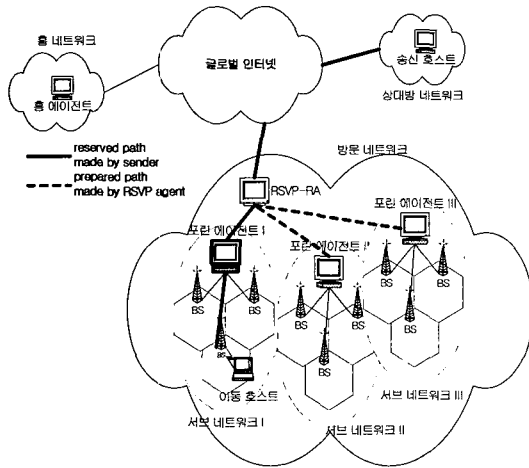


그림 1 RSVP 에이전트를 이용한 자원 예약 구조

그림 1에서 RSVP 에이전트는 포린 에이전트(FA : foreign agent) I,II,III의 상위계층에 존재하고 있으며, 자신으로부터 FA까지 현재 등록된 이동 호스트를 위하여 자원을 예약하여 주는 일을 수행한다. 송신 호스트가 RSVP를 이용하여 PATH 메시지를 전송하면 RSVP 에이전트는 자신이 관리하는 이동호스트에 대한 메시지인 경우 그 PATH 메시지를 가로챈다. 그리고 이동호스트의 현재 FA - 그림 1에서 포린 에이전트(foreign agent) I - 에게 전달해 주고, 핸드오프 후의 자원을 미리 예약하기 위하여 그 이웃에 있는 FA들 - 그림 1에서 포린 에이전트(foreign agent) II,III - 에게도 그 PATH 메시지를 전달하여 주게 된다. 그렇게 함으로써 송신자와 이동 호스트 사이에 자원이 예약되고, RSVP 에이전트와 이웃의 FA사이에도 자원이 예약되어진다. 또한 이러한 자원예약 동작의 투명성(transparency)을

type	S	B	D	M	G	V	Q	lifetime
Home address								
Home agent								
Care-of address								
identification								
Extensions... ( authentication , Rspec )								

그림 2 registration request 메시지 포맷

송신자나 이동 호스트에게 제공함으로써 [2]의 시그널링 오버헤드 및 MSPEC 생성 문제, [3][4]의 패킷 지연 문제 등을 효과적으로 해결하였다.

RSVP-FA 프로토콜은 mobile IP와 함께 동작하고 있기 때문에, 약간의 mobile IP 변경을 필요로 한다. [5]에서와 비슷한 방법으로 등록 요구(registration request) 메시지의 포맷을 바꾸어서, 이동 호스트가 등록 절차 시에 원하는 QoS에 대한 정보를 넘겨주도록 한다-그림 2. 그림에서 'Q' 비트가 1로 세팅된 등록요구 메시지를 받은 FA는 이동 호스트의 홈 에이전트(home agent)에게 등록요구 메시지를 전달해 준다. 그리고 확장부분에 덧붙여져 있는 Rspec을 자신을 관리하는 RSVP 에이전트에게 전달해 준다.

RSVP 에이전트가 이동 호스트를 대신에서 자원예약을 관리하기 위해서 다음 4가지의 새로운 메시지를 추가 하였다;

- 1) *requestQoS* : FA(Foreign Agent)가 자신을 관리하는 RSVP 에이전트에게 보내는 메시지로서 이동 호스트가 원하는 Rspec을 담고 있다.
- 2) *sendRspec* : RSVP 에이전트는 FA로부터 받은 이동 호스트의 Rspec을 주변의 FA들에게로 보내준다.
- 3) *prepareResv* : prepared 예약을 만들기 위해 이웃의 FA들이 RSVP 에이전트에게로 보내는 RESV메시지로서 prepared 예약을 위한 플로우 명세서(Flowspec)와 필터명세(Filterspec)를 포함하는 플로우 식별자(Flow Descriptor)를 담고 있다.
- 4) *switchFlow* : 핸드오프가 일어날 때마다 reserved와 prepared 예약의 변환을 위해서 RSVP 에이전트에 의해서 만들어지는 메시지이다.

표 1은 4종류의 메시지를 사용하여 변경 가능한 자원의 형태를 정리해서 보여주고 있다. 표 1에서 reserved와 prepared 형태의 자원은 모두 PathTear 메시지에 의해 free형태의 자원으로 변하게 되어 있으며, 소프트 스테이트(soft state) 메카니즘을 택한 까닭에 타이머가 종료하면 free형태의 자원으로 바뀌게 되어 있다. prepared와 reserved 사이의 예약 변환은 switchFlow 메시지에 의

표 1 자원 형태의 변경

자원 형태의 변경	해당 메시지
free->reserved	Path-Resv
free->prepared	Path-prepareResv
prepared->reserved	switchFlow
reserved->prepared	switchFlow
prepared->free	PathTear, time-out
reserved->free	PathTear, time-out

해 수행된다. 이상의 추가된 메시지를 사용한 자원 예약 절차의 상세한 과정은 본 장의 뒷부분에서 다시 상세히 다루기로 한다.

다음은 RSVP-RA 프로토콜을 사용하는 주요한 엔터티의 기능과 동작에 대하여 알아본다. RSVP-RA 프로토콜은 4가지 엔터티인 송신자, 이동 호스트, FA(Foreign Agent) 그리고 RSVP 에이전트를 통하여 모바일 네트워크에서 QoS의 지원을 위한 자원 예약을 수행한다.

1) 송신자(sender)

경로최적화[10]를 이용하여 통신하고 있을 때, 이동 호스트와 통신하는 상대방은 이동 호스트의 COA(Care of address)를 관리하는 바인딩 캐쉬(binding cache)를 가지고 있게 된다. 송신자는 이동 호스트와의 통신 경로 예약을 위해 이동 호스트를 향해 PATH 메시지를 전송하게 되는데, FA(Foreign Agent)까지는 RSVP 터널을 통해서 전달하게 되고, FA부터 이동 호스트까지는 바로 전달이 된다. 송신자는 이동 호스트의 COA를 향해 RSVP 터널을 만들게 되는데, 인캡슐레이터(encapsulator)는 송신자 자신이 되고 디캡슐레이터(decapsulator)는 이동 호스트가 현재 위치한 곳의 FA 주소 즉, COA가 된다. 한번 RSVP 터널이 형성되면 그 터널을 통해서 통신이 이루어지게 될 것이다. 후에 이동 호스트가 핸드오프를 하게 되어 새로운 COA를 알게 되면 이동 호스트는 RSVP의 갱신주기까지 기다리지 않고 그 새로운 COA를 향해서 PATH 메시지를 전송하여 RSVP 터널을 만들어야 한다. 이때부터 새로운 RSVP 터널이 만들어지기 전까지는 통신 경로로 미리 만들어져 있는 RSVP 터널을 이용하여야 하는데, 이를 지원해 주기 위해서 송신자는 지연 바인딩(delay binding)을 수행하여야 한다. 지연 바인딩(delay binding)이란 데이터를 전송할 때 바깥쪽 IP 목적지 주소로 이전의 COA를 사용하는 것이다. 중간의 라우터들은 예약된 플로우를 확인하기 위해서 IP 헤더의 목적지 주소를 검사하여 예약된 플로우인지 아닌지를 판단할 뿐, 해당 패킷이 인캡슐레이션(encapsulation) 되어 있는지는 관심이 없다. 따라서 바깥쪽 IP가 새로운 COA로 바뀌어

있으면 중간의 라우터들은 예약된 플로우로 판단할 수가 없게 된다. 그러므로 이전에 예약된 자원을 이용하려면 송신자는 바깥쪽 IP주소를 새로운 COA가 아닌 이전의 COA로 설정해야 한다. 이렇게 지연 바인딩(delay binding)을 통해서 전송된 데이터들은 RSVP 에이전트에 의해 가로채어져서 이동 호스트가 현재 위치한 새로운 FA에게 전달해 줌으로써, 새로운 RSVP 터널이 형성되기까지 예약된 경로의 자원을 사용할 수 있게 되어 QoS서비스의 저하 현상이 일어나지 않게 된다. 새로운 COA를 향하는 RSVP 터널이 형성된 후에는 새로운 COA로 자원이 예약되었으므로 지연 바인딩(delay binding)의 수행을 멈추고 바인딩 캐쉬(binding cache)의 값을 새로운 COA로 저장하고 데이터 전송의 바깥쪽 IP 목적지 주소도 새로운 COA로 설정한다.

2) 이동 호스트(mobile host)

이동 호스트는 FA(Foreign Agent)에게 QoS를 신청하도록 한다. 이는 위에서 언급한 mobile IP[6]의 등록 과정 중의 일부로 진행된다. 등록요청 메시지의 'Q' 비트 필드를 1로 설정하고, 확장부분에 자신이 원하는 QoS 스펙(spec)을 담은 Rspec을 덧붙여 전송함으로써 이루어진다.

3) 포린 에이전트(FA: Foreign Agent)

QoS를 제공해 줄 수 있는 모든 FA들은 RSVP 에이전트에 의해서 관리되게 된다. 따라서 각 FA마다 자신을 관리하는 RSVP 에이전트를 알고 있어야 한다. 이는 관리자의 설정으로 정해질 수도 있고, 동적으로 찾을 수도 있을 것이다. 본 논문에서는 동적으로 찾는 프로토콜은 범위에 벗어나므로 기술하지 않도록 하겠다. 등록요청 메시지의 'Q' 비트가 1로 설정된 메시지를 받게 되면, FA는 RSVP 에이전트에게로 이동 호스트가 QoS신청을 했음을 알리는 requestQoS 메시지를 전송하게 된다. FA가 이동성을 예측하는 알고리즘이 수행 중이면 핸드오프가 예측되는 이웃의 FA를 담아서 보내 줄 수도 있다. 이 경우에는 RSVP 에이전트가 이웃의 FA들과 형성해야 하는 prepared 형태의 자원예약의 부하를 줄일 수 있게 된다.

4) RSVP 에이전트

RSVP 에이전트는 FA를 통해서 QoS신청을 해 온 이동 호스트에 관한 리스트를 관리하고 있다. 일단 이동 호스트의 QoS신청을 받아 들이면 그 이동 호스트가 이동하리라고 예상되는 지역으로 RSVP 터널을 준비한다. FA가 예측 알고리즘이 수행 중이면 핸드오프가 예상되는 지역에 대한 정보를 제공하므로 그 지역으로만 RSVP 터널을 형성한다. 그러나 예측 알고리즘이 수행

중이지 않아서 이동이 예측되는 셀에 대한 정보를 주지 않는다면, 현재 이동 호스트가 위치한 지역의 모든 이웃 셀로 RSVP 터널을 만들도록 한다. RSVP 에이전트는 FA가 보내온 requestQoS 메시지에 담겨있는 Rspec을 sendRspec 메시지에 담아서 이동이 예측되는 셀(예측 알고리즘이 동작할 경우)이나 주변의 모든 셀(예측 알고리즘이 동작하지 않을 경우)로 전송한다. 이 메시지를 받은 FA는 그 메시지에 있는 이동 호스트와 Rspec을 기초로 해서 후에 관련된 PATH 메시지에 응답하게 된다. 이렇게 RSVP 터널을 형성할 준비를 끝내놓은 후, 이동 호스트를 향하는 PATH 메시지가 도착하게 되면 RSVP 에이전트는 현재 FA에게 전달해 줌과 동시에 sendRspec 메시지를 전송한 FA들에게로 PATH 메시지를 전송해 준다. 이 PATH 메시지의 SESSION\_ASSOC 오브젝트[11]안의 중단간 세션은 '송신자와 이동 호스트'가 되고 터널 세션은 'RSVP 에이전트와 FA'가 된다. 이때 RSVP 터널이 형성되는데, R<sub>entry</sub>는 RSVP 에이전트가 되고 R<sub>exit</sub>는 각 FA들이 된다. 이것은 RSVP 에이전트가 자신을 지나는 RSVP 터널의 내부를 훑쳐봄으로써 가능하게 된다. 후에 핸드오프가 일어나게 되면 RSVP 에이전트는 switchFlow 메시지를 통해서 미리 준비해 놓은 RSVP 터널을 이용하여 끊김이 없는 QoS를 제공해 줄 수 있게 된다. 이 때 RSVP 에이전트는 패킷 포워딩(packet forwarding) 서비스를 통하여 핸드오프를 지원해 주게 되는데, 그 자세한 설명은 본 장의 후반에서 자세히 설명하고자 한다.

**3.1 RSVP-FA를 사용한 자원 예약절차**

본 절에서는 앞에서 언급한 바와 같이 RSVP-FA에서의 자원 예약절차에 대하여 상세히 설명하고자 한다. 제안하는 RSVP-RA 프로토콜의 자원 예약의 과정을 다음 그림 3에 보였다. 자원의 예약과정은 다음의 7단계로 설명할 수 있다;

0. 이동 호스트는 등록 과정 중에 FA(Foreign Agent)에게 QoS신청을 한다.
1. FA는 자신의 RSVP 에이전트에게 requestQoS 메시지에 QoS스펙을 담아서 전송한다.
2. requestQoS 메시지를 받은 RSVP 에이전트는 이동 호스트의 핸드오프가 예상되는 지역의 FA에게 이동 호스트가 원하는 QoS 스펙(spec)을 담아서 sendRspec 메시지를 전송한다.
3. RSVP 에이전트는 등록된 이동 호스트의 COA를 향한 PATH 메시지를 받으면 현재의 FA에게 전달해 주면서 메시지를 복사해 놓는다.
4. 복사한 PATH 메시지의 터널세션을 'RSVP 에이

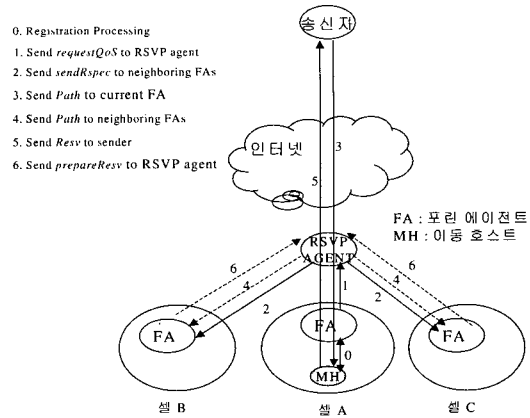


그림 3 자원 예약 설정 과정

전트의 이웃 FA'로 수정하여 전송한다.

5. PATH 메시지를 받은 이동 호스트는 Resv 메시지로 응답하게 된다. 이를 통해 reserved 자원이 형성된다.

6. PATH 메시지를 받은 이웃의 FA들은 RSVP 에이전트로부터 이전에 받은 Flowspec을 기초로 하여 prepareResv 메시지로 응답한다. 이 과정을 통해서 prepared 형태의 자원이 형성된다.

**3.2 핸드오프 절차 및 패킷 포워딩 서비스**

그림 4에 이러한 핸드오프시의 경로 변경 과정을 설명하고 있다. 이동 호스트가 새로운 지역으로 이동을 하고 나면, 이전의 COA(Care Of Address)를 터널세션으로 하는 자원 예약 - 그림 4에서 핸드오프 이전의 reserved 자원 - 은 더 이상 의미가 없게 된다. 따라서 송신자가 이동 호스트의 이동으로 인해서 바깥쪽 IP 목적지 주소를 새로운COA로 바꾸어 데이터를 전송하면 데이터 통신 경로를 따라 예약된 자원을 이용할 수 가 없게 된다. 그러므로 새로운 COA를 플로우 아이디로 하는 새로운 예약이 이루어지기 전까지는 이전에 예약된 경로와 자원

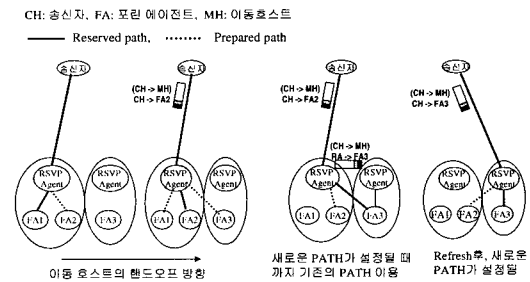


그림 4 핸드오프시의 경로변경

을 이용하게 함으로써 QoS 지원이 끊기지 않도록 해야 한다. 이를 지원하기 위해서 송신자는 새로운 경로의 예약이 이루어지기 전까지 이전의 COA를 목적지 주소로 하는 지연 바인딩을 통해서 패킷을 전송한다. 이 패킷은 데이터 통신 경로를 따라 전달되다가 RSVP 에이전트에게 도착하게 된다. 이 때 RSVP 에이전트는 이미 requestQoS 메시지에 의해 해당 패킷의 수신자인 이동 호스트가 핸드오프한 사실을 알고 있기 때문에 미리 예약을 해 둔 새로운 셀의 FA를 목적지로 하는 패킷으로 바꾸어 예약된 자원을 이용하며 전달되게 한다. 이 과정을 패킷 포워딩(packet forwarding)이라 부른다 - 그림 4의 세 번째 경우. 이러한 패킷 포워딩 서비스는 우선 이전의 COA를 목적지로 하는 바깥쪽 IP 헤더를 디캡슐레이션(decapsulation) 한 후, 새로운 COA를 목적지로 하는 바깥쪽 IP 헤더로 인캡슐레이션(encapsulation) 함으로써 이루어지게 된다. 그리고 원래의 패킷을 이전의 COA에게도 보내줌으로써 바인딩 경고(binding warning), 바인딩 갱신[10]을 유발시켜서 송신자로 하여금 새로운 COA를 알 수 있도록 한다. 새로운 COA를 알게 된 송신자가 새로운 자원 예약을 완료하기까지 패킷 포워딩 서비스를 받게 된다. 이로써 RSVP-RA 프로토콜은 호스트의 이동시에도 끊김이 없는(seamless) QoS를 제공할 수 있다.

4. 성능분석

본 장에서는 제안하는 RSVP-RA 프로토콜의 성능을 시뮬레이션을 통하여 분석해 본다. 시뮬레이션은 이동성에 의한 패킷 지연(delay), 예약되는 대역폭의 양, 라우터에서의 자원예약 상태의 유지 비용의 측면에서 시뮬레이션을 수행하였다. 제안하는 프로토콜을 [2]의 MRSVP, [5]의 SimpleQoS와 비교하여 성능을 평가하였다.

그림 5는 시뮬레이션에 사용된 네트워크 구조를 보여주고 있다. 한 로컬 네트워크가 36개의 셀로 구성되어 있고, 이동 호스트는 셀 사이에서 네 방향으로 랜덤하게 이동할 수 있다. 실험에서 두 노드간에 링크의 대역폭은 무제한으로 가정하였고, 송신자는 46의 위치에 고정되어 있는 것으로 가정하였다. 시뮬레이션에서의 시간은 상대적인 시간 개념을 사용하여 시뮬레이션 타임(simulation time)과 지연시간(delay time) 단위(unit)를 사용한다. 한 시뮬레이션 타임 단위 동안 성공적으로 송신자가 목적지를 향해 데이터를 전송하게 된다. 이 때 걸리는 시간은 각 링크 지연의 합으로 계산되는데, 각 링크 지연은 로컬 네트워크에서는 1에서 5사이의 값을 가지게 되

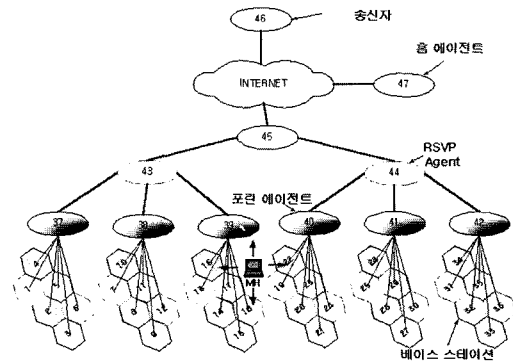
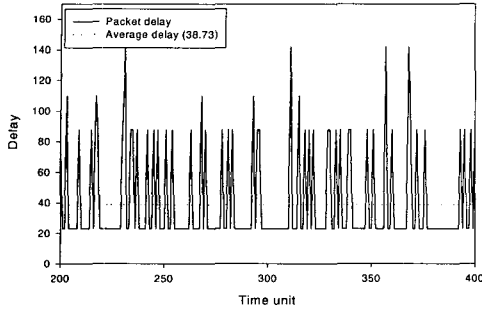


그림 5 시뮬레이션에 사용된 네트워크 구조

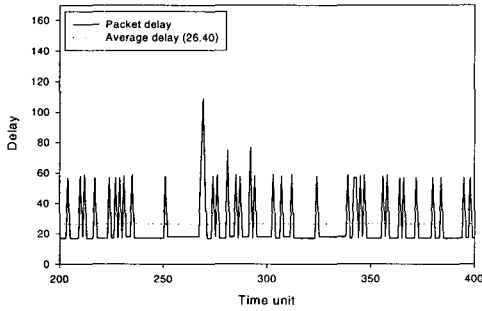
고, 인터넷을 지나는 동안에는 7에서 10사이의 지연시간(delay time)을 가지고 있다. 각 플로우가 가지는 서비스 클래스는 3가지로 정의하였다. 각 서비스 클래스 1Mbps, 2Mbps, 4Mbps를 요구하는 플로우가 전체 트래픽의 20%, 50%, 30%가 되도록 구성하였다. RSVP Agent의 위치는 그림 5의 두 군데 (43, 44)에서 동작하도록 하였다. 통신 중인 플로우의 수는 1~30까지 다양한 환경에서 실험하였고, 각 플로우의 라이프 타임(life time)은 지수분포를 따르고 평균 750 시뮬레이션 타임(simulation time)을 가진다. RSVP 메시지의 갱신을 위해서는 5 시뮬레이션 타임을 적용하였으며, 전체 시뮬레이션은 1000 시뮬레이션 타임동안 수행하였다.

1) 패킷 지연(packet delay) 시간 비교

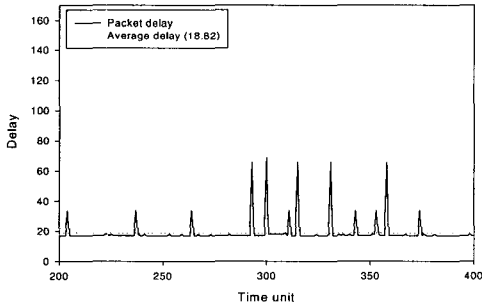
그림 6은 핸드오프 확률이 20%일 때의 패킷 지연 시간에 대해 보여주고 있다. 그림에서 볼 수 있듯이 핸드오프가 없을 때에는 패킷 지연은 20 정도의 값을 가지게 된다. 그림 6(a)에서 볼 수 있듯이, Simple-QoS는 23에서 최대 140까지의 지연 변화를 보여 주고 있다. 그림 6(b)에서처럼 MRSVP는 18에서 110정도의 지연시간을 보이는 반면, 그림 6(c)에서 볼 수 있듯이 RSVP-RA는 18에서 60정도의 값으로 제한될 뿐 아니라 변화의 폭도 그리 심하지 않다. 이는 RSVP 에이전트의 패킷 포워딩(packet forwarding) 서비스로 인해서 이러한 결과를 얻을 수 있었다. 200시뮬레이션 시간 동안 RSVP-RA 프로토콜의 평균 지연은 18.82이었으며, MRSVP와 Simple-QoS는 각각 26.40과 38.73의 값을 가졌다. 그리고 핸드오프 확률이 80%일 때도 비슷한 결과를 얻을 수 있었고, 그 결과는 그림 7에 나와 있다. SimpleQoS는 평균 지연시간 108.37의 높은 지연시간을 나타냈고, MRSVP의 평균 지연시간은 66.94. RSVP- RA는



(a) Simple-QoS

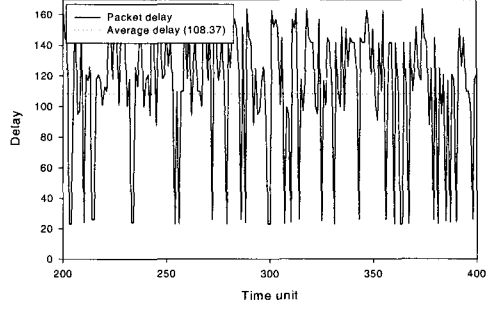


(b) MRSVP

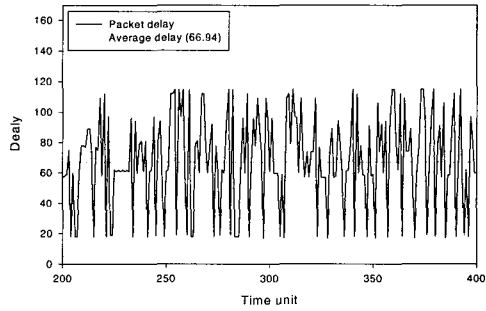


(c) RSVP-RA

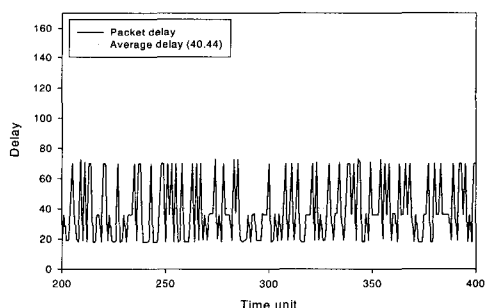
그림 6 패킷 지연 시간 비교 - handoff rate 20%



(a) Simple-QoS



(b) MRSVP



(c) RSVP-RA

그림 7 패킷 지연 시간 비교 - handoff rate 80%

40.44의 적은 패킷 지연을 보였다. 한편 각각의 핸드오프 확률별로 각 프로토콜의 평균 지연시간 변화는 그림 8에 보였다. 그림 8에서 전체적으로 핸드오프의 확률이 높아질수록 각 프로토콜이 갖는 패킷 지연이 증가함을 볼 수 있는데, RSVP-RA는 그 증가율이 낮음을 확인할 수 있다.

2) 예약되는 대역폭의 오버헤드의 비교

그림 9는 MSPEC의 수가 5, 10, 15, 20일 때의 예약되는 대역폭의 오버헤드를 보여주고 있다. 그림에서 예약되는 대역폭의 형태는 RSVP-RA에서는 prepared 형태의 자원이고, MRSVP에서는 passive 자원의 예약이

다. Simple-QoS의 경우 미리 예약되는 자원이 없기 때문에 비교에서 제외하였다. 전체적으로 플로우 수가 증가할수록 예약되는 대역폭 오버헤드가 증가하는 경향을 볼 수 있다. 이는 플로우의 수만큼 예약하는 대역폭이 늘어나므로 당연한 결과라고 할 수 있다. 그림에서 볼 수 있듯이 MSPEC의 수가 5일 때에는 MRSVP에서 불필요하게 예약되는 대역폭이 적어 더 좋은 성능을 보이고 있다. 이는 MRSVP에서 passive 예약은 MSPEC의 수에 비례해서 증가하게 되어 있고, RSVP-RA에서 prepared 예약은 주변 셀의 수에 비례하기 때문이다. 따라서 MSPEC의 수가 적을수록 MRSVP에서 passive

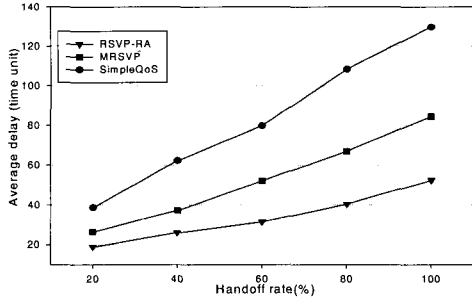


그림 8 평균 패킷 지연 시간 비교

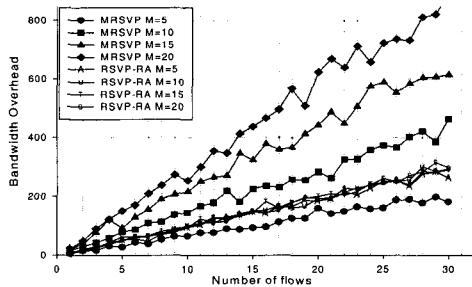


그림 9 대역폭 오버헤드

예약의 양은 줄게 된다. 그러나 MSPEC의 수가 증가할 수록 엄청난 양의 대역폭을 예약하게 됨을 볼 수 있다. 하지만 RSVP-RA는 주변의 셀의 숫자가 고정되어 있으므로 prepared로 예약되는 양은 고정된다. 따라서 이동 호스트의 이동성의 특성에 상관없이, 예약되는 대역폭의 양이 일정하였다.

3) RSVP 메시지 수의 비교

그림 10과 그림 11은 예약된 자원의 상태를 유지하고 관리하기 위해서 주고 받아야하는 RSVP 메시지의 수를 보여주고 있다. 그림 10에서 볼 수 있듯이 핸드오프 확률이 20%일 때 RSVP-RA 프로토콜은 Simple-QoS와 비슷한 성능을 보였으며, MRSVP에 비해 월등하게 적은 양의 메시지를 주고받는 것을 확인할 수 있다(그림 10, 그림 11). 그리고 플로우 수의 증가에 따라서도 제어 메시지 수의 증가율이 낮다는 것을 볼 수 있다. Simple-QoS[14]는 현재 사용되는 하나의 경로로만 자원을 예약하기 때문에 본 논문에서 제안하는 RSVP-RA에 비해 시그널링 오버헤드는 더 향상된 성능을 보일 것으로 예측하였으나, 이동성이 증가하게 되면 오히려 RSVP-RA보다 오버헤드가 증가함을 보였다(그림 11). RSVP-RA 프로토콜에서는 로컬 네트워크에서 RSVP 에이전트와 일정한 수의 FA들만이 RSVP 메시지를 주고받으면 되므로 핸드오프 확률에 관계없이 메시지의 수가 일정하였다.

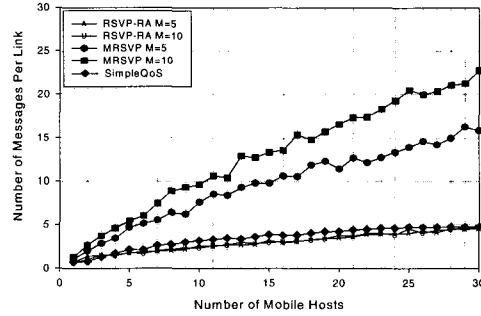


그림 10 각 링크 당 평균 RSVP메시지의 수 handoff rate 20%

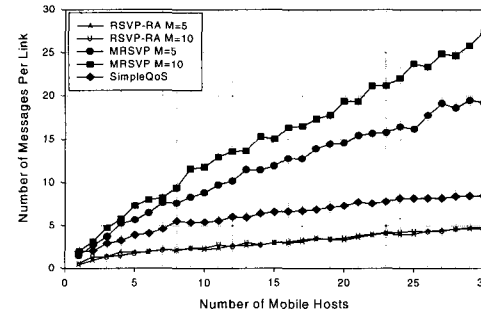


그림 11 각 링크 당 평균 RSVP메시지의 수 handoff rate 80%

5. 결론 및 향후 연구

지금까지 시뮬레이션을 통해 제안된 프로토콜과 Simple-QoS, MRSVP와의 성능을 비교하여 보았다. 시뮬레이션을 수행한 결과, 패킷 지연은 RSVP-RA의 패킷 포워딩(packet forwarding) 서비스에 의하여 개선된 성능을 보임을 확인할 수 있었다. 그리고 대역폭과 시그널링 오버헤드 측면에서도 RSVP-RA가 우수한 성능을 보임을 확인하였다. 이 두 가지의 성능 개선은 로컬 네트워크에서의 자원 예약 관리에 의한 성능 향상으로 정리 될 수 있다. 그러나 대역폭 오버헤드는 이동 호스트의 이동 지역의 특성에 따라서 기존 프로토콜인 MRSVP가, 우수한 성능을 보이기도 하였다. 이동 호스트의 이동 지역이 지극히 제한적인 경우, 이웃의 모든 지역에 자원을 예약해야 하는 RSVP-RA보다 나은 성능을 보임을 확인할 수 있었다. 그러나 이러한 경우를 제외하고는 모든 면에서 제안하는 RSVP-RA 프로토콜이 일정수준의 QoS를 보장하면서 효율적인 성능을 보이고 있음을 확인하였다.



제안된 RSVP-RA 프로토콜은 다음과 같은 장점을 가진다. 첫째는 RSVP 에이전트의 패킷 포워딩 서비스와 송신자의 지연 바인딩을 통해 패킷 지연을 줄였다는 것이다. 두 번째는 송신자와 FA(Foreign Agent)가 아닌 RSVP 에이전트와 FA간에만 prepared 자원이 예약되도록 함으로써 이동성을 지원해 주기 위해 불필요하게 많이 예약되는 대역폭의 양을 줄일 수 있었다. 세 번째는 자원 예약을 유지하고 관리하기 위해 주고받아야 하는 RSVP 메시지의 수를 줄임으로써 네트워크에 주는 프로세싱 오버헤드를 줄였다는 점이다. 마지막으로 네 번째로는 이동성의 특성에 관계되는 MSPEC같은 정보가 필요하지 않다는 점이다.

현재 RSVP-RA 프로토콜이 가지는 오버헤드를 줄일 수 있는 방안으로 분산구조의 RSVP-RA에 대한 연구를 수행 중에 있으며, 차후 All-IP 망에서 seamless handoff 지원을 위해 이동성 관리 프로토콜(mobility management protocol) 과의 연동방안에 대한 연구를 수행할 계획이다.

### 참 고 문 헌

- [1] R. Braden, L. Zhang, Resource ReSerVation Protocol(RSVP) -- Version 1 Message Processing Rules, RFC 2209, September 1997.
- [2] A. K. Talukdar, B. R. Badrinath and A. Acharya, MRSVP: A Reservation protocol for an Integrated Services Packet Networks with Mobile hosts, Tech report TR-337. Rutgers university
- [3] I. Mahadevan and K. M. Sivalingam, An Architecture for QoS guarantees and Routing in Wireless/Mobile Networks, WOWMOM 98.
- [4] I. Mahadevan and K. M. Sivalingam, Architecture and experimental results for quality of service in mobile networks using RSVP and CBQ, Wireless Networks Vol.6, 2000, pp. 221-234.
- [5] A. Terzis, M. Srivastava and L. Zhang, A simple QoS signaling Protocol for Mobile hosts in the Integrated Services Internet, IEEE Infocom'99.
- [6] C. Perkins, IP mobility Support, RFC 2002.
- [7] W. T. Chen and L. C. Huang, RSVP Mobility Support: A Signaling Protocol for Integrated Services Internet with Mobile Hosts, IEEE Infocom'2000.
- [8] S. K. Das, R. Jayaram, N. K. Kakani and S. K. Sen, A Resource Reservation Mechanism for Mobile Nodes in the Internet, IEEE 49th Vehicular Technology Conference(VTC'99).
- [9] S. K. Das, M. Chatterjee and N. K. Kakani, QoS Provisioning in Wireless Multimedia Networks, IEEE WCNC'99.
- [10] C. Perkins, D. B. Johnson, Route Optimization in Mobile IP, Internet Draft, work in progress, Feb 2000.
- [11] A. Terzis, J. Krawczyk, J. Wrocalwski and L. Zang, RSVP Operation Over IP Tunnels, RFC 2746
- [12] R. Braden, D. Clark, and S. Shenker, Integrated Service in the Internet Architecture: an Overview, RFC 1633, June 1994.
- [13] P. Paul, RSVP and Integrated Services in the Internet: A tutorial, IEEE Communications Magazine, May. 1997.
- [14] X. Xiao, L. M. Ni, Internet QoS: A Big Picture, IEEE Network, Mar. 1999.
- [15] C. Metz, RSVP: General-purpose signaling for IP, IEEE Internet Computing, May. 1999.
- [16] L. Zhang, Resource ReSerVation Protocol(RSVP) - Version 1 Functional Specification, RFC 2205.
- [17] D.B. Johnson, C. Perkins, Mobility Support in IPv6, Internet Draft, work in progress, Apr 2000.
- [18] Daniel O. A., Emmanuel Agu, Mobile Extensions to RSVP, in the Proceedings of the Sixth International Conference on Computer Communications and Networks(ICCCN'97)
- [19] Rohit Ghai and Suresh Singh, An Architecture and Communication Protocol for Picocellular Networks, IEEE Personal Communications Magazine Vol.1(3), 1994, pp.36-46.



김민선

1999년 2월 숭실대학교 컴퓨터학부 졸업(학사). 2001년 2월 포항공과대학교 컴퓨터공학과 졸업(석사). 2001년 2월 ~ 현재 한국루슨트테크놀로지스 Bell Labs-Korea 연구원. 관심분야는 무선네트워크에서의 QoS, 이동성관리, MPLS, Mobile IP, 이동통신(IMT-2000).

서영주

정보과학회논문지 : 정보통신  
제 29 권 제 3 호 참조

안성욱

정보과학회논문지 : 정보통신  
제 29 권 제 3 호 참조