

HMIPv6에서 게임 이론을 이용한 MAP 부하 분산 기법

정희원 기범도*, 종신회원 김승욱**

Game Theoretic MAP Load Balancing Scheme in HMIPv6

Bum-do Ki* *Regular Member*, Sung-wook Kim** *Lifelong Member*

요약

HMIPv6는 로컬 네트워크상에 이동성 지원 포인트(Mobility Anchor Point ; MAP)라는 기능을 도입하여 MAP 하부에서의 이동은 MAP가 관리함으로써 지연과 오버헤드를 줄이고 이동 노드의 핸드오프를 효율적으로 관리하는 방법이다. 그러나 이동 노드가 새로운 도메인에 들어올 경우 최상위 MAP에 이동 노드의 등록이 집중되어 성능을 저하시키는 단점이 있다. 본 논문에서는 게임이론 중에서 내쉬 교섭 해(Nash Bargaining Solution ; NBS) 이론을 적용하여 HMIPv6상에서 상위 MAP에 이동 노드의 등록이 집중되지 않고 거리(Distance), 등록된 노드 수, 데이터 전송량 등의 MAP 자원 사용 비율에 따라 여러 MAP로 부하를 분산시키는 알고리즘을 제안하였다. 시뮬레이션을 통해 상위 MAP로의 부하 집중을 방지할 수 있었고, MAP자원의 사용비율에 따라 가중치를 실시간으로 그리고 능동적으로 부여함으로써 도메인내의 MAP를 효율적으로 사용하여 성능을 향상시킬 수 있었다.

Key Words : HMIP, MAP, Load Balancing, Game Theory, Nash Bargaining Solution

ABSTRACT

The Hierarchical Mobile IPv6 (HMIPv6) has been proposed to accommodate frequent mobility of the Mobile Node. HMIPv6 can effectively reduce the signaling overhead and latency. However, it has a problem that the registration of a mobile node concentrates on the furthest MAP(Mobility Anchor Point) when the mobile node enters into a new domain. This paper proposes a new load distribution mechanism by using the concept of Nash Bargaining Solution. The main advantage of the proposed scheme can prevent load concentration from being registered to the specified MAP based on the weight value according to the available resource-ratio of a MAP. With a simulation study, the proposed scheme can improve network performance under widely diverse traffic load intensities.

1. 서론

최근 수년간 무선 단말이 소형화되고 무선 통신 기술이 발전함에 따라서 이런 무선 환경에서 인터넷을 이용하고자 하는 욕구가 증가하고 있다. 그 결과, 무선 네트워크 인프라의 구축과 이를 이용하는 서비스는 점차 증가하고 있고, 이에 따라 IP 기반 이동성 지원기술에 대한 연구가 활발히 진행되고 있다. Snoop TCP, M TCP, Mobile IP 등이 그 예이다. 특히 차세대 네트워크가

All-IP망으로 진화함에 따라 IP 기반의 이동성 관리 프로토콜인 Mobile IP의 중요성은 더욱 커지고 있다.

IP 기반 이동성 제공을 위한 연구는 이동 IPv4를 거쳐 Mobile IPv6에 까지 이르게 되었고, Mobile IPv6는 Mobile IPv4의 여러 단점을 보완하였다. 잦은 핸드오프로 인한 데이터의 손실과 지연, 삼각 라우팅(Triangular Routing), 인입 트래픽 필터링(Ingress Filtering)문제와 주소체계를 128 bit로 확장하여 주소 부족 문제를 해결한 것이 Mobile IPv6에서 보완한 기술이다. 그러나 실시간

* 서강대학교 정보통신대학원, ** 서강대학교 컴퓨터공학과(swkim01@sogang.ac.kr)

논문번호 : KICS2009-12-618, 접수일자 : 2009년 12월 13일, 최종논문접수일자 : 2010년 06월 30일

통신을 위해 보완해야 할 문제점은 여전히 남아 있다. MIPv6의 핸드오프 지연을 줄이기 위해 FMIPv6와 HMIPv6가 IETF MIPSHOP WG에 의해 제안되어 표준화가 완료되었다. FMIPv6는 링크 계층에서 단말의 이동을 미리 감지하여 주소 핸드오프 시간을 줄여주는 방법이고, HMIPv6는 MAP를 이용하여 도메인 안에서의 이동은 MAP에서 관리하게 하여 시그널링 오버헤드(signaling overhead)를 줄이는 것이다. 그리고 이 두 가지를 결합한 F-HMIPv6는 두 방법의 장점만을 결합하여 더 효율적으로 핸드오프를 관리하는 방법이다.

새로운 네트워크에 진입한 이동 노드는 자신의 홈 에이전트 역할을 담당할 MAP를 선택해야 하는데, 현재 IETF에서 제안한 HMIPv6^[1]는 상대적으로 비용이 많이 드는 MAP 도메인간 이동 확률을 줄이기 위하여 거리가 반으로 가장 먼 MAP를 선택하는 거리기반 MAP 선택 알고리즘을 제안하였다. 이 알고리즘은 빈번한 도메인간의 핸드오프를 줄이기 위하여 가장 먼 곳에 위치한 MAP를 선택한다. 그러나 먼 거리에 있는 MAP는 도메인이 커짐에 따라 성능을 저하시키는 병목현상을 유발할 수 있다. 이는 MAP가 더 많은 트래픽 터널링을 해야 할 뿐 아니라 바인딩 업데이트(Binding Update; BU)로 인한 트래픽도 증가하기 때문이다^[2].

본 논문은 분산 다계층 HMIPv6 네트워크 모형에서 MAP 거리, MAP에 등록된 노드 수, MAP 별 통신량을 정량화 하고, 이를 기반으로 투자비용 대비 최대 잉여 값을 갖는 MAP를 선택할 수 있는 방법을 제안한다. 잉여 값 산출 방식은 내쉬 교섭 해(Nash Bargaining Solution) 이론을 적용하여 구하고, 각 이동 노드는 잉여 산출식을 기반으로 계층구조상에 분산되어 있는 각 MAP들에 대해 생명주기(Life Time)와 선호도(Preference) 값이 유효할 때 전체 잉여가 가장 큰 MAP를 선택한다. 이를 통해 자원이 양호한 MAP를 선택하여 특정 MAP로 부하가 집중되는 것을 방지할 수 있고 이동 노드를 MAP에 적절히 분배하여 성능을 향상시킬 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 네트워크 계층의 이동성 관리 기술인 Mobile IP와 게임 이론에 관련된 기존 연구를 살펴보고, 3장에서는 분석적 모델을 위한 분산 다계층 HMIPv6 기본 모형을 제시하고, 잉여 값 산출을 위해 MAP 자원을 정량화 하여 NBS를 적용하는 방안을 제시한다. 4장에서는 성능평가 모델을 이용해 성능 평가를 수행하여 결과를 비교 및 분석하고, 5장에서 결론을 맺는다.

II. 관련 연구

2.1 HMIPv6

이동 IPv6는 홈 에이전트(Home Agent; HA)로의 BU, 대응 노드(Correspondent Node; CN)로의 복귀 라우팅 가능(Return Routability), 그리고 바인딩 캐시 엔트리(Binding Cache Entry)의 빈번한 업데이트를 수반하는데, 이런 이동 IPv6의 기능들은 어떤 측면에서 추가적인 시그널링 오버헤드로 비춰진다. HMIPv6는 주로 이러한 오버헤드를 다루기 위해 설계된 프로토콜이다^[3]. 이런 문제점들을 개선하기 위해서 HMIPv6에서는 로컬 네트워크 상에 MAP를 두어 이러한 지연과 오버헤드를 줄이고 효율적으로 이동 노드(Mobile Node; MN)의 핸드오프를 수행할 수 있도록 하고 있다.

계층적 이동성 관리기법은 도메인 내(Intra-domain)의 이동성 관리인 미시적 이동성 관리와 도메인 간(Inter-domain)의 이동성 관리인 거시적 이동성 관리로 구분된다. 이 기법의 중심 요소는 MAP인데, 이는 보통 자신의 네트워크 도메인을 이루고 있는 접속 라우터(Access Router; AR)들의 상위에 위치하는 라우터 혹은 라우터들의 집합이며, 자신과 자신의 도메인을 방문하고 있는 이동 노드들 간의 바인딩을 관리한다. 이동 노드가 AR들 사이를 이동할 때마다 MIPv6에서는 위에 기술한 것과 같은 시그널링 오버헤드가 큰 반면, HMIPv6에서는 이동 노드가 도메인 내를 이동할 때에는 MAP에 단 한번의 BU를 수행하면 되기 때문에 HA와 CN사이의 신호 양이 줄어들게 된다. HMIPv6의 경우 HA와 CN에게 이동 노드의 LCoA가 아닌 RCoA를 등록하기 때문에 CN이나 HA는 이동 노드의 정확한 위치를 파악할 수 없게 된다. 따라서 HMIPv6는 경로 최적화뿐만 아니라 이동 노드의 위치 기밀성(Location Privacy)을 보장할 수 있다^[4]. 그러나 통신하는 모든 패킷이 MAP를 거치기 때문에 MAP 장애 및 병목현상과 같은 상황에 취약한 단점을 가지고 있다.

2.2 HMIPv6에서의 MAP 선택 기법

HMIPv6에서 이동 노드가 무선 접속 라우터를 옮겨 갈 때 새 무선 접속 라우터가 속하는 여러 MAP 도메인 중 어느 것을 선택할 것인가 하는 문제는 통신 성능에 큰 영향을 주는 요소이다. 기존의 MAP 선택 기법에는 거리기반의 MAP 선택 기법, 이동 노드의 속도를 기반으로 한 MAP 선택 기법, 망 토폴로지(Topology)를 기반으로 한 MAP 선택 기법, 망 토폴로지(Topology)를 기반으로 한 MAP 선택 기법, 적응적 MAP 선택 기법 등이 있다.

거리기반의 MAP 선택 기법에서 이동 노드는 잦은 MAP간 핸드오프로 인한 지연 및 패킷 손실을 피하기 위

해 최대한 큰 영역을 관리하는 MAP를 선택하도록 하고 있다. MAP의 크기를 판단하기 위해 무선 접속 라우터가 이동 단말에게 보내주는 라우터 광고 메시지 내의 MAP 옵션 중에 거리(Distance) 필드 값을 이용한다. 여기에서 거리 필드의 값은 단말과 MAP 사이의 홉수를 나타낸다. 그러나 이런 단순한 거리기반의 MAP 선택 기법을 사용하면 모든 단말이 항상 거리가 가장 먼 MAP를 선택하기 때문에 동일한 선택이 발생하여 하나의 MAP로 모든 부하가 집중되는 부작용을 낳을 수 있다. 또한 단말의 이동 속도가 작을 경우에도 큰 MAP 영역을 경유하여 통신하기 때문에 먼 거리를 통해 패킷 전송이 이루어지는 비효율이 발생할 수 있다⁵⁾.

이동 노드의 속도에 따른 MAP 선택기법은 이동 노드의 속도가 빠를수록 시그널 발생이 많아진다는 점에 착안하여 제안된 기법이지만 프로토콜을 수정해야 하는 단점이 있다. 또한 속도가 빠른 이동 노드가 먼 거리의 MAP를 선택하더라도 패킷 전송률이 높은 경우에는 선택된 MAP의 부하가 증가하는 단점이 있다.

망 토폴로지(Topology)를 기반으로 한 MAP 선택 기법은 이동 노드와 가장 가까운 MAP를 선택하도록 하고 있다. 계층 구조의 토폴로지뿐 아니라 다양한 토폴로지에서도 동작이 가능하고 노드나 링크의 장애에 대처가 용이한 장점이 있는 반면에 MAP 옵션 및 라우터 광고 메시지를 수정해서 사용해야 하며 무선 접속 라우터에서 망의 모든 MAP 정보를 관리해야 하는 부담이 있다.

적응적 MAP 선택 기법은 세션 도착율과 이동 노드의 이동 특성을 이용한 SMR (Session-to-Mobility Ratio)이라는 개념을 도입하고 각 비용함수를 통하여 SMR의 값에 따라 MAP를 선택하는 기법이다. 지속적인 MAP 선택과정을 수행하고 최소비용으로 부하 분산이 가능하지만 이동 노드와 라우터 사이의 추가적인 메시지 전달이 필요하거나 프로토콜의 수정이 필요하다.

2.3 게임이론과 무선통신망

게임상황은 사람들 간의 상호작용(interaction)이 존재하는 상황을 말한다. 이러한 상황에서는 각 사람은 자신의 입장에서만 생각해서는 최적의 결과를 얻을 수 없다. 각 사람은 자신의 행동뿐만 아니라 상대방의 행동에 의해서도 영향을 받는다. 게임이론은 이러한 상호작용이 존재하는 상황에서 어떤 결과가 나타날 지에 관해 체계적으로 예측을 하기 위해 개발된 이론이라고 볼 수 있다. 게임이론은 사회현상을 게임상황으로 모형화하고, 그 모형화된 게임에서 경기자의 전략적 행동을 이해하는 틀을 제공하고 있다⁶⁾.

게임의 구성요소에는 게임에 참가하는 경제주체인 참

가자(Player, 예: 기업, 개인 등)와 참가자들이 자신의 이윤극대화를 위하여 선택할 수 있는 대안인 전략(Strategy), 게임의 결과로 각 참가자가 얻게 되는 보수(Payoff)가 있는데 이를 게임의 3요소라고 한다. 이 외의 요소로는 전략이 계속 유지되는 상태인 균형, 그리고 순서와 선택할 수 있는 대안을 제시하는 게임규칙 등을 들 수 있다.

게임은 크게 협조적 게임과 비협조적 게임으로 구분할 수 있는데, 협조적 게임(cooperative game)은 게임을 하기 이전에 게임에 참여하는 선수들이 완전히 구속력 있는 협약(full and binding agreement)을 맺고 하는 게임이다. 내쉬 교섭 해와 샤플리 벨류(Sharply Value)가 여기에 속한다. 비협조적 게임(non cooperative game)은 서로가 사전에 어떤 구속력 있는 협약이 없이 선수들이 주어진 전략집합 하에서 자신의 효용을 극대화하기 위해 합리적으로 자신의 최선의 전략을 찾으려는 형태의 게임이며, 내쉬 균형(Nash Equilibrium)이 여기에 속한다.

게임이론은 무선통신망 상에서 자치 에이전트(autonomous agent)들의 상호 작용에 대한 연구를 하는 것이다. 분산된 프로토콜로 동작하는 노드는 자신이 모든 것을 결정해야 한다. 이런 결정은 프로토콜의 규칙이나 알고리즘에 의해서 제한되기도 하지만 궁극적으로 각 노드는 파라미터를 설정하고 동작 모드를 변경하는데 약간의 여지는 있다. 이런 노드들이 전력, 패킷 전달, 백오프(backoff) 시간에 대한 의사 결정을 하는 자치 에이전트들이다.

무선 통신망에 게임 이론을 적용하는 데에는 많은 함정이 잠재해 있다. 그중의 하나가 협조적 게임 이론과 비협조적 게임 이론의 혼동이다. 경기자는 누구이고, 어떤 행동이 가능하고, 경기자의 목표가 무엇이고, 주어진 게임 상황에 균형은 존재하는가, 그리고 유일한 해가 존재하는가, 경기자가 전략을 바꾸면 능동적으로 처리하는가? 이런 의문점이 정확히 정의 되어야 한다. 게임 이론은 여러 문제들에 접근하는데 있어서 유용한 틀이다. 요점은 이것이 신중하게 그리고 문제에 적합하게 사용되어야 한다는 것이다.

게임이론의 적용분야는 CDMA 시스템의 업링크 전력제어(uplink power control)나 네트워크의 라우팅 문제, 그리고 개별 노드가 다른 노드 대신에 서비스 수행을 예상할 수 있느냐 하는 신용관리(trust management) 등을 예로 들 수 있다⁷⁾.

III. 제안하는 MAP 부하 분산 기법

이동 노드가 새로운 MAP 도메인에 들어오게 되면

AR로부터 라우터 광고(Router Advertisement; RA) 메시지를 받고 RA에 포함된 하나 이상의 MAP 옵션들 중에 거리가 가장 큰 MAP를 선택하여 자신의 로컬 HA기능을 담당하도록 한다. 그러므로 상위 MAP로 부하가 집중되는 문제를 야기한다. 본 논문에서는 거리 외에 이동 노드가 MAP를 선택하기 위한 MAP 자원의 가용량을 계산하기 위해 노드 수와 통신량을 추가하여 MAP의 가치를 평가함으로써 거리 기반의 MAP 선택에서 문제가 되는 특정 MAP로의 부하 집중을 방지하고자 한다.

MAP의 가치 평가는 MAP에서 먼저 노드 수와 통신량의 가중치를 구하고, 이후 이동 노드에서는 계산된 가중치와 거리를 이용하여 최종적으로 MAP를 선택하도록 한다.

거리가 4인 MAP 도메인이 있고 각 MAP의 최대 관리 가능한 노드 수는 100, 통신 대역폭은 100Mbps라고 임의로 설정한다. 표 1은 MAP들의 사용자원을 나타낸다.

표 1. MAP별 사용자원

	Distance	노드 수	통신량
MAP1	4	50	40
MAP2	3	32	40
MAP3	2	26	30
MAP4	1	66	60

3.1 MAP의 동작

MAP에서의 동작은 간단하다. 라우터 광고 메시지 전송시 MAP 자신의 가용 자원을 계산하여 확장된 MAP 옵션의 'Weight' 필드에 실어서 보낸다. 계산식은 아래와 같다.

$$Nn = (1 - n/N) \tag{1}$$

$$Db = (1 - d/B) \tag{2}$$

Nn은 노드 가용량, N은 최대 노드수, 그리고 n은 등록된 노드 수를 나타내며, Db는 가용 대역폭, B는 최대 대역폭, 그리고 d는 데이터 통신량을 나타낸다.

Db는 노드 수와 통신량이 반드시 비례하는 것이 아니기 때문에 비용 산출에 사용하는 것이다. 위의 식 (1)과 (2)에서는 가용한 자원을 측정하기 위해 1에서 사용 자원의 양을 뺀 것이다. 각 요소의 비용은 동일하다고 가정한다.

위의 식으로부터 노드 수와 대역폭에 대한 가용자원을 구할 수 있고 이 두 값의 평균을 이동 노드에게 전달한다. 계산식은 아래와 같다.

$$(Nn + Db) * 100 / 2 \tag{3}$$

확장된 MAP 옵션 형식에서 'Weight' 필드의 크기는 7-bit로 제한되어 있어서 부동 소수점 형태의 값이나 정수 128을 초과하는 값은 저장할 수 없다. 따라서 위의 식 (3)에서 100을 곱하여 노드 가용량(Nn)과 가용 대역폭(Db)의 값을 구하고, 이 두 값의 평균을 구하기 위해 2로 나눈다. 이 평균은 0~100 사이의 값이므로 'Weight' 필드에 적용하는데 무리가 없다.

표 1의 MAP1을 예로 들면 계산 방식은 아래와 같다.

$$\text{노드 가용량} = (1 - 50/100) = 0.5$$

$$\text{가용 대역폭} = (1 - 40/100) = 0.6$$

이 값을 이용하여 이동 노드에게 전달할 값은 아래와 같이 계산한다.

$$(0.5 + 0.6) * 100 / 2 = 55$$

이 결과를 MAP 옵션의 'Weight' 필드에 실어서 보낸다. 확장된 MAP 옵션 형식은 아래 그림 1과 같다.

0		1		2		3	
0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
Type	Length	Dist	Pref.	R	Weight		
Valid Lifetime							
Global IP Address for MAP							

그림 1. 확장된 MAP 옵션 형식

3.2 이동노드의 동작

각 MAP로부터 계산 결과를 수신한 이동 노드의 MAP 수신 정보는 표 2와 같다.

거리에 대한 투자비용은 아래와 같이 임의로 정의 한다.

$$Dh = d/D * 100 / D \tag{4}$$

Dh는 투자비용, d는 거리, 그리고 D는 최대거리를 나타낸다. 식 (4)에서 100을 곱하는 이유는 식 (3)과의 형평성을 위한 것이다. 식 (3)에서는 7-bit의 'Weight' 필드에 적용 가능하도록 하기 위해 100을 곱하여 정수로 만들었기 때문이다. 그리고 최대거리(D)로 나누는 것은 MAP의 깊이(Depth)에 따라 투자비용을 유동적으로 적용하

표 2. MAP별 평균 가용 자원

	거리	평균 가용 자원
MAP1	4	55
MAP2	3	69
MAP3	2	72
MAP4	1	37

기 위한 것이고, 또한 거리가MAP를 선택하는데 너무 많은 비중을 차지하지 않도록 하기 위함이다. 식 (3)과의 형평성을 위해 일률적으로 2로 나누게 되면 상위 MAP가 상대적으로 큰 값을 갖게 되므로 부하의 효과적인 분산을 기대하기 어렵다.

이제 위의 계산식 (4)를 이용하여 각 MAP의 총 투자비용을 계산하는 과정에 대해서 살펴보도록 한다.

최대 거리가 4 일 경우 MAP1의 가용자원 계산은 아래와 같다.

$$\text{거리 투자비용} = 4/4 * 100 / 4 = 25$$

$$\text{표 2의 MAP1 평균 가용자원} = 55$$

그러므로 MAP1의 총 투자비용은 거리 투자비용과 평균 가용자원을 합한 80 이 된다.

이와 같은 방법으로 각 MAP별 투자비용을 계산하면 아래 표 3과 같다. 이 결과를 바탕으로 내쉬 교섭 해를 구한다. 각각의 MAP가 경기가 되며, 여기에서는 2인 경기라고 가정한다.

제안하는 기법을 설명하기에 앞서 내쉬 교섭 해를 구하는 방법에 대해 알아보도록 한다.

모든 교섭상황은 두 가지 공통점이 있다. 첫째는 그룹 합리성(Group Rationality)이다. 선수 각각이 혼자서 이루어 낼 수 있는 보상보다 공동으로 이루어 낼 수 있는 보상이 같거나 커야 한다. 둘째는 제로섬(Zero-sum) 게임이 아니라는 것이다. 즉, 발생할 수 있는 잉여를 어떻게 나눌 것인가가 중요하다. 서로 더 많이 가져가려고 하면 다른 사람이 적게 가져가므로 제로섬 게임처럼 보이지만, 협정이 이루어 지지 못하면 아무도 이 잉여를 가져가지 못하게 된다는 점에서 제로섬 게임이 아니다. 따라서

표 3. MAP별 투자비용

	투자비용
MAP1	80
MAP2	88
MAP3	85
MAP4	43

위협(threat)이 전략으로써 중요해진다.

내쉬의 비협조적 게임의 교섭모형은 균형이 정해지지 않아 교섭상황을 설명하는데 적합하지 않다. 물론 순차적 게임을 설정하여 균형을 잡아낼 수 있으나 모형에 많은 가정과 수정을 필요로 하게 된다. 따라서 유일한 균형이 존재하는 협조적 교섭모형이 본 논문에서 제안하는 특정 MAP를 선택하는데 적합한 방법이라 할 수 있다.

V 라는 잉여를 A와 B가 나누려고 할 경우, 둘이 협정을 맺지 못하면, A는 a, B는 b를 가져간다. 이 a와 b를 BATNAs(Best Alternative to a Negotiated Agreement; Harvard Negotiation Project; Reference Point)라고 부른다. 많은 경우에 있어 BATNA는 0 이다. 이를 위해서는 다음 식이 성립하여야 한다.

$$a+b < V \text{ 또는 } V > 0 \rightarrow (V-a-b > 0)$$

A는 잉여의 h, B는 잉여의 k 의 비율로 잉여를 분할한다고 가정한다. 그러므로 $h + k = 1$ 이 된다. A가 최종적으로 가져가는 액수를 x, B가 최종적으로 가져가는 액수를 y 라고 하면, 잉여분할공식은 아래와 같다.

$$x = a + h(v - a - b) \rightarrow x - a = h(v - a - b) \tag{5}$$

$$y = b + k(v - a - b) \rightarrow y - b = k(v - a - b) \tag{6}$$

그러므로 식 (5)와 (6)을 정리하면 아래와 같다.

$$\frac{y - b}{x - a} = \frac{k}{h} \tag{7}$$

$$y = b + \frac{k}{h}(x - a) = \left(b - \frac{ak}{h} \right) + \frac{k}{h}x$$

위의 공식에 대한 제약은 아래와 같다.

$$x + y = v \tag{8}$$

식 (7)과 식 (8)을 모두 만족하는 점이 내쉬의 공조적 게임의 해이다. 내쉬의 공조적 교섭해 즉, 제약식과 분할공식을 모두 만족하는 내쉬의 해는 그림 2와 같다. x 축과 y축을 잇는 직선이 $h + k = 1$ 을 만족하는 효용가능곡선이고, 협상이 결렬되었을 경우 서로의 이익이 0이 되는 협상 결렬점 P이다.

그림 2에서 잉여분할공식의 기울기 k/h 는 각각의 상대적 교섭력을 의미한다. 만일, 교섭결과를 알고 있으면

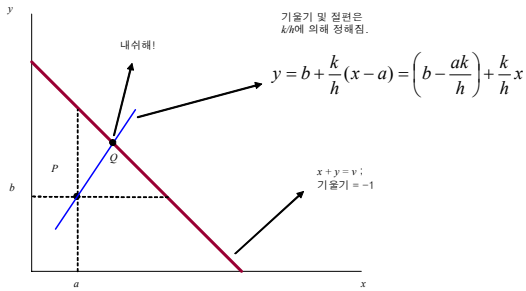


그림 2. 내쉬의 공조적 교섭 해

위의 공식을 이용하여 교섭력을 계산해 낼 수 있다. 즉, 아래 식 (9)에서 x, y, a, b 를 모두 알고 있으면 k/h 를 알 수 있다. 또한 $k + h = 1$ 이므로 k 와 h 값을 계산할 수 있다.

$$\frac{y-b}{x-a} = \frac{k}{h} \quad (9)$$

즉, 교섭력은 BATNA와 최종 분배액의 함수이다. 이를 상대적 교섭력이라 하며, 정치경제학에서는 부의 분배가 집단의 정치력에 의하여 결정된다는 가정 하에서 정치력(political power)라고도 부른다.

지금부터는 위에서 설명한 기법을 기반으로 표3을 이용하여 내쉬 교섭 해를 구하는 방법에 대해 알아본다. 각 MAP의 조합 중에서 MAP1과 MAP2에 대한 것을 예로 들도록 한다.

내쉬 교섭 해를 구하기 위해서 두 경기자 MAP1과 MAP2가 협조하여 얻을 수 있는 잉여는 각각 100씩 200이라고 가정한다. 표3의 초기 투자비용을 기반으로 잉여를 분할한다고 하면 MAP1 : MAP2는 48 : 52(168 : 80 = 100 : x)의 비율로 배분할 수 있다. MAP1이 최종적으로 가져가는 액수를 x , MAP2가 최종적으로 가져가는 액수를 y 라고 할 때, 이를 내쉬 교섭 해를 구하는 식에 적용하면 아래와 같이 계산할 수 있다.

$$\text{총 비용} = 80+88+100+10 = 368 \quad (10)$$

$$\text{잉여 배분 비율} = 48 : 52 \quad (11)$$

식 (10)의 총비용과식 (11)의 잉여 배분비율을 식 (9)에 적용하여 x 와 y 값을 구하는 방법은 다음과 같다.

$$\begin{aligned} 52(x-80) &= 48(y-88) \\ x &= 0.9y - 1.2 \end{aligned} \quad (12)$$

최종적으로 가져가는 액수 x 와 y 의 합이 총 비용이므로 x 는 총 비용에서 y 를 뺀 값과 같으므로 $x = 368 - y$ 가 된다. 이를 식 (12)에 대입하면

$$0.9y - 1.2 = 368 - y \text{ 가 된다. 그러므로}$$

$$y = 194 \text{ 이고,}$$

$$x = 368 - 194 = 174 \text{ 가 된다.}$$

산출된 x 와 y 값을 이용하여 MAP1과 MAP2의 순이익 a 와 b 는 표3의 각각의 투자비용을 제외한 값이 되며, 이는 아래와 같이 계산할 수 있다.

$$a = 174 - 80 = 94$$

$$b = 194 - 88 = 106$$

위와 같은 방법으로 나머지 MAP의 조합별 순이익을 구하면 아래 표4와 같다. 내쉬 교섭 해를 적용하여 계산한 결과, 최대 잉여 값을 갖는 MAP는 MAP2이므로 이동 노드는 MAP2를 선택한다.

평가 대상이 모든 MAP 이므로 합리성과 타당성을 제공할 수 있도록 가장 큰 잉여가 아닌 잉여 값의 합계 증가 큰 값을 선택하도록 한다.

MAP를 선택하는 알고리즘은 그림 3과 같다. 이동 노드의 RCoA나 LCoA가 라우터로부터 수신한 RCoA나 LCoA와 틀리면 이동 노드가 다른 네트워크로 이동했다는 것을 의미하므로 MAP를 선택해야 한다. 이 때 수신

```

if ( ( RCoA of MN != New RCoA ) or
    ( LCoA of MN != New LCoA ) ) {
  for ( i=0; i<Max distance; i++ ) {
    if ( pref. value == 0 or valid lifetime == 0 )
      Remove the MAP;
  }
  각 MAP의 조합별 잉여 계산(NBS 이용);
  if ( MAP with largest value == 1 ) {
    Select the MAP;
  }
  else if ( MAP with largest value > 1 ) {
    Select the furthest MAP of the MAPs;
  }
}
    
```

그림 3. MAP 선택 알고리즘

표 4. MAP 조합별 잉여 계산 결과

	MAP1	MAP2	MAP3	MAP4
1	94	106	-	-
2	94	-	106	-
3	128	-	-	72
4	-	99	101	-
5	-	133	-	67
6	-	-	130	70
합계	316	338	337	209

된 MAP 옵션 중에서 선호도(preference)나 생명주기(lifetime)의 값이 0 이라면 유효하지 않은 MAP 이므로 이 MAP는 선택 대상에서 제외하고 나머지 MAP들의 조합에 따라 내쉬 교집 해를 구한다. 계산 결과 중 가장 큰 값을 갖는 MAP가 하나이면 그 MAP를 선택하고 둘 이상이면 그 중에서 가장 거리가 먼 MAP를 선택한다.

IV. 성능 평가

4.1 시뮬레이션 환경

시뮬레이션은 리눅스에서 C를 이용하여 간단히 구현하였다. 시뮬레이션을 위한 하드웨어 및 소프트웨어 플랫폼은 아래 표 5와 같다.

그림 4는 시뮬레이션에서 사용될 네트워크 구조이다. 이는 크게 세 부분으로 구분할 수 있다. A영역은 최상위 MAP인 게이트웨이와 HA/CN을 연결하는 인터넷 구간, B영역은 계층화된 분산 MAP 구간, C영역은 이동 노드와 직접 연결을 담당하는 무선 인터페이스 구간이다. A 및 B 구간의 대역폭은 100Mbps 이다. C 구간의 대역폭은 10Mbps이고 이동 노드가 생성하는 트래픽은 최대 10Mbps이며, 이 트래픽 양은 임의로 생성된다고 가정한다.

실험 시나리오는 다음과 같다.

- 시뮬레이션에 사용되는 이동 노드의 수는 100개로 한다.
- 각 이동 노드는 일정한 간격(1 초)을 두고 AR2로 이동한다.

표 5. 시뮬레이션 환경

Lap Top	CPU	Intel Core Duo 2.0 Ghz
	Memory	3 Gb
OS	Windows XP	Prof. Edition v5.1
	VMware Server	v1.0.5
	Ubuntu	v8.10
Simulation	Linux C	

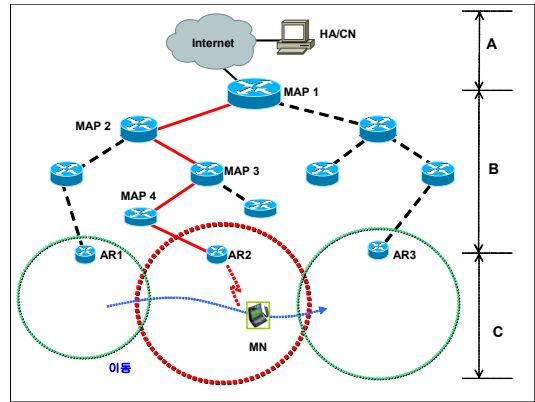


그림 4. 시뮬레이션 네트워크 구조

- 이동 노드는 AR2의 가장 긴 구간을 통과한다.
- MAP가 수용 가능한 노드 수를 초과할 경우 다른 MAP를 선택한다.
- 통신량은 무작위로 생성하여 이동 노드별로 다르게 적용한다.
- 이동 노드의 분산 정도와 트래픽 강도를 측정하고 비교 및 분석한다.

위와 같은 방법으로 시뮬레이션을 수행하여 [1]에서 제안한 거리 기반 MAP 선택기법과 [8]에서 제안한 MAP 선택기법, 그리고 본 논문에서 제안된 NBS 기반 MAP 선택기법의 결과를 비교 및 분석한다. 참고로, [8]에서 제안한 MAP 선택 계산 방법은 그림 5와 같이 임의로 'Metric 기반 MAP 선택 기법'이라 칭한다.

거리 가중치 : $D_h = \text{hop}$ (수동 조정)
 노드 가용량 : $N_n = 1 - n/N$
 가용 대역폭 : $D_b = 1 - D/B$

$$\text{Value} = D_h + N_n + D_b$$

N : 노드 임계값 n : 노드 수
 B : 대역폭 D : data 사용량

그림 5. Metric 기반 가치 계산 공식

4.2 성능 평가

이동 노드가 AR2 영역을 통과하도록 하여 MAP의 수용 가능한 자원 비율이 증감함에 따라 부하가 적절히 분산되는가를 측정하도록 한다.

이동 노드의 수는 100개로 하고 이동 노드의 최대 속도는 5m/s로 하며 속도는 임의로 지정한다. 특정 AR 이 영향을 미치는 범위는 100M이고 범위의 가장 긴 거리를

이동 노드가 지나간다고 가정한다. 각 이동 노드는 최대 10 Mbps의 트래픽을 발생한다. 동일한 조건 아래서 각 기법별 실행 결과는 아래 표 7과 같다. 이는 마지막 이동 노드가 AR 범위 안에 도착했을 때의 MAP 상태를 나타낸 것이다.

거리 기반 기법은 상위 MAP에 이동 노드가 집중된 것을 볼 수 있다. 단지 일정 시간 후에 AR2 범위를 벗어난 이동 노드가 사용했던 MAP의 자원이 복원됨에 따라 계속해서 상위 MAP를 선택했기 때문에 하위 MAP에는 이동 노드가 등록되지 않은 것을 볼 수 있다. Metric 기반 기법은 거리 기반 기법보다 양호한 분포를 보이기는 하지만 역시 상위 MAP로 집중된 모습을 보이고 있다. NBS 기반 기법은 다른 기법에 비해 더 양호한 분포도를 보이고 있다. 여기에서는 각 MAP에 이동 노드가 분포된 정도와 트래픽 강도(traffic intensity)에 대해서만 비교하도록 한다.

아래 그림 6은 이동 노드에 임의의 속도를 부여하고 이동할 경우 각 MAP에 노드가 분산된 정도를 나타낸다.

표 7. 이동성을 부여한 시뮬레이션 실행 결과

MAP ID	거리 기반 기법		Metric 기반 기법		NBS 기반 기법	
	등록된 Node 수	Data크기	등록된 Node 수	Data크기	등록된 Node 수	Data크기
0	27	14868	16	8143	17	8737
1	21	9522	19	9129	13	7180
2	0	0	6	3563	10	5019
3	0	0	7	3555	8	3454



그림 6. MAP 별 노드 수 변경 추이

X축은 시간, Y 축은 MAP에 등록된 노드 수를 나타낸다.

그림 6에서 거리 기반 기법과 metric 기반 기법에서는 상위 MAP에 이동 노드가 집중되었고, 최대수용 가능한 노드 수를 초과한 경우 차상위 MAP에 이동 노드가 위치한 것을 볼 수 있다. 트래픽이 적은 경우는 성능 향상을 기대할 수 있지만, 많은 트래픽이 집중될 경우에는 특정 MAP로 부하가 집중되기 때문에 지연은 급격히 증가하고 장애의 빌미를 제공하므로 전체 네트워크의 성능 및 안정성에 바람직하다고 할 수 없다. NBS 기반 기법에서는 MAP 가용자원에 따라 전체 MAP에 노드가 적절히 분배되어 이런 문제를 해결할 수 있다.

아래 그림 7은 이동 노드의 MAP 선택 추이를 나타낸다. X축은 시간을 나타내고, Y 축은 노드 수를 나타낸다. 거리 기반 기법 (a)는 상위 MAP 두 개에 모든 노드가 집중되었다. 최상위 MAP가 한계 노드 수에 도달한 후 차상위 MAP에 이동 노드가 할당되는 모습을 보이고 있다. 이후 두 MAP에 번갈아 등록되는 현상을 보이고 있는데, 이는 시간이 지남에 따라 AR2의 범위를 벗어난 노드가 발생하고, 이로 인해 MAP의 가용 자원이 환원됨으로써 최상위 MAP에 수용 가능한 노드 수가 변경되기 때문이다. 하위 MAP 두 개는 전혀 사용되지 않고 있다. Metric 기반 기법 (b)에서는 어느 정도 분산된 형태를 보이고 있지만 상위 MAP로의 쏠림 현상은 여전하다. 그러나 NBS 기반 기법 (c)에서는 시간이 지남에 따라 전체 MAP에 이동 노드가 골고루 분포된 모습을 보이고 있다. 즉, MAP의 할당 및 환원된 가용 자원을 정확하게 계산해서 MAP를 선택하기 때문에 전체 MAP에 골고루 분포가 되었다.

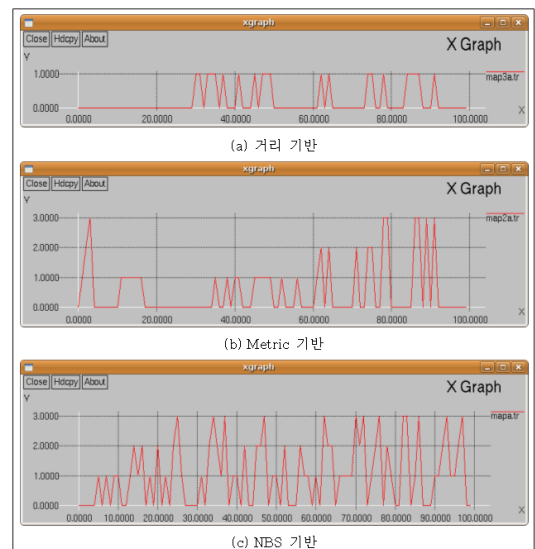


그림 7. MAP 선택 추이

트래픽 강도는 L_a/R 로 표현하는데, 이는 큐잉 지연의 정도를 측정하는데 매우 중요한 요소이다. 패킷이 큐에 도착하는 평균율을 a 라고하며, 단위는 패킷/초 이다. R 은 링크의 전송률이고 L 은 패킷의 크기(bit)이다. $L_a/R > 1$ 이면 비트가 큐에 도착하는 평균율이 전송되는 비율을 초과하는 것이다. 이럴 경우, 큐는 끝없이 증가하고 큐잉 지연은 무한대에 도달한다. 그러므로 트래픽 공학의 주요 규칙 중 하나는 "트래픽 강도가 1보다 크지 않게 설계하라"는 것이다⁹⁾.

아래 그림 8은 MAP의 트래픽 강도 변화 추이를 나타낸다. X축은 시간을 나타내고, Y 축은 트래픽 강도를 나타낸다. 거리 기반 기법은 상위 2개의 MAP의 트래픽 강도가 모두 1을 넘어서고 있다. 트래픽 강도가 1을 넘는다는 것은 위에서 언급한 바와 같이 큐잉 지연의 급격한 증가를 의미하고, 이는 곧 패킷 손실을 의미한다. 그러므로 종단간 지연이 클 수밖에 없다. Metric 기반 기법도 트래픽 강도가 1에 가깝다. 거리 기반 기법보다는 적은 수치를 보이지만 이 방법 역시 많은 큐잉 지연을 초래한다. NBS 기반 기법은 다른 두 기법에 비해서 양호한 트래픽 강도를 보이고 각각의 MAP에 고루 분산되었으며, 트래픽 강도 또한 낮은 수치를 나타내고 있다. 이는 곧 MAP의 부하가 다른 기법에 비해 고르게 분포된 것을 나타낸다.

결과적으로, 본 논문에서 제안한 NBS 기반의 MAP 선택 기법은 MAP의 사용 가능한 자원을 정확히 계산하고 이를 바탕으로 부하를 적절히 분배함으로써 MAP의 부하 분산은 물론이고 전체 네트워크의 안정성을 도모할

수 있는 한 가지 방법이 될 수 있다. 또한 저비용 고효율의 비용 효율적인 네트워크를 설계하고 구축하는데 유용한 방법을 제공할 수 있을 것이다.

V. 결 론

본 논문에서는 특정 라우터에 이동 단말이 집중되는 것을 방지하기 위해 각 라우터의 사용 가능한 자원(거리, 데이터 통신량, 관리되는 이동 단말의 수)을 정량화하였으며, 이 자료를 기반으로 내쉬 교섭 해를 구하여 라우터를 선택하도록 함으로써 특정 라우터로 집중되는 부하를 방지하고자 하였다.

이동 노드가 AR2를 횡단하도록 한 경우에 거리 기반 기법이나 metric 기반 기법에서는 상위 MAP에 등록이 집중되었고 하위 MAP는 전혀 사용되지 않거나 그 사용이 미미하다. 그러나 NBS 기반 기법은 전체 MAP에 고루 분포하고 있다. 위와 같은 이동 노드의 분포에 기초하여 각 기법별로 트래픽 강도를 측정 한 결과, 거리 기반 기법과 metric 기반 기법은 상위 MAP의 트래픽 강도가 1을 초과하거나 1에 근접한 값을 나타내는 반면, NBS 기반 기법은 전체 MAP가 0.4~0.8 사이의 트래픽 강도를 보임으로써 전체 MAP에 부하가 분산된 것을 볼 수 있다. 결과적으로 NBS 기반 MAP 선택 기법은 MAP 자원을 평가하는데 있어 객관성, 정밀성, 타당성을 제공함으로써 최대 성능을 발휘할 수 있는 MAP를 선택할 수 있도록 했으며, 시뮬레이션을 통해 모든 MAP에 부하가 적절히 분배되었음을 확인하였다.

참 고 문 헌

- [1] H. Soliman, C. Castelluccia, K. El Malki and L. Bellier, *Hierarchical Mobile IPv6 Mobility Management (HMIPv6), RFC4140*, Aug. 2005.
- [2] 정원식, 이수경, "적응적 MAP 선택을 통한 HMIPv6 네트워크의 성능 향상 알고리즘", *한국통신학회논문지 '06-11 Vol.31 No.11B*, pp.945-946, 2006.
- [3] Rajeev S. Koodli and Charles E. Perkins, *MOBILEINTER-NETWORKING WITH IPv6 Concepts, Principles, and Practices*, Wiley-Interscience, 2007.
- [4] 성기혁, *HMIPv6 상에서의 MAP 부하 분산 알고리즘*, 한양대학교 대학원, pp.13-14, 2007.
- [5] 노명화, *계층적 이동 IPv6 환경에서 이동단말의 특성을 고려한 비용 효율적인 MAP 선택 기법*,

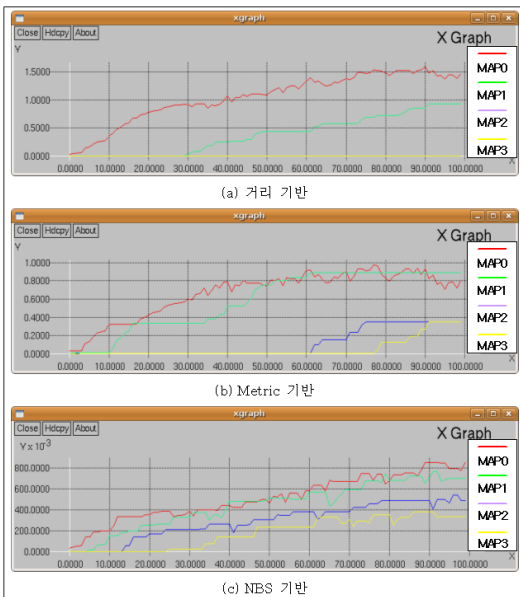


그림 8. 트래픽 강도 변화 추이

강원대학교 대학원, pp.6-7, 2007.

- [6] 한동근, *게임이론-전략적 의사결정의 이론과 응용*, 경문사, 1997.
- [7] Allen B. MacKenzie and Luiz A. DaSilva, *Game Theory for Wireless Engineers*, Morgan & Claypool Publishers, 2006.
- [8] 이승중, *HMIPv6 환경에서의 효율적인 MAP 선택 방안*, 건국대학교 대학원, pp.29-32, 2005.
- [9] James F. Kurose and Keith W. Ross, *Computer Networking : A Top-Down Approach, 4th Edition*, Addison-Wesley, 2008.

김 승 욱 (Sungwook Kim)

중신회원



2003년 12월 Syracuse University, computer science 박사
 2006년 2월 중앙대학교 컴퓨터 공학부 전임강사
 2009년 서강대학교 컴퓨터공학과 조교수
 2010년~현재 서강대학교 컴퓨터공학과 부교수

<관심분야> QoS, 실시간 제어처리, 셀룰러 네트워크 자원관리

기 범 도 (Bum-do Ki)

정회원



1991년 2월 상지대학교 자원학과 이학사
 2009년 8월 서강대학교 정보통신대학원 공학석사
 <관심분야> Mobile IP, IP기반 이동성 관리, Load Balancing, QoS