

다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치 관리 및 메시지 전달 기법*

최성진⁰ 강인성[†] 백맹순[†] 손진곤[‡] 황종선[†]

고려대학교 컴퓨터학과 분산시스템 연구실, 한국방송대학 컴퓨터과학과[‡]
{lotieye⁰, iskang[†], msbak[†], hwang[†]}@disys.korea.ac.kr, jgshon@mail.knou.ac.kr[‡]

Location Management & Message Delivery Scheme for Mobile Agent in Multi-region Mobile Agent Computing Environment

SungJin Choi⁰, InSung Kang, MaengSoon Baik, ChongSun Hwang
Distributed System Lab. Dept. of Computer Science & Engineering, Korea University
JinGon Shon
Dept. of Computer Science, Korea National Open University

요 약

다중 지역 이동 에이전트 컴퓨팅 환경을 기반하여 이동 에이전트 시스템을 개발하는데 있어서, 이동 에이전트를 관리하고 이동 에이전트들간의 메시지 전달을 보장하기 위한 위치 관리 및 메시지 전달 기법은 중요한 고려사항이다. 그러나 기존의 관련연구들은 여러 지역으로 구성된 다중 지역 이동 에이전트 컴퓨팅 환경에 적용하기에는 위치 갱신 비용 및 메시지 전달 비용이 상대적으로 증가한다는 단점과 추적문제(Following Problem)를 발생시킨다는 한계를 가지고 있다. 이에 본 논문에서는 기존의 홈 프락시 기법과 블랙보드 기법을 변형·혼합(Hybrid)하는 새로운 이동 에이전트 위치 관리 및 메시지 전달 기법인 HB 프로토콜을 제안하고자 한다. HB 프로토콜은 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 위치 변경시 발생하는 위치 갱신 비용과 이동 에이전트들간의 메시지 전달 비용을 효과적으로 줄이며, 추적문제를 낮은 통신비용으로 해결한다.

1. 서 론

이동 에이전트(Mobile agent)는 이질적인 네트워크 환경에서 노드들을 이동하면서 사용자를 대신하여 자율적으로 주어진 일을 처리하는 프로그램이다[6]. 이동 에이전트 수행중 이동 에이전트의 제어와 이동 에이전트들간의 메시지 전달을 보장하는 위치 관리 및 메시지 전달 기법은 이동 에이전트 시스템(Mobile agent system)을 개발하는데 있어서 중요한 고려사항 중의 하나이다[1, 3].

이동 에이전트 환경에서 이동 에이전트의 위치 관리 및 메시지 전달 기법에 관련된 기존 관련 연구들은 크게 홈 프락시(Home-Proxy), 추적 프락시(Follower-Proxy), 브로드캐스트(Broadcast), 블랙보드(Blackboard) 그리고 이메일(E-Mail) 기법으로 분류된다[1, 3, 4]. 이러한 관련 연구들 중 블랙보드와 이메일 기법은 주로 메시지 전달에 사용되지만, 나머지 기법들은 위치 관리와 메시지 전달에 동시에 사용된다.

여러 지역으로 구성된 다중 지역 이동 에이전트 컴퓨팅 환경에서 위 기법들을 적용하면, 위치 갱신 비용 및 메시지 전달 비용이 증가하게 되며 또한, 이동 에이전트에게 해당 메시지를 보내지 못하고 이동 에이전트가 방금 떠난 노드들만 계속해서 탐색하는 문제, 추적문제(Following Problem)가 발생한다. 따라서 효율적으로 이동 에이전트의 위치를 추적하고 메시지를 전달하기 위한 기법이 필요하다.

본 논문에서는 홈 프락시 기법과 블랙보드 기법을 변형·혼합(Hybrid)하는 새로운 이동 에이전트 위치 관리 및 메시지 전달 기법인 HB 프로토콜을 제안한다. HB 프로토콜은 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트 위치관리 및 메시지 전달시 발생하는 위치 갱신 비용 및 메시지 전달 비용을 줄이며, 추적문제를 낮은 통신비용으로 해결한다.

2. 관련 연구

기존 이동 에이전트의 메시지 전달 기법들 중 홈 프락시 기법[1,4]은 이동 에이전트의 위치 정보를 이동 에이전트가 생성된 노드, 홈 노드나 특정 서버에 저장한 다음 이 정보를 이용하여 이동 에이전트를 찾거나 메시지를 전달 하는 기법이다. 추적 프락시 기법[1, 4]은 위치 정보를 방문한 노드들에 저장한

다음 경로 프락시(Path proxies)를 따라가서 이동 에이전트를 찾거나 메시지를 경로 프락시를 따라 계속해서 포워딩(Forwarding)하는 기법이다. 브로드캐스트 기법[1,4]은 정해진 지역이나 이동계획(Itinerary)의 모든 노드들에게 메시지를 보내어 위치를 찾거나 메시지를 전달하는 기법이다. 블랙보드 기법[1, 5]은 각 이동 에이전트 시스템에 블랙보드를 두어 메시지를 교환하는 기법이다. 이메일 기법[1]은 이메일 서버와 유사한 방법으로 매일 서버에 메시지를 남겨 놓으면 이동 에이전트가 접속하여 메시지를 얻는 기법이다. 그러나 이러한 기법들은 여러 지역(region)으로 다중 지역 이동 에이전트 컴퓨팅 환경에서 적용하였을 경우 많은 문제들이 발생한다. 즉, 이동 에이전트의 현재 노드와 홈 노드와의 거리에 따라 위치 갱신 비용 및 메시지 전달 비용이 증가하는 문제점과 이동 에이전트에게 해당 메시지를 전달하지 못하고 이동 에이전트가 방문한 노드들만 계속해서 탐색하는 문제가 발생한다. Antonella Di Stefanof[2]는 SPC 프로토콜을 통하여 이동 에이전트의 위치 관리 문제점을 해결하려고 하였으나, 경로 프락시(Path Proxies)를 사용함으로써 인해 위치 관리 및 메시지 전달하는데 많은 비용이 발생하게 된다.

3. 이동 에이전트 컴퓨팅 환경

본 논문에서 가정하는 이동 에이전트 시스템은 이동 에이전트 a_i , 실행장소 p_i 로 구성된다. 이동 에이전트 a_i 는 각 노드의 실행 결과에 따라 다음 노드를 동적으로 선택하여 이동하거나, 이동계획(itinerary)에 따라 노드와 노드사이를 옮겨다니며 작업을 수행한다. 이때 노드에서는 에이전트가 수행할 수 있는 실행장소 p_i 를 제공한다[6]. 실행장소는 에이전트에게 특정 서비스를 제공한다. 한 노드에는 여러 실행장소가 있을 수 있다.

이동 에이전트 시스템들 중에서 같은 권한을 가진 이동 에이전트 시스템들의 집합을 지역 R_i 이라고 한다[6]. 본 논문에서는 이동 에이전트가 여러 지역으로 구성된 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동하면서 작업을 수행하는 경로가 정한다.

각 지역에는 지역의 권한을 책임지는 지역 서버(Region Server)가 존재한다[6]. 그리고 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 전역 네이밍 서비스를 제공하는 룩업 서버(Lookup server)가 존재한다. 그림 1은 본 논문에서 가정하는 다중 지역 이동 에이전트 컴퓨팅 환경을 나타내고 있다.

* 본 연구는 한국과학재단 목적기초연구(R01-2002-000-00235-0)지원으로 수행되었음.

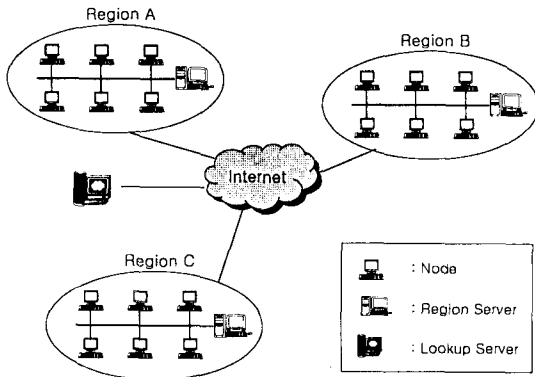


그림 1. 다중 지역 이동 에이전트 컴퓨팅 환경

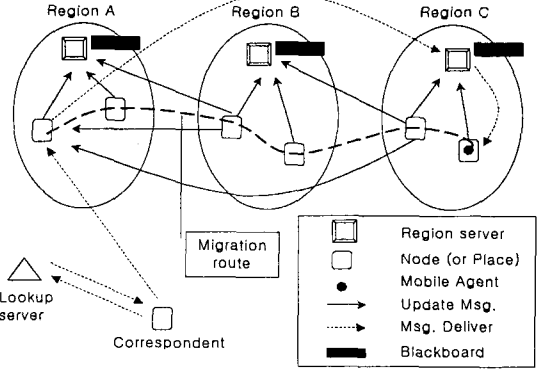


그림 2. HB 프로토콜의 전반적인 개요

4. 제안 기법

4.1 홈 프락시 기법과 블랙 보드 기법의 변형 · 혼합

여러 지역으로 구성된 다중 이동 에이전트 컴퓨팅 환경에서 홈 프락시 기법을 사용하여 이동 에이전트의 위치관리를 수행하면 위치 갱신 비용이 증가한다. 왜냐하면 다중 이동 에이전트 컴퓨팅 환경에서 이동 에이전트가 이동할 수 있는 이동 지역이 지역적으로 한정되어 있지 않기 때문이다[2]. 또한 이동 에이전트 탐색과정 및 메시지 전달과정에서 위치 갱신 정보가 최신 것이 아니면 해당 이동 에이전트를 찾아 계속 탐색하는 경우가 발생하여 해당 이동 에이전트를 찾아 메시지를 전달하는 것이 실패하게 된다.

이에 반해, 블랙보드 기법은 이동 에이전트가 메시지를 전달하기 위해서는 직접 해당 노드로 이동해야 하는데, 이동 에이전트가 해당 노드로 이동하는 패턴이 일정하지 않고 임의로 이루어진다. 따라서, 여러 지역으로 구성된 다중 이동 에이전트 컴퓨팅 환경에서 블랙 보드 기법을 적용하면 이동 에이전트가 메시지를 전달받기 위해서 여러 지역을 돌아다녀야 하므로 메시지를 전달 비용이 증가한다. 그리고 간단한 메시지의 경우, 직접 이동 에이전트가 해당 노드로 이동하여 처리하는 시점에서 불필요한 통신 부하가 발생한다.

따라서, 본 논문에서는 각 기법의 단점을 보완하여 이동 에이전트의 위치 관리 및 메시지 전달 비용을 줄이기 위해 두 기법을 변형 · 혼합하는 HB 프로토콜을 제안하고자 한다. 홈 프락시 기법의 문제점 즉, 이동 에이전트의 위치를 찾지 못하고 계속해서 이동 에이전트가 다니다간 곳을 탐색하는 문제점을 해결하기 위해 블랙보드 기법을 적용한다. 즉, 각 지역서버에 있는 블랙보드에 위치 추적 메시지나 전달 메시지를 남겨 놓은 다음 이동 에이전트가 이 메시지를 블랙보드로부터 읽어 추적 문제를 해결하고 메시지 전달을 보장받게 된다. 또한, 블랙보드 기법의 문제점 즉 메시지를 적절한 시기에 전달하지 못하는 문제점을 해결하기 위해 홈 프락시 기법을 적용하고자 한다. 즉, 이동 에이전트의 위치를 주기적으로 갱신할 때 메시지를 전달하는 방법으로 문제를 해결한다.

4.2 HB 프로토콜

HB 프로토콜은 크게 이주(migration)과정과 메시지 전달(Message delivery) 과정으로 구분된다. 이주과정은 이동 에이전트가 이동할 때 홈 노드나 지역 서버에게 자신의 위치를 알리는 등록과정을 말한다. 메시지 전달과정은 이동 에이전트를 찾은 다음 해당 메시지를 이동 에이전트에게 전달하는 과정을 말한다. 그림 2는 이주과정과 메시지 전달 과정을 설명한 HB 프로토콜의 전체적인 개요이다. 모든 노드들이 블랙보드를 갖지 않고 각 지역을 대표하는 지역서버만이 블랙보드를 가지게 된다. 지역 서버는 해당 지역 내에 들어오는 이동 에이전트에 대한 위치 정보를 유지하고 블랙보드를 이용하여 이동 에이전트의 추적 문제를 해결하며 이동 에이전트들에게 메시지를 전달하는 역할을 한다.

4.2.1 이주 과정 알고리즘

이주과정 알고리즘은 크게 같은 지역 내에서의 이주(Intra-region migration)와 다른 지역으로의 이주(Inter-region migration)로 구분된다.

같은 지역 내로 이동 에이전트가 이주할 때 이동 에이전트는 각 지역서버에게 위치갱신 메시지를 보낸다. 기존의 홈 프락시 방법에서는 이동 에이전트가 이주시에는 무조건 홈 노드에게 위치 갱신 메시지를 보내게 된다. 지역적 위치를 고려하지 않고 무조건 홈 노드에게 위치갱신 메시지를 보내면 위치갱신 비용은 높아지게 된다. 즉, 만약 홈노드와 이동 에이전트가 서로 다른 지역에 있다면 홈노드와 이동 에이전트의 거리를 고려할 때 직접 홈노드로 위치정보 메시지를 보내는 것은 같은 지역에 있는 지역서버에게 메시지를 보내는 것보다 상대적으로 비용이 높다. 예를 들어, LAN의 메시지 지연시간과 인터넷의 지연시간은 상당한 차이가 있다[7]. 또한 이동 에이전트가 이주할 때마다 지역 서버에게 위치갱신 메시지를 보낼 때 지역서버의 블랙보드에 있는 메시지를 체크할 수가 있어 블랙보드 기법에서 부족했던 메시지 전달의 주기성을 보완한다.

다른 지역으로 이동 에이전트가 이주할 때 이동 에이전트는 자기가 있는 현재 지역서버와 전에 있었던 지역 서버 그리고 홈 노드에게 위치갱신 메시지를 보내게 된다. 홈 노드에게 메시지를 보내는 것은 홈 노드가 현재 에이전트가 있는 지역을 알 수 있게 한다. 이전 지역서버에게 갱신 메시지를 보내는 것은 이동 에이전트의 현재 지역 위치가 홈 노드에게 알려지기 전에 보냈던 메시지의 전달을 보장하기 위해서 이다.

```

CloneNo = 0;
Child = false;
if ai : Li -> Li+1 then
    Compare(RS of Li, RS of Li+1);
    if SRS(Same Region Server) then
        if(isCloned(ai)) then
            Update.CloneNo = GetCloneNo(ai);
        fi;
        if(isChild(ai)) then
            Update.Child = true;
            SendMsg(Update, HN of Pai);
        fi;
        SendMsg(Update, RS of Li);
    fi;
    if DRS(Different Region Server) then
        if(isCloned(ai)) then
            Update.CloneNo = GetCloneNo(ai);
        fi;
        if(isChild(ai)) then
            Update.Child = true;
            SendMsg(Update, HN of Pai);
        fi;
        SendMsg(Update, RS of Li);
        SendMsg(Update, HN of ai);
        SendMsg(Update, RS of Li+1);
    fi;
fi;
    
```

그림 3. HB 프로토콜의 이주과정 알고리즘

이동 주체가 호스트인 이동 호스트 환경(Mobile IP, PCS)과는 달리 이동 에이전트 환경의 이동주체는 소프트웨어 프로그램

램이다[2]. 즉, 이동 에이전트는 사용자의 요구에 따라 복제(clone)될 수 있고 또한 자식 이동 에이전트를 생성할 수 있다. HB 프로토콜은 이동 에이전트의 홈 노드나 지역 서버에 복제된 이동 에이전트 관련 정보와 부모-자식 이동 에이전트의 관련 정보를 유지한다. 만약 이동 에이전트가 중간 연산 과정에서 자식 이동 에이전트를 생성한다면 자식 이동 에이전트의 위치는 부모 이동 에이전트의 홈 노드에도 기록되어야 한다. 왜냐하면 부모 이동 에이전트와 자식 이동 에이전트의 위치가 서로 다를 때 두 이동 에이전트에게 메시지를 보내야 하기 때문이다. 또한 복제된 이동 에이전트들은 홈노드나 지역 서버에서 그룹으로 묶어서 위치를 관리해야 한다. 복제된 이동 에이전트에게 메시지를 보낼 때에는 그 그룹에 속한 모든 이동 에이전트들에게 메시지를 보내게 된다.

그림 3은 같은 지역내에서의 이주와 다른 지역으로의 이주를 포함한 전체 이주과정 알고리즘이다.

4.2.2 메시지 전달과정 알고리즘

그림 4는 메시지 전달과정에 대한 알고리즘이다. 이 알고리즘을 통한 메시지 전달과정의 절차는 다음과 같이 크게 다섯 단계로 구분된다.

- 1) 먼저 Lookup 서버를 통하여 이동 에이전트의 홈노드를 알아낸다.
- 2) 홈노드에 접속하여 이동 에이전트가 있는 현재 지역 서버를 알아낸다.
- 3) 현재 지역서버에 탐색 메시지를 보내고 지역서버의 블랙보드에 탐색 메시지를 저장한다.
- 4) 지역서버에 있는 이동 에이전트가 위치한 노드에 메시지를 보낸다.
- 5) 만약 이동 에이전트 위치 정보가 틀려 이동 에이전트에게 메시지를 전달하지 못한 경우 다음 이동 에이전트 위치갱신 메시지가 올 때까지 기다리다가 해당 메시지를 보낸다.

```
//LookupServer를 통하여 Home Node의 주소를 알아내는 과정
HNAddr = CN.SendMsg(LookUp, a, LookupSrv);
CN.SendMsg(Msg, a, HNAddr);

//Home node(HN)이 상대자로부터 메시지를 받은 경우
RSAddr = FindAddr(a);
if (isCloned(a) or isChild(a)) then
    GroupAgents = FindGroupAgents(a);
    GroupAddr = FindGroupAddr(a);
    HN.SendGroupMsg(Msg, GroupAgents, GroupAddr);
else
    HN.SendMsg(Msg, a, RSAddr);
fi;

//Resion Server(RS)가 HN으로부터 메시지를 받은 경우
RNAddr = RS.FindAddr(a);
BlackBoard.in(Msg, a);
if (RS.SendMsg(Msg, ai, RNAddr) = ACK) then
    BlackBoard.out(Msg, a);
else
    WaitUntilNextRegistration();
    NewMsg=BlackBoard.out(Msg, a);
    SendMsg(NewMsg, a, RNAddr);
fi;
```

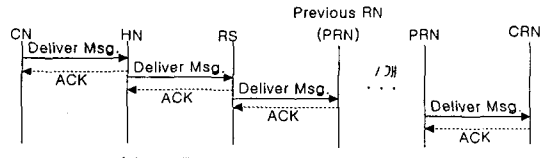
그림 4. HB 프로토콜의 메시지 전달과정 알고리즘

5. 평가 및 토론

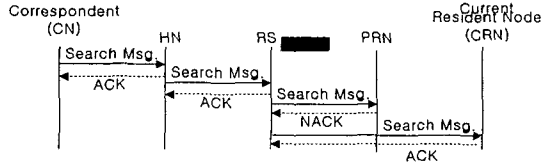
SPC 프로토콜[2]과 HB 프로토콜의 위치 갱신 메시지 복잡도 비용을 비교하면, 같은 지역내의 이주가 m 개이고 다른 지역으로의 이주가 n 개일 때 SPC 프로토콜은 $2m+4n$ 개이며 HB 프로토콜은 $m+3n$ 개이다. 따라서 HB 프로토콜은 $m+n$ 개만큼의 메시지 전달 비용을 줄일 수 있다.

이동 에이전트 메시지 전달과정이 실패했을 때 추적 프락시 방법을 사용하는 SPC 프로토콜과 블랙보드를 사용하는 HB 프로토콜에 대해서 각각의 메시지 비용에 대해서 살펴보면, 추적 프락시를 사용하는 프로토콜은 지역 서버에 저장되어 있는 잘못된 이동 에이전트 위치에서부터 이동 에이전트가 실제로 현재 있는 위치까지 이동 에이전트가 방문한 노드의 개수만큼 계속 반복하여 메시지를 보내야 하지만, 블랙보드 기법을 사용하는 프로토콜은 지역 서버의 블랙보드에 메시지를 보낸 다음 이동 에이전트가 다음번 위치갱신 메시지를 등록할 때까지 기다린 후에 메시지를 현재 이동 에이전트가 있는 노드에게 전달하게 된다. 그림 5에서 볼 수 있듯이 1만큼 메시지 전달 비용이 차이가 나게 된다. 표 1은 SPC 프로토콜과 HB 프로토콜의 메시지 복잡도 비용을 정리한 것이다.

또한 추적 프락시 기법을 사용하는 프로토콜은 이동 에이전트가 방문한 모든 노드에 위치관련 정보를 저장해야 하고 메시지가 왔을 때 포워딩을 해줘야 하므로 계속해서 포워딩 데몬을 유지해야 하는 오버헤드가 있다.



(1) 추적 프락시 방법을 사용하는 SPC 프로토콜



(2) 블랙보드를 사용하는 HB 프로토콜

그림 5. SPC프로토콜과 HB프로토콜의 메시지 전달 비용 비교

구분	SPC 프로토콜	HB 프로토콜
위치갱신 메시지 비용	$2m+4n$	$m+3n$
메시지 전달 비용(메시지 전달과정이 성공한 경우)	3	3
메시지 전달 비용(메시지 전달과정이 실패했을 경우)	$4+l$	4

표 2. SPC 프로토콜과 HB 프로토콜의 메시지 비용 비교

6. 결론 및 향후 연구과제

본 논문은 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 위치 관리 및 메시지 전달 기법에서 발생하는 높은 위치갱신 및 메시지 전달 비용과 추적문제를 해결하기 위해 새로운 HB 프로토콜을 제안했다. 제안 기법은 여러 지역으로 구성된 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 이주시 발생하는 이동 에이전트 위치관리 비용과 이동 에이전트들간의 메시지 전달 비용을 줄이며, 추적문제를 낮은 통신 비용으로 해결한다.

앞으로 제안된 HB 프로토콜을 현재 진행중인 "ODDUGI" 프로젝트에 구현하고, HB 프로토콜에 현재 연구 중인 보안 기법을 결합하여 안전한 위치 관리 및 메시지 전달 프로토콜을 구현하고자 한다.

참고문헌

- [1] Dwight Deugol, "Mobile Agent Messaging Models", In Proc. 5th International Symposium on Autonomous Decentralized Systems, pp. 278 -286, 2001.
- [2] Antonella Di Stefano and Corrado Santoro, "Locating Mobile Agents in a Wide Distributed Environment", IEEE Transactions on Parallel and Distributed Systems, Vol. 13, No. 8, 2002.
- [3] J. Baumann, "A Comparison of Mechanisms for Locating Mobile Agents", IBM Research Report 3333, 1999.
- [4] Pawel T. Wojciechowski, "Algorithms for Location-Independent Communication between Mobile Agents" Technical Report DSC-2001/13, Departement Systemes de Communication, EPFL, 2001.
- [5] G. Cabri, L. Leonardi, F. Zambonelli, "Mobile-Agent Coordination Models for Internet Applications", IEEE Computer, Vol. 33, No. 2, Feb. 2000.
- [6] Object Management Group, "Mobile agent system interoperability facilities specification", OMG TC Document orbos/97-10-05, 1997.
- [7] G. Coulouris, J. Dollimore and T. Kindberg, "Distributed Systems Concepts and Design", Addison-Wesley, third edition, 2001.