(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl.
G06F 15/16 (2006.01)

(11) 공개번호 10-2007-0002422
(43) 공개일자 2007년01월05일

(21) 출원번호 10-2005-0057959
(22) 출원일자 2005년06월30일
심사청구일자 2005년06월30일

(71) 출원인 고려대학교 산학협력단
서울 성북구 안암동5가1 고려대학교 내

(72) 발명자 황종선
서울 성북구 안암동5가 고려대학교 아산이학관 216호
최봉대
서울 성북구 안암동5가 고려대학교 아산이학관 522호
백맹순
서울 성북구 안암동5가 고려대학교 아산이학관 226호 분산시스템연구실
최성진
서울 성북구 안암동5가 고려대학교 226호 분산시스템 연구실
김홍수
서울 성북구 안암동5가 고려대학교 아산이학과 226호 분산시스템연구실
변은정
서울 성북구 안암동5가 고려대학교 아산이학관 226호 분산시스템연구실

(74) 대리인 현종철

전체 청구항 수 : 총 30 항

(54) 다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트위치관리 및 메시지 전달 방법 및 그 시스템

(57) 요약

다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법 및 그 시스템이 개시된다. 그 이동 에이전트 위치관리 및 메시지 전달 방법은, 이동 에이전트가 생성될 때 룩업서버와 지역서버의 협력하에 이동 에이전트에게 적어도 식별자를 부여하며 자신의 위치를 룩업서버와 지역서버에 등록하는 이동 에이전트 생성단계; 이동 에이전트가 이동할 때 이동에이전트가 생성된 홈 노드나 이동에이전트가 현재 속해있는 지역의 지역 서버에게 자신의 위치를 알리는 이동 에이전트 이주단계; 및 이동 에이전트를 찾은 후 해당 메시지를 상기 이동 에이전트에게 전달하는 이동 에이전트 메시지 전달 단계를 포함함을 특징으로 한다.

본 발명에 의하면, 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트 위치관리 및 메시지 전달 시에 발생하는 위치 갱신 비용 및 메시지 전달 비용을 줄이며, 추적문제를 낮은 통신비용으로 해결한다. 또한, 복제된 이동 에이전트와 부모-자식 이동 에이전트에 대한 위치 관리 및 메시지 전달에 대해서 다뤄, 복제 이동 에이전트들과 부모-자식 이동 에이전트에 대한 메시지 전달을 보장하며 중복된 메시지 전달을 줄여 낮은 통신 비용으로 메시지를 전달한다.

대표도

도 2

특허청구의 범위

청구항 1.

네트워크로 연결되는 적어도 두 개의 지역을 포함하며 상기 지역 각각은 적어도 하나의 노드 및 지역서버를 구비하는, 다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템에 있어서,

상기 노드의 실행 결과에 따라 다음 노드를 동적으로 선택하여 이동하며, 이동계획에 따라 노드와 노드 사이를 이동하며 주어진 작업을 스스로 수행하는 이동 에이전트;

상기 이동에이전트가 수행할 수 있는 실행 환경에 해당하는 실행장소를 제공하고, 이동 에이전트를 생성, 실행, 이주시킬 수 있는 기능을 제공하는 노드;

같은 권한을 가진 노드들의 집합을 지역이라 할 때, 자신의 지역에서 생성된 이동 에이전트에 대해서 룩업서버와 협력하여 네이밍 서비스를 제공하고, 자신의 지역에 위치한 이동 에이전트들에 대한 위치관리 및 메시지 전달을 수행하는 지역서버; 및

다중지역에서 이동 에이전트의 전역 네이밍 서비스와, 이동 에이전트에 대한 룩업서비스를 제공하고, 각 지역에서 생성된 이동 에이전트에 대한 위치관리 정보를 유지 및 관리하는 룩업서버를 포함함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 2.

제1항에 있어서, 상기 지역서버는

위치 추적 메시지나 전달 메시지를 남겨 놓는 블랙보드를 가지며, 상기 위치추적 메시지나 전달 메시지는 이동 에이전트에 의해 읽히는 것을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 3.

제2항에 있어서,

이동 에이전트가 생성될 때는 상기 룩업서버와 지역서버의 협력하에 이동 에이전트에게 적어도 식별자를 부여하며 자신의 위치를 룩업서버와 지역서버에 등록하며, 이동 에이전트가 이동할 때는 이동에이전트가 생성된 홈 노드나 이동에이전트가 현재 속해있는 지역의 지역 서버에게 자신의 위치를 알리며, 메시지 전달할 때는 이동 에이전트를 찾은 후 블랙보드를 사용하여 해당 메시지를 상기 이동 에이전트에게 전달하는 이동 에이전트 메시지 전달함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 4.

제3항에 있어서, 상기 이동에이전트는

사용자의 요구에 따라 복제되거나 자식 이동 에이전트를 생성할 수 있음을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 5.

제4항에 있어서,

홈 노드에서 이동 에이전트가 복제된 경우, 홈 노드에서 복제된 이동 에이전트들에 대한 위치를 관리함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 6.

제4항에 있어서,

이동 에이전트가 연산 수행 중에 복제된 경우, 복제된 이동 에이전트들은 생성 노드에서 이동 에이전트들에 대한 위치를 관리하고 이동 에이전트를 복제한 생성노드는 홈 노드에게 이동 에이전트가 복제되었음을 알려 홈 노드와 생성노드에서 복제된 이동 에이전트들에 대한 위치를 관리함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 7.

제4항에 있어서,

만약 이동 에이전트가 연산 수행 중에 자식 이동 에이전트를 생성한다면, 자식 이동 에이전트의 위치는 부모 이동 에이전트의 홈 노드에 기록됨을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 8.

제4항에 있어서,

홈 노드는 적어도 모든 복제된 이동 에이전트들과 모든 부모-자식 이동 에이전트들 중 하나에 대한 위치 관리 및 메시지 전달을 담당함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 9.

제7항에 있어서,

연산 중간 과정에서 생긴 복제 이동 에이전트 또는 자식 이동 에이전트에 대한 위치관리는 맨 처음 이동 에이전트를 생성한 홈 노드에서 전체적으로 관리하며, 모든 복제 이동 에이전트 또는 자식 이동 에이전트는 위치 갱신 메시지를 홈 노드에게 보냄을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 10.

제9항에 있어서, 상기 홈 노드는

{*AgentID, CloneMAs, Parent_ChildMAs, RSaddrs*} (여기서, 상기 *CloneMAs*는 이동 에이전트가 복제된 경우 복제 이동 에이전트들에 대한 목록으로서, 원래 이동 에이전트(Original mobile agent : OMA) 또는 복제된 이동 에이전트들(Cloned mobile agent : CLMA)을 나타내고, 상기 *Parent_ChildMAs*는 자식 이동 에이전트를 생성한 경우 부모-자식 이동 에이전트(Chikd mobile agent : CHMA)를 나타내고, *RSaddrs*는 O-CLMAs(OMA와 CLMA) 또는 P-CHMAs(PMA와 CHMA)들에 대한 각각의 지역서버의 주소를 나타낸다.) 정보를 유지함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 11.

제10항에 있어서, 복제 이동 에이전트 생성은

CLMAs 또는 CHMAs는 생성 메시지 {*AgentID, HNaddr*}을 지역서버에게 보내고, 지역서버는 다시 상기 생성 메시지를 룩업서버에게 보내며, CLMAs 또는 CHMAs는 HN에게 생성 메시지 {*AgentID, OMA_ID, PMA_ID, RSaddr*}(여기서 *OMA_ID*는 CLMA에 대한 원래 이동 에이전트의 식별자를 나타내고, *PAM_ID*는 CHMA 에 대한 원래 이동 에이전트의 식별자를 나타낸다.)를 보냄에 의해 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 12.

제11항에 있어서, 상기 복제 이동 에이전트 이주는

Intra-RM에서는 CLMAs 또는 CHMAs가 위치 갱신 메시지 {*AgnetID, RNaddr*}를 지역서버에게 보내고, Inter-RM의 경우에는 CLMAs 또는 CHMAs가 위치 갱신 메시지 {*AgentID, RSaddr*}를 HN에게 보내고, {*AgentID, NextRSaddr*}를 이전 지역서버에게 보내고, {*AgentID, RNaddr*}를 현재 지역서버에게 보내어 이주가 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 13.

제12항에 있어서, 상기 복제 이동 에이전트의 메시지 전달은

홈 노드에서 유지관리하고 있는 모든 O-CLMAs 또는 O-CHMAs에 대한 지역 서버들의 주소를 알아낸 다음 메시지를 복제하여 모든 지역서 버들에게 보내어 메시지를 전달함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 14.

제4항에 있어서,

복제된 이동 에이전트들 또는 자식 이동 에이전트에 대한 위치관리를 홈 노드와 중간 연산 과정에서 복제 이동 에이전트를 생성한 생성노드(CN)에서 위치관리 및 메시지 전달을 수행함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 15.

제14항에 있어서,

상기 홈 노드는 복제된 이동 에이전트들 또는 자식 이동 에이전트들을 생성한 생성노드에 대한 위치정보만 가지고 있고 이동 에이전트 자체에 대한 위치정보는 가지고 있지 않으며, 생성노드가 복제 이동 에이전트들 또는 자식 이동 에이전트들에

대한 실제 위치정보를 유지. 관리하며, 모든 복제 이동 에이전트들 또는 자식 이동 에이전트들은 위치갱신 메시지를 생성노드에게 보내고, 생성노드에 의해서 메시지 전달 과정이 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 16.

제15항에 있어서,

상기 홈 노드는 {*AgentID, RSaddr, CNaddr*} (여기서, *RSaddr*는 O-CLMAs(OMA와 CLMA)에 대한 지역 서버의 주소를 나타내고, *CNaddr*은 복제이동 에이전트르 생성한 생성노드의 주소를 나타낸다..) 정보를 유지하고,

상기 생성노드는 {*AgentID, CloneMAs, Parent_ChikdMAs, RSaddrs*} (여기서, 상기 *CloneMAs*는 이동 에이전트가 복제된 경우 복제 이동 에이전트들에 대한 목록으로서, 원래 이동 에이전트(Original mobile agent : OMA) 또는 복제된 이동 에이전트들(Cloned mobile agent : CLMA)을 나타내고, *Parent_ChildMAs*는 자식 이동 에이전트를 생성한 경우 부모-자식 이동 에이전트(Chikd mobile agent : CHMA)를 나타내고, *RSaddrs*는 O-CLMAs(OMA와 CLMA) 또는 P-CHMAs (PMA와 CHMA)들에 대한 각각의 지역 서버의 주소를 나타낸다.) 정보를 유지함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 17.

제16항에 있어서, 상기 복제 이동 에이전트 또는 자식 이동 에이전트의 생성은

CLMAs 또는 CHMSs가 생성 메시지 {*AgentID, HNaddr*}을 지역서버에게 보내고, 지역서버는 다시 상기 생성 메시지를 룩업서버에게 보내며, CLMAs 또는 CHMAs는 HN에게 생성메시지 {*AgentID, OMA_ID, PMA_ID, CNaddr*}를 보냄에 의해 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 18.

제17항에 있어서, 상기 복제 이동 에이전트 또는 자식 이동 에이전트의 이주는

Intra-RM에서는 CLMAs 또는 CHMAs가 위치 갱신 메시지 {*AgentID, RNaddr*}를 지역서버에게 보내고, Inter-RM의 경우에는 CLMAs 또는 CHMAs는 위치 갱신 메시지 {*AgentID, RSaddr*}를 CN에게 보내고, {*AgentID, NextRSaddr*}를 이전 지역서버에게 보내고, {*AgentID, RNaddr*}를 현재 지역서버에게 보냄에 의해 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 19.

제18항에 있어서, 상기 복제 이동 에이전트 또는 자식 이동 에이전트의 메시지 전달은

홈 노드에서 유지관리하고 있는 모든 O-CLMAs 또는 모든 P-CHMAs에 대한 지역 서버들의 주소를 알아낸 다음 메시지를 복제하여 모든 지역 서버들에게 보냄에 의해 이루어짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 시스템.

청구항 20.

네트워크로 연결되는 적어도 두 개의 지역을 포함하며 상기 지역 각각은 적어도 하나의 노드 및 지역서버를 구비하는, 다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법에 있어서,

이동 에이전트가 생성될 때 룩업서버와 지역서버의 협력하에 이동 에이전트에게 적어도 식별자를 부여하며 자신의 위치를 룩업서버와 지역서버에 등록하는 이동 에이전트 생성단계;

이동 에이전트가 이동할 때 이동에이전트가 생성된 홈 노드나 이동에이전트가 현재 속해있는 지역의 지역 서버에게 자신의 위치를 알리는 이동 에이전트 이주단계; 및

이동 에이전트를 찾은 후 해당 메시지를 상기 이동 에이전트에게 전달하는 이동 에이전트 메시지 전달 단계를 포함함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 21.

제20항에 있어서, 상기 이동 에이전트 생성단계는

노드에서 이동 에이전트를 생성하는 단계;

홈 노드가 지역서버에게 이동 에이전트 생성 메시지를 보내어 지역서버에 등록하는 단계; 및

지역서버가 룩업서버에게 이동 에이전트의 생성정보를 보내어 룩업 서버에 등록하는 단계를 포함함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 22.

제20항에 있어서, 상기 이동 에이전트 이주단계는

이동 에이전트가 현재 자신이 위치한 지역서버에게 위치갱신 메시지를 보내는 동일한 지역 내에서의 이주(Intra-RM)와, 다른 지역으로의 이주(Inter-RM)를 구비함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 23.

제22항에 있어서, 상기 Inter-RM은

이동 에이전트가 위치 갱신 메시지를 홈 노드에게 보내어 홈 노드가 현재 에이전트가 있는 지역을 알 수 있게 하는 단계;

이동 에이전트가 이전 지역서버에게 위치갱신 메시지를 보내어 이동 에이전트의 현재 지역 위치가 홈 노드에게 알려지기 전에 이동 에이전트에게 보냈던 메시지의 전달을 보장하는 단계; 및

이동 에이전트가 위치갱신 메시지를 현재 지역서버에게 보내어 이동 에이전트 자신이 현재 지역에 이동했다는 것을 알리는 단계를 포함함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 24.

제20항에 있어서, 상기 이동 에이전트 메시지 전달 단계는

이주하는 이동 에이전트에게 비동기적으로 메시지를 전달하기 위해 블랙보드를 사용하고, 각 지역서버는 블랙보드를 두어 자신의 지역에 있는 모든 이동 에이전트들에 대한 메시지 전달을 책임짐을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 25.

제20항에 있어서, 상기 이동 에이전트 메시지 전달 단계는

지역서버가 이동 에이전트로부터 위치 갱신 메시지를 받을 때 메시지 전달과정이 수행되는 반응적(Reactive) 메시지 전달 단계; 및

응급 메시지 전달을 위한 것으로, 지역서버가 메시지를 받자 마자 메시지 전달 과정이 수행되는 적극적(Proactive) 메시지 전달 단계 중 적어도 하나를 구비함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 26.

제25항에 있어서, 상기 반응적 메시지 전달 단계는

송신자(sender)는 먼저 룩업서버를 통하여 수신 이동 에이전트의 홈 노드를 알아낸 후 메시지를 홈 노드에게 전송하는 단계;

홈 노드는 수신 이동 에이전트가 있는 현재 지역서버를 알아낸 후 메시지를 지역서버로 보내는 단계;

지역서버는 자신의 블랙보드에 해당 메시지를 저장하는 단계;

지역서버는 이동 에이전트로부터 위치갱신 메시지를 받을 때, 해당 이동 에이전트에게 전달할 메시지가 블랙보드에 있는지를 검사하는 단계; 및

만약 메시지가 있다면, 블랙보드에서 메시지를 꺼낸 다음 해당 이동 에이전트에게 메시지를 전달하는 단계를 구비함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 27.

제26항에 있어서,

이동 에이전트는 자신에게 전달될 메시지가 있는지 없는지를 이주할 때 마다 검사함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 28.

제25항에 있어서, 상기 적극적 메시지 전달 단계는

지역서버가 메시지를 받으면, 메시지를 블랙보드에 저장한 다음 바로 메시지를 현재 이동 에이전트가 있는 노드에게 전송함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 29.

제28항에 있어서, 상기 적극적 메시지 전달 단계는

송신자(sender)는 먼저 룩업서버를 통하여 수신 이동 에이전트의 홈 노드를 알아낸 후 메시지를 홈 노드에게 전송하는 단계;

홈 노드는 수신 이동 에이전트가 있는 현재 지역서버를 알아낸 후 메시지를 지역서버로 보내는 단계;

지역서버는 자신의 블랙보드에 해당 메시지를 저장하는 단계; 및

지역서버는 현재 수신 이동 에이전트가 상주하고 있는 노드의 주소를 알아낸 후, 메시지를 수신 이동 에이전트에게 전달하는 단계를 구비함을 특징으로 하는 다중 지역 이동 에이전트 위치관리 및 메시지 전달 방법.

청구항 30.

제20항 내지 29항 중 어느 한 항에 기재된 발명을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 다중지역 이동 에이전트에 관한 것으로서, 특히 다중 지역 이동 에이전트 컴퓨터 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법 및 그 시스템에 관한 것이다.

이동 에이전트(Mobile agent)는 이질적인 네트워크 환경에서 노드들을 이동하면서 사용자를 대신하여 자율적으로 주어진 일을 처리하는 프로그램이다. 이동 에이전트 수행 중 이동 에이전트의 제어와 이동 에이전트들 간의 메시지 전달을 보장하는 위치 관리 및 메시지 전달 기법은 이동 에이전트 시스템(Mobile agent system)을 개발하는데 있어서 중요한 고려사항 중의 하나이다.

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. 이들 프로토콜들을 결합한 쉐도우(Shadow), SPC(Search-by-Path-Chase) 기법으로 분류된다. 이러한 관련 연구들 중 블랙보드와 이메일 기법은 주로 메시지 전달에 사용되지만, 나머지 기법들은 위치 관리와 메시지 전달에 동시에 사용된다.

상기 홈 프락시 기법은 이동 에이전트의 위치정보를 이동 에이전트가 생성된 노드, 즉 홈 노드(HN)나 특정 서버에 저장한 다음, 상기 위치정보를 이용하여 이동 에이전트를 찾거나 메시지를 전달하는 기법이다. 이동 에이전트는 자신이 메시지를 전송하고자 하는 이동 에이전트에 대한 홈 노드의 위치를 룩업서버(Lookup Server : LS)를 통해 찾은 후, 해당 홈 노드에게 메시지를 전달한다. 홈 노드는 현재 이동 에이전트의 위치정보에 따라 전달된 메시지를 해당 이동 에이전트에게 전달하게 된다. 홈 프락시 기법은 홈 노드나 특정 서버가 이동 에이전트에 대한 위치 정보를 관리하기 때문에 만약 홈 노드와 이동 에이전트의 거리가 멀어지면 이동 에이전트의 위치정보 변경시 마다 홈 노드로 갱신 메시지를 전송할 때나, 해당 이동 에이전트로 메시지를 전송할 때 통신비용이 증가하는 단점을 가지고 있다. 또한, 이동 에이전트의 수가 많아지고 이동 에이전트들간의 통신과 이동 에이전트의 이주가 자주 일어나는 상황에서는 홈 노드나 특정 서버는 병목현상(Bottleneck)을 겪게 된다.

상기 추적 프락시 기법은 이동 에이전트에 대한 위치 정보를 이동 에이전트가 방문한 노드들에 저장한 다음, 각 노드에 저장된 위치정보 경로, 경로 프락시(Path proxies)를 따라가서 이동 에이전트를 찾거나 메시지를 경로 프락시를 따라 계속해서 포워딩(Forwarding)하여 해당 이동 에이전트에게 메시지를 전달하는 기법이다. 이동 에이전트는 자신이 메시지를 전송하고자 하는 이동 에이전트에 대한 홈 노드의 위치를 룩업서버를 통해 찾은 후, 해당 홈 노드에게 메시지를 전달한다. 홈 노드는 이동 에이전트가 이동한 다음 노드에게 해당 메시지를 다시 전달한다. 다음 노드는 이동 에이전트가 이동한 그 다음 노드에게 메시지를 전달한다. 이와 같이 추적 프락시 기법은 포워딩 과정을 계속 되풀이하면서 이동 에이전트를 찾거나 메시지를 전달한다. 추적 프락시 기법은 홈 노드나 특정 서버의 부하는 줄어드는 반면에 메시지의 전달을 위해서 이동 에이전트가 이동한 경로 상의 모든 노드가 참가해야 하는 단점을 가지고 있다. 즉, 이동 에이전트가 방문한 모든 노드에 위치관련 정보를 저장해야 하고 메시지를 포워딩을 해주는 포워딩 데몬을 유지해야 하는 오버헤드가 있다. 또한 이동 에이전

트가 이동할 때마다 경로 프락시가 증가하게 되어 메시지 전달하는데 높은 통신비용이 발생하며, 경로 프락시를 이루는 중간 노드가 고장이 나면 경로 프락시를 이루는 체인(Chain)이 깨져 이동 에이전트에게 메시지를 전달할 수 없는 상황이 발생한다. 또한, 이동 에이전트에게 메시지를 전달하지 못하고 이동 에이전트가 방문한 노드들을 계속해서 탐색하는 문제가 발생한다.

상기 브로드캐스트 기법은 정해진 지역이나 이동계획(Itinerary)의 모든 노드들에게 메시지를 보내어 이동 에이전트의 위치를 찾거나 메시지를 전달하는 기법이다. 브로드 캐스팅 기법은 이동 에이전트 수행과정의 모든 노드들에게 해당 메시지를 브로드캐스팅 해야하기 때문에 높은 통신비용이 발생한다.

상기 블랙보드 기법은 각 노드에 블랙보드를 두어 메시지를 교환하는 기법이다. 메시지를 전송하고자 하는 이동 에이전트는 현재 노드의 블랙보드에 메시지를 남긴다. 이후 메시지를 수신하고자 하는 이동 에이전트가 해당 노드를 방문하여 자신의 메시지를 수신한다. 블랙보드 기법은 모든 노드들이 메시지를 저장할 기억장소를 가지고 있어야 하는 단점을 가지고 있다. 또한, 이동 에이전트가 메시지를 전달받기 위해서는 직접 해당 노드로 이동해야 하는데 메시지를 수신하는 이동 에이전트가 해당 노드로 이동하는 패턴이 일정하지 않고 임의로 이루어지기 때문에 메시지 전달이 즉각적으로 이루어지지 않는 단점을 가지고 있다. 그리고 간단한 메시지의 경우, 직접 이동 에이전트가 해당 노드로 이동하여 처리해야 하기 때문에 메시지를 처리하는 과정에서 불필요한 통신 부하가 발생한다.

상기 이메일 기법은 이메일 서버와 유사한 방법으로 메시지를 전송하고자 하는 이동 에이전트는 메일 서버에 메시지를 남겨 놓으면 메시지를 수신하고자 하는 이동 에이전트가 서버에 접속하여 메시지를 얻는 기법이다. 이동 에이전트의 위치정보를 이용하지 않고, 서버에 이동 에이전트를 위한 메시지 큐를 두어 메시지를 전달하는 이메일 기법은 메시지의 전달이 즉각적으로 이루어지지 않는다는 단점을 가지고 있다.

상기 쉐도우(Shadow) 기법은 홈 프락시 프로토콜과 추적 프락시 프로토콜을 결합한 프로토콜이다. 이동 에이전트는 자신의 현재 위치를 생존시간 (TTL : Time to live)에 따라 자신의 쉐도우에 갱신을 한다. 다시 말하면, TTL이 지나지 않으면 이동 에이전트는 추적 프락시 방법을 사용을 하여 경로 프락시를 유지한다. TTL이 지나면 이동 에이전트는 홈 프락시 방법을 사용하여 자신의 쉐도우에 위치 갱신 메시지를 보냄으로써 경로 프락시를 줄이게 된다. 메시지 전달 과정은 경로 프락시를 이용하거나 쉐도우에 저장된 이동 에이전트의 위치를 이용하여 이루어진다. 쉐도우 프로토콜은 추적 프락시 프로토콜을 사용함으로 인해 위치 관리 및 메시지 전달하는데 많은 통신비용이 발생하며 이동 에이전트가 지나온 경로만 계속 탐색하는 추적 문제가 발생한다.

상기 SPC(Search-by-Path-Case) 프로토콜은 홈 프락시 프로토콜과 추적 프락시 프로토콜을 결합한 프로토콜로써 다중 지역 이동 에이전트 컴퓨팅 환경을 고려하고 있다. 이동 에이전트의 위치관리 정보는 RAR(Regin Agent Register)이나 SAR(Site Agent Register)에서 분산된 방법으로 유지된다. 메시지 전달 과정은 RAR과 SAR을 기반으로 이동 에이전트가 남긴 경로를 따라 이루어지게 된다. 그러나, SPC 프로토콜도 경로 프락시(Path Proxies)를 사용함으로 인해 위치 관리 및 메시지 전달하는데 많은 통신비용이 발생하며 이동 에이전트가 지나온 경로만 계속 탐색하는 추적 문제가 발생한다.

이와 같이 기존의 위치관리 및 메시지 전달 프로토콜들은 크게 다음과 같은 문제점을 가지고 있다. 첫째, 여러 지역으로 다중 지역 이동 에이전트 컴퓨팅 환경에서 기존 기법들을 적용하였을 경우 이동 에이전트의 현재 노드와 홈 노드와의 거리에 따라 위치 갱신 비용 및 메시지 전달 비용이 증가하는 문제점이 발생한다. 둘째, 이동 에이전트에게 해당 메시지를 전달하지 못하고 이동 에이전트가 방문한 노드들만 계속해서 탐색하는 문제가 발생한다. 마지막으로 복제된 이동 에이전트와 부모-자식 이동 에이전트에 대한 위치관리 및 메시지 전달에 대해서 다루지 않아, 부모-자식 이동 에이전트에 대한 메시지 전달을 보장하지 못하며, 복제 이동 에이전트들에 대한 중복 메시지 전달이 발생하여 통신비용이 크다. 따라서 효율적으로 이동 에이전트의 위치를 추적하고 메시지를 전달하기 위한 기법이 필요하다.

발명이 이루고자 하는 기술적 과제

본 발명이 이루고자 하는 기술적 과제는 홈 프락시 기법과 블랙보드 기법을 다중 지역 이동 에이전트 컴퓨팅 환경에 맞게 변형 및 혼합(Hybrid)하는 HB프로토콜을 이용함으로써, 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법 및 시스템을 제공하는 것이다.

발명의 구성

상기 기술적 과제를 이루기 위한 본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템은, 네트워크로 연결되는 적어도 두 개의 지역을 포함하며 상기 지역 각각은 적어도 하나의 노드 및 지

역서버를 구비하는, 다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템에 있어서, 상기 노드의 실행 결과에 따라 다음 노드를 동적으로 선택하여 이동하며, 이동계획에 따라 노드와 노드 사이를 이동하며 주어진 작업을 스스로 수행하는 이동에이전트; 상기 이동에이전트가 수행할 수 있는 실행 환경에 해당하는 실행장소를 제공하고, 이동 에이전트를 생성, 실행, 이주시킬 수 있는 기능을 제공하는 노드; 같은 권한을 가진 노드들의 집합을 지역이라 할 때, 자신의 지역에서 생성된 이동 에이전트에 대해서 룩업서버와 협력하여 네이밍 서비스를 제공하고, 자신의 지역에 위치한 이동 에이전트들에 대한 위치관리 및 메시지 전달을 수행하는 지역서버; 및 다중지역에서 이동 에이전트의 전역 네이밍 서비스와, 이동 에이전트에 대한 룩업서비스를 제공하고, 각 지역에서 생성된 이동 에이전트에 대한 위치관리 정보를 유지 및 관리하는 룩업서버를 포함함을 특징으로 한다.

상기 지역서버는 위치 추적 메시지나 전달 메시지를 남겨 놓는 블랙보드를 가지며, 상기 위치추적 메시지나 전달 메시지는 이동 에이전트에 의해 읽히는 것이 바람직하다.

상기 본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템은, 이동 에이전트가 생성될 때는 상기 룩업서버와 지역서버의 협력하에 이동 에이전트에게 적어도 식별자를 부여하며 자신의 위치를 룩업서버와 지역서버에 등록하며, 이동 에이전트가 이동할 때는 이동에이전트가 생성된 홈 노드나 이동에이전트가 현재 속해있는 지역의 지역 서버에게 자신의 위치를 알리며, 메시지 전달할 때는 이동 에이전트를 찾은 후 블랙보드를 사용하여 해당 메시지를 상기 이동 에이전트에게 전달하는 이동 에이전트 메시지 전달함이 바람직하다. 상기 이동에이전트는 사용자의 요구에 따라 복제되거나 자식 이동 에이전트를 생성할 수 있음이 바람직하다.

상기 본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템은, 홈 노드에서 이동 에이전트가 복제된 경우, 홈 노드에서 복제된 이동 에이전트들에 대한 위치를 관리함이 바람직하다.

상기 본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 시스템은, 이동 에이전트가 연산 수행 중에 복제된 경우, 복제된 이동 에이전트들은 생성 노드에서 이동 에이전트들에 대한 위치를 관리하고 이동 에이전트를 복제한 생성노드는 홈 노드에게 이동 에이전트가 복제되었음을 알려 홈 노드와 생성노드에서 복제된 이동 에이전트들에 대한 위치를 관리함이 바람직하다.

또한 상기 본 발명에 의한 시스템은 만약 이동 에이전트가 연산 수행 중에 자식 이동 에이전트를 생성한다면, 자식 이동 에이전트의 위치는 부모 이동 에이전트의 홈 노드에 기록됨이 바람직하다.

또한 상기 본 발명에 의한 시스템은 홈 노드는 적어도 모든 복제된 이동 에이전트들과 모든 부모-자식 이동 에 이전트들 중 하나에 대한 위치 관리 및 메시지 전달을 담당함이 바람직하다. 상기 본 발명에 의한 시스템은 연산 중간 과정에서 생긴 복제 이동 에이전트 또는 자식 이동 에이전트에 대한 위치관리는 맨 처음 이동 에이전트를 생성한 홈 노드에서 전체적으로 관리하며, 모든 복제 이동 에이전트 또는 자식 이동 에이전트는 위치 갱신 메시지를 홈 노드에게 보냄이 바람직하다. 또한 상기 본 발명에 의한 시스템은 복제된 이동 에이전트들 또는 자식 이동 에이전트에 대한 위치관리를 홈 노드와 중간 연산 과정에서 복제 이동 에이전트를 생성한 생성노드(CN)에서 위치관리 및 메시지 전달을 수행함이 바람직하다. 상기 본 발명에 의한 시스템은 상기 홈 노드는 복제된 이동 에이전트들 또는 자식 이동 에이전트들을 생성한 생성노드에 대한 위치정보만 가지고 있고 이동 에이전트 자체에 대한 위치정보는 가지고 있지 않으며, 생성노드가 복제 이동 에이전트들 또는 자식 이동 에이전트들에 대한 실제 위치정보를 유지. 관리하며, 모든 복제 이동 에이전트들 또는 자식 이동 에이전트들은 위치 갱신 메시지를 생성노드에게 보내고, 생성노드에 의해서 메시지 전달 과정이 이루어짐이 바람직하다.

상기 기술적 과제를 이루기 위한 본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법은, 네트워크로 연결되는 적어도 두 개의 지역을 포함하며 상기 지역 각각은 적어도 하나의 노드 및 지역서버를 구비하는, 다중 지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법에 있어서, 이동 에이전트가 생성될 때 룩업서버와 지역서버의 협력하에 이동 에이전트에게 적어도 식별자를 부여하며 자신의 위치를 룩업서버와 지역서버에 등록하는 이동 에이전트 생성단계; 이동 에이전트가 이동할 때 이동에이전트가 생성된 홈 노드나 이동에이전트가 현재 속해있는 지역의 지역서버에게 자신의 위치를 알리는 이동 에이전트 이주단계; 및 이동 에이전트를 찾은 후 해당 메시지를 상기 이동 에이전트에게 전달하는 이동 에이전트 메시지 전달 단계를 포함함을 특징으로 한다. 상기 이동 에이전트 생성단계는 노드에서 이동 에이전트를 생성하는 단계; 홈 노드가 지역서버에게 이동 에이전트 생성 메시지를 보내어 지역서버에 등록하는 단계; 및 지역서버가 룩업서버에게 이동 에이전트의 생성정보를 보내어 룩업 서버에 등록하는 단계를 포함함이 바람직하다. 상기 이동 에이전트 이주단계는 이동 에이전트가 현재 자신이 위치한 지역서버에게 위치갱신 메시지를 보내는 동일한 지역 내에서의 이주(Intra-RM)와, 다른 지역으로의 이주(Inter-RM)를 구비함이 바람직하다. 상기 Inter-RM은 이동 에이전트가 위치 갱신 메시지를 홈 노드에게 보내어 홈 노드가 현재 에이전트가 있는 지역을 알 수 있게 하는 단계; 이동 에이전트가 이전 지역서버에게 위치갱신 메시지를 보내어 이동 에이전트의 현재 지역 위치가

홈 노드에게 알려지기 전에 이동 에이전트에게 보냈던 메시지의 전달을 보장하는 단계; 및 이동 에이전트가 위치갱신 메시지를 현재 지역서버에게 보내어 이동 에이전트 자신이 현재 지역에 이동했다는 것을 알리는 단계를 포함함이 바람직하다. 상기 이동 에이전트 메시지 전달 단계는 이주하는 이동 에이전트에게 비동기적으로 메시지를 전달하기 위해 블랙보드를 사용하고, 각 지역서버는 블랙보드를 두어 자신의 지역에 있는 모든 이동 에이전트들에 대한 메시지 전달을 책임짐이 바람직하다. 상기 이동 에이전트 메시지 전달 단계는 지역서버가 이동 에이전트로부터 위치 갱신 메시지를 받을 때 메시지 전달과정이 수행되는 반응적(Reactive) 메시지 전달 단계; 및 응급 메시지 전달을 위한 것으로, 지역서버가 메시지를 받자 마자 메시지 전달 과정이 수행되는 적극적(Proactive) 메시지 전달 단계 중 적어도 하나를 구비함이 바람직하다. 상기 반응적 메시지 전달 단계는 송신자(sender)는 먼저 룩업서버를 통하여 수신 이동 에이전트의 홈 노드를 알아낸 후 메시지를 홈 노드에게 전송하는 단계; 홈 노드는 수신 이동 에이전트가 있는 현재 지역서버를 알아낸 후 메시지를 지역서버로 보내는 단계; 지역서버는 자신의 블랙보드에 해당 메시지를 저장하는 단계; 지역서버는 이동 에이전트로부터 위치갱신 메시지를 받을 때, 해당 이동 에이전트에게 전달할 메시지가 블랙보드에 있는지를 검사하는 단계; 및 만약 메시지가 있다면, 블랙보드에서 메시지를 꺼낸 다음 해당 이동 에이전트에게 메시지를 전달하는 단계를 구비함이 바람직하다.

상기 본 발명에 의한 이동 에이전트 위치관리 및 메시지 전달 방법은, 이동 에이전트는 자신에게 전달될 메시지가 있는지 없는지를 이주할 때 마다 검사함이 바람직하다. 상기 적극적 메시지 전달 단계는 지역서버가 메시지를 받으면, 메시지를 블랙보드에 저장한 다음 바로 메시지를 현재 이동 에이전트가 있는 노드에게 전송함이 바람직하다. 상기 적극적 메시지 전달 단계는 송신자(sender)는 먼저 룩업서버를 통하여 수신 이동 에이전트의 홈 노드를 알아낸 후 메시지를 홈 노드에게 전송하는 단계; 홈 노드는 수신 이동 에이전트가 있는 현재 지역서버를 알아낸 후 메시지를 지역서버로 보내는 단계; 지역서버는 자신의 블랙보드에 해당 메시지를 저장하는 단계; 및 지역서버는 현재 수신 이동 에이전트가 상주하고 있는 노드의 주소를 알아낸 후, 메시지를 수신 이동 에이전트에게 전달하는 단계를 구비함이 바람직하다.

그리고 상기 기재된 발명을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공한다.

이하, 첨부된 도면들을 참조하여 본 발명에 따른 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법 및 시스템에 대해 상세히 설명한다.

먼저, 다중지역 이동 에이전트 컴퓨팅 환경을 설명하기로 한다. 본 발명에서는 여러 지역으로 구성된 다중 지역 이동 에이전트 컴퓨팅 환경을 가정하고 있다. 도 1은 다중 지역 이동 에이전트 컴퓨팅 환경을 도시한 것이다. 다중 지역 이동 에이전트 컴퓨팅 환경은 크게 이동 에이전트, 노드(또는 실행장소(Place)), 지역(Region), 지역 서버(Region server : RS), 룩업 서버(Lookup Server : LS)로 구성된다.

상기 이동 에이전트는 코드, 데이터, 실행 상태, 이동 메타데이터로 구성된 이동 객체이다. 이동 에이전트는 각 노드의 실행 결과에 따라 다음 노드를 동적으로 선택하여 이동하거나, 이동계획(itinerary)에 따라 노드와 노드사이를 이동하며 주어진 작업을 자치적으로 수행한다. 이동 에이전트가 이동을 시작한 출발지에서부터 목적지까지를 이동에이전트의 이주 경로(Migration Path)라 한다.

상기 노드(실행장소)는 이동 에이전트가 수행할 수 있는 실행환경인 실행장소를 제공한다. 다시 말하면, 노드에는 이동 에이전트 시스템이 설치되어 있어 이동 에이전트를 생성, 실행, 이주, 종료시킬 수 있는 기능을 제공한다. 이동 에이전트를 맨 처음 생성한 노드를 홈 노드 (Home node : HN)이라고 한다.

같은 권한을 가진 노드들의 집합을 지역이라고 한다. 그리고 각 지역의 권한을 책임지는 서버를 지역 서버(Region Server : RS)라 한다. 지역 서버는 자신의 지역에서 생성된 이동 에이전트에 대해서 룩업 서버와 협력하여 네이밍 서비스를 제공한다. 또한 자신의 지역에 위치한 이동 에이전트들에 대한 위치 관리 및 메시지 전달을 수행한다.

상기 룩업 서버(Lookup server : LS)는 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 전역 네이밍 서비스를 제공하며 또한 이동 에이전트에 대한 록업 서비스를 제공한다. 룩업 서버는 모든 지역에서 생성된 이동 에이전트에 대한 위치 관리 정보를 유지 관리 한다.

다음으로 HB프로토콜을 설명하기로 한다. 여러 지역으로 구성된 다중 이동 에이전트 컴퓨팅 환경에서 홈 프락시 기법을 사용하여 이동 에이전트의 위치관리를 수행하면 위치 갱신 비용이 증가한다. 다중 이동 에이전트 환경에서 이동 에이전트는 전 세계적으로 연결된 네트워크를 통하여 지리적 제한을 받지 않고 이동하기 때문에 이동 에이전트가 이동할 수 있는 지역이 한정되지 않아 이동 에이전트와 홈 노드와 멀리 떨어지면 위치 갱신 비용은 증가하게 된다. 또한 이동 에이전트 탐

색과정 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. 또한, 이동 에이전트의 수가 많아지고 이동 에이전트들간의 통신과 이동 에이전트의 이주가 자주 일어나는 상황에서는 홈 노드나 특정 서버는 병목현상을 겪게 된다.

이에 반해, 블랙보드 기법은 이동 에이전트가 메시지를 전달받기 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 메시지 전달 비용이 증가한다. 그리고, 간단한 메시지의 경우 직접 이동 에이전트가 해당 노드로 이동하여 처리하는 시점에서 불필요한 통신 부하가 발생한다.

따라서, 본 발명에서는 각 기법의 단점을 보완하고 이동 에이전트의 위치관리 및 메시지 전달 비용을 줄이기 위해 두 기법을 변형 및 혼합하는 HB 프로토콜을 제공한다.

홈 프락시 기법의 문제점 즉, 증가된 위치 갱신 비용 문제점과 서버의 병목현상을 해결하기 위해 홈 프락시 기법을 변형하고 블랙보드 기법을 적용한다. 증가된 위치 갱신 비용 문제점과 홈 노드나 서버의 병목현상을 해결하기 위해 각 지역을 관리하는 지역서버가 해당 지역에 있는 모든 이동 에이전트에 대한 위치 관리 및 메시지 전달을 관리한다. 따라서, 지역적으로 가까운 지역 서버에 위치 갱신 메시지를 보내기 때문에 위치 갱신 비용은 줄어들게 된다. 각 지역 서버가 이동 에이전트에 대한 위치 관리 및 메시지 전달에 참여함으로써 홈 노드나 서버에서 발생하는 병목 현상을 줄여준다.

블랙보드 기법의 문제점 즉, 메시지를 적절한 시기에 전달하지 못하는 문제점과 이동 에이전트가 직접 해당 노드에 찾아가서 메시지를 전달받는 시점에서의 통신부하 문제점을 해결하기 위해 홈 프락시 기법을 적용한다. 메시지를 적절한 시기에 전달하기 위해서 이동 에이전트가 위치를 주기적으로 갱신할 때 해당 메시지를 전달하는 방법으로 문제를 해결한다. 이동 에이전트가 직접 해당 노드에 찾아가지 않고도 메시지를 전달받기 위해 지역 서버에 블랙보드를 두어 지역 서버가 관리하는 이동 에이전트에게 메시지를 전달한다.

그리고 이동 에이전트의 현재 위치를 찾지 못하고 계속해서 이동 에이전트가 다녀간 곳을 탐색하는 추적 문제를 해결하기 위해 홈 프락시 기법에 블랙보드 기법을 변형 및 혼합한다. 기존의 블랙보드 기법처럼 모든 노드에 블랙보드를 두는 것이 아니라 HB 프로토콜에서는 각 지역 서버에 블랙보드를 두어 각 지역서버에 있는 블랙보드에 위치 추적 메시지나 전달 메시지를 남겨 놓은 다음 이동 에이전트가 이 메시지를 블랙보드로부터 읽어 추적 문제를 해결하고 메시지 전달을 보장하게 한다.

HB 프로토콜은 크게 생성 (Creation), 이주 (Migration), 메시지 전달 (Message Delivery) 과정으로 구분된다. 생성 과정에서는 이동 에이전트가 생성될 때 룩업 서버와 지역서버의 협력하에 이동 에이전트에게 유일한 이름 및 식별자 (global name or identifier)를 부여하며 자신의 위치를 룩업 서버와 지역 서버에 등록하는 과정을 말한다. 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.

도 2는 본 발명에 의한 다중 지역 이동 에이전트 컴퓨팅 환경에서 HB 프로토콜을 이용한 이동 에이전트 위치관리 및 메시지 전달 시스템의 구성을 블록도로 도시한 것이다.

룩업 서버는 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트의 전역 네이밍 서비스를 제공한다. 룩업 서버는 *AgentID, HNaddr* 엔트리를 유지한다. *AgentID*는 이동 에이전트의 식별자를 나타낸다. 그리고 *HNaddr*는 이동 에이전트의 홈 노드 주소를 나타낸다. 지역 서버는 해당 지역 내에 있는 모든 이동 에이전트에 대한 위치관리 및 메시지 전달에 관련된 정보를 유지?관리한다. 기존의 블랙보드 기법과 달리 모든 노드들이 블랙보드를 갖지 않고 각 지역을 대표하는 지역 서버만이 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 수행한다. 지역서버는 *AgentID, RNaddr, NextRSaddr, Msgs* 엔트리를 유지한다. *RNaddr*은 현재 이동 에이전트가 상주해 있는 노드(Resident Node)의 위치를 나타낸다. *NextRSaddr*은 이동 에이전트가 다음 지역으로 이동했을 경우 이동 에이전트가 위치하고 있는 지역 서버의 주소를 나타낸다. *Msgs*는 블랙보드에 저장되어 있는 메시지들을 나타낸다. 홈 노드는 이동 에이전트가 맨 처음 생성된 노드이다. 홈 노드는 *AgentID, RSaddr* 엔트리를 유지한다. RSaddr은 현재 이동 에이전트가 위치하고 있는 지역 서버의 주소를 나타낸다.

이동 에이전트의 생성에 대해 설명하기로 한다. 도 3은 이동 에이전트 생성과정을 흐름도로 도시한 것이다. 이동 에이전트의 생성 과정은 이동 에이전트가 생성될 때 룩업 서버와 지역서버의 협력하에 이동 에이전트에게 유일한 이름 및 식별자

(global name or identifier)를 부여하며 자신의 위치를 룩업 서버와 지역 서버에 등록하는 과정을 말한다. 지역 서버는 자신의 지역에서 생성된 이동 에이전트에게 네이밍 서비스를 제공한다. 그리고 난 다음, 지역서버는 이동 에이전트의 생성 정보, 즉 *AgentID, HNAddr*를 룩업 서버에게 등록한다. 룩업 서버는 지역 서버에서 제공한 이동 에이전트의 식별자가 서로 같을 때 전역적 네이밍 서비스를 제공하여 이 문제를 해결한다. 이와 같이 이동 에이전트가 생성될 때 생성 메시지(Creation message)를 지역서버가 룩업서버에게 보냄으로써, 여러 이동 에이전트가 동시에 지역 서버에서 생성된 경우, 지역 서버는 여러 이동 에이전트들의 생성 메시지들을 여러 번 보내지 않고 하나의 메시지 형태로 룩업 서버에 보냄으로써 이동 에이전트가 생성 될 때마다 보내는 생성 메시지 비용을 줄일 수 있다. 도 2에서 지역 A에서 생성된 이동 에이전트에 대한 생성 메시지 (Creation message)가 지역서버 RSA에게 보내진 후 룩업서버 LS에게 보내지는 과정이 녹색의 점선으로 설명되어 있다. 이를 설명하면, 먼저 홈 노드가 이동 에이전트를 생성한다.(300단계) 또한 홈 노드는 지역서버(RS)에게 이동 에이전트 생성 메시지를 보낸다.(320단계) 지역 서버(RS)가 룩업서버(RS)에게 이동 에이전트 생성 메시지를 보낸다.(340단계)

이동 에이전트의 이주에 대해 설명하기로 한다. 이동 에이전트의 이주과정은 이동 에이전트가 이동할 때 홈 노드나 지역 서버에게 자신의 위치를 알리는 등록과정을 말한다. 이주과정 알고리즘은 크게 같은 지역 내에서의 이주(Intra-region migration : Intra-RM)와 다른 지역으로의 이주(Inter-region migration : Inter-RM)로 구분된다. 이주 과정에 전체 과정은 그림 2에서 파란 실선으로 나타나 있다. 도 4는 이동 에이전트의 이주과정을 흐름도로 도시한 것이다. 이동 에이전트가 노드 A에서 다른 노드 B로 이주한다.(400단계)

노드 A와 노드 B가 같은 지역에 있는지 판단한다.(410단계) 같은 지역 내에서 이동 에이전트가 이주할 때 (즉 Intra-RM의 경우), 이동 에이전트는 현재 자신이 위치한 지역서버(Current RS)에게 위치갱신 메시지를 보낸다.(420단계) 다시 말해, 이동 에이전트는 *AgentID, RNaddr*를 포함한 위치 갱신 메시지(location update message)를 지역서버에게 전송한다. 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 홈 노드와 이동 에이전트가 서로 다른 지역에 있다면 홈 노드와 이동 에이전트의 거리를 고려할 때 직접 홈 노드로 위치정보 메시지를 보내는 것은 같은 지역에 있는 지역서버에게 메시지를 보내는 것보다 상대적으로 비용이 높다. 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 (즉, Inter-RM의 경우), 이동 에이전트는 홈 노드(HN)와 자기가 전에 있었던 지역 서버(previous RS)와 현재 있는 지역서버(Current RS)에게 위치갱신 메시지를 보내게 된다.(430, 440, 450단계) 다시 말하면, 이동 에이전트는 *AgentID, RNaddr*를 포함한 위치 갱신 메시지를 현재 지역서버에게 보내고, *AgentID, NextRSaddr*를 포함한 위치 갱신 메시지를 전에 있었던 지역서버에게 보내고, *AgentID, RSaddr*를 홈 노드에게 보내게 된다. 이렇게 위치 갱신 메시지를 보냄으로써 다음과 같은 효과를 얻을 수 있다. 첫째, 홈 노드에게 메시지를 보내는 것은 홈 노드가 현재 에이전트가 있는 지역을 알 수 있게 한다. 둘째, 이전 지역서버에게 갱신 메시지를 보내는 것은 이동 에이전트의 현재 지역 위치가 홈 노드에게 알려지기 전에 이동 에이전트에게 보냈던 메시지의 전달을 보장하기 위해서이다. 마지막으로, 자기가 현재 있는 지역서버에게 메시지를 보내는 것은 이동 에이전트 자신이 현재 지역에 이동했다는 것을 알리기 위함이다.

이동 에이전트의 메시지 전달에 대해 자세히 설명한다. 메시지 전달과정은 수신 이동 에이전트(Receiving mobile agent)를 찾은 다음 해당 메시지를 수신 이동 에이전트에게 전달하는 과정을 말한다. 메시지 전달 과정에서는 이주하는 이동 에이전트에게 비동기적으로 메시지를 전달하기 위하여 블랙보드를 사용한다. 각 지역 서버는 블랙보드를 두어 자신의 지역에 있는 모든 이동 에이전트들에 대한 메시지 전달을 책임진다. 도 5는 이동 에이전트의 메시지 전달 방법을 흐름도로 도시한 것이다. 먼저 송신자(sender)는 LS를 접촉하여 홈 노드의 어드레스를 찾는다.(500단계) sender는 홈 노드에게 메시지를 보낸다.(505단계) 홈 노드는 현재 이동 에이전트가 있는 지역서버(RS)에게 메시지를 보낸다.(510단계) 상기 지역서버(RS)는 블랙보드에 메시지를 넣는다.(515단계) 그리고 나서 메시지 전달이 응급한 (emergent)지 체크한다.(520단계)

메시지 전달 과정은 크게 두 가지 방법, 반응적 메시지 전달 (Reactive message delivery :RMD)와 전극적 메시지 전달 (Proactive message delivery : PMD)로 구분된다. 반응적 메시지 전달 (RMD) 방법에서는 지역 서버가 이동 에이전트로부터 위치 갱신 메시지를 받을 때 메시지 전달 과정이 수행된다. 이와 달리, 적극적 메시지 전달(PMD) 방법에는 응급 메시지 전달을 위한 것으로 지역 서버가 메시지를 받자마자 메시지 전달 과정이 수행된다.

반응적 메시지 전달 (RMD) 방법에서의 메시지 전달과정의 절차는 다음과 같다.

1) 송신자(Sender)는 먼저 룩업 서버를 통하여 수신 이동 에이전트의 홈 노드를 알아낸 후 메시지를 홈 노드에게 전송한다.

2) 홈 노드는 수신 이동 에이전트가 있는 현재 지역 서버를 알아낸 후 메시지를 지역 서버로 보낸다.

3) 지역서버는 자신의 블랙보드에 해당 메시지를 저장한다.

4) 지역 서버는 이동 에이전트로부터 위치갱신 메시지를 받을 때, 해당 이동 에이전트에게 전달할 메시지가 블랙보드에 있는지를 검사한다. 만약 메시지가 있다면, 블랙보드에서 메시지를 꺼낸 다음 해당 이동 에이전트에게 메시지를 전달한다. 이를 보다 상세히 설명하면, 지역서버는 이동 에이전트로부터의 위치갱신 메시지를 기다리고 있다가(540단계), 위치갱신 메시지가 도착하던 상기 지역서버는 자기에게 있는 블랙보드를 체크한다.(545단계) 블랙보드에 전달할 메시지가 있는지 체크한다.(550단계) 전달할 메시지가 존재하면, 지역서버는 블랙보드로부터 메시지를 얻는다.(555단계) 그리고 나서 지역서버는 수신 이동 에이전트에게 상기 메시지를 보낸다.(560단계)

반응적 메시지 전달 방법에서는 이동 에이전트는 자신에게 전달될 메시지가 있는지 없는지를 이주할 때마다 검사할 수 있다. 다시 말하면 반응적 메시지 전달방법은 메시지 전달의 주기성을 제공한다. 또한 메시지를 블랙보드에 저장함으로써 이주중인 이동 에이전트에 대해서 비동기적으로 메시지를 전달할 수 있는 방법을 제공한다.

만약 이동 에이전트가 메시지가 도착하기 전에 위치 갱신을 수행하고 한 노드에 오랫동안 상주해 있는 경우, 응급 메시지를 전달하고자 하는 경우에는 반응적 메시지 전달 방법은 문제점이 존재한다. 다시 말해, 이동 에이전트가 다른 노드로 이주할 때까지 메시지 전달이 지연되는 문제점이 발생한다. 이러한 문제점을 해결하기 위해 적극적 메시지 전달 방법을 제안한다. 적극적 메시지 전달 방법에서는 지역 서버가 메시지를 받으면, 메시지를 블랙보드에 저장한 다음 바로 메시지를 현재 이동 에이전트가 있는 노드에게 전송을 한다. 적극적 메시지 전달 방법에서의 메시지 전달 과정은 다음과 같다.

1) - 3) 과정은 반응적 메시지 전달 방법과 같다.

4) 지역 서버는 현재 수신 이동 에이전트가 상주하고 있는 노드의 주소 (즉 *RNaddr*)를 알아낸 후, 메시지를 수신 이동 에이전트에게 전달한다.

즉, 블랙보드에 있는 메시지가 응급메시지이면, 지역서버는 엔트리에서 이동 에이전트에 대한 Rnaddr를 조사한다.(525단계) 그리고 나서 지역서버는 수신 이동 에이전트에게 메시지를 보낸다.(530단계) 메시지 전달이 성공하지 않았으면(535단계), 상기 540단계로 가서 위치갱신 메시지를 기다린다.

적극적 메시지 전달 방법은 즉각적인 메시지 전달을 제공한다. 그러나 적극적 메시지 전달 방법은 이동 에이전트가 자주 그리고 빨리 노드와 노드 사이를 이주하는 경우, 메시지 전달이 실패하게 되어 메시지를 다시 재전송해야 하는 경우가 자주 발생한다. 따라서 메시지 전달 비용이 증가하게 되는 결과를 초래한다. 이런 경우에는 반응적 메시지 전달 방법이 적극적 메시지 전달 방법보다 더 알맞다.

블랙보드를 기반으로 메시지 전달 방법에 대한 전체 과정은 도 2에서 갈색 실선으로 표현되어 있다. 도 5에서 적극적 메시지 전달 방법에서 메시지 전달이 실패하면 메시지를 재전송하기 않고 반응적 메시지 전달 방법이 수행된다. 왜냐하면 이동 에이전트가 위치 갱신 메시지를 지역서버에게 전송한 후 정확한 위치를 기반으로 메시지를 전달하는 것이 메시지 전달 비용을 감소시키기 때문이다.

한편, 이동 에이전트의 복제와 부모-자식 이동 에이전트에 대해 설명하기로 한다. HB 프로토콜은 복제 이동 에이전트와 부모-자식 이동 에이전트에 대한 위치 관리 및 메시지 전달 방법을 제공한다. 이동 주체가 호스트인 이동 호스트 환경(Mobile IP, PCS)과는 달리, 이동 에이전트 컴퓨팅 환경의 이동주체는 소프트웨어 프로그램이 될 수 있다. 즉, 이동 에이전트는 사용자의 요구에 따라 복제될 수 있고 또한 자식 이동 에이전트를 생성할 수 있다. HB 프로토콜은 이동 에이전트의 홈 노드나 복제 및 자식 이동 에이전트를 생성한 생성 노드(Creation node : CN)에서 복제된 이동 에이전트와 부모-자식 이동 에이전트의 관련 정보를 유지한다.

만약 이동 에이전트가 연산 수행 중에 자식 이동 에이전트를 생성한다면 자식 이동 에이전트의 위치는 부모 이동 에이전트의 홈 노드에도 기록되어야 한다. 만약 부모 이동 에이전트의 홈 노드가 자식 이동 에이전트에 대한 정보를 가지고 있지 않으면 부모 이동 에이전트를 생성한 홈 노드와 자식 이동 에이전트를 생성한 생성 노드사이에 아무런 관계가 성립되지 않는

다. 따라서, 만약 부모 이동 에이전트와 자식 이동 에이전트의 위치가 서로 다를 때 부모-자식 이동 에이전트들에게 보낸 메시지는 두 이동 에이전트들에게 전달되지 못하는 문제점이 발생한다. 예를 들어, 부모 이동 에이전트에게 종료 메시지를 보낸다고 한다면 종료 메시지는 부모 이동 에이전트와 자식 이동 에이전트에게 보내져야 한다. 만약 부모 이동 에이전트에게만 보내진다면 자식 이동 에이전트는 고아(orphan) 이동 에이전트가 되어 문제를 발생시킨다. 따라서 부모 이동 에이전트를 생성한 홈 노드에서는 자식 이동 에이전트에 대한 위치 관리 및 메시지 전달도 책임져야 한다. 즉, 부모 이동 에이전트의 홈 노드는 자식 이동 에이전트에 대한 위치 정보를 유지해야 한다.

또한 복제된 이동 에이전트들은 홈 노드에서 위치 정보를 관리 유지해야 한다. 즉, 복제된 이동 에이전트들을 그룹으로 묶어서 위치를 관리해야 한다. 다시 말해, 홈 노드에서 이동 에이전트가 복제된 경우에는 홈 노드에서 복제된 이동 에이전트들에 대한 위치를 관리한다. 이동 에이전트가 연산 수행 중간에서 복제된 경우 복제된 이동 에이전트들은 생성 노드에서 이동 에이전트들에 대한 위치를 관리하고 이동 에이전트를 복제한 생성 노드는 홈 노드에게 이동 에이전트가 복제되었음을 알려야 한다. 즉, 홈 노드와 생성 노드에서 복제된 이동 에이전트들에 대한 위치를 관리해야 한다. 이렇게 복제된 이동 에이전트를 홈 노드나 생성 노드에서 종합적으로 관리를 함으로써 복제 이동 에이전트가 같은 지역에서 작업을 수행하고 있을 때 복제 이동 에이전트들에게 메시지를 보내는 비용을 절감한다. 또한 복제된 이동 에이전트에게 메시지를 보낼 때 그 그룹에 속한 모든 복제 이동 에이전트들에게 메시지를 보낼 수 있다.

복제된 이동 에이전트와 부모-자식 이동 에이전트의 위치를 관리하는 기법은 크게 두 가지 기법으로 분류된다. 중앙 집중형 기법과 분산형 기법으로 구성된다.

다음으로 중앙 집중형 위치 관리 및 메시지 전달에 대해 설명하기로 한다. 중앙 집중형 기법에서는 모든 복제된 이동 에이전트들이나 부모-자식 이동 에이전트들에 대한 위치 관리 및 메시지 전달을 홈 노드에서 수행한다. 즉, 연산 중간 과정에서 생긴 복제 이동 에이전트나 자식 이동 에이전트에 대한 위치 관리는 맨 처음 이동 에이전트를 생성한 홈 노드에서 전체적으로 관리를 한다. 따라서 모든 복제 이동 에이전트와 자식 이동 에이전트는 위치 갱신 메시지를 홈 노드에게 보내게 된다. 이렇게 홈 노드가 모든 이동 에이전트들에 대한 위치를 관리하기 때문에 복제 이동 에이전트나 자식 이동에이전트에게 메시지를 보낼 때 해당 이동 에이전트들에게 한꺼번에 한 번의 과정으로 메시지를 보낼 수 있다. 홈 노드에서는 {*AgentID, CloneMAs, Parent_ChildMAs, RSaddrs*} 목록을 유지한다. *CloneMAs*는 이동 에이전트가 복제된 경우 복제 이동 에이전트들에 대한 목록 즉, 원래 이동 에이전트(Original mobile agent : OMA) 또는 복제된 이동 에이전트들(Cloned mobile agent : CLMA)을 나타낸다. *Parent_ChildMAs*는 자식 이동 에이전트를 생성한 경우 부모-자식 이동 에이전트들에 대한 목록, 즉 부모 에이전트 (Parent mobile agent ; PMA) 또는 자식 이동 에이전트 (Child mobile agent : CHMA)를 나타낸다. *RSaddrs*는 O-CLMAs(OMA와 CLMA) 또는 P-CHMAs(PMA와 CHMA)들에 대한 각각의 지역서버의 주소를 나타낸다.

생성 과정에서는 CLMAs와 CHMA는 생성 메시지 *AgentID, HNaddr*을 RS에게 보낸다. RS는 다시 이 생성 메시지를 LS에게 보내게 된다. 또한 CLMAs와 CHMA는 HN에게 생성 메시지 *AgentID, OMA_ID or PMA_ID, RSaddr*를 보내게 된다. 여기서 *OMA_ID*와 *PMA_ID*는 각각 CLMA와 CHMA에 대한 원래 이동 에이전트의 식별자와 부모 에이전트에 대한 식별자를 나타낸다.

이주 과정 중 Intra-RM에서는 CLMAs와 CHMA는 위치 갱신 메시지 *AgentID, RNaddr*를 RS에게 보내게 된다. Inter-RM의 경우에는 CLMAs와 CHMA는 위치 갱신 메시지 *AgentID, RSaddr*를 HN에게 보내고, *AgentID, NextRSaddr*를 이전 RS에게 보내고, *AgentID, RNaddr*를 현재 RS에게 보낸다.

메시지 전달 과정에서는 홈 노드에서 유지관리하고 있는 모든 O-CLMAs와 P-CHMAs에 대한 지역 서버들의 주소를 알아낸 다음 메시지를 복제하여 모든 지역서버들에게 보내게 된다.

분산형 위치 관리 및 메시지 전달에 대해 설명하기로 한다. 분산형 기법은 복제된 이동 에이전트들과 자식 이동 에이전트에 대한 위치 관리를 홈 노드와 중간 연산 과정에서 복제 이동 에이전트와 자식 이동 에이전트를 생성한 생성 노드(Creation node : CN)에서 위치 관리 및 메시지 전달을 수행한다. 홈 노드는 각각 복제된 이동 에이전트들과 자식 이동 에이전트를 생성한 생성 노드에 대한 위치 정보만 가지고 있고 각각 이동 에이전트 자체에 대한 위치 정보는 가지고 있지 않는다. 생성 노드가 복제 이동 에이전트와 자식 이동 에이전트에 대한 실제 위치 정보를 유지 관리한다. 따라서 모든 복제 이동 에이전트들과 부모-자식 이동 에이전트들은 위치 갱신 메시지를 생성 노드에게 보내게 된다. 또한, 중앙 집중형 방법과는 달리 생성 노드에 의해서 메시지 전달 과정이 이루어진다. 분산형 기법에서 홈 노드에서는 *AgentID, RSaddr, CNaddr* 목록을 유지 관리한다. *CNaddr*은 복제 이동 에이전트나 자식 이동 에이전트를 생성한 생성 노드의 주소를 나타낸다. 생성 노드에서는 *AgentID, CloneMAs, Parent_ChildMAs, RSaddrs* 목록을 유지관리 한다.

생성 과정에서는 CLMAs와 CHMA는 생성 메시지 *AgentID, HNaddr*을 RS에게 보낸다. RS는 다시 이 생성 메시지를 LS에게 보내게 된다. 또한 CLMAs와 CHMA는 HN에게 생성 메시지 *AgentID, OMA_ID or PMA_ID, CNaddr*를 보내게 된다.

이주 과정 중 Intra-RM에서는 CLMAs와 CHMA는 위치 갱신 메시지 *AgentID, RNaddr*를 RS에게 보내게 된다. Inter-RM의 경우에는 CLMAs와 CHMA는 위치 갱신 메시지 *AgentID, RSaddr*를 CN에게 보내고, *AgentID, NextRSaddr*를 이전 RS에게 보내고, *AgentID, RNaddr*를 현재 RS에게 보낸다.

메시지 전달 과정에서는 생성 노드에서 유지관리하고 있는 모든 O-CLMAs와 P-CHMAs에 대한 지역 서버들의 주소를 알아낸 다음 메시지를 복제하여 모든 지역서버들에게 보내게 된다.

한편, 본 발명의 실제적인 구현에 대해 설명하기로 한다. 본 발명에 의한 HB 프로토콜을 오뚜기(ODDUGI) 이동 에이전트 시스템을 기반으로 구현하였다. 상기 오뚜기 이동 에이전트 시스템은 이동 에이전트를 생성, 수행, 이주, 종료시킬 수 있는 플랫폼으로써 이동 에이전트를 개발하고 관리하는 다양한 기능을 제공한다. 또한 이동 에이전트의 안정적인 연산 수행을 위해 결함 포용 기법 및 보안 기법을 제공한다. 특히, 본 발명에 의한 위치 관리 및 메시지 전달 방법을 제공한다. 오뚜기 이동 에이전트 시스템은 JDK 1.3.1 버전을 기반으로 개발되었다 (JDK 1.3.1 버전 및 이후 버전에서 동작 가능하다).

룩업 서버, 지역 서버, 홈 노드, 각 노드에 본 발명에 따른 HB 프로토콜 알고리즘을 적용하여 구현하였다. 특히 룩업 서버, 지역 서버, 홈 노드에서 위치 관리 및 메시지 전달을 쉽게 유지 관리하기 위해 모니터링 툴을 개발하였다. 도 6은 룩업 서버, 지역 서버, 홈 노드에서 동작하는 모니터링 툴을 나타낸다.

다음으로 종래 방법과의 비교 및 성능평가는 다음과 같다. 먼저, 위치관리 및 메시지 전달 기법들에 대해 비교하기로 한다. 홈 프락시 기법, 추적 프락시 기법, 쉐도우(Shadow) 프로토콜, SPC 프로토콜과 HB 프로토콜의 위치 관리 및 메시지 전달 비용을 비교할 때, 각 프로토콜의 메시지 복잡도 비용을 계산하기 위한 메시지 비용들을 살펴보면 다음과 같다.

M_{RS} : 이동 에이전트의 현재 지역서버(*RS*)에게 보내는 메시지 비용

M_{PN} : 이동 에이전트가 이주하기 전에 있던 이전 노드(*PN*)에게 보내는 메시지 비용

M_{PRS} : 이동 에이전트가 이주하기 전에 있던 지역서버(*PRS*)에게 보내는 메시지 비용

M_{HRS} : 이동 에이전트가 생성된 홈 지역서버(*HRS*)에게 보내는 메시지 비용

M_{HN} : 이동 에이전트가 생성된 홈 노드(*HN*)에게 보내는 메시지 비용

M_{RN} : 이동 에이전트가 현재 있는 노드(*RN*)에게 보내는 메시지 비용

표 1은 같은 지역내의 이주가 m개이고 다른 지역으로의 이주가 n개일 때 각 프로토콜들의 위치 관리 및 메시지 전달 기법에 관하여 비교, 정리한 것이다.

[표 1]

구 분	Home- Proxy 기법	Follower- Proxy 기법	Shadow 프로토콜	SPC 프로토콜	HB 프로토콜
위치갱신 메시지 비용	$M_{HN}(m+n)$	없음	$j(0\le j\le M_{HN}\times(m+n))$	$((M_{RS}+M_{PN})\times m)+((M_{RS}+M_{PN}+M_{PRS}+M_{HRS})\times n)$	$(M_{RS}\times m)+((M_{RS}+M_{PRS}+M_{HRN})\times n)$
메시지 전달 비용(메시지 전달과정이 성공한 경우)	$M_{HN}+M_{RN}$	$M_{HN}+M_{PN}(m+n-1)+M_{RN}$	$k(M_{HN}+M_{RN}\le k\le M_{HN}+M_{PN}(m+n-1)+M_{RN})$	$(M_{HRS}+M_{RS}+M_{RN})$	$(M_{HN}+M_{RS}+M_{RN})$
메시지 전달 비용(메시지 전달과정이 실패한 경우)	$M_{HN}+(h\times M_{PN})+M_{RN}(h\ge 1)$	$M_{HN}+M_{PN}(m+n-1)+M_{RN}$	$k(M_{HN}+M_{RN}+(M_{PN}\times l)\le k\le M_{HN}+M_{PN}(m+n-1)+M_{RN})$	IntraRegion :$(M_{HRS}+M_{RS}+(M_{PRN}\times l)+M_{RN})$ $(l\ge 1)$ InterRegion : $M_{HRS}+M_{PRS}+M_{RS}+(M_{PRN}\times l)+M_{RN}(l\ge 1)$	IntraRegion : $M_{HN}+M_{RS}+M_{RN}$ InterRegion: $M_{HN}+M_{PRS}+M_{RS}+M_{RN}$
추적 문제 해결	△	×	×	×	○
복제 이동 에이전트에 대한 위치관리 및 메시지 전달	×	×	×	×	○
부모-자식 이동 에이전트에 대한 위치 관리 및 메시지 전달	×	×	△	×	○

홈 프락시 기법에서 위치 갱신 메시지 비용은 $M_{HN}(m+n)$와 같다. 메시지 전달 과정이 성공한 경우 메시지 전달 비용은 $(M_{HN}+M_{RN})$과 같다. 메시지 전달 과정이 실패한 경우 메시지 전달 비용은 $(M_{HN}+(h\times M_{PN})+M_{PN})$ 비용이 든다. 여기서 h는 이동 에이전트가 이주하기 전에 있던 이전 노드(PN)들의 수를 가리킨다. 홈 프락시 기법은 추적 문제가 발생은 하지만 추적 프락시 기법에서 발생하는 것만큼 길게 발생하지 않는다. 왜냐하면 이동 에이전트가 다른 노드로 이주할 때 항상 홈 프락시에게 위치 갱신 메시지를 보내기 때문에 그때까지 기다리다가 메시지를 전달하면 추적 문제는 해결된다. 홈 프락시 기법은 복제 이동 에이전트와 자식 이동 에이전트에 대한 위치 관리 및 메시지 전달에 대해서는 다루지 않고 있다.

추적 프락시 기법에서는 이동 에이전트가 이동할 때마다 기존 노드에 프락시를 두기때문에 위치 갱신 메시지 비용은 들지 않는다. 하지만, 메시지를 전달할 때에 프락시 경로를 따라 메시지를 계속 포워딩 하게 되므로 메시지 비용은 $M_{HN}+M_{PN}(m+n-1)+M_{RN}$과 같다. 추적 프락시 기법은 프락시 경로를 따라 메시지를 계속 포워딩하기 때문에 추적 문제를 해결하기가 힘들다. 추적 프락시 기법도 복제 이동 에이전트와 자식 이동 에이전트에 대한 위치 관리 및 메시지 전달 기법에 대해서는 다루지 않는다.

쉐도우(Shadow) 프로토콜[6, 20]에서 발생하는 위치 갱신 메시지 비용은 이동 에이전트가 가지고 있는 생존시간(ttl)에 따라 다르다. 쉐도우 프로토콜에서는 이동 에이전트의 생존시간이 0일 될 때 자신의 쉐도우에게 위치 갱신 메시지를 보내게 된다. 따라서 생존시간에 따라 최소 0부터 $M_{HN}(m+n)$까지의 위치 갱신 메시지 비용이 든다. 메시지 전달 비용도 생존시간에 따라 다르다. 쉐도우 프로토콜은 이동 에이전트의 생존시간(ttl)이 남아 있는 상태에서는 프락시 기법과 비슷하고 생존시간이 0이 되면 홈 프락시 기법과 비슷하여 최소 $(M_{HN}+M_{RN})$부터 최대 $M_{HN}+M_{PN}(m+n-1)+M_{RN}$까지의 메시지 전달 비용이 든다. 메시지 전달 과정이 실패를 하면 프락시 경로를 따라 계속해서 메시지를 포워딩 함으로 인해 최소 $M_{HN}+M_{RN}+(M_{PN}\times 1)$부터 최대 $M_{HN}+M_{PN}(m+n-1)+M_{RN}$까지 비용이 든다. 여기서 1은 메시지 전달을 실패한 노드다음부터 현재 이동 에이전트가 위치한 노드까지의 이동 에이전트가 이동한 노드 수를 나타낸다. 쉐도우 프로토콜은 프락시 방법을 사용하므로 추적 문제를 해결할 수가 없다. 쉐도우 프로토콜에서 부모 에이전트가 자식 이동 에이전트를 생성하면 자식 이동 에이전트는 부모 에이전트의 쉐도우에 의존 관계가 성립된다. 자식 이동 에이전트가 다른 노드로 이주할 때 자식 이동 에이전트는 자신의 생존시간(ttl)이 남아 있는 상태에서는 현 노드에 에이전트 프락시(Agent Proxy)를 남겨놓으나 생존시간이 경과하면 부모 이동 에이전트의 그림자에게 자신의 위치 정보를 보내게 된다. 그러나, Shadow 프로토콜에서는 고아(Orphan) 발견에 초점을 맞추고 있고 위치 관리 및 메시지 전달에 대한 자세한 내용은 다루지 않고 있으며 또한 복제 이동 에이전트에 대한 위치 관리 및 메시지 전달에 대해서도 다루지 않고 있다.

SPC 프로토콜과 HB 프로토콜의 위치 갱신 메시지 복잡도 비용을 비교하면, SPC 프로토콜은 같은 지역내에서의 이주 과정에서는 이동 에이전트의 현재 지역 서버와 이동 에이전트가 이주하기 전에 있던 전 노드에게 위치 갱신 메시지를 보내게 된다. 다른 지역으로의 이주과정에서는 이동 에이전트가 이주한 현재 지역 서버, 이동 에이전트가 이주하기 전에 있던 이전 노드, 이동 에이전트가 이주하기 전에 있던 지역 서버, 그리고 이동 에이전트가 생성된 홈 지역 서버에게 위치 갱신 메시지를 보내게 된다. 즉, SPC 프로토콜의 위치 갱신 메시지 복잡도는 $((M_{RS}+M_{PN})\times m)+((M_{RS}+M_{PN}+M_{PRS}+M_{HRS})\times n)$ 와 같고, HB 프로토콜의 위치 갱신 메시지 복잡도는 $(M_{RS}\times m)+((M_{RS}+M_{PRS}+M_{HN})\times n)$ 와 같다.

이동 에이전트 메시지 전달과정에서 SPC 프로토콜과 HB 프로토콜의 메시지 비용에 대해서 살펴보면 다음과 같다. SPC 프로토콜은 홈 지역 서버에게 메시지를 전달을 한다. 홈 지역 서버는 메시지를 현재 이동 에이전트가 있는 지역 서버에게 전달해 주고 지역 서버는 메시지를 현재 이동 에이전트가 있는 노드에게 전달한다. HB 프로토콜은 룩업 서버를 통해 이동 에이전트의 홈 노드에게 메시지를 보내게 된다. 홈 노드는 메시지를 현재 이동 에이전트가 있는 지역 서버에게 메시지를 전달을 하게 되고 지역 서버는 이 메시지를 블랙보드에 저장하게 된다. 블랙보드에 저장된 메시지는 이동 에이전트가 다음번 위치 갱신 메시지를 보낼 때 이동 에이전트에게 전달된다. 따라서, SPC 프로토콜의 메시지 전달 비용은 $(M_{HRS}+M_{RS}+M_{RN})$ 와 같다. HB 프로토콜의 메시지 전달 비용은 $(M_{HN}+M_{RS}+M_{RN})$ 와 같다.

이동 에이전트 메시지 전달과정이 실패했을 때 추적 프락시 방법을 사용하는 SPC 프로토콜과 블랙보드를 사용하는 HB 프로토콜에 대해서 각각의 메시지 비용에 대해서 살펴보자. 먼저, 같은 지역내에서의 이주과정에서 메시지 전달이 실패했을 때 추적 프락시를 사용하는 SPC 프로토콜은 지역 서버에 저장되어 있는 잘못된 이동에이전트 위치에서부터 이동 에이전트가 실제로 현재 있는 위치까지 이동 에이전트가 방문한 노드의 개수(I, $I\geq 1$)만큼 계속 반복하여 메시지를 보내야 하지만, 블랙보드 기법을 사용하는 HB 프로토콜은 지역 서버의 블랙보드에 메시지를 보낸 다음 이동 에이전트가 다음번 위치갱신 메시지를 등록할 때까지 기다린 후에 메시지를 현재 이동 에이전트가 있는 노드에게 전달하게 되므로 같은 지역내에서의 이주과정동안 발생하는 메시지 전달은 실패하지 않는다. 같은 지역내에서의 이주과정에서 메시지 전달이 실패했을 때 SPC 프로토콜의 메시지 전달 비용은 $(M_{HRS}+M_{RS}+(M_{PRN}\times I)+M_{RN})$ 와 같다. HB 프로토콜의 메시지 전달 비용은 $(M_{HN}+M_{RS}+M_{RN})$ 와 같다. 다른 지역으로의 이주과정에서 메시지 전달이 실패했을 때 추적 프락시를 사용하는 SPC 프로토콜은 이동 에이전트가 전에 이주했던 지역 서버에서 현재 이동 에이전트가 있는 지역서버에게 메시지를 보낸 다음 현재 지역 서버에서 이동 에이전트가 이주했던 노드들의 개수(I)만큼 계속 반복하여 메시지를 포워딩하게 되지만, 블랙보드를 사용하는 HB 프로토콜 기법은 이동 에이전트가 전에 이주했던 지역 서버에서 현재 지역 서버로 메시지를 보낸 다음 다음번 위치갱신 메시지를 등록할 때까지 기다린 후에 메시지를 현재 이동 에이전트가 있는 노드에게 전달하게 된다. 다른 지역으로의 이주에서 메시지 전달이 실패했을 때 SPC 프로토콜의 메시지 전달 비용은 $(M_{HRS}+M_{PRS}+M_{RS}+(M_{PRN}\times I)+M_{RN})$ 와 같다. HB 프로토콜의 메시지 전달 비용은 $(M_{HN}+M_{PRS}+M_{RS}+M_{RN})$ 와 같다.

한편, 성능평가는 다음과 같다. 위치 관리 및 메시지 전달 기법의 성능 평가를 위해 성능 평가 기준 매개변수를 다음과 같이 고려한다. 여기서, C_U 는 이동 에이전트가 이주과정에 있을 때 발생하는 위치 갱신 비용이고 C_D 는 메시지 전달과정에서 발생하는 메시지 전달 비용이다.

위치 관리 오버헤드 O_U : O_U는 이동 에이전트가 위치 갱신 메시지를 보낼 때 발생하는 비용, C_U 의 네트워크 지연시간

메시지 전달 오버헤드 O_D : O_D는 이동 에이전트에게 메시지를 보낼 때 발생하는 비용, C_D 의 네트워크 지연시간

가용성(Availability) α: 시스템이 임의의 순간에 정확하게 동작할 확률을 가용성이라 한다. 가용성은 다음 수학식 1에 의해 계산된다.

[수학식 1]

$$\alpha = \frac{MTTF}{MTTF + MTTR}$$

여기서, $MTTF$는 Mean Time To Failure를 나타내고, $MTTR$는 Mean Time To Repair를 나타낸다.

위치 관리 및 메시지 전달 오버헤드는 다음과 같다. 각 위치 관리 및 메시지 전달 기법을 수행하는데 전체 메시지 비용 C은 이동 에이전트가 이주과정에 있을 때 발생하는 위치 갱신 비용 C_U과 메시지 전달과정에서 발생하는 메시지 전달 비용 C_D으로 구성된다. 이동 에이전트의 전체 이주 횟수가 i일 때 같은 지역내의 이주 확률이 p_{sr}이고 다른 지역으로의 이주 확률이 p_{dr}라고 가정하고, 이동 에이전트가 이동할 때 홈 지역 서버내에서 이주할 확률을 p_h라고 가정한다. 여기서, $p_h \leq p_{sr}$이며, $p_{sr} + p_{dr} = 1$이다.

각 기법의 C_U에 대해서 살펴보면 다음과 같다.

[수학식 2]

$$C_U^{HP} = (M_{lHN} \times (p_h \times i)) + (M_{rHN} \times ((1 - p_h) \times i))$$

$$C_U^{FP} = 0$$

$$C_U^{S} = 0 \ or \ (M_{lHN} \times (p_h \times i)) + (M_{rHN} \times ((1 - p_h) \times i))$$

$$C_U^{SPC} = ((M_{lRS} + M_{lPN}) \times (p_{sr} \times i)) + ((M_{lRS} + M_{rPN} + M_{rPRS} + M_{rHRS}) \times (p_{dr} \times i))$$

$$C_U^{HB} = (M_{lRS} \times (p_{sr} \times i)) + ((M_{lRS} + M_{rPRS} + M_{rHN}) \times (p_{dr} \times i))$$

각 메시지 비용에 l과 r이 의미하는 것은 l은 같은 지역내에서 메시지를 보낸다는 것을 의미하고, r은 같은 지역을 벗어난 다른 지역으로 메시지를 보낸다는 것을 의미한다. 예를 들어, M_{lPN}은 같은 지역에 있는 이전 노드에게 메시지를 보내는 것을 의미하고 M_{rPN}은 다른 지역에 있는 이전 노드에게 메시지를 보내는 것을 의미한다. C_U^S는 이동 에이전트의 생존시간에 따라 최소 0부터 최대 $M_{lHN} \times (p_h \times i) + M_{rHN} \times (1 - p_h) \times i)$ 만큼의 범위를 갖는다.

이동 에이전트에게 메시지를 전송하려는 상대 이동 에이전트는 홈 지역 서버와 다른 지역에 있고, 현재 이동 에이전트의 위치는 현재 홈 지역서버와 다른 곳이 있다고 가정할 때, 각 기법의 C_D에 대해서 살펴보면 다음과 같다. 여기서, C_D^{FP}와 C_D^S에서 M_{RN}는 이동 에이전트의 이동 상황에 따라 M_{lRN}기거나 M_{rRN}이 된다.

[수학식 3]

$$C_D^{HP} = M_{rHN} + M_{rRN}$$

$$C_D^{FP} = M_{rHN} + (M_{lPN} \times (p_{sr} \times (i - 1))) + (M_{rPN} \times (p_{dr} \times (i - 1)) + M_{RN})$$

$$C_D^{S} = M_{rHN} + (M_{lPN} \times (p_{sr} \times (i - 1))) + (M_{rPN} \times (p_{dr} \times (i - 1)) + M_{RN}) \ or \ \ M_{rHN} + M_{rRN}$$

$$C_D^{SPC} = M_{rHRS} + M_{rRS} + M_{lRN}$$

$$C_D^{HB} = M_{rHN} + M_{rRS} + M_{lRN}$$

LAN의 메시지 지연시간과 인터넷의 메시지 지연시간은 상당한 차이가 있다. 10 ~ 1000Mbps의 대역폭을 갖는 LAN에서 지연시간(latency)은 보통 1 ~ 10ms를 갖고 0.010 ~ 600Mbps의 대역폭을 갖는 WAN이나 0.010 ~ 2Mbps 대역폭을 갖는 인터넷 환경에서의 지연시간은 100 ~ 500ms를 갖는다. 이런 환경에서 p_{sr} = 0.2, p_{dr}=0.8 인 경우, 위치 관리 오버헤드와 메시지 전달 오버헤드를 살펴보면 다음과 같다. 도 7에서는 각 기법의 위치관리 오버헤드를 나타내고 있고 도 8은 각 기법의 메시지 전달 오버헤드를 나타내고 있다.

위치 관리 및 메시지 전달 기법의 오버헤드에 대한 성능 평가를 위해 다음과 같은 세 개의 상황을 구성한다. 첫 번째 상황은 같은 지역내에서의 이주가 다른 지역으로의 이주보다 많은 경우 즉 $p_{sr} > p_{dr}$인 경우, 두 번째 상황은 같은 지역 내에서의 이주와 다른 지역으로의 이주가 같은 경우 즉 $p_{sr} = p_{dr}$인 경우, 세 번째 상황은 같은 지역내에서의 이주가 다른 지역으로의 이주보다 적은 경우, 즉 $p_{sr} < p_{dr}$인 경우에 대하여 각 성능 평가 기준 매개변수에 대한 각 기법들의 성능평가한 결과는 다음과 같다.

도 9, 10, 11에서 볼 수 있듯이 이동 에이전트의 위치관리 및 메시지 전달 오버헤드는 *psr*> *pdr* 인 경우보다는 *psr*< *pdr* 인 경우 더 크게 나타난다. 즉, 같은 지역내에서의 이주가 많을수록 이동 에이전트의 위치 관리 및 메시지 전달 오버헤드는 감소하게 된다. 또한, 다중 지역 이동 에이전트 컴퓨팅 환경에 맞게 설계된 SPC 프로토콜과 HB 프로토콜을 비교해 보면 HB 프로토콜이 SPC 프로토콜보다 적은 위치관리 및 메시지 전달 오버헤드를 가짐을 알 수 있다.

한편, 가용성은 다음과 같다. 가용성은 *MTTR*와 *MTTR*에 크게 의존한다. *MTTF = 1/λ*로 정의된다. 여기서 λ는 시스템의 고장 비율(failure rate)을 나타낸다. λn 은 각 노드의 고장 비율을 나타내고 λc 은 통신 채널의 고장 비율을 나타낸다. 가용성 α는 위치 관리 과정에서 발생하는 위치관리 가용성 α*U* 과 메시지 전달 과정에서 발생하는 메시지 전달 가용성 α*D*으로 구분된다. 이동 에이전트가 이주한 횟수를 *i*라하고, *MTTR* = 0.5 hour이고, λn 과 λc 을 각각 0.02/hour 라고 가정할 때, 위치 관리 및 메시지 전달 기법들의 가용성은 다음과 같다.

홈 프락시 기법의 가용성을 살펴보면 다음과 같다.

[수학식 4]

$$\alpha_U^{HP} = \frac{\frac{1}{\lambda_n + \lambda_c}}{\frac{1}{\lambda_n + \lambda_c} + 0.5} = \frac{1}{1 + 0.5(\lambda_n + \lambda_c)}$$

$$\alpha_D^{HP} = \frac{\frac{1}{2\lambda_n + 2\lambda_c}}{\frac{1}{2\lambda_n + 2\lambda_c} + 0.5} = \frac{1}{1 + (\lambda_n + \lambda_c)}$$

추적 프락시 기법의 가용성을 살펴보면 다음과 같다.

[수학식 5]

$$\alpha_U^{FP} = \frac{\frac{1}{\lambda_n}}{\frac{1}{\lambda_n} + 0.5} = \frac{1}{1 + 0.5\lambda_n}$$

$$\alpha_D^{FP} = \frac{\frac{1}{(i+1)\lambda_n + (i+1)\lambda_c}}{\frac{1}{(i+1)\lambda_n + (i+1)\lambda_c} + 0.5} = \frac{1}{1 + 0.5(i+1)(\lambda_n + \lambda_c)}$$

쉐도우 기법의 가용성은 이동 에이전트의 생존시간에 따라 다르다. 최소 홈 프락시 가용성부터 최대 추적 프락시 가용성을 갖는다. 다음 식은 쉐도우 기법의 평균 가용성을 나타낸다.

[수학식 6]

$$\alpha_U^S = \frac{1}{i}\sum_{k=1}^{i} \frac{1}{1 + 0.5\lambda_n}\left(i - \left\lfloor \frac{i}{k} \right\rfloor\right) + \frac{1}{1 + 0.5(\lambda_n + \lambda_c)}\left\lfloor \frac{i}{k} \right\rfloor$$

$$\alpha_D^S = \frac{1}{1 + \lambda_n + \lambda_c} \text{ or } \frac{1}{i}\sum_{k=1}^{i} \frac{1}{1 + 0.5\{(k+1)(\lambda_n + \lambda_c)\}}$$

SPC 프로토콜의 가용성은 같은 지역내에서의 이주 확률이 *psr* 이고 다른 지역으로의 이주 확률이 *pdr*일 때 다음과 같다.

[수학식 7]

$$\alpha_U^{SPC} = \frac{p_{sr}\frac{1}{2\lambda_n + 2\lambda_c} + p_{dr}\frac{1}{4\lambda_n + 4\lambda_c}}{p_{sr}\frac{1}{2\lambda_n + 2\lambda_c} + p_{dr}\frac{1}{4\lambda_n + 4\lambda_c} + 0.5} = \frac{p_{sr} + 1}{p_{sr} + 1 + 2(\lambda_n + \lambda_c)}$$

$$\alpha_D^{SPC} = \frac{\frac{1}{3\lambda_n + 3\lambda_c}}{\frac{1}{3\lambda_n + 3\lambda_c} + 0.5} = \frac{1}{1 + 1.5(\lambda_n + \lambda_c)}$$

[수학식 8]

$$\alpha_U^{HB} = \frac{p_{sr}\frac{1}{\lambda_n + \lambda_c} + p_{dr}\frac{1}{3\lambda_n + 3\lambda_c}}{p_{sr}\frac{1}{\lambda_n + \lambda_c} + p_{dr}\frac{1}{3\lambda_n + 3\lambda_c} + 0.5} = \frac{2p_{sr} + 1}{2p_{sr} + 1 + 1.5(\lambda_n + \lambda_c)}$$

$$\alpha_D^{HB} = \frac{\frac{1}{3\lambda_n + 3\lambda_c}}{\frac{1}{3\lambda_n + 3\lambda_c} + 0.5} = \frac{1}{1 + 1.5(\lambda_n + \lambda_c)}$$

HP 프로토콜의 가용성은 같은 지역내에서의 이주 확률이 *psr* 이고 다른 지역으로의 이주 확률이 *pdr*일 때 다음과 같다.

본 발명은 컴퓨터로 읽을 수 있는 기록 매체에 컴퓨터(정보 처리 기능을 갖는 장치를 모두 포함한다)가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 장치의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광데이터 저장장치 등이 있다.

본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

발명의 효과

본 발명에 의한 다중지역 이동 에이전트 컴퓨팅 환경에서의 이동 에이전트 위치관리 및 메시지 전달 방법 및 시스템에 의하면, 다중 지역 이동 에이전트 컴퓨팅 환경에서 이동 에이전트 위치관리 및 메시지 전달 시에 발생하는 위치 갱신 비용 및 메시지 전달 비용을 줄이며, 추적문제를 낮은 통신비용으로 해결한다. 또한, 복제된 이동 에이전트와 부모-자식 이동 에이전트에 대한 위치 관리 및 메시지 전달에 대해서 다뤄, 복제 이동 에이전트들과 부모-자식 이동 에이전트에 대한 메시지 전달을 보장하며 중복된 메시지 전달을 줄여 낮은 통신 비용으로 메시지를 전달한다.

도면의 간단한 설명

도 1은 다중 지역 이동 에이전트 컴퓨팅 환경을 도시한 것이다.

도 2는 본 발명에 의한 다중 지역 이동 에이전트 컴퓨팅 환경에서 HB 프로토콜을 이용한 이동 에이전트 위치관리 및 메시지 전달 시스템의 구성을 블록도로 도시한 것이다.

도 3은 이동 에이전트 생성 과정에 대한 전체적인 알고리즘을 순서도로 표현한 것이다.

도 4는 이동 에이전트 이주 과정에 대한 전체적인 알고리즘을 순서도로 표현한 것이다.

도 5는 이동 에이전트 메시지 전달 과정에 대한 전체적인 알고리즘을 순서도로 표현한 것이다.

도 6은 룩업 서버, 지역 서버, 홈 노드에서 동작하는 모니터링 툴을 나타낸다.

도 7는 각 기법의 위치관리 오버헤드를 나타내고 있다.

도 8은 각 기법의 메시지 전달 오버헤드를 나타내고 있다.

도 9는 $p_{sr} > p_{dr}$ ($p_{sr} = 0.8$, $p_{dr} = 0.2$ 인 경우, 이동 에이전트의 위치관리 및 메시지 전달 오버헤드를 그래프로 도시한 것이다.

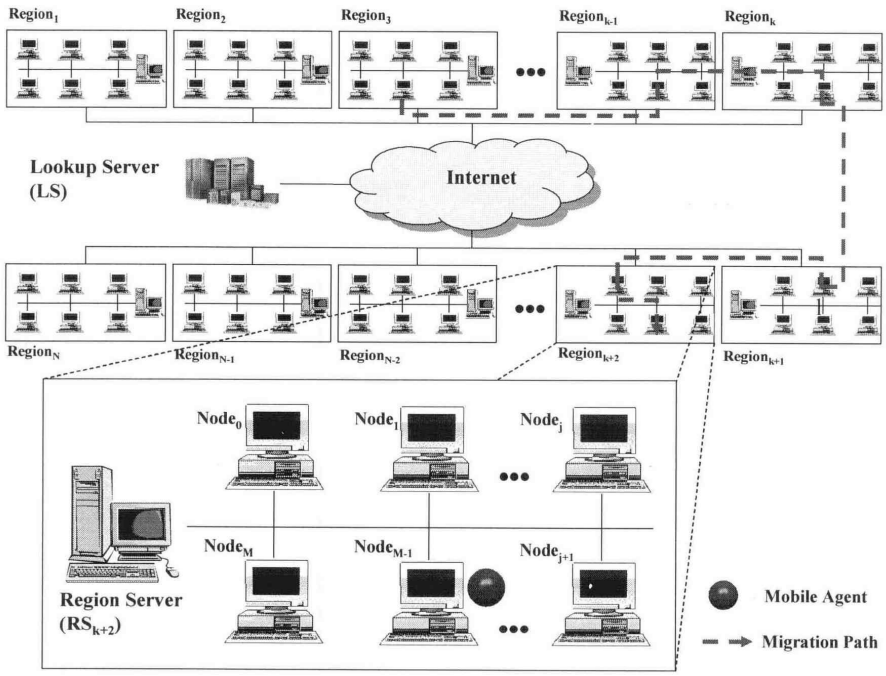
도 10는 $p_{sr} = p_{dr}$ ($p_{sr} = 0.5$, $p_{dr} = 0.5$ 인 경우, 이동 에이전트의 위치관리 및 메시지 전달 오버헤드를 그래프로 도시한 것이다.

도 11은 $p_{sr} < p_{dr}$ ($p_{sr} = 0.2$, $p_{dr} = 0.8$ 인 경우, 이동 에이전트의 위치관리 및 메시지 전달 오버헤드를 그래프로 도시한 것이다.

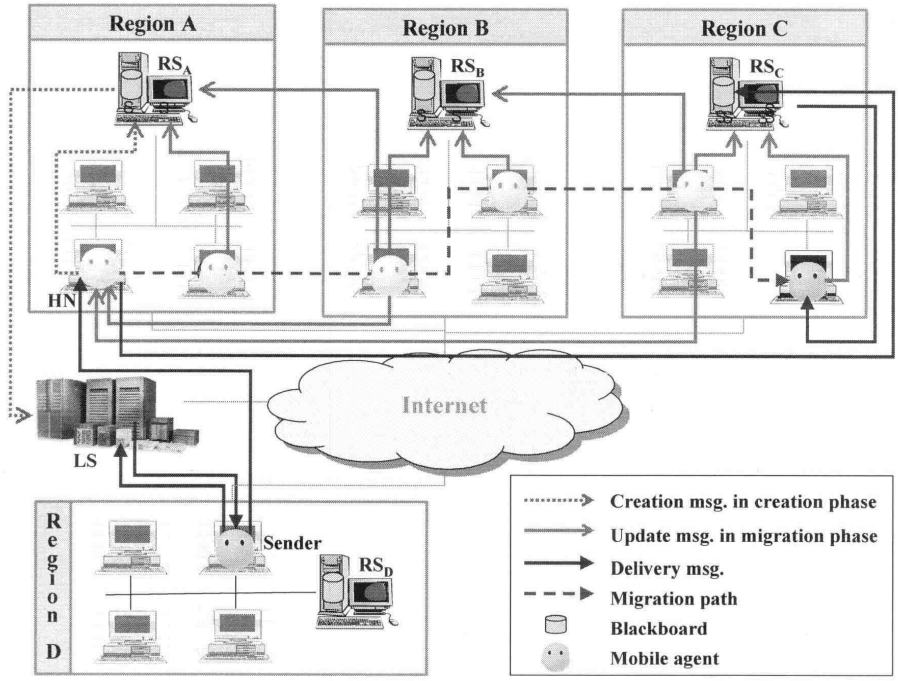
도 12는 위치 관리 및 메시지 전달 기법들의 가용성을 그래프로 도시한 것이다.

도면

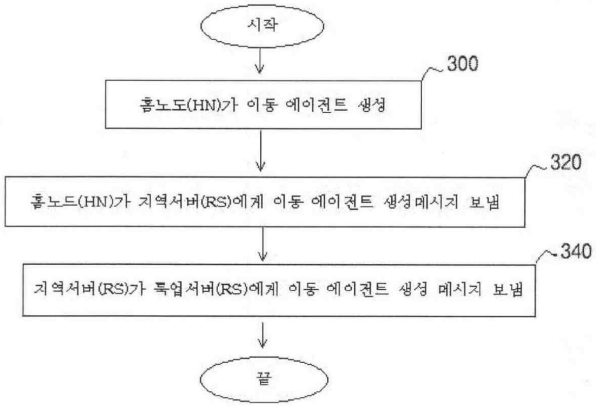
도면1



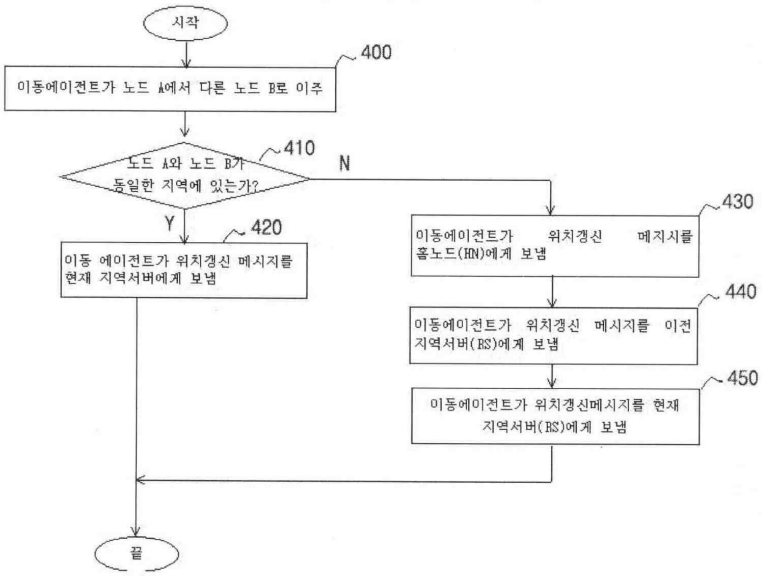
도면2


Region A
Region B
Region C
RS_A
RS_B
RS_C
HN
Internet
LS
Region D
Sender
RS_D
Creation msg. in creation phase
Update msg. in migration phase
Delivery msg.
Migration path
Blackboard
Mobile agent

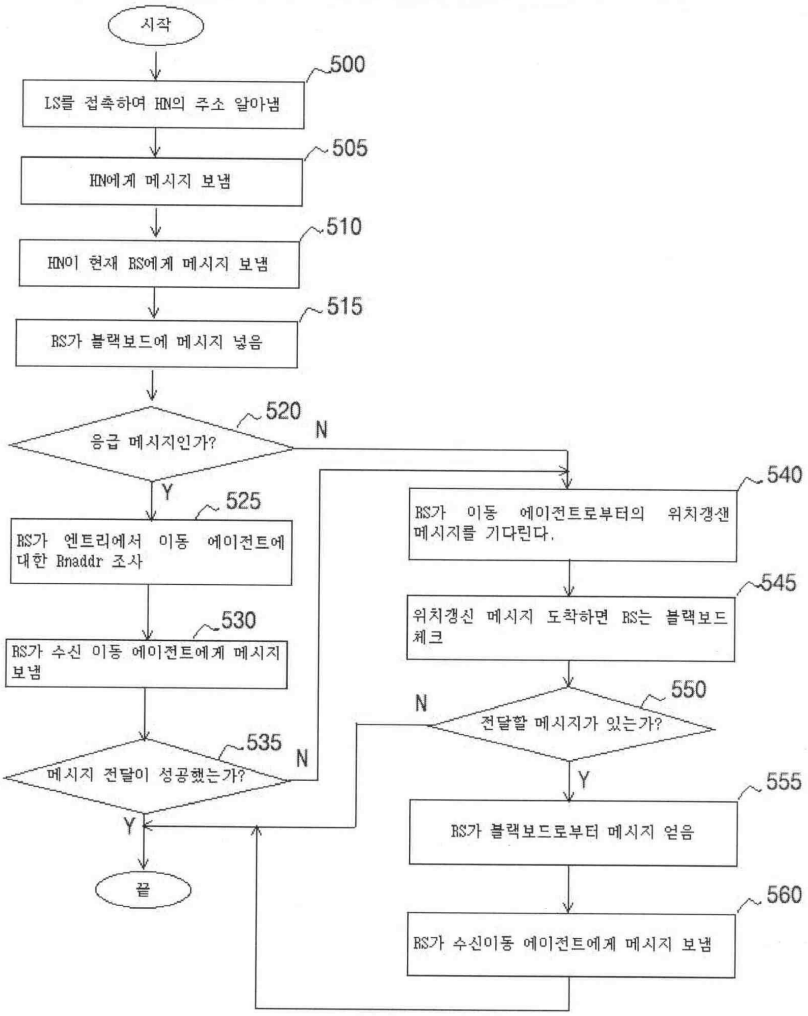
도면3


시작
300
홈노도(HN)가 이동 에이전트 생성
320
홈노드(HN)가 지역서버(RS)에게 이동 에이전트 생성메시지 보냄
340
지역서버(RS)가 특업서버(RS)에게 이동 에이전트 생성 메시지 보냄
끝

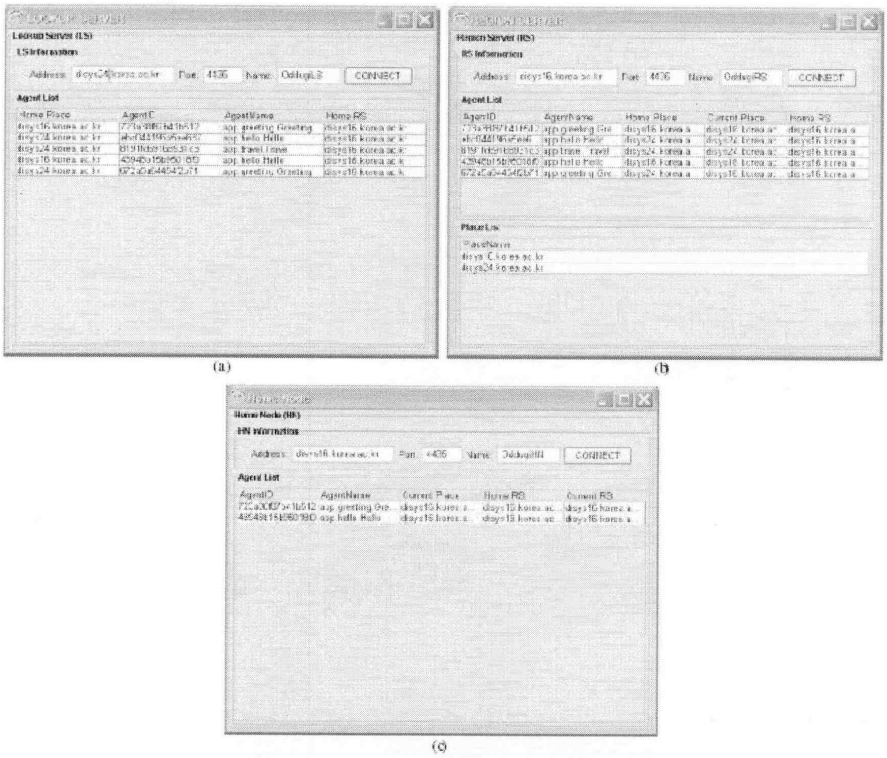
도면4


시작
400
이동에이전트가 노드 A에서 다른 노드 B로 이주
410
노드 A와 노드 B가 동일한 지역에 있는가?
N
Y
420
이동 에이전트가 위치갱신 메시지를 현재 지역서버에게 보냄
430
이동에이전트가 위치갱신 메시지를 홈노드(HN)에게 보냄
440
이동에이전트가 위치갱신 메시지를 이전 지역서버(BS)에게 보냄
450
이동에이전트가 위치갱신메시지를 현재 지역서버(BS)에게 보냄
끝

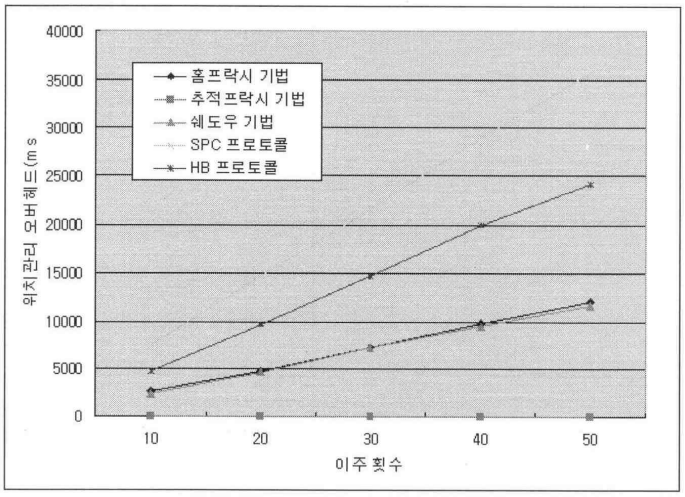
도면5


시작
500
LS를 접촉하여 HN의 주소 알아냄
505
HN에게 메시지 보냄
510
HN이 현재 BS에게 메시지 보냄
515
BS가 블랙보드에 메시지 넣음
520
응급 메시지인가?
N
Y
525
BS가 엔트리에서 이동 에이전트에 대한 Rnaddr 조사
530
BS가 수신 이동 에이전트에게 메시지 보냄
535
메시지 전달이 성공했는가?
N
Y
끝
540
BS가 이동 에이전트로부터의 위치갱신 메시지를 기다린다.
545
위치갱신 메시지 도착하면 BS는 블랙보드 체크
550
N
전달할 메시지가 있는가?
Y
555
BS가 블랙보드로부터 메시지 얻음
560
BS가 수신이동 에이전트에게 메시지 보냄

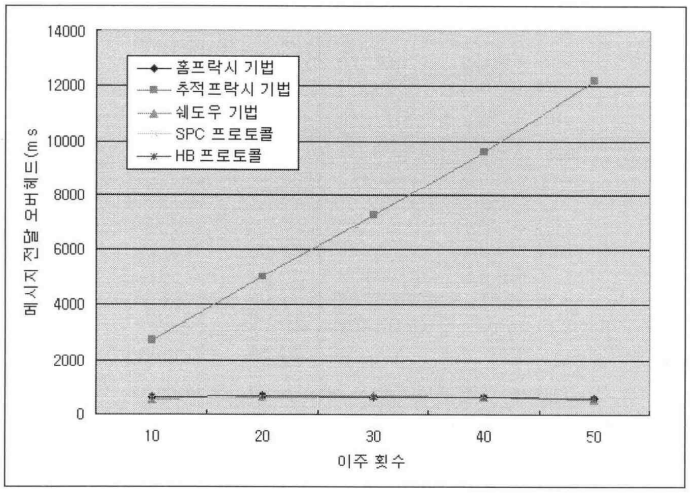
도면6



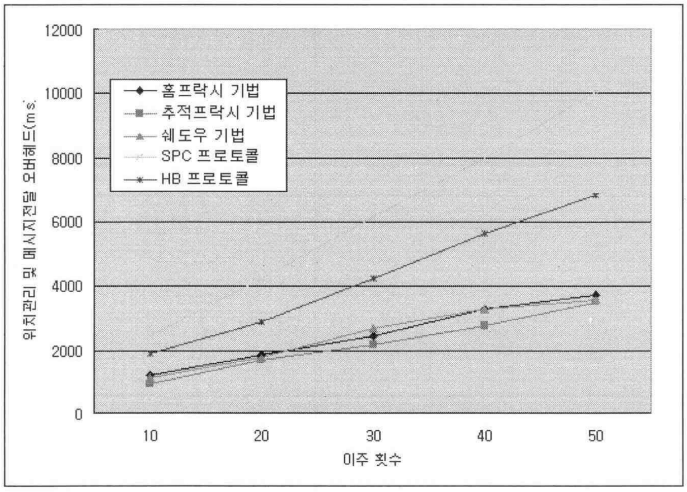
도면7


40000
35000
30000
25000
20000
15000
10000
5000
0
위치관리 오버헤드(ms
10
20
30
40
50
이주 횟수
홈 프락시 기법
추적 프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜

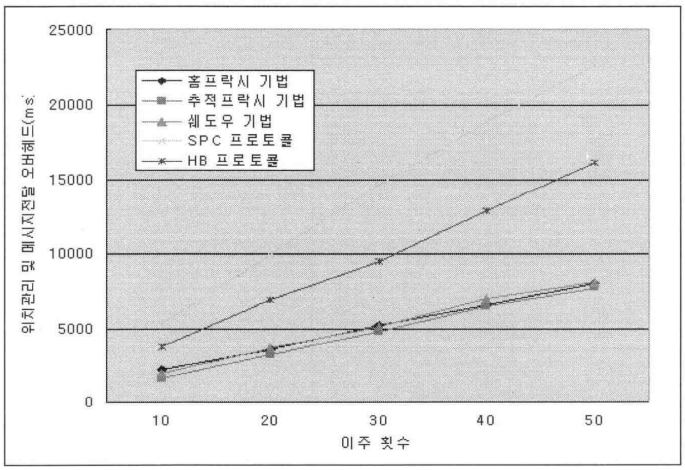
도면8


14000
12000
10000
8000
6000
4000
2000
0
메시지 전달 오버헤드(m s
홈프락시 기법
추적프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜
10
20
30
40
50
이주 횟수

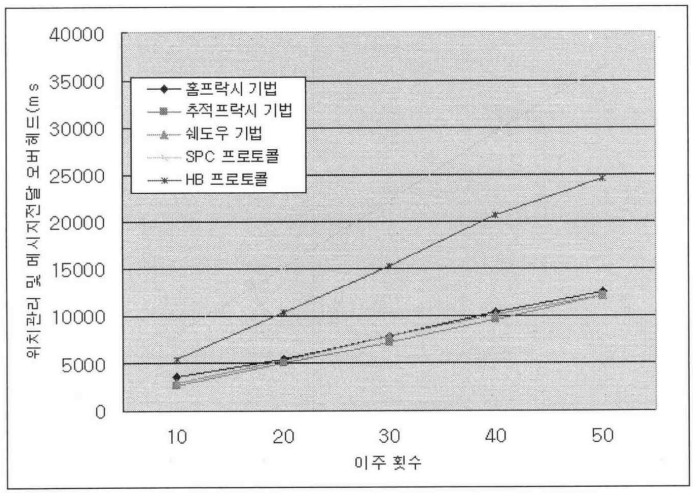
도면9


12000
10000
8000
6000
4000
2000
0
위치관리 및 메시지전달 오버헤드(m's
홈프락시 기법
추적프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜
10
20
30
40
50
이주 횟수

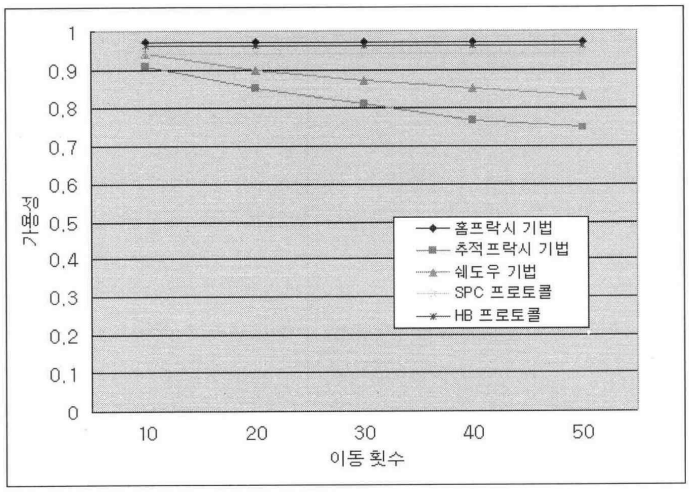
도면10


25000
20000
15000
10000
5000
0
위치관리 및 메시지전달 오버헤드(m's
홈프락시 기법
추적프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜
10
20
30
40
50
이주 횟수

도면11


40000
35000
30000
25000
20000
15000
10000
5000
0
위치관리 및 메시지전달 오버헤드(ms
홈프락시 기법
추적프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜
10
20
30
40
50
이주 횟수

도면12


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
가용성
홈프락시 기법
추적프락시 기법
쉐도우 기법
SPC 프로토콜
HB 프로토콜
10
20
30
40
50
이동 횟수