이동 에이전트의 작업 시간을 고려한 작업 노드 선택법에 대한 연구

김용호° 김영균 오길호
{kimyh°, ygkim, gilho}@cespc1.kumoh.ac.kr

A Study of a Work-Node Selection Method in Consideration for Mobile Agent's Work Time

Yong-Ho Kim° Young-Gyun Kim Gil-Ho Oh
School of Computer Engineering, Kumoh National University of Technology

요 약

이동 에이전트는 네트워크 내에서 스스로 이동하면서 사용자 또는 다른 개체 대신 행동할 수 있는 컴퓨터 프로그램을 말하며 이동 에이전트가 적용될 수 있는 응용 분야로는 전자 상거래, 정보 검색, 병렬 처리, 네트워크 관리 등이 있다. 이러한 응용 분야에서 이동 에이전트가 작업을 수행하기 위해 이동할 때 작업 시간을 고려한 후 적절한 방법으로 목표 노드를 선택한다면 전체적인 작업 처리 시간은 단축되어 시스템의 성능은 향상될 것이다. 본 논문에서는 이동 에이전트의 목표 노드 선택 방법을 정적인 노드 선택법, 동적인 노드 선택법, 단순 순회법으로 구분하고 이동 에이전트의 작업 시간에 따라 세 가지 방법을 적용하여 수행 성능을 측정하고 결과를 통해서 이동 에이전트의 작업 시간에 따라 적절한 노드 선택법을 사용할 때 전체적인 성능이 향상되는 것을 알 수 있었다.

1.서론

이동 에이전트 환경에서 작업에 참여하는 각 노드의 성능이 동일하고 부하가 일정하며 이동 에이전트가 수행하는 작업의 종류가 한정되어 있다면 시스템의 전체적인 성능에 영향을 끼치는 요소는 이동 에이전트의 효율적인 운용일 것이다.

본 논문에서는 이동 에이전트 기반의 간단한 데이터 관리 시스템을 구현하여 각 노드에서 작업을 수행하는 이동 에이전트에게 정적인 노드 선택법, 동적인 노드 선택법, 단순 순회법을 적용하여 전체적인 처리시간을 측정, 비교하였다. 반복적인 실험 결과, 이동 에이전트의 종류에 맞는 적절한 작업 선택법을 사용할 경우 시스템의 전체적인 성능이 향상되는 것을 알 수 있었다.

2.관련연구

2.1 이동 에이전트(Mobile Agent)

이동 에이전트는 네트워크 내에서 스스로 이동하면서 사용자 또는 다른 개체 대신 행동할 수 있는 컴퓨터 프로그램을 말한다. 이동 에이전트는 분산 컴퓨팅 환경에 동적으로 이식이 가능하며(portability), 사용자가 요구하는 작업을 특정한 호스트로 이동하여 수행할 수 있으며(code mobility), 다른 에이전트의 직접적인 지시나 간섭 없이도 스스로 판단하여 행동하는 자율성(autonomy)을 가진다[1,2].

2.2 연구 모델의 특성

본 논문에서는 실험을 위해 간단한 데이터 관리 시스템을 구현하였다. 각 노드들은 동일한 데이터를 보유하고 있으며 데이터를 항상 최신의 것으로 유지해야 한다고 가정한다.

본 시스템은 관리를 맡고 있는 마스터 노드와 관리를 받는 슬래이브 노드들로 구성된다. 마스터 노드는 등록된 슬래이브 노드로 이동 에이전트를 파견하여 데이터를 검사하고 필요 시 업데이트를 수행하기 위한 이동 에이전트를 노드로 파견한다. 슬래이브 노드들은 동일한 성능을 가지는 고정된 수의 컴퓨터들이며 유휴시간을 이용하여 작업을 수행하므로 이동 에이전트에 의한 작업 외에 별도의 부하가 없다고 가정한다.

3.시스템의 설계 및 구현

마스터 노드는 이동 에이전트를 이용하여 슬래이브 노드들이 보유하고 있는 데이터를 검사하고 필요 시 업데이트를 수행하기 위한 이동 에이전트를 파견한다.

3.1 데이터 관리

각각의 슬래이브 노드들이 보유하고 있는 데이터는 동일하다 데이터를 검사하는 이동 에이전트는 슬래이브 노드들을 순회하면서 데이터의 버전을 검사하며 필요할 경우 작업 에이전트가 데이터 업데이트 작업을 수행한다. 업데이트는 데이터의 버전에 따라 순차적으로 수행되어야 한다.

3.2 이동 에이전트를 위한 노드 선택법

본 논문에서는 데이터 업데이트를 수행하는 에이전트가 각 슬래이브 노드들을 순회하는 동안, 새로 추가되는 목표 노드들에 대한 고려 유무에 따라 정적인 선택법, 동적인 선택법, 단순 순회법으로 구분하였다.

3.2.1 정적인 노드 선택법

마스터 노드는 현재 시점에서 목표가 되는 모든 노드들의 주소를 작업 에이전트에게 전달한 후 파견한다. 작업 에이전트는 목표 노드들을 순회하며 작업을 수행하고 마스터 노드로 귀환한 후 새로운 목표 노드들의 주소를 전달 받는다. 이 방식은

작업 에이전트가 주어진 목표 노드들을 순회하는 동안 추가되는 새로운 목표 노드들을 고려하지 않는다.

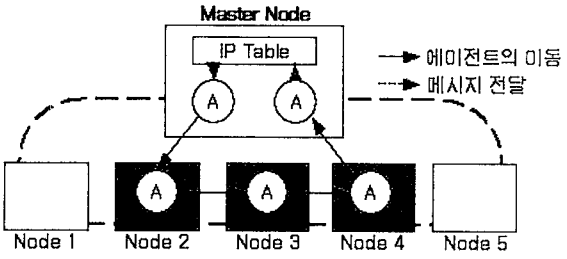

그림 1. 정적인 노드 선택법

그림1은 정적인 노드 선택법의 예를 보여준다. 에이전트 A는 마스터 노드에서 목표 노드의 주소를 모두 전달 받아 출발한다. 노드3에서 작업 중 노드5에 대한 작업 요구가 추가 발생하더라도 A는 이에 대한 고려가 없으므로 노드5로 이동하지 않는다.

3.2.2 동적인 노드 선택법

마스터 노드는 현재 시점에서 목표가 되는 노드들 중 첫 번째 노드의 주소만 작업 에이전트에게 전달하여 파견한다. 에이전트는 목표 노드로 이동하여 작업을 수행한 후 마스터 노드에게 작업 종료를 알리고 새로운 목표 노드 주소 하나를 전달 받는다. 이 방식은 작업 에이전트가 마스터 노드의 지시에 따라 목표 노드들을 순회하는 동안 새롭게 추가되는 슬래이브 노드들을 고려한다.

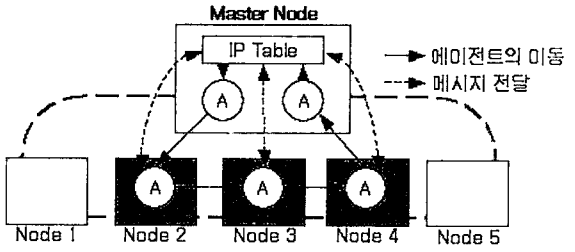

그림 2. 동적인 노드 선택법

그림2는 동적인 노드 선택법의 예를 보여준다. 에이전트 A는 최초에 노드2의 주소만 전달 받아 출발하며 작업 종료 사실을 마스터 노드에게 전달하고 다음 목표 노드의 주소를 전달 받는다. 노드3에서 작업 중 노드5에 대한 작업 요구가 추가 발생하면 A는 노드4에서 작업을 마친 후 노드5로 이동한다.

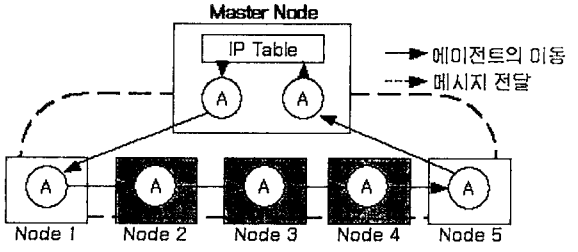

그림 3. 단순 순회법

3.2.3 단순 순회법

데이터에 대한 검사 및 업데이트 작업을 구분하지 않고 하나의 에이전트가 모든 작업을 수행하는 방법이다. 이를 위해 이동 에이전트는 관리 대상인 모든 슬래이브 노드들을 순회하면서 데이터를 검사하고 작업이 필요할 경우 업데이트 작업을 수행한다. 그림 3은 에이전트 A가 1번에서 5번 노드까지 순회하면서 데이터를 검사하고 2,3,4번 노드에서 업데이트 작업을 수행하는 과정을 보여준다.

3.3 처리할 작업의 종류

슬래이브 노드에서 수행되는 작업은 데이터를 최신의 것으로 유지하는 업데이트 작업이다. 작업을 수행하기 위한 이동 에이전트는 목표 노드에 도착 시 데이터의 헤더 정보를 검사하여 최신 여부를 조사한 후 필요할 경우 다음 버전으로 업데이트 작업을 수행한다.

3.4 구현한 시스템의 구조

JDK(Java Development Kit) 1.1.8과 IBM의 ASDK(Aglets Software Development Kit) 1.1 Beta3를 이용하여 구현하였다[3,4]. 시스템은 관리 에이전트, 검사 에이전트, 작업 에이전트로 구성되어지며 각각의 역할은 다음과 같다.

▪관리 에이전트 (Manager Agent)

마스터 노드에 정적으로 존재하며 전체 슬래이브 노드들의 IP 주소 테이블, 목표 슬래이브 노드들의 IP 주소 테이블, 각 노드들의 데이터에 대한 버전 정보 테이블을 가진다. 전체 슬래이브 노드를 순회하며 데이터의 버전을 검사하는 검사 에이전트를 생성하여 파견하고 작업이 필요한 목표 노드로 작업 에이전트를 생성하여 파견한다.

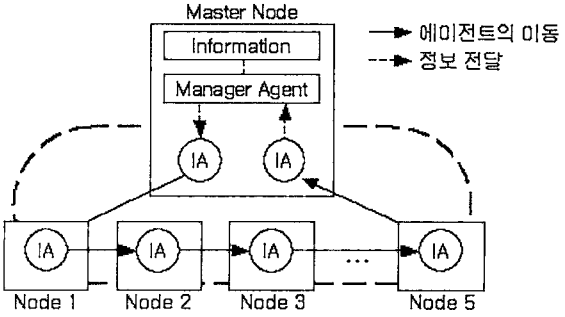

그림 4. 검사 에이전트의 동작 구조

▪검사 에이전트 (Inspection Agent)

검사 에이전트는 그림4처럼 각 슬래이브 노드들을 순회하면서 데이터의 버전을 검사하고 마스터 노드로 귀환하여 결과를 관리 에이전트에게 전달한다.

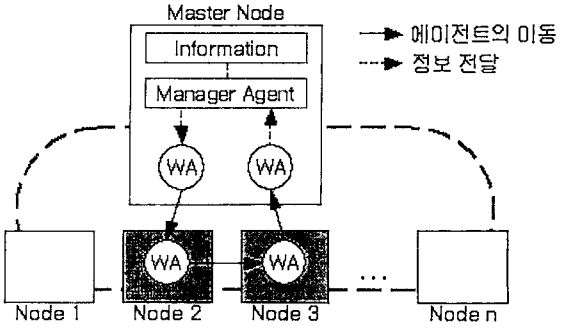

그림 5. 작업 에이전트의 동작 구조

▪작업 에이전트 (Worker Agent)

관리 에이전트가 지정한 작업 노드로 이동하면서 데이터를

업데이트한다. 정적인 노드 선택법일 경우 관리 에이전트로부터 전달 받은 목표 노드 주소 리스트를 이용하여 각각의 슬래이브 노드들을 순회하며, 동적인 노드 선택법일 경우 슬래이브 노드에서 작업을 수행한 후 관리 에이전트에게 작업 종료를 알리고 다음 목표 노드의 주소를 전달 받는다. 그림 4는 작업이 필요한 목표 노드들만 방문하는 작업 에이전트의 동작 과정을 보여준다.

4.성능 평가

성능 평가는 표1과 같은 환경에서 수행되었다. 관리 대상인 슬래이브 노드의 수를 8개로 고정하고, 작업 에이전트의 노드별 수행 시간은 검사 에이전트의 순회 주기를 기준으로 하여 작거나 비슷하거나 큰 경우로 분류하여 다섯 가지로 구분한 뒤 세가지 노드 선택법을 적용하여 전체 처리 시간을 측정하고 비교하였다. 검사 에이전트의 순회 주기는 900ms로 측정되었다.

작업 에이전트의 노드에서의 수행시간은 표2와 같다. 데이터에 대한 업데이트 작업은 세 번 수행되며 검사 에이전트가 한 번 순회할 때 작업 요구가 발생할 확률은 50%로 설정하였다.

표1. 실험 환경

마스터 노드	펜티엄3-733, 256MB	1대
슬래이브 노드	펜티엄2-350, 64MB	8대
네트워크	10Mbps Ethernet	

표2. 작업 에이전트의 노드 수행 시간

300, 600, 900, 1200, 1500	ms

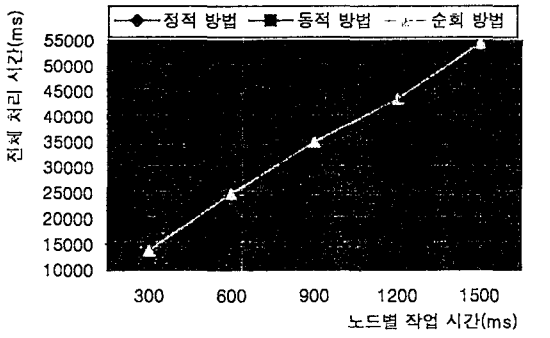

그림 6. 작업 에이전트의 노드별 수행 시간에 따른 전체 처리 시간

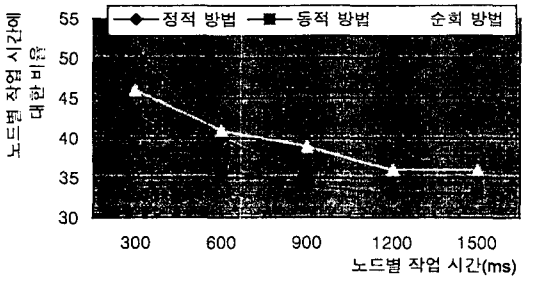

그림 7. 작업 에이전트의 노드별 수행 시간 대비 전체 처리 시간

수행 결과는 그림6,7과 같다. 결과를 보면 전체적으로 동적인 노드 선택법이 다른 두 가지 방법보다 우수한 성능을 보여주는 것을 볼 수가 있다. 노드별 수행 시간이 300ms일 경우 정적인 노드 선택법이 가장 떨어지는 성능을 보여주었다. 이는 작업 수행 시간에 비해 작업 요구가 느리게 반영되기 때문이다. 그러나 노드별 수행 시간이 600ms일 경우 정적인 노드 선택법이 단순 순회법보다 더 좋은 성능을 보이기 시작했다.

노드별 수행 시간이 검사 에이전트의 순회 주기와 같은 900ms일 때 동적인 노드 선택법과 정적인 노드 선택법 두 가지의 경우에서 가장 뛰어난 성능 향상이 있었으며 노드별 작업 시간이 900ms보다 클 경우에 세 가지 방법 모두 작업 시간이 크게 단축되지 않았다.

검사 에이전트의 순회 주기를 기준으로 결과를 정리해 보면 노드별 작업 시간이 순회 주기에 비해 50% 미만으로 작다면 오히려 구현이 간단한 단순 순회법을 적용하는 것이 가장 적절하며 50%보다 크거나 비슷할 경우는 동적인 방법이 적절하고 작업 시간이 순회 주기를 상회할 경우 상대적으로 구현이 쉬운 정적인 방법을 사용하는 것이 적절하다. 만약 전체적인 노드에서의 작업 시간이 논문 구현처럼 일정하지 않고 다양하다면 동적인 노드 선택법을 사용하는 것이 적절할 것이다.

5.결론

본 논문에서는 동일한 성능을 가지는 컴퓨터들로 구성된 이동 에이전트 환경에서 이동 에이전트의 노드에서의 수행시간을 고려한 작업 노드 선택법에 대해서 연구했다. 이를 위해 간단한 형태의 데이터 관리 시스템을 구현하였고 실험결과 이동 에이전트의 종류를 고려해서 적절한 작업 노드 선택법을 사용할 경우 시스템의 성능이 향상된다는 것을 알 수가 있었다.

이번 연구에서는 호스트의 수를 고정하고 이동 에이전트의 수행 시간만 고려하여 실험하였으나 차후에는 호스트의 수가 가변적으로 변하는 상황에서 다양한 수행 시간과 크기의 이동 에이전트를 혼합 운용하는 방향으로 연구할 계획이다.

참고문헌

[1] OMG, "Mobile Agent Facility Specification", http://www.omg.org/, pp12-13, 2000.

[2] D. Chess, C. Harrison, A. Kershenbaum, "Mobile agents: Are they a good idea?, In Mobile Object Systems: Towards the Programmable Internet", Vol. 1222 of Lecture Notes in Computer Science, Springer-Verlag, 1997.

[3] Aglet workbench, http://www.trl.ibm.com/aglets, IBM Japan, 2000.

[4] D. Lange, M. Oshima," Programming and Deploying Java Mobile Agents with Aglets" , Addison Wesley, 1998.

[5] L. Ismail, D. Hagimont," A Performance Evaluation of the Mobile Agent Paradigm" , Proceeding of the 1999 ACM SIGPLAN Conference on OO programming, Systems, Languages, and Applications, 1999.

[6] 김용호, 김영균, 오길호, " 이동 에이전트 환경에서 최적화된 작업할당 방법에 대한 연구" , 한국정보과학회 가을 학술발표논문집, 28권, pp622-624, 10월, 2001년.