

모바일 그리드 시스템의 개발 및 성능평가

김 태 경[†] · 김 희 승^{**} · 이 현 주^{**} · 정 태 명^{***}

요 약

무선 네트워크 및 이동단말기의 성능이 급속도로 발전하고 있으며, Mobile IP나 ad-hoc 기술 같은 이동성을 관리하는 기술들도 개발됨에 따라 언제 어디서든 혹은 이동 중에도 분산 컴퓨팅 같은 작업을 수행하는 요구가 크게 증가하고 있다. 이러한 무선 네트워크 환경에서 안정적인 분산 어플리케이션 수행을 위해서 모바일 그리드시스템에 대한 연구를 수행하였으며, 모바일 그리드는 자원 공유를 가능하게 하는 새로운 방식으로 센서나 모바일 단말기 혹은 무선의 다른 이동 장비들을 포함한 유선의 자원을 공유하는 방식이다. 본 논문에서는 모바일 네트워크 환경에서 그리드 서비스를 제공할 수 있도록 모바일 그리드 시스템의 구체적인 설계 및 구현을 수행하였다. 또한 실제 구현된 환경에서 응용·프로그램을 수행하여 모바일 그리드 시스템의 성능평가를 수행하였다.

키워드 : 모바일 그리드, 설계, 구현, 성능평가

The Development of Mobile Grid System and Performance Evaluation

Tae-Kyung Kim[†] · Hee-Seung Kim^{**} · Hyun-Joo Lee^{**} · Tai-Myung Chung^{***}

ABSTRACT

The performance of mobile devices and wireless networks is improved rapidly and Mobility management skills such as Mobile IP and ad-hoc technology were developed. So the requirement of processing the distributed computing is highly increased in any place and any time or in the state of movement. Therefore we studied the mobile grid system to process the distributed applications properly in wireless networks. In addition to typical computational resources, Mobile Grid brings new resources such as sensor, mobile devices or other wireless devices to distributed computing for the purpose of resource-sharing. In this paper, we design and implement the mobile grid system in wireless lan network environment for the providing the grid service. Also, we evaluate the performance of mobile grid system using the processing the distributed applications in implemented mobile grid environments.

Key Word : Mobile Grid, Design, Implementation, Performance Evaluation

1. 서 론

모바일 그리드는 슈퍼 컴퓨팅과 고속 네트워크 개념을 확장하고, 급속히 발전하고 있는 무선이동 통신망 기술을 이용하여, 그리드 서비스 이용자에게 이동성 및 편의성을 제공하는 것이다. 즉, 시간과 장소에 관계없이 그리드 서비스가 가능하도록 해주는 기술이다. 그리드는 지리학적으로 분산되어 있는 고성능 컴퓨팅 자원을 네트워크로 상호 연동하여 조직과 지역에 관계없이 사용할 수 있도록 하는 환경을 말하며, 네트워크, 통신, 연산, 정보자원을 통합하여 동일한 방식으로 연산과 데이터 관리를 위한 가상의 플랫폼을 제공한다 [1]. 모바일 그리드의 기능은 다음과 같이 정의할 수 있다.

- 그리드의 자원 모델을 무선 모바일 네트워크로 확대
- 유비쿼터스 모바일 장비를 이용하여 그리드 시스템에 접근하는 것을 허용
- 무선 센서 네트워크를 이용하여 정보를 모으는 것이 가능

여기서 무선 네트워크를 그리드에서 사용하는 이유로는 모바일 장비의 사용이 폭발적으로 증가하고 있는 것과 네트워크 성능의 향상을 들 수 있다. 아래의 <표 1>은 무선 랜 장비의 예측 현황을 나타낸 것이다.

PDA의 경우, CPU의 성능이 600Mhz에 이르고 있으며, 저장 공간도 8G급 NAND 플래시 메모리가 개발되어 사용되고 있다. 이외에도 소형 단말기를 위한 메모리 확장 방안이 연구되고 있는데, 이는 추가적인 메모리 슬롯을 만들어 소형 추가 메모리를 자유롭게 탈착이 가능하도록 하는 방식이다. 무선 네트워크의 경우에는 어디에서도 사용자가 원하는 곳에서 쉽게 이용이 가능하다는 장점 이외에도 대역폭의 성능 향상을

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성·지원사업의 연구결과로 수행되었음.

† 준 회 원 : 성균관대학교 전기전자및컴퓨터공학부 박사

** 준 회 원 : 성균관대학교 컴퓨터공학과 석사과정

*** 종신회원 : 성균관대학교 정보통신공학부 교수

논문접수 : 2005년 1월 4일, 심사완료 : 2005년 5월 23일

〈표 1〉 전 세계 무선랜 장비 예측 (단위: 천 개)

장비 명	2004	2005	2006	2007
Adapter	30,764	41,417	50,415	56,931
AP	3,919	4,851	5,837	6,555
Broadband gateway	3,365	5,550	7,941	9,472
다른 무선 랜 장비	105	132	158	179
전체	38,153	51,950	64,351	73,137

자료출처: Gartner Dataquest 2002. 7.

들 수 있는데, IEEE 802.15.3a의 기술의 경우에는 10m 거리에 110Mbps를 지원하며 4m 거리에서는 최소 200Mbps 그리고 최대 480Mbps를 지원이 가능하다. 기술적인 면에서도 모바일 단말기의 효율적인 이동을 돕기 위해 Mobile IP [2]와 애드혹 라우팅 [3]의 기술들이 개발 및 연구되고 있다.

그러나 모바일 그리드를 고려할 때 기존의 그리드와 다른 점들을 고려해야 한다. 유선의 그리드에서는 대용량의 메모리와 빠른 프로세서를 가진 강력한 자원의 성능을 제공할 수 있으며, 고성능 대역폭의 링크를 제공한다. 그리고 네트워크의 단절은 장비의 이상이나 성능 이상으로 인해 발생하나 그 경우가 거의 미비하다는 특성을 가지고 있다. 그에 비하여 무선의 그리드에서는 낮은 CPU 속도, 적은 메모리 용량, 낮은 배터리 성능, 작은 스크린 크기 등의 성능이 제한되어 있으며, 예측하지 못하는 네트워크의 단절도 일반적인 무선 통신 상태로 간주되며, 호스트들의 빠른 이동이 가능하고 대역폭과 네트워크가 제공하는 성능의 변화폭이 크다는 특성을 가지고 있다.

이러한 환경에서 효율적인 그리드 작업 수행을 위해서, 본 논문에서는 2장에서 모바일 그리드와 관련된 연구에 대해서 정리를 하였으며, 3장에서는 모바일 그리드 구현을 위한 설계 및 실제 구현에 대한 설명을 하였다. 그리고 4장에서는 프락시 기반의 모바일 그리드 시스템에 적합한 작업 할당 알고리즘을 제시하였으며, 5장에서는 모바일 그리드 구조에 대한 성능평가에 대한 결과를 기술하였고, 6장에서는 결론 및 향후 연구과제에 대하여 언급하였다.

2. 관련 연구

관련연구에서는 모바일 그리드를 구축하기 위한 연구 프로젝트들의 접근방법 및 장·단점에 대해서 분석하였다. 현재 모바일 그리드 분야에서 추진되고 있는 방식으로는 큰 분류로 프락시 기반의 클러스터 구조의 방식과 모바일 에이전트를 이용하는 두 가지 방식이 있다.

2.1 모바일 에이전트를 이용한 방식

모바일 에이전트를 이용하는 방식은 이동 단말을 이용하여 그리드 코어 하부 구조 망에 액세스 하기 위한 방안을 에이전트 기반의 접근에 기초하여 설계한 것이다 [4]. 에이전트 기반 방식의 일반적인 장점은 다음과 같다 [5].

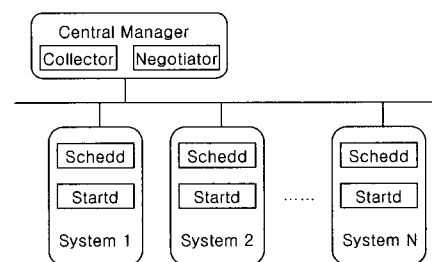
- 에이전트가 서버로 이동해서 해당되는 일련의 과정을 수행하고 그 최종결과만을 클라이언트에 돌려주기 때문에 대역폭 사용의 효율성을 가져올 수 있다.
- 이동 단말의 경우 망에서의 단절이 상대적으로 잦으며 낮은 대역폭인 경우가 많은데, 모바일 에이전트의 경우 원격에서 수행한 작업의 결과를 필터링하고 압축하여 전송이 가능하므로 무선환경에 유리하다.
- 모바일 에이전트는 이질적인 플랫폼간에 이동이 자유로우며 특정 응용에 관련된 모든 코드를 함께 가지고 다니기 때문에 분산 작업이 망 상황에 따라 동적으로 할당할 수 있다.

이러한 모바일 에이전트를 이용한 대표적인 모바일 그리드 시스템으로는 모바일 클라이언트를 위한 Condor distributed system의 확장과 SETI@Anywhere 등이 있다.

• Mobile Condor

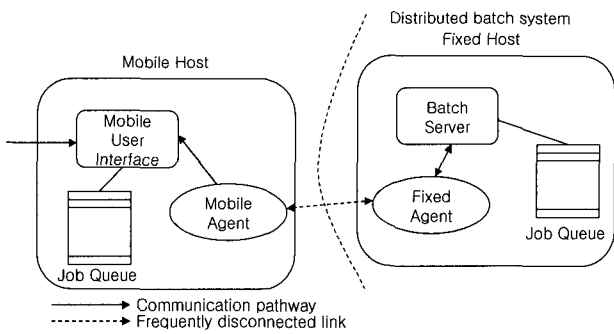
이 시스템은 이동 컴퓨팅을 사용하는 사용자가 언제 어디서나 Condor pool과의 연결 상태에 관계없이 Condor 환경과 상호 작용할 수 있도록 하기 위한 것으로, 기존 Condor 시스템에 영향을 미치지 않고 단말에만 Mobile Condor Agent를 추가한 것이다 [6]. Condor란 네트워크에 연결된 컴퓨터들간에 작업을 공유하기 위한 분산 배치 시스템으로, Condor pool 내의 모든 장비들은 "schedd"와 "startd"라는 데몬을 가지고 있다. 이중 스케줄러 데몬인 "schedd"는 주어진 장비에 제출된 모든 작업을 관리하는 역할을 담당하는데 queue manager와 scheduler로 구성된다. 또한 "startd" 데몬은 "Central Manager(CM)"가 할당한 작업을 시작 및 감시하며 종료하는 일을 하고, 키보드나 마우스 작업 또는 CPU 부하와 같이 Condor 작업을 수행하기 위해 필요한 장비의 정보를 감시한다. CM은 pool내의 모든 자원들과 작업을 감시, 관리한다. 이를 위해 pool 내의 모든 schedd와 startd 데몬들은 CM 상의 "collector"라는 데몬에게 그 정보를 보고한다. CM 내의 또 다른 데몬인 "negotiator"는 주기적으로 collector로부터 정보를 받아 기다리고 있는 작업을 위해 적절한 장비를 선택한다. Condor 시스템의 구조는 아래의 (그림 1)과 같다.

이러한 시스템에서 무선 단말로 액세스하여 작업을 제출하고 그 결과를 보기 위해서는 무선 단말의 제약성을 고려해야 한다. 이러한 상황을 고려하여 제안된 것이 mobile Condor 시스템이다. 이 시스템에서는 무선 단말에 Queue Manager와



(그림 1) Condor 시스템 구조

Mobile Condor Agent가 탑재되며 이를 통해 Condor 시스템에 제출하게 된다. 먼저 이동단말은 수행할 작업을 제출하고 이는 이동단말의 큐에 저장된다. 그 후 이동단말은 고정 장비 내의 "schedd" 프로세스와 연결을 맺게 되는데, 이때 Mobile Condor Agent는 Queue Manager를 통해 제출할 작업을 선택한 후 해당 파일을 고정 장비로 전송한다. 고정 장비는 이를 저장한 후 일반적인 Condor 동작 절차에 따라 다른 실행 장비에 제출하게 되고, 결과를 얻어 이동 단말로 알려주게 된다. 또한 이동단말의 유형에 따라 작업 제출이나 제출 후 작업의 수정, 진행 상황 검색 등의 기능이 선택적으로 구현되었다. 모바일 Condor 시스템의 구조는 아래의 (그림 2)와 같다.



(그림 2) Mobile Condor 시스템 구조

이 시스템은 모바일 장비를 기존의 시스템과 연결하여 작업의 수행을 요구할 수는 있으나, 모바일 장비를 어플리케이션을 수행하기 위한 자원으로 제공할 수 없다는 단점이 있다.

• SETI@Anywhere

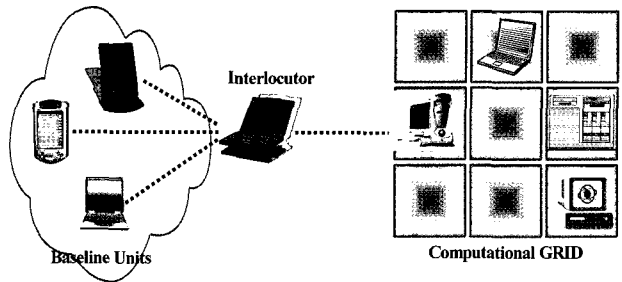
SETI@Anywhere는 기존 유선 PC를 기반으로 하여 전파 망원경의 신호를 분석하는데 사용하던 SETI@Home을 확장시켜 이동 단말기까지 참여할 수 있도록 한 데모 시스템이다. 이 시스템에서 사용한 단말은 Compaq iPaq 3670을 사용하였으며, 운영체제는 Linux를 이용하였고 GPRS 망을 통한 인터넷 연결을 가지도록 하였다. 이러한 환경에서 SETI가 iPaq 상에서 3개월 동안 성공적으로 운영됨을 보임으로써 모바일 그리드의 가능성을 검토하였다 [7]. 이외에도 모바일 에이전트를 이용한 모바일 그리드 프로젝트로는 InviNet(Invisible Network) [8], Agile computing[9] 등이 있다.

모바일 에이전트를 이용하는 방식에서 고려해야 될 점은 에이전트 자체에 대한 보안적인 문제 및 에이전트 이동에 따른 부하의 문제를 고려해야 한다. 그 이외에도 모바일 에이전트가 이동하여 수행되기 위해서는 JVM(Java Virtual Machine) 등의 환경을 구성해야 하는 등의 문제가 있다.

2.2 프락시 기반의 방식

프락시 기반의 모바일 그리드의 방식은 Leveraging Every Existing Computer out tHere Project(LEECH)[10]의 일환으로 연구된 것으로 여러 곳에 있는 PDA나 랩탑과 같이 인터넷에 연결된 무선 이동 단말의 가용성을 염두에 두고, 이런

자원들을 그리드 컴퓨팅에 사용하기 위한 방안이다. 프락시 기반의 클러스터 구조는 아래의 (그림 3)과 같다.



(그림 3) 프락시 기반의 클러스터 구조

여기서 Baseline unit은 Barely Adequate Systems Leveraging Internet Networking의 약자로, 그리드 노드로 사용될 무선 단말을 의미하며, Interlocutor는 baseline unit을 대신해서 그리드 망에 특성(처리 능력, 성능 등)을 등록하는 역할을 수행한다. 또한 글로버스[11]와 같은 미들웨어를 탑재하여, 그리드 망 내에서 자원 요청이 오면 이를 각 Baseline Unit에게 적당하게 분배하는 일을 하며, 분배되어 수행된 결과의 값을 각각의 Baseline Unit으로부터 취합하여 결과값을 요청자에게 되돌려 주는 기능을 수행한다.

이외에도 Mustafa Sanfar [12] 등이 LEECH에서 제시한 프락시 기반의 방식을 발전시켜 여러 계층의 수직적인 Hierarchical Dispatcher Model에 대해서 연구를 수행하였는데, 이 방식은 LEECH의 Interlocutor의 부하를 계층적인 모델을 이용하여 분산시키는 특징을 가지고 있다.

프락시 기반의 방식에서는 이동 단말을 대신해 interlocutor가 그리드 시스템에 참여하므로, 이동 단말의 이질성이 감추어 질 수 있으며, 이동 단말들은 자신의 정책에 따라 자유로이 그리드 세션에 참여 및 탈퇴가 가능하다. 이 때 interlocutor가 요청된 작업을 가용한 자원에 능동적으로 분배하여야 하는데, 이에 대한 추가적인 연구가 필요하다. 본 논문에서는 프락시 기반의 모바일 그리드 구조를 이용하여 모바일 그리드 시스템을 설계 및 구현하였다.

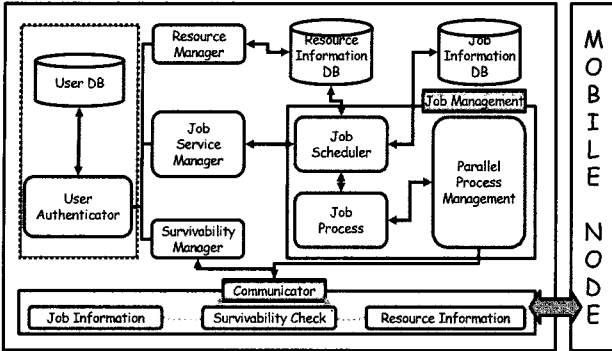
3. 모바일 그리드 시스템의 설계 및 구현

3.1 모바일 그리드 시스템의 설계

구현한 내용은 크게 모바일 그리드 구조와 모바일 그리드 프로토콜로 나눌 수 있다. 설계함에 있어서 다음과 같은 사항들을 고려하였다.

- 능동적인 자원관리: 무선환경을 고려한 능동적인 모바일 단말기의 자원의 발견 및 관리
- 서비스 품질의 보장: 신뢰성 있게 작업의 요청 및 분배 그리고 수행된 결과의 안정적인 반환
- 보안: 무선환경에 취약한 보안을 고려해서 사용자의 인증 및 암호화 알고리즘을 이용한 메시지의 보안 관리

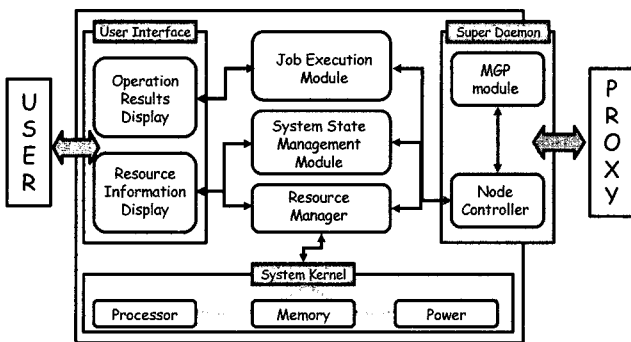
모바일 그리드 구조(Mobile Grid Architecture : MGP)는 크게 프락시 부분과 모바일 노드로 구성되며 각각의 구조는 다음과 같다.



(그림 4) 프락시의 설계

(그림 4)에서 User Authenticator는 User DB 정보에 있는 무선 단말기 사용자의 정보를 이용하여 인증 작업을 수행하게 되며, Resource Information DB는 현재 접속한 사용자의 단말기에 대한 자원 정보를 가지고 있다. Job Information DB에는 작업에 대한 정보를 가지고 있으며, 작업 분배와 수행을 위한 작업 실행파일의 절대경로 및 작업분배를 위한 작업의 실행시간에 대한 정보를 가지고 있다. Job management에서는 단말기에서 요청한 작업을 수신하여 job scheduler를 이용하여 작업 분배 정책을 수립하고, MPICH 모듈을 이용해 각 단말기들로 작업을 분배하게 된다. 분배된 작업이 다시 MPICH 모듈에 의해 수집되면 그 결과를 작업 요청을 한 단말기로 보내주게 된다. Communicator는 단말기와 Proxy간의 소켓 통신을 위한 모듈이다.

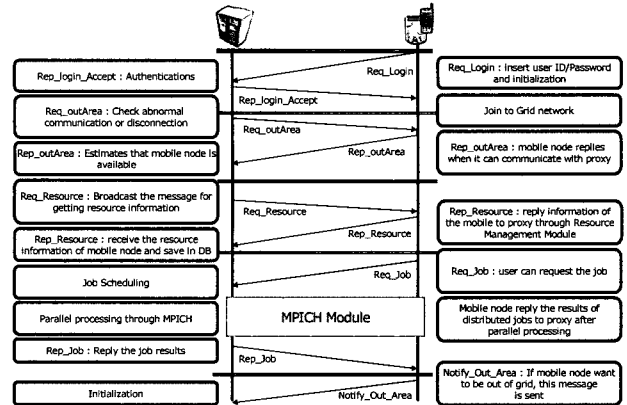
(그림 5)의 사용자 인터페이스에서는 프로그램의 시작에서부터 프로그램이 종료할 때까지의 정보를 사용자에게 제공한다. Super Daemon은 프락시와 통신을 수행하는 기능을 하며, Job Execution Module은 실질적인 분산 컴퓨팅을 수행하는 기능을 하고, System State Management Module은 자원 정보를 가공하여 사용자와 proxy에게 제공하는 역할을 한다. Resource Manager는 시스템 커널을 이용하여 직접적으로 시스템 및 가용 자원에 대한 정보를 취합하며, Node Controller는 모바일 노드의 전체적인 제어의 기능을 수행한다. 여기서



(그림 5) 모바일 노드의 설계

MGP(Mobile Grid Protocol)는 모바일 그리드에서 사용하는 통신모듈을 의미한다.

다음은 MGP 메시지의 동작과정을 나타낸 것이다.



(그림 6) MGP의 동작과정

(그림 6)에서 각각의 메시지의 정의는 다음과 같다.

- Req_Login : 모바일 단말기가 그리드 네트워크 내에 참가하기 위해 보내는 메시지이다. 여기엔 사용자 인증에 필요한 정보(사용자 ID, 인증정보(key or password))가 포함되며 단말기 인식 정보(IP, Port) 등의 정보를 알 수 있다.
- Rep_Login_Accept : 사용자의 접근에 대해 프락시는 인증을 수행한 후, 참여 가능 여부를 결정한다. 만약 인증이 성공적으로 수행되고 사용자의 참여를 허락할 경우, 해당 메시지를 전송하며 프락시는 해당 단말기에 대한 정보를 데이터베이스에 저장하며 이때, 각 노드를 구분하기 용이하게 하기 위한 Machine_ID를 할당한다.
- Rep_Login_Reject : 사용자의 접근에 대해 프락시의 인증이 실패한 경우, 해당 메시지가 전송되고 모바일 노드는 그리드 네트워크에 참여하는 것이 거부된다. 이 경우, 인증 정보를 다시 입력하여 Req_Login 과정부터 다시 수행해야 한다.
- Req_Resource : 사용자 단말이 그리드 네트워크에 참여한 후, 프락시는 주기적으로 단말기에 대해 필요한 자원 정보를 요청한다. 이 과정에 대한 응답은 요청된 자원 정보가 되며, 응답이 없을 경우 Req_Out_Area 메시지를 전송하여 해당 단말기가 송수신 범위 내에 있는지를 확인하게 된다. 요청에 대해 응답된 자원 정보는 데이터베이스에 저장되며 단말기가 전송 범위 내에 존재하고 있지 않다고 인식되면 해당 내용을 삭제한다.
- Rep_Resource : Req_Resource 메시지에 대한 자원정보 응답이다. 메시지의 간략화를 위해 요청된 내역에 대한 자원 정보만을 제공한다.
- Req_Job : 모바일 노드에서 작업 요청을 위해 보내는 메시지이다. 해당 메시지는 작업 내용을 포함하고 있다.
- Rep_Job : 요청되었던 작업에 대한 결과를 반환하는 메시

지이다.

- **Notify_Out_Area**: Mobile node가 사용자의 의도에 의해 그리드 네트워크에서 제외되고자 할 때 전송하는 메시지이다. 이 메시지를 받은 서버는 해당 노드에 대한 정보를 데이터베이스에서 삭제한다.
- **Req_Out_Area**: 모바일 노드가 정상적인 응답을 송신하지 않을 경우, 아직 그리드 네트워크 송수신 범위 내에 존재하는지를 확인하기 위한 메시지이다. 응답이 없을 경우에는 모바일 노드가 송수신 범위 외에 있다고 판단하고 해당 노드에 대한 정보를 데이터베이스에서 삭제한다.
- **Rep_Out_Area**: 모바일 노드가 Req_Out_Area 메시지를 받았을 경우, 이에 대한 응답을 보냄으로써 아직 활동이 가능하다는 것을 알려주는 메시지이다. 이 메시지를 받았을 경우, 프락시는 해당 모바일 노드가 정상적으로 동작 가능하다고 판단하여 계속 작업을 진행한다.

3.2 모바일 그리드 시스템의 구현

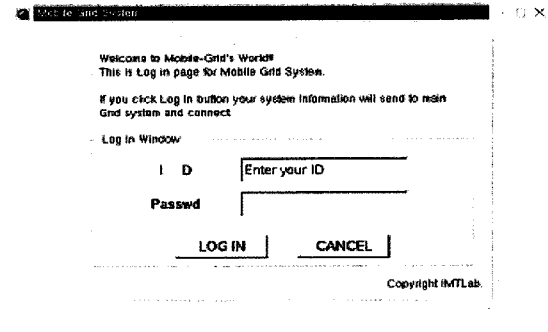
모바일 그리드 시스템을 구현하기 위해서 네트워크의 환경은 802.11b 무선환경에서 구현을 하였으며, 모바일 노드로는 PDA를 사용하였다. 프락시 시스템의 개발환경은 다음과 같다.

- OS: Redhat Linux 7.3 Kernel 2.4.18-3
- CPU: Intel Pentium III (Coppermine) 1GHz
- RAM: 256 Mbytes
- DataBase: mysql Ver 11.16 Distrib 3.23.49
- Compiler: gcc version 2.96

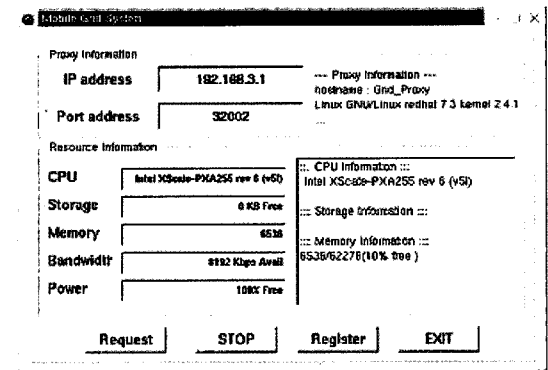
모바일 노드의 경우에는 PC와는 다른 운영체제 및 하드웨어로 구성이 되어 있고, 벤더마다 다른 구성을 하고 있어서 개발하기 전에 우선 Target Device를 선정하는 것이 중요하다. 일반적으로 PC의 경우에는 운영체제가 동일하면 개발환경이 유사하지만 PDA의 경우에는 다양한 종류의 Device들이 존재하고, 개발환경이 열악하기 때문에 단말기에 종속적인 방법을 통해 구현을 해야 한다. 본 논문에서는 Sharp에서 생산하는 PDA 제품군인 zaurus 시리즈를 Target Device로 선정하였으며, GUI는 Trolltech사에서 개발한 Qt/Embedded 라이브러리를 이용해 C++로 구현 하였다. 크로스 컴파일을 수행한 장비로는 Intel Pentium 3 1.7GHz사양의 PC에서 RedHat Linux 7.3를 설치해서 개발을 하였다.

아래의 그림은 사용자 인터페이스를 나타낸 것이다. 구현한 사용자 인터페이스로는 로그인 UI, 자원정보 표시 UI, 작업 요청 UI가 있다.

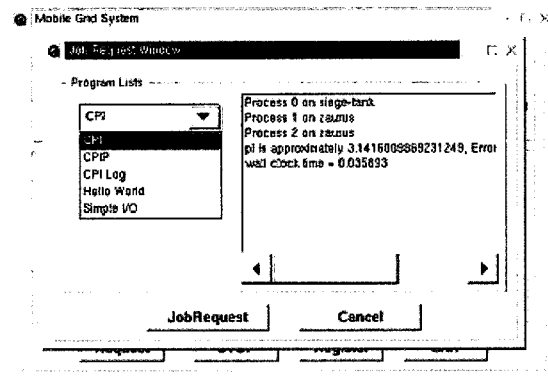
모바일 그리드 시스템의 자원으로써 작업을 수행한 모바일 노드의 로그 파일은 (그림 10)과 같다. (그림 10)에서는 로그를 통해서 성공적으로 프락시 시스템에 로그인을 하고 자원 정보를 얻어오며, 얻어온 정보는 프락시가 이용할 수 있는 형식으로 변환해서 보내주고 있음을 볼 수 있다. 또한 주기적인 프락시의 요청에 알맞게 대응하고 있는 것과 요청한 작업에 대한 결과를 받아오는 것을 볼 수 있다.



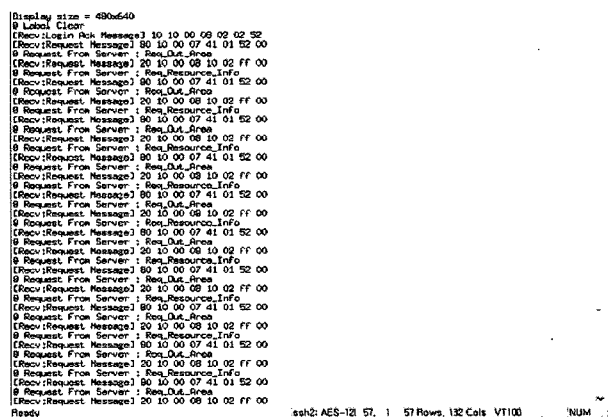
(그림 7) 로그인 UI



(그림 8) 자원정보 표시 UI

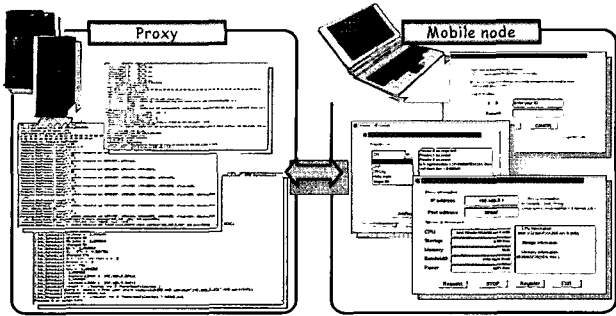


(그림 9) 작업요청 UI



(그림 10) 모바일 단말기의 로그 파일

모바일 그리드 시스템을 구현한 전체의 구성은 (그림 11)과 같다.



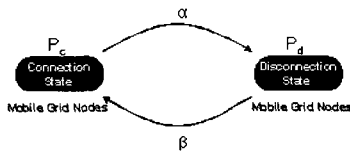
(그림 11) 모바일 그리드 시스템의 전체 구성

4. 작업 할당 스케줄링 알고리즘

작업의 할당시에 무선 네트워크의 단절성을 고려해서 다음과 같이 스케줄링 알고리즘을 작성하였다.

4.1 네트워크 단절성 고려

각각의 모바일 단말기에서 작업을 수행하기 위해서는 작업을 요청하는 패킷의 전송 및 작업이 완료된 뒤에 결과를 전달 받아야 된다. 그러나 무선 네트워크 환경에서는 이동으로 인한 네트워크의 단절이 발생할 수 있으므로 그에 대한 분석이 필요하다. 프락시와 모바일 단말기와의 연결 상태에 있을 확률을 P_c , 단절 상태에 있을 확률을 P_d 라고 하면 아래의 그림과 같이 나타낼 수 있다.



(그림 12) 모바일 단말기의 연결상태 전이도

여기서 α 를 disconnection rate라 하고, β 를 re-connection rate 라고 하면, P_c 와 P_d 의 값은 steady state property에 의하여 다음과 같이 나타낼 수 있다.

$$P_c + P_d = 1, -\alpha P_c + \beta P_d = 0$$

$$P_c = \frac{\beta}{\beta + \alpha}, P_d = \frac{\alpha}{\beta + \alpha}$$

여기서 작업 수행을 요청하기 위해 패킷을 전송하는 시간과 작업의 완료 후에 다시 결과를 재전송하는 시간을 단위시간 t 라고 가정하면, 전송시간 $f(t)$ 는 다음과 같이 나타낼 수 있다.

$$f(t) = \left(\frac{1}{P_c}\right)t = \left(\frac{\alpha + \beta}{\beta}\right)t \quad (1)$$

모바일 단말기에서의 작업 수행시간 예측은 다음과 같이 수행할 수 있다.

- W: 분산 프로그램을 수행하기 위해 필요한 전체 시간
- w_k : k번째 모바일 단말기에서 분산 작업 수행을 위한 필요 시간
- m: 가용한 모바일 단말기의 수
- S: 단말기 소유자의 개인 작업으로 인한 분산 작업의 중단 횟수
- λ_k : k번째 모바일 단말기에서 포아송 분포의 작업 arrival rate
- μ_k : k번째 모바일 단말기에서 작업의 service rate
- ρ_k : k번째 모바일 단말기에서 작업의 utilization rate
- T_k : k번째 모바일 단말기에서의 분산 작업의 완료 시간

k번째 모바일 단말기에서의 작업 수행 시간 (T_k)는 다음과 같이 나타낼 수 있다.

$$T_k = X_1 + Y_1 + X_2 + Y_2 + \dots + X_s + Y_s + Z$$

여기서, X_i : 분산 컴퓨팅 작업에 의해 소요된 컴퓨팅 시간이라 하고, Y_i : 단말기 소유자의 작업에 의해 소요된 컴퓨팅 시간이라고 정의하면

$$w = X_1 + X_2 + \dots + X_s + Z$$

$$T_k = w + Y_1 + Y_2 + \dots + Y_s$$

가 된다.

$\Gamma(0) = 0, \Gamma(S) = X_1 + X_2 + \dots + X_s$ 라고 하면,

$\Gamma(S) < w, \Gamma(S+1) \geq w$ 가 되어 S는 변수 λw 에 대해 포아송 분포를 따르게 된다.

$$\Pr(S = s) = \Pr(\Gamma(s) < w, \Gamma(s+1) \geq w)$$

$$= \frac{(\lambda w)^s 2^{-\lambda w}}{s!} \quad S = s > 0, E(S) = \lambda w$$

또한 $Y_j (j=1, 2, \dots, S)$ 는 i.i.d 이므로 다음과 같이 나타낼 수 있다.

$$E(Y_j) = \frac{1}{\mu - \lambda}$$

그러므로 작업 수행시간(T_k)은 다음과 같이 나타낼 수 있다.

$$E(T_k) = E(T_k | S) = E(w_k + Y_1 + Y_2 + \dots + Y_s | S)$$

$$= E(w_k + SE(Y_1)) = w_k + \lambda w_k E(Y_1) = \frac{1}{1 - \rho_k} w_k \quad (2)$$

```

Begin
proxy server checks the available mobile nodes;
collecting the available resources information of mobile nodes;
ordering the mobile nodes according to CPU, bandwidth, disk,
memory;
calculating network response time using (1) +(2) in each mobile
nodes;
partitioning and allocating subtasks to each mobile node
according to response time;
End
    
```

(그림 13) 작업 할당 스케줄링 알고리즘

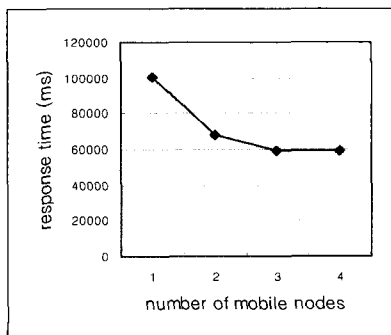
무선 네트워크의 전송시간 및 모바일 단말기에서의 작업 수행시간을 이용하여 작업을 모바일 단말기에 할당 알고리즘을 (그림 13)과 같이 설정하였다.

5. 모바일 그리드 시스템의 성능평가

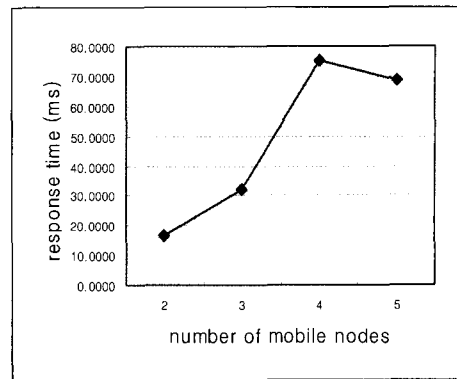
모바일 그리드 시스템의 성능평가를 위해서 본 논문에서는 모바일 그리드 노드에서의 수행 처리 속도와 비-그리드 환경 (일반 컴퓨팅 환경)에서의 수행 처리 속도를 비교하였다.

성능평가를 위해 사용된 프로그램은 지정된 범위 안에 있는 소수(prime number)를 구하는 프로그램으로써 1~100000 사이의 소수를 계산해내는 프로그램이다. 단일 노드에서 소수를 구하는 프로그램을 실행한 결과와 같은 프로그램을 모바일 노드의 개수를 증가시키면서 측정한 응답시간은 다음의 (그림 14)와 같다.

(그림 14)에 나타난 것처럼 하나의 노드에서 수행한 작업과 모바일 그리드 상황에서 수행한 작업의 응답시간을 비교해보면, 모바일 그리드에서 수행한 작업이 노드의 개수가 증가할수록 빨리 수행된 것을 볼 수 있다. 우선적으로 하나의 단말기에서 작업을 수행하였을 때의 응답시간은 100,475초에 수행이 되었으나 이 작업을 구현한 모바일 그리드 시스템에서 수행한 결과값은 모바일 단말기의 개수가 2개일 때 68,006초에 수행이 되었으며, 3개일 때는 59,224초에 그리고 4개일 때는 59,083초에 수행이 되었다. 즉 모바일 그리드 시스템을 이용하여 컴퓨팅을 수행하는 것이 효율적인 것임을 알 수 있다. 그러나 모바일 그리드에서 한가지 고려해야 될 점은 모바일 그리드에서 작업수행을 하고자 할 때에는 적절한 어플리



(그림 14) 응용 프로그램의 수행시간 비교

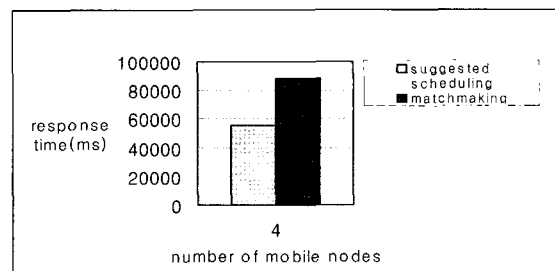


(그림 15) 파이 연산의 수행시간

케이션을 선정해야 한다는 것이다. 다음의 (그림 15)는 간단한 연산인 파이(π)의 값을 구하는 프로그램을 모바일 그리드 상에서 수행한 결과를 나타낸 것이다.

이 연산에서는 작업의 수행시간이 극히 작기 때문에 연산으로 인한 분산 컴퓨팅의 장점보다는 무선 네트워크 부하로 인한 시간지연이 더 크게 영향을 미치는 것을 알 수 있다. 또한 모바일 단말기의 개수가 5개로 증가하면서 응답시간이 감소하게 되는데 이는 단말기의 개수가 많아지면서 응답시간이 감소하는 것으로 나타났다.

또한 제시한 스케줄링 알고리즘과 Condor/G[13]에서 사용되는 matchmaking 방식의 스케줄링 알고리즘과의 성능 비교를 수행하였다. 시뮬레이션 환경은 위에서 수행한 환경과 같으며, 다만 두 개의 모바일 단말기에 비전용(non-dedicated) 환경을 설정하기 위해 평균적으로 50~60%의 CPU의 부하를 차지하는 작업을 수행하였다.



(그림 16) 스케줄링 알고리즘 수행시간 비교

프락시의 단점인 요청된 작업을 가용한 자원에 능동적으로 분배하는 것을 수행하기 위해서 네트워크에서의 전송시간 및 모바일 단말기에서의 작업수행 시간을 예측하여 작업을 할당하는 스케줄링 알고리즘을 제시하였으며, 이 스케줄링 알고리즘의 성능평가 결과 (그림 16)에서 제시된 바와 같이, 그리드에서 사용되고 있는 matchmaking 알고리즘보다 빠른 응답시간을 가져온 것을 알 수 있다.

6. 결론 및 향후 계획

슈퍼 컴퓨팅과 고속 네트워크 개념을 확장하고, 급속히 발

전하고 있는 무선이동 통신망 기술을 이용하여 이동성을 보장하는 것이 모바일 그리드이다. 모바일 그리드는 이동 단말의 급속한 증가와 성능의 향상으로 인하여 그 필요성이 더욱 증대되고 있다. 본 논문에서는 무선 네트워크 환경에서 그리드 작업을 수행할 수 있도록, 모바일 그리드 시스템의 설계 및 구현을 수행하였다. 또한 모바일 그리드 서비스의 유용성을 제시하기 위해서 성능평가를 수행하였다. 성능평가의 결과로 작업의 수행시간이 오랫동안 요구되는 프로그램의 경우 연산에 참여하는 모바일 단말기의 개수가 증가함에 따라 그 수행시간이 단축되는 것으로 나타났다.

향후 연구계획으로는 모바일 그리드 시스템 상에서 효율적으로 보안을 제공할 수 있는 방안에 대한 연구와 유·무선 그리드 통합 방안에 대한 연구를 수행할 예정이다. 이러한 모바일 그리드 기술은 유·무선의 자원에 대하여 공유를 가능하게 하는 기술적인 기반을 제공하며, 다양한 종류의 어플리케이션의 제공을 가능하게 한다.

참 고 문 헌

[1] F. Berman, G. Fox, T. Hey, The Grid: past, present, future, Grid Computing - Making the Global Infrastructure a Reality, Wiley and Sons.

[2] W. Ma, Y. Fang, "Dynamic hierarchical mobility management strategy for mobile IP networks", Selected Areas in Communications, IEEE Journal on Volume: 22, Issue: 4, May, 2004

[3] B. Strulo, J. Farr, and A. Smith, "Securing mobile ad hoc networks-a motivational approach", BT Technology Journal, 2003.

[4] Dario Bruneo, Marco Scarpa, Angelo Zaia, and Antonio Puliafito, "Communication Paradigm for Mobile Grid Users", Proceedings of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid, 2003.

[5] Tae-Kyung Kim, Dong-Young Lee, and T. M. Chung, "Mobile Agent-Based Misuse Intrusion Detection Rule Propagation Model For Distributed System", Lecture Notes in Computer Science: EurAsia ICT 2002, Tehran, Iran, LNCS2510 pp.842-849, October, 2002.

[6] S. Y. Yi and M. Livny, "Extending the Condor Distributed System for Mobile Clients", ACM Mobile Computing and Communications Review, 1999.

[7] T. Hayton, "The mobile Grid", Global Grid Forum 5.

[8] M. Migliardi, M. Maheswaran, B. Maniymaran, P. Card, and F. Azzedin, "Mobile Interfaces to Computational, Data, and Service Grid System", Mobile Computing and Communication Review, Volume1, Number 2.

[9] N. Suri, J. M. Bradshaw, M. M. Carvalho, and T. B. Cowin, "Agile Computing: Bridging the Gap between Grid Computing and Ad-hoc Peer-to-Peer Resource Sharing", Proceedings of the 3rd IEEE/ACM Interna-

tional Symposium on Cluster Computing and the Grid, 2003.

[10] Thomas Phan, Lloyd Huang, and Chris Dulan, "Challenge: Integrating Mobile Wireless Devices Into the Computational Grid", MOBICOM 02, September 23~26, Atlanta Georgia, USA.

[11] Globus-http://www.globus.org/

[12] M. Sanver, S. P. Durairju, A. Gupta, "Should one incorporate Mobile-ware in Parallel and Distributed Computation?", HiPC 2003.

[13] Condor-G: http://www.cs.wisc.edu/condor/condorg



김 태 경

e-mail : tkkim@imtl.skku.ac.kr
 1997년 단국대학교 수학교육(학사)
 2001년 성균관대학교 정보통신공학(석사)
 1996년~1997년 기아정보시스템 사원
 1997년~2001년 서울신학대학교 종합전산실 주임대리
 2001년~2005년 성균관대학교 전기전자및 컴퓨터공학부(박사)

관심분야: 그리드 네트워크, 네트워크 보안, 네트워크 QoS, 모바일 그리드



김 희 승

e-mail : hskim@imtl.skku.ac.kr
 2003년 성균관대학교 정보통신공학부(학사)
 2003년~현재 성균관대학교 컴퓨터공학과 석사과정
 관심분야: 그리드 네트워크, 시스템 보안, 네트워크 보안



이 현 주

e-mail : hjlee98@imtl.skku.ac.kr
 1979년 성균관대학교 정보통신공학부(학사)
 2003년~현재 성균관대학교 컴퓨터공학과 석사과정
 관심분야: 그리드 네트워크, 네트워크 관리, 네트워크 보안, AAA



정 태 명

e-mail : tmchung@ece.skku.ac.kr
 1981년 연세대학교 전기공학(학사)
 1984년 University of Illinois Chicago, 전자계산학과(학사)
 1987년 University of Illinois Chicago, 컴퓨터공학과(석사)
 1995년 Purdue University, 컴퓨터공학 박사

1985년~1987년 Waldner and Co., System Engineer
 1987년~1990년 Bolt Bernek and Newman Labs., Staff Scientist
 현재 성균관대학교 정보통신공학부 교수
 관심분야: 실시간시스템, 네트워크 관리, 시스템 보안, 네트워크 보안, 전자상거래, 그리드 네트워크