

프록시기반 모바일 그리드에서 자원관리 기법

조인석[†] · 이대원[†] · 이화민^{††} · 길준민^{†††} · 유헌창^{††††}

요 약

그리드 컴퓨팅은 지리적으로 분산되어 있는 기관들의 자원을 공유하여 협동 작업을 가능하게 하는 대규모 가상 컴퓨팅 환경이다. 최근 들어, 무선 통신망과 모바일 기기의 발달은 모바일 기기를 자원으로 이용 가능하도록 하고 있다. 하지만, 그리드 컴퓨팅에서 모바일 기기를 자원으로 이용하고자 하는 시도는 모바일 기기의 제약사항인, 낮은 프로세서 성능, 작은 메모리 용량, 한정된 배터리 용량, 낮은 통신 대역폭 등으로 어려움이 많다. 이에 본 논문에서는 모바일 그리드 환경에서 발생하는 제약사항의 극복을 위하여 프록시 기반 모바일 그리드 시스템을 제안하였다. 제안한 프록시 기반 모바일 그리드 시스템은 모바일 라우터를 사용하여 모바일 그리드 컴퓨팅 환경에서 모바일 기기가 자원 이용자 역할 뿐만 아니라 자원 제공자 역할도 수행할 수 있도록 하여, 기존 유선 그리드와의 협업 작업이 가능하도록 하였다. 또한, 본 논문에서 제안한 시스템에서 모바일 장치의 상태 변화에 대처할 수 있는 적응적 작업 스케줄링 기법을 제안한다. 그리고 제안한 작업 스케줄링 기법의 타당성을 검증하기 위해 SimGrid 시뮬레이션 도구를 사용하여 다른 작업 스케줄링 기법과 비교 분석하였다.

주제어 : 모바일그리드, 스케줄링, 프록시

Resource Management Scheme in Proxy-Based Mobile Grid

InSeock Cho[†] · DaeWon Lee[†] · HwaMin Lee^{††}
JoonMin Gil^{†††} · HeonChang Yu^{††††}

ABSTRACT

Grid computing has a large scale virtual computing environment that enables a collaborative processing through sharing resources of geographically distributed organizations. In recent years, the development of wireless networks and mobile devices enables mobile devices to consider as a resource of the grids. However, there are some problems such as low performance of processors, small capacity of storages, limited capacity of battery, and low bandwidth. In this paper, to overcome these limitations occurred in mobile grid environments, we proposed a proxy-based mobile grid system. Our proposed system enables mobile devices to play roles as a resource consumer and a resource provider and to collaborate with wired grids through a mobile router. Also, we propose an adaptive job scheduling scheme to cope with context changes of mobile devices and compare our scheduling scheme with other scheduling schemes using a simulation tool, SimGrid, to verify the validity of our job scheduling scheme.

Keywords : Mobile Grid, Scheduling, Proxy

[†] 종신회원: 고려대학교 컴퓨터교육학과

^{††} 종신회원: 순천향대학교 컴퓨터학부

논문접수: 2008년 8월 13일, 심사완료: 2008년 9월 20일

* 이 논문은 2006년 교육인적자원부 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2006-311-D00173).

^{†††} 종신회원: 대구카톨릭대학교 컴퓨터교육과 교수

^{††††} 종신회원: 고려대학교 컴퓨터교육과 교수(교신저자)

1. 서 론

그리드 컴퓨팅은 지리적으로 분산되어 있고 보안상으로 분리되어 있는 기관들을 가상 조직(VO: Virtual Organization)으로 만들어 자원들을 공유하여 협동 작업을 가능하게 하는 대규모 가상 컴퓨팅 환경이다[1][2]. 그리드 컴퓨팅은 원격 자원들의 통합을 가능하게 함으로써 연산 집약적 작업이나 데이터 집약적 작업을 처리하는데 주로 이용되고 있다. 즉, 그리드 컴퓨팅은 통신망으로 연결된 고성능 슈퍼컴퓨터들의 가용 자원들을 공유하도록 하여 컴퓨팅의 효과를 극대화하도록 하였다. 이와 같은 그리드 컴퓨팅의 패러다임은 개인용 컴퓨터의 가용 자원을 그리드 컴퓨팅의 자원으로 이용하는 데스크탑 그리드에 대한 연구로부터 모바일 기기를 잠재적인 자원으로 이용하는 모바일 그리드에 대한 연구로 변화하고 있다.

그리드 컴퓨팅에서 모바일 기기를 자원으로 이용하고자 하는 시도는 모바일 기기의 제약사항인, 낮은 프로세서 성능, 작은 메모리 용량, 한정된 배터리 용량, 낮은 통신 대역폭 등으로 어려움이 많았다. 하지만 최근 들어, 무선 통신망과 모바일 기기의 발달로 인해 모바일 기기를 자원으로 이용하여 컴퓨팅 성능을 개선하려고 하는 모바일 그리드에 대한 연구가 활발히 진행되고 있다[3][4][5][6][7]. 기존의 그리드 컴퓨팅 분야의 응용은 수치해석, 고에너지 물리학, 분자 모델링 등의 고성능 컴퓨팅 분야였지만 현재의 그리드 컴퓨팅 환경은 모바일 기기 자원의 출현으로 모바일 자원이나 사용자의 이동성을 응용하는 지진 관측, 생태 환경 관측, 기상 관측, 재난 방지, u-health, u-learning 등의 좀 더 지식 집약적이고 적응적인 작업 수행을 요구하는 분야에 적용되고 있다.

한편, 모바일 그리드에 대한 연구는 주로 기존 모바일 컴퓨팅에서 수행하였던 많은 연구 결과들을 토대로 이루어져 왔고 그리드 컴퓨팅에서의 모바일 기기는 주로 자원 이용자의 역할을 수행해 왔다. 그러나 최근에는 모바일 기기의 성능이 향상됨에 따라 자원 이용자의 역할 뿐 아니라 자원 제공자의 역할까지도 수행함으로써 모바일 기

기가 그리드 환경에 보다 적극적인 참여 형태로 변화하고 있다. 이러한 추세에 맞추어 모바일 그리드 환경에 적절한 특징을 반영하고 효과적으로 이를 지원해 줄 수 있는 방법이 요구된다.

따라서, 본 논문에서는 모바일 그리드 환경에서 발생하는 제약사항을 극복하기 위해 프록시 서비스 기반 모바일 그리드 시스템을 제안한다. 그리고 모바일 기기의 자원 제공자 역할로 유선 그리드와의 협업 작업도 지원하는 적응적 스케줄링 기법을 제안하고자 한다.

2. 관련연구

2.1 모바일 기기의 제약사항 및 역할

모바일 그리드 환경에서 모바일 기기를 자원 이용자 역할 이외에 자원 제공자 역할을 수행하도록 하기 위해서는 모바일 기기의 제한된 하드웨어적 특성을 해결하여야 한다. 모바일 기기의 제한적 특성이란 모바일 기기의 이동성, 배터리 용량, 신뢰성, 그리고 이동성 등이 있다.

먼저, 모바일 기기의 이동성(mobility)에서는 모바일 기기가 무선으로 연결을 유지하고 있는 동안 기기 사용자의 이동으로 인해, AP(Access Point)와의 연결이 약해지는 현상과 지형이나 날씨 등의 방해물에 따른 접속 끊김 현상에 의해 간헐적 접속이 발생할 수 있다. 따라서 모바일 그리드 컴퓨팅에서는 모바일 기기의 이동성을 고려한 작업 처리 기법이 필요하다.

모바일 기기의 전원은 충전을 통해 이루어짐으로써 매우 제한적으로 전원이 공급되어 모바일 기기의 사용 시간이 정해져 있다. 유선으로 연결된 고정 장치보다 낮은 전원이 공급됨으로 인해 모바일 기기를 이용함에 있어 효율적인 자원 사용 방법이 필요하다. 즉, 모바일 그리드 컴퓨팅 환경에서는 스케줄링에 있어 모바일 기기의 남은 전원에 따른 동적 자원 변경(dynamic resources change)을 반영하여, 정적 자원 정보뿐만 아니라 동적 자원 정보를 고려하는 적응적 작업 처리 기법이 필요하다.

그리드 컴퓨팅에 참여하는 장치들은 유선자원을 그리드에 제공하는 구조로 신뢰성이 매우 높

았다. 그러나 모바일 그리드 컴퓨팅에서는 정적 자원뿐만 아니라 동적 자원의 변화를 고려해야하므로 자원 변화에 따른 의도하지 않은 문제가 발생할 수 있다. 즉, 모바일 기기의 동적 자원의 변화에 따른 개별 신뢰성을 고려하여야 한다.

모바일 그리드 컴퓨팅 환경에서 고려해야 하는 이동성은 사용자 이동성(user mobility), 단말기 이동성(terminal mobility), 그리고 서비스 이동(service mobility)등으로 분류된다. 사용자 이동성은 사용자가 작업하던 환경의 이동으로 예를 들어 회사의 워크스테이션에서 사용하던 작업을 자신의 집의 개인 컴퓨터(PC)에서 사용하도록 이동하는 것을 들 수 있다. 단말기 이동성은 예를 들어 핸드폰을 가진 사용자가 이동함으로써 단말기 자체가 이동하는 경우이다. 마지막으로 서비스 이동성은 모바일 기기나 사용자의 위치나 이동에 관계없이 사용자가 원하는 서비스를 네트워크에서 제공해 줄 수 있도록 독립적인 서비스를 제공해주고 관리해주는 능력을 의미한다. 이 경우, 작업 처리 시 모바일 기기의 이동성에 따른 자원 상태 변화도 적극적으로 고려해야 한다.

모바일 기기를 그리드 컴퓨팅 환경에 적용하기 위한 연구는 미국과 유럽을 중심으로 진행되어 왔다. 그러나 초기 무선 그리드 연구에서는 모바일 기기는 자원 제공자로서가 아닌 자원 사용자의 관점에서 연구되었으며 모바일 기기가 그리드 컴퓨팅 환경에 통합되기 위한 몇가지 도전 과제들을 제시하는 수준이었다. 하지만 현재 무선 환경의 발달과 모바일 기기의 성능 향상으로 모바일 기기를 유선 그리드와 통합하기 위한 방법이 필요하다.

[4]에서는 특정 프록시 서비스를 두는 것이 아닌 소규모의 모바일 웹서버를 모바일 기기에 설치하고 설치된 모바일 웹서버에서 그리드 서비스를 구동하도록 하고 있다. [5]에서는 프록시 역할을 하는 그리드 호스트를 선택하고 여기에 모바일 객체를 복제하여 유선망에서의 그리드 객체의 공유를 제안하고 있다. [4][5]의 방법은 모바일 기기와 유선 그리드를 통합하기 위한 방법이다. 본 논문에서는 [4][5]와 달리 프록시 서버와 상호 연합하고 복제 기법을 통해 관리하는 프록시 관리 기법을 제안한다.

2.2 모바일 그리드에서 스케줄링 기법

모바일 그리드 컴퓨팅 환경의 모바일 기기에서 작업을 수행할 때 고려해야 할 것은 작업이 안전하게 수행될 수 있도록 하는 것이다. 즉, 모바일 기기가 작업이 수행되는 시간(모바일 기기에 작업 할당하는데 걸리는 시간, 실제 작업 처리 시간, 모바일 기기에서 수행된 작업의 결과를 받는 시간)동안 아무런 문제없이 작동하도록 유지시켜 주는 것에 초점을 두고 있다.

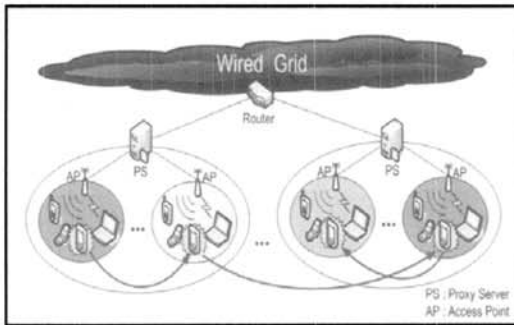
기존의 모바일 그리드에서는 모바일 기기의 이동성에 기반을 둔 성능 예측 방법에 의한 스케줄링 기법을 제안하고 있으나, 모바일 기기의 불규칙적인 이동 특징 때문에 정확도가 떨어진다[8]. 반면에, [9]에서는 모바일 기기의 전력 사용량에 기반을 둔 성능 예측 방법에 의한 스케줄링 기법을 제안하고 있으나, 모바일 기기의 통신에 대한 측면은 고려하지 않았다. [10]에서는 모바일 기기가 연결이 끊기더라도 작업은 모바일 기기에서 개별적으로 수행되며 모바일 기기가 네트워크에 다시 연결될 때 수행 결과를 돌려줄 수 있으므로 모바일 기기의 간헐적 접속 문제를 해결한 것처럼 보였다. 그러나 모바일 기기가 재접속하지 않을 경우에는 모바일 기기에서 수행된 작업 결과를 항상 되돌려 받을 수 없다는 단점이 있다.

따라서 모바일 그리드 컴퓨팅 환경에서 모바일 기기가 작업을 수행하기 위해 스케줄링을 할 때 모바일 기기의 상태 변화에 따라 현재 상황에 최적화된 적응적 스케줄링 기법이 필요하다. 본 논문에서는 프록시 서버 역할을 수행하는 모바일 라우터가 모바일 기기의 현재 자원 상태 정보를 기반으로 모바일 기기의 우선순위에 따라 작업을 할당하는 방식을 사용한다.

3. 모바일 그리드 시스템 모델

모바일 그리드 시스템은 기존의 유선 그리드 환경에 무선 환경이 연결된 이동 환경이다. 모바일 기기의 이동은 크게 프록시 서버를 기준으로 프록시 서버 관리 영역 내부이동과 새로운 프록시 서버 관리 영역으로 이동하는 2가지 이동이 존재

한다. <그림 1>은 모바일 기기의 이동성을 고려하는 그리드 환경을 보여준다.



<그림 1> 모바일 그리드 시스템

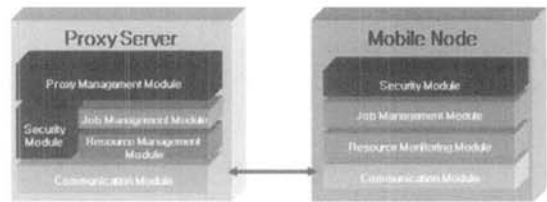
모바일 그리드 시스템 환경은 대학 캠퍼스와 같이 고정된 환경 내 모바일 기기의 이동성을 가지는 고정 환경 시스템과 이동하는 기차와 같이 모바일 기기를 포함하는 환경 자체가 이동성을 가지는 이동 환경 시스템으로 나눌 수 있다.

고정 환경 시스템은 모바일 기기를 가진 사용자의 이동성을 고려하는 환경으로서, 모바일 기기 개별과 프록시 서버 역할을 하는 모바일 라우터와의 연결을 고려해야 한다. 대학 캠퍼스와 같은 고정 환경은 일정지역(건물별 혹은 단과대학별)마다 AP가 있어서 일정 범위내의 모바일 기기들의 정보를 알 수 있으며, 캠퍼스 모바일 라우터에 프록시 서버 기능을 두어 캠퍼스 내의 AP들을 관리하는 환경이다.

이동 환경 시스템은 비행기나 기차 같이 모바일 기기를 포함하는 환경 자체의 이동성을 고려하는 환경이다. 노트북, PDA 등과 같은 모바일 기기를 가진 사용자가 서울에서 부산으로 가는 KTX 열차를 타고 이동한다고 가정해 보자. KTX는 서울에서 부산으로 이동하지만 모바일 기기를 가진 사용자는 기차 안에서 항상 머물고 있다. 한 기차 칸에 많은 모바일 기기 사용자가 있을 수 있으며, 이런 다양한 모바일 기기들의 자원들을 통합하여 그리드 환경에서 필요한 작업을 해결하는 데 사용할 수 있다. 기차가 출발지에서 도착지까지 이동하는 일정 시간 동안 모바일 기기 자체는 기차 외부의 라우터와의 연결을 고려하지 않아도 된다. 기차내의 모바일 라우터

가 외부 라우터와의 연결을 담당하므로 모바일 기기는 단지 기차내의 모바일 라우터와의 연결만을 고려하면 된다. 이런 이동 환경 시스템은 기차 내의 한 칸마다 AP가 있어서 그 칸 내의 모바일 기기들의 정보를 알 수 있으며, 기차의 모바일 라우터에 프록시 서버 기능을 두어 전체 기차 내부의 AP들을 관리하는 환경이다.

본 논문에서는 모바일 그리드 시스템 구조에서 핵심이 되는 부분을 모바일 라우터(프록시 서버)와 모바일 기기로 각각 모듈화 하여 구성요소들을 정의하였다. <그림 2>는 모바일 그리드 시스템에서 모바일 라우터(프록시 서버)와 모바일 기기에서 사용되는 구성요소들을 나타낸다.



<그림 2> 모바일 그리드 시스템 구성요소

모바일 그리드 시스템 구성요소는 다음과 같다.

- ① 프록시 관리 모듈(Proxy Management Module)은 모바일 그리드 시스템에 속하는 모든 라우터들 간의 협업을 담당하는 부분으로, 다음 2가지 모듈로 나뉜다.
 - 프록시 통합 모듈(Proxy Integration Module): 유선 그리드에 있는 라우터와 프록시 서버 역할을 담당하는 모바일 라우터 간의 협업을 담당하는 부분
 - 모바일 프록시 모듈(Mobile Proxy Module): 모바일 라우터들 간의 협업을 담당하는 부분
- ② 작업 관리 모듈(Job Management Module)은 사용자가 요청한 작업에 대한 정보, 스케줄링, 실행, 분배, 통합 등을 담당하는 부분이다.
- ③ 자원 관리 모듈(Resource Management Module)은 모바일 기기의 자원에 대한 전반적인 기능을 담당하는 부분으로, 다음 2가지 모듈로 나뉜다.
 - 자원 정보 모듈(Resource Information Module): 모바일 기기의 정적 자원 정보를 관리하는 부분
 - 자원 감시 모듈(Resource Monitoring Module): 모바일 기기의 자원 변화를 체크하는 부분
- ④ 통신 모듈(Communication Module)은 모바일 라우터(프록시 서버)와 모바일 기기간의 통신을 담당하

는 부분이다.

- ⑤ 보안 모듈(Security Module)은 작업 및 자원에 대한 사용 권한 여부 검사를 담당하는 부분이다.

본 논문에서는 프록시 서버 역할을 하는 모바일 라우터의 관리를 위해 모바일 라우터 간에 성능 차이에 기반을 둔 계층적 방법과 성능이 비슷한 모바일 라우터들 간의 협업을 가능하게 하는 리스트 방법을 사용한다.

계층적 방법은 2가지 모바일 라우터를 사용하는데, 전체 시스템을 총괄하는 슈퍼 스케줄러 역할을 하는 마스터 모바일 라우터(MMR: master mobile router)와 부분 시스템을 담당하고 서브 스케줄러 역할을 하는 서브 모바일 라우터(SMR: Sub mobile router)이다. MMR은 일반적으로 유선망에 존재하며 안정적이고 신뢰성이 높으며, 무선망에 존재하는 모든 SMR을 관리한다. 만약 하나의 SMR이 작업 수행 도중 결함이 발생하였을 경우 해당 SMR이 속한 MMR이 자신의 범위에 속하는 SMR 중 차순위 SMR에게 해당 작업을 넘겨준다. SMR의 순위는 SMR에 속하고 있는 모바일 기기의 신뢰도와 자원 상태 정보를 기준으로 판단한다.

리스트 방법은 모바일 라우터의 성능들이 비슷할 때 사용하는 방법이다. 임의의 모바일 라우터가 작업 수행 중 작업 시간 초과, 연결 단절 등으로 기능을 수행하지 못할 경우 주변의 다른 라우터에게 자신이 담당하고 있던 노드 및 작업에 관련된 정보를 넘겨준다. 즉, 모든 모바일 라우터는 자신의 주변에 있는 모바일 라우터의 거리에 따른 연결 정보를 리스트로 관리하고 있으며, 가장 가까운 모바일 라우터가 해당 리스트의 가장 윗부분에 존재한다.

4. 적응적 스케줄링 기법

유선 그리드 컴퓨팅에서의 주요 응용은 많은 데이터를 다루며 데이터 집약적인 작업을 하는 응용(data-intensive application)과 빠른 계산능력을 필요로 하는 연산 집약적인 작업을 주로 하는 응용(computation-intensive application)으로 나뉜다.

각 응용은 CPU와 메모리간 관계를 가지고 있으며, 사용자가 지정한 QoS(Quality of Service)에 맞추어 부적관계(negative relationship)를 갖는 상관관계 계수($\alpha + \beta = 1, 0 \leq \alpha, \beta \leq 1$; α : CPU의 관계 계수, β : 메모리의 관계 계수)를 설정한다.

작업 스케줄링을 위해 모바일 기기의 처리 능력과 통신 능력에 기반하여 모바일 기기의 자원을 관리한다. 즉, 본 논문에서는 모바일 기기의 CPU, 메모리, 대역폭(bandwidth), 지연시간(latency), 전력량(battery amount)을 각각 매개변수로 하여 현재 자원의 정보를 나타내는 값(IVR: Information Value of Resources)과 통신 신뢰도(CR: Communication Reliability)를 각각 식 (1)과 (2)와 같이 계산한다.

$$IVR(i) = \alpha * IV_{Cpu}(i) + \beta * IV_{Mem}(i) \dots (1)$$

- IVR(i): 모바일 기기 i의 현재 자원 정보 값
- IV_{Cpu}(i): 모바일 기기 i의 현재 CPU 정보 값
- IV_{Mem}(i): 모바일 기기 i의 현재 메모리 정보 값
- ($0 \leq IVR(i), IV_{Cpu}(i), IV_{Mem}(i) \leq 1$)

$$CR(i) = (B(i)/M_b(i)) + (C(i)/M_c(i)) - L(i) \dots (2)$$

- CR(i): 모바일 기기 i의 통신 신뢰도 ($0 \leq CR(i) \leq 2$)
- B(i): 모바일 기기 i의 현재 가용한 전력량
- M_b(i): 모바일 기기 i의 최대 가용한 전력량
- C(i): 모바일 기기 i의 현재 네트워크 대역폭
- M_c(i): 모바일 기기 i의 최대 네트워크 대역폭
- L(i): 모바일 기기 i의 네트워크 지연 확률

모바일 기기 i가 현재 네트워크에 접속되어 있지 않으면, 자원 정보 값(IVR(i))은 0으로 설정되며, 네트워크에 접속되어 있으면 각각 현재 가용한 자원과 최대 사용가능한 자원의 비율로 얻어진다. 즉, CPU 정보 값(IV_{Cpu})은 최대 사용가능한 CPU 양에 대해 현재 사용가능한 CPU 양의 비율이고, 메모리 정보 값(IV_{Mem})은 최대 사용가능한 메모리 양에 대한 현재 사용가능한 메모리 양의 비율이다. 즉, 현재 자원의 동적 자원 정보와 정적 자원 정보를 활용하였다.

본 논문에서는 모바일 기기의 자원 정보 값(IVR)과 통신 신뢰도(CR)를 기반으로 모바일 기기에 대한 순위에 따라 작업을 할당하는 스케줄링 기법을 사용한다. 사용자가 요구하는 모바일 기기의 자원 사용률에 맞추어 작업 복제 개수를 정하고, 현재 모바일 기기의 상태를 기반으로 스


```

while(모바일 라우터 실행 중)
{
while(작업 리스트에 있는 작업의 수 > 0) {
작업 리스트 중 가장 빠른 순서의 작업 가져옴;
if(작업 수행 권한이 없을 경우) {
작업 리스트의 다음 작업 수행;
} else {
모바일 기기들의 현재 신뢰도 계산
모바일 기기의 현재 자원 상태 정보 계산
if(현재 모바일 라우터가 관리하는 총 자원 < 작업
처리 시 필요한 자원) || (작업 수행 예상 시간
> 작업 요청 시 요구한 작업 수행 시간)
{
(1) 현재 모바일 라우터가 관리하는 자원의 신뢰
도 계산;
(2) 유선 그리드 망의 신뢰도 계산;
(3) 주변의 다른 모바일 라우터가 관리하는 자원
의 신뢰도 계산;
N = (1) + (2) + (3);
현재 모바일 라우터에게 ((1)/N)만큼 작업 분배;
유선 그리드 망에게 ((2)/N)만큼 작업 분배;
주변의 다른 모바일 라우터에게 ((3)/N)만큼 작업
분배;
}
(자원 상태 정보, 신뢰도)로 작업 복제 개수 계산;
자신이 관리하는 모바일 기기들에게 작업 복제;
}
}
if(모바일 기기에서 작업 결과 > 0)
작업복제번호와 작업번호를 검토후 작업결과 통합;
if(통합 작업 결과 > 0)
if(작업 번호에 해당하는 통합 작업 결과 = 1)
작업 번호의 첫 번째 작업 결과 값(FR(JobNo))에
통합 작업 결과 저장;
if(작업 번호에 해당하는 통합 작업 결과 = 2)
작업 번호의 두 번째 작업 결과 값(SR(JobNo))에
통합 작업 결과 저장;
if(FR(JobNo) = SR(JobNo)) {
해당 작업 번호를 수행하는 모든 자원들에게 작업
종료 수행;
작업 번호의 작업 결과 반환;
}
}
}
    
```

<그림 5> 스케줄링 알고리즘

5.1 모의실험 환경

모바일 기기의 개수와 수행되는 작업의 개수는 100개 단위로 하여 100~1000개까지 설정하였으며, 작업의 크기는 10~100 사이의 임의의 값(10 단위로 무작위)으로 설정하였다. 모바일 기기의 정적 자원 상태는 동일하다고 가정하였으며 모의 실험에 사용된 개별 모바일 기기의 사양은 <표 1>과 같다.

<표 1> 모의실험에서의 모바일 기기의 사양

| | 최소값 | 최대값 | 단위 설정 |
|------|--------|-------|---------------------|
| CPU | 100MHz | 1GHz | 100MHz |
| 메모리 | 64M | 1024M | 64,128,256,512,1024 |
| 대역폭 | 0.001 | 0.999 | random |
| 지연시간 | 0.0001 | 0.001 | random |

1회 최대 수행 시간은 1시간(3600초)으로 하였으며, 100MHz 64M의 모바일 기기가 대역폭(bandwidth)이 1이고 지연시간(latency)이 0일 때 1초에 처리할 수 있는 작업의 양을 1로 정하였다. 네트워크 그룹은 10개로 하였으며 하나의 네트워크 그룹에 속할 수 있는 모바일 기기의 수는 최소 1개에서 최대 100개로 하였다. 실험에서는 유선 그리드 망의 자원의 신뢰도는 0.9, 1초에 처리할 수 있는 작업의 양은 1000으로 설정하였다.

모바일 그리드 컴퓨팅 환경에서 특정 응용의 작업 완료 시간(T_{Comp})은 작업 통신 시간(T_{Comm})과 작업 수행 시간(T_{Exec})으로 나누어지며, 식 (3)과 같다.

$$T_{Comp} = T_{Comm} + T_{Exec} \dots\dots (3)$$

식 (3)에서 작업 통신 시간(T_{Comm})은 작업을 수행하기 위해 해당 장치에 전달하는 시간(T_{C_input})과 작업 수행 후 결과를 받는 시간(T_{C_output})으로 계산되며, 식 (4)와 같다.

$$T_{Comm} = T_{C_input} + T_{C_output} \dots\dots (4)$$

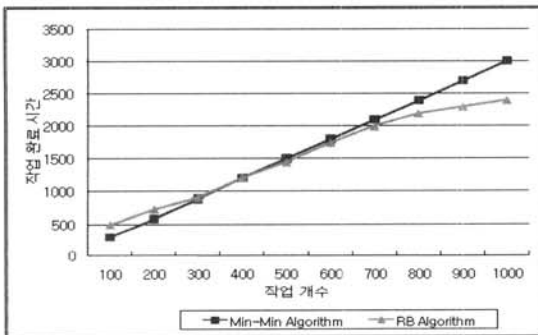
식 (3)에 나타난 작업 수행 시간(T_{Exec})은 대규모의 작업을 작은 규모의 작업으로 쪼개는 데 걸리는 시간($Sub(task(i))_{T_{Part}}$), 작은 규모로 쪼개진 작업들이 각기 수행되어 종료하는데까지 걸리는 시간($Sub(task(i))_{T_{Exec}}$), 최종 결과를 취하기 위해 분할된 작업들의 결과를 모으는데 걸리는 시간($Sub(task(i))_{T_{Coll}}$)으로 계산된다. 따라서 한 응용(task(i))의 작업 수행 시간(task(i) $_{T_{Exec}}$)은 식 (5)와 같다.

$$task(i)_{T_{Exec}} = Sub(task(i))_{T_{Part}} + Sub(task(i))_{T_{Exec}} + Sub(task(i))_{T_{Coll}} \dots (5)$$

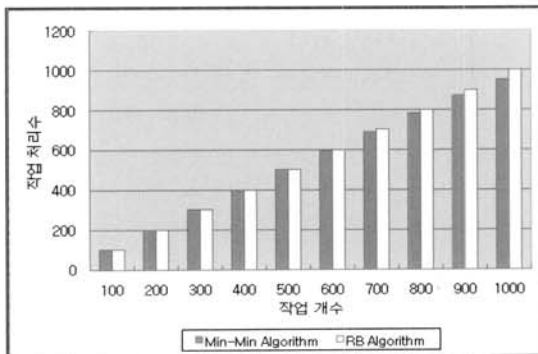
5.2 모의실험 결과

성능 분석을 위해 본 논문에서는 작업 개수의 변화와 모바일 기기의 개수 변화에 대한 작업 완료 시간과 작업 처리 수를 비교하였다.

작업 개수의 변화에 따른 작업 완료 시간은 <그림 6>에서 보여주며, 작업 개수의 변화에 따른 작업 처리 수는 <그림 7>에서 보여준다. 두 그림의 결과를 살펴보면, 본 논문에서 제안한 알고리즘(RB 알고리즘)이 작업의 개수가 많아질수록 작요 완료 시간 측면에서 Min-Min 알고리즘보다 점점 줄어들음을 알 수 있다. 또한, 작업 처리 측면에서도 본 논문에서 제안한 알고리즘이 작업 개수가 많아질수록 작업 처리 수 또한 증가하므로 모바일 그리드 시스템에서 더욱 신뢰적으로 작업을 처리할 수 있음을 보여준다.



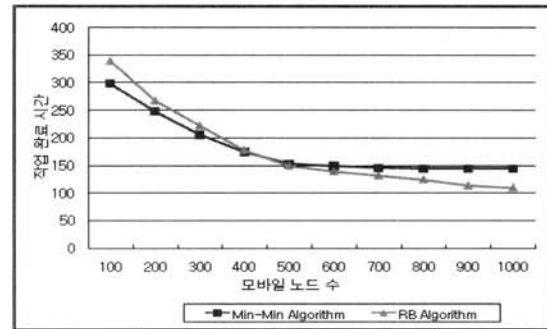
<그림 6> 작업 개수에 따른 작업 완료 시간



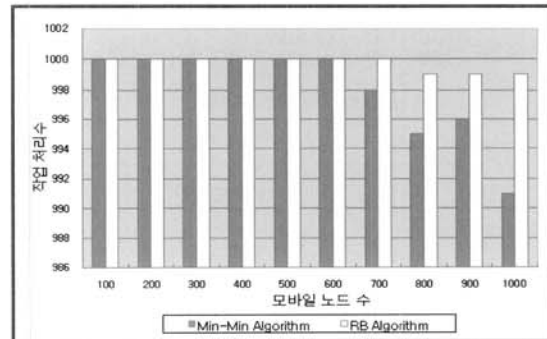
<그림 7> 작업 개수에 따른 작업 처리 수

<그림 8>과 <그림 9>는 각각 모바일 기기의

개수에 따른 작업 완료 시간과 작업 처리 수를 보여준다. 본 논문에서 제안한 알고리즘이 Min-Min 알고리즘보다 작업 완료 시간 측면에서 모바일 기기 수가 적을 때에는 작업 완료 시간이 높지만, 일정 수준 이상으로 모바일 기기의 개수가 많아질수록 성능이 향상되는 결과를 볼 수 있다. 또한 작업 처리 수 측면에서도 모바일 기기의 개수에 따라 제안한 알고리즘이 더 좋은 성능을 보여 주고 있음을 보여준다.



<그림 8> 모바일 기기 수에 따른 작업 완료 시간



<그림 9> 모바일 기기 수에 따른 작업 처리 수

6. 결론 및 향후 연구과제

본 논문에서는 모바일 그리드 컴퓨팅 환경에서 발생하는 제약사항을 극복하기 위해 프록시 기반 모바일 그리드 시스템을 제안하고, 모바일 기기의 동적 자원 변경에 따른 작업을 처리할 수 있도록 적응적 작업 처리 기법을 제안하였다. 이는 모바일 기기의 역할을 단순한 그리드 자원 이용자 역할 뿐만 아니라 그리드 자원 제공자 역할도

수행이 가능하도록 하였다. 또한, 프록시 서버 역할을 하는 모바일 라우터를 두어 기존 유선 그리드와의 협업 작업 시 모바일 기기의 결함을 포함하여 안정성과 신뢰성을 제공하였다. 성능 평가 결과 성능이 좋은 서비스에 작업을 할당하는 Min-Min 기법보다 조건에 따른 작업 복제, 분배 알고리즘을 이용한 적응적 작업 처리 기법이 더 효율적임을 알 수 있었다.

향후 연구과제로는 모바일 기기의 개수에 따른 성능 평가에서 보였듯이 본 논문에서 제안하는 작업 처리 기법을 경제적인 관점에서의 작업 처리 기법으로 보완하는 연구와 실제 NWS 응용과 같은 데이터를 이용한 실험을 통한 연구가 요구된다.

참 고 문 헌

[1] Foster, I., & Kesselman, C., & Tuecke, S. (2001). The Anatomy of the Grid: Enabling scalable Virtual Organizations. *International Journal Supercomputer Applications*, 15(3).

[2] Foster, I., & Kesselman, C., & Nick, J. M., & Tuecke, S. (2002). The Physiology of the Grid. <http://www.globus.org/research/papers/ogsa.pdf>.

[3] Kalim, U., & Jameel, H., & Sajjad, A., & Lee, S. (2005). Mobile-to-Grid Middleware: An Approach for Breaching the Divide Between Mobile and Grid Environments. *ICN 2005, LNCS 3420*, 1-8.

[4] Kianil, S. L., & Riaz, M., & Lee, S., & Jeon, T., & Kim, H. (2005). Grid Access Middleware for Handheld Devices. *EGC 2005, LNCS 3470*, 1002-1011.

[5] Isaiadis S., & Getov, V. (2005). A Lightweight Platform for Integration of Mobile Devices into Pervasive Grids. *HPCC 2005, LNCS 3726*, 1058-1063.

[6] Isaiadis S., & Getov, V. (2005). Integrating Mobile Devices into the Grid: Design Considerations and Evaluation. *Euro-Par*

2005, LNCS 3648, 1080-1088.

[7] The AKOGRIMO project: <http://www.akogrimo.org>

[8] Huang, C., & Zhu, Z., & Wu, Y., Xiao, & Z. (2006). Power-Aware Hierarchical Scheduling with Respect to Resource Intermittence in Wireless Grids. *MLC Conference*.

[9] Umar, F., & Wajeeha, K. (2006). A Generic Mobility Model for Resource Prediction in Mobile Grids. *IEEE CTS*, 189-193.

[10] Park, S. M., & Ko, Y. B., & Kim, J. H. (2003). Disconnected operation service in mobile grid computing. *ICSOC*, 499-513.

[11] Casanova, H., & Legrand, A., & Zagorodnov, D., & Berman, F. (2000). Heuristics for Scheduling Parameter Sweep Applications in Grid Environments. *Proceedings of the 9th Heterogeneous Computing Workshop (HCW'00)*, 349-363.

조 인 석



2006 고려대학교 사범대학
컴퓨터교육과(이학사)
2006~현재 고려대학교
컴퓨터교육과 석사과정

관심분야: 그리드컴퓨팅, 분산시스템
E-Mail: violet@comedu.korea.ac.kr

이 대 원



2001 순천향대학교 공과대학
전기전자공학부(공학사)
2003 고려대학교
컴퓨터교육과(교육학석사)

2003~현재 고려대학교 사범대학 컴퓨터교육과
박사과정
관심분야: 그리드컴퓨팅, 분산시스템, 이동컴퓨팅
E-Mail: daelee@comedu.korea.ac.kr

이 화 민



2000 고려대학교 사범대학
컴퓨터교육과(이학사)
2002 고려대학교
컴퓨터교육과(교육학석사)

2006 고려대학교 컴퓨터교육과(이학박사)
2006~2007 특허청 전기전자심사본부 통신사무관
2007~현재 순천향대학교 컴퓨터학부 교수
관심분야: 분산 컴퓨팅, 그리드 컴퓨팅,
결함 포용, 멀티에이전트 시스템

E-Mail: leehm@sch.ac.kr

길 준 민



1994 고려대학교 전산학과
(이학사)
1996 고려대학교 전산학과
(이학석사)

2000 고려대학교 컴퓨터학과 (이학박사)
2001~2002 미국 일리노이대(시카고) Post-Doc
2002~2006 한국과학기술정보연구원 선임연구원
2006~현재 대구카톨릭대학교 컴퓨터교육과 교수
관심분야: 그리드컴퓨팅, 컴퓨터교육

E-Mail: jmgil@cu.ac.kr

유 현 창



1989 고려대학교 이과대학
전산학과 (이학사)
1991 고려대학교 이과대학
전산학과 (이학석사)

1994 고려대학교 이과대학
전산학과 (이학박사)
1998~현재 고려대학교 컴퓨터교육과 교수
2006~현재 한국컴퓨터교육학회 부회장
2004 조지아텍 방문교수
관심분야: 그리드컴퓨팅, 분산시스템, 결함포용
시스템, u-learning

E-Mail: yuhc@comedu.korea.ac.kr