

## P2P Grid 컴퓨팅을 위한 에이전트 기반 보안 관리

김진택\*, 송오영\*, 박세현\*

\*중앙대학교 전자전기공학부

### A Study of Secure Agent-Based Management Model for P2P Grid Computing

Jin-taek Kim\*, Oh-young Song\*, Se-hyun Park\*

\*Department of Electrical and Electronics Engineering, Chung-Ang Univ.

#### 요 약

본 논문에서는 인터넷 기반 P2P 네트워크를 이용한 Grid 컴퓨팅에서의 효율적인 보안 관리 모델을 제시한다. 하나는 P2P 네트워크를 이용한 에러 복구 과정과 다른 하나는 에이전트간 신뢰관계 구성 방안이다. 또한 P2P 네트워크에서 다양한 네트워크 간 보안 통신을 보장하기 위해 SOAP을 이용한 보안 메시지 프로토콜을 제안한다. 제안된 새로운 보안 관리 모델은 P2P 네트워크로 구성된 Grid 컴퓨팅 네트워크의 성능을 향상시킬 것으로 기대된다.

#### I. 서론

오늘날 우주항공, 기상, 의료, 조선 등 다양한 분야에서 제품 설계 및 성능 분석등 대용량의 데이터 처리를 위해 슈퍼 컴퓨팅을 필요로 하고 있다. 하지만 슈퍼컴퓨터는 초기 도입 단가와 유지보수에 많은 비용을 요구하여, 일반 기업체 및 연구 기관에서의 사용이 어렵다. 최근 이에 대한 대체 기술로서 여러 대의 컴퓨터를 네트워크로 묶어 슈퍼 컴퓨팅을 수행하도록 하는 그리드 컴퓨팅(Grid Computing)이 주목받고 있다.

이러한 Grid Computing의 대표적인 플랫폼은 클루버스[1]로서 현재 많은 곳에서 이용되고 있다. 클루버스 플랫폼은 초기 공동의 연구 목적을 가진 사용자끼리의 컴퓨팅 자원의 공유를 위해 구성되었으나[2], 인터넷 사용자가 증가하고 컴퓨터 환경이 복잡해지면서, 분배한 작업의 안전한 결과값 반환을 보장하는 부분에 있어 잦은 Error recovery로 인하여 전반적인 컴퓨팅 효율을 저하시키고 있다.[3] 따라서 앞으로 Grid Computing에서 인터넷 기반의 망을 네트워크 구조로 가지게

될 경우 효율적인 작업 관리를 위한 새로운 작업 관리 모델이 요구되고 있다. 이러한 새로운 작업 관리 모델을 적용한 P2P형태의 Grid 컴퓨팅 환경은 서버 플랫폼 기반의 중앙 집중적인 자원 및 작업관리를 최소화하고, 작업을 수행하는 에이전트끼리의 P2P 프로토콜을 통한 협력을 통해 인터넷으로 연결된 개인 PC환경에서 효율적인 Grid Computing을 실현할 수 있다.

하지만 인터넷 망에서 P2P 네트워크를 구성하기 위해 새로운 작업 관리 모델을 위한 상호간 신뢰를 구축할 수 있는 방안이 선결되어야 한다. 따라서 본 논문에서는 인터넷망으로 연결되는 P2P 네트워크에 필수적인 에이전트간 상호인증 및 안전한 보안 구조를 P2P구조에 적용하여 새로운 형태의 Grid 작업 수행 모델을 제안하고자 한다.

#### II. 본문

##### 1. Grid Computing

오늘날, 컴퓨터의 처리속도와 네트워크 대역폭은 무어의 법칙 이상으로 빠른 발전을 거듭하고

있다. 그러나, 최신 슈퍼컴퓨터를 사용해서도 풀기 어려운 연산 문제들이 있다. 이런 어려운 연산 문제를 해결하기 위한 방법으로, 분산된 자원을 연결해 하나의 시스템으로 사용하고자 하는 노력이 진행됐다. 이를 메타컴퓨팅, 또는 P2P 슈퍼컴퓨팅이나 그리드(Grid) 컴퓨팅이라고 한다.

초기 클러스터링 컴퓨팅은 특정 지역적 폐쇄적인 네트워크를 이용하여 구성되었다. 비교적 적은 수의 시스템을 연결하였기 때문에 연산 성능이 별로 좋지 못했다. 그러나 분산 자원 관리 기술에 대한 연구가 진행되면서 많은 수의 시스템을 효과적으로 관리할 수 있게 되었다. 또한, 인터넷의 대역폭 확장으로 인해 LAN이나 자체 인터페이스를 이용하는 클러스터링에서 벗어나 국가나 전세계를 포함하는 대형 분산 시스템이 속속 등장하고 있다.[4]

이러한 대형 분산 시스템에는 단순히 컴퓨터만 Peer로 참여하는 것이 아니라, 대용량 저장장치, 다양한 고성능 연구 장비들이 포함되고 있는데, 이렇게 통합된 형태를 그리드(Grid)라고 한다. 그래서 그리드는 네트워크로 연결된 가상의 슈퍼컴퓨터이다. 그리드는 협업 업무, 컴퓨터를 이용한 정밀 실험, 원격 데이터셋의 검색, 원격 소프트웨어의 사용, 데이터 중심의 컴퓨팅, 대형 시뮬레이션, 무수한 변수가 사용되는 학술연구등에 활용될 수 있다. 다음 그림에서 그리드 컴퓨팅의 기본적인 개념을 보여주고 있다.

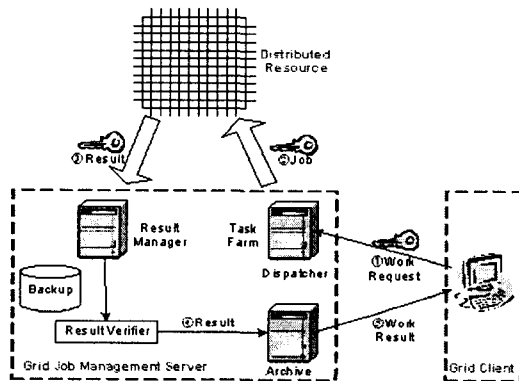


그림 1 Grid Computing의 기본개념

그림에서 Grid 클라이언트는 Grid 작업 관리 서버에게 작업을 요청하고, 그리드 네트워크 망에 소속된 많은 분산 에이전트들은 분배된 작업에 대한 결과를 반환한다. 이 과정에서 필요한 작업의 분배 수집과 같은 통제는 그리드 작업 관리 서버

가 수행하게 된다.

그리드 작업 관리 서버는 작업을 에이전트의 상태에 적합하도록 분배해야한다. 그래서 그리드 자원 관리서버의 지원을 필요로 한다. 그리드 자원 관리 서버에서는 사용 가능한 분산 자원의 상태를 관리한다. 그러나 그리드 네트워크를 구성하는 에이전트의 상태나 망 연결이 매우 유동적이기 때문에 사용 가능한 자원의 실시간 업데이트가 필수적이다. 이러한 실시간 업데이트는 그리드 네트워크를 중앙 집중적으로 만들게 된다. 따라서 자원 관리서버에 걸리는 부하가 증가되어 시스템 효율성이 감소한다.

이러한 시스템 효율성 감소를 해결하기 위해 본 논문에서는 그리드 구조에 P2P 네트워크 구조를 도입해 새로운 P2P 그리드 네트워크 구조를 제안하고자 한다. 또한 P2P 그리드 네트워크에 적합한 보안 작업 관리 모델을 제안한다.

다음은 Grid 컴퓨팅에 P2P 네트워크를 도입한 P2P Grid Computing Network Architecture이다.

## 2. P2P Grid 컴퓨팅 네트워크

인터넷 기반의 P2P 네트워크는 다양성과 불확실성이라는 두 가지 큰 특징을 가진다. 에이전트가 사용하는 인터넷 망은 신뢰성이 낮고, 에이전트의 자원은 따라서 그리드 컴퓨팅을 이용하기 위해서는 적합한 관리 모델이 필요하다.

또한 P2P 네트워크에서 구성원들은 다양한 네트워크 망에 접속되고, Firewall이나 Proxy와 같이 NAT 장비를 거치게 되어 주소정보가 변경되기 때문에 에이전트 정보의 실시간 업데이트를 전달할 서버가 필요하다.

그래서 P2P 그리드 컴퓨팅 네트워크의 기본 구성요소를 세가지로 정의했다. 첫번째 구성요소인 작업 관리 서버는 에이전트에게 적당한 작업을 분배하고, 그 결과를 보고 받아 정리한다. 두번째 구성요소인 에이전트들은 분배된 작업을 수행하고, 필요한 작업간 의존성과 오류 복구를 위해 다른 에이전트에게 P2P 프로토콜을 이용해 협력을 요청한다. 세번째 구성요소인 P2P 그리드 서버는 에이전트의 상태를 실시간 관리하고, 에이전트에게 작업간 의존성과 오류 에이전트의 작업 복구에 관련된 제어를 수행한다.

다음 그림은 P2P Grid 컴퓨팅 네트워크 구조를 나타내고 있다.

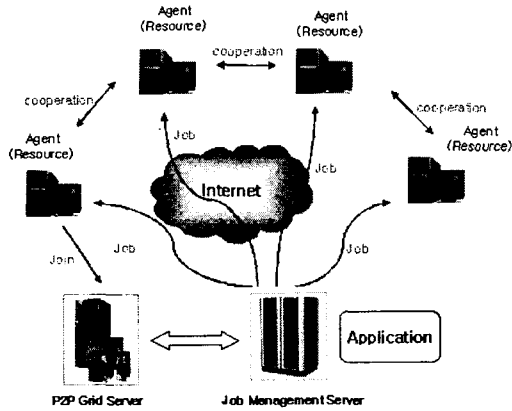


그림 2 P2P Grid 컴퓨팅 네트워크의 구조

만약 P2P 그리드 컴퓨팅 네트워크가 구성된다면, 에이전트들 간의 상호 협력 체계를 이루어 Job을 분산하여 처리할 수 있게 된다. 또한 P2P 형태의 에이전트 간 협력 체계와 효율적인 작업의 분산이 가능해진다.

하지만 인터넷 망 기반의 P2P네트워크를 사용하기 때문에 발생하는 보안적인 보완이 필수적이다.[5][6] 다음은 P2P Grid 구조에 필요한 보안을 제시하여 안전하고 효율적인 새로운 Grid 네트워크를 제안한다.

### 3. 보안 아키텍처

앞에서 설명한 P2P Grid 분산컴퓨팅 플랫폼에서 구성요소간 메시지의 비밀성, 무결성, 부인방지 그리고 상호인증이 필요하다.[7][8] 이러한 보안 기능을 제공해줄 수 있는 PKI 시스템을 도입한 전체 구성도를 제안한다. 다음 그림은 새로운 P2P Grid 분산컴퓨팅 구조이다.

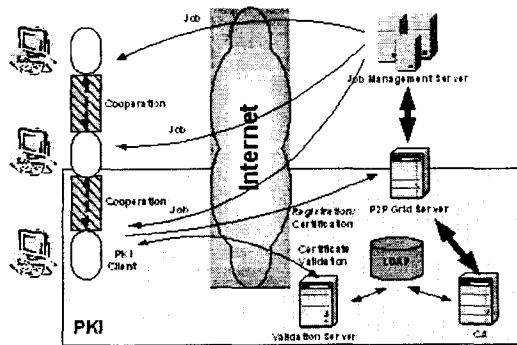


그림 3 새로운 P2P Grid 분산 컴퓨팅 구조

제안된 구조에서, P2P 그리드 서버는 에이전트의 등록과 인증에서 인증서를 이용한다. 작업관리 서버는 에이전트와 안전한 SSL 통신을 한다. 에이전트들은 안전한 통신을 위해 SSL을 이용하고, P2P 그리드 서버와 인증서 기반 인증을 수행한다. 또한 각 에이전트들은 에이전트간 안전한 통신과 신뢰관계 구축을 위해 인증서 기반 상호인증을 수행한다. 여기서 사용되는 인증서 기반 인증 및 상호인증은 PKI에서 사용되는 방법과 동일하다.

P2P 그리드 컴퓨팅 네트워크에 도입할 PKI의 모듈과 각 모듈들의 역할을 간단히 설명하면 다음과 같다.

■ CA 서버 : 인증서의 발행 및 폐기, 갱신 등을 수행하고 폐기에 따른 인증서 폐기 목록을 관리한다. CA 서버는 일관된 인증 정책을 적용해서 인증서를 발행한다. 또한 다른 공인 인증기관과의 상호인증을 기반으로 인증서 패스를 설정할 수 있으므로 인증서 사용의 효율성을 극대화 할 수 있다. 발행되는 인증서는 공개키 기반 인증서로 X.509 표준을 따르고 있다.

■ 등록서버 : 인증서 발행 대상자의 요청을 받아서 인증서 발행 및 폐기, 갱신 요청서를 CA에게 전달하는 역할을 한다. 또한 발행 대상자의 신원을 조사하고 확인하는 역할을 하므로써 CA가 인증서를 발행할 때 올바른지 않은 대상자에게 인증서를 발행하는 것을 막는다. 또한 인증서의 목적을 P2P 그리드 컴퓨팅의 에이전트 인증에 사용될 것이라고 인증서 내부에 명시한다.

■ 검증서버 : 인증서는 일정한 기간이 지나거나, 인증서 소유자의 요청으로 인하거나, 혹은 인증서 소유자의 키가 노출되었을 때 폐기되기 때문에 보안 모듈에서 사용하기 전에 사용하려는 인증서가 올바른지를 검사해야 한다. 통합적이고 체계적인 검증 정책을 유지하기 위해서 필요하다.

■ PKI 클라이언트 : 인증서를 가지고 등록서버나, 검증서버에 접속해서 인증서와 관련된 동작을 수행하거나 각종 암호화 기능을 수행하기 위해서 에이전트의 시스템 내부에 설치되는 프로그램을 말한다.

다음 그림은 간단한 P2P 에이전트간 메시지 무결성을 검증하는 방안을 보여주고 있다. 에이전트는 P2P Grid 서버에 등록될 때 각자의 인증서를 발행받고, 발행받은 인증서는 P2P Grid 서버로의 인증 때나 다른 에이전트와의 SSL이나 메시지 무결성을 제공하는데 이용한다. PKI 기반의 메시지 비밀성, 무결성, 부인방지등을 P2P 에이전트간 제공하므로써 보안적으로 안전한 통신을 제공하고

에이전트간 신뢰관계를 구축하여 P2P 협업관계를 구현할 수 있다.

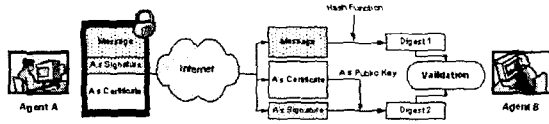


그림 4 에이전트간 메시지 무결성 검사

제안된 보안 구조를 적용하면 보안적으로 안전한 P2P Grid 네트워크를 구성하게 된다. 그래서 P2P 에이전트간 협업 시 신뢰성을 잃지 않기 때문에 다음절에서 제안하는 P2P Grid 보안 구조에서 사용되는 작업 관리 과정이 가능하게 된다.

#### 4. 보안 관리 모델 제안

지금부터는 제시된 보안구조로 인해 P2P간 신뢰관계가 구축된 다음, 에이전트간 협업형태로 작업을 분배할 수 있는 방법에 대해 설명한다. 제안하고자 하는 보안 관리 모델을 제안한 목적은 작업 관리 서버 부하를 분산시키는데 첫 번째 목적이 있고, P2P 에이전트간 협력으로 보다 간단하고 효율적인 에러복구가 가능하게 하는데 두 번째 목적이 있다. 또한 협업에 필요한 결과 데이터들간의 연관성을 효율적으로 해결할 수 있는데 세 번째 목적이 있다.

대부분의 Grid 작업은 반복적인 수치계산이 대부분을 차지한다.[9] 이러한 경우 에이전트가 수행하는 프로그램들은 원자성을 가진 프로세스로 구성되기 마련이다. 따라서 에이전트간 협업을 위해서는 원자성이 제공될 수 있고 반복적인 수치계산에 적합하도록 미리 컴파일 된 파일로 전송하는 것이 유리하다. 하지만 결과 값들 사이에 연관성이 발생하는 경우에 활용하기 위해서는 결과 값들 사이에 수행 순서와 입출력의 관계가 필요하다.

따라서 본 논문에서는 효율적인 작업 관리와 P2P 구조에서 사용이 가능하도록 모든 에이전트들에게 소스형태가 아닌 DLL, EXE와 같은 실행 파일 형태의 프로그램과 데이터를 제공한다. 또한 다양한 네트워크망에서 안전하게 에이전트까지 전달되기 위해 데이터와 실행 파일등은 XML 문서에 포장되어 SOAP으로 전송된다. 따라서 에이전트간 전송된 값에서 다른 부분은 데이터만 달라지게 된다. 그리고 XML 문서를 이용한 환경 설정을 통해 P2P 에이전트간 작업 연관성을 정의할 수 있다. 그로 인해 데이터들간의 연관성을 효과적으

로 연산할 수 있는 것이다.

또한 데이터를 골고루 분산시켜 이중화 내지는 삼중화로 전송하게 되면, 분산된 데이터들은 에이전트의 에러 발생시 P2P 프로토콜에 의한 Backup 프로세스의 데이터로 사용될 수 있다. 다음 그림은 제안된 모델을 적용한 경우 가능한 에러복구 시나리오이다.

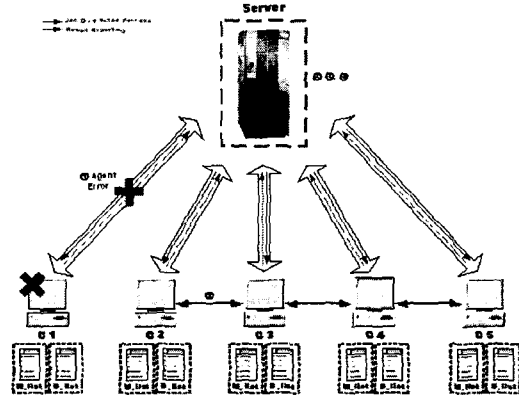


그림 5 에러 복구 시나리오

- ① 에러발생
- ② 차단되거나 작업을 수행하지 않는 에이전트를 찾아내고 해당 에이전트가 수행하지 않은 남아 있는 Job Unit을 계산한다.
- ③ 수행에 여유가 있는 에이전트를 보고된 상태 값에서 찾아낸다.
- ④ 찾아낸 에이전트에게 Job Unit을 할당한다.
- ⑤ 에이전트는 Backup Job Unit 요청에 따라 P2P 프로토콜을 사용해 다른 에이전트가 보유하고 있는 Backup List를 받아 작업을 수행 후 결과 값을 리턴 한다. 사용되는 Job List는 에이전트의 등급에 따라 Master List와 Backup List를 적합하게 할당한다. P2P Grid 서버는 각 에이전트의 상태 즉 자원의 량과 사용 가능한 우선 순위 등을 조합한 등급을 작업 관리 서버에게 보내고, 작업 관리 서버는 Job List를 작성하고 전송한다.

에이전트들은 전송 받은 Job Unit 중에 Backup List는 에이전트의 에러가 발생 시 사용되므로 초기 작업 수행 시에는 Master List에 있는 작업부터 수행한다. 에이전트들이 작업을 수행하면 하나의 데이터를 처리할 때마다 작업 관리 서버 쪽으로 결과 값을 리턴 하게 되는데, 모든 결과 값이 리턴되지 않고 P2P Grid 서버에서 에이전트의 상태가 변경되면 에러 복구 과정을 수행하게 된다.

작업 관리 서버에서는 에러가 발생한 에이전트에서 수행하고 있던 데이터 범위에 대해 P2P Grid 서버 쪽으로 전달한다. 전달받은 값으로 Backup List를 가지고 있는 에이전트의 아이디와 범위를 상태 값으로 선정된 에이전트에게 Backup List Job 요청을 통해 전달한다. 받은 에이전트는 Backup List를 가지고 있는 에이전트에게 메시지를 P2P Grid 서버를 경유해 보내게 된다. 받은 에이전트는 Backup List에서 해당 데이터를 추출하고 자신의 List에서 지운 다음 요청 에이전트에게 전송한다. 만일 여기서 에러가 발생했던 에이전트가 다시 수행될 경우 새롭게 P2P Grid 서버와 인증과정을 수행하게 되는데, 이미 에러가 발생했던 에이전트임을 구별해 수행되고 있는 Backup List 범위를 보내서 에이전트의 Master List를 수정하도록 한다.

제안된 방법은 기존에 서버가 모든 일을 하던 것에서 에러의 감시와 할당만 에이전트에게 전달하면 에이전트간 P2P 네트워크 구조를 이용해 에러가 발생한 작업을 연계해 수행할 수 있게 하였다. 그로 인해 작업 관리 서버의 부하를 감소시킬 수 있게 된다. 또한 관리 서버의 부하가 감소되면 Grid 네트워크 구성에 따른 비용의 절감과 함께 확장성이 증대된다.

또한 제안된 P2P간 신뢰 관계가 구축된 보안 모델에서는 XML 기반의 환경 파일을 이용해 계산 결과사이의 연관성을 지원할 수 있다. 다음 간단한 작업 할당 XML 코드를 보면 에이전트사이 에 계산된 결과 값을 다른 에이전트의 입력으로 사용할 수 있다. P2P를 통해 에러를 복구할 수도 있지만 다양하게 발생할 수 있는 데이터간 연관성을 XML 문서형태로 정의하여 쉽게 지원할 수 있다.

```
<?xml version=1.0?>
<workspace version=1.0>
  <project>Title of Project</project>
  <work>Work name</work>
  <serverset>
    <server address=Address>P2P Grid Server name
    </server>
  </serverset>
  <watcherset>
    <watcher name=Agent ID
    address=IP_address></watcher>
  <watcher name=Agent ID address=IP_address></watcher>
  <watcher name=Agent ID address= IP_address>
  </watcher>
</watcherset>
```

```
<fileset type=exec>
<file filename=programs name size=8>LOCATIONE_PATH
</file>
</fileset>
<fileset type=iibrary>
<file filename=DLL#1 size=8>LOCATIONE_PATH</file>
<file filename=DLL#2 size=8>LOCATIONE_PATH</file>
<file filename=DLL#3 size=8>LOCATIONE_PATH</file>
</fileset>
<fileset type=cfg>
<file filename=configure file name size=8>LOCATIONE_PATH
</file>
</fileset>
<fileset type=parameter>
<file filename=parameter file name size=8>LOCATIONE_PATH
</file>
</fileset>
<fileset type=data>
<file filename=data#0 size=8>LOCATION_PATH</file>
<file filename=data#1 size=8>LOCATION_PATH</file>
<file filename=data#n size=8>LOCATION_PATH</file>
</fileset>
<fileset type=result_data>
<file filename=result#0 size=8>LOCATION_PATH</file>
<file filename=result#1 size=8>LOCATION_PATH</file>
<file filename=result#n size=8>LOCATION_PATH</file>
</fileset>
<fileset type=backup_data>
<file filename=b_data#0
size=8>LOCATION_PATH</file>
<file filename=b_data#1
size=8>LOCATION_PATH</file>
<file filename=b_data#n
size=8>LOCATION_PATH</file>
</fileset>
</workspace>
```

표 1 작업 할당 XML 코드

serverset 부분은 P2P Grid 서버의 이름과 주소를 정의하는 부분이다. 다음의 watchset 부분에서는 연관성 있는 데이터를 가지고 있거나, 또는 백업 데이터를 가지고 있는 에이전트끼리 정의하는 부분이다. 그리고 각각의 fileset은 필요한 실행 파일, dll, 데이터 파일 및 백업 파일, 그리고 결과파일의 이름을 정의하는 부분이다.

이렇게 정의된 XML 문서를 기반으로 필요한 에이전트끼리 협력을 할 수 있고, 차례대로 데이터를 처리할 수 있다. 또한 해당 결과 값이 만일

다른 에이전트의 입력 값으로 쓰인다면 다음과 같은 XML 구문의 추가로 정의가 가능하다.

```

Agent ID: 1
<barrier type=cooperate>
<file filename=data#1 size=8>LOCATIONE_PATH</file>
<send agent_id=2 type=input filename=result#1
size=8>LOCATIONE_PATH </send>
</barrier>
Agent ID:2
<barrier type=cooperate>
<receive agent_id=1 type=input filename=result#1
size=8>LOCATIONE_PATH</receive>
<file filename=result#1 size=8> LOCATIONE_PATH
</file>
</barrier>
    
```

표 2 에이전트간 해당 결과 값이 다른 경우

Data#1을 가지고 계산된 결과 값 result#1을 에이전트 2에게 입력 값으로 사용될 것을 정의하는 구문이다. <barrier>로 정의된 구문은 해당 결과 값을 서버로 보내지 않고 해당 에이전트에게 직접 P2P 메시지로 전송하게 된다. 물론 받는 쪽 에이전트도 정의되어야 한다. 기존의 순차적으로 수행되는 데이터와 결과 값 쌍과는 별도로 연관성 있는 데이터만 별도로 정의한다. 따라서 데이터의 연관성이 있는 모델에도 쉽게 P2P Grid 컴퓨팅을 적용할 수 있을 것으로 기대한다.

### 5. 시뮬레이션 결과

제안된 새로운 Grid 보안 관리 모델의 성능을 비교 검증한다.

기존의 Grid 모델은 Work가 증가할수록 작업 관리 서버에서 담당하고 있는 Job의 개수도 비례적으로 증가하며 에러가 발생 할 때마다 추가적인 Job의 개수가 증가한다. 그러나 제안된 모델에서는 Work가 증가하더라도 초기 Work에서 필요한 실행과일을 생성한 다음 Job Unit의 분배이후에는 결과값의 수집만 수행되므로 에러가 발생하더라도 안정된 부하를 유지하게 된다.

기존의 Grid 모델에서는 Work의 증가와 상관없이 에이전트의 상태나 등록만을 담당했었던 Grid 관리 서버의 경우 제안된 모델에서는 에이전트간 에러 복구를 위한 메시지를 주고받게 된다. 그러나 Grid 서버에서 에러 발생에 따른 에이전트의 선정과 단 하나의 P2P Backup List 요청 메시지의 전달만을 수행하기 때문에 기존 Grid 서버에 비해 부하가 증가되지 않는다.

다음은 제안된 보안 관리 모델의 성능을 검증하기 위한 시뮬레이션 시나리오이다.

동일한 1000개의 에이전트를 수용하고 에이전트들은 랜덤한 주기로 에러 발생에 따른 재접속이 시도된다고 가정한다. 동일한 에이전트의 수행 패턴을 기존 방법과 제안된 방법에 적용하였을 경우 작업 관리 서버에 걸리는 부하의 량을 비교의 대상으로 한다. 여기서 기존 Grid 구조에서 작업을 분배하고 수집하는 모든 유닛을 하나의 서버에 있다고 가정하여 제안된 구조와의 차이점을 보완해 비교한다. 다음 표는 Grid 모델을 시뮬레이션 했을 때 Job의 증가 시 작업 관리서버 부하를 보여주는 표이다.

작업수	Grid 아키텍처	제안된 Grid 아키텍처
100	10	7
200	19	13
300	29	20
400	38	27
500	46	34
600	55	40
700	64	47
800	76	54
900	85	61
1000	95	67

표 3 작업 관리 서버의 부하 비교

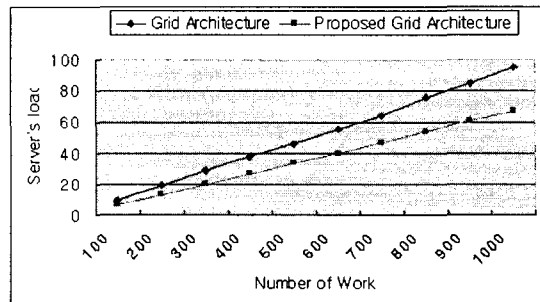


그림 6 시뮬레이션 결과

그림에서 제안된 모델은 작업의 수가 증가하더라도 에러에 의한 추가 부담이 없어 기존 모델에 비해 부하가 감소함을 알 수 있다. 에러가 발생하더라도 제안된 P2P Grid 네트워크에서는 에이전트 간 에러가 발생한 Backup List의 교환으로 작업 관리 서버의 부하를 주지 않으면서 에러를 복

구하게 된다.

또한, 기존 Grid 컴퓨팅에서는 데이터 사이의 연관성을 보장해 줄 수 없지만 제안된 P2P Grid 구조에서는 에이전트간 데이터 교환 및 결과 값 전송에 따라 연관성 있는 응용 모델에서도 Grid 컴퓨팅을 적용할 수 있다.

### III. 결론

본 논문에서 제안된 새로운 보안 Grid 관리 모델은 크게 세 가지의 특징을 가진다. 첫째로는 P2P를 도입함으로써 작업 관리 서버의 부하를 감소시켜 서버 응답성을 증가시키고, 부하의 예측을 통한 시스템 확장에 유리하다. 두 번째로는 P2P 에이전트간 인증서 기반의 보안 서비스를 제공해 에이전트 간 신뢰관계를 구축할 수 있다는 점이다. 세 번째로는 데이터간의 연관성을 가지는 경우에서도 Grid 컴퓨팅을 적용해 슈퍼 컴퓨팅이 가능하다는 점이다.

본 논문에서 제시하고 있는 Job Unit List의 다중화와 관리 과정, 보안 구조는 새로운 인터넷 기반의 P2P Grid 구조를 효율적으로 만들어준다. 또한 P2P 협업을 통한 데이터간 연계를 통해 다양한 응용 분야를 Grid에 적용할 수 있도록 해준다.

제안된 새로운 작업 수행 모델을 적용한 Grid Computing은 인터넷 기반 분산 컴퓨팅 환경을 조성하고, 슈퍼컴퓨터 또는 클러스터 시스템의 응용 기술 개발에 도움이 될 것으로 기대한다.

### 참고문헌

[1] <http://www.globus.org>  
 [2] A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, S. Tuecke, "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets". *Journal of Network and Computer Applications*, 23:187-200, 2001  
 [3] D. Angulo, I. Foster, C. Liu, and L. Yang. "Design and Evaluation of a Resource Selection Framework for Grid Applications," *Proceedings of IEEE International Symposium on High Performance Distributed Computing (HPDC-11)*, Edinburgh, Scotland, July 2002.  
 [4] K. Czajkowski, I. Foster, and C. Kesselman. "Resource Co-Allocation in Computational Grids." *Proceedings of the Eighth IEEE*

*International Symposium on High Performance Distributed Computing (HPDC-8)*, pp. 219-228, 1999.

[5] I. Foster, C. Kesselman, G. Tsudik, S. Tuecke. "A Security Architecture for Computational Grids." *Proc. 5th ACM Conference on Computer and Communications Security Conference*, pp. 83-92, 1998.  
 [6] Azzedin, F.; Maheswaran, M.; "Towards trust-aware resource management in Grid computing systems," *Cluster Computing and the Grid 2nd IEEE/ACM International Symposium CCGRID2002*, 21-24 May 2002, Page(s): 419 -424  
 [7] Novotny, J, Tuecke, S. and Welch, V., "An online credential repository for the Grid: MyProxy," *High Performance Distributed Computing, 2001. Proceedings. 10th IEEE International Symposium on* , 7-9 Aug. 2001, Page(s): 104 -111  
 [8] Johnston, W.E., Jackson, K.R. and Talwar, S., "Overview of security considerations for computational and data grids," *High Performance Distributed Computing, 2001. Proceedings. 10th IEEE International Symposium on* , 7-9 Aug. 2001, Page(s): 439 -440  
 [9] Butt, A.R., Adabala, S., Kapadia, N.H. and Figueiredo, R., "Fine-grain access control for securing shared resources in computational grids," *Parallel and Distributed Processing Symposium, Proceedings International, IPDPS, 2002, Abstracts and CD-ROM* , 15-19 April 2002, Page(s): 22 -29