그리드에서 서비스 기반 가상 탐색 시스템 설계 및 구현

(Design and Implementation of Service based Virtual Screening System in Grids)

이 화 민 † 진 성 호 †† 이 종 혁 ††

(HwaMin Lee) (SungHo Chin) (JongHyuk Lee)

이 대 원 †† 박 성 빈 ††† 유 헌 창 ††††

(DaeWon Lee) (Seongbin Park) (HeonChang Yu)

요 약 가상 탐색은 대규모의 화학분자 데이타베이스의 화학분자 데이타들을 분자 다킹과 같은 컴퓨팅 기술을 이용하여 한정된 소규모의 화학분자만을 스크리닝하는 과정으로, 대규모 컴퓨팅 파워와 데이타 저장 용량을 요구하는 대표적인 대규모의 과학 어플리케이션이다. AutoDock, FlexX, Glide, DOCK, LigandFit, ViSION 등과 같은 기존의 분자 다킹 소프트웨어나 어플리케이션들은 슈퍼 컴퓨터, 단일 클러스터, 또는 단일 워크스테이션 등을 이용하여 작업을 수행하도록 개발되었다. 하지만 슈퍼컴퓨터를 이용한 가상 탐색은 너무 많은 비용이 든다는 문제점이 있고, 단일 클러스터나 워크스테이션을 이용한 가상 탐색은 오랜 수행 시간이 요구되는 문제점을 가지고 있다. 이에 본 논문에서는 대규모의 데이타 집약적인 연산을 지원하는 그리드 컴퓨팅 기술을 이용하는 서비스 기반 가상 탐색 시스템을 제안한다. 이를 위해 본 논문에서는 가상 탐색을 위한 3차원 화학 데이타베이스를 구축하였다. 그리고 효율적인 분자 다킹 서비스를 제공하기 위해 자원 브로커와 데이타 브로커를 설계하고 가상 탐색을 위한 다양한 서비스들을 제안하였다. 본 논문에서는 DOCK 5.0과 Globus 3.2를 이용하여 서비스 기반 가상 탐색 시스템을 구현하고 성능 평가를 실시하였다. 본 논문에서 구현한 서비스 기반 가상 탐색 시스템은 신약 개발이나 신소재 개발 과정에서 연구 개발 기간을 단축하고 개발 비용을 절감할 수 있다.

키워드 : 가상 탐색, 분자 다킹, 그리드 컴퓨팅, 웹 서비스

Abstract A virtual screening is the process of reducing an unmanageable number of compounds to a limited number of compounds for the target of interest by means of computational techniques such as molecular docking. And it is one of a large-scale scientific application that requires large computing power and data storage capability. Previous applications or softwares for molecular docking such as AutoDock, FlexX, Glide, DOCK, LigandFit, ViSION were developed to be run on a supercomputer, a workstation, or a cluster-computer. However the virtual screening using a supercomputer has a problem that a supercomputer is very expensive and the virtual screening using a workstation or a cluster-computer requires a long execution time. Thus we propose a service-based virtual screening system using Grid computing technology which supports a large data intensive operation. We constructed 3-dimensional chemical molecular database for virtual screening. And we designed a resource broker

† 정 회 원 : 순천향대학교 컴퓨터학부 교수
leehm@sch.ac.kr
†† 비 회 원 : 고려대학교 대학원 컴퓨터교육학과
wingtop@comedu.korea.ac.kr
spurt@comedu.korea.ac.kr
ldw1996@comedu.korea.ac.kr
††† 비 회 원 : 고려대학교 컴퓨터교육과 교수
psb@comedu.korea.ac.kr
†††† 종신회원 : 고려대학교 컴퓨터교육과 교수
yuhc@comedu.korea.ac.kr
(Corresponding author임)

논문접수 : 2005년 8월 1일
심사완료 : 2008년 3월 16일

Copyright@2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 시스템 및 이론 제35권 제6호(2008.6)

and a data broker for supporting efficient molecular docking service and proposed various services for virtual screening. We implemented service based virtual screening system with DOCK 5.0 and Globus 3.2 toolkit. Our system can reduce a timeline and cost of drug or new material design.

Key words : Virtual screening, molecular docking, Grid computing, web service

1. 서 론

분자 모델링은 수용체(receptor)나 효소와 같은 거대 분자의 3차원 구조와 기존에 유사성을 갖는 것으로 알려진 단백질 구조 정보를 이용하여, 이들 거대 분자에 결합하여 활성을 조절할 수 있는 리간드(ligand)를 탐색하는 방법이다. 분자 모델링 방법 중 가상 탐색(virtual screening)은 컴퓨터를 이용하여 200만-300만개 정도의 대규모 화학분자 데이타베이스의 화학분자 데이타들을 표적 수용체에 다킹(docking)[1]시켜 1만-2만개의 화학분자만을 스크리닝하는 방법이다. 신약, 신소재, 고분자의 개발에 있어서 가상 탐색의 이용은 연구 개발 기간을 단축시킬 수 있기에 생화학분야에서는 매우 중요한 기술이다. 2004년 현재 PDB(Protein Data Bank) 데이타베이스에는 25,000개가 넘는 단백질 분자 정보가 존재한다. 가상 탐색 과정에서는 수용체와 리간드 분자의 구조를 분석하고 두 분자를 결합시켜 그 결합에너지를 계산하는 과정이 이루어진다. 가상 탐색은 이처럼 많은 분자들을 대상으로 복잡한 계산 과정을 수행해야 하기에 강력한 컴퓨팅 능력을 요구한다. AutoDock[2], FlexX[3], DOCK[4], LigandFit[5], Hex[6] 등과 같은 기존의 분자 다킹 어플리케이션들은 슈퍼 컴퓨터, 단일 클러스터, 또는 단일 워크스테이션 등을 이용하여 작업을 수행하도록 설계되고 구현되었다. 하지만 슈퍼컴퓨터를 이용한 가상 탐색은 너무 많은 비용이 든다는 문제점이 있고, 단일 클러스터나 워크스테이션을 이용한 가상 탐색은 오랜 수행 시간이 요구되는 문제점을 가지고 있다.

그리드(Grid)는 1990년대 중반 등장한 개념으로 슈퍼컴퓨터, PC, 저장 시스템, 데이타베이스, 데이타 소스, 다른 기관 소유의 특성화된 장치 등 지리적으로 분산되어 있는 광범위한 자원들을 공유하여 장시간 소요되는 컴퓨팅 작업의 성능 향상 및 비용 절감을 목적으로 하고 있다[7,8]. 최근 들어 그리드 컴퓨팅에 관한 많은 연구가 진행되어 왔고 기존의 많은 연구들이 주목할 만한 성과를 이루어내면서 그리드는 차세대 인터넷의 핵심을 이룰 기술로 기대를 모으고 있다. 현재 그리드 컴퓨팅은 여러 분야에서 응용되고 있는데, 그 중 분자 모델링, 기상관측, 수치해석, 핵물리 실험 등 대규모 컴퓨팅 능력을 요구하는 생화학, 물리학, 생물학 등 과학 분야에 가장 활발히 이용되고 있다[8].

가상 탐색 과정은 연산 수행 시간이 작게는 며칠에서부터 길게는 몇 년 까지 걸리는 대표적인 대규모 과학 애플리케이션이다[1,8]. 이에 본 논문에서는 대규모의 데이타 집약적인 연산을 지원하는 그리드 컴퓨팅 기술을 이용하여 서비스 기반 가상 탐색 시스템을 제안한다. 이를 위해 본 논문에서는 가상 탐색을 위한 3차원 화학분자 데이타베이스를 구축하였다. 또한 효율적인 분자 다킹 서비스를 제공하기 위해 자원 브로커와 데이타 브로커를 설계하고 가상 탐색을 위한 각종 서비스들을 제안하였다. 그리고 DOCK 5.0[5]과 Globus 3.2[9,10]를 기반으로 서비스 기반 가상 탐색 시스템을 구현하였다. 본 시스템은 신약 개발이나 신소재 개발 과정에서 개발 기간을 단축하고 비용 절감을 가져올 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 분자 다킹과 그리드 컴퓨팅에서 가상 탐색 관련 연구들을 살펴본다. 3장에서는 가상 탐색을 위한 데이타베이스 구축 방법을 설명하고 4장에서는 서비스 기반 가상 탐색 시스템의 구조를 제시한다. 5장에서는 본 논문에서 설계한 자원 브로커와 데이타 브로커에 대해 설명하고 6장에서는 가상 탐색을 위한 서비스들을 소개한다. 7장에서는 구현 및 실험 결과를 제시한다. 마지막으로 8장에서 본 연구의 결과와 향후 연구 과제를 제시한다.

2. 관련 연구

분자 모델링에서 목표 수용체에 대해 화학 데이타베이스(CDB, Chemical Database)에 있는 수많은 리간드 또는 합성 분자와의 결합 적합성을 심사하는데 이 과정을 분자 다킹(molecular docking)이라고 부른다. 분자 다킹은 분자 모델링의 핵심 부분으로 주어진 두 분자들 사이의 상호작용 과정에서 두 분자들 사이의 상대적인 위치를 구하는 문제이다. 일반적으로 분자 다킹은 작은 리간드 분자와 상대적으로 크기가 큰 수용체(receptor) 분자들을 결합하는 것을 말한다. 그림 1은 일반적인 분자 다킹의 모습을 나타내는 것이다. 진하게 나타낸 부분들이 분자 다킹에서 가장 중요하게 다루어지는 바인딩 사이트들을 나타낸다. 분자 다킹 문제를 해결하기 위해서는 탐색 알고리즘을 통한 분자 결합 상태 탐색과 스코어링 함수를 통한 스코어링 과정이 이루어져야 한다. 하지만 이 과정들은 매우 복잡하고 오랜 시간이 요구되는 문제들이다. 따라서 이러한 문제들을 컴퓨터를 이용하여 효율적으로 수행하기 위해서 분자 다킹을 지원해주는 많은 프로그램들이 출시되었다. 현재 출시되어 있

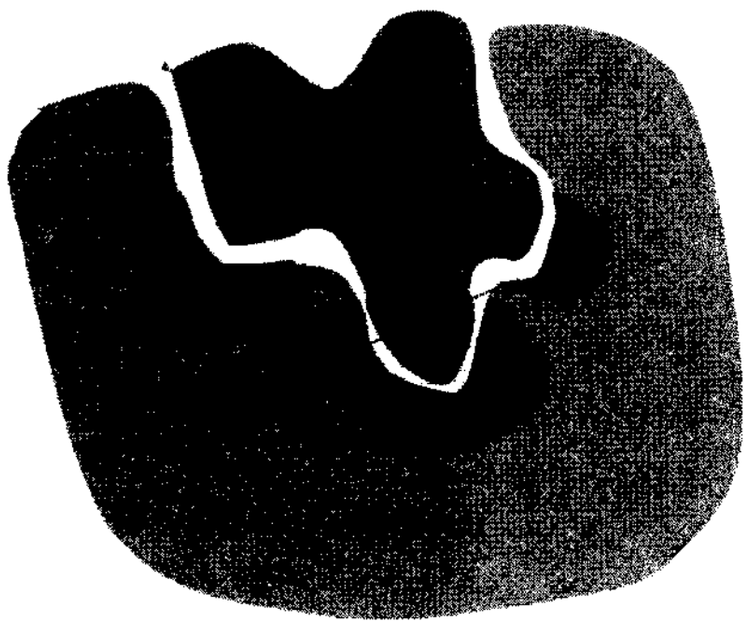


그림 1 일반적인 분자 다킹 모습

는 프로그램 중 어느 것도 분자 다킹 어플리케이션의 표준으로 자리 잡고 있지는 않지만 AutoDock, DOCK, Hex 등과 같은 프로그램들은 분자 다킹을 연구하는 연구자들에 의해서 가장 널리 사용되고 있다.

최근 들어 그리드 컴퓨팅에서 가상 탐색과 관련된 많은 연구들이 이루어지고 있다. 대표적인 연구들을 살펴보면 다음과 같다.

[11]에서는 그리드 환경에서 Nimrod-G을 이용하여 분자 다킹 프로세스들이 파라메타 스윕 어플리케이션(Parameter Sweep Application) 형태로 수행되는 환경을 제공한다. 이를 위해 [11]에서는 원격으로 작은 규모의 CDB에 접근할 수 있게 하는 CDB 관리 및 접근 툴과 Nimrod-G 자원 브로커를 제안하였다. 그러나 [11]은 글로버스 툴킷 2.2를 기반으로 구현이 되어 웹 서비스를 기반하는 글로버스 3.2 환경에는 적합하지 않다. 그리고 제안된 CDB 관리 및 접근 툴에서 제공되는 기능이 제약적이고, 그리드 컴퓨팅 환경과 같은 대용량의 이질적인 CDB에서 필요한 이질적인 CDB 통합 관리 기능과 분산된 CDB에서의 검색 기능 등을 제공하지 않는다.

[12]에서는 그리드 컴퓨팅 환경에서 단백질 구조 비교를 위한 툴을 제안하였다. 이를 위해 3차원 단백질 구조들의 다양한 변형 성질들을 테이블에 저장하는 색인 기법을 기반한 비교 알고리즘을 개발하고 글로버스 MPICH를 이용하여 4개의 노드에서 실험하였다. 하지만 이 논문은 단백질 구조 비교를 위해 PDB에서 단백질 구조 정보를 보다 빠르게 검색하고 저장하는 방법 제시에만 연구가 한정되어 있다.

[13]에서는 다양한 과학 및 공학 분야에서 대규모의 독립된 작업들을 수행해야 하는 파라메타 스윕 어플리케이션들을 그리드 컴퓨팅 환경에서 수행하기 위해 APST (AppLeS Parameter Seep Template) 소프트웨어를 제시한다. 하지만 APST는 일반적인 파라메타 스윕 어플리케이션 템플릿으로 실제 가상 탐색 시스템에 적용하기 위해서는 많은 수정 사항이 요구된다.

BioSimGrid[14] 프로젝트에서는 바이오분자 시뮬레이션을 위한 그리드 인프라를 구축을 목적으로 한다. 이를 위해 유럽의 각 연구소에 분산된 바이오분자들을 통합 관리하는 BioSimGrid 데이타베이스를 구축하고, OGSA-DAI[15]와 SOA(Service Oriented Architecure)을 기반으로 구축된 BioSimGrid 데이타베이스를 이용할 수 있도록 하는 그리드 포탈 서비스를 구현하였다. 이 논문에서는 그리드 컴퓨팅 환경에서 이질적으로 분산된 데이타베이스 구축 및 접근 서비스만을 제시했을 뿐 가상 탐색과 같은 어플리케이션의 적용 방법은 제시하지 못하고 있다.

이에 본 논문에서는 그리드 컴퓨팅에서 이질적으로 분산된 화학 데이타베이스를 통합 관리하는 3차원 화학 데이타베이스를 구축하고 웹 서비스를 기반한 가상 탐색 시스템을 설계하고 구현하고자 한다.

3. 가상 탐색을 위한 3차원 화학 데이타베이스 구축

분자 모델링을 위한 화학 데이타베이스는 화합물의 다양한 구조를 포함하고 있는 이형체(conformer) 데이타베이스 형태로 구축하였다. 기존의 화학 데이타베이스는 파일의 형태로 존재했다. 하지만 화합물의 종류가 기하급수적으로 증가하면서 파일 용량이 커지게 되어 다루기가 힘들고 정보 검색 면에서 비경제적이고 용이하지 않았다. 또한 기존의 화학 데이타베이스들은 제조 회사마다 필드들의 순서와 정도들이 상이하게 구성된 파일의 형태로 존재했다. 이에 화합물의 종류가 많아지면서 파일의 크기가 커지게 되어 화학 데이타의 정보 검색 및 추가/삭제 등 데이타 관리에 있어서 어려움이 증가되었다. 그리고 각 회사마다 파일의 필드들과 필드 값들이 이질적으로 구성되어 여러 화학 데이타의 통합 관리가 어려웠다. 이에 본 연구에서는 기존의 여러 화학 데이타베이스 파일들을 MySQL을 이용하여 하나의 통합 데이타베이스로 구축하였다. 본 논문에서 구축한 데이타베이스 테이블 Protomer (protein과 conformer의 합성어)의 스키마는 표 1과 같다.

본 논문에서 제안하는 서비스 기반 가상 탐색 시스템에서는 총 32,889개의 화학분자에 대한 데이타베이스를 구축하였다. 이 시스템은 구축된 화학 데이타베이스에서 가상 탐색을 위해 반드시 필요한 필드들을 검색하여 검색된 필드 정보들을 이용해 자동으로 mol2 파일을 구성해준다. 이를 위해 구축된 데이타베이스는 Query Evaluation Service Factory, Query Evaluation Service 등 두 가지의 서비스를 제공한다.

4. 서비스 기반 가상 탐색 시스템 구조

표 1 Protomer 데이타베이스 테이블 스키마

Attribute	Description
Prot_ID	ZINC_ID
Type	Subset Type ex. Fragment-like, Drug-like
Mol_name	분자명
LogP	P=[옥탄올 층의 화합물의 농도]/ [물 층의 화합물의 농도]
Apolar_desolvation	무극성 탈용매화
Polar_desolvation	극성 탈용매화
H_bond_donors	수소 결합 donor 개수
H_bond_acceptors	수소 결합 Acceptor 개수
Charge	Charge
Molecular_weight	분자량
Rotable_bond	Rotable bond 개수
Content	Mol2 파일 전체 내용

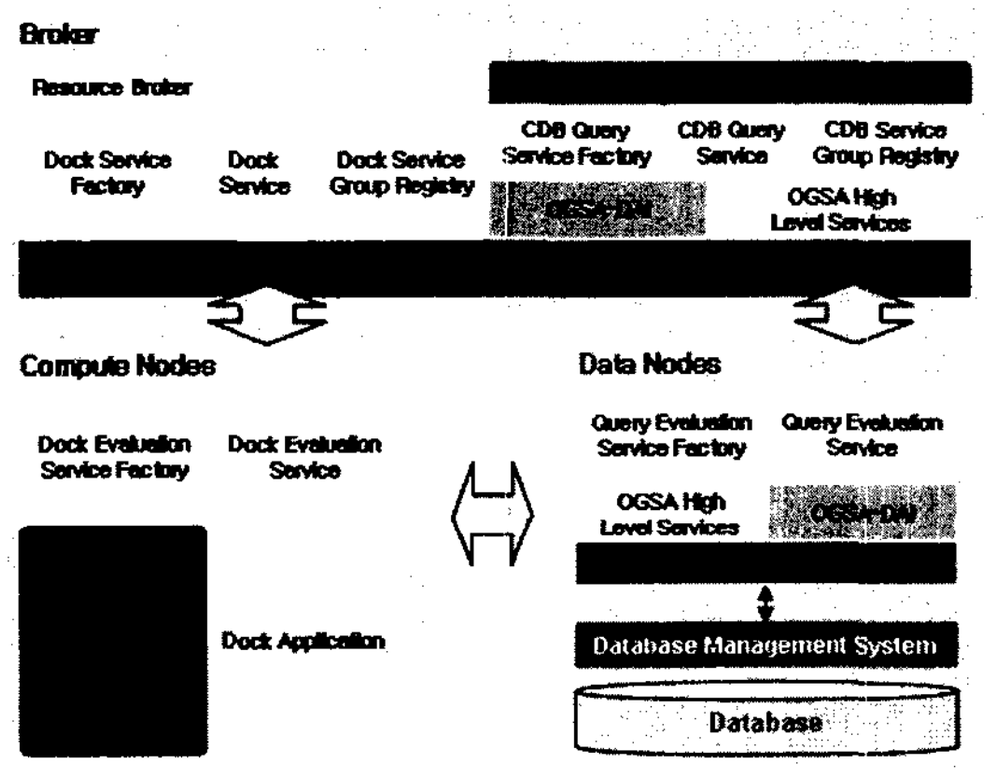


그림 2 그리드 서비스 기반 가상 탐색 시스템의 전체 구조

그림 2는 본 연구에서 제안하는 서비스 기반 가상 탐색 시스템의 전체 구조이다. 서비스 기반 가상 탐색 시스템은 크게 브로커(broker), 계산 자원(computation resource), 데이타 자원(data resource)으로 구성된다. 서비스 기반 가상 탐색 시스템의 각 구성요소의 역할은 다음과 같다.

1) 자원 브로커(Resource Broker)

그리드에서 분자 다킹을 수행하기 위해서 자원 브로커는 사용자의 요구에 따라 다킹 작업을 복제하고 큰 덩어리의 작업을 작은 규모의 다킹 작업으로 세분화시키는 기능이 필요하다. 그리고 세분화된 각각의 다킹 작업에 대해서 작업의 특징이나 자원의 특징에 따라 매핑되어진 정보를 고려해 작업의 흐름(workflow)을 작성하는 기능과 세분화된 작업을 정해진 원격 자원에 할당(dispatch)시키는 기능을 제공해야 한다. 이를 위해 자원 브로커는 작업의 실행 파일, 환경 변수, 초기 입력 데이타 값 등을 원격 호스트에 전송할 수 있어야 한다. 그리고 자원 브로커는 정보 서비스를 이용하여 각 원격 호스트에서 수행되고 있는 세부작업들의 상태를 모니터링 하여 작업의 상태에 따라 신뢰성 있는 다킹 작업이 수행될 수 있도록 관리해야 한다. 마지막으로 자원 브로커는 연산 결과의 수집과 수집된 연산 결과를 종합하여 사용자나 그리드 포탈에 전달하는 기능을 제공해야 한다. 본 연구에서 자원 브로커는 MDS(Metacomputing Directory Service)를 이용하여 계산 노드의 메모리, CPU 사용량 정보를 기반으로 가상 탐색을 수행할 계산 노드를 결정한다. 이를 위해 본 연구에서 구현한 자원 브로커는 Dock Service Factory, Dock Service, Dock Service Group Registry와 같은 세 가지 서비스를 제공한다.

2) 데이타 브로커(Data Broker)

데이타 브로커는 Docking 작업 수행 시 계산 자원에서 요구하는 데이타에 대해 적절한 CDB 서비스를 선택하는 역할을 담당한다. 또한 동일한 데이타에 대한 복제본(replica)들이 존재하는 경우 최적의 복제본을 선택해 주는 기능도 제공한다. 이를 위해 데이타 브로커는 복제본 카탈로그(replica catalogue)와 데이타베이스 정보 서비스를 이용해야 한다. 본 연구에서 데이타 브로커는 가상 탐색 수행을 위해 화학 데이타를 제공할 데이타 자원을 선정할 때 네트워크 대역폭 정보를 이용한다. 본 연구에서 제안하는 데이타 브로커는 CDB Query Service Factory, CDB Query Service, CDB Service Group Registry와 같은 세 가지 서비스를 제공한다.

3) 복제본 카탈로그(Replica Catalogue)

복제본 카탈로그는 데이타 브로커가 적절한 CDB 자원 검색을 할 수 있도록 복제되어 있는 CDB에 대한 정보를 관리한다.

4) 데이타베이스 정보(Database Information) 서비스

데이타베이스 정보 서비스는 CDB에 대한 정보를 관리한다. 현재의 그리드 정보 서비스는 그리드 자원들의 장착된 메모리량, CPU 종류등과 같은 정적 정보와 CPU 사용량, 네트워크 대역폭의 변화 추이 등과 같은 동적 정보를 제공하고 있다. 이와 같이 기본적으로 제공되는 그리드 자원에 대한 정보 이외에 다킹 작업의 스케줄링 시 요구하는 정보는 가상 데이타베이스에 대한 정보 즉, 가상 데이타베이스의 물리적 정보, 가상 데이타베이스의 데이타 구조, 가상 데이타베이스에서 제공할 수 있는 분자의 종류 등과 같은 정보를 제공하여야 한다. 또한, 원격 호스트에서 수행되고 작업의 상태에 대한 정보들을 저장하고 있다가 데이타 브로커 서비스에서 질의(query)가 들어오면 그 정보를 제공해 줄 수 있는 기능이 필요하다. 복제본 카탈로그와 데이타베이스 정보 서비스는 데이타 브로커의 선택 알고리즘에 이용된다.

5) 계산 자원(Computation Resource)

각 계산 자원에서는 Dock 서비스가 있어 브로커에서 요구하는 다킹 작업을 처리하고 그 결과를 브로커에 돌려준다.

6) 데이타 자원(Data Resource)

그리드 컴퓨팅 환경에서는 다양한 종류의 파일 시스템, 데이타베이스, XML 데이타베이스, 계층 저장 시스템, 디렉토리 서비스를 제공한다. 이에 서비스 기반 가상 탐색 시스템에서는 데이타 브로커를 이용하여 데이타 자원을 사용하길 원하는 서비스들이 동일한 방법으로(uniform methods) 이질적인 데이타 자원들에 접근하여 사용할 수 있게 한다. 또한 서비스 기반 가상 탐색 시스템은 OGSA-DAI[15]와 OGSA-DQP[16]를 이용하여 계산 자원 노드에서 수행되는 다킹 서비스가 데이타 자원들에 쉽게 액세스할 수 있도록 한다.

5. 자원 브로커와 데이타 브로커 설계

5장에서는 본 논문에서 제안하는 자원 브로커와 데이타 브로커에 대해 설명한다.

5.1 자원 브로커

그리드에서 작업을 수행하기 위해서는 사용자가 제공하는 입력 데이타나 파라미터 값에 따라 큰 작업을 세부작업으로 나누는 과정이 필요하다. 따라서 그리드에서 다킹을 수행하기 위해서 자원 브로커는 사용자의 요구에 따라서 다킹 작업을 복제하거나 상세화하여 큰 덩어리의 다킹 작업을 작은 규모의 다킹 작업으로 세분화시키는 기능이 필요하다. 그리고 세분화된 각각의 다킹 작업에 대해서 작업의 특징이나 자원의 특징에 따라 매핑되어진 정보에 따라 작업의 흐름(workflow)을 작성하는 기능과 세분화된 작업을 정해진 원격 자원에 할당 시키는 기능을 제공해야 한다. 작업의 실행 파일, 환경 변수, 초기 입력 데이타 값 등을 원격 호스트에 전송하고 원격 호스트에서 올바르게 작업이 시작될 수 있도록 관리하여야 하고 세부 작업이 올바르게 수행되는가에 대한 관리 기능이 필요하다. 정보 서비스를 통하여 각 원격 호스트에서 수행되고 있는 세부작업들의 상태를 모니터링 하여 작업의 상태에 따라 신뢰성 있는 다킹 작업이 수행될 수 있도록 관리하여야 한다. 마지막으로 연산 결과의 수집과 수집된 연산 결과를 종합하여 사용자나 그리드 포탈에 전달하는 기능을 제공하여야 한다. 자원 브로커[17]는 MDS를 이용하여 계산 노드의 메모리, CPU 사용량 정보를 기반으로 가상 탐색을 수행할 계산 노드를 결정한다.

자원 브로커는 다음과 같은 세 가지 서비스를 제공한다.

1) Dock Service Factory

Dock Service Factory는 Dock Service를 생성하여 클라이언트가 다킹 서비스 자원과 상호작용할 수 있도록 한다.

2) Dock Service

Dock Service는 클라이언트와 상호작용하는 서비스이다. Dock Service는 분자 다킹을 실행하는 노드의 메타데이타와 자원 정보를 얻기 위해 Dock Service Group Registry 서비스에 접근한다. 이는 클라이언트로부터 받은 요구 작업을 분석하여 작업을 분할, 스케줄링하기 위함이다.

3) Dock Service Group Registry

Dock Service Group Registry는 실제 다킹 서비스를 생성하는 Dock Evaluation Service Factory를 등록하는 서비스이다. 이를 통하여 Dock Service에서는 어떤 Dock Evaluation Service Factory가 있는지 알 수 있고 각각의 Dock Evaluation Service Factory가 제공하는 기능과 자원에 대해 알 수 있다.

5.2 데이타 브로커

데이타 브로커는 분산되어 있는 화학 데이타베이스에서 사용자나 분자 다킹 어플리케이션이 요구하는 데이타 자원을 검색하는 기능과 검색된 데이타들을 실제 연산 작업과 매핑시키는 역할을 담당한다. 그리고 복제되어 존재하는 데이타들 중에서 효율적으로 분자 다킹이 실행될 수 있는 자원을 선택하는 기능을 제공한다. 마지막으로 데이타 브로커는 사용자 작업이 올바르게 수행될 수 있도록 선택된 데이타들을 실제 작업이 일어나는 그리드 노드로 이동시키고 작업을 위해 데이타를 할당할 수 있는 기능을 제공한다.

본 논문에서 제안하는 데이타 브로커는 다음 세 가지 서비스를 제공한다.

1) CDB Query Service Factory

CDB Query Service Factory는 CDB Query Service를 생성하여 Dock Service가 화학 데이타 자원과 상호작용할 수 있도록 한다.

2) CDB Query Service

CDB Query Service는 Dock Service와 상호작용하는 서비스이다. CDB Query Service는 데이타 노드의 자원 정보를 얻기 위해 CDB Service Group Registry 서비스에 접근한다. 이는 Dock Service로부터 받은 요구 작업을 분석하여 가장 적절한 데이타베이스를 연결하기 위함이다.

3) CDB Service Group Registry

CDB Service Group Registry는 실제 데이타베이스 서비스를 생성하는 Query Evaluation Service Factory를 등록하는 서비스이다. 이를 통하여 CDB Query Service에서는 어떤 Query Evaluation Service Factory가 있는지 알 수 있고 각각의 Query Evaluation Service

Factory가 제공하는 기능과 자원에 대해 알 수 있다.

6. 가상 탐색을 위한 서비스 설계

6장에서는 서비스 기반 가상 탐색 시스템을 구성하는 서비스들의 상세 스펙에 대해서 설명한다. 그림 3은 본 논문에서 가상 탐색 시스템을 위해 설계한 서비스들의 전체 구조를 보여준다.

1) Dock Service Group Registry(DSGR)

DSGR 서비스는 계산 노드에서 가상 탐색을 실행하는 Dock Evaluation Service Factory(DESF)의 등록을 담당하는 영속적인(persistent) 서비스이다. DSGR 서비스는 OGSI(Open Grid Services Infrastructure)의 PortType인 GridService, Notification Source, ServiceGroup, 그리고 ServiceGroupRegistration으로부터 파생된 DockService GroupRegistry PortType을 제공한다. DESF는 ServiceGroup Registration PortType을 이용하여 DSGR에 서비스를 등록하고 삭제할 수 있다. DSGR은 ServiceGroupEntry 서비스를 생성하여 등록된 서비스들 각각에 대한 존속 시간을 관리할 수 있다. 그리고 DSGR은 GridService PortType을 이용하여 등록된 서비스에 대한 정보를 질의할 수 있다. 또한 Notification-Source PortType을 이용하여 DESF에게 상태가 변경된 서비스의 정보를 통지할 수 있도록 한다. 위와 같이 DSGR이 제공하는 기능들을 이용하여 Dock Service는 가상 탐색 수행을 위한 스케줄링에 필요한 계산 노드의 정보를 검색할 수 있다.

2) Dock Service Factory(DSF)

DSF는 클라이언트가 Dock 서비스를 요청하는 영속하

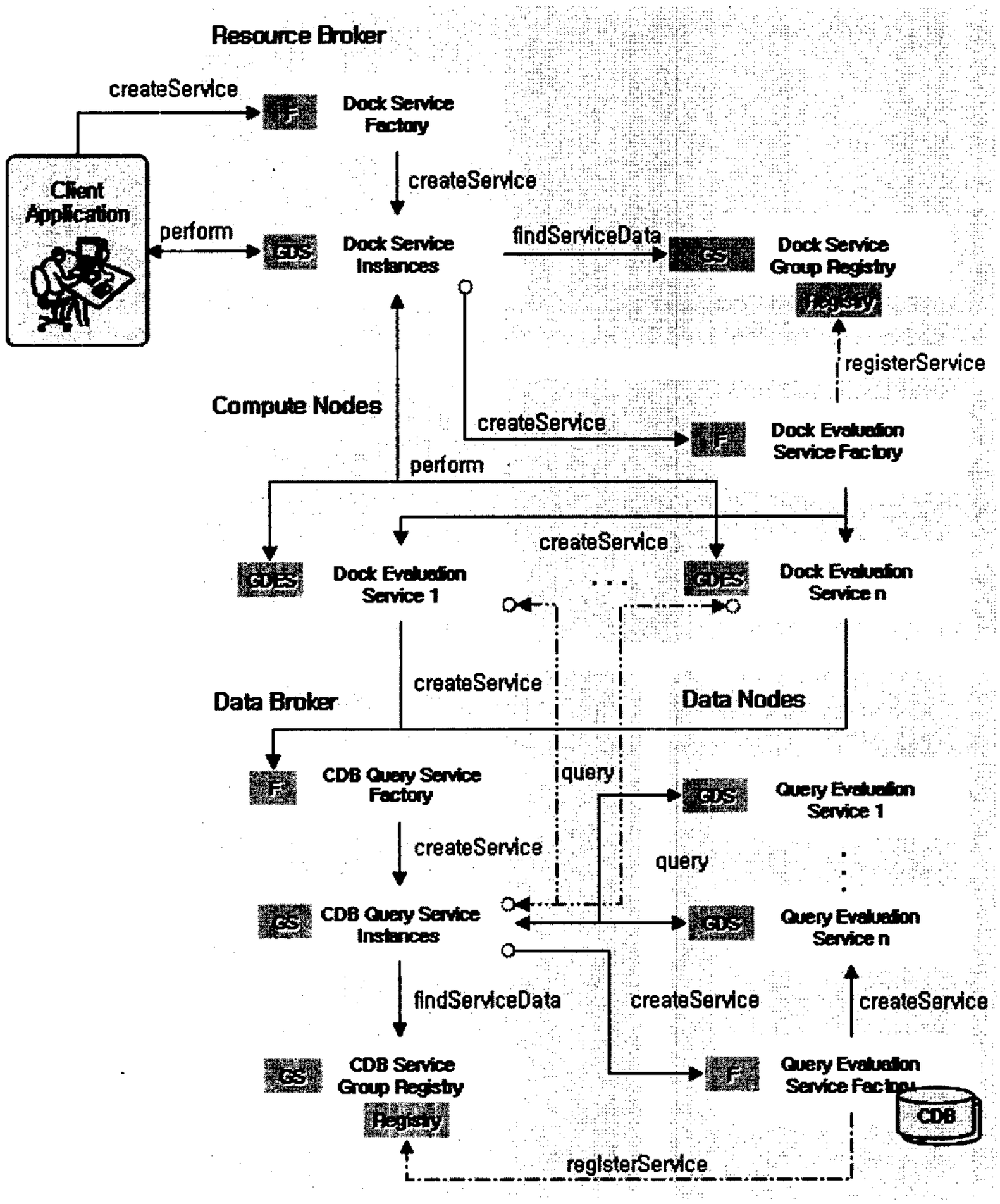


그림 3 가상 탐색 시스템을 위한 서비스들의 전체 구조

는 서비스이다. DFS는 OGSI PortType인 GridService와 Factory로부터 파생된 DockServiceFactoryPortType을 제공한다. DSF는 클라이언트 요청에 대해 하나의 Dock Service Instance (DSI)를 생성하는 서비스이다. DSI는 Dock Service Group Registry(DSGR)를 통하여 가장 적절한 계산 노드를 찾아낸 후, 계산 노드의 Dock Evaluation Service Factory에 서비스를 요청한다.

3) Dock Evaluation Service Factory(DESF)

DESF는 Dock Service Instance(DSI)가 요청한 수용체와 리간드에 대하여 실제로 다킹 어플리케이션을 실행하는 영속적인 서비스이다. DESF는 OGSI PortType인 GridService와 Factory로부터 파생된 DockEvaluationServiceFactory PortType을 제공한다. DESF는 DSI의 요청에 대해 하나의 Dock Evaluation Service Instance(DESI)를 생성하여 서비스한다. DESI는 CDB Query Service Factory(CDBQSF)를 통하여 리간드 정보를 조회한다.

4) CDB Service Group Registry(CDBSGR)

CDBSGR는 데이타 노드(Data Node)에서 CDB를 조회하는 기능을 가진 Query Evaluation Service Factory (QESF)를 등록하는 영속적인 서비스이다. CDBSGR은 OGSI의 PortType인 GridService, NotificationSource, ServiceGroup, ServiceGroupRegistration으로부터 파생된 CDBServiceGroupRegistryPortType PortType을 제공한다. Q-ESF는 ServiceGroupRegistration PortType을 이용하여 DSGR에 서비스를 등록하고 삭제할 수 있으며 CDBSGR은 ServiceGroupEntry 서비스를 생성하여 각각의 등록된 서비스에 대한 존속(duration) 시간을 관리할 수 있다. 그리고 CDBSGR은 GridService PortType을 이용하여 등록된 서비스에 대한 정보를 질의할 수 있다. 끝으로, CDBSGR은 Notification Source PortType을 이용하여 DESF가 상태가 변경된 서비스의 정보를 통지받을 수 있는 기능을 제공한다. 즉, CDBSGR이 제공하는 위와 같은 기능들을 이용하여 CDB Query Service는 가상 탐색을 위한 스케줄링에 필요한 데이타 노드의 정보를 검색할 수 있다.

5) CDB Query Service Factory(CDBQSF)

CDBQSF는 Dock Evaluation Service Instance (DESI)가 요청한 리간드 데이타의 정보를 조회하는 영속적인 서비스이다. CDBQSF는 OGSI PortType인 GridService와 Factory로부터 파생된 CDBQueryServiceFactory PortType을 제공한다. CDBQSF는 DESI의 요청에 대해 하나의 CDB Query Service Instance (CDBQSI)를 생성하여 서비스한다. CDBQSI는 Query Evaluation Service Factory(QESF)를 통하여 리간드 정보를 조회한다.

6) Query Evaluation Service Factory(QESF)

QESF는 CDB Query Service Instance(CDBQSI)가 요청한 리간드에 대해 실제로 데이타베이스를 조회하는 영속적인 서비스이다. QESF는 OGSI PortType인 GridService와 Factory, OGSA-DAI PortType인 GDSPortType으로부터 파생된 QueryEvaluation ServiceFactory PortType을 제공한다. QESF는 CDBQSI의 요청에 대해 하나의 Query Evaluation Service Instance(QESI)를 생성하여 서비스한다. QESI는 OGSA-DAI를 이용하여 데이타베이스를 조회한다.

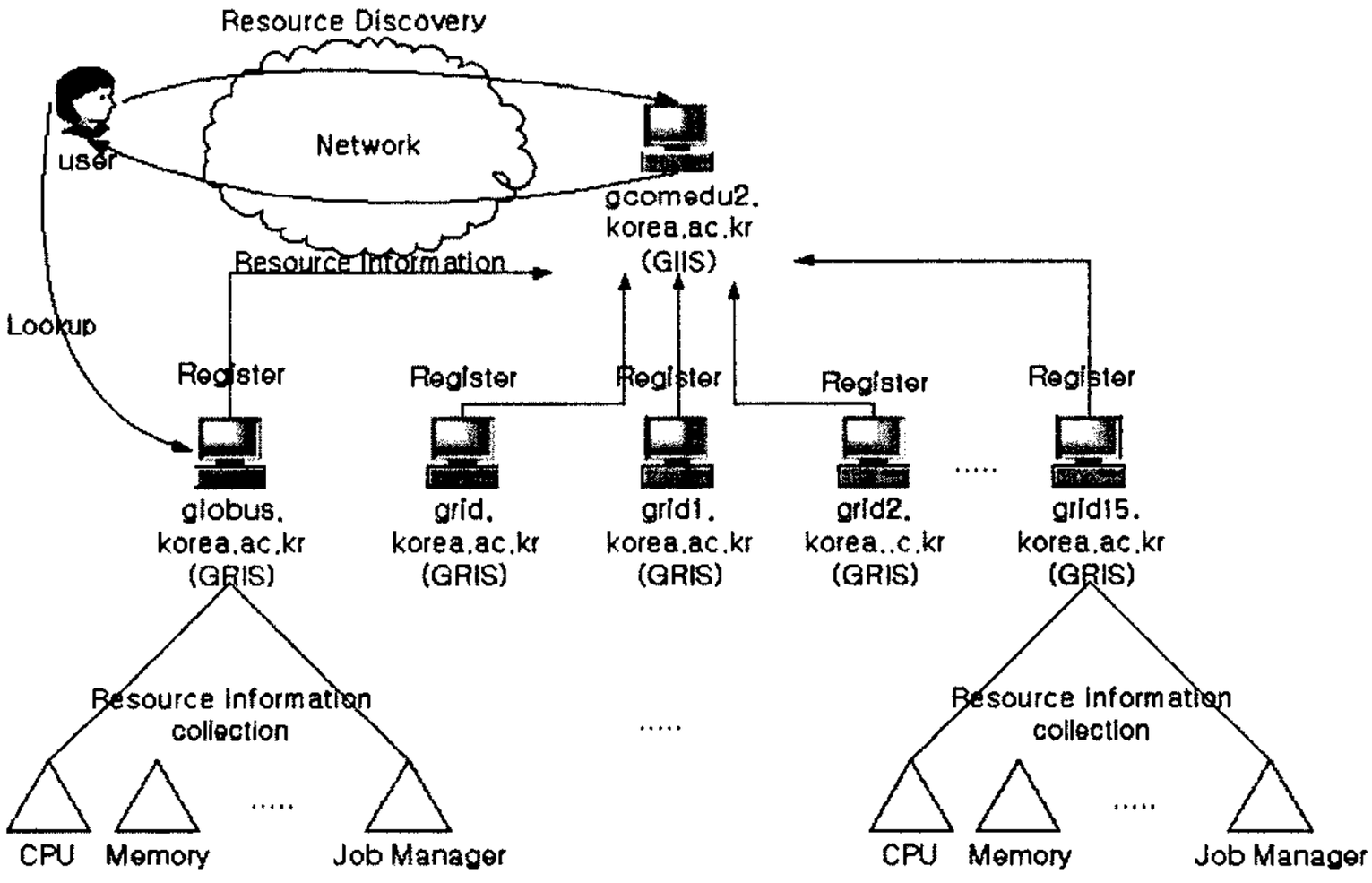


그림 4 KU 테스트베드의 MDS 구조

7. 구현 및 실험

본 논문에서는 서비스 기반 가상 탐색 시스템의 구현 및 실험을 위해 DOCK 5.0과 Globus 3.2를 이용하였고 서버 부분과 클라이언트 부분으로 나누어 자바로 구현하였다. 그리고 MySQL을 이용하여 3차원 화학데이타베이스를 구축하였고 가상 탐색 시스템의 각 서비스는 XML과 WSDL을 이용하여 정의하였다. 그리고 구현한 가상 탐색 시스템의 성능을 평가하기 위해 18개의 노드로 구성된 KU 테스트베드를 구축하였다. 그림 4는 KU 테스트베드의 MDS 구조를 보여준다[17].

7.1 구현 결과

1) 서버

그리드 서비스들은 그림 5와 같이 그리드 컨테이너에 의해 수행된다. 그림 5에서 빨간 박스로 표시된 서비스들이 본 연구에서 개발한 분자 다킹을 위한 서비스들이다.

그림 6은 클라이언트로부터 요청받은 수용체와 리간드의 분자 다킹이 수행된 결과를 보여준다.

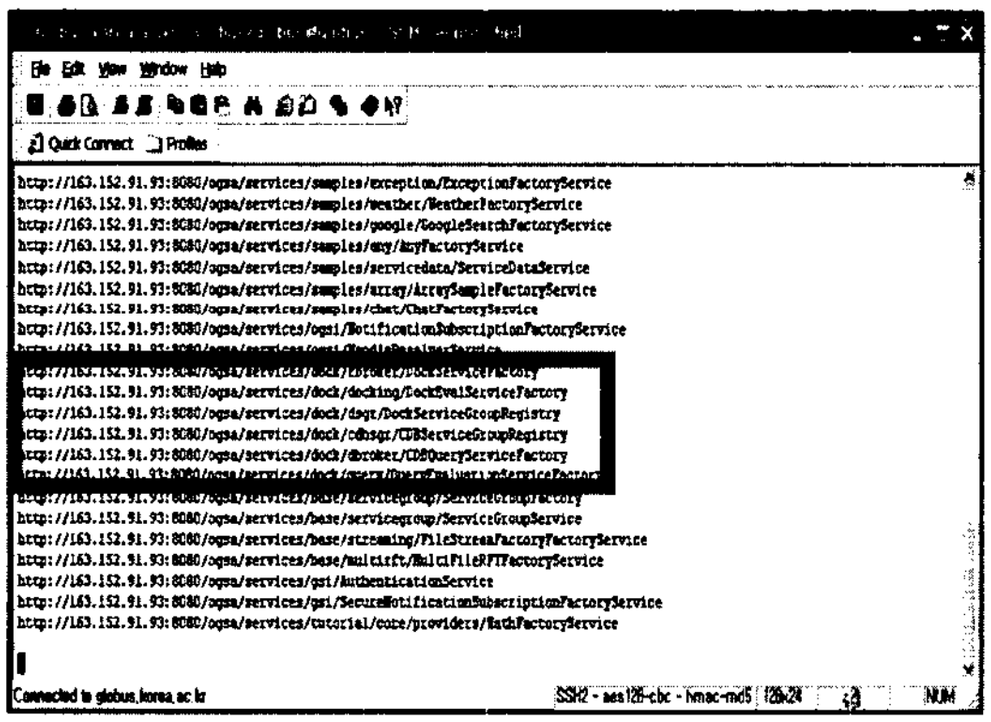


그림 5 컨테이너에 의한 서비스들의 수행 과정

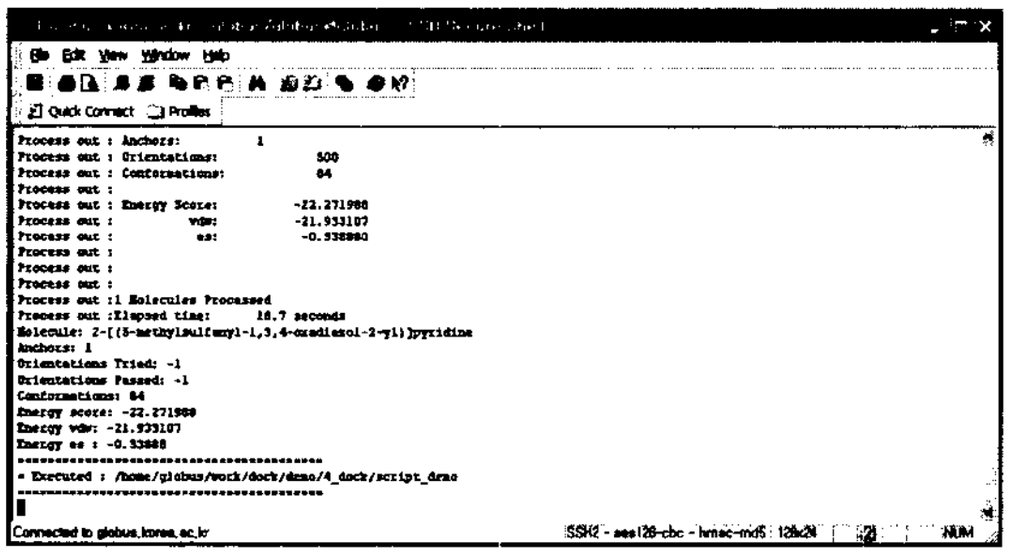


그림 6 분자 다킹 수행 결과

2) 클라이언트

본 연구에서는 사용자가 그리드에서 분자 다킹 서비스를 쉽게 사용할 수 있게 하기 위해서 그림 7과 같은 분자 다킹 클라이언트를 개발하였다. 다킹 클라이언트의

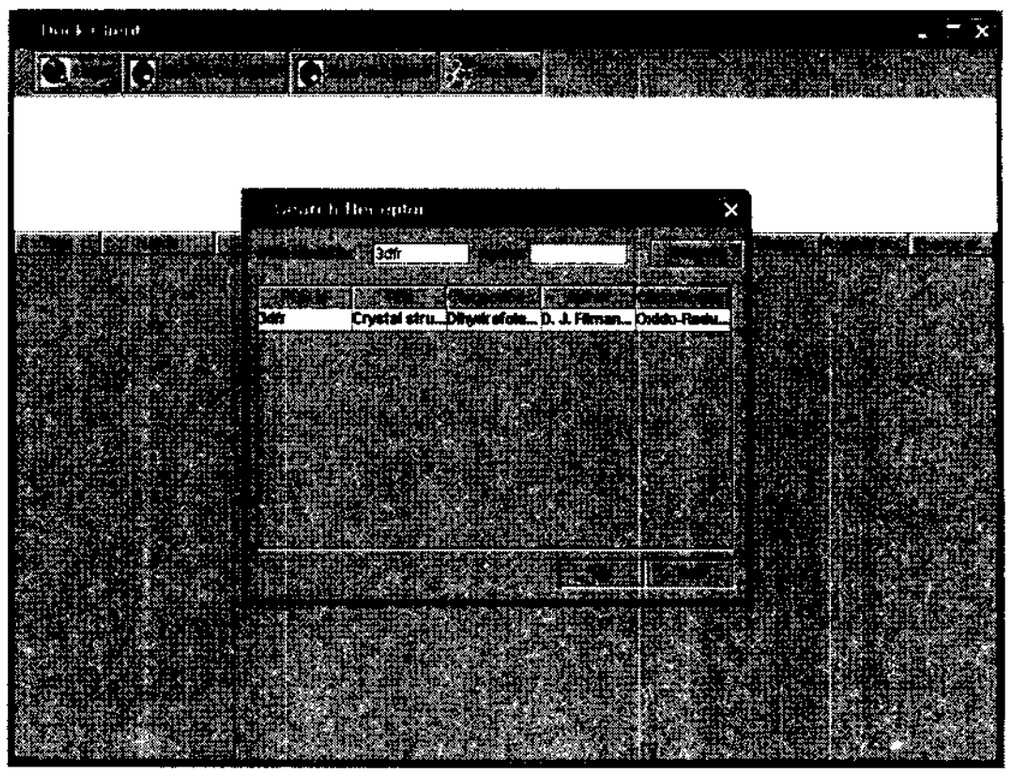


그림 7 수용체의 검색

메인 화면에는 다킹 서비스를 이용하는 사용자의 인증을 담당하는 'Login', 수용체 데이타를 찾기 위한 'Search Receptor', 리간드 데이타를 찾기 위한 'Search Ligand', 검색된 수용체와 리간드를 다킹하기 위한 'Docking' 메뉴로 구성된다. 그리고 현재 프로세스의 상태를 알 수 있는 상황창과 검색된 데이타의 결과값을 보여주기 위한 창으로 다킹 클라이언트가 구성된다. 그림 7은 수용체를 검색하는 화면이다. 수용체는 PDB ID와 저자로 검색할 수 있으며 만약 한 개 이상의 수용체가 결과값으로 나올 경우 테이블에서 선택할 수 있다.

그림 8은 다킹 클라이언트를 통해서 분자 다킹에 이용할 리간드를 검색하는 화면이다. 리간드의 검색을 위해서는 수용체의 경우와 달리 다양한 검색 조건을 이용하여 리간드를 검색할 수 있다. 수용체와 마찬가지로 검색된 리간드가 한 개 이상인 경우 테이블에서 다킹할 리간드를 선택할 수 있다. 이렇게 선택된 리간드는 메인 화면에 표시된다.

그림 9는 선택된 수용체와 리간드 사이의 분자 다킹

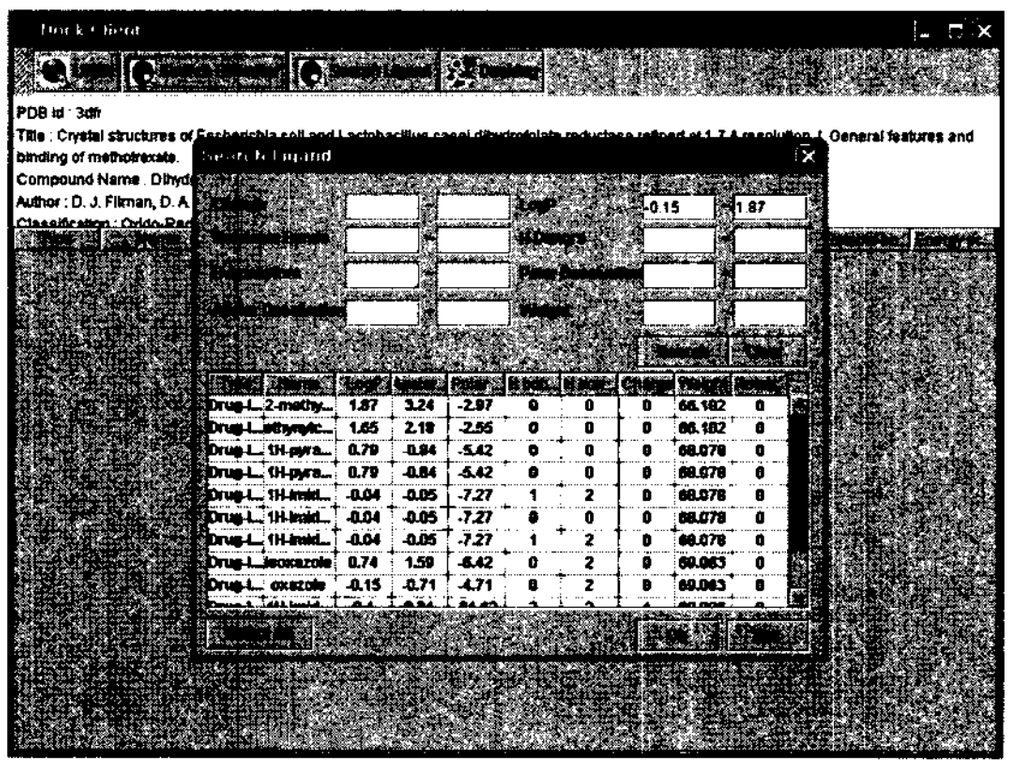


그림 8 리간드의 검색

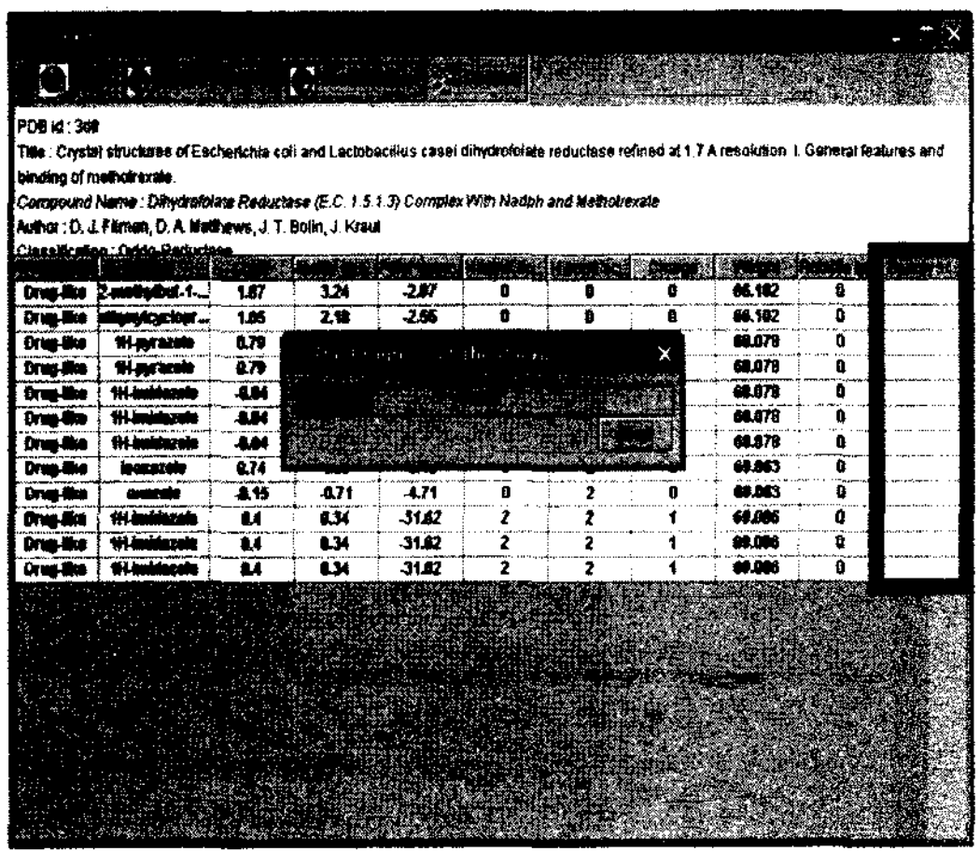


그림 9 선택된 수용체와 리간드의 분자 다킹 수행 과정

을 수행하는 화면이다. 분자 다킹을 통해 계산된 스코어 값이 그림 9의 빨간 박스내에 에너지 값으로 표시되며, 테이블 칼럼을 클릭하여 실험된 리간드들을 정렬하고 에너지 값이 작은 리간드들을 대상으로 직접 화학 실험을 수행하게 된다.

7.2 실험 결과

그림 10은 임의 선택(random selection), 최상 선택(best selection), 본 논문에서 제안하는 최적 선택 등 세 가지 다른 자원 선택 방법에 따른 분자 다킹의 실행 시간을 비교한 것이다. 자원의 임의 선택 방법은 사용자가 무작위로 자원을 선택하여 다킹 작업을 수행시킨 실행 시간이고, 최상 선택 방법은 최상의 컴퓨팅 성능을 가지는 노드들을 선택하여 다킹 작업을 수행시킨 실행 시간이다. 최적 자원 선택 방법은 자원들의 QoS와 예측된 실행 시간을 고려하는 자원 선택 알고리즘을 이용하여 다킹 작업을 수행시킨 실행 시간이다. 실험을 위해 본 논문에서는 하나의 타켓 수용체에 대해 10,000개의 리간드 분자를 대상으로 분자 다킹 작업을 수행시켰다.

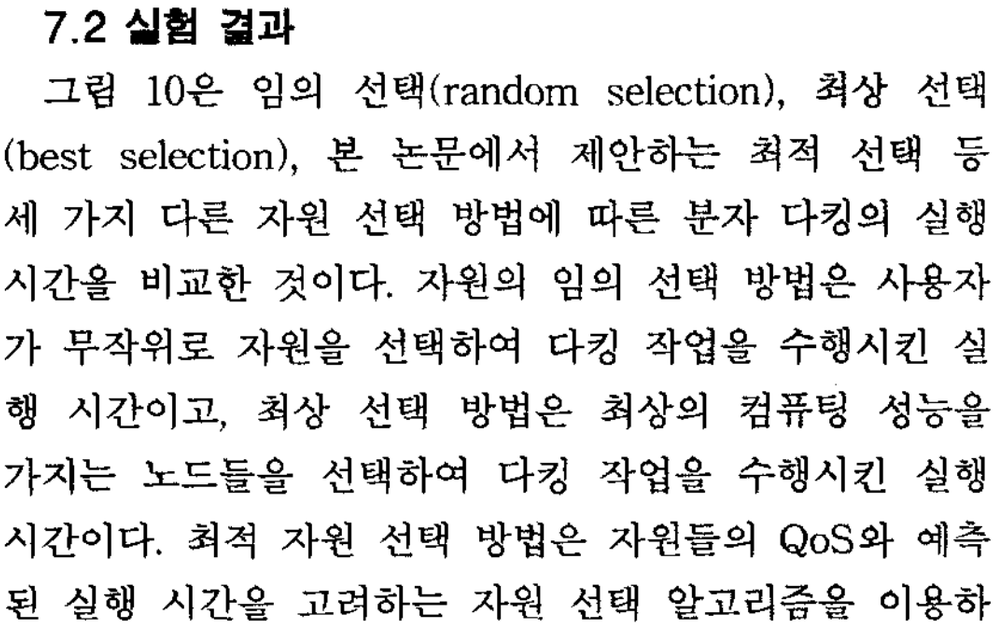


그림 10 자원 선택 방법에 따른 실행 시간 비교

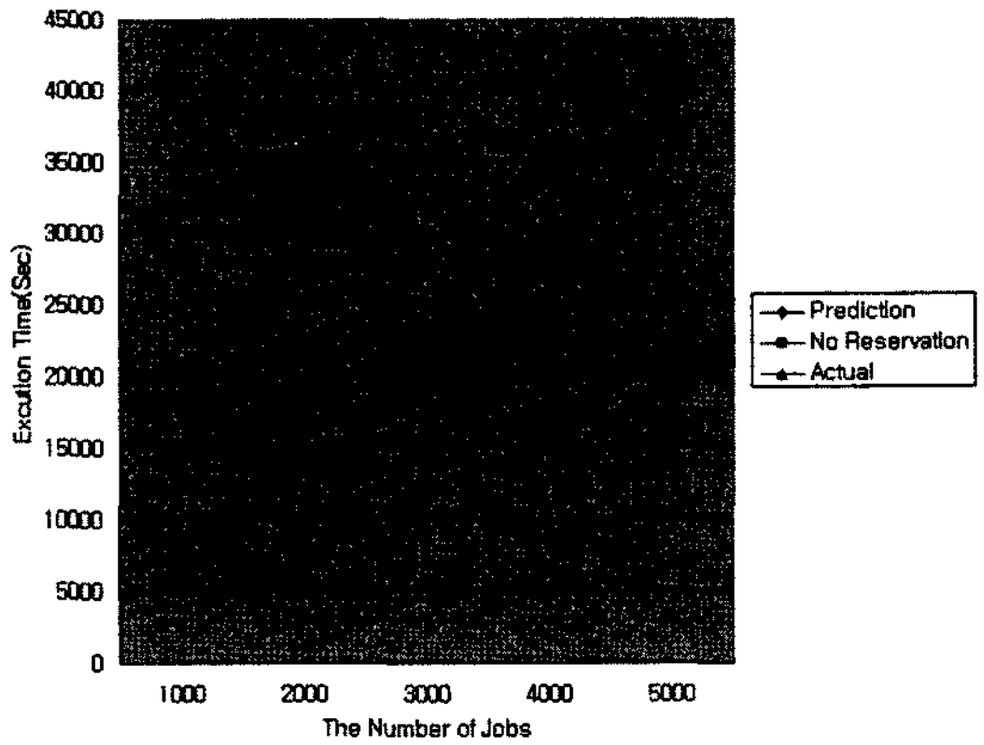


그림 11 작업 수 증가에 따른 실행시간 비교

실험 결과 그림 10과 같이 본 논문에서 제안하는 최적 자원 선택 방법이 임의 선택 방법이나 최상 선택 방법보다 좋은 성능을 보였다. 또한 그림 10에서 분자 다킹 작업을 수행하는 노드의 수가 증가할수록 실행 시간이 감소하였다.

그림 11은 분자 다킹 작업의 수가 증가할 때 세 가지 작업 실행 시간을 비교한 것이다. 첫 번째 실행 시간은 본 논문에서 제안한 최적 자원 선택 브로커가 계산한 분자 다킹 작업의 예측 실행 시간이고, 두 번째 실행 시간은 최적 자원 선택 브로커가 자원 예약(resource reservation)없이 최적 자원 선택 브로커가 선택한 자원에서 분자 다킹 작업을 수행한 실행 시간이다. 세 번째 실행 시간은 최적 자원 선택 브로커가 자원 예약을 사용하여 분자 다킹 작업을 수행한 실제 실행 시간이다. 실험 결과 최적 자원 선택 브로커가 예측한 실행 시간과 실제 실행 시간의 평균 차이는 예측한 실행 시간의 3%에 불과하였다.

본 논문의 구현 및 실험 결과는 본 논문에서 제안하는 그리드 서비스를 기반한 가상 탐색 시스템이 실제 실행 시간과 거의 정확하게 실행 시간을 예측하여 최적 자원을 선택하고, 이러한 최적 자원의 선택을 통해 기존의 하나의 서버에서 DOCK 프로그램을 실행하는 방법보다 실행 시간을 단축하여 연구개발 기간을 단축하는 것을 보여준다.

8. 결론 및 향후 과제

가상 탐색은 컴퓨터를 이용하여 대규모의 화학분자 데이타베이스의 화학분자 데이타들을 표적 수용체에 다킹(docking)시켜 1만-2만개의 화학분자만을 스크리닝해 주기 때문에, 신약, 신소재, 고분자의 개발에 있어서 분자 모델링의 이용은 연구 개발 비용을 절감하고 연구 기간을 단축시킬 수 있는 매우 중요한 기술이다. 기존의

가상 탐색 어플리케이션들은 슈퍼컴퓨터나 단일 클러스터, 단일 워크스테이션 등을 이용하여 작업을 수행하도록 설계되고 구현되었다. 하지만 슈퍼컴퓨터를 이용한 분자 모델링은 너무 많은 비용이 든다는 문제점이 있고, 단일 클러스터나 워크스테이션을 이용한 가상 탐색은 오랜 수행 시간이 요구되는 문제점을 가지고 있다. 이에 본 연구에서는 대규모의 데이타 집약적인 연산을 지원하는 그리드 컴퓨팅 기술을 이용하여 서비스 기반 가상 탐색 시스템을 개발하였다. 이를 위해 본 연구에서는 MySQL을 이용하여 Protomer라는 3차원 화학분자 데이타베이스를 구축하였고 자원 브로커, 데이타 브로커, 복제본 카탈로그, 데이타베이스 정보 서비스, 계산 자원과 데이타 자원으로 구성된 가상 탐색 시스템을 설계하였다. 또한 효율적인 분자 다킹 서비스를 제공하기 위해 자원 브로커와 데이타 브로커를 설계하고 가상 탐색을 위한 각종 서비스들을 제안하였다. 그리고 DOCK 5.0과 Globus 3.2를 기반으로 서비스 기반 가상 탐색 시스템을 구현하였다. 본 연구에서 구현한 가상 탐색 시스템은 그리드에서 서비스를 이용한 가상 탐색 시스템의 구현에 관한 최초의 연구라는 점에서 중요한 의미를 가지며 신약 개발이나 신소재 개발 과정에서 연구 개발 기간을 단축하고 개발 비용을 절감할 수 있다는 장점을 갖는다.

향후 연구 과제는 다양한 테스트 데이타를 이용하여 본 연구에서 개발한 가상 탐색 시스템과 기존의 여러 분자 모델링 어플리케이션의 수행 성능을 비교하는 실험을 실시하는 것이다. 그리고 신약 개발이나 신소재 개발 분야의 연구자들이 본 연구에서 개발한 분자 모델링 어플리케이션을 편리하게 사용하기 위해서 PSE(problem solving environment)를 설계하고 구현하는 것이다.

참 고 문 헌

[1] Shoichet, Bodian, and Kuntz, Molecular docking using shape descriptors, Journal of Computational Chemistry, Vol.13, No.3, pp. 380-397, 1992.

[2] Morris, Goodsell, Huey, and Olson, Distributed automated docking of flexible ligands to proteins: Parallel applications of AutoDock 2.4, Journal of Computer-Aided Molecular Design, Vol.10, pp. 293-304, 1996.

[3] H. Claußen, C. Buning, M. Rarey, and T. Lengauer, FlexE: Efficient Molecular Docking into Flexible Protein Structures, Journal of Molecular Biology, Vol.308, pp. 377-395, 2001.

[4] S. Makino and I.D. Kuntz. Automated Flexible Ligand Docking Method and Its Application for Database Search, J. Comp. Chem. Vol.18, No.14, pp. 1812-1825, 1997.

[5] C.M. Venkatachalam, X. Jiang, T. Oldfield, and M. Waldman, LigandFit: A Novel method for the Shape-Directed Rapid Docking of Ligands to Protein Active Sites, J. Mol. Graphics Modeling, 2002.

[6] D.W. Ritchie, Evaluation of Protein Docking Predictions Using Hex 3.1 in CAPRI Rounds 1 and 2, PROTEINS: Struct. Funct. Genet. Vol.52, No.1, pp. 98-106, 2003.

[7] I. Foster, C. Kesselman and S. Tuecke, The Anatomy of the Grid : Enabling Scalable Virtual Organizations, International Supercomputer Applications, Vol.15, No.3, 2001.

[8] Ian Foster, and Carl Kesselman, The Grid : Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers, 1998.

[9] Thomas Sandhol and Jarek Gawor, Globus Toolkit 3 Core - A Grid Service Container Framework, Globus Project, 2003.

[10] I. Foster, C. Kesselman, J. Nick and S. Tuecke, The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Global Grid Forum, 2002.

[11] Rajkumar Buyya, Kim Branson, Jon Giddy, and David Abramson, The Virtual Laboratory: a toolset to enable distributed molecular modelling for drug design on the World-Wide Grid, Concurrency and Computation: Practice and Experience, Vol.15, No.1, pp. 1-25, 2003.

[12] C. Ferrari, Concettina Guerra, G. Canotti, A grid-aware approach to protein structure comparison, Journal of Parallel and Distributed Computing Vol. 63, Issue 7-8, pp. 728-737, 2003.

[13] H. Casanova, F. Berman, Parameter Sweeps on the Grid with APST, chapter 33 in Grid Computing: Making the Global Infrastructure a Reality, Wiley Publisher, Inc., 2002.

[14] Wu, Tai, Murdock, Ng, Johnston, Fangohr, Jeffreys, Cox, Essex, and Sansom, BioSimGRID: a distributed database for biomolecular simulations, Proc. UK e-Science All Hands Meeting, pp. 412-419, 2003.

[15] Mario Antonioletti, et al., Experiences of Designing and Implementing Grid Database Services in the OGSA-DAI project, Available: http://www.ogsadai.org.uk/

[16] M.Nedim Alpdemir, Arijit Mukherjee, Anastasios Gounaris, Norman W.Paton, Paul Watson, Alvaro A.A.Fernandes, Jim Smith, OGSA-DQP: A service-Based Distributed Query Processor for the Grid, Proceedings of UK e-Science, 2003.

[17] HwaMin Lee, KwangSik Chung SungHo Chin, JongHyuk Lee, DaeWon Lee, Sungbin Park, HeonChang Yu, A Resource Management and Fault Tolerance Services in Grid Computing, Journal of Parallel and Distributed Computing, Vol.65, Issue 11, 2005.

이 화 민

2000년 고려대학교 컴퓨터교육과 졸업(이학사). 2002년 고려대학교 대학원 컴퓨터교육학과 졸업(교육학석사). 2006년 고려대학교 대학원 컴퓨터교육학과 졸업(이학박사). 2006년~2007년 특허청 전자상거래심사팀 통신사무관. 2007년~현재 순천향대학교 컴퓨터학부 전임강사. 관심분야는 분산 컴퓨팅, 그리드 컴퓨팅, 결함포용, 자원 스케줄링 등

진 성 호

2002년 고려대학교 컴퓨터교육과 졸업(이학사). 2004년 고려대학교 대학원 컴퓨터교육학과 졸업(교육학석사). 2004년~현재 고려대학교 대학원 컴퓨터교육과 박사과정. 관심분야는 분산 컴퓨팅, 그리드 컴퓨팅, 자원 관리 등

이 종 혁

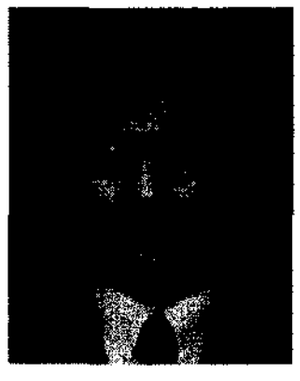
2004년 고려대학교 컴퓨터교육과 졸업(이학사). 2006년 고려대학교 대학원 컴퓨터교육학과 졸업(이학석사). 2006년~현재 고려대학교 대학원 컴퓨터교육과 박사과정. 관심분야는 분산 컴퓨팅, 그리드 컴퓨팅, 자원 관리 등

이 대 원

2001년 순천향대학교 전자공학과 졸업(공학사). 2004년 고려대학교 대학원 컴퓨터교육학과 졸업(교육학석사). 2004년~현재 고려대학교 대학원 컴퓨터교육과 박사과정. 관심분야는 분산 컴퓨팅, 그리드 컴퓨팅, 자원 관리 등

박 성 빈

1990년 고려대학교 전산과학과(이학사) 1993 University of Southern California(전산학 석사). 1999 University of Southern California(전산학 박사). 2003~현재 고려대학교 컴퓨터교육과 부교수 관심분야는 하이퍼텍스트, 알고리즘, 계산이론, 컴퓨터교육 등



유 헌 창

1989년 고려대학교 이과대학 컴퓨터학과 졸업(이학사). 1991년 고려대학교 대학원 컴퓨터학과 졸업(이학석사). 1994년 고려대학교 대학원 컴퓨터학과 졸업(이학박사). 1995년~1998년 서경대학교 컴퓨터공학과 조교수. 1998년~현재 고려대학교 컴퓨터교육과 교수. 관심분야는 분산 시스템, 그리드 컴퓨팅, 모바일 컴퓨팅, 결함포용시스템 등