그리드에서 Molecular Docking 어플리케이션 설계

진성호*O 이화민* 이대원* 이종혁* 박성빈* 유헌창*
* 고려대학교 대학원 컴퓨터교육과
{wingtop^O, zelkova, daelee, spurt, psb, yuhc}@comedu.korea.ac.kr

A Design of Molecular Docking Application in Grids

SungHo Chin HwaMin Lee DaeWon Lee JongHyuk Lee Seongbin Park HeonChang Yu
* Dept. of Computer Science Education, Korea University

요 약

Molecular modelling은 시뮬레이션을 통해 온도, 압력 등과 같은 분자 운동에 영향을 미칠 수 있는 요소를 설정한 후 분자의 움직임을 관찰하는 방법으로 신약, 신소재, 고분자 개발에 있어서 연구 개발 기간을 단축하는 효과적인 방법이다. 기존의 molecular modelling 어플리케이션들은 슈퍼 컴퓨터나 단일 클러스터를 이용하여 작업을 수행하도록 설계되어, 비용과 성능 측면에서 문제점을 가지고 있다. 1990년대 중반 지리적으로 분산되어 있는 광범위한 자원들을 공유하여 장기간 소요되는 컴퓨팅 작업의 성능 향상 및 비용 절감을 목적으로 하는 그리드(grid)가 등장하였다. 이에 본 연구에서는 효율적이면서도 저비용을 갖는 molecular modelling 어플리케이션 개발을 위해 그리드를 기반으로 최적 자원 선택 브로커를 이용하는 molecular docking 어플리케이션을 제안한다. 이를 위해 우리는 molecular docking을 수행하는 그리드 환경의 계층 구조를 설계하고 효율적 작업 수행을 위한 최적 자원 선택 브로커를 설계하였다. 그리고 그리드 환경에서 molecular docking 어플리케이션의 효과적인 수행을 위해 molecular docking 연산 모델을 정의하고 필요한 molecular docking 어플리케이션의 요소들을 설계하였다.

1. 서론[1)]

Molecular modelling은 실제 실험을 통하지 않고 시뮬레이션을 통해 온도, 압력 등과 같은 분자 운동에 영향을 미칠 수 있는 요소를 설정한 후 분자의 움직임을 관찰하는 방법이다[1]. 신약, 신소재, 고분자의 개발에 있어서 molecular modelling의 이용은 연구 개발 기간 단축을 가져올 수 있기에 상당히 중요한 문제가 아닐 수 없다. 1980년대부터 컴퓨터와 통신 기술의 발전을 통해 생화학, 의학, 약학 등 다양한 분야에서 신물질의 발견 과정은 CAMD(Computer Assisted Molecular Design)를 이용하고 있다. Molecular modelling의 세부 분야는 molecular graphics, molecular docking, computational chemistry, statistical modelling, molecular 데이터와 정보 관리 및 검색 등으로 분류되고 모든 과정들이 컴퓨터를 통해 이루어지기 때문에 CAMD라고 부른다[1]. CAMD는 분자들의 구조, 행동과 상호 작용들을 컴퓨터로 시뮬레이션을 수행하고, 그래픽을 이용한 분자 모의실험을 통해 다양한 가능성들을 가상적으로 시도해 볼 수 있게 한다. 그리고 다양한 가능성 중 가장 확률이 높은 것을 추출하여 중점적으로 실제 실험이 가능할 수 있도록 해준다.

기존의 molecular modelling 어플리케이션들은 슈퍼컴퓨터나 단일 클러스터, 혹은 단일 워크스테이션을 이용하여 작업을 수행하도록 설계 구현되었다. 하지만 슈퍼컴퓨터를 이용한 molecular modelling은 너무 많은 비용이 든다는 문제점이 있고, 단일 클러스터나 워크스테이션을 이용한 molecular modelling은 성능적인 측면에서 오랜 수행 시간이 요구되는 문제점을 가지고 있다. 따라서 본 논문에서는 molecular modeling의 세부 분야 중에서 molecular docking을 수행하는 어플리케이션을 수정하여 효율적이고 저비용을 갖는 그리드 기반 molecular docking 어플리케이션을 설계하고자 한다.

그리드(grid)는 1990년대 중반 등장한 개념으로 슈퍼컴퓨터, PC, 저장 시스템, 데이터베이스, 데이터 소스, 다른 기관 소유의 특성화된 장치 등 지리적으로 분산되어 있는 광범위한 자원들을 공유하여 장시간 소요되는 컴퓨팅 작업의 성능의 향상 및 비용 절감을 목적으로 하고 있다[2]. 최근에 들어 그리드 컴퓨팅에 관한 많은 연구가 진행되어 왔고 기존의 많은 연구들이 주목할만한 성과를 이루어내면서 그리드는 차세대 인터넷의 핵심을 이룰 기술로 기대를 모으고 있다. 그리드 컴퓨팅이 다루는 문제로는 기상관측, 수치해석, 핵물리 실험 등 대규모 컴퓨팅 능력을 요구하는 다양한 과학 분야에 걸쳐 있다[3].

Molecular docking 또한 거대한 컴퓨팅 능력을 요구하는 대표적인 과학 어플리케이션[3]으로, 대규모의 데이터 집약적인 연산을 지원하는 그리드와 같은 고성능 컴퓨팅 기술을 이용하여 master-worker 병렬 어플리케이션으로 구현될 수 있다. 특히, 본 연구에서 다루고자 하는 molecular docking은 문제 해결을 위한 연산 수행 시간이 짧게는 몇 일에서 길게는 몇 년 까지 걸리는 대규모 작업이다[1]. 긴 연산시간을 요구하는 작업이 그리드를 이용하여 효율적인 작업 수행이 이루어지기 위해서는 작업과 대상 자원을 연결해 줄 수 있는 자원 브로커가 필수적이다. 따라서 본 연구에서는 최적 자원 선택 브로커와 화학 데이터베이스 브로커를 이용하여 그리드에서 효율적인 작업 수행 환경을 제공함으로써 molecular docking 어플리케이션의 성능을 향상시키고자 한다.

2. 관련 연구

그리드 컴퓨팅은 네트워크로 연결된 가상의 슈퍼 컴퓨팅 시스템을 사용자에게 제공함으로써 지금까지 해결하지 못했던 대규모의 문제를 해결할 수 있도록 지원한다. 특히, 계산 그리드(computational grid)는 지리적으로 분산되어진 컴퓨팅 리소스들을 데이터 집약적인 문제나 과학, 공학, 경영 등에서 다루는 여러 가지 큰 문제를 해결하기 위해서 사용된다. 그러나 molecular modeling을 위한 어플리케이션 개발자들의 입장에서 볼 때, 이와 같은 환경에서의 문제를 풀기 위해서는 자원 관리나 자원의 스케줄링과 같은 문제를 극복해야 한다. [4]에서는 현재의 그리드 기술들을 통합하여 지리적으로 분산된 자원을 이용하여 신약 개발을 위한 molecular docking이 가능한 virtual laboratory라는 환경을 제공한다. 또한 기존의 molecular docking 어플리케이션을 분산 어플리케이션에서 구

1) "이 논문은 2003년도 한국학술진흥재단의 지원에 의하여 연구되었음." (KRF-20003-D00469)

동되게 하기 위해서 Nimrod-G 파라미터 명세 언어[5]를 이용하였다. 그리고 월드 와이드 그리드 환경에서의 어플리케이션 수행 스케쥴링 문제 해결을 위하여 CDB(Chemical DataBase) 브로커와 Nimrod-G 자원 브로커를 이용하였다. 하지만, [4]에서는 molecular modelling에 관한 연구보다는 Nimrod-G 라는 자원 브로커의 스케쥴링 기법과 성능 측정에 관해 중점을 두어 연구가 이루어 졌다. 그러므로 이 [4]에서 제시하는 기법들을 그리드를 이용한 molecular docking 소프트웨어를 개발을 위해 사용하는 것은 적합하지 않다.

컴퓨터를 이용한 molecular docking에 관한 지원과 연구가 활발히 이루어짐에 따라 컴퓨터를 이용한 molecular docking을 지원해 줄 수 있는 각종 소프트웨어들이 제작되어 배포되었다. 이러한 소프트웨어들은 Unix, 윈도우즈, MacOS 등의 운영체제를 사용하는 단일 노드에서 수행되는 소프트웨어들이 대부분이다. 하지만, 향후 바이오 기술과 나노 기술의 발전으로 molecular docking에 사용되는 데이터의 양은 단일 노드가 가지고 있는 저장 장치의 한계를 넘어 설 것이다. 그리고 현재의 소프트웨어들은 단일 노드의 연산 능력만을 이용하여 실험이 진행됨으로 많은 양의 데이터 처리나 복잡한 연산을 효율적으로 수행하지 못한다.

3. Molecular Docking

Molecular docking은 특정 분자들의 결합을 예측하는 과정이다. Molecular docking 과정에서 보통 고정적으로 사용되는 분자를 receptor라고 하며 이 분자는 실험이 이루어지는 동안 변하지 않고 사용되게 된다. 그리고 receptor와 결합하기 위해서 화학 데이터베이스에서 검색한 분자들을 ligand라 한다. 즉, molecular docking은 receptor와 ligand 사이의 결합을 예측하는 과정이라고 정의할 수 있다.

Molecular docking 문제 해결을 위해서 다음 두 단계의 과정을 거친다. 먼저 주어진 두 분자(receptor와 ligand)들의 3차원 좌표 및 surface complementarity를 이용하여 기하학적으로 가능한 결합 구조들을 찾는 기하학적 단계와 결합된 구조들 중 에너지가 낮은 상태의 구조들을 골라내는 에너지 기반 단계이다. 이를 통해 두 분자가 결합하는 구조를 찾을 수 있다. Ligand와 receptor의 어떤 부분에 결합할 수 있을지를 알려주는 active site 정보가 있을 경우 기하적인 단계에서 이루어지는 공간 탐색에 도움이 된다.

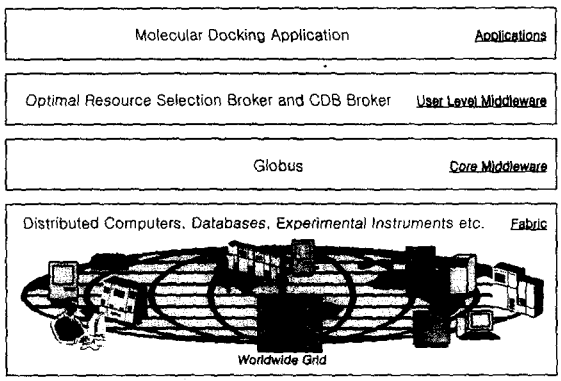
Molecular docking을 위해 개발된 프로그램들은 많이 있는데 그들 중 가장 보편적으로 이용되는 UCSF의 DOCK[6]은 receptor에서 ligand가 어떤 방향으로 결합되는가를 찾는 루틴들과 ligand의 방향을 평가하는 루틴들로 구성되어 있다. Sphgen 프로그램은 receptor에서 active site를 찾은 후 그곳을 채우는 sphere center들을 생성한다. DOCK 프로그램에서는 Sphgen에서 만들어진 sphere들을 ligand atom들과 매치시키면서 그들의 적합정도를 평가한다. 또한 DOCK 프로그램은 에너지가 최저인 상태를 찾는다.

4. 그리드 기반 Molecular Docking 어플리케이션 설계

Molecular docking은 기술선진국에서 신물질 개발연구나 신약 개발, 신소재 개발 등 다양한 분야에서 핵심적인 기술로 널리 활용되어 획기적인 기여를 하고 있는 기술이다. Molecular docking의 주요 활용 분야 중 신약 발견의 경우, 새로운 치료약의 시장 출시를 위해서 실험실에서 다양한 합성과 실험에 15년이나 소요될 만큼 광대한 작업 시간이 필요하다[1]. Molecular docking은 긴 시간이 소요되는 작업을 슈퍼컴퓨터나 병렬컴퓨터 등을 이용하여 짧은 시간 안에 결과를 얻게 해주는 엄청난 시간 단축의 이익을 창출한다.

계산 그리드는 대규모 작업을 수행하기 위한 가상 조직(VO)을 만들기 위해 다른 조직에 의해 소유된 슈퍼컴퓨터, 저장 시스템, 데이터베이스, 각종 과학 및 의학 전문 장치들을 포함하여 지리적으로 분산되어 있는 자원들의 공유를 가능하게 한다. 이러한 계산 그리드의 컴퓨팅 파워는 거대한 데이터 집합들을 다루는 대규모 문제를 해결하게 해준다. 따라서, 그리드 컴퓨팅을 이용하면 molecular docking과 같이 데이터 집합적이고 고성능의 연산 자원을 요구하는 어플리케이션을 효율적으로 수행할 수 있다.

4.1 Molecular docking 어플리케이션을 위한 그리드 계층 구조 설계



<그림 1> Molecular Docking을 위한 그리드 계층 구조

<그림 1>은 본 연구에서 재구조화하는 molecular docking 어플리케이션의 계층 구조를 보여준다. 가장 하위의 Fabric 계층은 그리드에서 제어되는 공유 자원들을 제공하는 역할을 담당한다. 공유되는 자원은 분산된 컴퓨터 풀, 클러스터 컴퓨터, 슈퍼컴퓨터, 데이터베이스, 저장 시스템, 각종 실험 장치 등 다양하다. 두 번째 Core Middleware 계층은 분산된 이기종 컴퓨팅 자원들을 하나의 가상 슈퍼컴퓨터처럼 사용할 수 있는 미들웨어이다. 본 연구에서는 그리드 응용 서비스 개발에서 가장 보편적으로 사용하고 있는 글로버스를 이용한다. 세 번째 계층인 User Level Middleware는 본 연구에서 재구조화하는 molecular modelling 어플리케이션의 수행과 성능 향상을 위해 필요한 최적 자원 선택 브로커(optimal resource selection broker)와 화학 데이터베이스 브로커(CDB broker)로 구성된다. 마지막 계층은 가상 조직내에서 동작하는 사용자 어플리케이션 계층으로 본 연구에서 설계하고자 하는 molecular docking 어플리케이션이 위치하게 된다.

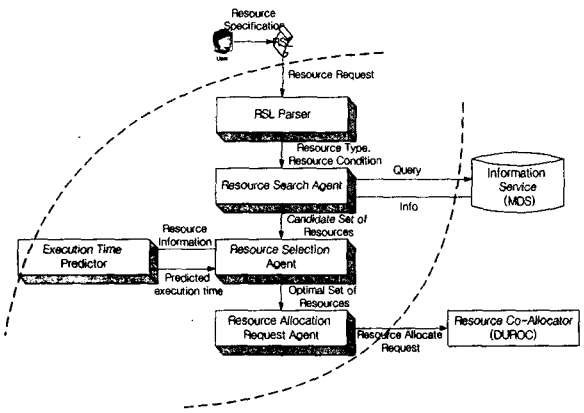
4.2 최적 자원 선택 브로커 설계

<그림 2>는 본 연구에서 사용하는 최적 자원 선택 브로커의 구조를 보여주는 것이다. 최적 자원 선택 브로커는 작업이 가장 효율적으로 수행 될 수 있는 자원을 선택함으로, molecular docking 어플리케이션의 성능을 향상 시킨다.

최적 자원 선택 브로커의 구성 요소들의 역할은 다음과 같다.

(1) RSL Parser

최적 자원 선택 브로커에게 사용자의 요구가 기술된 RSL이 전달되면, RSL Parser는 사용자가 요구하는 자원에 관한 정보를 추출하여 자원 검색 에이전트에게 전달한다.



<그림 2> 최적 자원 선택 브로커의 구조

(2) 자원 검색 에이전트

자원 검색 에이전트는 RSL Parser를 통해 얻어진 사용자의 요구 자원에 대한 정보를 이용하여 글로버스에서 자원 정보 검색을 지원하는 MDS를 통해 적절한 자원을 검색하여 자원 선택 에이전트에게 전달한다.

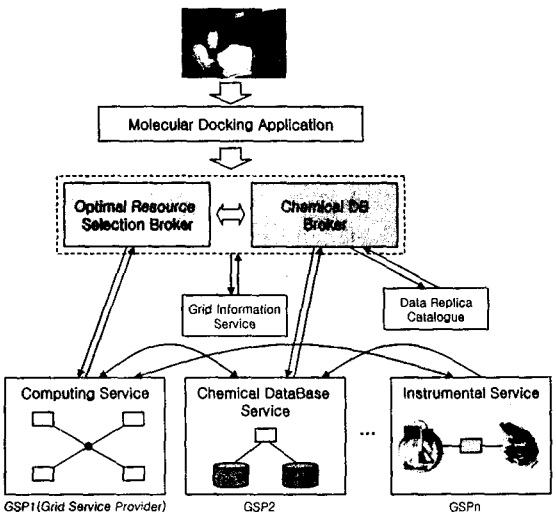
(3) 자원 선택 에이전트

자원 선택 에이전트는 후보 자원 중에서 작업 수행을 위해 사용할 자원을 선택하여 자원 할당 요청 에이전트에게 최적 자원 집합을 전달한다.

(4) 자원 할당 요청 에이전트

자원 할당 요청 에이전트는 자원 선택 에이전트에서 넘어온 최적 자원 집합에 속한 자원들에 대해서 DUROC을 통해 각 자원 관리자에게 자원 할당을 요청한다.

4.3 Molecular Docking 의 연산 모델 설계



<그림 3> Molecular Docking의 연산 모델

<그림 3>은 그리드를 이용한 molecular docking 어플리케이션의 연산 모델을 보여주고 있다. Molecular docking을 수행하는 사용자는 새로운 분자를 찾기 위해 molecular docking 어플리케이션을 통해 필요한 파라메타들을 입력하여 molecular docking 문제를 정형화한다. 그리고 문제 해결을 위해 요구되는 수행시간과 소요 비용을 상세화하여 그리드 자원 브로커에게 필요한 컴퓨팅 자원과 데이터를 요청한다.

본 연구에서 설계한 연산 모델에서는 두 가지의 브로커를 이용한다. 작업 수행을 위한 최적의 자원을 검색해주고 할당을 담당하는 최적 자원 선택 브로커와 화학 데이터베이스에 대한 접근과 그리드상에 분산되어 있는 데이터베이스의 통합을 담당하는 CDB 브로커를 이용하여 molecular docking 어플리케이션이 효율적으로 수행될 수 있는 환경을 제공한다. 최적 자원 선택 브로커와 CDB 브로커는 그리드 정보 서비스를 이용하여 molecular docking을 위해 요청한 작업을 수행하기에 적합한 자원을 찾고, 각 자원의 가용 상태를 확인한 후 요청된 작업을 스케줄링한다. 브로커에 의해 스케줄링되어 각 자원에 할당된 작업은 지역 자원 관리자에 의해 수행된다. 끝으로, 요청된 모든 작업의 수행이 끝나면 molecular docking 어플리케이션은 각 작업들의 연산 결과들을 통합하여 사용자에게 결과값을 반환한다.

5. 결론 및 향후 연구과제

본 논문에서는 그리드를 기반으로 하는 molecular docking 어플리케이션을 설계하였다. Molecular docking 어플리케이션이 효율적으로 수행되기 위해서는 대용량 화학 데이터베이스에 대한 관리와 작업 수행에 필요한 연산자원에 대한 관리가 필요하다. 그러나, 기존 molecular docking 어플리케이션들은 연산 자원이 충분하지 못한 단일 노드에서 수행되도록 설계, 구현되어 결과값을 얻기에 오랜 시간이 소요되거나, 정밀하지 못한 결과값을 얻는다. 따라서, 본 논문에서는 연산 집약적이며 데이터 집약적인 molecular docking 문제를 그리드에서 수행하여 저비용으로 짧은 시간내에 정확한 결과값을 얻을 수 있도록 설계하였다.

향후 연구과제로는 본 논문에서 설계한 내용을 바탕으로 어플리케이션을 구현하고 구현된 molecular docking 어플리케이션을 통해 실제 데이터 값을 이용한 실험과 성능 측정이 요구된다.

참고 문헌

[1] Shoichet B, Bodian D, Kuntz I. Molecular docking using shape descriptors. Journal of Computational Chemistry 1992: 13(3):380-397.

[2] I. Foster, C. Kesselman and S. Tuecke, "The Anatomy of the Grid : Enabling Scalable Virtual Organizations", International Supercomputer Applications, 15(3), 2001.

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

[4] R. Buyya, K. Branson, J. Giddy, D. Abramson, The virtual laboratory: a toolset to enable distibuted molecular modelling for drug design on the World-Wide Grid, Concurrency and Computation, 2003, 15:1-25.

[5] Abramson D, Giddy J, Kotler L. High performance parametric modelling with Nimrod/G: Killer application for the Global Grid, Proceedings International Parallel and Distributed Processing Symposium (IPDPS). IEEE Computer Society Press: Los Alamitos, CA, 2000.

[6] Ewing A (ed.). DOCK Version 4.0 Reference Manual. University of California at San Francisco (UCSF), U.S.A., 1998. http://www.cmpharm.ucsf.edu/kuntz/dock.html.