

그리드 기반 대규모 신약후보물질탐색환경의 성능 개선

이세훈 황순욱

한국과학기술정보연구원

sehooi@kisti.re.kr, hwang@kisti.re.kr

Performance enhancement of Grid-based Large-scale Virtual Screening Environment

Sehoon Lee Soonwook Hwang

Korea Institute of Science and Technology Information

요 약

그리드 기반 대규모 신약후보물질탐색환경에서는 (1) 대규모의 도킹 작업을 그리드 상에 제출한 후 이들을 지속적으로 관리하고, (2) 실행된 각 작업들은 동시에 그리드 스토리지 및 메타데이터 카탈로그 서비스에 접근한다. 우리는 위의 두 부분에서 성능의 개선이 필요함을 발견하였다. (1)의 과정에서 제출된 작업의 상태 정보를 얻기 위해 그리드 스케줄러를 사용하던 기존의 방식은 대규모 작업 제출에 의한 부하 문제로 정확한 정보를 신속하게 얻는 데 어려움이 있어 제출된 작업의 효율성이 떨어지는 문제가 있었다. 그리고 (2)의 과정에서는 동시에 실행된 대규모 작업들의 접근 요청이 그리드 스토리지 및 메타데이터 카탈로그 서비스에 집중되면서 이로 인해 성능 저하가 일어났다. 본 연구에서는 기존 환경에서 찾아낸 위의 두 가지 주요 부분에서 발견한 성능 이슈를 밝히고 또 이에 대한 해결 방법을 제시한다.

1. 서 론

1.1 가상 스크리닝

가상 스크리닝(Virtual Screening)은 컴퓨터의 계산 능력이 비약적으로 향상되면서 단백질 및 화합물의 특성을 컴퓨터에서 정밀하게 시뮬레이션 할 수 있게 되면서 생겨난 방법으로, 주어진 단백질에 대해 화합물이 적합하게 붙을 수 있는 지를 시험(도킹, docking)하고, 이를 통해 최적의 신약 후보 화합물을 탐색하는 컴퓨팅 테크닉이다. 일반적으로 스크리닝은 인비트로(in vitro) 실험으로도 가능하지만 수백만 개에 달하는 모든 후보 화합물에 대해 인비트로 실험을 행하는 것은 비용적인 측면을 생각해 볼 때 현실적이지 못하다. 실제로 가상 스크리닝은 실제 인비트로 실험의 대상을 크게 줄이기 위한 용도로 널리 활용되고 있다. 가상 스크리닝에 사용되는 방법은 여러 가지이나, 주로 단백질 도킹(Protein Docking)과 분자동역학 시뮬레이션(Molecular Dynamics Simulation)이 사용된다.

1.2 그리드 기반 대규모 가상 스크리닝

전 세계적으로 수백만에서 수천만 개의 후보 화합물들이 존재하기 때문에 이들 모두에 대해 가상 스크리닝을 실행하기 위해서는 수많은 컴퓨팅 자원이 필요하다. 국내 제약회사를 비롯한 신약 개발을 위한 연구팀들은 주로 연구실에 있는 클러스터를 이용하여 가상 스크리닝을 수행하여 새로운 신약 물질들을

개발하고 있다. 하지만 연구실 규모의 클러스터만으로는 수백만 개의 화합물을 도킹하는데 최소 수년이 걸리기 때문에 실제로 국내의 신약 개발 연구팀들은 수천에서 수만 가지의 화합물을 선택하고 이에 대해서만 도킹을 수행하고 있다. 대규모의 자원을 제공하는 공공의 그리드 인프라는 이와 같이 필요한 규모의 자원을 확보하지 못한 국내 연구자들에게 큰 도움이 될 수 있다. 연구자들은 그리드 인증서를 발급 받고, 관련 분야의 VO(가상 조직)에 가입함으로써 해당 VO에서 제공하는 계산 자원 및 스토리지를 자유롭게 이용할 수 있다. 유럽의 WISDOM(Wide in Silico Docking on Malaria) 프로젝트는 공공의 인프라를 신약 개발 연구에 활용한 대표적인 예로, 실제 BIOMED VO가 제공하는 EGEE[1] 인프라(80,000 CPU core, 20Pbyte storage)를 이용해 그리드 기반 대규모 가상 스크리닝을 수행함으로써 개발 속도와 비용 절감 등에서 큰 기여를 할 수 있음을 입증했다.

2. 관련연구

2.1 WISDOM Production Environment

WISDOM(Wide in Silico Docking on Malaria)은 그리드 인프라를 활용하여 소외된 질병이나 긴급하게 퍼져 나가는 질병에 대한 신약후보물질을 탐색하는 프로젝트이다 [2]. WISDOM Production Environment(이하 WISDOM 환경)은 대규모 그리드

인프라에서 한정된 시간 내에 수많은 도킹을 처리하기 위해 개발되었다. 2005년 처음 개발된 이후 그리드 서비스와 자원을 효율적으로 활용하기 위해 점진적으로 업그레이드 되어 왔다. 2005년 여름, 처음으로 수행한 대규모 가상 스크리닝에서 말라리아를 대상으로 6주 동안 42백만 개의 도킹을 처리했다. 2006년에는 다시 변종 말라리아를 대상으로 2차 대규모 가상 스크리닝을 수행했고, 2달 반 동안 약 1.4억 개의 도킹을 처리했다 [3]. 이 수치는 시간 당 약 8만개의 도킹이 처리되었음을 의미한다. 2007년에는 그리드 상에서 작업 실행의 효율성을 향상시키기 위해 새로운 WISDOM 환경이 개발되었다. 기존에는 전체 후보 화합물을 일정 수의 작업에 미리 할당하고, 제출된 각 작업은 미리 할당된 후보 화합물에 대해서만 도킹을 수행하는 방식(푸시 모델, Push Model)을 사용하였다. 새로운 환경에서는 풀 모델(Pull Model)을 적용하는데 이는 후보 화합물을 미리 할당 받지 않고 일단 작업이 제출되어 실행된 후에 실시간으로 중앙 큐로부터 실행할 도킹을 동적으로 가져가 실행하는 방식이다. 이와 같은 풀 모델 방식은 실시간에 워크로드(Workload)를 관리할 수 있다는 장점이 있다. 예를 들어, 성능이 좋은 시스템에서 실행되는 작업은 상대적으로 다른 작업보다 정해진 시간에 더 많은 도킹을 할당 받아 처리할 수 있다.

2.2 DIANE

스위스 CERN에서 개발한 DIANE(Distributed ANalysis Environment)은 기존 그리드 미들웨어 위에서 동작하는 상위 수준 미들웨어로, 하위 수준의 컴퓨팅 자원들과 최상위의 응용 프로그램을 연결하는 경량 그리드 프레임워크이다 [4]. DIANE은 분산된 마스터/워커 모델(풀 모델)에 기반하여 응용 프로그램의 워크플로우를 관리한다. 이는 모든 응용에 적용될 수는 없지만 전체 작업이 여러 개의 작은 독립적인 하위 작업으로 나누어 질 수 있는 고에너지물리나 바이오 인포매틱스 등의 분야에 적합하다. 클라이언트가 플래너(Planner)에게 작업 정보를 보내면 플래너는 작업을 여러 개의 하위 작업들로 나누어 워커(Worker)들을 통해 실행시킨다. 실행된 결과는 인테그레이터(Integrator)에 의해 취합되고, 취합된 최종 결과는 클라이언트에게 제공된다. DIANE은 분산 응용 프로그램을 빠르고 쉽게 개발하고 구축할 수 있는 상위 수준의 기능을 제공한다. 응용프로그램 개발자는 통신 방법 등을 구체적으로 코딩할 필요 없이 단지 데이터 메시지의 입력과 출력 내용을 기술하는 애플리케이션 컴포넌트를 기술하고 콜백(callback) 인터페이스만 구현하면 된다. 대신 DIANE 프레임워크(framework)가 워크플로우 관리와 메시지 전송 등에 대한 구체적인 내용을 책임진다.

DIANE은 다양한 응용프로그램들에 의해

테스트되었지만, 가장 대규모의 응용에 활용된 것은 신약후보물질탐색 분야이다. 2006년 대만 ASGC(Academia Sinica Grid Computing)센터를 중심으로 진행된 대규모 가상 스크리닝에서는 조류독감에 대한 30만개의 화합물 중 2,000개의 후보 화합물을 추출하는데 성공했다. 이를 위해 약 4주간 2580개의 작업들을 통해 16.7년의 CPU 시간 동안 도킹을 실행하였다 [5].

3. 기존 환경의 문제점

앞에서 소개한 기존 환경을 활용한 지난 대규모 신약후보물질탐색을 통해 발견된 개선사항에 대해 설명한다.

3.1 병목으로 인한 그리드 스토리지의 성능 저하

WISDOM 환경에서 도킹을 위해 자주 사용되는 데이터에는 도킹 소프트웨어(Autodock), 목표단백질(Protein), 후보화합물(Ligand), 도킹 결과(DLG 파일, Coordinates 파일)가 있다. 기존의 방식에서는 이와 같은 파일들을 그리드 스토리지인 Storage Element(이하 SE)에 저장하고, 그에 대한 메타데이터인 물리적 위치 정보를 AMGA에 저장한다. 하지만 그리드 스토리지는 대규모의 작업 요청이 동시에 집중되는 경우 성능 저하가 눈에 띄게 나타났다. 복잡한 프로토콜을 간소화 하기 위해 gLite에서 채택해 쓰이고 있는 LCG File Catalog(LFC)를 거치지 않고 하위 레벨의 GSIFTP 프로토콜을 이용해 데이터에 바로 접근하더라도 병목현상은 여전히 나타났다. 실제 여러 대규모 가상 스크리닝을 수행하는 동안 그리드 스토리지에 대한 병목 현상은 진행 속도를 떨어뜨리는 주요 원인으로 나타났다.

3.2 배포된 작업의 효율성 저하

대규모 가상 스크리닝을 수행하는 작업 실행 환경의 가장 큰 목표는 "최대한 빠른 시간에 주어진 작업을 모두 완료하는 것"이다. 이를 위해서는 작업 관리자는 정해진 그리드 자원을 효율적으로 활용해야 한다. 따라서 "대기 시간이 짧고 또 오랫동안 작업을 수행할 수 있는 자원을 찾는 것"이 가장 중요하다고 볼 수 있다. 풀 모델에서는 실제로 활용 가능한 CPU의 수를 초과하는 작업을 제출해서 이 숫자를 계속 유지한다면 그리드 자원을 최대한 활용할 수 있지만 작업 관리자가 많은 수의 작업 실행 에이전트를 지속적으로 관리해야 하는 부담이 있고, 또 그리드 스케줄러인 Resource Broker(이하 RB)에도 많은 부담을 주기 때문에 적합하지 않은 방법이다. 따라서 우리는 실제 활용 가능한 CPU의 수 이하의 작업 실행 에이전트를 실행하는 경우에 가장 효율적으로 그리드 자원을 활용하는 방법으로 한정하여 생각하기로 한다. 배포된 작업의 효율성이 저하되는 경우는 다음의

3가지를 생각해 볼 수 있다. 첫째, RB가 제공하는 상태정보를 활용해 제출된 작업을 관리하는 경우이다. 그리드 스케줄러인 Resource Broker(이하 RB)는 제출된 작업을 적절하게 스케줄링하여 등록된 Computing Element(이하 CE) 중 하나를 선택하여 작업을 전달한다. 그리고 제출된 작업에 대한 기본적인 상태정보도 제공한다. 기존 WISDOM 환경은 RB가 제공하는 상태정보를 활용해 작업을 관리한다. 하지만 기본적으로 RB가 제공하는 작업 상태정보는 실제 상태와 약간의 시간 차이가 있고, 게다가 많은 수의 작업이 동시에 제출되면 RB에 집중되는 부하로 인해 성능이 저하된다. 때문에 제출된 작업의 최신 상태정보를 얻는데 어려움이 있어 실패하거나 취소된 작업을 다시 제출하는데 시간이 소요된다. 그 시간 동안은 그리드를 활용하지 못하게 되므로 작업의 효율성이 저하된다. 둘째, 지금 바로 실행가능한 자원이 있음에도 불구하고 대기시간이 긴 자원에 작업이 제출되는 경우이다. 만약 자원별로 현재 평균 대기시간이 어느 정도 되는지 알 수 있다면 보다 효율적으로 그리드 자원을 활용할 수 있다. 하지만 기존 WISDOM 환경은 이러한 점을 고려하지 않고 있다. 셋째, 기본 실행시간이 작업 소요시간보다 짧은 자원에 작업이 제출되는 경우이다. 예를 들어, 30분만 실행할 수 있는 자원에 1시간이 걸리는 작업을 제출하게 되면 대기시간을 포함해 30분이라는 시간을 낭비하게 된다. 기존 환경에서는 CE의 큐 이름을 이용해 이를 구분했지만 (큐 이름에 "short"나 "30min"이 포함되어 있으면 제외하는 방식), 큐 이름이 정확한 실행시간을 반영하지 않는 경우가 많다. 배포된 작업의 효율성을 높여 최대한 빠른 시간에 주어진 작업을 모두 완료하기 위해서는 이와 같은 문제점을 해결하는 것이 반드시 필요하다.

4. 개선사항

문제점을 해결하기 위해 새롭게 적용한 성능개선 사항에 대해 설명한다. 그림 1은 새롭게 개선된 작업 실행 환경을 활용한 대규모 신약후보물질탐색의 진행 과정을 보여준다. User Interface (이하 UI)에 설치된 작업 관리자 (Job Manager)를 통해 전체 작업을 시작한다. 작업 관리자는 먼저 다수의 작업 실행 에이전트(Job Execution Agent)를 제출하고 준비된 모든 도킹이 종료될 때까지 이들의 상태를 주기적으로 모니터링 하면서 관리한다. 만약 제출된 작업 실행 에이전트가 시작되지 못하거나 시작되더라도 도중에 실패하는 경우 이를 감지하여 취소하고 또다시 새로운 작업 실행 에이전트를 제출함으로써 항상 일정한 수의 작업 실행 에이전트를 그리드 상에 유지한다. 일단 작업 실행 에이전트가 그리드 상의 Worker Node (이하 WN)에서 실행되면 먼저 태스크 관리자인 AMGA에게 태스크를 요청, 동적으로 할당 받는다. 가져온 태스크를

참고하여 도킹 소프트웨어, 단백질, 화합물을 WN로 가져온다. 이 때 단백질을 제외한 소프트웨어, 화합물은 모두 AMGA에서 가져오고, 단백질은 Storage Element(이하 SE)에서 가져온다. 이후 도킹을 실행하고 정상적으로 끝나게 되면 그 결과를 AMGA에 저장하고 다시 새로운 태스크를 요청하는 식으로 WN에서 실행되는 동안 이 과정을 계속 반복한다.

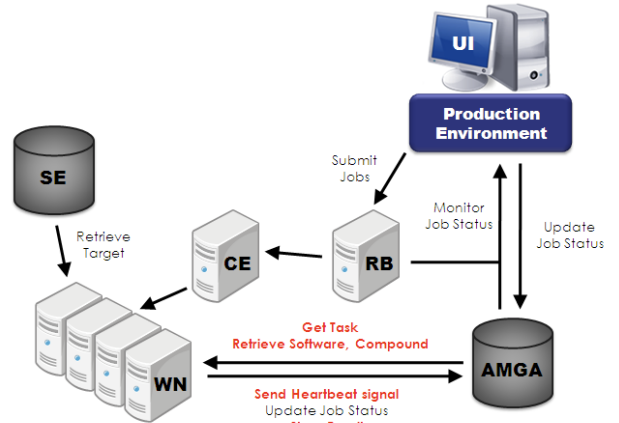


그림 1 개선된 작업 실행 환경의 동작 원리

기존의 방식과 비교하여 크게 달라진 부분은 첫째, 작업 관리자가 RB와 같은 그리드 스케줄러를 활용하지 않고 작업 실행 에이전트가 제공하는 정보를 활용해 직접 스케줄링을 한다는 점이다. 둘째로 메타데이터를 저장소 뿐만 아니라 실제 데이터 저장소로서 AMGA를 활용한다. 기존의 AMGA에 메타데이터가 아닌 데이터를 직접 저장함으로써 성능에 큰 향상을 가져왔다.

4.1 작업 배포 효율 향상을 위한 기능 개선

제출한 작업의 효율성을 높이기 위해서는 작업을 제출할 때 현재 가장 작업을 실행하기 적합한 CE를 찾는 것이 가장 중요하다. 이를 위해서는 CE에 대한 보다 더 자세한 정보가 필요하다. 이와 같은 정보를 얻기 위해 우리는 워커노드(Worker Node)에서 실행되는 작업 실행 에이전트로부터 정보를 얻어 AMGA에 저장하고 이를 활용한다. 먼저, 작업 실행 에이전트는 주기적으로 타임 스탬프(Time Stamp)를 AMGA에 저장하고, 작업 관리자는 이를 참조하여 제출한 작업이 WN에서 시작됐는지, 또 아직 살아있는지에 대한 실시간 정보를 얻을 수 있다. 만약 제출한 작업이 일정 시간 후에도 시작되지 않는다면 이를 감지하고 취소 후 다시 제출한다. 또 시작됐던 작업이라도 주기적인 신호가 감지되지 않는다면 동작이 멈추었다고 판단하고 역시 취소 후 다시 제출한다.

4.2 데이터 저장소로서의 AMGA의 활용

도킹 소프트웨어, 단백질, 후보물질, 도킹 결과 등 모든 중요 데이터를 그리드 스토리지로부터 읽고 또

저장하는 기존의 방식은 대규모 작업 실행에 따른 병목 현상으로 인해 성능 저하의 주요 원인이었다. 우리는 이를 해결하기 위해 AMGA를 데이터 저장소로 활용하는 방식을 제안한다. AMGA는 데이터베이스를 기반으로 하고 있기 때문에 SE에 비해 많은 양의 데이터 요청을 처리하는 데 있어 좋은 성능을 보여준다. 이는 전체 가상 스크리닝의 동시 처리량을 늘려주기 때문에 SE에 나타나던 병목 현상을 해결할 수 있다. AMGA에서 제공하는 Text 데이터 타입을 활용하면 최대 2GByte까지의 데이터를 저장할 수 있다. 하지만 ODBC를 통한 데이터 통신 버퍼의 문제로 현재는 1.6MByte까지의 데이터를 저장이 가능하다. 따라서 크기가 큰 목표 단백질은 기존의 방식대로 SE에 저장하고, 이를 제외한 도킹 소프트웨어, 후보물질, 도킹 결과는 모두 AMGA에 저장하는 방식을 적용했다.

그림 2는 작업 실행 에이전트 수의 증가에 따른 AMGA와 SE의 성능을 비교해 보여준다. SE는 도킹 결과를 저장하는 경우와 저장된 데이터를 읽어오는 경우 모두 작업 실행 에이전트 수가 증가함에 따라 소요시간이 엄청나게 증가함을 알 수 있다. 4000개의 에이전트가 동시에 접근하는 경우 65KByte 정도의 도킹 결과를 저장하는데 SE는 약 4343초가 걸리는 반면 AMGA는 약 8초 밖에 걸리지 않았다. 읽기 연산에서는 더 많은 차이를 보였다. 역시 4000개의 에이전트가 동시에 접근하는 경우 7Kbyte 정도의 후보물질을 읽는데 약 1107초가 걸리는 반면 AMGA는 약 5초 밖에 걸리지 않았다. 이와 같은 성능의 차이는 전체 도킹의 진행속도에도 영향을 미친다는 것을 알 수 있다.

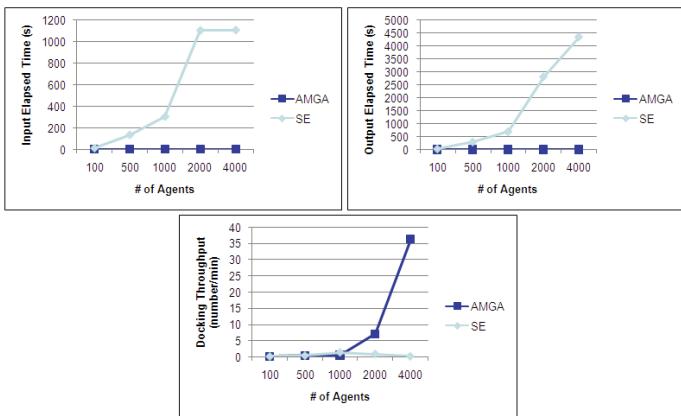


그림 2 AMGA와 SE의 성능 비교

SE를 사용하는 경우에는 최대 분당 1.5개의 도킹 밖에 수행하지 못하는 반면 AMGA는 분당 36개의 도킹을 수행하고 있고 여전히 확장 가능성을 보여주고 있다. 이는 약 24배 정도의 성능 차이를 보여주고 있다. 실험 결과에서 알 수 있듯이 데이터 저장소로서의 AMGA의 활용은 대규모 가상 스크리닝의 성능에 큰 향상을 가져왔다. 최대 저장용량을 늘려서 단백질까지도 AMGA에 저장할 수 있다면 더 나은 성능을 얻을 수

있으리라 본다. 아주 큰 용량의 데이터를 저장하는 경우를 제외한 작은 양의 데이터를 빈번하게 읽고 쓰는 경우에는 SE 대신 AMGA를 활용하는 것이 더 바람직하다고 판단할 수 있다.

5. 개선된 환경을 통한 대규모 가상 스크리닝의 결과

개선된 환경의 성능을 검증하기 위해 제2형 당뇨병 목표 단백질에 대한 대규모 신약후보물질탐색을 수행했다. 현재 신약후보물질탐색은 프랑스의 WISDOM, 한국의 KISTI, 전남대의 국제 협력으로 진행되고 있다. WISDOM과 전남대로부터 주요 목표 단백질과 후보 화합물들을 제공 받아, KISTI의 작업 실행 환경을 통해 이에 대한 신약후보물질탐색을 수행한다. 결과로 얻어진 유효 화합물 리스트는 다시 WISDOM과 전남대에 전달되고, 인비트로 시험을 통해 유효한 신약 후보 화합물을 실제로 도출하고, 또 이를 통해 가상 스크리닝 방식의 유효성을 시험한다. 2008년 수행된 제2형 당뇨병 목표 단백질에 대한 대규모 신약후보물질탐색을 통해 본 논문에서 새로 개선한 작업 실행 환경의 성능 향상을 확인할 수 있었다. 표 1을 보면 총 308,310개의 후보물질에 대해 최대 약 7400 CPU를 동시에 활용하여 2.4일 만에 완료했다. 이는 기존 WISDOM 환경과 비교했을 때 동시 처리량(최대 동시 활용 CPU)은 1.5배, 작업 효율성(그리드 활용률)은 2배로 향상되었음을 확인할 수 있다. 이는 "최대한 빠른 시간에 주어진 작업을 모두 완료"하고자 하는 그리드 기반 대규모 가상 스크리닝의 목표를 충분히 만족시킨 결과로 볼 수 있다.

	WISDOM-II	DIANE	KISTI
Total number of dockings	156,407,400	308,585	308,310
Estimated duration on 1 CPU	413 years	16.7 years	39.0 years
Duration of the experiment	76 days	30 days	2.4 days
Cumulative number of Grid jobs	77,504	2,580	103,583
Maximum number of concurrent CPUs	5000	240	7,370
Number of used Computing Elements	98	36	127
Crunching Factor	1983	203	5937
Distribution Efficiency	39%	84%	81%

표 1 개선된 환경을 통한 대규모 가상 스크리닝의 결과

6. 결론

본 연구에서는 신약후보물질탐색 분야에 활용되고 있는 그리드 기반 대규모 신약후보물질탐색환경의 성능 저하 이슈를 제기하고 이에 대한 해결 방법을 제시했다. 개선된 환경을 활용해 대규모 신약후보물질탐색을 수행한 결과, 기존 WISDOM 환경의 성능을 평균 활용 CPU 수(Crunching factor)에서 3배, 동시 처리량(최대 동시 활용 CPU 수)에서 1.5배, 작업 분배 효율 면에서 2배 이상의 성능 개선을 얻을 수 있음을 확인했다.

개선된 환경을 통해 제공되는 서비스는 국내외 신약 물질 연구자들에게 그리드 인프라를 이용하여 “최대한 빠른 시간에 주어진 작업을 모두 완료”하여 새로운 신약 물질을 탐색할 수 있는 환경을 제공할 수 있을 것이다.

참고문헌

- [1] EGEE: Enabling grids for e-science in Europe, <http://public.eu-egee.org/>
- [2] WISDOM: Wide in-silico docking on malaria, <http://wisdom.eu-egee.fr/>
- [3] Vincent Breton et al, “Grid added value to address malaria”, in Proc. 6th IEEE/ACM CCCGrid Conf., Vol. 2, P. 40, 2006
- [4] DIANE: Distributed analysis environment, <http://it-proj-diane.web.cern.ch/it-proj-diane/>
- [5] Hurng-Chun Lee et al, “Grid-Enabled High-Throughput In Silico Screening Against Influenza A Neuraminidase”, IEEE Transactions on Nanobioscience, Vol. 5, No. 4, 2006