그리드 컴퓨팅을 이용한 BLAST 성능개선 및 유전체 서열분석 시스템 구현

Performance Improvement of BLAST using Grid Computing and Implementation of Genome Sequence Analysis System

김동욱, 최한석
목포대학교 멀티미디어공학전공

Dong-Wook Kim(dwkim@kribb.re.kr), Han Suk Choi(chs@mokpo.ac.kr)

요약

본 논문에서는 현재 생물정보학 연구에서 가장 많이 사용하고 있는 BLAST의 문제점을 분석하고 이에 따른 해결책을 제시하기 위하여 그리드 컴퓨팅을 이용한 G-BLAST(Grid Computing을 이용한 Basic Local Alignment Search Tool)를 제안한다. 본 연구에서 제안하고 있는 G-BLAST을 이용한 시스템은 이기종 분산 환경에서 수행이 가능한 서열분석 통합 소프트웨어 패키지이며 기존 서열분석 서비스의 취약점인 검색 성능을 개선하여 BLAST 검색 기능을 강화 하였다. 또한, BLAST 결과를 사용자가 관리 및 분석이 용이하도록 데이터베이스 및 유전체 서열분석 서비스 시스템을 구현하였다.

본 논문에서는 G-BLAST시스템의 성능확인을 위하여 병렬컴퓨팅 성능테스트 기법을 도입하여 구현된 시스템을 기존 BLAST와 속도 및 효율부분에서 비교하여 성능개선을 확인하였으며 서열결과 분석에 필요한 자료를 사용자관점에서 제공해주고 있다.

■ 중심어 : | 바이오인포매틱스 | 블라스트 | 그리드컴퓨팅 | EST분석 | 데이터베이스 | 파서 |

Abstract

This paper proposes a G-BLAST(BLAST using Grid Computing) system, an integrated software package for BLAST searches operated in heterogeneous distributed environment. G-BLAST employed 'database splicing' method to improve the performance of BLAST searches using exists computing resources. G-BLAST is a basic local alignment search tool of DNA Sequence using grid computing in heterogeneous distributed environment.

The G-BLAST improved the existing BLAST search performance in gene sequence analysis. Also G-BLAST implemented the pipeline and data management method for users to easily manage and analyze the BLAST search results. The proposed G-BLAST system has been confirmed the speed and efficiency of BLAST search performance in heterogeneous distributed computing.

■ keyword : | Bioinformatics | BLAST | Grid-computing | EST Analysis | Database | Parser |

I. 서 론

1990년대 인간 유전체 프로젝트(Human Genome Project)이후 염기나 단백질의 서열을 자동으로 분석할 수 있는 각종기기(DNA Sequencer, DNA Microarray 등)들의 획기적인 발전으로 인해 생물학적 정보의 양은

접수번호 : #100407-003
접수일자 : 2010년 04월 07일
심사완료일 : 2010년 04월 26일
교신저자 : 최한석, e-mail : chs@mokpo.ac.kr

[그림 1]과 같이 기하급수적으로 증가하고 있다.

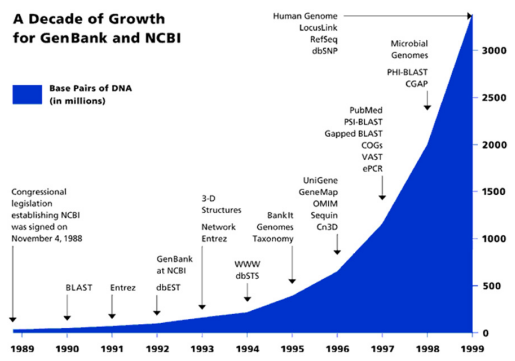


그림 1. 1990년대 Genebank와 NCBI 생물정보 증가 추세[2]

이와 같이 급증하고 있는 생물정보를 처리하기 위하여 생물정보학이 활발히 연구되고 있고 특히, 생물정보학(Bioinformatics) 연구에서 가장 많이 사용되고 있는 방법은 BLAST(Basic Local Alignment Search Tool)이다. 잘 모르고 있는 핵산(DNA 또는 RNA) 서열이나 단백질 서열이 주어지면, 먼저 분석하고자 하는 서열과 관계되는 서열들을 서열 데이터베이스(sequence database)로부터 찾아낸다. 그 다음에 데이터베이스에서 찾아진 서열들로부터 분석하고자 하는 서열의 성질들을 유추해낸다[1]. 이러한 서열 데이터베이스 검색에서 가장 대중적으로 사용되고 있는 방법이 BLAST 이며 BLAST는 대부분의 생물정보학 온라인 검색서버의 중심을 이루는 알고리즘이다[2]. BLAST는 전체서열간의 최적 글로벌 정렬을 수행하는 것이 아니라 지역적 유사성(Local Similarity)이 있는 부분을 찾아 서열의 짝짓기 비교를 수행한다[1].

BLAST는 몇 분 내에 수백에서 심지어는 수천회의 서열비교를 실행할 수 있다. 그리고 유사한 서열을 찾기 위하여 수 시간 안에 전체 데이터베이스에 대해 질의 서열을 비교할 수 있다. BLAST는 워낙 대중적이어서 생물정보학 분야에서는 “이 서열을 Genbank에 대해서 BLAST 하였으며 세 개의 매치를 찾았다” 고 말하는 것에서 볼 수 있듯이 생물정보학에서는 일반 동사화될 정도로 중요한 시스템이 되었다[2].

BLAST 를 사용하는 방법에는 두 가지가 있다[3]. 첫 번째는 미국 NCBI 사이트(http://www.ncbi.nlm.nih.gov/blast/)에서 웹을 통해서 BLAST 검색을 하는 것이다. 하지만 분석하고자 하는 서열들의 수가 아주 많을 때는 BLAST를 개인(또는 연구실) 컴퓨터에 설치하여 검색하는 것이 훨씬 효율적이다. 그러나 BLAST 검색에서 찾고자하는 서열의 수가 많아지면 검색시간이 기하급수적으로 증가한다. 일반 PC에서 네트워크를 통해 2,000여개의 EST서열의 BLAST 검색에 필요한 시간은 48시간이상이며 이마저도 네트워크상태가 안정적일 때 가능하다. 이런 이유로 많은 생물학자들이 연구절차상의 다른 작업을 수행하기 앞서 많은 시간을 대기해야 하는 문제점이 발생한다. 이러한 문제를 해결하기위해서는 고속 및 대용량의 하드웨어를 설치하여 빠른 결과를 탐색하는 것이 바람직하나 일반 유전체 연구자들의 실정으로는 가능하지 않은 것이 현실이다.

따라서 본 논문에서는 이러한 문제를 최신 IT기술인 그리드방식과 자바 프로그래밍 기법을 이용하여 최소한의 비용으로 기 보유한 저가의 시스템들을 재활용하여 최대한의 효과를 얻기 위하여 BLAST 검색시스템을 설계하였으며 검색결과를 연구자가 쉽게 사용할 수 있도록 Excel 형식으로 변경할 수 있는 파서 및 검색결과가 자동으로 데이터베이스에 저장되어 논문 작성에 필요한 각종 분석자료를 서비스하도록 설계하였다.

본 논문에서는 위와 같이 유전체 서열분석시스템의 효율적인 구조를 제안하고, 구축된 시스템의 성능평가를 통해 향후 개선방향을 살펴본다.

II. 관련연구

1. BLAST의 병렬화

1장에서 살펴본 바와 같이 유전자 서열의 급속한 증가로 인해 BLAST 검색시간의 단축이 생물정보연구에서 매우 중요한 요소로 자리잡게 되었다. 이러한 문제를 해결하기 위한 방법 중 하나가 BLAST의 병렬화이다. BLAST의 병렬화에는 세가지의 방식[4]이 있는데

세립형(fine grained)병렬화는 다중정렬비교를 통해 상호 독립적으로 이루어지기 때문에 효과적이나 Decyper 같은 고가의 특정 H/W를 사용해야 하는 문제점이 있다. 조립형(coarse grained) 병렬화는 BLAST 엔진 소스의 수정없이 각 계산 노드에 데이터베이스를 모두 저장해 두고, 사용자 질의를 분할하여 배치처리(batch processing)하는 것이다. 이 방식은 단일 사용자 질의의 검색시간은 동일하나 웹서비스와 같이 동시다발적인 다수의 사용자 질의 검색서비스에 유용하다. 그러나 전체 서열데이터베이스가 각 노드에 적재되기 위해서는 많은 양의 디스크공간이 필요하고, 응답시간의 향상이 없고 서열 데이터베이스의 일관성을 관리해야하는 단점이 있다. 중립형(medium grained) 병렬화는 노드별로 서열데이터베이스를 분할하는 것이다. 서버노드에서 계산 노드로 사용자질의를 전달하고 검색완료 후 결과를 취합한다. 데이터베이스 분할을 통해 노드의 메모리 사용을 감소시킬 수 있고 빠른 응답시간을 얻을 수 있으나 구현하기가 까다롭다. 본 논문에서는 저렴한 비용으로 고효율을 얻기 위하여 중립형 병렬화를 이용하여 시스템을 설계하였다.

2. 그리드방식을 이용한 BLAST

최근들어 그리드 방식 및 클러스터 방식을 이용한 BLAST 성능개선이 다양하게 시도되고 있다[5-8]. 이러한 시스템들은 기본적으로 앞에서 언급한 BLAST 병렬화를 통한 BLAST의 성능개선을 위하여 시도되고 있고 모두 유사한 방식을 적용하여 개발되었다. 각각의 시스템은 구현환경과 실험환경이 달라 객관적인 비교를 할 수 없으나 LOCAL BLAST보다 월등한 성능을 자랑하는 것이 사실이다.

본 논문에서 제안하는 G-BLAST도 같은 방식의 시스템이다. 그러나 제안하는 시스템은 Grid Computing 방식의 병렬화 뿐만 아니라 BLAST 결과를 자동 분석 및 조작할 수 있는 시스템을 추가하여 사용자 편의성을 증대하였다.

Ⅲ. 시스템 설계 및 구현

1. 시스템 개요

유전체 서열분석 시스템의 주요 기능은 [그림 2]와 같이 설계되었다.

G-BLAST 서버측 데몬과 클라이언트측 데몬은 RMI 통신을 통하여 상태정보와 쿼리 및 BLAST 결과 정보를 송수신한다. 이 과정을 통하여 나온 서열검색 결과를 데이터베이스에 저장하고 분석결과를 사용자에게 서비스한다.

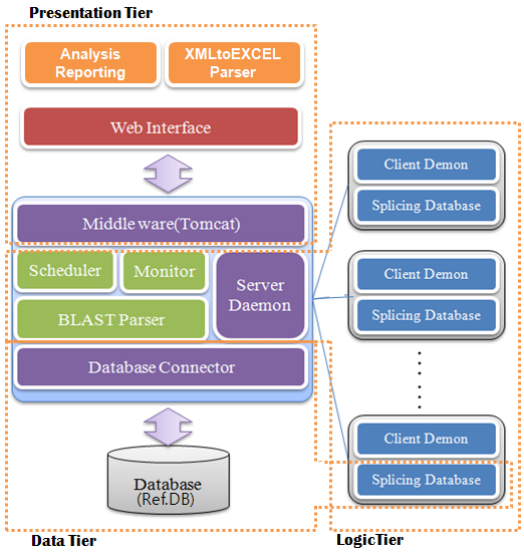


그림 2. 유전체 서열분석 시스템 기능설계

2. G-BLAST 동작 알고리즘

본 논문에서 구현한 G-BLAST 시스템은 3-Tier 방식으로 구성되었고 Presentation Tier, Logic Tier, Data Tier 로 나누어져 있다. 각각의 동작 절차는 다음과 같다.

2.1 Presentation Tier

사용자 인터페이스를 제공하고 있으며 서버의 동작과 클라이언트의 동작을 감시하며 사용자로부터 입력을 받아들여 Logic Tier에 전달하고 Logic Tier에서 전달한 결과 값을 제공 할 수 있다.

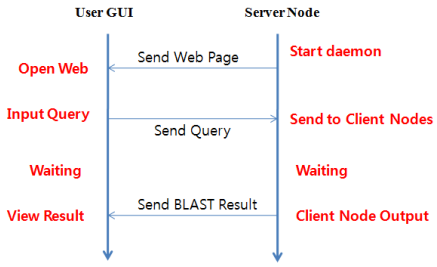


그림 3. 사용자와 서버 노드간 동작과정

2.2 Logic Tier

클라이언트 노드와 RMI 통신을 통해 클라이언트의 상태를 통보 받고 Load Balancing과 Job Scheduling 기능을 수행하며 클라이언트 노드에 쿼리를 전송한 후 클라이언트 노드로부터 수행된 BLAST 검색결과를 전송받아 정렬한 후 사용자에게 서비스한다.

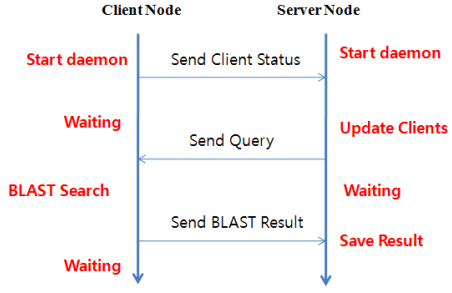


그림 4. Client Node – Server Node 동작과정

2.3 Data Tier

G-BLAST 시스템의 클라이언트 노드측 분할데이터베이스[9][10]에는 [그림 5]와 같은 알고리즘을 이용하여 NCBI(National Center for Biotechnology Information)에서 다운받은 핵산(nucleotide)서열 nr, nt 데이터(6.8Gb, 4.9Gb)를 노드수로 분할하여 formatDB[3] 프로그램을 활용 각각 저장하였고 서버측 데이터베이스에는 클라이언트 노드에서 전송한 BLAST 결과를 mysql[11]에 저장하는 Ref.DB를 설치하였다. Ref DB에는 BLAST 검색에서 나온 결과를 각 Sequence 별로 상위 50위까지 자동 저장되게 구성하였다. 또한, Client Node의 IP와 클라이언트 상태 정보를 저장할 수 있도록 구성하였다.

```
Input : NCBI nr/nt Database, DB connection Properities
Output : 분할된 DB (nr/nt DB)
START {
Step1. DB connection module 호출
Step2. DB connection Properities을 로드
Step3. DB 분할 정보를 입력
Step4. DB 크기 로드
Step4.1 DB 크기를 입력받은 분할정보로 계산
Step4.2 계산된 라인까지 버퍼에 로드
(분할된 라인값 변수로 저장)
Step4.3 LOOP { 분할정보
Step4.3.1 첫줄 '>'의 유무 계산
Step4.3.1.1 IF 1'st char = '<' Then 계산된 라인 -1
Step4.3.1.2 IF 1'st char = '<' Then 계산된 라인
Step4.3.2 DB name+분할정보로 저장
Step4.4 } END LOOP
Step5. 클라이언트 노드에 DB로드
Step5.1 Loop { 분할정보
Step5.1.1 NCBI에서 제공하는 DB 로드 Tool formatDB 실행
Step5.2 } END LOOP
Step6. Main Server DB에 분할정보 입력
} END
```

그림 5. 데이터베이스 분할 알고리즘

3. XMLtoEXCEL 파서

BLAST 결과로 나오는 NCBI format의 XML문서는 매우 유용한 문서이나 일반생물학자가 이를 원하는 형태로 변경하여 원하는 값을 얻어내기에는 쉽지 않은 것이 사실이다[12]. 본 논문에서 구현하고 있는 XMLtoExcel 파서는 이러한 문제점을 해결하기 위해 BLAST 결과 XML 문서를 입력하고 옵션 값을 선택하여 사용자가 원하는 결과만을 Excel 문서로 다운로드 할 수 있도록 파서를 구성하였다. 본 논문에서 구현된 XMLtoEXCEL 파서는 NCBI BLAST DTD를 사용하여 파싱하였고 특히, 유효한 XML 포맷 뿐만 아니라 유효하지 않은 XML 포맷도 결과를 받아볼 수 있도록 하였다. BLAST 결과 파일이 대용량(50G 이상)일 경우나 타 프로그램의 XML 결과가 유효한 문서로 파싱되지 않는 경우가 있어 유효하지 않은 XML문서도 파싱하도

록 [그림 6]과 같이 구현하였다.

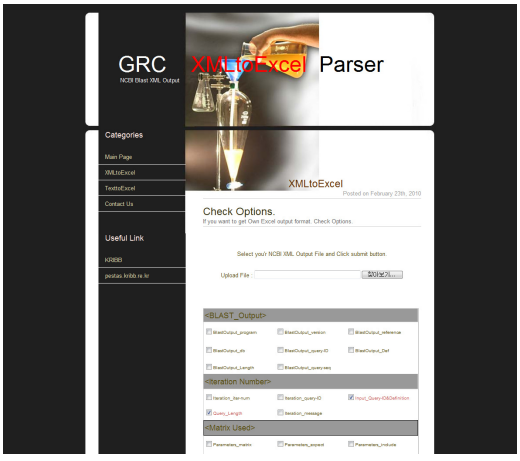


그림 6. XMLtoEXCEL 파서

4. 분석결과 레포팅

[그림 7]은 G-BLAST에 의해 분석된 유전체 서열정렬 자료 데이터베이스를 이용하여 유전체 연구자들이 한눈에 결과를 알아볼 수 있도록 결과 자동 분석 인터페이스를 구현하였고 논문작성시 필요한 분석결과를 Excel 파일로 제공할 수 있도록 구현하였다.

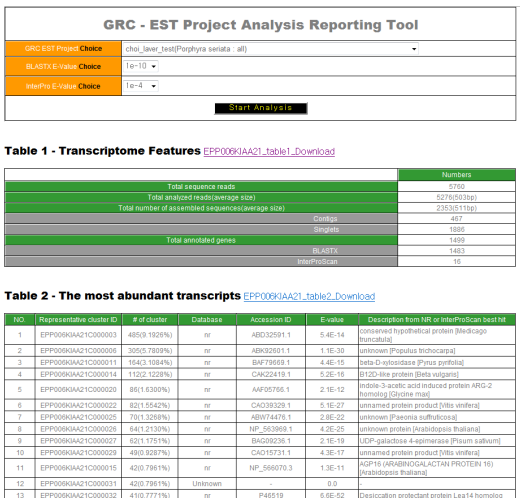


그림 7. 분석결과 Reporting

Ⅳ. 평가

1. G-BLAST 성능평가

본 논문에서 구현한 G-BLAST(20node)의 성능개선 정도를 테스트하기위해 G-BLAST와 기존 NCBI BLAST를 Local에 설치하였으며 한국생명공학연구원에서 시퀀싱한 평균 511bp인 감초 EST 2,000개를 이용 BLASTX를 20개로 분할된 nr DB(각 340Mb)를 포함하는 G-BLAST와 Local BLAST nr DB (6.5Gb)에 각각 질의한 후 성능비교를 시행하였다.

1.1 개발 H/W 환경

본 논문에서 사용한 개발환경은 다음의 [표 1]과 같다. 또한, 성능비교를 위한 Local BLAST의 설치환경은 Intel Pentium IV 1.86Ghz (single core) 1대를 사용하였다.

표 1. H/W사양

구분	사양	비고
CPU	Intel PentiumIV, 1.86Ghz (double core)	10대 20node
O/S	Linux version 2.6.12-1.1381_FC3smp	
Network	100Mbps	
Memory	1024 KB/Node	

1.2 테스트 결과

Local BLAST에서 1,500개와 2,000개 질의는 시스템의 성능부족으로 인하여 시행하지 못하였다.

[표 3]의 결과를 이용하여 병렬 컴퓨팅 성능평가 척도를 이용하여 수식 (1), (2)의 속도향상률과 효율성을 평가하였다[10].

$$속도향상률 = \frac{Local\ BLAST\ 속도}{G-BLAST\ 속도} \quad (1)$$

$$효율성 = \frac{Local\ BLAST\ 검색속도}{G-BLAST\ 검색속도 \times Node\ Count} \quad (2)$$

표 2. BLAST 성능 비교테스트 결과

입력 EST갯수	Local BLAST	G-BLAST
10 EST	3.5min	0.3min
20 EST	35min	1.2min
50 EST	120min	2.5min
100 EST	250min	6min
500 EST	1,250min	18min
1,000 EST	2,800min	32min
1,500 EST	-	70min
2,000 EST	-	75min

표 3. G-BLAST 성능테스트 결과

입력 EST갯수	속도향상률	효율성
10 EST	11.67	0.58
20 EST	29.17	1.46
50 EST	100	2.4
100 EST	41.7	2.08
500 EST	69.44	3.47
1,000 EST	87.50	4.38

BLAST 검색은 전체서열간의 최적 글로벌 정렬을 수행하는 것이 아니라 지역적 유사성(Local similarity)이 있는 부분을 찾아 서열의 짝짓기 비교를 수행한다. [표 2]의 결과에서 보이듯이 입력서열이 증가할수록 계산의 복잡도는 증가하여 서열정렬 속도향상에 컴퓨터의 성능이 밀접한 연관이 있음을 보여준다. [표 3]에서는 1,000 EST(평균 511bp × 1,000 EST= 0.51메가)의 입력 값 일때 최대의 효율성을 보여주고 있다. 이 것은 입력값이 커질수록 효율성이 증가하여 BLAST 성능개선이 이루어졌다는 것을 나타낸다.

V. 결론 및 향후 연구

본 논문에서는 자바와 그리드 컴퓨팅을 이용하여 최소한의 비용으로 BLAST 검색 속도를 개선시키고 이를 이용한 유전체 서열분석 시스템을 제안했다.

BLAST 검색서비스의 속도개선은 비용 투자를 통해 얻을 수 있다. 그러나 많은 유전체 연구자들은 속도개선을 위해 비용투자는 생각하고 있지 않은 것이 현실이다[13]. 이러한 문제점을 해결하기 위해 G-BLAST는 이미 확보하고 있는 저가의 개인용 PC 또는 리눅스, 유닉스 서버를 모두 묶어 BLAST 검색에 활용 할 수 있도록 운영체제에 독립적인 자바프로그래밍을 사용하였고 설치 및 통신이 용이한 RMI(Remote Method Invocation) 방식을 이용하였다. 또한, 무료 DBMS인 Mysql을 이용하여 데이터베이스를 구성하여 비용을 최소화 하였다.

G-BLAST의 개선된 성능은 [표 2]와 [표 3]의 결과를 통해 기존 LOCAL BLAST보다 성능과 효율측면에서 우수하다는 것을 검증하였으며 이를 이용하여 사용자관점의 분석 인터페이스를 구현하였다.

본 논문에서 구축된 유전체분석 시스템을 이용하면 생명정보학에 지식이 없는 일반 유전체 연구자들도 최소한의 비용으로 고성능 BLAST 서비스를 제공 받을 수 있게 되며 서열 분석에 필요한 자료들을 구현된 시스템으로부터 제공받을 수 있어 유전체 연구에 많은 도움이 될 수 있을 것이다.

향후 연구에서는 최신의 분석기법을 적용하여 더 많은 분석자료를 제공할 수 있도록 진행 할 예정이다.

참 고 문 헌

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic Local Alignment Search Tool", Journal of Molecular Biology, Vol.215, No.3, pp.403-410, 1990.

[2] Cincia gibus and Per Jambeck, *Developing bioinformatics computer skills*, O'Reilly Media, 2002.

[3] http://www.ncbi.nlm.nih.gov

[4] Anne Julich, "Implementations BLAST for Parallel Computers," CABIOS, Vol.11, No.14, pp.3-6, 1995.

[5] A.E.Darling, Lucas Carey and Wu-chun Feng, "The Design, Implementation and Evaluation of mpiBLAST," ClusterWorld 2003 conference,

2003.

[6] Vincent Breton, Eddy Caron, Frederic Desprez and Gael Le Mahec, "BLAST Application with Data-Aware Desktop Grid Middleware," 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009.

[7] Enis Afgan and Purushotham Bangalore, "Dynamic BLAST - a Grid Enabled BLAST," IJCSNS International Journal of Computer Science and Network Security, Vol.9, No.4, 2009.

[8] David R. Mathog, "Parallel BLAST on split databases," BIOINFORMATICS APPLICATIONS NOTE, Vol.19, No.14, pp.1865-1866, 2003.

[9] A.E.Darling, Lucas Carey and Wu-chun Feng, "The Design, Implementation and Evaluation of mpiBLAST," ClusterWorld 2003 conference, 2003.

[10] 김태경, 조완섭, "고성능 BLAST 구현을 위한 E-Cluster 기반 데이터 분할 및 질의 라우팅 기법", 한국컴퓨터정보학회논문지, 제14권, 제2호, pp.139-147, 2009.

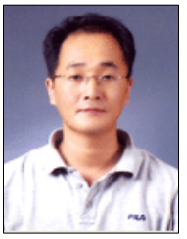
[11] http://www.mysql.org

[12] T. H. Lee, Y. K. Kim, and Baek Hie Nahm, "GBParsy: A GenBank flatfile parser library with high speed," BMC Bioinformatics, Vol.9, No.321, pp.1-6, 2008.

[13] 공재근, 좌용권, 박정선, 유선주, 이문상, "효율적인 생물정보 서열검색을 위한 PC-클러스터 시스템의 구현", 한국정보과학회 제31회 춘계학술발표회, 2004.

저 자 소 개

김 동 욱(Dong-Wook Kim) 정회원



- 2000년 2월 : 목포대학교 전산통계학과 졸업(이학석사)
- 2009년 2월 : 목포대학교 멀티미디어공학과 박사수료
- 2009년 9월 ~ 현재 : 한국생명공학연구원 유전체자원센터 연구원

<관심분야> : Bioinformatics, Database

최 한 석(Han Seok Choi) 정회원

- 1986년 8월 : (미)웨스턴일리노이대 전산과 전산학(이학석사)
- 1987년 5월 : (미)웨스턴일리노이대 수학과 수학(이학석사)
- 1997년 2월 : 전북대학교 전산과 전산학(이학박사)
- 2000년 4월 ~ 현재 : 목포대학교 멀티미디어학과 교수

<관심분야> : Bioinformatics, IT, Web2.0, IPTV