The Korean Journal of Microbiology, Vol. 39, No. 2, June 2003, p. 67-75

녹섹(NOGSEC): A NOnparametric method for Genome SEquence Clustering

이영복[1,2,*] · 김판규[1,3] · 조환규[1,3]

[1]부산대학교 전자계산학과 그래픽스응용연구실, [2]포항공과대학교 생물학전문연구정보센터,
[3]부산대학교 컴퓨터 및 정보통신연구소

비교유전체학의 주요 주제 중 유전자서열을 분류하고 단백질기능을 예측하는 연구가 있으며, 이를 위해 단백질 구조, 공통 서열 및 바인딩 위치 예측 등의 방법과 함께, 전유전체 서열에서 구해지는 유사도 그래프를 분석해 상동유전자를 검색하는 계산학적인 접근방법이 있다. 유사도 그래프를 사용한 방법은 서열에 대한 기존 지식에 의존하지 않는 장점이 있지만 유사도 하한값과 같은 주관적인 임계값이 필요한 단점이 있다. 본 논문에서는 반복적으로 그래프를 분해하는 이전의 방법을 일반화시켜, 유사도 그래프에 기반한 유전자 서열 군집분석 방법론과 객관적이고 안정적인 파라미터 임계값 계산 방법을 제안한다. 제시된 방법으로 알려진 미생물 유전체 서열을 분석하여 이전의 방법인 BAG 알고리즘 결과와 비교했다.

Key words □ **function prediction, genome sequence clustering, graph decomposition**

유전체 서열 분석 작업은 많은 실험과 비용을 필요로 한다. 때문에 실험 대상을 줄이고 새로운 정보를 추출해 내는 목적으로 계산생물학, 생물정보학이 발생하였다. 이중 비교유전체학의 주요 목적은 서로 다른 여러 유전체에서 같은 기능을 하는 보존 유전자 집합을 발견하는 것이다. 유전체들을 교차 비교해 단백질 기능면에서의 유사성을 보이는 유전자 집합을 예측할 수 있으며, 컴퓨터 실험만으로 유전자나 단백질에 대한 정보를 추출하거나 알려진 유전체와 비교해 유전자의 기능을 추정할 수 있다.

기존의 유전체 기능 주석달기(functional annotation) 작업이 문제자체에서 또는 분석툴과 연구방법에서 발생하는 오류요소들을 포함하고 있음은 Devos 등의 논문에서 언급된 바 있다(6). 입력 데이터에 오류를 포함하고 있지만 오류여부를 판단할 수 없으며 처리해야 할 데이터의 양이 너무 많은 문제를 계산학적으로 해결하는 것이 바람직하다. 본 논문에서는 계산학적인 방법으로 서열 유사성 그래프를 분석해 유전체의 기능 주석을 예측하는 이전의 연구와 그에 대한 신뢰도 개선 방법을 다루고 있으며, 유전체 서열 기능 예측의 자동화 및 기존의 기능 주석에 대한 검증의 의의가 있다.

유전체 서열 군집분석

유전체 서열 분석 작업 중 생물정보학의 중요한 주제인 유전자의 기능을 예측하는 문제가 있다. 유전체의 전체 염기 서열 중 발현되는 것으로 추측된 염기 서열의 단백질군(5, 11)을 찾는 작업은 이미 많은 부분이 연구자들의 실험과 수작업, 계산학적인 방법으로 밝혀지고 데이터베이스화 되어 있으며, 전체 염기서열이 밝혀진 유전체의 수가 많아진 현재, 새로 획득한 전유전체 서열의 유전자 지도작성에 유사한 종에서 축적된 염기서열-단백질 정보를 활용하는 계산학적인 방법이 연구되어졌다. 이러한 계산학적인 접근방법은 상동성 검색에 의한 기능주석 예측방법(function annotation prediction) 혹은 접근방법에 따라 유전체 서열 군집분석(genome sequence clustering)이라 명명한다.

본 논문에서는 유전체 서열의 비교분석을 통해 유전자의 기능을 예측하는데 목적을 두고 있다. 이와 같은 연구방법이 필요한 이유는 유전체 서열에서 단백질 생산물과 기능을 예측하는 실험비용이 크기 때문이다. 알려진 ACRs (ancient conserved regions, 8)이 많아지면서 아직 규명되지 않은 개체 유전자의 기능을 예측하기 위해서 이미 조사되고 정리된 유전자 서열과의 유사성을 검사하는 접근방법이 사용되었다. 이 중 유전체 서열 군집분석을 통해 유전자의 기능을 적절한 해상도로 예측하는 연구(7, 9, 12, 20)가 있었다. 이들 연구의 대부분은 유전자 서열간의 유사도 그래프에서 연결도가 높은 부분 그래프(highly connected subgraph)를 찾는 방식을 택했으며, 유사도 그래프를 희소화하기 위해 유사도에 대한 임계값과 같은 주관적인 파라미터를 사용하였다.

알려진 서열들에 대한 상동성 검색으로 연관성 있는 서열을 알아내는 작업은 생물학자들이 가장 빈번히 사용하는 분석방법이다. 이와 같은 분석을 위해 제안된 Smith-Waterman 알고리즘은 문자열에 대한 편집거리(edit distance)라는 계산학적인 모델에 바탕을 두고 있으며(10, 19), 주어진 모델에 대한 최적해를 구할 수 있지만 계산비용이 실용적이지 않은 문제가 있다. 이전 연구(7, 9, 12, 20)에서 유전자 서열간의 유사성 측정을 위해 사용한 FASTA, BLAST와 같은 유사 서열 검색 방법은 Smith-Waterman 알고리즘의 계산

*To whom correspondence should be addressed.
Tel: 054-279-8194, Fax: 054-279-5540
E-mail: yblee@bric.postech.ac.kr or yblee@pearl.cs.pusan.ac.kr

비용을 줄이기 위해 해싱(hashing), 오토마타(automata)의 휴리스틱(heuristic)을 적용한 예이며, 계산비용이 실용적이라는 이유로 민감도와 선택성을 희생하며 사용되고 있다(2, 16).

유전자를 기능을 중심으로 군집분석한 대표적인 연구로 COGs (Clusters of Orthologous Groups of proteins, 20)를 들 수 있다. COGs는 bacteria, archaea, eukaryote의 유전체에서 발현되는 유전자를 모아 orthologous 보존성향을 기준으로 군집분석한 결과이다. 이때 유전자 서열이 모인 각각의 군집을 COG라 부르며, 각 COG에 속한 단백질 서열은 공통되는 기능을 가지거나, ACRs를 포함하고 있을 빈도가 높다. 여러 종의 전유전체 염기서열에서 추출한 단백질 서열들을 분석하기 위해 COGs에서는 여러 전유전체에서 유래한 단백질 집합의 모든 가능한 짝에 대해서 BLAST 검색을 수행, 상호 최근접 유사성(reciprocal similarity) 관계를 보이는 서열들을 찾는다. 강한 유사성을 보이는 유전자 서열들을 묶고, 진화계통도나 BLAST 검색 등을 수행해 일일이 확인하는 과정을 거쳐 초기 유전자 집합을 생성했다. 이와 같은 과정을 거쳐 orthologous하게 보존되거나, 최소 3계통을 대표하는 단백질 집합을 정의할 수 있다.

GeneRAGE 방법(7)은 유전자 서열간의 유사성에 대해 BLAST 방법의 E-value 임계값 10^{-10}과 Smith-Waterman 방법의 Z-score 임계값 10을 기준으로 강한 유사성을 찾아내어 묶어주고, 나머지에 대해 계층적인 군집분석을 행하였다. Matsuda등이 1999년 제안한 *p*-quasi 완전그래프 방법(12)은 공통된 기능을 가지는 단백질 집합의 이상적인 모델은 완전그래프(complete graph)일 것이라는 가정 하에, 유전자 서열에 대한 유사성 그래프의 부분 그래프 중 각 유전자가 적어도 *p*개 이상의 다른 유전자와 에지(edge)를 가질 때에 해당하는 *p*-quasi 완전 그래프 집합을 응용하였다.

유전체의 기능 주석달기에 대한 계산학적인 접근 방법들은 많은 비용이 드는 실험들과 연구자들의 수작업에 의한 주석 작업을 줄인 의의가 있지만 정확성면에서의 문제점을 가진다. 이는 계산학적인 접근 방법들에서 가정한 모델들이 실제 생명체의 작용 방법과 차이가 있으며, 기존의 축적된 서열의 기능에 대한 주석 정보 자체가 오류를 포함할 수 있는 이유도 있지만 보다 큰 요인은 기존의 계산학적인 방법들이 특성값에 대한 주관적인 임계값을 가정하고 있는 것이다.

연구자의 주관적인 임계값이 필요한 분석 방법은 그 결과가 안정적이지 않으며, 분석결과만으로는 타당성과 설득력이 부족하다. Kim이 2002년 제안한 BAG (Biconnected components and Articulation points based Grouping of sequences, 9)은 각 유전자 집합을 결정하는 임계값을 동적으로 찾는 방법을 제안하였으며, 주관적인 임계값 사용을 완전히 배제하지는 못했지만, 최소 공통 서열 길이 등과 같이 연구자가 제공하는 파라미터에 대해 분석 결과가 상대적으로 안정적이다. BAG에서는 두 유전체의 유전자 집합에 대한 FASTA 검색 결과를 여러 단계의 Z-score 임계값으로 희소화시키는 실험을 하였다. Z-score 임계값에 대해 서로 연결된 컴포넌트(biconnected component) 수가 최대가 되는 Z-score 값을 유효 임계값의 하한값으로, 접합점(articulation point)의 수가 일정 비율 이상으로 감소하기 시작하는 Z-score 임계값을 상한값으로 가정하였다. 서열을 정렬하였을 때 알 수 있는 공통 부분 서열의 크기와 위치를 기준으로 주어진 부분그래프가 하나의 단백질 집합으로 충분한지, 혹은 여러 단백질 집합으로 나뉘어야 할 것인지를 검사하고 반복적으로 분할/병합한다. BAG 방법에서는 직관적이지 않은 서열 유사성에 대한 임계값 사용을 줄이고 공통 부분서열의 길이에 대한 임계값을 사용하였다.

기능 주석달기에 대한 기존의 서열 군집분석 알고리즘은 유전자 상동성을 판단하는 주관적인 임계값을 필요로 하며, 본 논문에서는 연구자가 부여하는 임계값의 입력에 따라 결과가 안정적이지 않은 기존 방법들의 단점을 보완해 입력서열 집합에 적합한 임계값을 동적으로 계산하는 안정적인 유전체 서열 군집분석 방법론을 제안하고 제시된 방법을 실제 데이터에 적용해 정확성과 안정성을 보이고자 한다.

Nonparametric method for Genome Sequence clustering

본 논문에서는 여러 유전체의 단백질 생성 유전자를 상호 비교해 생물학적으로 충분한 상동성을 보이는지의 기준에 따라 유전자 집합을 군집분석하는 녹섹(NOGSEC, NOnparametric method for Genome Sequence clustering) 알고리즘을 정의하고 유전자 서열 군집분석 시스템을 개발하였다. 유전자 서열 군집분석 시스템의 입력은 각 유전자의 기능에 대한 충분한 주석달기가 이루어진 유전체의 서열과 기능 주석달기를 원하는 유전체의 서열이며, 서열간의 유사성 비교를 위해 FASTA 3.3(16)을 사용하였다. FASTA 툴로 두 유전체의 유전자간에 상호 비교를 행하고 그 결과에 충분히 엄격한 임계값을 적용해 생물학적으로 의미가 있다고 보여지는 유사성 그래프를 작성한다. 반복적인 그래프 분화를 통해 상동성이 있다고 추측되는 유전자 부분집합을 구하게 되며, 이 부분집합이 충분한 해상도로 단백질 서열의 기능을 결정할 수 있는지 확인한다.

유전자 서열 군집분석 시스템의 목적은 군집분석방법으로 유전체의 기능주석을 예측해 주는 것이다. 임의 두 유전자 서열의 유사도는 두 유전자의 진화거리가 가까운 정도를 의미하며, 유전자 서열을 의미하는 두 문자열간의 편집거리(edit distance, 10, 19)를 계산하는 것으로 두 유전자의 유사도를 추정할 수 있으며 이는 계산생물학 및 생물정보학에서 흔히 사용되고 있는 방법이다.

다른 시각에서 임의 두 유전자가 상동성이 있다는 것은 두 유전자가 공통된 부분 서열을 가지고 있으며, 이는 두 유전자 외에 이와 유사한 기능을 하는 모든 유전자에서 공통적으로 발견되는 보존 서열(conserved sequence) 혹은 도메인(domain)이 있음을 의미한다. 따라서 공통 도메인의 데이터베이스를 구축하면 패턴 검색으로 유전자의 기능을 추측할 수 있으며, 이는 PROSITE, BLOCK, PRINTS, SMART와 같은 도메인(혹은 pattern, profile) 데이타베이스에서 사용되고 있는 방법이다(3, 4, 17, 18).

많은 유전체를 분석 대상에 포함시킬수록 더욱 정확한 공통 도메인을 정의할 수 있을 것이고, 검색할 수 있는 도메인의 수도 많아지겠지만, 계산 생물학에서 흔히 사용되는 서열 검색 방식이 시간복잡도가 높기 때문에 분석된 모든 유전체를 분석대상으로 하는 것은 불가능하다. 또다른 문제점으로 공통 도메인을 정의하

기 위해서는 공통된 기능을 하는 동류의 유전자 집합을 알고 있어야 하는데, 이는 도메인 검색외의 방법으로 유전자 집합을 밝혀내는 방법이 필요함을 의미한다. 이와 같은 이유로 적은 수의 유전체를 비교 분석해 새로운 유전체의 기능을 예측해 주는 유전자 서열 군집분석이 연구되었으며 COG(20), BAG(9) 등과 같은 방법들이 제안되었다.

본 논문에서는 적은 수의 유전체, 특히 기능주석달기가 되어 있는 하나의 유전체와 분석되기 전단계의 유전체 하나를 상호 비교해 유사한 기능을 하는 유전자들로 구성된 유전자 집합을 밝혀내는 방법을 연구하였다. 분석되기 전의 유전자가 기능이나 생산되는 단백질이 알려진 유전자와 충분한 상동성을 보인다고 밝혀지면 이를 통해 새로운 유전자의 기능을 추측할 수 있다.

사용할 수 있는 입력은 각 유전자간에 상호 비교된 유사성 행렬이다. 이는 각 유전자가 정점에 해당하고 유사성이 에지의 가중치인 가중치그래프(weighted graph)로 볼 수 있으며, 상동성 있는 유전자 집합을 찾는 문제는 전체 그래프의 적절한 부분그래프를 구하는 문제가 된다. 이때 각 서열마다 진화거리가 다르기 때문에 하나의 임계값을 적용해 전체 그래프를 분화할 수 없다. 이는 그래프의 각 컴포넌트 내부 유사성과 컴포넌트사이의 유사성을 비교했을 때, 컴포넌트 내부 유사성이 더 작을 수도 있음을 의미하며, 유전자의 유사도 그래프를 분석하는데 있어 단순히 임계값만을 사용하지 않고 그래프의 구조를 참고로 해야 함을 의미한다. 따라서 서열 군집분석 알고리즘은 진화거리가 먼 상동 유전자들을 찾아낼 수 있어야 하며, 이때의 상동유전자들은 먼 유사성(remotely homology)을 나타낸다고 한다(15).

상동성 있는 유전자 집합을 찾기 위해 두번째로 고려해야 할 것은 다중도메인(multi domain)문제이다. 유전체의 진화과정에서 하나의 유전자가 다른 유전자를 일부 포함하는 현상이 일어난다. 이는 유전자의 주요 도메인 부분이 변이에 의해 다른 유전자에 삽입될 수 있음을 의미하며, 이와 같이 여러 유전자의 도메인을 하나의 유전자가 가지고 있을 때 이를 다중도메인이라 한다. 다중도메인을 표현할 수 있는 전산학적 모델이 필요하며, 서열 군집분석 알고리즘은 다중도메인을 밝혀낼 수 있어야 한다. 본 논문에서 구현하고자 하는 유전체 서열 군집분석은 유전자간의 유사성 행렬을 바탕으로 먼 유사성을 가지는 상동 유전자 집합을 찾아내는 문제로 볼 수 있으며, 이전 연구를 적용했을 때 다중도메인 문제로 확장될 수 있음을 보인다.

방 법

시스템은 크게 유사도 행렬 생성과 분석을 담당한 두 부분으로 구성되어 있다. 시스템은 문자열 처리에 적합한 펄(perl)언어와 자바(java)로 구현되었으며, Fig. 1과 같이 각 기능을 담당하는 모듈 별로 독립되어 파일 혹은 파이프라인 방식으로 데이터를 상호 전달하게 된다.

FASTA를 이용한 유전자 교차비교단계

유전자 교차비교단계에서는 두 유전체의 유전자집합에서 모든

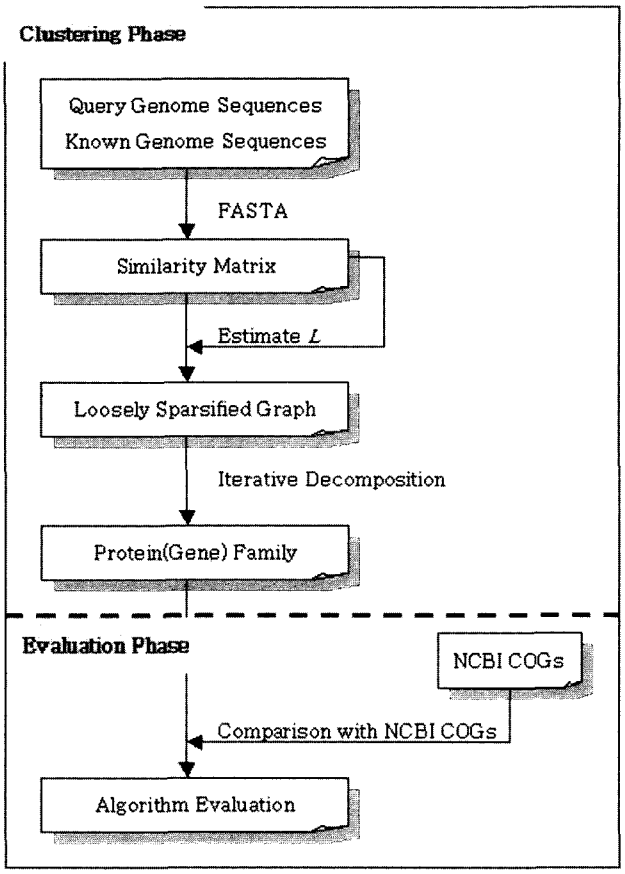

Fig. 1. The system overview. Clustering phase composed of calculation of similarity matrix, calculation of sparsifying threshold and iterative decomposition mudules.

가능한 쌍의 유전자에 대한 유사도를 계산하여 이를 유사도 행렬에 저장하게 된다. 유사도 행렬의 값 중 대부분의 값은 두 유전자의 유사성이 없음을 나타내기 때문에 실제 구현에서는 모든 유전자 쌍에 대한 유사도를 계산하지 않으며, 이와 같은 대규모 서열 비교를 위해 개발된 FASTA 툴을 사용한다. FASTA 툴은 대규모 서열 비교를 할 때 휴리스틱를 적용해 검색 공간을 줄이는 방식으로 서열 정렬 문제에 대한 근접한 해를 제공한다. FASTA 툴은 서열 데이타베이스를 검색해 주어진 질의 서열과 충분히 유사한 서열들을 찾아주며, FASTA 결과를 파싱해 인접 리스트 방식으로 저장한다.

절단 임계값 추정단계

유전자 집합에 대한 유사성 행렬이 구해지면, 이에 적당한 임계값을 적용해 유사성이 충분히 높은 유전자 쌍만을 선택한다. 이는 각 유전자가 그래프의 정점이 되고 두 유전자간의 유사도 값이 그래프에서 두 정점간 간선의 가중치인 완전그래프를 임계값 L로 희소화시키는 작업으로 볼 수 있다. 본 논문에서는 유전자 집합을 분석하기 앞서 유사도 만으로도 충분히 상동성이 있다고 판단할 수 있는 유전자들을 연관지어 주기 위해 충분히 엄격한 임계값을 구하고 이를 사용해 그래프로 표현된 유사성 행

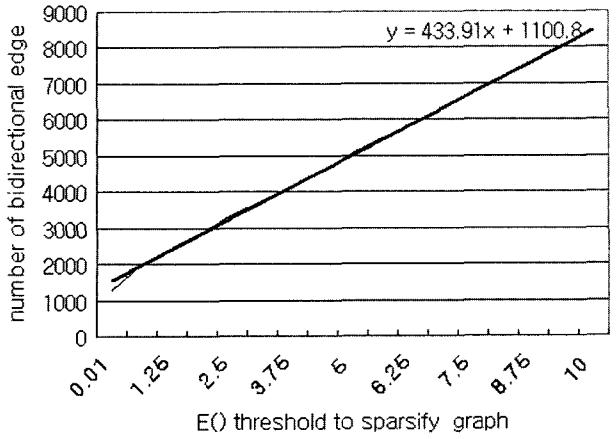

Fig. 2. Estimation of stable parameter L. Regression on the number of bi-directional edges with threshold parameter E() in the sparified similarity graph shows expected number of FASTA hits with zero E-value.

```
Sequence_Cluster_Analysis: 유전체의 서열 집합엣거 유전자를 기능별로 분류한다.
Program Sequence_Cluster_Analysis( Genome1, Genome2 ):

begin
        /* 유사도 행렬을 계산한다 */
        GenomeDB := Genome1 ∪ Genome2;
        {V,E} := FASTA(GenomeDB ⊗ GenomeDB);

        /* cut-off threshold L을 결정하고 그래프를 최소화시킨다 */
        L = cutoff_estimation( {V,E});
        E' := L-th smallest weighted edges from E;

        /* 반복적으로 completeness가 낮은 component를 분할한다 */
        queue := find compoents from {V',E'};
        foreach Ci in queue
        begin
            if |Ci| ≦ 4, report "protein family Ci found";
            else if test_decomposable(Ci), add decompose(Ci) on queue;
        end
end
```

Fig. 3. Pseudocode for the genome sequence cluster analysis method.

렬을 희소화시킨다. 이를 위해 유사성 행렬에 대해 다음의 실험을 행한다. 즉 FASTA의 결과값 중 E-value 값이 0.1 이하인 FASTA 검색 결과의 총 수를 구하고 E-value 값이 10.0 이하인 FASTA 검색 결과의 총 수를 구한다. E-value 값은 해당하는 검색 결과가 무작위 문자열에 대해 검색했을 때와 같이 우연히 발생할 기대값을 의미한다. FASTA 알고리즘의 특성상 E-value 값과 검색 결과수는 선형 관계를 가지며, 선형관계를 이용해 선분의 절편, 즉 E-value 값을 0으로 제한했을 때의 예상되는 검색 결과의 수를 추정하고 이를 그래프를 희소화 시키는 임계값 L로 사용한다.

이 과정을 Fig. 2로 설명하면 그림에서 E(), 즉 E-value 값이 0.1에서 10.0까지 변함에 따라 양 방향으로 동시에 강한 유사성을 보이는 양방향의(bi-directional) 간선 수는 선형으로 증가한다. 선형 회귀 분석에 의해 임계값 E()와 양방향의 간선 수는 y=433.91x+1100.8의 관계를 보인다. 준 수식의 y절편은 그래프의 절단 임계값 E()를 충분히 낮추었을 때, 즉 0으로 주어질 때의 예상되는 간선 개수이며 명확한 유사성을 보이는 핵심(core) 간선이라 할 수 있다.

유전자 집합 분석 단계

임계값으로 희소화 시키는 과정에서 그래프는 컴포넌트 집합으로 나뉘게 된다. 이때의 각각의 컴포넌트는 느슨한 임계값으로 인해 충분히 분리되지 못한 상태이며, 절단 임계값 추정단계에서의 설명과 같이 충분히 강한 유사성을 가지는 두 유전자간의 관계가 반영되어 있는 그래프가 된다. 따라서 각각의 컴포넌트에 대해 하나의 단백질 그룹으로 보고하거나, 혹은 단백질 그룹이 될수 있는 더 작은 컴포넌트들로 분할하는 반복적인 작업을 한다. 그 결과로 그래프를 구성하는 초기 컴포넌트집합을 각각의 단백질 그룹에 해당하는 컴포넌트 집합으로 나눌 수 있다.

임의 컴포넌트가 하나의 단백질 그룹으로서 충분한지, 아니면 여러 단백질 그룹이 섞여있어 추가로 분할해주어야 하는지를 판단하는 기준은 컴포넌트 내부의 연결성(connectivity), 즉 컴포넌트의 정점의 차수(degree)의 분포를 조사하여 결정한다. 다시 말해서 차수의 변이가 심한 컴포넌트는 여러 개의 단백질 그룹을 포함하고 있고, 차수의 분포가 일정하며 높은 값을 가지는 성향이 있다면 주어진 컴포넌트는 하나의 단백질 그룹이라 말할 수 있다. 이러한 방법으로 구해진 단백질 그룹은 절단 임계값 추정단계에서 정의된 핵심 간선을 바탕으로 하기 때문에 생물학적으로도 높은 정확성을 보인다.

유전자 집합 분석 알고리즘

FASTA 검색으로 얻어진 유사성 행렬에 의해 정의되어진 초기 그래프 G={V,E}는 절단 임계값 추정단계에 해당하는 알고리즘 cutoff_estimation의 반환값인 임계값 L에 의해 희소화 되어 그래프 G의 부분그래프(sub-graph)인 G'={V',E'}, |E'|=L를 생성하게 된다. 희소화된 부분그래프 G'는 임계값 L에 의해 분리된 컴포넌트 집합으로 볼 수 있는데, 두 개의 유전체를 비교 분석했을 때 나타나는 상동 유전자 집합의 크기가 2-4정도일 때 해상도가 적당하며 컴포넌트의 크기 $|C_i|\leq 4$일 때는 C_i가 충분한 completeness를 가지고 있으며, C_i에 포함된 유전자들이 충분한 유사성을 보인다고 할 수 있다. 반면에 컴포넌트의 크기가 클 때는 해당 컴포넌트에 여러 단백질 집합이 포함되어 있을 가능성이 크다. 주어진 컴포넌트가 completeness가 높은 컴포넌트(sub-component)로 나뉠 수 있는지 테스트하는 test_decomposable 알고리즘이 사용된다. 알고리즘 test_decomposable에 의해 컴포넌트로 나뉘어야 한다고 판단될 때는 분할 알고리즘을 사용해 컴포넌트를 나누게 되는데, 나뉘어진 각각의 컴포넌트에 대해서도 또한 반복적으로 test_decomposable와 분할 단계를 수행한다. 이와 같이 유전자 집합 분석 알고리즘은 cutoff_estimation 알고리즘으로 결정되는 임계값 L을 유사도 행렬에 적용시켜 컴포넌트 집합 $C=\{C_1,...,C_k\}$를 생성하고, 각 C_i가 충분히 분할 되었는지를

```
Algorithm cutoff_estimation: 강한 상동성을 안정적으로 보존하는 cutoff임계값을
계산
Program cutoff_estimation ({V,E});

begin
        /* 측정된 1차 방정식의 y절편을 구한다 */
        y(0,1) := |E'| where weight(E'i) < 0.1;
        y(10,0) := |E'| where weight(E'i) < 10.0
        return L := -1/99 y(10,0) + 10/9.9 y(0,1);
end
```

Fig. 4. Algorithm to calculate stable cut-off threshold L.

```
Algorithm test_decomposable: 주어진 component를 분할해야 하는지 검사
Program test_decomposable ({V,E});

begin
        /* 그래프 degree의 분산값이 1보다 클 때 decomposable하다 */
        STDEV := standard deviation of degree of V;
        if STDEV > 1, return true;
end

Algorithm decompose: 주어진 컴포넌트를 decompose해 준다
Program decompose ({V,E});

begin
        /* weigh가 작은edge부터 병합해서 두 개의 subcompoent 생성 */
        sort(E);
        E' = ∅, V' := V;
        foreach Ei in E
        begin
            E' := E' ∪ Ei;
            if #subcomponent =2, return subcomponents;
        end
end
```

Fig. 5. The simplest case of algorithm test_decomposable and decompose.

$|C_i|$와 test_decomposable 알고리즘 수행을 통해 판단해 필요한 경우 decompose 알고리즘을 수행한다. 이에 대한 가상코드를 Fig. 3에서 볼 수 있으며, 반복계산에 의해 임계값 L을 계산하는 cutoff_estimation 알고리즘을 Fig. 4에서 볼 수 있다.

알고리즘 test_decomposable과 decompose을 일반화시킬 수 있다. 즉 test_decomposable 알고리즘은 컴포넌트 내부의 차수 분포를 참고로 해 컴포넌트를 분할해야 하는지를 판단할 수 있는 어떤 방법이라도 사용이 가능하다. 또한 decompose 알고리즘은 주어진 컴포넌트를 여러 가지 평가함수가 최대화되는 방향으로 분할하도록 정의할 수 있다. 가장 간단한 형태의 test_decomposable과 decompose 알고리즘은 단순히 컴포넌트의 차수값들에 대한 분산값이 임계값 내에 있는지를 확인하고, 분산값이 클 때에는 컴포넌트가 양분될 때 까지 간선의 가중치를 기준으로 컴포넌트를 희소화시키는 것이다. 이때의 test_decomposable과 decompose 각각에 대한 알고리즘 정의는 Fig. 5에서 볼 수 있다.

```
Algorithm test_decomposable: 주어진 component를 분할해야 하는지 검사
Program test_decomposable ({V,E});

begin
        /* maximal p-quasi subgraph를 구해 p값이 충분히 큰지 확인한다 */
        p= maximal p-quasi subgraph of {V,E};
        if p'|V'| > P, return true;
end

Algorithm decompose: 주어진 컴포넌트를 decompose해 준다
Program decompose ({V,E});

begin
        /* p-quasi subgraph를 찾아 multidomain정보를 포함하여 분할한다 */
        C(0) := maximal p-quasi subgraph of {V,E};
        if test_decomposable ({V,E}- C(0) ) then
        begin
            C(1),C(2) := maximal bipartite subgraph of {V,E}- C(0);
            report "C(0) is candidate multidomain family";
            return C(0), C(1), C(2);
        end
        else return C(0), {V,E}-C(0);
end
```

Fig. 6. Algorithm extension example using *p*-quasi subgraph (12) for test_decomposable and decompose.

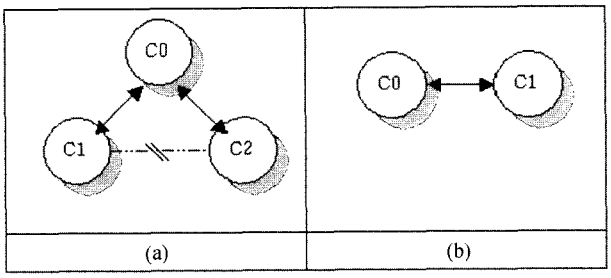

Fig. 7. Extended decompose algorithm using *p*-quasi subgraph (12) representing (a) protein families (C1, C2), multi-domain protein family (C0), and (b) two distinct protein families (C0, C1).

test_decomposable과 decompose 알고리즘을 확장 정의해 주어진 컴포넌트가 얼마나 큰 *p*-quasi 부분그래프를 가지고 있는지에 따라 분할 가능한지 판단하는 Matsuda의 방법(12)를 적용할 수 있다. 알고리즘 decompose의 보다 복잡한 형태는 Kim의 연구(9)에서와 같이 최대 이분 부분그래프(maximal bipartite subgraph)로 분할하는 것일 수도 있고 Matsuda(12)와 같이 최대 *p*-quasi 부분그래프로 분할 할 수도 있다. 최대 *p*-quasi 부분그래프(12)를 이용해 test_decomposable과 decompose 알고리즘을 정의하는 예는 Fig. 6에서 볼 수 있는데, 입력 컴포넌트에서 최대 *p*-quasi 부분그래프를 제외한 그래프가 분할 가능하다면 Fig. 7의 (a)와 같이 세 개의 부분 컴포넌트로 나눠지게 되며 특히 다중도메인 그룹일 가능성이 있는 C0를 찾을 수 있다. 만약 *p*-quasi 부분그래프를 제외한 부분그래프에 대해 test_decomposable이 실패한다면 이는 Fig. 7(b)와 같이 두 개의 서로 다른 유사한 컴포넌트로 분

리되고 각각에 대해서 반복적인 분할 과정이 적용된다. 이와같이 다중도메인을 표현하기 위해 *p*-quasi 그래프개념을 도입하고 응용분야에 맞는 test_decomposable과 decompose 알고리즘을 정의할 수 있다.

결 과

본 논문에서 소개된 두 개의 유전체의 상호비교를 통한 유전자 서열 군집분석방법을 BAG 실험(9)에 쓰였던 두 미생물 유전체에 적용하였고, 각 유전자에 대해 단백질 고유 ID (pid), 아미노산 서열, 관련된 대표적인 단백질 이름을 사용하였다. 이미 많은 부분 분석이 완료된 이들 미생물 유전체를 실험에 사용함으로 해서 논문에서 제시한 유전자 군집분석 알고리즘의 정확도를 간접적으로 알 수 있다.

실험에 사용된 하나의 유전체는 이미 알려진 지표로서의 유전체 역할을 하며, 나머지 하나는 아직 그 유전자가 분석되기 전이라 대규모의 계산학적인 상동성 검사가 필요하다고 가정한다. 입력으로 사용된 유전체는 (9)에서 서열 분석 알고리즘의 검증을 위해 사용한 *Borrelia burgdorferifull* (GENBANK accession number AE000783, 850 proteins)와 *Treponema pallidum* (GENBANK accession number AE000520, 1031 proteins)으로 총 1881개의 중복되지 않은 유전자를 NCBI에서 구하였다.

NCBI에서 얻은 단백질 라이브러리에 대해 FASTA 검색을 수행한다. 사용된 FASTA 버전은 3.3이며, fasta33-A-B-H-Q의 옵션을 사용하였다. FASTA 검색 결과는 각 유전자에 대해 유사성이 높다고 보여지는 몇몇 이웃 유전자의 목록과 그때의 일치하는 정도를 나타내는 Z-score 값 및 확률 기대치 E(), pairwise alignment를 제공하는데, 이와 같은 결과 파일에서 26,113개의 E() 및 정규화되지 않은 Z-score만을 파싱해 분석하기 쉬운 인접리스트 형태로 변환하였으며, 이는 유전자 서열 분석 시스템의 알고리즘 설명에서 나왔던 전체 유사성 행렬에 해당한다.

따라서 FASTA 검색의 결과로 E(), Z-score로 구성된 유사성 행렬이 있으며, 이를 cutoff_estimation 모듈에서 분석한 결과 절단 임계값 L은 1100으로 구해졌다. 임계값 L은 기존 방법들(7, 9, 12, 20)에서 사용했던 E-value, z-score 혹은 공통서열의 길이와 같은 임계값이 시행착오법(trial and error)과 같은 주관적인 방법으로 결정된 데 반해 유전자 유사도 분포를 기반으로 계산된 값이며, 생물학적인 상동성을 보장한다. 인접 리스트의 목록을 그래프로 보았을 때, 알고리즘의 초기 희소화 방법에 따라 가중치가 가장 작은 1100개의 에지로 구성된 E'와 767개의 유전자로 구성된 V'에 대해 Fig. 5에서 소개되었던 비교적 간단한 test_decomposable과 decompose 알고리즘을 적용하여 반복적인 그래프 분화를 한다. 끝으로 생성된 유전자 집합이 충분한 해상도와 정확도를 가지는 지를 검증하기 위한 실험이 행해졌다.

고 찰

본 논문에서는 주석달기가 되어 있는 두 유전체를 유전자 서열 군집 분석 알고리즘에 적용하여 기존에 알려져 있는 사실과 일치하는지 확인하였다. 입력으로 사용된 유전체에 대한 주석달기 정보는 "DNA-directed RNA polymerase (rpoA)"와 같이 구체적인 정보를 포함하고 있을 때도 있으며 "*T. pallidum* predicted coding region TP0134"과 같이 단지 단백질을 생성하는 것으로 추측된다는 정도의 정보를 가지고 있을 수도 있다. 따라서 유전체 서열에 주석달기가 완성된 정보 만으로는 기존 유전체가 어떻게 무리지어 지는지 충분히 세밀한 해상도로 알아내기가 힘들다. 본 논문에서는 유전체 서열 파일에 기록된 주석달기정보 외

Table. 1. Running example representing correct protein families with proper resolution.

OUTPUT	PID	COG ID	Protein Name
Grp125	3322698	COG 115	D-alanine glycine permease (dagA)
	3323324	COG 115	sodium/proton-dependent alanine transporter
Grp126	2688353	COG 706	inner membrane protein
	3323271	COG 706	membrane protein
Grp170	2688023	COG 1174	glycine betaine, L-proline ABC transporter, permease protein (pro-W)
	2688113	COG 0581	phosphate ABC transporter, permease protein (pstA)
	2688563	COG 1176	spermidine/putrescine ABC transporter, permease protein (pstB)
	2688564	COG 1177	spermidine/putrescine ABC transporter, permease protein (pstC)
	3322334	COG1175	sugar ABC transporter, permease protein (y4oQ)
	3322335	COG 0395	sugar ABC transporter, permease protein (y4oR)
	3322383	COG 2011	amino acid ABC transporter, permease protein (yaeE)
	3322954	COG 1176	spermidine/putrescine ABC transporter, permease protein (potB)
	3322955	COG 1177	spermidine/putrescine ABC transporter, permease protein (potC)
Grp2	3322264	N/A	tpr protein B (tprB)
	3322380	N/A	tpr protein C (tprC)
	3323357	N/A	tpr protein L (tprL)
Grp180	2687952	COG 0552	cell division protein, putative
	3322872	COG 0552	cell division protein (ftsY)
	2688634	COG 0541	signal recognition particle protein (ffh)
	3322699	COG 0541	signal recognition particle protein (ffh)

에 NCBI에서 유전체 서열과 함께 제공해 주는 단백질 ID/COG ID 대조파일을 추가로 사용한다.

단백질 ID/COG ID 대조파일에는 유전자의 ID, 유전체 내에서의 위치, COG group ID, 주요 단백질 생산물의 이름이 단백질을 생성할 것으로 기대되는 모든 유전자에 대해 나열되어 있다. 따라서 주어진 알고리즘의 결과와 기존의 주석달기 정보와의 일치 여부는 그룹 지어진 모든 유전자에 대해 서열 정보에서 단백질 생산물을 나타내는 단어들이 일치하거나 혹은 NCBI의 단백질 ID/COG ID 대조표에서 같은 COG에 해당하는지를 확인할 수 있으며, 기존에 구축된 문헌정보를 활용해 잘못된 답(false positive)를 측정할 수 있다. Table 1은 유전자 서열 군집분석 시스템의 결과에 원본 서열의 주석달기 정보와 COG 정보(1)를 조합해 비교가 되도록 한 출력 결과 예시이다. 성능평가 방법에 따라 문헌정보와 알고리즘의 출력물을 비교하였을 때, 알고리즘의 결과물이 정확성 면에서 좋은 성능을 보이는 것이 관찰되었다. 즉 알고리즘이 모든 유전자 집합을 찾아내지는 않지만 찾아낸 유전자 집합과 그 원소 유전자의 정확성이 높아 잘못된 답의 수가 적다.

논문에서 제시한 알고리즘으로 *B. burgdorferifull*와 *T. pallidum* 두 유전체를 분석해 총 433개의 유전자 집합을 찾아낼 수 있었으며 이는 Kim이(9) 찾아낸 441개에 근접한다. Table 1에서 단백질 생성물이나 COG 그룹과 같이 상세한 기준으로 분류된 유전자 군집의 예시를 보였다. Table 1에서 알 수 있듯이 Grp125 집합에는 같은 유전체의 gi3322698과 gi3323324의 두 유전자가 속해 있는 것으로 예측되었으며, 두 유전자는 기능주석의 키워드에서는 차이를 보이지만 같은 COG 그룹에 속한다고 명기되어 있다. Grp126에 속하는 두개의 유전자는 서로 다른 유전체에 속해 있으면서 동일한 기능을 수행하는 것으로 주석처리되어 있으며 COG 분석에서도 일치했다. Grp170에 속하는 9개의 유전자들은 COG 분석에서 일치하지 않았다. 하지만 기능주석을 확인했을 때 모든 9개의 유전자가 ABC transporter 단백질과 관련이 있는 것으로 나타나, 본 논문의 유전자 집합 분석 시스템이 충분히 세밀한 분류 기준으로 유전자들을 분류하는 데 성공했음을 보여준다. Grp2에서 볼 수 있는 세 가지 유전자는 기능주석을 확인했을 때 "tpr 단백질 그룹"에 속한 것으로 확인되었으며, 이 경우 각 서열에 COG 그룹 번호가 부여되지 않은 것으로 보아 본 논문의 유전자 서열 군집 분석 시스템이 COG 생성 알고리즘으로 분류할 수 없는 유전자 집합에 대해서도 분석이 가능하여 상보적으로 쓰일 수 있음을 보여준다.

알고리즘으로 생성된 유전자 집합 433개 중에서 COG 번호나 주석달기 상으로는 직접적인 상동성을 찾을 수 없는 10개의 유전자 집합에 대해 SMART (a simple modular architecture research tool, 18) 패턴 분석을 행했을 때 공통된 도메인을 찾을 수 있을 뿐 아니라 여러 개의 도메인이 조합되어 있는 것으로 보아 10개의 유전자 집합 모두 다중도메인 집합일 가능성을 보였다.

반면에 Grp180에 속하는 네 개의 유전자는 각각 두 개씩의 단백질 그룹이 합쳐진 것인데, 두 단백질 그룹간에 유사성이 있는지 여부는 주석달기 정보나 COG 정보로는 판단할 수 없다.

Table. 2. Running example representing clusters with improper resolution.

Name	Family with BAG[12]	Family with NOGSEC	Name
7.1	2688217, 2688606, 3322724, 322641	2688217, 2688606, 3322641	427
7.2	2687931, 2688317, 3322929, 3322641	2687931, 2688317, 3322929	100
		2688491, 3322724	227
18.1	2687964, 2688449, 2688636, 3322737, 3322802, 3323075	2687964, 2688449, 2688636, 3322737, 3322802, 3323075, 2688415, 3322451, 2688747, 3323208	25
18.2	2688415, 3322451		
18.3	2688747, 3323208		
452.1	3322593, 3322915, 3322925	3322593, 3322915, 3322925, 3322394, 3322594, 3322924	426
452.2	3322394, 3322594, 3322924, 3322925		
454.1	3322756, 3322400	3322756, 3322400, 3322399, 3322413, 3322755	129
454.2	3322399, 3322413, 3322755, 3322400		
465.1	3323180, 3323181	3323180, 3323181, 3323179, 3322844, 3323176, 3323177	146
465.2	3323179, 3323181		
465.3	3322844, 3323176, 3323177, 3323181		
58.1	2688707, 3322341, 3322811	2688314, 2688322, 2688460, 2688488, 2688489, 2688604, 2688707, 3322643, 3322930	432
58.2	2688488, 3322811		
58.3	2688488, 3322643		
58.4	2688460, 2688604, 3322643		
		3322341	430
		3322811	211

Left two columns are 6 multidomain protein families from BAG(9) algorithm, right two columns are from NOGSEC. Similarity of these list argues that imported decomposition method from BAG algorithm may improve ours.

Grp180에 속하는 유전자를 PROSITE 데이타베이스에 검색했을 때 각각의 유전자 서열이 네 다섯 군데의 보존 도메인을 가진 것으로 예측되었지만 발견되는 도메인들이 유전자 서열 전체에 걸쳐 빈번하게 발생하는 것으로 알려져 있어 그 중요성이 낮으므로 이런 경우 도메인 검색 방법(3, 4, 17, 18)을 통해서도 두 유전자의 관계를 정확히 알 수 없다.

Kim(9)이 제시한 유전자 군집분석 알고리즘이 다중도메인 성질을 띄는 유전자를 찾아낼 수 있는지를 확인하기 위해 Kim이(9) BAG 알고리즘으로 밝혀 낸 다중도메인 서열 목록과 비교하였다. Table 2에서 왼쪽은 BAG 알고리즘으로 찾아진 다중도메인 그룹의 목록이며 오른쪽 유전자 목록은 본 논문에서 제시한 방법으로 찾아진 그룹의 목록이다. Kim의 연구에서(9) BAG 방법으로 확인한 다중도메인 그룹은 모두 6개로, 본 논문에서

찾아진 유전자 그룹과 비교했을 때, 마지막 유전자 집합 58을 제외한 나머지 다중도메인 집합을 정확하게 찾을 수 있었으며, Kim의 연구에서 7.1집합에 속한다고 보고된 gi3322724 유전자를 실제 이와 가장 유사한 *B. burgdorferifull* 유전자 gi2688491과 그룹지어 독립된 단백질 집합으로 보고하였다. Kim의 연구에서 다중도메인을 찾아내기 위해 선정된 후보군이 본 논문에서 찾아 진 유전자 목록과 거의 일치하기 때문에, Kim이 다중도메인을 찾는 방법을 본 논문의 결과에 적용해 다중도메인을 찾도록 확장할 수 있음을 알 수 있다.

논문에서 제시한 NOGSEC 알고리즘은 전체 유사성 행렬 중 신뢰성이 높은 상위 일부 값만을 사용해 유전자 집합의 핵심집합을 생성하고, 여러 유전자 집합으로 나뉠 수 있는 컴포넌트를 찾아 분화시켰다. 논문에서 사용된 방법은 사용자의 명시적인 임계값이 필요한 기존의 방법들을 개선하여 유사도 행렬 자체에서 신뢰성 있는 엄격한 임계값을 계산하였으며, 그 결과로 해공간이 줄어드는 반면 그 정확도를 보장할 수 있었다. 논문에서 제시한 비교적 단순한 알고리즘만으로 Kim의 논문에서의 유전자 집합 수 441개에 근접한 433개의 집합을 찾을 수 있고 Kim의 연구에서 제시된 6개의 다중도메인 유전자 집합 중 5개를 유사하게 찾을 수 있음을 보였다.

그래프 이론을 적용해 유전자의 기능을 예측하는 기존의 방법들은 그래프를 희소화 시키는 주요 임계값들을 사용자로부터 입력받아야 하는 단점이 있었으며, 본 논문에서는 동류의 알고리즘에서 사용되는 유사도에 대한 임계값을 근접하게 계산할 수 있는 방법을 제시하였다.

계산된 임계값은 신뢰성이 있다고 믿어지는 유전자 유사도만을 반영하고 있으며, 엄격한 임계값으로 인해 결과의 신뢰성이 높지만 분석할 수 있는 유전자의 수가 줄어드는 장단점이 있다.

향후 유전자 기능 예측을 보다 정확하고 완전성이 있게 하기 위해서는 도메인 패턴 데이타베이스를 활용하고 기존 주석달기 정보를 충분히 활용한 복합적인 시스템이 구축되어야 하며, 연구자들에게 실험 결과를 정확하고 효율적으로 전달할 수 있는 GUI나 웹 인터페이스가 설계되어야 한다. 그리고 초기 희소화 그래프 생성으로 제외된 유전자를 본 방법론에서의 결과물인 유전자 집합와 비교해 근접한 유전자 집합을 찾아주는 연구가 필요하다.

감사의 글

본 연구는 부산대학교 연구비 지원을 받았습니다.

참고문헌

1. http://www.ncbi.nlm.nih.gov.
2. Altschul, S.F. 1990. Basic local alignment search tool. *J. Mol. Biol.* 215, 403-10.
3. Attwood, T.K., M.E. Beck, A.J. Bleasby, K. Degtyarenko, A.D. Michie, and D.J. Parry-Smith. 1997. Novel developments with the PRINTS protein fingerprint database. *Nucl. Acids Res.* 25, 707-710.
4. Bucher, P. and A. Bairoch. 1994. A generalized profile syntax for biomolecular sequences motifs and its function in automatic sequence interpretation. *Proc. Int. Conf. Inteli. Syst. Mol. Biol.* 53-61.
5. Dayhoff, M.O. 1978. Survey of new data and computer methods of analysis. Atlas of protein sequence and structure.
6. Devos, D., and A. Valencia. 2000. Practical limits of function prediction. *PROTEINS, Structure, Function, and Genetics* 41, 98-107.
7. Enright, A.J. and C.A. Ouzounis. 2000. GeneRAGE: A robust algorithm for sequence clustering and domain detection. *Bioinformatics* 16, 451-457.
8. Green, P., D. Lipman, L. Hillier, R. Waterston, D. State, and J.-M. Claverie. 1993. Ancient conserved regions in new gene sequences and the protein databases. *Science* 259, 1711-1716.
9. Kim, S. 2003. Graph theoretic sequence clustering algorithms and their applications to genome comparison, pp. 81-116. *In* J.T.L. Wang, C.H. Wu, and P.P. Wang (eds), Computational Biology and Genome Informatics. World Scientific Publishing Company, New Jersey.
10. Levenshtein, V.I. 1966. Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady, 707-710.
11. Mount, D.W. 2001. Bioinformatics: Sequence and Genome Analysis. Cold Spring Harbor, New York.
12. Matsuda, H., T. Ishihara, and A. Hashimoto. 1999. Classifying molecular sequences using a linkage graph with their pairwise similarities. Theor. Comp. Sci. 210, 305-325.
13. Matsuda, H. 2000. Detection of conserved domains in protein sequences using a maximum-density subgraph algorithm. *IEICE* Trans. *Fundamentals* E83-A, 713-721.
14. Needleman, S.B. and C.D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequences of two proteins. *J. Mol. Biol.* 48, 443-453.
15. Pavel, A. Pevzner. 2000. Computational Molecular Biology, An Algorithmic Approach. MIT Press, Cambridge, Massachusetts.
16. Pearson, W.R. and D.J. Lipman. 1988. Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci.* 85, 2444-2448.
17. Shmuel, P., J.G. Henikoff., and S. Henikoff. 1996. The block database-a system for protein classification. *Nucl. Acids Res.* 24, 197-200.
18. Schultz, J., F. Milpetz, P. Bork, and C.P. Ponting. 1998. Smart, a simple modular architecture research tool: identification of signaling domains. *Proc. Natl. Acad. Sci.* 95, 5857-5864.
19. Smith, T.F. and M.S. Waterman. 1981. Identification of common molecular subsequences. *J. Mol. Biol.* 147, 195-197.
20. Tatusov, R.L., M.Y. Galperin, D.A. Natale, and E.V. Koonin. 2000. The cog database: a tool for genome-scale analysis of protein functions and evolution. *Nucl. Acids Res.* 28, 22-28.
21. Tatusov, R.L., D.A. Natale, I.V. Garkavtsev, T.A. Tatusova, U.T. Shankavaram, B.S. Rao, B. Kiryutin, M.Y. Galperin, N.D. Fedorova, and E.V. Koonin. 2001. The cog database: new developments in phylogenetic classification of proteins from complete genomes. *Nucl. Acids Res.* 29, 22-28.

(Received February 4, 2003 /Accepted April 1, 2003)

ABSTRACT : NOGSEC: A NOnparametric method for Genome SEquence Clustering

Young-Bock Lee[1,2,*], Pan-Gyu Kim[1,3] and Hwan-Gue Cho[1,3] ([1]Department of Computer Science, Pusan National University, [2]Biological Research Information Center, Pohang University of Science and Technology, Research Institute of Computer Information and Communication, Pusan National University)

One large topic in comparative genomics is to predict functional annotation by classifying protein sequences. Computational approaches for function prediction include protein structure prediction, sequence alignment and domain prediction or binding site prediction. This paper is on another computational approach searching for sets of homologous sequences from sequence similarity graph. Methods based on similarity graph do not need previous knowledges about sequences, but largely depend on the researcher's subjective threshold settings. In this paper, we propose a genome sequence clustering method of iterative testing and graph decomposition, and a simple method to calculate a strict threshold having biochemical meaning. Proposed method was applied to known bacterial genome sequences and the result was shown with the BAG algorithm's. Result clusters are lacking some completeness, but the confidence level is very high and the method does not need user-defined thresholds.