구조적 특징에 기반한 대사 경로 드로잉 알고리즘

(An Algorithm for Drawing Metabolic Pathways based on Structural Characteristics)

이 소 희 † 송 은 하 †† 이 상 호 ††† 박 현 석 ††††
(So-Hee Lee) (Eun-Ha Song) (Sang-Ho Lee) (Hyun-Seok Park)

요 약 '생물정보학'이란 생물학적 데이타를 처리, 가공하여 정보를 얻어내는 연구 분야로 이 중 대사체학은 대사 경로 네트워크를 가시화하여 생명 활동을 이해하고자 하는 분야로, 대사 경로 내의 흐름을 한 눈에 알 수 있도록 가시화하여 보여 줄 수 있는 도구가 반드시 필요하다. 따라서 본 논문에서는 새로운 '대사 경로 드로잉 알고리즘'을 제안하였다. 대사 경로 그래프의 구조로는 이분 그래프를 이용하여 가독성을 높였으며, 이 그래프가 척도 없는(scale-free) 네트워크 구조라는 것과 구조적으로 환형, 계층적, 선형 컴포넌트를 가진다는 것을 고려하여 사이즈가 큰 그래프도 적절하게 드로잉 하도록 하였다.

키워드 : 생물정보학, 대사체학, 대사 경로 드로잉 알고리즘, 가독성, 이분그래프

Abstract Bioinformatics is concerned with the creation and development of advanced information and computational technologies for problems in biology. It is divided into genomics, proteomics and metabolimics. In metabolimics, an organism is represented by metabolic pathway, i.e., well-displayed graph, and so the graph drawing tool to draw pathway well is necessary to understand it comprehensively. In this paper, we design an improved drawing algorithm. It enhances the readability by making use of the bipartite graph. Also it is possible to draw large graph properly by considering the facts that metabolic pathway graph is scale-free network and is composed of circular components, hierarchic components and linear components.

Key words : bioinformatics, metabolic pathway, graph drawing, readability, bipartite graph

1. 서 론

2003년 4월 14일 '휴먼 지놈 프로젝트'가 종료되면서, 생물학적 데이타를 처리, 가공하여 유용한 정보를 얻어내는 연구 분야인 '생물정보학'이 더욱 부각되고 있다. 이 분야는 크게 유전체학, 단백질체학, 대사체학으로 나눠진다. 이 중, 대사 경로 네트워크를 그래프로 가시화하여 생명 활동을 총체적으로 이해하는 분야인 대사체학의 중요성이 부각되고 있다[1].

생명 활동을 유지하기 위한 기능들은 세포 내외에 존재하는 여러 물질들 간의 복잡한 네트워크를 통해 수행된다. 이 네트워크를 이해하는 것은 각 물질의 역할을 이해하고 그와 관련된 질병의 원인을 알아내는 데에도 중요하다. 이러한 네트워크는 그 기능과 참여하는 물질에 따라서 대사 경로, 조절 경로, 신호 전달 경로의 세 가지로 나뉜다[2]. 이 중 가장 기본적인 대사 경로는 화합물이 효소에 의해 다른 화합물로 변화하는 과정을 보여주는 네트워크로서 이미 많은 부분이 밝혀져 있다. 이러한 정보를 저장하고 있는 데이타베이스로는 KEGG, EcoCyc, WIT 등이 있다[3-5]. 연구를 수행하는 생물학자들은 이러한 대사 경로 정보를 검색하여 정보를 얻어낸다. 그런데 그 결과를 텍스트로 보게 되면 대사 경로의 본질적 복잡성 때문에 그 내용을 이해하기가 어렵다. 그러므로 생물학자들에게는 대사 경로의 흐름을 한 눈에 알 수 있도록 가시화하여 보여 주는 도구가 반드시

· 이 논문은 정보통신 선도기반기술개발산업의 연구비 지원에 의한 연구 결과임
† 비 회 원 : Impulse Japan사 근무
ssoy5502@hanmail.net
†† 학생회원 : 이화여자대학교 컴퓨터학과
ehsong@ewha.ac.kr
††† 종신회원 : 이화여자대학교 컴퓨터학과 교수
shlee@ewha.ac.kr
†††† 비 회 원 : 이화여자대학교 컴퓨터학과 교수
neo@ewha.ac.kr
논문접수 : 2004년 2월 6일
심사완료 : 2004년 8월 23일

필요하다.

기존의 '대사 경로 가시화 시스템'은 크게 정적인 시스템과 동적인 시스템으로 나눌 수 있다[1,3,6]. 이 시스템들은 대사 경로 네트워크를 그래프로 표현하여 본질적인 복잡성을 효과적으로 다루고 있다. 정적인 시스템의 예로는 대사 경로 다이어그램을 수작업으로 그려 비트맵 이미지 파일로 저장하고 있는 온라인 데이타베이스 시스템인 KEGG와 Boehringer Mannheim의 'Biochemical Pathways'를 스캔하여 웹으로 제공하는 ExPASy Molecular Biology Server가 대표적이다[3,7]. 그리고 그래프 드로잉 알고리즘을 적용하여 그 결과를 보여주는 동적인 시스템의 예로는 EcoCyc이 있다[4,6]. 이 시스템에서는 그래프 형태로 표현이 되어 있는 대사 경로에 적합한 그래프 드로잉 알고리즘을 적용하여 대사 경로상의 노드들을 적절하게 배치시킨 후, 그 결과를 보여준다.

생물학자들의 연구를 돕고자 제공되고 있는 이러한 '대사 경로 가시화 시스템'에는 아직 많은 문제점들이 존재한다. KEGG와 같은 정적인 시스템의 경우 대사 경로 정보가 업데이트 되었을 때 이미지를 다시 그려줘야 하고 각 대사 경로에 대해 준비되어 있는 이미지 하나로만 그 정보를 제공한다는 단점이 있다[1,3,7]. 또한 EcoCyc은 동적인 방법을 이용함으로써 대사 경로 정보가 업데이트 되는 경우에도 새로운 이미지를 그려서 보여 주는 것이 가능하지만 제공되는 이미지들이 컴포넌트들로 나누어진 작은 대사 경로들을 표현해주고 있기 때문에, 규모가 큰 대사 경로를 표현해주지 못 한다는 문제점이 있다[1,4,6].

본 논문에서는 동적인 방법에 초점을 맞추어 대사 경로 정보의 업데이트를 안정적으로 반영해 줄 뿐만 아니라 최근 크게 부각되고 있는 대사경로 그래프가 '척도없는(scale-free) 네트워크'라는 특징도 고려하여, 여러 형태의 컴포넌트가 복합적으로 존재하는 그래프도 그려줄 수 있는 개선된 대사 경로 드로잉 알고리즘을 설계 및 구현한다.

2. 대사 경로의 특징 및 기존 연구

2.1 대사 경로의 특징

2.1.1 척도 없는 네트워크

척도 없는 네트워크란 주어진 네트워크에 계속하여 추가되는 새로운 노드들이 특정한 소수의 노드에 링크되려는 성질을 가지는 네트워크를 말한다[9,10].

척도 없는 네트워크의 위상을 살펴보면 매우 높은 연결성을 가지는 소수의 노드들과 이 노드들에 연결되어 있는 낮은 연결성을 가지는 다수의 노드들로 이루어져 있다(그림 1 참조).

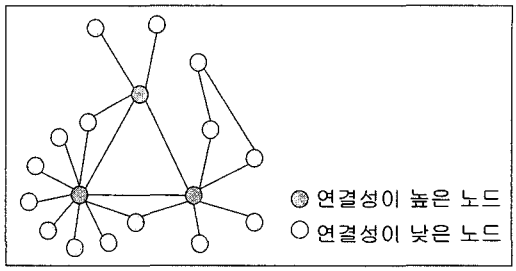


그림 1 척도없는 네트워크의 예

WIT(What Is There)내의 43개의 생물체의 대사 경로 네트워크를 분석해보면 대부분 척도 없는 네트워크에 속한다는 것을 알 수 있다[8,11]. 따라서 대사 경로 그래프를 레이아웃 해 줄 때, 연결성이 높은 단백질을 중심에 위치시킴으로써 가독성을 높일 수 있다[12].

2.1.2 구조적인 특징

일반적인 생화학 책에 실려 있는 대사 경로들을 살펴보면 대부분 환형 컴포넌트들과 계층적 컴포넌트들로 이루어져 있음을 알 수 있다[1]. 주어진 대사 경로에 환형 컴포넌트가 존재하는 경우 이 컴포넌트에 '환형 레이아웃 알고리즘'을 적용하고, 이 환형 컴포넌트와 연결이 되어 있는 다른 컴포넌트들을 환형 컴포넌트의 내·외부에 적당히 레이아웃 해 줄 수 있다[1,6]. 계층적 컴포넌트에는 일반적으로 많이 사용되는 '계층적 레이아웃 알고리즘'을 적용해 주는 것이 적합하다[1,6]. 또한, 대사 경로 내에서 빈번히 나타나는 긴 선형 컴포넌트는 '스네이크 레이아웃 알고리즘'을 적용해 주는 것이 적합하다[6].

2.2 기존 연구

2.2.1 EcoCyc에 적용된 알고리즘[4]

대사 경로 그래프를 표현하는 방법은 크게 화합물 그래프, 결합 그래프, 그리고 이분 그래프로 나뉜다[2]. 이 중 화합물 그래프란 반응물과 생성물로 이루어진 화합물의 집합을 노드 집합으로 하고 화합물들 간의 관계를 에지로 연결한 것으로 에지 상에는 효소의 이름이 라벨로 표현된다. EcoCyc에서는 대사 경로를 이러한 화합물 그래프로 표현하여 보여준다[4].

이 알고리즘은 주어진 대사 경로 그래프의 위상이 환형일 때는 '환형 레이아웃 알고리즘'을, 계층형일 때는 '계층적 레이아웃 알고리즘'을, 선형일 때는 '선형 레이아웃 알고리즘'을 적용한다. 복합형일 때는 그래프 내의 가장 큰 환형 컴포넌트를 찾아 이를 제외한 나머지 부분을 연결 성분들로 나눈다. 그리고 연결 성분들 중 환형 컴포넌트와 둘 이상의 에지로 연결되어 있는 것을 찾아 그 위상에 맞게 레이아웃 한 뒤 그 주위에 가장 큰 환형 컴포넌트를 레이아웃 하고 이 연결 성분과 환형 컴포넌트를 묶어서 슈퍼노드로 간주한다. 마지막으로

나머지 연결 성분들을 각각 그 위상에 맞게 레이아웃 한 뒤 각각을 슈퍼노드로 간주하여 슈퍼노드들로 이루어진 전체 대사 경로 그래프에 '트리 레이아웃 알고리즘'을 적용한다[6].

이 알고리즘은 '트리 레이아웃 알고리즘'을 적용할 때, 슈퍼노드 내의 노드들의 실제적 배치를 고려하지 않기 때문에, 발생할 에지 크로싱을 미리 예측하여 제거하는 것이 불가능하므로 전체적인 레이아웃 결과가 적절하지 않을 수 있다는 단점이 있다.

2.2.2 환형·계층적인 특징에 기반을 둔 알고리즘[1]

이 레이아웃 알고리즘은 논문 [1]에 소개되어 있는 알고리즘으로 EcoCyc과 마찬가지로 일반적으로 많이 사용되는 화합물 그래프로 대사 경로 그래프를 보여준다.

이 알고리즘에서는 주어진 대사 경로 그래프 내에 환형 컴포넌트가 없는 경우 '계층적 레이아웃 알고리즘'을, 그래프 자체가 환형 컴포넌트인 경우 '환형 레이아웃 알고리즘'을 적용한다. 복합적인 경우에는 그래프 내의 가장 큰 환형 컴포넌트를 찾아 이를 제외한 나머지 부분을 연결 성분들로 나눈 뒤, 환형 컴포넌트와 두 개 이상의 에지로 연결되어 있는 것을 내부 컴포넌트라 하고 나머지를 외부 컴포넌트라 한다. 그 다음 환형 컴포넌트의 내부에 내부 컴포넌트들을 위치시키고 외부에 외부 컴포넌트들을 위치시킨 뒤, 환형 컴포넌트에는 '환형 레이아웃 알고리즘'을 적용하고, 각각의 내부, 외부 컴포넌트들을 그 위상에 맞게 레이아웃 한다. 그리고 환형 컴포넌트를 제외한 모든 컴포넌트들을 슈퍼노드로 간주하고, 환형 컴포넌트의 위치를 고정시킨 채로 '스프링 임베딩 알고리즘'을 적용한다. 마지막으로 각 슈퍼노드 내에 있는 연결 성분들을 X, Y축을 기준으로 대칭시켜 본 후, 결과가 가장 적합한 것을 선택하여 그 슈퍼노드 내의 레이아웃으로 결정한다[1].

이 알고리즘은 화합물 그래프를 이용하여 대사 경로 그래프를 표현하고 있다. 화합물 그래프로 대사 경로 그래프를 표현하게 되면 화합물 간의 다(多) 대 다(多)관계를 표현하기 위해 하이퍼에지가 나타난다. 이러한 하이퍼에지는 레이아웃 결과에 잦은 에지 크로싱을 발생시킴으로써 가독성을 떨어뜨린다는 문제점이 있다[1].

3. 구조적 특징에 기반한 대사 경로 드로잉 알고리즘

3.1 고려 사항

위에서 살펴본 바와 같이 기존의 시스템들은 일반적으로 대사 경로 그래프를 화합물 그래프로 표현하는데, 이 화합물 그래프는 다음과 같은 단점들을 가지고 있다.

첫째, 해석이 모호한 경우가 많다[2].

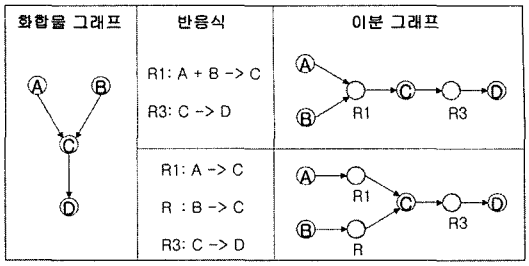


그림 2 화합물 그래프와 이분 그래프

그림 2의 화합물 그래프는 두 개의 반응식으로 해석될 수 있다. 이는 A와 B가 C와 함께 작용하는 것인지, 따로 작용하는 것인지 구분할 수 없기 때문이다. 이를 해결하기 위해서는 하나의 노드 집합은 화합물들로, 다른 하나의 노드 집합은 반응 노드들로 이루어져 있는 이분 그래프로 표현해 주는 것이 적합하다. 오른쪽 위의 그림은 A와 B가 함께 C에 작용한다는 것을 반응 노드 R1로 표현하고, 밑의 그림은 A와 B가 따로따로 C에 작용한다는 것을 반응 노드 R1, R2로 표현해 줌으로써 모호성을 제거하였다[2].

둘째, 하이퍼에지가 빈번하게 나타난다[1].

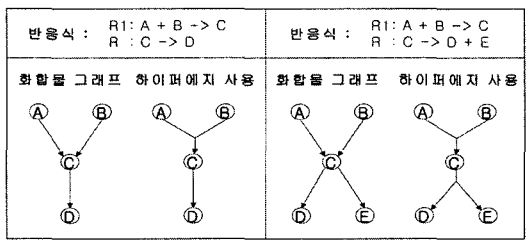


그림 3 하이퍼에지를 사용한 그래프

그림 3은 이러한 하이퍼에지의 예를 보여주고 있다. 이는 화합물 그래프의 단점을 해결하기 위해 화합물 간의 다(多) 대 다(多)관계를 에지를 확장하여 표현하는 과정에서 발생된다. 하지만 하이퍼에지를 이용한 표현은 해석의 모호성은 해결해주나 레이아웃 결과의 가독성을 떨어뜨리게 된다. 그림 4와 같은 경우 전체적인 레이아웃에 영향을 받아 불가피하게 일어난 에지 크로싱으로 인하여 실제적인 정보를 직관적으로 이해하기가 힘들게 되었음을 알 수 있다. 이러한 문제점은 하이퍼에지에 연결되어 있는 화합물의 수가 더 많아질수록 심각해진다.

본 논문에서 제안된 알고리즘에서는 이러한 단점들을 극복하기 위하여 이분 그래프를 대사 경로 그래프 구조로 사용하였다[2]. 기존에 연구된 이분 그래프에서는 하나의 노드 집합은 화합물들로, 다른 하나의 노드 집합은 반응을 표현하는 반응 노드들로 표현하였으나, 본 연구

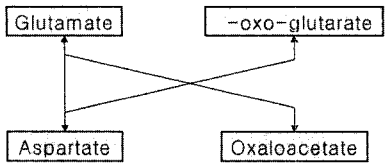


그림 4 에지 크로싱이 일어난 예

에서는 반응 노드 대신 효소들을 표현하는 노드 집합을 사용함으로써 하이퍼에지를 제거하여 가독성을 높혔고 하이퍼에지로 인해 발생하는 에지 크로싱을 줄여 대사 경로 그래프를 알아보기 쉽도록 하였다.

3.2 알고리즘

3.2.1 개요

본 알고리즘은 입력으로 들어온 대사 경로 그래프의 구조적인 특성을 고려하여 적합한 레이아웃 모듈을 적용한다.

알고리즘: 구조적 특징을 고려한 대사 경로 드로잉 알고리즘
입력: 대사 경로 그래프
출력: 각 노드의 좌표
과정:
1. 연결성이 높은 노드의 존재 여부 검사
 1.1 연결성이 높은 노드가 존재할 경우 '노드의 연결성을 고려한 레이아웃' 적용
 1.2 연결성이 높은 노드가 존재하지 않을 경우
  1.2.1 노드 사이즈가 6 이상인 환형 컴포넌트의 존재 여부 검사
   1.2.1.1 존재할 경우 '환형·계층적인 특징을 고려한 레이아웃' 적용
   1.2.2.2 존재하지 않을 경우 '계층적 구성요소의 레이아웃' 적용

입력으로 들어온 대사 경로 그래프에 대해 우선 고려되는 특성은 노드의 연결성이다. 실제 KEGG 내의 대사 경로 그래프의 노드들을 차수 별로 그룹을 지어 보았을 때, 차수가 5 이하인 노드들의 개수보다 6 이상인 노드들의 개수가 매우 적다. 그러므로 먼저 차수가 6 이상인 노드들을 '연결성이 높은 노드'의 후보로 지정한다. 다음으로는 레이아웃 결과의 가독성을 높여주기 위해 각각의 연결 성분에 에지가 최대 3개까지 연결되어 있을 경우에만 '노드의 연결성을 고려한 레이아웃'을 적용한다. 각 연결 성분에 연결된 에지의 수를 확인하는 방법으로는 시간적인 효율성을 위해 에지의 총 수와 연결 성분의 개수를 비교하는 방법을 사용하였다. 즉, '연결성이 높은 노드'의 후보와 인접한 에지의 수가 총 k라 하였을 경우 연결 성분의 개수가 k × 2/3개보다 크면 '노드의 연결성을 고려한 레이아웃'을 적용한다.

입력으로 들어온 대사 경로 그래프에 대해 두 번째로 고려되는 특성은 노드 사이즈가 6 이상인 환형 컴포넌트의 존재 유무이다. 이를 만족하면, '환형·계층적인 특징을 고려한 레이아웃' 알고리즘을 적용하고, 만족하지 않으면 '계층적 구성요소의 레이아웃'을 적용하고 마친다. 이 때 노드 사이즈가 6 미만이면 환형 컴포넌트에 내에 화합물이 두 개만 포함되므로 사이클이라는 의미가 없어지기 때문에 노드 사이즈가 6 이상인 것을 찾는다. 이 단계에서 좋은 레이아웃 결과를 얻기 위해서는, 가장 큰 환형 컴포넌트를 찾는 것이 중요하다. 그러나 이 문제를 푸는 것은 NP-complete 문제로 알려져 있다 [13]. 그러므로 본 구현에서는 휴리스틱한 방법을 이용하여 가장 크다고 여겨지는 환형 컴포넌트를 찾도록 하였다.

3.2.2 노드의 연결성을 고려한 레이아웃

노드의 연결성을 고려한 레이아웃 알고리즘은 대사 경로 그래프 상에 존재하는 연결성이 높은 노드들을 적절하게 위치시키자는 아이디어를 기반으로 제안되었다.

알고리즘: 노드의 연결성을 고려한 레이아웃
입력: 대사 경로 그래프, 차수가 6 이상인 노드의 집합(N_d)
출력: 각 노드의 좌표
과정:
1. N_d에 인접한 에지 제거 후 연결 성분들로 그룹핑
2. 각 연결 성분에 대해서
 2.1 노드 사이즈가 6 이상인 환형 컴포넌트의 존재 여부 검사
  2.1.1 존재할 경우 '환형·계층적인 특징을 고려한 레이아웃' 적용
  2.1.2 존재하지 않을 경우 '계층적 구성요소의 레이아웃' 적용
 2.2 연결 성분의 가로, 세로의 길이를 구하여 슈퍼노드로 저장
3. 입력 그래프를 N_d와 슈퍼노드, 제거되었던 에지들로 변형하여 '스프링 임베딩 알고리즘' 적용
4. 각 슈퍼노드에 대해서
 4.1 $k1$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 4.2 슈퍼노드를 X축 대칭 후, $k2$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 4.3 슈퍼노드를 Y축 대칭 후, $k3$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 4.4 슈퍼노드를 X축 대칭 후, $k4$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 4.5 $k1$, $k2$, $k3$, $k4$값의 대소 비교
  4.5.1 $k1$이 가장 적을 경우 슈퍼노드를 그대로 위치시킴
  4.5.2 $k2$가 가장 적을 경우 슈퍼노드를 X축 대칭하여 위치시킴
  4.5.3 $k3$이 가장 적을 경우 슈퍼노드를 X, Y축 대칭하여 위치시킴
  4.5.4 $k4$가 가장 적을 경우 슈퍼노드를 Y축 대칭하여 위치시킴

첫 단계로는 그림 5와 같이 연결성이 높은 노드들에 인접한 에지들을 제거한 후, 남은 부분들을 연결 성분들로 나누어 각각의 연결 성분을 각각 하나의 슈퍼노드로 간주한다.

그림 5와 같이 선택된 노드를 제외한 연결 성분들을

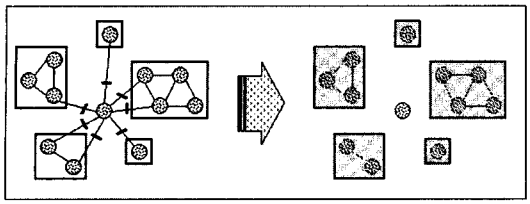


그림 5 연결 성분을 슈퍼노드로 그룹핑

각각 하나의 슈퍼노드로 간주하기 위해서 먼저 각 연결 성분에 그 구조에 적합한 레이아웃 알고리즘을 적용해 준다. 각 연결 성분에는 노드의 사이즈가 6 이상인 환형 컴포넌트가 존재할 경우 '환형·계층적인 특징을 고려한 레이아웃' 알고리즘이 적용되고, 존재하지 않을 경우 '계층적 구성요소의 레이아웃' 알고리즘이 적용된다. 연결 성분을 레이아웃 해 준 후 각 연결 성분의 가로, 세로의 값을 이용하여 해당하는 슈퍼노드의 가로, 세로의 길이로 지정해 준다.

그 다음 변형된 입력 그래프에 '스프링 임베딩 알고리즘'을 적용한다[14,15]. 그림 6에서 볼 수 있듯이 입력 그래프는 선택된 노드와 슈퍼노드, 그리고 제거했던 에지들로 이루어진 그래프이다. 이 때, 연결성이 높은 노드와 각 슈퍼노드를 연결하는 에지의 길이는 슈퍼노드들이 서로 겹치는 경우를 최대한 줄일 수 있도록 넉넉하게 잡아준다.

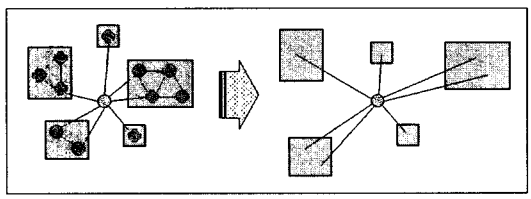


그림 6 슈퍼노드를 이용한 그래프의 변환

마지막 단계로 슈퍼노드 내의 레이아웃 되어 있는 연결 성분을 적절하게 대칭 시켜준다. 이는 에지 크로싱을 최소화하기 위한 휴리스틱한 시도이다.

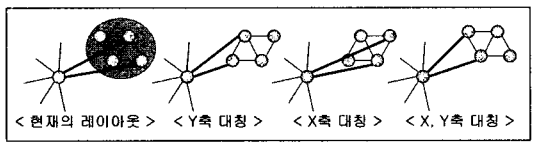


그림 7 슈퍼노드 내의 배치 결정

각 슈퍼노드에 해당하는 연결 성분에 대해 현재의 레이아웃, X축 대칭, Y축 대칭, 그리고 X, Y축 대칭의 4가지를 고려한다. 그리고 각각의 경우에서 이 슈퍼노드 내의 노드와 선택된 노드 간을 연결하는 에지들의 길이의 합이 가장 작은 것을 골라 그 경우를 연결 성분의 배치 상태로 결정한다. 그림 7과 같은 경우에는 'X, Y 축 대칭'을 한 레이아웃이 선택된다.

3.2.3 환형·계층적인 특징을 고려한 레이아웃

환형·계층적인 특징을 고려한 레이아웃 알고리즘은 대사 경로 그래프 상에 존재하는 환형 컴포넌트를 적절하게 배치하기 위하여 고안된 알고리즘이다.

알고리즘: 환형·계층적인 특징을 고려한 레이아웃
입력: 대사 경로 그래프, 노드 사이즈가 6 이상인 환형 컴포넌트(N_c)
출력: 각 노드의 좌표
과정:
1. N_c에 인접한 에지 제거 후 연결 성분들로 그룹핑
2. 각 연결 성분에 대해서
 2.1 노드 사이즈가 6 이상인 환형 컴포넌트의 존재 여부 검사
  2.1.1 존재할 경우 '환형·계층적인 특징을 고려한 레이아웃' 적용
  2.1.2 존재하지 않을 경우 '계층적 구성요소의 레이아웃' 적용
 2.2 연결 성분의 가로, 세로의 길이를 구하여 슈퍼노드로 저장
3. N_c에 '환형 레이아웃 알고리즘'을 적용
4. 각 슈퍼노드와 N_c의 무게 중심을 구함
5. 가장 큰 슈퍼노드의 무게 중심 방향에 N_c를 위치시키고, N_c를 그 무게 중심이 가장 큰 슈퍼노드 쪽으로 향하도록 회전
6. 입력 그래프를 N_c와 슈퍼노드, 제거되었던 에지들로 변형
7. N_c의 위치를 고정하고 변형된 입력 그래프에 '스프링 임베딩 알고리즘' 적용
8. 각 슈퍼노드에 대해서
 8.1 $k1$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 8.2 슈퍼노드를 X축 대칭 후, $k2$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 8.3 슈퍼노드를 Y축 대칭 후, $k3$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 8.4 슈퍼노드를 X축 대칭 후, $k4$=∑슈퍼노드 내의 노드와 N_d 사이의 에지들의 길이
 8.5 $k1$, $k2$, $k3$, $k4$값의 대소 비교
  8.5.1 $k1$이 가장 적을 경우 슈퍼노드를 그대로 위치시킴
  8.5.2 $k2$가 가장 적을 경우 슈퍼노드를 X축 대칭하여 위치시킴
  8.5.3 $k3$이 가장 적을 경우 슈퍼노드를 X, Y축 대칭하여 위치시킴
  8.5.4 $k4$가 가장 적을 경우 슈퍼노드를 Y축 대칭하여 위치시킴

첫 단계로는 그림 8과 같이 선택된 환형 컴포넌트에 인접한 에지들을 제거한 후, 그래프의 남은 부분들을 연결 성분으로 나눈다. 그리고 환형 컴포넌트를 제외한 각각의 연결 성분을 각각 하나의 슈퍼노드로 간주한다.

이 경우에도 각각의 연결 성분에 그 위상에 따라 적합한 레이아웃 알고리즘을 적용하여 연결 성분의 가로, 세로의 길이를 얻은 후, 슈퍼노드의 정보로 저장한다.

다음에는 환형 컴포넌트에 '환형 레이아웃 알고리즘'을 적용한 후, 이 컴포넌트를 회전시킨다. 이를 위해 먼저 각 슈퍼노드의 무게 중심을 구한다. 각 슈퍼노드의

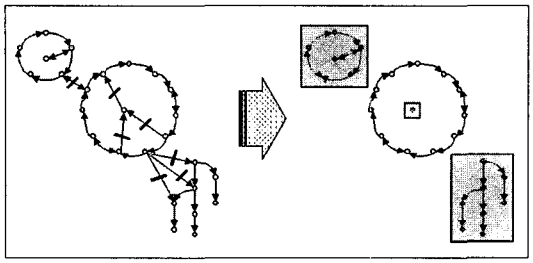


그림 8 연결 성분을 슈퍼노드로 그룹핑

무게 중심은 그 슈퍼노드 내의 노드들 중, 환형 컴포넌트에 연결되어 있는 노드들의 평균 좌표로부터 구한다. 각 슈퍼노드의 무게 중심은 평균 좌표의 위치에 따라 동, 서, 남, 북, 중심 중에서 하나로 결정된다. 이 과정에서, 노드의 수가 3개 이하이면서 환형 컴포넌트에 연결된 에지의 수가 가장 많은 하나의 슈퍼노드를 내부 컴포넌트로 체크해 둔다. 그 다음 가장 큰 슈퍼노드를 선택하여 이 슈퍼노드의 무게 중심 쪽에 환형 컴포넌트를 위치시킨다. 그리고 이 슈퍼노드에 대한 환형 컴포넌트의 무게 중심을 구한 후, 이 무게 중심이 슈퍼노드의 무게 중심에 최대한 가까워지도록 환형 컴포넌트를 회전한다. 그림 9를 보면, 왼쪽의 선택되어 있는 슈퍼노드의 무게 중심이 '동'이므로 환형 컴포넌트를 이 슈퍼노드의 동쪽에 위치시킨다. 그리고 계산한 환형 컴포넌트의 무게 중심이 슈퍼노드의 바로 앞으로 오도록 화살표만큼 회전시킨다.

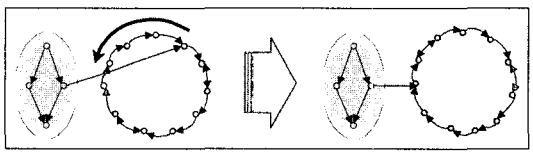


그림 9 환형 컴포넌트의 회전

레이아웃 된 환형 컴포넌트는 고정한 채로 '스프링 임베딩 알고리즘'을 적용한다. 입력 그래프는 위치가 고정되어 있는 환형 컴포넌트와 슈퍼노드, 그리고 제거했던 에지들로 이루어진다. 우선 슈퍼노드의 초기 위치를 지정해주는데, 내부 컴포넌트로 체크된 슈퍼노드는 환형 컴포넌트의 중심에 위치시켜주고, 다른 슈퍼노드들에 대해서는 각 슈퍼노드에 대한 환형 컴포넌트의 무게 중심을 구하여 그 무게 중심의 방향에 그 슈퍼노드를 위치시킨다. 연결된 에지의 길이는 내부 컴포넌트의 경우에는 짧게, 다른 슈퍼노드들에 대해서는 넉넉하게 지정해 준다. 이렇게 초기 설정을 해 준 후 적용한 결과는 그림 10과 같이 나타난다.

마지막 단계로 슈퍼노드 내의 연결 성분을 적절하게

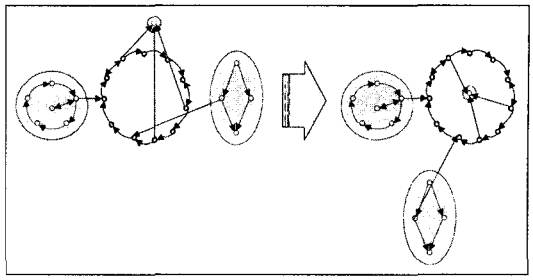


그림 10 스프링 임베딩 알고리즘 적용

대칭 시켜준다. 이는 '노드의 연결성을 고려한 레이아웃'의 경우와 동일하다.

3.2.4 계층적 구성요소의 레이아웃

계층적 구성요소의 레이아웃 알고리즘은 위의 두 조건을 만족시키지 못한 그래프에 대해 적용되는 알고리즘으로, 입력 그래프의 구조적 특성에 따라서 그래프를 레이아웃한다.

알고리즘: 계층적 구성요소의 레이아웃
입력: 대사 경로 그래프
출력: 각 노드의 좌표
과정:
1. 대사 경로 그래프가 선형일 경우
   1.1 노드 사이즈가 10 이하일 경우 '직선형 레이아웃'을 적용
   1.2 노드 사이즈가 10 이하가 아닐 경우 '스네이크 레이아웃'을 적용
2. 대사 경로 그래프가 트리 구조일 경우 '트리 레이아웃'을 적용
3. 대사 경로 그래프가 선형도 트리 구조도 아닐 경우 '계층적 레이아웃'을 적용

계층적 구성요소는 선형, 트리, 그리고 이 두 가지에 속하지 않는 세 가지의 구조로 구분된다. 선형 구조는 그 길이에 따라 노드 사이즈가 10 이하일 경우에는 '직선형 레이아웃'을, 10보다 클 경우에는 '스네이크 레이아웃'을 적용한다. 트리 구조일 경우에는 일반적인 '트리 레이아웃'을, 아무 조건에도 속하지 않는 구조일 경우에는 '계층적 레이아웃'을 적용한다[14,15].

4. 실험 및 실험 결과

4.1 사용 언어와 실행 환경

본 실험은 마이크로소프트 윈도우 98 환경에서 수행하였으며, Java 프로그래밍 언어와 Java 기반 그래프 라이브러리인 YFiles를 사용하여 구현하였다[16].

4.2 레이아웃 알고리즘의 적용

그림 11의 경우 노드의 연결성을 고려한 레이아웃이 적절하게 적용되어 있음을 볼 수 있다. 또한, 그림 12를 보면 복합적인 경우에도 환형·계층적인 특징을 고려한 레이아웃, 계층적 구성요소의 레이아웃이 적절하게 적용되어 있음을 볼 수 있다.

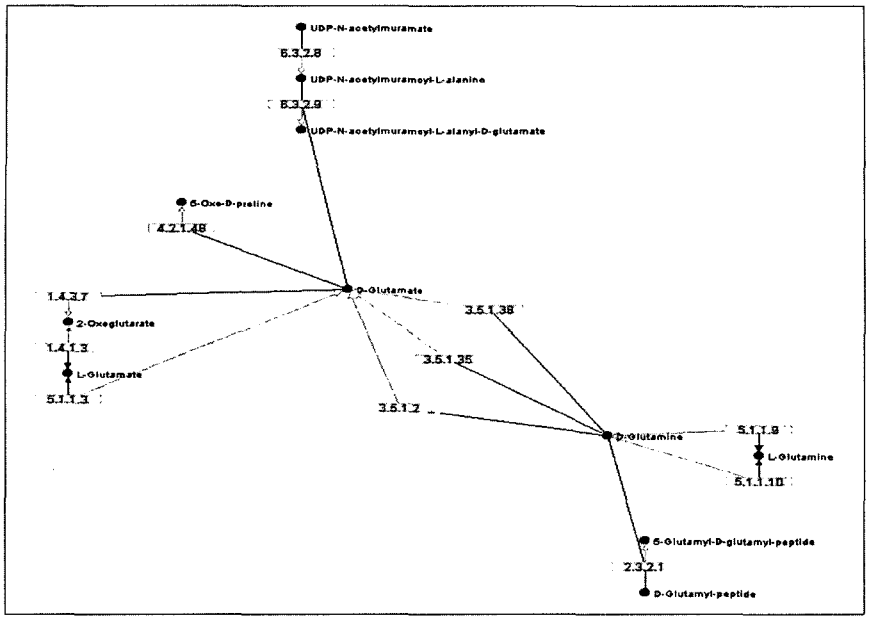


그림 11 노드의 연결성을 고려한 레이아웃 적용

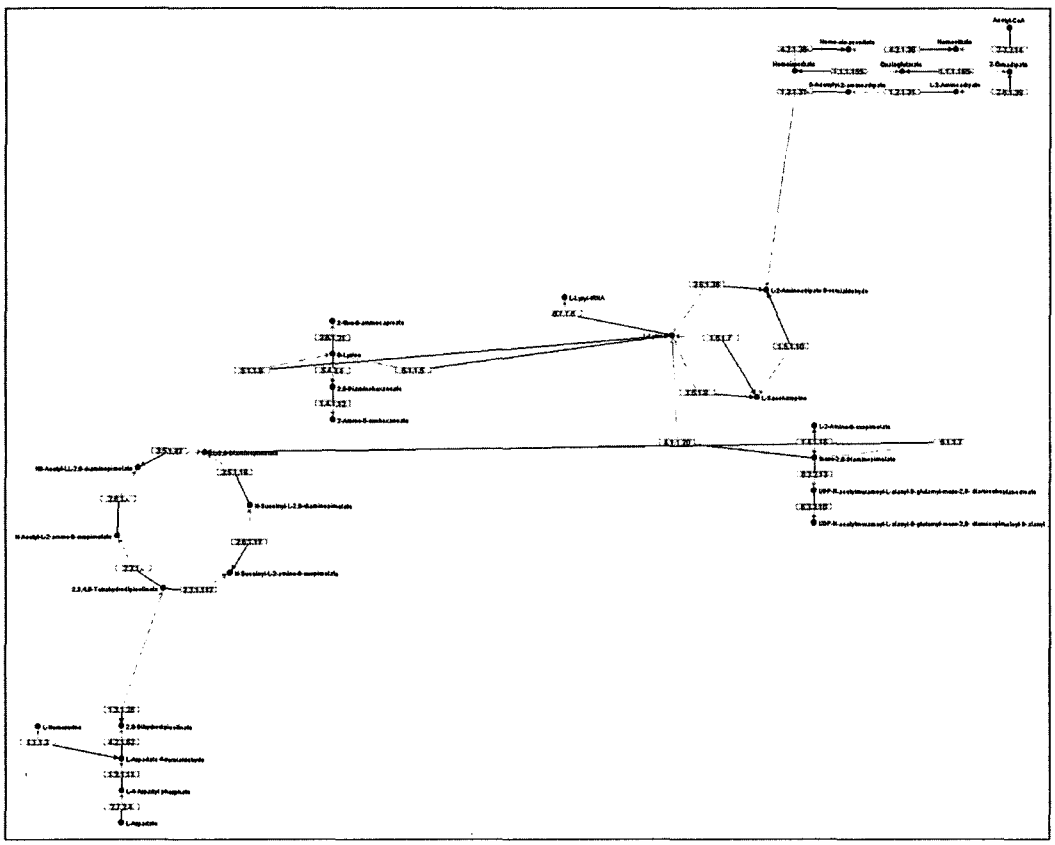
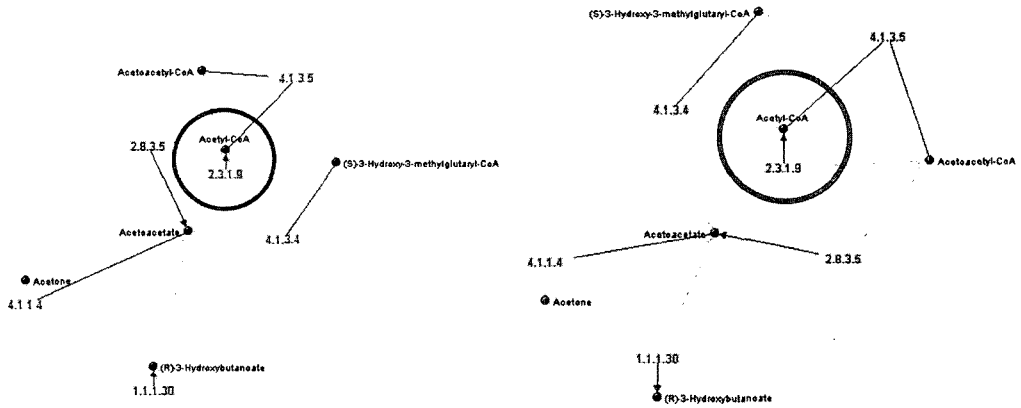
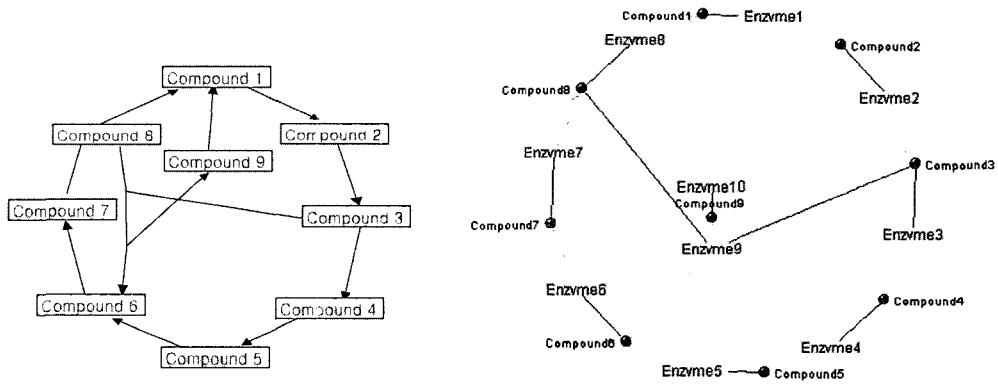


그림 12 복합적인 경우의 레이아웃 결과



(a) EcoCyc에 적용된 알고리즘 (b) 본 논문에서 제안한 알고리즘

그림 13 EcoCyc에 적용된 알고리즘과의 비교



(a) 환형·계층적 특징에 기반을 둔 알고리즘 (논문 [1]) (b) 본 논문에서 제안한 알고리즘

그림 14 환형·계층적인 특징에 기반을 둔 알고리즘과의 비교

4.3 기존 시스템과의 비교

4.3.1 EcoCyc에 적용된 알고리즘

EcoCyc에 적용된 알고리즘에서는 슈퍼노드 내의 노드들의 배치를 고려하지 않으므로 전체 레이아웃이 항상 적절하지는 않다. 그림 13을 보면 본 논문에서 제안한 알고리즘을 적용 했을 때 노드와 에지의 겹쳐짐이 줄어든 결과를 얻을 수 있다는 것을 알 수 있다.

4.3.2 환형·계층적인 특징에 기반을 둔 알고리즘

이 알고리즘은 하이퍼에지의 표현에 있어서 에지 크로싱이 발생했을 뿐만 아니라, 효소에 대한 정보도 전혀 포함하고 있지 않다. 그림 14를 보면 본 논문에서 제안한 알고리즘을 적용하여 효소에 대한 정보를 포함시키고, 하이퍼에지를 제거하여 가독성을 높였음을 알 수 있다.

4.4 알고리즘 수행 시간

본 논문의 제안 알고리즘의 수행 시간을 노드 사이즈에 따라서 측정하였다. 측정한 수행 시간은 각 노드 사이즈에 대한 평균 수행 시간 값이다.

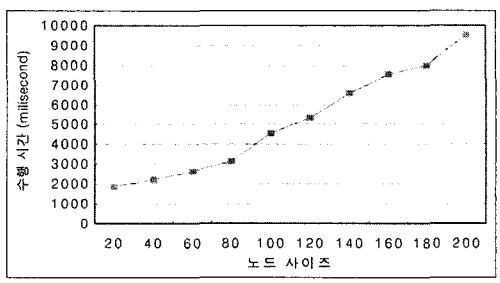


그림 15 레이아웃 알고리즘 수행 시간

그림 14의 그래프를 보면 노드 사이즈가 약 200인 경우 약 10초의 시간이 걸리는 것을 알 수 있다. KEGG 내의 대사 경로의 노드 사이즈는 대부분 100 내외 이므로, 드로잉 결과를 얻기 위해 그다지 많은 시간이 소요되지는 않는다.

5. 결론 및 향후 연구과제

본 논문에서는 대사체학에 있어서 필수적인 '대사 경로 가시화 시스템' 구축에 있어서 필요한 그래프 레이아웃 알고리즘을 제안 및 구현하였다. 이 알고리즘은 기존에 구현되어 있는 시스템들의 단점들을 극복하기 위하여 이분 그래프 구조의 대사 경로 그래프를 입력으로 사용하였으며, 각각의 대사 경로 그래프의 특징에 적합하게 레이아웃을 해 주었다. 그러나 모든 경우의 그래프에 대해 적절한 결과를 보여주는 완벽한 그래프 레이아웃 알고리즘이 존재하지 않듯이, 이 레이아웃 알고리즘도 노드의 사이즈가 커질수록 그 결과의 가독성이 떨어지게 된다. 이러한 결과는 비록 완벽하지 않다 하더라도 최소의 수정을 통하여 가장 적합한 드로잉 결과를 얻을 수 있는 초기 드로잉으로써 유용하게 사용될 수 있다.

향후 과제로는 대사 경로뿐만 아니라, 조절 경로와 신호 전달 경로 또한 같이 레이아웃 해 줄 수 있는 그래프의 구조 및 레이아웃 알고리즘의 연구가 필요하다. 생물체 내의 생명 활동을 총체적으로 이해하기 위해서는 조절 부분과 신호 전달 부분도 함께 통합적으로 파악해야만 하기 때문에 이 부분은 앞으로의 중요한 연구 과제가 될 것이다.

참 고 문 헌

[1] M. Y. Becker and I. Rojas, "A Graph Layout Algorithm for Drawing Metabolic Pathways," BIOINFORMATICS, Vol. 17, No. 5, pp.461-467, 2001.

[2] Y. Deville, D. Gilbert, J. Helden and S. Wodak, "An Overview of Data Models for the Analysis of Biochemical Pathways," Proceedings of International Workshop on Computational Methods in System Biology, p.174, 2003.

[3] M. Kanehisa, S. Goto, S. Kawashima and A. Nakaya, "The KEGG Databases at GenomeNet," Nucleic Acids Reasearch, Vol. 30, No. 1, pp.42-46, 2002.

[4] P. D. Karp, M. Riley, M. Saier, I. T. Paulsen, J. Collado-Vides, S. M. Paley, A. Pellegrini-Toole, C. Bonavides and S. Gama-Castro, "The EcoCyc Database," Nucleic Acids Reasearch, Vol. 30, No. 1, pp.56-58, 2002.

[5] R. Overbeek, N. Larsen, G. D. Pusch, M. D'Souza, N. Selkov, N. Kyrpides, M. Fonstein, N. Maltsev and E. Selkov, "WIT: Integrated System for High-throughput Genome Sequence Analysis and Metabolic Reconstruction," Nucleic Acids Research, Vol. 28, No. 1, pp.123-125, 2000.

[6] P. D. Karp and S. Paley, "Automated Drawing of Metabolic Pathways," Third International Conference on Bioinformatics and Genome Research, pp.225-238, 1994.

[7] R. Appel, A. Bairoch and D. Hochstrasser, "A New Generation of Information Retrieval Tools for Biologists: The Example of the ExPASy WWW Server," Trends in Biochemical Sciences, Vol. 19, pp.258-260, 1994.

[8] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai and A. -L. Barabasi, "The Large-scale Organization of Metabolic Networks," NATURE, Vol. 407, pp.651-654, 2000.

[9] A. Barabasi and R. Albert, "Emergence of Scaling in Random Networks," SCIENCE, Vol. 286, pp. 509-512, 1999.

[10] K. -I. Goh, B. Kahng and D. Kim, "Universal Behavior of Load Distribution in Scale-Free Networks," Physical Review Letters, Vol. 87, No. 27, 2001.

[11] 정하웅, 강병남, "복잡계의 이해-네트워크의 구조적 성질 및 그 응용", 물리학과 첨단기술 10권 23, 2001.

[12] P. Holme, M. Huss and H. Jeong, "Subnetwork Hierarchies of Biochemical Pathways," BIOINFORMATICS, Vol. 19, No. 4, pp.532-538, 2003.

[13] Douglas B. West, Introduction to Graph Theory, pp.232-245, PRENTICE HALL, 1996.

[14] G. D. Battista, P.Eades, R. Tamassia and I. G. Tollis, Graph Drawing : Algorithms for the visualization of Graphs, PRENTICE HALL, 1999.

[15] M. Kaufmann, D. Wagner(Eds.), Drawing Graphs : Methods an Models, Springer, 1998.

[16] R. Wiese, M. Eiglsperger and P. Schabert, "The Y-files Graph Library: Documentation and Code available at http://www-pr.informatik.uni-tuebingen.de/yfiles," 2000.

이 소 희

2002년 2월 이화여자대학교 생물과학과 &컴퓨터학과 학사. 2004년 2월 이화여자대학교 과학기술대학원 컴퓨터학과 석사 2004년~일본 Impulse Japan사 근무. 관심분야는 생물정보학, 데이타마이닝 등

송 은 하

1996년 2월 가톨릭대학교 수학과 학사 1999년 2월 이화여자대학교 공과대학 컴퓨터학과 석사. 1999년~2001년 한국과학기술연구원(KIST) 연구원. 2002년~현재 이화여자대학교 과학기술대학원 컴퓨터학과 박사과정. 관심분야는 생물정보학, 데이타마이닝, 그래프 드로잉 등

이 상 호
정보과학회논문지 : 소프트웨어 및 응용
제 31 권 제 8 호 참조

박 현 석
서울대학교 전자공학 학사 & 캔사스주립대 전산학 학사. 펜실베니아대(U. of Pennsylvania) 전산정보학 석사. 캠브리지대(U. of Cambridge) 전산학 박사 동경대(U. of Tokyo) post-doctoral fellow. 1989, 캔사스주립대 Golden Key National Honor Society 회원(Honor Roll). 1994년 5월~1994년 12월 펜실베니아대 Research Fellow(미국방성 DARPA 후원). 1997년 10월~1998년 10월 동경대 정보과학과 Research Associate(일본학술진흥회 JSPS 후원) 1998년 12월~1999년 2월 동경대 정보과학과 방문교수 1998년 9월~2000년 2월 성신여자대학교 대우전임강사(정보통신부후원). 2000년 3월~2002년 11월 세종대학교 컴퓨터공학과 조교수(& 바이오인포매틱스 연구소장). 1999년 7월~현재, ㈜마크로젠 이사 & 서울의대 유전자이식연구소 연구위원. 2002년 12월~현재, 이화여자대학교 컴퓨터학과 조교수