유전자 온톨로지와 연계한 단백질 상호작용 네트워크 시각화 시스템

(Protein Interaction Network Visualization System Combined with Gene Ontology)

최 윤 규 † 김 석 †† 이 관 수 ††† 박 진 아 ††††
(YunKyu Choi) (Seok Kim) (Gwan-Su Yi) (Jinah Park)

요 약 단백질 상호작용 네트워크는 어떤 단백질들 간에 상호 작용 관계가 있는지를 네트워크 형태로 나타낸 것이며 단백질 상호작용을 발견하거나 분석하는 것은 생명 공학에서 중요한 연구분야이다. 본 논문에서는 방대한 단백질 상호작용 데이터를 유전자 온톨로지와 연계한 시각화를 통하여 효과적으로 직관을 얻을 수 있는 효율적인 단백질 상호작용 네트워크 분석시스템을 다룬다. 단백질 상호작용 네트워크는 데이터 양이 매우 방대하기 때문에 이를 효율적으로 분석하는 방법과 효과적인 시각화 기법이 요구된다. 본 연구에서는 이를 위하여 동적이고 상호작용 가능한 그래프와 관심 노드와 그 주변 노드를 표시하며 점진적으로 탐색할 수 있는 컨텍스트 기반 탐색 기법을 도입하였다. 이 밖에도 특화된 기능으로써 단백질 상호작용과 유전자 온톨로지 간의 빠르고 자유로운 상호참조 기능과 최소 공통 조상을 사용한 유전자 온톨로지 분석 기능 등을 지원한다. 인터페이스 측면에서는 상호참조 기능을 효과적으로 사용하게 하기 위하여 유전자 온톨로지 그래프와 단백질 상호작용의 시각화 결과를 2차원 윈도우로 나란히 보여주는 인터페이스를 디자인 하였다.

키워드 : 단백질 상호작용, 유전자 온톨로지, 동적 그래프, 컨텍스트 기반 탐색, 최소 공통 조상

Abstract Analyzing protein-protein interactions(PPI) is an important task in bioinformatics as it can help in new drugs' discovery process. However, due to vast amount of PPI data and their complexity, efficient visualization of the data is still remained as a challenging problem. We have developed efficient and effective visualization system that integrates Gene Ontology(GO) and PPI network to provide better insights to scientists. To provide efficient data visualization, we have employed dynamic interactive graph drawing methods and context-based browsing strategy. In addition, quick and flexible cross-reference system between GO and PPI; LCA(Least Common Ancestor) finding for GO; and etc are supported as special features. In terms of interface, our visualization system provides two separate graphical windows side-by-side for GO graphs and PPI network, and also provides cross-reference functions between them.

Key words : Protein-protein interaction, Gene Ontology, Dynamic Graph, Context-based browsing, Least Common Ancestor

· 이 논문은 2008 한국컴퓨터종합학술대회에서 '유전자 온톨로지와 연계한 단백질 상호작용 네트워크 시각화 시스템에 관한 연구'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 한국과학기술원 정보통신공학과
ckyun777@kaist.ac.kr
†† 학생회원 : 한국과학기술원 전산학과
skim0103@kaist.ac.kr
††† 정 회 원 : 한국과학기술원 바이오 및 뇌공학과 교수
gsyi@kaist.ac.kr
†††† 정 회 원 : 한국과학기술원 전산학과 교수
jinahpark@kaist.ac.kr

논문접수 : 2008년 4월 22일
심사완료 : 2009년 1월 23일

정보과학회논문지: 시스템 및 이론 제36권 제2호(2009.4)

1. 서 론

단백질 상호작용(Protein-Protein Interaction: 이하 PPI) Network는 각 단백질을 노드로 단백질 사이의 상호작용 유무를 에지로써 표현한 Network이고, 이를 분석하는 일은 신약 개발 등에 이용될 수 있다. 유전자 온톨로지(Gene Ontology: 이하 GO)는 생물의 유전자의 정보를 나타내기 위한 전산화된 데이터 모델로써, 생물학적 어휘나 개념, 그리고 이들 사이의 관계로 구성된다.

단백질 상호작용을 분석하고 시각화 해주는 시스템에 대한 연구는 국내외에서 꾸준히 연구되고 있으며 국내에서 연구된 시스템으로는 Protenica[1], PIVS[2], CPIN[3], BioINet[4,5]등이 있으며 국외에의 사례로는 Osprey[6], Cytoscape, BiNGO[7]등이 있다. 이러한 시스템들은 각기 특징을 가지고 있으며 일부는 GO와 연계하여 PPI를 군집화 해주고 시각화와 분석 기능을 가지고 있지만, GO와 PPI를 동시에 그래프로 시각화 해서 보여주며 서로 쉽고 빠르게 연계하여 사용 할 수 있는 기능의 지원은 부족하다. 한 예로써 BiNGO는 선택된 단백들들에 연관된 GO그래프 구성 및 시각화가 가능하지만, GO그래프에서 상호참조를 통한 PPI 네트워크의 확장은 불가능하다.

이러한 점들을 보완하여 GO와 PPI를 연계하여 단백질 상호작용에 대한 직관을 얻고 원하는 단백질 또는 단백질 상호작용을 찾는데 도움을 줄 수 있는 시각화 시스템인 PINGO(Protein Interaction Network and Gene Ontology)를 개발하였다. 본 논문에서는 GO와 PPI를 동시에 효율적으로 시각화 하고 연계하여 분석하기 위하여 효율적인 PPI 네트워크, GO의 시각화에 관한 연구와 이들을 연계하여 분석 가능하게 해주는 시스템 구현에 대한 내용을 다룬다.

2. 배 경

2.1 GO 시각화

GO는 유전자 관련 개념들에 대한 전산처리를 가능하게 해주는 분류를 제공한다. 이를 통해 이전까지 통일성 없이 관리되던 생물학 정보들이 GO를 통해 통합 관리되면서 하나의 체계화된 모델을 형성하게 됐으며 자동화된 처리가 용이 하게 되었다. GO는 크게 세 가지 부분으로 나누어지는데, 유전자가 가지고 있는 능력에 대해 다루는 분자 기능(molecular function: MF), 하나 혹은 그 이상의 분자 기능이 특정 순서에 의해 조합되어 달성되는 생물학적 목적에 대해 다루는 생물학적 과정(biological process: BP), 그리고 세포 안에서 유전자 생산물이 활동하는 위치에 대해 다루는 세포 요소(cellular component: CC)가 그것이다. 각 온톨로지의 관계는 'is-a'와 part-of'로 나타내어지며, 형태상으로는 방향성 비순환 그래프(directed acyclic graph: 이하 DAG)로 나타내어진다.

GO분야에서 가장 중요한 문제 중의 하나는 방대한 GO 용어(term)들을 어떻게 효과적으로 사용자들에게 보여주는가 이다. 현재까지 나온 GO 관련 시스템들은 GO를 보여주는 방법을 기준으로 크게 리스트형(list type), 그래프형(graph type), 그리고 혼합형(hybrid type)으로 나눌 수 있다.

eGOn[8], EASE[9], ErmineJ[10]로 대변되는 리스트형은 일반적으로 트리 구조(Tree structure)로 GO 데이터 모델을 표현하고 있으며, 텍스트 기반의 입력을 채용하고 있다. 이러한 툴은 텍스트 기반이라는 특징을 살려, 세세한 항목까지 사용자가 입력 및 수정을 할 수 있어 사용자는 매우 구체적인 사항까지 표현할 수 있다. 그러나 문제는 트리 구조로 GO를 표현하는 데는 한계가 있다. 사실상 GO 데이터 모델은 트리 구조가 아닌 DAG이다. DAG와 트리 구조의 가장 큰 차이는 DAG에서는 자식 노드가 여러 개의 부모 노드를 가질 수 있는 반면, 트리 구조에서는 하나의 자식 노드가 단 하나의 부모 노드만을 가질 수 있다. 즉, 리스트형에서는 하나의 GO 용어가 트리의 여러 군데에 존재할 수 있기 때문에, 하나의 자식 GO 용어의 모든 부모 GO 용어들을 파악하기가 매우 힘들다.

GOLEM[11]로 대변되는 그래프형은 시각적인 면에 치중한 형태로써, 2D DAG 그래프로 GO 데이터 모델을 표현하여 사용자가 보다 쉽게 구조를 파악할 수 있도록 하였다. 이에 더하여 그래프의 노드를 선택할 수 있게 하는 등 직관적인 인터페이스도 함께 제공하고 있다. 그래프형의 단점은 사용자가 원하는 노드를 선택하기 어렵다는 점이다. 즉, GOLEM에서는 프로그램 시작 시 루트(root) 노드만 화면상에 표시되어 있고, 화면상에 나타난 각 노드의 자식 노드들을 추가함으로써 DAG 그래프를 확장할 수 있게 되어있다. 따라서, 만약 사용자가 리프(leaf) 노드를 화면에 보고자 할 경우, 루트에서 리프까지 가는 모든 경우를 일일이 선택해주어야 한다.

마지막인 혼합형은 리스트형과 그래프형의 혼합한 형태를 띠고 있다. 대표적으로 BINGO와 WebGestalt[12]를 들 수 있다. 이들은 리스트형의 트리 구조를 통하여, 사용자가 특정 GO 용어의 계층적 위치를 쉽게 파악할 수 있게 하였을 뿐만 아니라, 검색 기능을 지원하여 더욱 쉽고 빠르게 작업할 수 있도록 하였다. 그리고 그래프를 사용하여 한 GO 용어에 연결된 모든 노드들을 쉽게 파악할 수 있도록 하였으며, 그래프에서 바로 입력 및 수정이 가능하게 하였다. 그러나 이들이 혼합형의 특징을 완전히 표현했다고 하기에는 무리가 있다. BINGO

는 프로그램 시작 시 그래프에 모든 GO 용어들을 표시하고, 리스트에서 선택된 GO 용어들을 하이라이트 시킬 뿐이기 때문이다.

또한, 위에서 언급한 GO 시스템들은 화면에 효과적으로 GO 용어들을 보여주는 것에만 초점을 맞추고 있어, 이들이 이루는 관계의 특징을 파악할 수 있도록 도와주는 것에는 미흡하다. 따라서, 사용자는 화면에 보여지는 GO 용어들 사이의 관계가 어떠한 특징을 갖는지 스스로 찾아야만 한다.

2.2 PPI 시각화

단백질 상호작용은 각각의 단백질을 노드로 단백질 사이의 상호작용의 유무를 에지로 하는 무 방향 네트워크 형태의 구조로 나타낼 수 있다. 이러한 네트워크 구조는 그래프의 일종이며, G=(V, E)로 나타낼 수 있다.

일반적인 네트워크 시각화에 관한 연구로는 [13,14] 등이 있으며 이들은 일반적인 네트워크 시각화에 관련된 개념과 이슈들, 그리고 필요한 요소들을 설명하고 있다. 이들이 다루는 공통적인 이슈 중 하나는 큰 규모의 네트워크를 시각화의 문제점이며 큰 규모의 네트워크 시각화의 문제점들은 다음과 같다.

(1) 네트워크 자체의 복잡함으로 인한 이해 불가
(2) 표시할 화면 공간의 부족
(3) 노드와 에지에 라벨표기의 어려움
(4) 네트워크 레이아웃 및 기타 계산시간의 증가

Schwikowski et al.[15]에서 보여주다시피 PPI 네트워크는 그 노드와 에지의 수가 많고 구조도 복잡함을 알 수 있다. 이러한 복잡한 네트워크는 그래픽적으로 시각화 하여도 이해하기가 난해 할 뿐만 아니라 공간 부족으로 인해 그래프를 이해하는데 가장 중요한 요소 중 하나인 라벨을 표기하기가 힘들어진다. 따라서 이를 해결하기 위한 일반적 해결책들로서 다음과 같은 방법들이 있다.

(1) Zoom / pan (2) Local view / Global view
(3) Filtering (4) Clustering
(5) Grouping (6) Context-based browsing

(1)~(3)은 일반적인 방법이므로 설명을 생략하고, (4)~(6)에 대해 본 논문에서의 관점은 다음과 같다. (4) Clustering은 여러 노드를 하나의 노드로 묶어 표시함으로써 그 표시 개수와 복잡성을 줄이는 방법을 이야기한다. (5) Grouping은 비슷한 속성을 갖는 노드들을 공간적으로 근접한 위치에 또는 유사한 색등으로 그룹화 하여 표시 함으로써 네트워크에 대한 이해도를 높이는 방법을 말한다. 많은 논문들에서 Clustering과 Grouping을 분리하지 않고 모두 Clustering으로 표시하고 있지만 여기서는 혼돈을 줄이기 위해 Clustering과 Grouping으로 나누어서 정의하였다. 마지막으로 (6) Context-based browsing(컨텍스트 기반 탐색)[16]은 전체 네트워크를 모두 표시하는 대신에 사용자가 한 노드를 선택하고 이 노드로부터 이 노드의 이웃 노드로 이동함으로써 표시하는 네트워크요소의 개수와 복잡도를 획기적으로 줄이고 점진적으로 네트워크 구조를 이해해 갈수 있도록 하는 방식을 의미하며 이를 적용한 예로는 Visualthesaurus (http://www.visualthesaurus.com/)가 있다.

PPI 네트워크 구조를 시각화 하는데 고려해야 할 다른 측면의 문제점은 노드들을 어떻게 배치해야 하는가에 대한 문제이다. 이러한 레이아웃에 대한 고려사항은 [13,14] 등에서 다루어지고 있으며, 여러 레이아웃 방법들이 소개되고 있다. 좋은 레이아웃의 기준들은 다음과 같다[13].

(1) Planarity: 교차하는 에지를 줄임
(2) Predictability: 같은 그래프에 대해서 항상 비슷한 형태로 레이아웃이 나옴
(3) Time complexity: 레이아웃을 거의 실시간으로 계산 가능한지
(4) Aesthetic: 좌우 대칭, 적은 에지 교차, 노드의 균일 간격 배치 등 미적인 측면

이러한 레이아웃과 기준들은 절대적인 것이 아니며 레이아웃 방식을 선택할 시 고려할 점은 레이아웃의 특성들이 목적에 얼마나 적합한가이다. Osprey, Cytoscape 등의 프로그램은 기본적으로 Circular, FDP[17] 등 여러 레이아웃을 지원하고 있다. Circular 레이아웃은 빠르게 레이아웃을 계산 가능하지만 대체적으로 Planarity와 Aesthetic 부분이 떨어진다. 단백질 상호작용에 많이 사용되는 FDP(force directed placement)는 그래프가 균형적이고 보기 좋은 형태로 표시되지만 계산시간이 길고 Predictability가 떨어진다는 단점을 갖는다. 또 새로운 노드를 추가 하였을 때 그 형태가 크게 변하는 단점이 있으며 이는 변화를 애니메이션으로 보여줌으로써 완화가 가능하다[14].

2.3 PPI와 GO 연동

GO와 PPI를 연계함으로써 PPI를 개념적으로 분류하고 이해하는 것이 가능해지며, GO를 이용한 PPI의 Filtering, Clustering, Grouping을 함으로써 효과적으로 PPI를 분석하고 이해하는데 도움을 줄 수 있다. 이러한 기능은 Osprey와 Cytoscape plug-in인 BINGO등에서 지원하고 있다. Osprey는 단백질을 표시하는 노드의 색을 GO의 Biological process에 따라 다른 색으로 표시하고 그룹화 하여 표시하는 것을 지원함으로써 PPI와 GO를 연관하여 분석을 가능하게 하였지만 GO를 그래프 형태로 보여주거나 탐색하는 기능은 지원하지 않는다. BINGO는 Cytoscape에 Plug-in 형태로 추가 되며 선택된 단백질들과 관련된 GO 노드들을 시각화해서 보

여주지만 GO를 탐색하고 GO와 PPI을 연동하는 기능 부분은 미약하다. 이에 대해 본 논문에서는 GO와 PPI 모두를 서로 상호참조 가능한 그래프 형태로 시각화 해 주며 효과적으로 탐색가능하고 연동 가능하게 하고자 하였다. 이에 관한 부분은 다음 단락인 구현 항목에서 다룬다.

3. 구 현

3.1 전체 시스템 구조 및 사용 데이터

본 연구에서 개발한 시스템인 PINGO(Protein Interaction Network and Gene Ontology)의 전체적인 구성은 그림 1 및 그림 2와 같다. 그림 1에서와 같이 GO에 관한 부분을 왼쪽에 PPI에 관한 부분을 오른쪽에 배치하였고, GO와 PPI측의 동일/유사한 기능을 좌우로 나란히 배치하여 쉽게 이해하고 사용할 수 있도록 하였다. 시스템에 관한 데모 및 정보는 다음 웹 페이지 (http://cgv.icu.ac.kr/pingo/)에서 제공되고 있다.

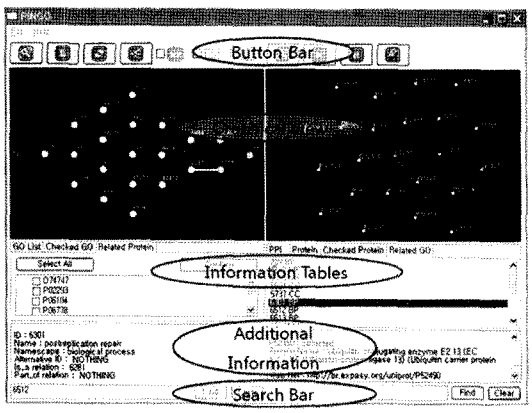

그림 1 PINGO 인터페이스

Graphical View 부분은 2D형태의 레이아웃을 사용하고 있지만 Scene graph기반의 Java3D로 구현되었으며, 이를 통해 3D효과 등을 포함한 여러 효과들이 차후에 추가 가능하게 하였다. 각 부분별 기능은 다음과 같다.

(1) Button Bar: View 조작, 각종 기능 버튼 및 체크 박스
(2) Graphical View: GO와 PPI를 각각 시각화된 그래프 형태로 표시해 주는 부분
(3) Information Tables: GO와 PPI에 관한 Tree List, 또는 테이블 형태의 텍스트 데이터를 표시해 주는 부분
(4) Additional Information: 선택된 노드의 정보와 프로그램 실행상태 관련 정보를 표시해주는 부분
(5) Search Bar: 각각 GO ID와 단백질을 검색해서 찾아주는 부분

시스템 구조는 GO측과 PPI측을 독립적 모듈로 구성하고 상위 GO-PPI 연동 계층에서 인터페이스를 통하여 양측을 모두 사용하는 기능을 수행하게 디자인 하였으며 그 구조는 다음과 같다.

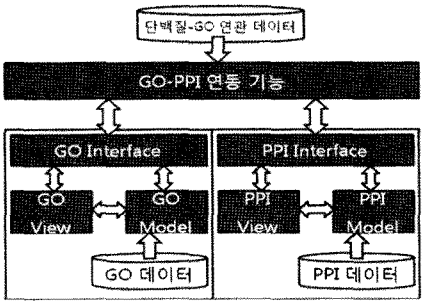

그림 2 PINGO 시스템 구조

테스트용으로 사용한 데이터에 관한 사항은 다음과 같다.

(1) Gene Ontology: OBO v1.2 format, 2008-05-06
(2) Protein Interaction: MINT, 2008-04-08
(3) Protein-GO 상호 참조: unfiltered UniProt GO Annotation @ EBI, 2008-4-11

3.2 사용자 인터페이스

PINGO의 GO 부분과 PPI 부분은 사용자가 쉽게 사용할 수 있도록 비슷한 인터페이스를 사용하고 있다. 즉, 특정 GO 용어 혹은 단백질을 선택하기 위해, 사용자는 다음 세 가지 방법 중 하나를 택할 수 있다.

(1) 검색을 통한 추가
  A. ID를 입력한다.
(2) 리스트에서 선택
  A. 체크박스에 체크한다.
(3) 그래프에서 직접 선택
  A. GO 부분: 노드를 더블 클릭한다.
  B. PPI 부분: 노드를 한번 클릭한다.

이렇게 선택된 결과는 다음과 같은 방식으로 표현된다.

(1) 리스트
  A. 선택된 GO 용어 혹은 단백질의 체크박스에 체크표시가 된다.
(2) 그래프
  A. GO 부분: 루트 GO 용어로부터 선택된 GO 용어까지의 모든 경로를 표시한다.
  B. PPI 부분: 선택된 단백질을 중심으로, 이와 연결된 다른 단백질들을 원형으로 표시한다.

이에 더하여, 선택된 GO 용어 및 단백질의 구체적인 정보를 정보 창(그림 1의 "Additional Information" 참

고)에 따로 표시하여, 사용자의 이해를 높이고 있다.

3.3 GO 그래프 시각화 구현

PINGO의 GO 부분은 앞서 언급한 리스트형, 그래프형, 혼합형 중 혼합형에 속한다. 그러나, 다른 혼합형 툴들과는 달리 리스트와 그래프가 동등한 지위를 가지면서 서로의 상호 작용을 보다 활발히 하는 진보된 형태를 띠고 있다. 즉, 다른 툴들과 마찬가지로 리스트부분에서는 트리 구조를 통해 모든 GO 용어들을 볼 수 있으며, 여기에서 선택된 GO 용어들의 트리 구조만이 그래프에 그려지게 되어 사용자가 더욱 쉽게 이해할 수 있도록 한다. 이처럼 선택된 GO 용어를 중심으로 그래프를 업데이트하므로, BINGO처럼 처음부터 모든 GO 용어들을 다 그릴 필요가 없어 보다 직관적으로 파악할 수 있다.

또한, GO 그래프는 수기야마(Sugiyama) 알고리즘[18]을 통해, 노들간의 교차(cross)를 줄이는 방향으로 레이아웃(layout)를 정하였다. 그리고 각종 버튼을 통해 사용자가 그래프를 좀더 효과적으로 컨트롤할 수 있도록 하였다. 그림 3은 여러 노드들이 순서대로 추가되었을 때, 수기야마 알고리즘에 의해 정렬되는 것을 보여준다.

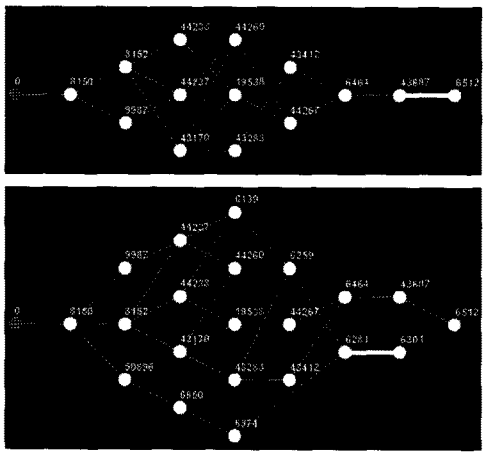

그림 3 노드 추가 시 동적으로 변하는 GO 그래프
(위 측에서 아래 측 그림 순, 'GO:6512' -> 'GO:6301' 순으로 추가 되었다. 숫자는 GO ID, 파란 노드는 루트, 초록 노드들은 그래프에 추가된 노드이며, 노란 노드들은 경로를 나타낸다.)

더 나아가, PINGO는 사용자들이 보다 쉽게 GO 용어들 사이의 관계가 가지는 특징을 유추하고 그래프를 단순화 하여 시각화 할 수 있는 방법을 제공한다. 이 방법이 바로 최소 공통 부모(Least Common Ancestor: 이하 LCA)이다. 최소 공통 부모란, 선택된 모든 GO 용어들의 공통 조상이면서 이들과 가장 가까이 있는 GO 용어를 일컫는다. LCA는 GO 용어들의 공통된 속성을 파악 하는데 매우 큰 도움이 되며, 특히 PINGO의 GO 부분처럼 한정된 GO용어들 사이의 관계 및 특징을 파악하는데 매우 유용하다. 이처럼, PINGO는 리스트와 그래프의 상호 작용 증대와 선택된 GO 용어들에 맞춰진 작업을 제공함으로써 다른 GO툴보다 나은 작업 환경을 제공하고 있다.

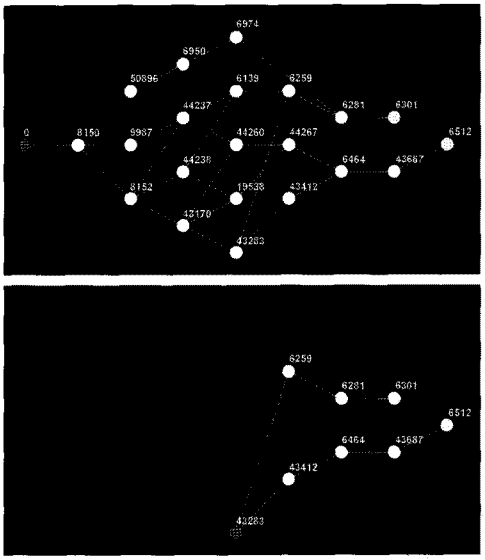

그림 4 LCA 기능을 활용한 GO 그래프 분석
(위의 그림은 기본 GO 그래프이며, 아래 그림은 LCA 기능을 적용한 결과이다. 아래 그림을 보면, 'GO:6301 postreplication repair'과 'GO:6512 Ubiquitin Cycle'의 LCA가 'GO:43283 biopolymer metabolic process'임을 알 수 있고, 이를 통해 이 두 GO 용어가 biopolymer metabolic process에 관여한다는 것을 알 수 있다.)

3.4 PPI 네트워크 시각화 구현

PPI 네트워크는 기본적으로 FDP에 기반하여 시각화하고 있으며 레이아웃 과정을 동적으로 애니메이션 하여 보여주고 있다. PPI의 Graphical View에서 지원하는 주요 기능들은 다음과 같다.

(1) FDP 레이아웃 애니메이션
(2) 컨텍스트 기반 탐색 기능
(3) 노드 위치 고정 기능
(4) 주변 노드 동적 추가 및 제거 기능(2촌 까지 가능)
(5) 연관 GO 상호 참조 기능

기능(1)은 새로운 노드가 추가됨에 따라 변하는 레이아웃을 애니메이션으로 보여줌으로써 새로운 노드가 추가됨에 따라 레이아웃이 크게 변하여도 기존 노드들의

위치가 어떻게 변하였는지 결과적으로 어느 위치에 존재하는지를 이해하는데 도움을 주도록 한 기능이다. 또 FDP 알고리즘은 알고리즘의 반복에 따라 점점 완성된 형태를 가지게 되는데 마지막 반복 단계에서는 레이아웃 전체에서 큰 변화는 없지만 중간에 멈출 경우 부정확한 레이아웃인 상태에서 중단될 수 있으며 너무 많은 반복을 실행하면 최종 레이아웃까지 기다리는 시간이 증가하여 사용성이 떨어진다. 그러므로, PINGO에서는 레이아웃이 실행되는 동안에 이를 애니메이션으로 보여주고 레이아웃 작업 중에도 유저와 동적으로 그래프와 상호작용이 가능하게 하였다.

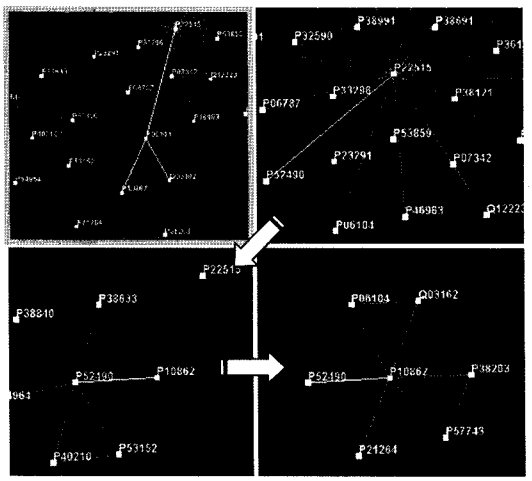

그림 5 'P22515' -> 'P52490' -> 'P10862' 순으로 컨텍스트 기반 탐색 결과(좌상은 세 노드를 모두 표시한 경우의 그래프이며, 우상부터 화살표 순으로 이동하였다. 초록색 노드는 현재 선택된 단백질을 나타내고 노란색 노드는 주변 노드들을 나타낸다.)

복잡하고 큰 네트워크를 탐색하기 위해 보통 Local/Global view를 많이 사용한다 하지만 이러한 탐색방법은 관심 노드의 이웃 노드들이 멀리 떨어져 있을 때 이들을 확인하기 힘들다. 이러한 문제 점을 해결하기 위하여 사용된 (2)컨텍스트 기반 탐색은 관심 노드와 이웃 노드들을 근거리에 표시하며 시작 노드에서부터 주변 노드로 중심을 옮기며 탐색함으로써 점진적으로 네트워크를 이해할 수 있도록 한 기능이다(그림 5 참고). (3)은 (2)의 기능을 구현하여 테스트 하던 중 몇 몇의 관심 노드들의 위치가 크게 변함으로써 전체 연결관계를 이해하기 어려워 지는 문제점을 해결하고자 FDP에 영향을 받지 않도록 노드의 위치를 고정시키는 기능을 추가한 것이다. (4)는 원하는 노드의 2촌 거리에 있는 노드들까지 확장/제거를 해주는 기능이다. (5)의 기능은 선택된 단백질 노드와 연관된 GO들을 보여주고 그 중 원하는 GO노드를 GO Graphical view에 추가 할 수 있도록 해주는 기능으로써 자세한 설명은 다음 3.5절에 기술 하였다.

3.5 PPI - GO 연동 구현

앞에서 언급하였듯이 PPI와 GO를 동시에 시각화 해서 보여주며 서로간에 자유롭게 상호작용이 가능한 시스템은 존재하지 않는 것으로 보인다. PINGO 시스템의 PPI와 GO간의 연동 구현은 PPI측과 GO측 양쪽에서 모드 가능하며, 양측이 서로 빠르게 상호작용 가능하고, 이를 동시에 시각화해서 보여준다. 이러한 연동기능은 PPI 측에서 사용시 선택된 단백질 노드에 관련된 GO ID를 목록 형태로 보여주고, 이 목록에서 원하는 GO ID를 선택함으로써 GO Graphical View에 선택한 GO가 추가되도록 함으로써 관심 단백질의 GO상에서의 분류와 관계를 빠르게 파악할 수 있도록 하였다. 반대로 GO측에서도 마찬가지로 선택한 GO의 관련 단백질 리스트를 보여주며 이 리스트에서 선택함으로써 즉각적으로 PPI측에 단백질 노드가 추가되게 하였다. 예를 들면 그림 3의 GO:6512는 ubiquitin cycle, GO:6301은 post-replication repair BP이며 그림 5의 P22525, P52490, P10862는 각각 ubiquitin E1, E2, E3클래스 단백질이다. 이들은 GO:6512부터 시작하여 GO측과 PPI측의 상호 참조를 통해 오가며 파악 될 수 있다. 이러한 방식으로 사용자는 어떤 GO로부터 또는 PPI 네트워크의 단백질로부터 시작하여 양측을 오가며 의미를 파악할 수 있다.

4. 향후 과제

PINGO 시스템이 보다 유용하기 위해서는 다음과 같은 4가지 영역을 개선하여 발전시키고자 한다. 첫째로 현재 버전의 PINGO 시스템은 GO와 PPI를 연계하여 보여주며 서로의 연관 관계를 목록 형태로 확인 가능하지만 어떤 단백질이 어떤 GO와 연계되어 있는지 그래픽적으로 시각화 하여서 보여주는 기능은 아직 지원되지 않고 있다. 따라서 이러한 부분은 GO에 따른 단백질의 그룹화나 Osprey와 비슷한 방식으로 색을 사용하여 GO-단백질 연관 관계를 표시함으로써 보완이 가능 할 것으로 본다. 두 번째 개선점은 [14]에서도 지적되었듯이 FDP 레이아웃 방법은 동일한 그래프 일지라도 생성시마다 다른 위치와 방향으로 생성됨으로써 Predictability가 매우 떨어진다. 이를 보완하기 위해 적절한 Force Field와 일관된 초기위치 선정 방법에 대한 연구가 필요하다. 세 번째로 개선하고자 하는 부분은 노드의 개수 증가에 따라 FDP 레이아웃 속도가 급격히 떨어질 수 있기 때문에[5] 이를 완화하기 위하여 현재 사용중인 동

적인 레이아웃에 적절한 고속화 알고리즘을 개발 적용하고자 한다. 마지막으로는 그래프의 편집과 노드, 에지의 다양한 선택 방법, 필터링 방법 등 사용자 편의 기능들을 보완하고자 한다.

본 논문에서는 유전자 온톨로지와 단백질 상호 작용 네트워크를 각각 좌 우측에 나란히 배치하고 서로 연계하여 효율적으로 단백질 상호 작용 네트워크를 분석 가능하게 하는 시스템을 제안하였다. 이를 위해 GO측의 시각화는 LCA분석 및 그래프형, 리스트형 시각화를 제공하며 PPI 측에서는 동적 확장 축소 가능 그래프와 노드고정과 결합한 컨텍스트 기반 탐색기능을 제공 한다. GO의 LCA 분석 기능은 선택된 용어들의 공동 분모를 보여 줌으로써 좀 더 효율적인 분석을 가능하게 한다. PPI의 동적 그래프 기능과 컨텍스트 기반 탐색 기능은 큰 네트워크를 효과적으로 시각화 하고 효율적으로 탐색을 가능하게 하는데 의미가 있다. 더 나아가 이러한 두 GO, PPI시각화와 탐색 기능을 연계가 가능하며 이러한 기능들은 단백질 상호작용 네트워크에 대해 효율적으로 지식을 얻는데 도움을 줄 수 있을 것이다.

참 고 문 헌

[1] Hee-Jeong Jin, Ji-Hyun Yoon, and Hwan-Gue Cho, "An Analysis System for Protein-Protein Interaction Data Based on Graph Theory," 한국정보과학회, 2006.

[2] Mi-Kyung Lee and Ki-Bong Kim, "A Visualization and Inference System for Protein-Protein Interaction," 한국정보과학회, 2004.

[3] Dong-Soo Han, Suk-Hoon Jung, Woo-Hyuk Jang, and Choon-Ho Lee, "Constraints Based Dynamic Protein Interaction Network," 한국정보과학회, 2005.

[4] http://www.meta-biz.net/html/product03.htm

[5] SunLee Bang, JaeHun Choi, JongMin Park, Soo Jun Park, "단백질 상호작용 네트워크의 개념 분류 레이아웃", 한국정보과학회 학술발표논문집, pp. 61-63 (3 pages), 2006.

[6] Bobby-Joe Breitkreutz, Chris Stark, Mike Tyers. "Osprey: a network visualization system," Genome Biology, 2003.

[7] Steven Maere, Karel Heymans and Martin Kuiper, "BiNGO: a Cytoscape plugin to assess overrepresentation of Gene Ontology categories in Biological Networks," Bioinformatics, 2005.

[8] Vidar Beisvag et al., "GeneTools - application for functional annotation and statistical hypothesis testing," BMC Bioinformatics, Vol.7, pp.470, 2006.

[9] Glynn Dennis Jr et al., "DAVID: Database for Annotation, Visualization, and Integrated Discovery," Genome Biology, Vol.4, Num.5, pp.3, 2003.

[10] Homin K Lee, William Braynen, Kiran Keshav and Paul Pavlidis, "ErmineJ: Tool for functional analysis of gene expression data sets," BMC Bioinformatics, Vol.6, pp.269, 2005.

[11] Rachel SG Sealfon, Matthew A Hibbs, Curtis Huttenhower, Chad L Myers and Olga G Troyanskaya, "GOLEM: an interactive graph-based gene-ontology navigation and analysis tool," BMC Bioinformatics, Vol.7, pp.443, 2006.

[12] Bing Zhang, Stefan Kirov and Jay Snoddy, "WebGestalt: an integrated system for exploring gene sets in various biological contexts," Nucleic Acis Research, Vol.33, pp. 741-748, 2005.

[13] Aaron Kershenbaum, Keitha Murray, "Visualization of network structures," Journal of Computing Sciences in Colleges, Volume 21, Issue 2, Pages: 59-71, 2005.

[14] Ivan Herman, Guy Melançon, M. Scott Marshall, "Graph Visualization and Navigation in Information Visualization: A Survey," IEEE Transactions on Visualization and Computer Graphics, Vol.6, No.1, pp. 24-43, January 2000.

[15] Benno Schwikowski, Peter Uetz, Stanley Fields, "A network of protein-protein interactions in yeast," Nature Biotechnology 18, 1257-1261, 2000.

[16] Yannis Tzitzikas, JeanLuc Hainaut, "On the Visualization of Large sized Ontologies," AVI, pp. 99-102, 2006.

[17] P. Eades, "A heuristic for graph drawing," Congressus Numerantium, Vol. 42, pp. 149-160, 1984.

[18] K Sugiyama, S Tagawa, M Toda, "Methods for Visual Understanding of Hierarchical System Structures," IEEE Transactions on Systems, Man and Cybernetics 11(2):109-125, 1981.

최 윤 규

2005년 건국대학교 소프트웨어학과(학사) 2006년~2009년 한국정보통신대학교 공학부 석사과정. 2009년~현재 한국과학기술원 정보통신공학과 석사과정. 관심분야는 컴퓨터 그래픽스, 정보 시각화, 컴퓨터 구조

김 석

2008년 한국정보통신대학교(학사). 2008년~2009년 한국정보통신대학교 공학부 석사과정. 2009년~현재 한국과학기술원 전산학과 석사과정. 관심분야는 physics-based virtual deformation, haptics

이 관 수

1988년 서울대학교 동물학(학사). 1990년 한국과학기술원 생물공학(석사). 1993년 한국과학기술원 생물공학(박사). 1993년~1994년 한국생명공학연구원 단백질 공학 그룹, 연구원. 1996년~1999년 한국기초과학지원연구원 자기공명그룹, 연구원. 1996년~1999년 미국 노스캐롤라이나 대학교, 생화학 및 생물물리학과, 연구원. 1999년~2001년 캐나다 토론토 대학교, 의생물물리학과 연구원. 2001년~2002년 캐나다 Affinium Pharmaceuticals(Integrative Proteomics), Inc. 책임 연구원. 2002년~2009년 한국정보통신대학교 부교수. 2009년~현재 한국과학기술원 부교수. 관심분야는 Bioinformatics, Computational systems biology, Synthetic biology, Structural bioinformatics, Medical informatics.

박 진 아

1988년 미국 콜럼비아대학교(학사). 1989년 미국 IBM Thomas J. Watson 연구센터 연구원. 1991년 미국 펜실바니아대학교(석사). 1996년 미국 펜실바니아대학교(박사). 1996년~1998년 미국 펜실바니아대학교 박사후과정. 1999년~2002년 한국과학기술원 대우교수 및 초빙교수. 2002년~2009년 한국정보통신대학교 조교수 및 부교수. 2009년~현재 한국과학기술원 부교수. 관심분야는 컴퓨터 그래픽스, 가변형 모델, 의료영상 데이터 가시화, 햅틱 렌더링, 정보가시화