IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 3, Ver. VI (May-Jun. 2014), PP 45-55
www.iosrjournals.org

Algorithms for Various Biological Networks

BhartiTalukdar

Abstract: *Biological networks are the networks which are used to represent different biological entities and relationship between the different entities. But due to the ongoing growth of knowledge in the life science their size and complexity is steadily increasing. For understanding biological networks several algorithms for lying out and graphically representing networks and network analysis results have been developed. However, current algorithms are specialized to particular layout styles and therefore various algorithms are required for representing different types of networks. This paper present a novel algorithm to visualize different biological networks and network analysis results in meaningful ways depending on network types and analysis outcome.*

I. Background

Networks play a crucial role in biological analysis of organisms. They are used to represent processes existing in biological systems and to represent interactions and dependencies between biological entities such as genes, transcripts, proteins and metabolites. One large application area for network-centered analysis and visualization is Systems Biology, an increasingly important research field which aims at a comprehensive understanding and remodeling of the processes in living beings [1,2]. Due to the steady growth of knowledge in the life sciences such networks are increasingly large and complex. To tackle this complexity and help in analyzing and interpreting the complicated web of interactions meaningful visualizations of biological networks are crucial.

Since last few years methods for automatic network visualization have gained increased attention from the research community over recent years and various layout algorithms have been developed, e. g. [3-11]. Often standard layout methods such as force directed [12,13], layered [14,15] and circular [16] approaches are used to draw these networks. However, the direct use of standard layout methods is somewhat unsatisfactory since biological networks often have specialized layout requirements reflecting the drawing conventions historically used in manually laid out diagrams (which have been developed to better emphasize relevant biological relationships and concepts). This has led to the development of network- and application-specific layout algorithms, for example, for signal transduction maps [17,18], protein interaction networks [3,6], metabolic pathways [4,10,19] and protein-domain interaction networks [20]. Advanced solutions combine different layout styles (such as linear, circular and branching layouts) for sub-networks or use specific layouts styles for particular network parts such as cycles [7,10,21].

However, current approaches for the automatic visualization of biological networks have four major drawbacks resulting from the specialized nature of these algorithms:

1. Different kinds of biological networks (e. g. protein interaction or metabolic networks) have different layout conventions and this requires the implementation and sometimes development of specialized layout algorithms for each convention.
2. It is not easy to combine networks with different layout conventions in the one drawing since the layout algorithms use quite different approaches and so cannot be easily combined.
3. The user cannot tailor the standard layout algorithms for their particular need or task by e. g. emphasizing the pathways of interest by making them straight.
4. The algorithms do not sufficiently support interactive network exploration. Usually with these algorithms small modifications in the network structure and re-layout of the network results in very different pictures.

However, such sudden and large changes destroy the user's mental map (i. e. the user's understanding of the network based on the previous view) and therefore hinder interactive understanding of the network.

Here I present a new algorithm for layout of biological networks that overcomes these limitations. It is based on a powerful new graph drawing technique, constrained graph layout [22]. Like force-directed layout [12,13] constrained graph layout works by minimizing an objective function that measures the quality of the layout. However it extends force-directed layout by allowing minimization of the objective to be done subject to placement constraints on the objects in the network. This is achieved by using mathematically rigorous optimization techniques based on gradient projection [23]. Efficient implementation is made possible by restricting the placement constraints to be separation constraints of the form $u + g \leq (=) v$, enforcing a minimum (or precise) gap g between the positions u and v of pairs of objects in either the x or y dimensions of the drawing.

The presented approach provides a generic, universal algorithm for layout of biological networks:
1. It greatly simplifies the implementation of layout methods for life sciences, systems and synthetic biology tools, which have previously had to utilize very different layout algorithms for different types of biological networks (or different layout requirements).
2. It allows the use of different layout styles for different parts of one large network.
3. It allows the user to customize the layout by adding separation constraints.
4. It lends itself to mental-map-preserving dynamic layout in interactive systems, thereby supporting interactive exploration of large and complex networks.

Introduction

A network is defined as a set of elements called vertices or nodes having connections among them called edges. Internet, the world wide web, Social networks(connection among individuals),networks of business relations, neural networks, food webs are examples of network.
The study of networks in the form of mathematical graph theory ,is one of the fundamental pillars of discrete mathematics .Euler's celebrated 1735 solution of the Konisberg bridge problem is cited as the first true proof in theory of netwoks.

Types of Networks

There are many ways of categorizing the network. Such as a network can have more than one type of different vertex or more than one different type of edge .If we take the example of social network of people, vertices may be men or women. People of different nationalities ,locations ,ages ,incomeset .Edges may represent friendship, animosity or geographical proximity.

They can carry weights ,representing how well two people know each other.They can also be directed ,pointing in only one direction .Graphs composed of directed edges are themselves called directed graphs or sometimes digraphs.

A graph representing telephone calls or email messages between individuals would be directed, Since each message goes in only one direction .Directed graphs can be cyclic or acyclic.
One can also have hyperedges-edges that join more than two vertices together. Graphs containing such edges are called hypergraphs .for example in social network-n individuals connect to each other by virtue of belonging to the same family can be represented by n-edge joining them.

Glossary of terms

Vertices- The fundamental unit of a network also called a site(physics), a node (Computer Science),or an actor(Sociology).
Edge-The line connecting two vertices . Also called a bond(physics),a link(Computer Science) or a tie(Sociology).

Directed/Undirected-An edge is directed if it runs in only one direction and undirected if it runs in both directions.

Degree-The number of edges connected to a vertex .A directed graph has both an in-degree and an out-degree for each vertex ,which are the numbers of incoming and outgoing edges.
Component-The component to which a vertex belongs is that set of vertices that can be reached from.

In a directed graph a vertex has both an in-component(set of vertices from which the vertex can be reached) and out-component(set of vertices which can be reached from it).Geodesic paths-Shortest path through the network from one vertex to another.Diameter-Length (number of edges) of the longest geodesic path between any two vertices.Social Network-

A Social network is a social structure made up of a set of social actors (such as individuals or organizations) and a set of dyadic ties between these actors. The social network perspective provides a set of methods for analyzing the structure of whole social entities as well as a variety of theories explaining the patterns observed in these structures .the study of these structures uses social network analysis to identify local and global patterns, locate influential entities and examine network dynamics.A social network is a set of people or groups of people with some pattern of contacts or interactions between them. The patterns of friendships between individuals, business relationships between companies, and intermarriages between families.

Information networks

Information networks sometimes called as knowledge networks. The classic example of an information network is the network of citations between academic papers. These citations form a network in which the vertices are articles and a directed edge from article A to article B indicates that A cites B. Citation networks are acyclic because papers can only cite other papers that have already been written, not those that have to be written.

Technological Networks

The man-made networks designed typically for distribution of resources such as electricity or Information for example electric power grid or Internet or telephone network.

Biological Networks

Biological processes are often represented in the form of networks such as protein-proteininteraction networks and metabolic pathways

II. Basic Network features

The Small World Effect

A node's degree or connectivity ,giving the number of links k the node has ,is the most elementary network measure. For example in following fig. nodes and I j have exactly three links(k=3).The overall graph is characterized by average degree <k>,which has the value <k>=2.6.

As in most networks, there are multiple paths between any two nodes i and j .A useful distance measure is the length of the shortest path $l_{ij.}$ The mean path length defined as

$$<l>= \quad 2/N(N-1)\sum_{I=1}^{N} l_{ij}$$

number of steps is often referred to display the 'small world' property .first illustrated on social networks ,indicating that two randomly chosen individuals can be connected by only six intermediate acquaintances.

Transitivity or Clustering

In many networks it is found that if it is found that if vertex A is connected to vertex and vertex B to vertex C,t hen there is a probability that vertex A will also be connected to vertex C.

In terms of network topology, transitivity means the presence of a number of triangles in the network sets of three vertices each of which is connected to each of which is connected to each of the others .It can be quantified by defining a clustering coefficient C thus:

$$C= \frac{3* \text{number of triangles in the network}}{\text{number of connected triples of vertices}}$$

Where a "Connected triple" means a single vertex with edges running to an unordered pair of others.

In effect, C measures the fraction of triples that have their third edge filled in to complete the triangle. The factor of three in the numerator accounts for the fact each triangle contributes to three triples and ensures that C lies in the range 0<=C<=1.In simple terms, C is the mean probability that two vertices that are network neighbors of the same other vertex will themselves be neighbors. It can also be written in the form.

C=6*number of triangles in the network/number of paths of length two.

Where a path of length two refers two refers to a directed path starting form a specified vertex.

Degree distributions

The degree of a vertex in a network is the number of edges incident on (i.e. connected) to that vertex. We define Pk to be the fraction of vertices in the network that have degree k .equivalently , Pk is the probability that a vertex chosen uniformly at random has degree k. If a network is directed, meaning that edges point in one direction from one node to another node, then nodes have two different degrees, the in-degree which is the number of incoming edges, and the out – degree which is the number of outgoing edges.

The degree distribution p(k) of a network is then defined to b the fraction of nodes in the network with degree k, Thus if there are n nodes in total in a network and nk of them have degree k ,we have p(k)=n k/n.

The degree distribution is very important in studying both real networks, such as the Internet and social networks , and theoretical networks .The simplest network model ,for example, the(Bernoulli) random graph ,in which each of n nodes is connected (or not) with independent probability p(or 1-p) , has a binomial distribution of degree k.(or Poisson in the limit of large n).Most networks in the real world ,however have degree distributions very different from this. Most are highly right skewed , meaning that a large majority of nodes have low degree but a small number ,known as "hubs" have high degree.

Biological Networks

Biological processes are often represented in the form of networks such as protein-proteininteraction networks and metabolic pathways. The study of biological networks , their modeling,analysis, and visualization are important tasks in life science today. An understandingof these networks is essential to make biological sense of much of the complex data that isnow being generated. This increasing importance of biological networks is also evidencedby the rapid increase in publications about network-related topics and the growing

numberof research groups dealing with this area. Most biological networks are still far from being complete and they are usually difficult to interpret due to the complexity of the relationshipsand the peculiarities of the data. Network visualization is a fundamental method that helpsscientists in understanding biological networks and in uncovering important properties ofthe underlying biochemical processes. This chapter therefore deals with major biologicalnetworks, their visualization requirements and useful layout methods. We start with somebasic biology and important biological networks

Molecular Biological FoundationsA cell consists of many different (bio-) chemical compounds. A crucial macromolecule inorganisms is DNA (deoxyribonucleic acid), which is the carrier of genetic information. ButDNA itself is not able to provide the structure of a cell, to act as a catalyst for chemicalreactions or to sense changes in the cell's environment. Such functions are carried out byproteins, large molecules which are built according to information stored in DNA sequences.

The central dogma of molecular biology deals with the information transfer from DNA toproteins. It states that proteins do not code for the production of other proteins, DNAor RNA (ribonucleic acid), i.e., that information cannot be transferred from one proteinto another protein directly or from a protein back to nucleic acid. Instead, the standardpathway of information flow is from DNA to RNA to protein. Genes represented by DNAsequences are transcribed into RNA sequences which are then translated into proteins, seeFigure 20.1. These proteins have different types such as structural components (whichgive cells their shape and help them move), transport proteins (which carry substancessuch as oxygen), enzymes (which catalyze most chemical processes in cells and help changemetabolites into each other) and regulatory proteins (which regulate the expression of othergenes). Crick summarized the standard pathway of information flow as "DNA makes RNA,RNA makes protein and proteins make us" [Kel00].

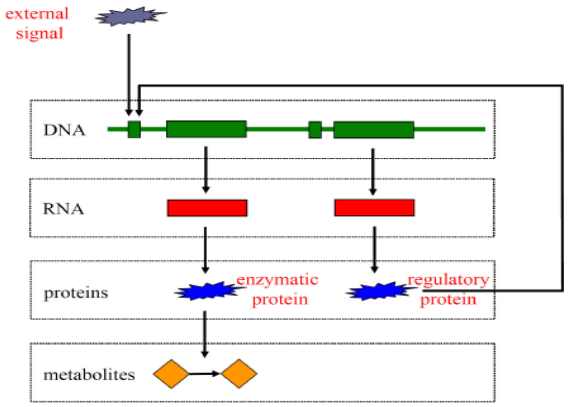


Fig:The Standard Pathway of Information Flow

Signal Transduction and Gene Regulatory Networks

A key issue in biology is the response of a cell to internal and external stimuli and thesubsequent regulation of its genetic activity. Signal transduction and gene regulatory pathwaysand networks describe processes to coordinate the cell's response to such stimuli. Herewe consider both networks together as the underlying mechanisms have many similarities,the networks share some common elements and both often result in the regulation of geneexpression. Consequently, similar visualization approaches are used for signal transductionand gene regulatory pathways and networks.

Definition

Signal transduction is a communication process within a cell to coordinate its responses toan environmental change. The stimulus comes from the cell's environment, e.g., moleculessuch as hormones. The response is a reaction of the cell, e.g., the activation of a gene orthe production of energy. A signal transduction pathway is a directed network of chemicalreactions in a cell from a stimulus (an external molecule which binds to a receptor on thecell membrane) to the response (e.g., the activation of a gene). Here we focus on signaltransduction pathways that aim at transcription factors and thus alter the expression ofgenes in a cell. The signal transduction network of a cell is the complete network of allsignal transduction pathways. A signaling cascade is a process where signal transductioninvolves an increasing number of molecules in the steps from the stimulus to the response.

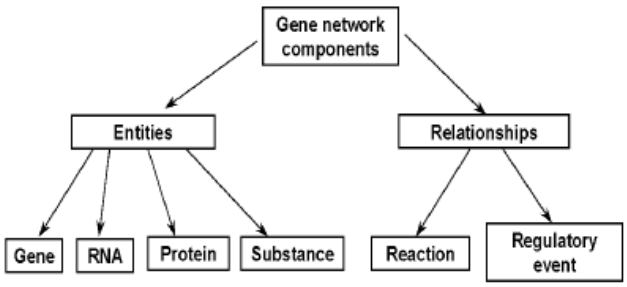
Gene regulation is a general term for cellular control of the synthesis of proteins at thetranscription step. Gene regulation can also be seen as the response of a cell to an internalstimulus. Often one gene is regulated by another gene via the corresponding protein (calledtranscription factor), thus gene regulation is coordinated in a gene regulatory network. Thisnetwork directs the level of expression for each gene in the cell by controlling whether andhow often that gene will be transcribed into RNA. Similar to signaling cascades in

signaltransduction networks a gene can activate more genes in turn and an initial stimulus cantrigger the expression of large sets of genes.

As mentioned above we study signal transduction and gene regulation together. Figure20.1 sketches both processes with signal transduction going from an external signal viaseveral steps to the activation of a gene as one possible response and gene regulation goingfrom a gene via a protein to another gene.

Events of signal transduction and gene regulatory processes occur in different parts of acell (cellular compartments). To represent compartments these networks can be modeled asclustered graphs. A clustered graph C = (G, T) consists of a directed graph G = (V,E) anda rooted tree T, such that the leaves of T are exactly the nodes of G. The nodes v ∈V ofthe graph are chemical and biochemical compounds (ranging from ions, to small molecules,macromolecules and genes) and the edges e ∈E are biochemical events (e.g., binding, transportationand reaction). The occurrence of signal transduction and gene regulatory eventsin different cellular compartments can be modeled be the tree T. Each node t ∈T representsa cluster of nodes of G consisting of the leaves of the sub tree rooted at t. The modelingof such networks based on clustered graphs can be used for cluster-preserving layout algorithms

[EH00]. However, as it is only partly known in which compartment an event occurs,signal transduction and gene regulatory processes are usually modeled by graphs. The pathwaysand networks can be derived from databases such as KEGG [KGKN02, KGH+06] and



Gene Regulatory Network

The entities are subdivided into 4 classes: 1)Protein or protein complex; 2) Gene; 3) RNA; 4) Nonproteinaceous Substance. Instances of eachclass are described in a separate table in the GeneNet database The components of a genenetwork are scattered throughout cellcompartments, cells

Two types of relationships between the entitiesare considered: Reaction, that is, formation of anew entity or acquisition of a new property bythe entity, and Regulatory event, that is, theeffect of an entity onto a certain reaction.

Protein Protein Interaction

While traditional biochemical experiments had generated asmall set of data for individual protein–protein interactions[34],the last three years have seen a rapid expansion of protein interactiondata due to the recent development of high-throughputinteraction detection methods such as yeast two-hybrid (Itoet al., 2000) and mass spectrometry techniques. The interactiondata is available either in text files or in databases. However, due to the volume of data, a graphical representationof protein interactions has proven to be much easier tounderstand than a long list of interacting proteins. Furthermore,a network of protein interactions provides us with aclear notion of protein function by showing a context withinwhich function can be interpreted.

Protein–protein interactions are typically visualized as anundirected graph G = (V ,E), where x, y ∈V represent

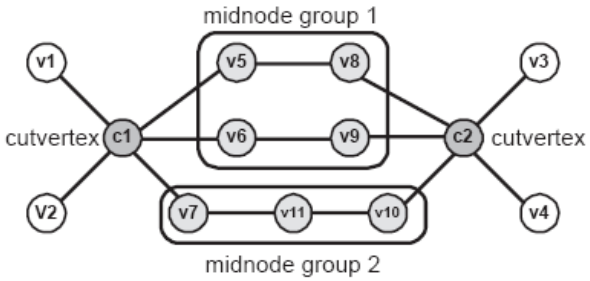
∗To whom correspondence should be addressedproteins and (x, y) ∈E represents an interaction betweenproteins x and y. Visualization of a graph is straightforwardwhen dealing with a small number of nodes and edges. Inpractice, protein–protein interaction networks often consistof thousands of nodes or more, which severely limit theusefulness of many graph drawing tools either because theyproduce cluttered drawings with many edge crossings or staticdrawings that are not easy to modify, they are too slowfor interactive analysis with large data sets, or because theyrequire input data to be in specific format rather than taking thedata directly from protein–protein interaction databases. Theultimate usefulness of a protein interaction network dependson the readability of the network, and therefore, a proteininteraction network should focus on conveying the interactioninformation quickly and clearly.

Force-directed layout algorithms have been the mostpopular methods for visualizing an undirected graph, whichproduce an optimal layout based on a force model. A simpleimplementation of a force-directed algorithm encounters realdifficulties when drawing graphs of more than a few hundrednodes. These difficulties originate from two sources. First,layout adjustment involves computation of force betweenevery pair of nodes at each step of the optimization process.Second, for large graphs the optimization process needs toomany iterations for transforming the initial random layout intoan optimal layout.Previously we developed a force-directed layout

program called InterViewer (Juet al., 2003). In this paper, I present a new program that efficiently produces a protein interaction network of good quality without computingforce between every pair of nodes. This improveson InterViewer in many ways: (1) while Interviewer produces

a drawing by computing force between every pair of nodesin each iteration of the optimization process, This produces a more pleasant drawing without computing forcebetween every pair of nodes, (2) This is faster than InterViewer, (3) This provides several abstractionoperations to reduce complex networks into simpler ones and(4) multiple protein interaction networks can be comparedfor common proteins and their interactions shared by all orpart of the networks.

Algorithm for protein interaction networks



Mid nodes (v5–v11) on three paths between a pair of enclosing Cutvertices (c1 and c2). Since the multiple paths have different lengths, mid nodes on the paths are grouped into two groups: mid node group 1 = {v5, v6, v8, v9}, mid node group 2 = {v7, v10, v11}.

III. Definitions

The degree of a node v is the number of its edges and is denotedby deg(v). A cutvertex (also called an articulation point) in

a graph G is a node whose removal disconnects G. A path in a graph G is a sequence (v1, v2, . . . ,vn) of distinct nodes

ofG, such that (vi , vi+1) ∈E for $1 \leq i \leq n - 1$. A graph$G_ = (V_, E_)$, such that $V\_ \subseteq V$ and $E\_ \subseteq E \cap (V\_ \times V\_)$, is

asubgraphof graph G = (V , E).

When multiple paths exist between a pair of cutvertices, wecall the nodes on the paths mid nodes. In Figure there are mid nodes (shown in yellow) on three paths between a pair of enclosing cut vertices (shown in blue). If the multiple pathsbetween a pair of cutvertices have different lengths, mid nodes on the paths of same length are grouped together.

What we call pivot nodes are the key nodes in the layout of agraph. In order to produce a layout of high quality efficiently,we select pivot nodes that are almost uniformly distributedin each connected component (see Fig. 2 for examples). Thenumber of pivot nodes and distance between them are determinedbased on the number of nodes and edges, and the diameterof a connected component (a diameter of a connected componentis the maximum distance between two nodes in thecomponent). In general, more pivot nodes are selected for aconnected component with a large diameter compared to thenumber of nodes than for a connected component with a smalldiameter compared to the number of nodes. For a small connectedcomponent with 100 nodes or fewer, we select morepivot nodes so that the distance between them may be 3 orless. However, each connected component can have at most100 pivot nodes in any case for the efficiency of the algorithm.

A detailed method for selecting pivot nodes and for computingthe distance between them is described in Algorithms 2and 3 later.

IV. The Algorithm

A common problem with many force-directed layoutalgorithms is that they become very slow when dealing with large graphs because layout adjustment at each step typicallyinvolves computation of force between every pair of nodes.Since a protein interaction network tends to be a disconnectedgraph with several connected components, we first computea layout of connected components and then compute a layoutof nodes within a connected component. Our experience is that this approach produces much better drawings in a shortertime than computing a layout of all nodes from the beginning.Our algorithm uses a multilevel technique to draw a graph.It is composed of two steps at the top level: grouping and layout.

In the grouping step, the algorithm first groups nodes ofa disconnected graph into connected components, and finds mid nodes and pivot nodes in each connected component. Inthe layout step, the coarsest graph is an initial layout of connectedcomponents based on their pivot nodes only. The layoutof each connected

component is then refined locally within theconnected component based on its mid nodes and neighbors of each node. Each step of the algorithm can be summarizedas follows.

1. Grouping

(a) Identify all connected components of an entirenetwork.

(b) For each connected component, determine its mid nodes and pivot nodes.

(c) Compute the distance of every node from the pivotnodes of the connected component to which the nodebelongs.

2. Layout

(a) Find a layout of connected components of an entirenetwork (layout between connected components).

(b) For each connected component find a layout ofnodes with respect to the pivot nodes of the connectedcomponent (global layout within a connectedcomponent).

(c) Refine the layout of each connected component by relocating the mid nodes adjacent to cut vertices with respect to the cut vertices and the cut vertices' direct neighbors (local layout of mid nodes within connected component).

(d) Refine the layout of each connected component byrelocating all nodes with respect to their neighborswithin distance 2 (local layout of all nodes within aconnected component).

Step 1(a) is straightforward, and Algorithm 1 describesstep 1(b). In Algorithm 1, a group represents a connectedcomponent. Since step 1(a) and Algorithm 1 are performed onnodes with at least one edge, nodes with no edge are positionedafter the connected components of size ≥2 are positioned instep 2(a). For a graph with $|V| = n$ nodes, the time complexity

of step 1(a) is O(n), and the time complexity of Algorithm 1

Fig. 2.(**a**) Pivot nodes (shown in green) selected from a mesh. (**b**) Pivot nodes (shown in green) selected from a protein interaction network.

Algorithm 1 Distance(v ,w)

1: DLast.Add(v, 0)
{Add v and its distance (= 0) from v to DLast}
2: DLast.First {Get the first node of DLst}
3: **repeat**
4: DLast.GetCurrent(v_, currentDist)
{Get the current node v_ and its distance from v}
5: **for all** neighbor u of v_ **do**
6: **if** u _∈DLast**then**
7: **if** w = u **then**
8: return currentDist+1{distance between v and u}
9: **end if**
10: DLast.Add(u, currentDist+1)
{Add u and its distance from v to DLast}
11: **end if**
12: **end for**
13: DLast.Next {Get the next node of DLast}
14: **until** DLst.Eof {until no more nodes exist in DLast}Selecting pivot nodes from each connected component instep 1(c) is done by Algorithms 2 and 3. When selectingpivot nodes, distances of the pivot nodes from all other nodesare also computed. Algorithms 2 and 3 take O(n) time for asingle pivot node, and therefore, the total time complexity forselecting all pivot nodes is $O(|PvN| \cdot n)$. Algorithm 3 examineswhether the current node v is already a pivot node; ifnot, it determines the possibility of including the node to the pivot node set PvN depending on the distance from existingpivot nodes, the structure of the connected component (i.e.diameter, number of nodes and edges of the connected component).

The current node v can be selected as a pivot node if

Algorithm 2 SelectPivotNodes

1: MaxDist←1
2: PvN.Add(V[0], DistTable.Create(V[0], 0))
{first node in a group}
3: PvN.First {Get the first node of PvN}
4: **repeat**
5: DLast.Clear {Initialize DLst as an empty list}
6: DLast.Add(PvN.CurrentPivotNode, 0)
{Add the current pivot node and its distance}
7: DLast.First {Get the first node of DLast}
8: **repeat** {distance from pivot nodes}

9: ChkDistance(DLast, PvN.CurrentDistTable,
MaxDist)
10: DLast.Next {Get the next node of DLast}
11: **until** DLast.Eof {until no more nodes exist in DLast}
12: PvN.Next {Get the next node of PvN}
13: **until** PvN.Eof {until no more nodes exist in PvN}
it satisfies the following rules (function ChkPvN(v)in step 16
of Algorithm 3).
1. In a connected component with <40 nodes, the distanceof v from all existing pivot nodes should be at least 2.
2. In a connected component with ≥40 and <100 nodes,the distance of v from all existing pivot nodes should be at least 3.
3. In a connected component with ≥100 nodes,
(a) if the diameter (d) of the connected component is<7,
degree(v) should be ≥3.
(b) if 7 ≤ d <15, degree(v) should be ≥4.
(c) if 15 ≤ d <20, degree(v) should be ≥5.

Algorithm for protein interaction networks
Algorithm 3 ChkDistance(DLast, DistTable, MaxDist)
1: DLast.GetCurrent(v, dist)
{Get a node v and its distance from a pivot node}
2: **if** (dist>MaxDist) **then**
3: MaxDist←dist {Update the maximum distance}
4: **end if**
5: bAddPvN←true {potential pivot node}
6: **for all** neighbor w of v **do**
7: **if** w ∈DLst**then** {distance of w from a pivot node ha snot been determined.}
8: bAddPvN←false {w cannot be a pivot node}
9: DLast.Add(w, dist+1)
{Add w and its distance from v to DLast}
10: DistTable(w)←dist+1{Store the distance of w from v in DistTable}
11: **end if**
12: **end for**
13: **if** MaxDist/3 = dist **then** {The node is at a distance of one third of the maximum distance}
14: bAddPvN←true {potential pivot node}
15: **end if**
16: **if** bAddPvN and ChkPvN(v) **then**
17: PvN.Add(v, DistTable'.Create(v, 0))
18: **end if**
(d) else, letR be the ratio of the diameter of the connectedcomponent to the number of nodes of the connectedcomponent.
(i) ifR <0.01, the distance of v from all existingpivot nodes should be at least 40.
(ii) if 0.01 ≤ R <0.02, the distance of v from allexisting pivot nodes should be at least 17. If the total number of nodes>1000, adjust the distanceto 30.
(iii) if 0.02 ≤ R <0.035, the distance of v from allexisting pivot nodes should be at least 13. If the total number of nodes>1000, adjust the distanceto 20.
(iv) if 0.035 ≤ R <0.07, the distance of v from allexisting pivot nodes should be at least 10.
(v) ifR ≥ 0.07, the distance of v from all existingpivot nodes should be at least 5.

Algorithm 4 provides a concise description of all layoutsof step 2, including both global layout and local layout. Theposition of v is always determined with respect to a referenceset V _, which is a subset of V . In step 2(a), the referenceset V _ is a set of pivot nodes of other connected components,to which v does not belong. The maximum diameter of allconnected components is used as the value of Distance(u, v)in step 4 of Algorithm 4, and therefore is constant for allnodes. In step 2(b), the reference set V _ is a set of pivot
Algorithm 4 Layout(v, V _)
1: D ← 0 {Initialize the position displacement D to 0}
2: **for all** u ∈V _ **do** {V _: subset of V }
3: _ ← pos[u] − pos[v] {pos[u]: position of node u}
4: D ← D + _(1 − Distance(u, v)/_)

{_: norm of a vector _}
5: **end for**
6: D ← D/|V _| {|V _|: number of nodes in V _}
7: pos[v] ← pos[v] + D
{Update the position of v by adding D.}nodes of the connected component to which v belongs. The
value of Distance(u, v) is available in the distance table, whichwas already computed by Algorithm 3 for each pivot node.
Steps 2(b) and 2(c) are repeated until the maximum edgelength of the connected component ≤ a threshold value.
In step 2(c), the reference set V _ of v is a set of its enclosingcutvertices and the cutvertices' direct neighbors, and v is a
Mid node that is directly adjacent to a cutvertex. The distance between a mid node and any node of its reference set is computed
by simple arithmetic. Suppose that node v5 of Figure 1is to be relocated in step 2(c) and that the path length between
its enclosing cutvertices be p. The reference set V _ of nodev5 becomes {c1, c2, v1−v10}. Then, the distance from v5 to
its near cutvertex c1 is 1, and that to c1_s neighbors v1, v2,v6, v7) is 2. The distance from v5 to its far cutvertex c2 is
p−1, that from v5 to any of v3, v4 or v10 is p, and that fromv5 to any of v8 or v9 is p − 2. Therefore, the distance from
a mid node to any node in its reference set is either 1, 2, pathlength (= p) of its enclosing cutvertices, p − 1, or p − 2.
In step 2(d), the reference set V _ of a node v is the neighborsof v within distance of 2, and v is any node in the network.
A single execution of Algorithm 4 takes O(|V _|) time, so thetotal time complexity of steps 2(a)–2(c) isO(n·|PvN|), where|PvN| is the number of pivot nodes. The worst time complexityof step 2(d) isO(n2) since the number of a node's neighbors
within a distance of 2 can be as large as O(n).

V. Abstraction Of Protein Interaction Networks

Alarge number of edges and nodes of a complex protein interactionnetwork often reduces the readability of the networkdue to cluttered edges and nodes. In general there are twoways to analyze such a complex network. One is to extract smaller sub networks from the entire network and to analyze each of the sub networks one by one. Another is to abstract theentire network into a simpler one. InterViewer3 can extract a sub network in several ways. For example, it can extract a sub network of proteins within specified interacting distancefrom one or more target proteins or a sub network of proteins

Metabolic Networks

Metabolic reactions are fundamental to life processes, e.g., for the production of energyand the synthesis of substances. A huge number of reactions occur at any time in livingcells and the product of one reaction is usually used by another reaction, thus metabolicreactions are strongly interconnected and form metabolic pathways and networks.

A metabolic reaction R is a transformation of chemical substances or metabolites (reactants) into other substances (products) usually catalyzed by enzymes. In general metabolicreactions are reversible, that is, they occur in both directions. Such reactions are characterizedby a steady state, i.e., if occurring isolated they reach a state where the amountof change in both directions is equal. A cell is in a constant exchange of substances withits environment. Furthermore, many reactions are regulated, i.e., they are suppressed orenhanced by other factors (allosteric control). This shifts the steady state and togetherwith the steady supply of substances from outside and their final use, e.g., by exportingthem from the cell, one can consider a main direction of a reaction. This is also expressedby the differentiation of substances into reactants and products. As already seen, metabolicreactions interact with each other, i.e., the product of one reaction is usually a reactant ofanother reaction. A metabolic path P = (R1, . . . ,Rn) is a sequence of metabolic reactionswhere for all $1 \leq i < n$ at least one product of reaction Ri is a reactant of reaction Ri+1.

The metabolic network or metabolism of a particular cell or an organism is the complete network of metabolic reactions of this cell or organism. A metabolic pathway is a connectedsub-network of the metabolic network either representing specific processes or defined byfunctional boundaries, e.g., the network between an initial and a final substance as shownin Figure 20.5.

From a formal point of view a metabolic pathway is a hyper-graph. The nodes representthe substances and the hyper-edges represent the reactions. A hyper-edge connectsall substances of a reaction, is directed from

reactants to products and is labeled with theenzymes that catalyze the reaction. Hyper-graphs can be represented by bipartite graphs.

Additionally to the nodes representing substances, the reactions are nodes (either labeledwith the enzymes or with further nodes for enzymes) and edges are binary relations connectingthe substances of a reaction with the corresponding reaction node. This is a commonmodeling of metabolic pathways, e.g., for their simulation using Petri-nets [HT98, RML93].

For the analysis and visualization of metabolic pathways substances are often divided intotwo types [MZ03]: main substances and co-substances. Co-substances are usually small orcurrent metabolites, e.g., ATP, ADP, H2O, NH3 and NADH. These substances normallytransfer electrons or functional groups such as phosphate and amino groups [NIS90]. Mainsubstances are all other metabolites. However, this is not a global property but is givenaccording to the reaction [MZ03], and a small metabolite such as ATP may be consideredas main substance in a particular reaction. For visualization purposes this distinction isimportant as main substances and co-substances are often differently visually represented.

Here a metabolic pathway is modeled as directed bipartite graph G = (VS, VR,E) withnodes u1, . . . , un,w1, . . . ,wm ∈VS representing substances, nodes v ∈VR representing reactions(including the enzyme(s) catalyzing the reaction) and directed edges (u1, v), . . . , (un, v),(v,w1), . . . , (v,wm) ∈E representing the transformation of substances u1, . . . , un to substancesw1, . . . ,wm by the reaction v. A reversible reaction does not contain backwardedges as in some models for simulation purposes, instead this property of an reaction isrepresented by an attribute. Another attribute is used to mark main and co-substances.

Types of Metabolic Networks

• Simplified metabolic network : A network which contains reactions, enzymes andmain substances, but no co-substances.
• Metabolite network and simplified metabolite network: A network which consistsonly of substances (metabolites); in the simplified case only of main substances.
• Enzyme network : A network which consists only of the enzymes catalyzing there actions.(a)

References

[1]. Appel, A. Bairoch, and D. F. Hochstrasser. A new generation ofinformation retrieval tools for biologists: The example of the ExPASyWWW server. Trends Biochemical Sciences, 19:258–260, 1994.
[2]. W. Basalaj. Incremental multidimensional scaling method for databasevisualization. In R. F. Erbacher, P. C. Chen, and C. M. Wittenbrink,editors, Visual Data Exploration and Analysis VI (Proc. SPIE), volume
[3]. ofProceedings of SPIE, pages 149–158, 1999.3[Bax03] A. D. Baxevanis. The molecular biology database collection: 2003 update.Nucleic Acids Research, 31(1):1–12, 2003.
[4]. C. Bachmaier, U. Brandes, and B. Schlieper. Drawings of phylogenetictrees (extended abstract). In X. Deng and D. Du, editors, Algorithmsand Computation, Proc. ISAAC 2005, volume 3827 of LNCS, pages 1110–1121. Springer, 2005.
[5]. G. D. Bader, D. Betel D, and C. W. Hogue. BIND: the biomolecularinteraction network database. Nucleic Acids Research, 31(1):248–250,2003.
[6]. Brandes, T. Dwyer, and F. Schreiber. Visual triangulation of networkbasedphylogenetic trees. In O. Deussen, C. Hansen, D. Keim, andD. Saupe, editors, Data Visualization (Proc. VisSym'04), pages 75–84.Eurographics Association, 2004.
[7]. U. Brandes, T. Dwyer, and F. Schreiber. Visual understanding ofmetabolic pathways across organisms using layout in two and a half dimensions.Journal of Integrative Bioinformatics, 1:2 (EPub), 2004.
[8]. W. Basalaj and K. Eilbeck. Straight-line drawings of protein interactions(system demonstration). In J. Kratochv´ıl, editor, Graph Drawing (Proc.GD '99), volume 1731 of Lecture Notes Comput. Sci., pages 259–266.Springer-Verlag, 1999.
[9]. F. J. Brandenburg, M. Forster, A. Pick, M. Raitner, and F. Schreiber.Graph Drawing Software, chapter BioPath – Exploration and Visualizationof Biochemical Pathways, pages 215–236. Springer Mathematics andVisualization Series, 2004.
[10]. L. Borisjuk, M.-R. Hajirezaei, C. Klukas, H. Rolletschek, andF. Schreiber. Integrating data from biological experiments into metabolicnetworks with the DBE information system. In Silico Biology, 5(2):93–102, 2005.
[11]. [BJDG+03] Z. Bar-Joseph, E. D. Demaine, D. K. Gifford, A. M. Hamel, T. S.
[12]. Jaakkola, and N. Srebro. K-ary clustering with optimal leaf orderingfor gene expression data. Bioinformatics, 19(9):1070–1078, 2003.
[13]. [BM02] V. Batagelj and A. Mrvar. Pajek – analysis and visualization of largenetworks. In P. Mutzel, M. J¨unger, and S. Leipert, editors, Graph Drawing (Proc. GD '01), volume 2265 of Lecture Notes Comput. Sci., pages477–478, 2002.
[14]. [BR01] M. Y. Becker and I. Rojas. A graph layout algorithm for drawingmetabolic pathways. Bioinformatics, 17(5):461–467, 2001.REFERENCES 647
[15]. [Cam96] N. A. Campbell. Biology. The Benjamin-Cummings Publishing Company,1996.
[16]. [Car04a] S. F. Carrizo. Phylogenetic trees: An information visualization perspective.
[17]. In Y.-P. Phoebe Chen, editor, Bioinformatics (Proc. APBC 2004),volume 29 of Conf. Res. Pract. Inform. Techn., pages 315–320, 2004.
[18]. [Car04b] S. F. Carrizo. A survey of phylogenetic researchers: Results.http://www.cs.usyd.edu.au/~scarrizo/Carrizo_PhylogeneticsSurveyResults.doc, January 2004.
[19]. [CG10] G. R. Cochrane and M. Y. Galperin. The 2010 Nucleic Acids Researchdatabase issue and online database collection: a community of data resources.Nucleic Acids Research, 38:D1–D4, 2010.
[20]. [DBD+02] E. Demir, O. Babur, U. Dogrus¨oz, A. G¨ursoy, G. Nisanci, R. C¸ etinAtalay,and M. Ozturk. PATIKA: an integrated visual environment for collaborativeconstruction and analysis of cellular pathways. Bioinformatics,18(7):996–1003, 2002.
[21]. [DETT99] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. Graph Drawing.Prentice Hall, Upper Saddle River, NJ, 1999.
[22]. [DRS04] T. Dwyer, H. Rolletschek, and F. Schreiber. Representing experimentalbiological data in metabolic networks. In Y. P. Chen, editor, Bioinformatics (Proc. APBC'04), volume 29 of Conf. Res. Pract. Inform. Techn.,pages 13–20, 2004.

[23]. T. Dwyer and F. Schreiber. Optimal leaf ordering for two and ahalf dimensional phylogenetic tree visualization. In N. Churcher andC. Churcher, editors, Information Visualisation (Proc. invis.au 2004),volume 35 of Conf. Res. Pract. Inform. Techn., pages 109–115, 2004.
[24]. P. Eades. A heuristic for graph drawing. Congr. Numer., 42:149–160,1984.
[25]. P. D. Eades. Drawing free trees. Bulletin of the Institute for Combinatorics and its Applications, 5:10–36, 1992.
[26]. J. Ellson, E. R. Gansner, E. Koutsofios, S. C. North, and G. Woodhull.Graphviz – open source graph drawing tools. In P. Mutzel, M. J¨ungerand S. Leipert, editors, Graph Drawing (Proc. GD'01), volume 2265 ofLecture Notes Comput. Sci., pages 483–484, 2001.
[27]. P. Eades and M. L. Huang. Navigating clustered graphs using forcedirectedmethods. Journal of Graph Algorithms Applications, 4(3):157–181, 2000.
[28]. L. B. Ellis, C. D. Hershberger, and L. P. Wackett. The university of Minnesota biocatalysis/biodegradation database: Microorganisms, genomicsand prediction. Nucleic Acids Research, 28(1):377–379, 2000.
[29]. J. Felsenstein. Maximum likelihood and minimum-steps methods for estimatingevolutionary trees from data on discrete characters. SystematicZoology, 22:240–249, 1973.
[30]. .J. Felsenstein. The newick tree format. http://evolution.gs.washington.edu/phylip/newicktree.html, 1995.
[31]. W. M. Fitch. Toward defining the course of evolution: Minimum changefor a specified tree topology. Systematic Zoology,
[32]. .C. Friedrich and F. Schreiber. Visualisation and navigation methodsfor typed protein-protein interaction networks. Applied Bioinformatics,2(S3):19–24, 2003.[FS04] C. Friedrich and F. Schreiber. Flexible layering in hierarchical drawings
[33]. **A Constrained, Force-Directed Layout Algorithm for Biological Pathways.** In Proc. International Symposium on Graph Drawing (GD'03), LNCS.Volume 2912. Edited by Liotta G. Perugia: Springer; 2003:**:**314-319.
[34]. **A fast layout algorithm for protein interaction.**Networks Kyunsgook Han and Byong-Hyon Ju School of Computer Science & Enginnering,Inha University,Inchon 402-751,Korea Received on May 1,2003;revised and accepted on July 15,2003