Discovering Temporal Relations in Molecular Pathways Using Protein-Protein Interactions

Martin Farach-Colton
Department of Computer Science, Rutgers University
Piscataway, NJ 08854, USA
farach@cs.rutgers.edu

Yang Huang
Department of Computer Science, Rutgers University
Piscataway, NJ 08854, USA
yahuang@paul.rutgers.edu

John L. Woolford
Department of Biological Science, Carnegie Mellon University
Pittsburgh, PA 15213, USA
jw17@andrew.cmu.edu

ABSTRACT

The availability of large-scale protein-protein interaction data provides us with many opportunities to study molecular pathways involving proteins. In this paper we propose to mine temporal relations in molecular pathways by protein-protein interaction data. In particular, we model the assembly pathways of protein complexes with interval graphs and determine the temporal order of joining the pathway for proteins by ordering the vertices in the interaction graph. We develop a tool called XRONOS to perform such a computation. We then apply XRONOS to the ribosome assembly pathway and present validation results for the obtained ordering. The results are promising and show the potential usage for XRONOS in the study of molecular pathways.

Categories and Subject Descriptors

J.3 [**Computer Applications**]: Life and Medical Sciences

General Terms

Algorithms, Experimentation

Keywords

Interval graphs, probe interval graphs, vertex ordering, protein-protein interaction, molecular pathways, ribosomal assembly pathway

1. INTRODUCTION

The ultimate goal of proteomics is to elucidate all protein-related molecular processes *in vivo*, which in turn requires full knowledge of which proteins participate in each cellular process and how they interact with each other during the process. Examples of such processes and components include: relatively stable protein complexes like the ribosome or the nucleosome; metabolic pathways; and protein complex assembly pathways, such as the process that leads to the assembly of the ribosome.

The recent advent of high-throughput biochemical methods for detecting protein-protein interactions (ppi) has provided a new approach for understanding known cellular processes and for discovering new complexes and pathways. We can model the set of interactions as a graph, which we refer to as the *PPI graph*, where the nodes represent proteins and each edge represents a detected (or inferred) ppi. Note that some lab techniques can determine not only that two proteins bind directly, but that they belong to the same complex. That is, the lab techniques can determine some part of the transitive closure of the direct interaction graph. The PPI graph would equal its transitive closure in the absence of lab errors if each protein participated in one complex and if the protein complexes didn't change over time.

One of the important ways in which complexes change over time is that protein complexes get assembled via some pathway. For example, the ribosome has many helper proteins – somewhat confusingly referred to as *nonribosomal proteins* – that participate in its assembly but don't appear in mature ribosomes, more about which later. While quite a bit of work has been done on mining the PPI graph for heretofore unknown protein complexes and pathways, we know of no work that tries to extract the temporal sequence of protein interactions in assembly pathways.

In this paper, we develop such a tool, which we call XRONOS, and show the results of applying XRONOS to the proteins from the ribosome assembly pathway. While a fair amount of *ad hoc* information is known about the assembly of the ribosome, XRONOS replies only on the PPI graph restricted to the proteins in the pathway, so we believe that XRONOS will generalize to other pathways.

We provide the road map of this paper as follows: In Section 2, we provide more background information, including some details of ribosomal assembly and the detection of ppi's. In Section 3, we describe our model of assembly pathways. Section 4 describes our algorithm and its implementation. Section 5 shows experimental results and provides two validation tests. Section 6 concludes this paper and addresses our future work.

2. BACKGROUND

2.1 Protein-protein interactions

RECOMB'04, March 27–31, 2004, San Diego, California, USA.

Molecular pathways are mainly determined by interactions among biological molecules, which include but are not limited to protein-protein interactions, protein-nucleic acid interactions and interactions between proteins and other small molecules. Among these, protein-protein interactions play important roles in almost every biological process, such as signal transduction, DNA replication, DNA repair, transcription and translation. Traditionally, people studied protein-protein interactions individually by biochemical and biophysical experiments. The experiments only focused on a few proteins at a time and their possible interactions. The rate of accumulation of protein interaction data was rather slow.

Recent techniques have led to an explosion in the quantity of such measured interactions. These techniques include correlated mRNA expression analysis [9], yeast two-hybrid systems [24, 12], and protein complex purification with mass spectrometry [6, 8] (*mass spec*, for short). Some authors have attempted to use genome analysis and other techniques to predict other interactions [14, 19, 18].

Our data come from a couple of large-scale studies that used mass spec, so we give a brief sketch of the method here. First, DNA sequences encoding a tandem affinity tag are recombined at the 3' end of target ORFs in the genome, to enable tandem affinity purification of the protein (and associated molecules) from cells. These target proteins are known as *bait proteins*. They then form protein complexes with other proteins *in vivo*. Each bait protein is then extracted from the yeast culture using its tag. The baits often remain in their complexes, so the non-bait proteins are brought along with the bait. Finally, the complexes are analyzed by mass spectrometry to identify their protein components.

Several specialized interaction databases have been built for depositing protein interaction data, such as DIP [25], BIND [1], GRID [3] and MIPS [16]. Most of the data in these databases were obtained using various high-throughput methods.

2.2 Prior work on the PPI Graph

Many groups have attempted to identify proteins that coordinate to achieve a specific biological task. Segal *et al.* [22] used a probabilistic network, which was learned from both protein interaction data and gene expression profiles using the EM algorithm. Ideker *et al.* [11] combined protein-protein interaction, protein-DNA interaction and mRNA expression profile data and used simulated annealing to identify active subnetworks, which are connected regions showing significant changes in expression. Friedman *et al.* [5] employed Bayesian Network to recover the the causal relations among proteins.

However, much of the previous work focuses on identifying groups of genes/proteins that are likely to interact with each other, or find the causal relationship among those proteins. To the best of our knowledge, no one tried to extract the temporal information from available data.

2.3 The Ribosome

Despite the central and highly conserved role of ribosomes in biology (catalyzing the synthesis of proteins by translation of the genetic code in messenger RNA), we know little about the assembly pathway of ribosomes. In eukaryotes, these ribonucleoprotein particles contain 80 different ribosomal proteins and 4 different ribosomal RNAs, which are assembled together with each other primarily in the nucleolar subcompartment of the nucleus of cells. Some late steps of assembly occur in the nucleoplasm or in the cytoplasm. Recent genetic and biochemical studies in yeast have identified more than 150 different "nonribosomal" proteins that are necessary for assembly of yeast ribosomes, but which do not end up in mature ribosomes [17]. Genetic analysis of yeast strains mutant for these proteins suggests steps in assembly of ribosomes requiring their function, thus providing some temporal coordinates for function. Subcellular localization of the proteins (nucleolar, nucleoplasmic, and/or cytoplasmic) also indicates spatial coordinates, and thus temporal coordinates. Proteins localized in the nucleolus are thought to function in early steps, nucleoplasmic proteins in middle steps, and cytoplasmic proteins in late steps.

The ribosomal assembly pathway consists of an initial segment before the ribosomal RNA is cleaved, followed by two separate pathways for two final ribosomal complexes, the so-called 40S and 60S particles. In our study below, we concentrate on proteins involved in the formation of the 60S particle.

Thus, we have some general ideas of the order in which proteins associate with the forming ribosome and the order in which they dissociate – in the case of "nonribosomal" proteins but this map is far from detailed. Such an ordered pathway will be a key step to learn the mechanisms of ribosome assembly in yeast and other eukaryotes. This pathway is intimately linked with the growth and proliferation of cells, and understanding ribosome biogenesis is a significant goal.

3. METHODS

3.1 Modeling molecular pathways

There are several types of models for molecular pathways, especially for gene signaling pathways [23]. We will consider only protein-complex assembly pathways, and so we can use a very simple model. On the time axis, the pathway starts at point t_{start}, which we can take to be time 0. Then at some point $t_{e,i}$ the protein p_i "enters" the pathway by binding to other proteins in the complex. Note that we assume that each protein only enters the pathway once. At time $t_{l,i}$, if protein i is a helper protein, it "leaves" the complex, that is, it stops binding to the other proteins in the complex. If it is a component of the complex, $t_{l,i}$ can be set to t_{end} since the protein will stay in the final product of the pathway. When the pathway completes, at time t_{end}, the complex is mature. Clearly, t_{start} and t_{end} can be arbitrary, as long as for $\forall i$, $t_{e,i} \geq t_{start}$ and $t_{l,i} \leq t_{end}$. One such pathway is showed in Figure 1(a).

Our simple model assumes that there is no branching in the pathway, which is not true of the ribosomal pathway. But as noted above, our experimental evaluation will deal with only one branch of the pathway, the 60S branch.

Given such a representation of the pathway, we may construct a graph G_t for the pathway, where vertex v_i corresponds to the time interval $(t_{e,i}, t_{l,i})$ associated with protein p_i, and there is an edge between two vertices if and only if the two corresponding intervals intersect. Such a graph will be be an *interval graph*. The definition of interval graph is:

DEFINITION 1. *Given V, a set of intervals on the real*

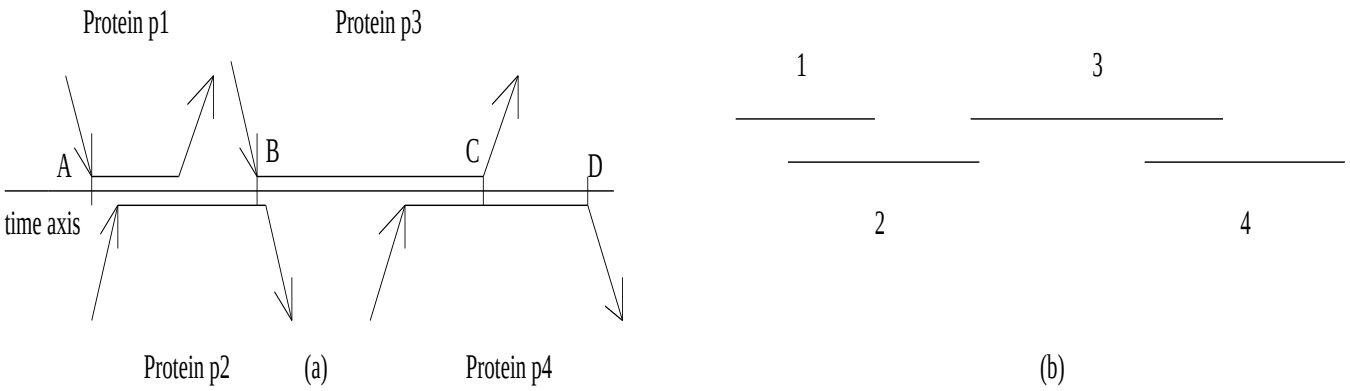

Figure 1: The left side of the figure is an imaginary pathway which involves four proteins: p1, p2, p3 and p4. On the time axis, the point A indicates $t_{e,1}$, the point B indicates $t_{e,3}$, the point C indicated $t_{l,3}$ and the point D indicates $t_{l,4}$. The right side of the figure is a correct arrangement of time intervals. The graph G_t will be a path connecting 4 vertices.

line, a graph $G = (V, E)$ is called an interval graph if the edge set is $E = \{(v_i, v_j) | v_i \cap v_j \neq \emptyset\}$, i.e., two vertices are connected by an edge if and only if the two intervals intersect. The set of intervals is called an interval representation, realization or model of the graph.

Suppose we consider mass spec data for proteins known to participate in some pathway. Then we can assume that two proteins, p_i and p_j, for which a mass spec interaction is detected have overlapping time intervals, that is v_i and v_j are adjacent in G_t. As a result, we are able to make the following observation:

OBSERVATION 1. *Given the complete interaction data about one linear pathway, the graph G_t of the pathway can be reconstructed.*

Similarly,

OBSERVATION 2. *Given some interaction data about one pathway containing complete interaction data about one period of time, during which the pathway is of linear shape, the subgraph of G_t corresponding to that period of time can be reconstructed.*

The above observation casts some light on discovering temporal relation of molecular pathway from protein-protein interactions. An important characterizations [4] of interval graphs is that a graph is interval iff there exists an *I-ordering* of its vertices. The definition of I-ordering is as follows:

DEFINITION 2. *Given a graph $V = (G, E)$, an* ordering *of V is a bijective function $O : V \to [1 \ldots |V|]$. For $v \in V$, $O(v)$ is called the vertex v's rank.*

DEFINITION 3. *Given a graph $G = (V, E)$, an ordering O of V is an* I-ordering *iff for any $u, v, w \in V$, with $O(u) < O(v) < O(w)$,*

$$(u, w) \in E \Longrightarrow (u, v) \in E \tag{1}$$

Suppose G_t is interval, and order the vertices by the left end point $t_{e,i}$ of their corresponding intervals. Such an ordering is, in fact, an I-ordering. Thus the I-ordering of the graph G_t(or G_i) lets us decide the correct arrangement of time intervals on the pathway. To summarize, We may discover the temporal order by which proteins enter the pathway and the order by which protein-protein interactions occur on the pathway by studying protein-protein interaction data.

So far, we have considered what happens when the protein interaction network is fully computed. But in mass spec, we measure interactions between bait proteins and all others in each experiment. In the absence of complete information, this type of data is better modeled by a *probe interval graph.*

DEFINITION 4. *Let $G = (V, E)$ be a graph, and let $P \subseteq V$. Then $G' = (V, P, E')$ is a* probe subset graph *if $E' = \{(u, v) | (u, v) \in E$ and $u, v \in P\}$, that is, it is the subgraph of a graph where we only know the adjacencies of probe nodes in P.*

$G' = (V, P, E')$ is a probe interval graph *if there exists an interval graph G such that G' is a probe subset graph of G. The vertices in P are called probes and the vertices in $N = V - P$ are called non-probes. The induced subgraph on P is an interval graph and N is an independent set.*

The probe interval graph [26, 27] was first proposed to model physical mapping of DNA sequences. McMorris *et al.* [15] discovered several important properties about them, such as that probe interval graphs are perfect and that their intrinsic matrices have consecutive-1's. Johnson and Spinrad [13] designed the first $O(n^2)$ algorithm for recognizing probe interval graphs when the probe/non-probe partition is given.

Given protein interaction data from mass spec, we can obtain an interaction graph, where bait proteins correspond to probes and other proteins correspond to non-probes. Obviously, every edge is incident on at least one probe. In the absence of error, the interaction graph will be a probe interval graph. By modeling our data as a probe interval graph, we can deal with partial information.

In the absence of errors, we could use Johnson and Spinrad's recognition algorithm to generate all possible interval representations corresponding to a probe interval graph. One of them should be the correct arrangement of time intervals $(t_{e,i}, t_{l,i})$ on the pathway.

3.2 Algorithm for vertex ordering

Suppose we get G_i from complete protein-protein interaction data, and so G_i is interval. The interval representation of G_i gives us the arrangement of proteins in their temporal order. Although there are many false-positives and false-negatives in current protein-protein interaction data, computing the ordering of vertices for interaction graph G_i is still a good start.

We apply the 5-sweep LBFS algorithm [21] on G_i to compute the ordering of the vertices. The LBFS algorithm recognizes an interval graph by generating and then checking one vertex ordering. The algorithm consists of two passes: first, produce an ordering of the vertices; second, determine whether the graph is interval by checking whether the ordering is an I-ordering. For our purposes, the crucial property of this algorithm is that if the underlying graph is not an interval graph, but some of its subgraphs are interval graphs, the algorithm will still output an ordering that will be an I-ordering for the interval subgraph, that is, the vertices of the interval subgraphs are correctly ordered. Which particular interval subgraphs get correctly ordered will depend on the order in which the nodes of the graph are encountered.

4. THE PARTICULARS OF XRONOS

We begin by considering the nonribosomal proteins only associating with precursors to the 60S subunit of the yeast ribosome. Recall that the nonribosomal proteins are the proteins that transiently associate with assembling ribosomes and perform some function necessary for the maturation of ribosomes. But none of them ends up in the mature ribosome. Note that the final product of the constructed pathway will not be 60S subunit. Instead, it will show the order in which those nonribosomal proteins join the pathway. About 50% of the protein data come from Dr. Woolford's lab or published papers. The other data only include proteins that are copurified with each epitope-tagged protein and account for the other 50%. These are only from the large-scale mass spec study of Gavin *et al.* [6]. The gene for each protein might have several names in the literature, so we canonicalize by ORF names, merging the adjacencies of proteins with the same ORF name.

The adjacency matrix is so derived, where the columns indicate which proteins bait proteins are adjacent with. However, this matrix needs further cleaning, since the submatrix induced by bait proteins need not be symmetric. We solve this problem by considering two proteins to be adjacent if either adjacency is present in the matrix.

Finally we obtain a 96×25 protein-protein interaction matrix. There are 25 bait proteins and a total of 96 proteins in the matrix. 514 interactions are found in the matrix among which 24 are self-interactions.

XRONOS**..** The input of XRONOS is the cleaned adjacency matrix. The output is a (hopefully small) set of orderings of the proteins, which will correspond to their temporal ordering. XRONOS randomly permutes the rows and columns of the interaction matrix. On each permutation, it applies the 5-sweep LBFS algorithm to compute a vertex ordering. Repeating the process many times, we record the rank of each vertex in each ordering.

5. EXPERIMENTAL RESULTS

We ran XRONOS on our input, using 5000 iterations. In the worst case, each protein could end up with 96 different ranks. Interestingly, though G_i is not an interval graph, we found that most of vertices only have two different ranks and can be arranged in two orderings. The vertices with degree 1 that share the same adjacent vertex have more than two ranks. Those vertices can be grouped according to their adjacent vertex. For example, YPR010C, YGL036W and YNL021W are in one group, since they are all adjacent only to YPR016C. They have 6 ranks: 46, 47, 48, 86, 87 and 88. If we substitute one pseudo-vertex for each of such group of vertices all vertices, including pseudo-vertices, can be arrange in those two orderings. Let the two orderings be O_1 and O_2, where O_1 appears 2947 times and O_2 appears 2053 times. Thus, while an ordering was not recovered completely unambiguously, there was enough of a signal so that only two orderings emerged, and one of these was dominant.

5.1 Validation tests for the ordering

Since there are no experiment data for us to check the accuracy of our computed ordering, we designed two validation tests to verify the orderings O_1 and O_2. First, we noticed that there are some protein-protein interactions in the GRID database that were not part of our input. Indeed, there are 54 such interactions. We refer to the interacting pairs in our input as $Pair_{c1}$ and the pairs from GRID not in our input set as $Pair_{c2}$. We note that this is not simply the "withhold 10% at random" type of validation, because the 54 interactions in $Pair_{c2}$ were obtained using different lab techniques than the 490 pairs (the final number of interactions after all the filtering was done) in $Pair_{c1}$.

We model the problem of measuring how one particular ordering arranges those two classes of interacting pairs as the *minimum linear arrangement problem*. The minimum linear arrangement problem can be stated as follows: given the graph $G = (V, E)$ and the function $f : V \rightarrow [1 \ldots |V|]$ are given, we try to find a vertex ordering to minimize the objective function

$$\frac{\sum_{(u,v)\in E} |f(u) - f(v)|}{|E|}|$$

It is well known that this problem is an NP-hard problem in general. There are only approximation algorithms [20, 2]. G_i is the input interaction graph, the ordering we obtain from G_i can be viewed as mapping f, the rank of the protein in that ordering can be viewed as the value of f on the corresponding vertex in the interaction graph, and the mean of distance of interacting pairs in G_i is just the objective function. Furthermore, we define the distance of one interacting pair, so that it corresponds to $|f(u) - f(v)|$.

DEFINITION 5. *Given the ordering O and the interacting protein pair (p_i, p_j), the distance of the pair in the ordering O is the difference between ranks of p_i and p_j in the ordering.*

The reason we want to measure the mean of distance of interacting pairs is based on the following intuition: the ordering of the true pathway should minimize the distance of interacting pairs somehow, because it tends to arrange proteins proximally those proteins that interact with one another in proximity. Hence, the ordering with a smaller objective function value indicates it can be a better approximation for the pathway.

	Ordering O_1		Ordering O_2	
	$Pair_{c1}$	$pair_{c2}$	$Pair_{c1}$	$Pair_{c2}$
mean	20.27	26.41	21.99	29.78
std. dev.	18.79	18.72	20.92	19.85

Table 1: Mean and standard deviation of distance of two classes of interacting pairs in orderings O_1 and O_2. $Pair_{c1}$ is the class of interacting pairs in our input dataset, and $Pair_{c2}$ is the class of interacting pairs which are not in the input dataset but found in the GRID database

For all those interacting pairs we compute their distance in each ordering. To give a simple example, consider an ordering with only four proteins. Let the ordering be $[p_3, p_4, p_2, p_1]$ where ranks for proteins p_1, p_2, p_3 and p_4 are 4, 3, 1 and 2 respectively. Suppose in the input data (p_1, p_4), (p_2, p_3) and (p_4, p_2) are interacting pairs. And suppose (p_1, p_3) and (p_3, p_4) are in the class $Pair_{c2}$. The distance for one interacting pair is defined to be the difference between ranks of two proteins. Thus the distances for (p_1, p_4), (p_2, p_3) and (p_4, p_2) in that ordering are $2(2 = |1 - 3|)$, $2(2 = |2 - 4|)$ and $1(1 = |3 - 2|)$, while the distances for (p_1, p_3) and (p_3, p_4) are 3 and 1. So the mean of distance of interacting pairs in the $Pair_{c1}$ is $\frac{2+2+1}{3}$, while the mean of distance of interacting pairs in the $Pair_{c2}$ is $\frac{3+1}{2}$. For those vertices that have more than two ranks we arrange them in O_1/O_2 by their ranks which get most/least hits during 5000 times of running. In the Table 1 we give the mean and standard deviation of distance of the two classes of interacting pairs in the two orderings.

The expected distance for an interacting pair in a random ordering is:

$$\frac{\sum_{k=1}^{n-1} k(n-k)}{\binom{n}{2}} = \frac{n+1}{3} \tag{2}$$

where k denotes distance of the pair, $n - k$ denotes the number of times such a distance can occur in all possible orderings, and $\binom{n}{2}$ is the number of possible orderings for n proteins without considering the relative order of the two proteins in that pair. So the expected distance is 32.33 when $n = 96$. The mean of distance of interacting pairs in orderings O_1 and O_2 are clearly less than the random orderings. The distribution of distance is shown in Figure 2 and Figure 3. Each data point (x, y) in the figure shows that there are y interacting pairs having distance x in one ordering. The Figure 2 shows the distribution for interacting pairs in O_1 and Figure 3 shows the distribution for interacting pairs in O_2. We make two observations here: Compared to interacting pairs only found in GRID, unseen in the input data, interacting pairs in the input data are arranged much closer in both orderings by the algorithm; Interacting pairs are closer in O_1 than O_2. Our conjecture is that O_1 is better approximation than O_2.

For the second validation test, we compute the orderings of vertices in random graphs and then compute the distance of interacting pairs in the the random graphs and the distance of interacting pairs only found in GRID. If XRONOS generated few orderings for random graphs, or gave orderings with small linear-arrangement scores, then the above scores would merely indicate that XRONOS does non-random things to graph, rather than that it is capturing interval structure of the graphs.

	$Pair_{c1}$	$Pair_{c2}$
test 1	22.77	30.40
test 2	24.23	33.89
test 3	23.10	33.95
test 4	22.73	31.23
test 5	22.03	30.19
test 6	23.19	29.21
test 7	21.94	31.20
test 8	22.42	30.30
test 9	22.50	28.82
test 10	22.50	29.76
mean	22.73	30.90
std. dev	0.66	1.77

Table 2: Mean of the distance of two classes of interacting pairs during 10-time random graph test. $Pair_{c1}$ is the class of interacting pairs in the random graph, and $Pair_{c2}$ is the class of interacting pairs which are not in the input dataset but found in the GRID database

This way we can test the null hypothesis that the interaction graph constructed from our protein interaction dataset is a random graph. We generate random graphs which have the same degree sequence with the input interaction graph. Based on the idea in [7], we implemented an algorithm to simulate a Markov chain on connected graphs with given degree sequence. It can be showed that if we simulate the Markov chain, in the limit the graphs generated by such a Markov chain with the given degree sequence will distribute uniformly at random. In one test we ran the Markov chain for 1 million steps to generate one random graph.

We then ran XRONOS on the random graph, once again iterating 5000 times. Each time we record the rank of each vertex. Finally, for each vertex, we take the rank which gets the most hits in 5000 times as the rank of the vertex in the output ordering. Finally we compute the mean of distance of "interacting pairs" in $Pair_{c1}$, which correspond to edges in the random graph, and interacting pairs in $Pair_{c2}$ based on the output ordering. We perform this test 10 times. In Table 2. we show the distance of interacting pairs in the $Pair_{c1}$ and the distance of interacting pairs in the $Pair_{c2}$

We perform a one-sample one-tailed T-test: $H_0 : \mu = 20.27$ $H_1 : \mu > 20.27$ where 20.27 is the mean of the distance of interacting pairs in our input dataset(see Table 1) in the ordering O_1, and μ is the mean of the distance of "interacting pairs" corresponding to edges in the random graph. The null hypotheses can be rejected with $P < 0.0001$.

6. CONCLUSIONS AND DISCUSSIONS

In this paper, we present techniques for analyzing an exciting new source of biological data – the burgeoning set of measured protein-protein interaction. We present a tool XRONOS for analyzing assembly pathways in the data, and show that the data contain of surprisingly robust temporal signal.

Here we discuss some of future directions regarding our work. Generally, assembly pathway can be divided into two classes: stepwise assembled pathway and the pathway with preassembled subunits. In the former, each protein in the

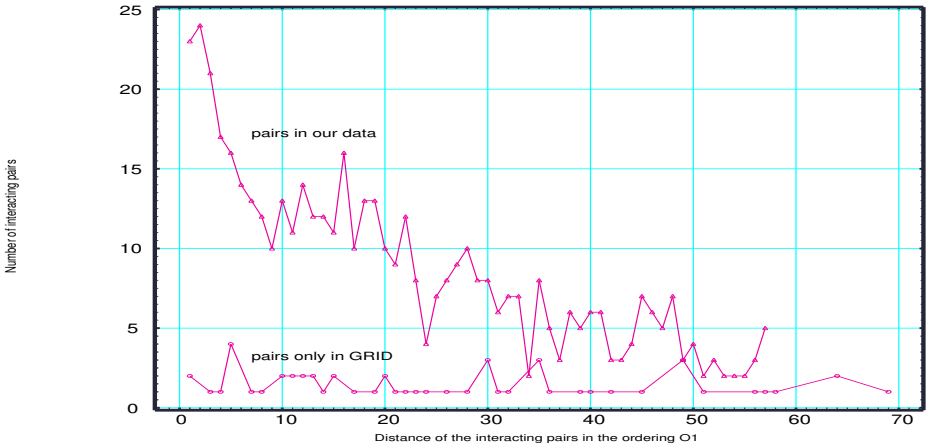

Figure 2: The distribution of distance of interacting pairs in the $Pair_{c1}$ and $Pair_{c2}$ in the ordering O_1

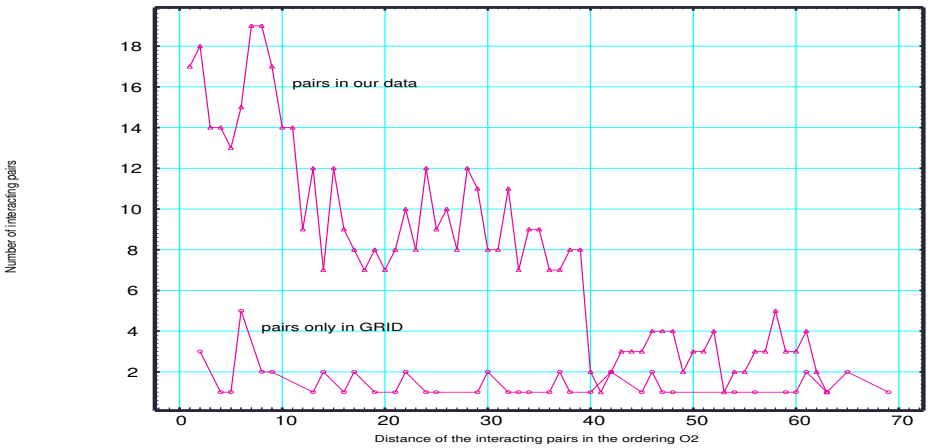

Figure 3: The distribution of distance of interacting pairs in the $Pair_{c1}$ and $Pair_{c2}$ in the ordering O_2

pathway joins the complex one by one by binding some proteins already in the premature complex. In the latter, some proteins bind together to form several preassembled subunits. Then those subunits interact with each other to form the mature complex. For the stepwise assembled pathway we can directly use our tool to recover the pathway. For the pathway with preassembled subunits, we need to first identify proteins involving each subunit, which can be done by analyzing interaction data with mRNA expression data. Then we apply the tool to construct the assembly pathway of subunits. Finally we try to assemble those subunits into the final product. Recently some large-scale experiments were carried out to decide the cellular locations of proteins [10]. Such spatial information will be very helpful to discover temporal relation in the pathway. As we have mentioned before, when two proteins interact they must be in the same cellular location. If we can find some efficient way to analyze protein location data with protein-protein interaction data at the same time, we are able to decide which interactions are more likely to be false positive and adjust the ordering accordingly. One of the key assumptions we made for our model is that the pathway is of linear shape. One of our future work is to consider a more general model to remove this assumption, for example, we may model the pathway by a tree or even a forest. That kind of model should qualify to study more pathways.

7. REFERENCES

[1] G. D. B. andD. Batel and C. W. Hogue. Bind: the biomolecular interaction network database. *Nucleic Acids Reseach*, 31(1):248–250, 2003.

[2] R. Bar-Yehuda, G. Even, J. Feldman, and S. Naora. Computing an optimal orientation of a balanced decomposition tree for linear arrangement problems. *Journal of Graph Algorithms and Applications*, 5(4):1–27, 2001.

[3] B.-J. Breitkreutz, C. Stark, and M. Tyers. The grid: the general repository for interaction datasets. *Genome Biology*, 4(3):R23, 2003.

[4] D. G. Corneil, S. Olariu, and L. Stewart. The lbfs structure and recognition of interval graphs. Submitted.

[5] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian networks to analyze expression data. *Journal of Computational Biology*, 7:601–620, 2000.

[6] A.-C. Gavin, M. Bösche, R. Krause, and etal. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415:141–147, 2002.

[7] C. Gkantsidis, M. Mihail, and E. Zegura. The markov chain simulation method for generating connected power law random graphs. In *Proceedings of the 5th Workshop on Algorithm Engineering and Experiments*, 2003.

[8] Y. Ho, A. Gruhler, A. Heilbut, and etal. Systematic identification of protein complexes in *saccharomyces cerevisiae* by mass spectrometry. *Nature*, 415:180–183, 2002.

[9] T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, M. J. Kidd, A. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000.

[10] W.-K. Huh, J. V. Falvo, L. C. Gerke, A. S. Carroll, R. W. Howson, J. S. Weissman, and E. K. O'Shea. Global analysis of protein localization in budding yeast. *Nature*, 425:686–691, 2003.

[11] T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18(Suppl 1):S233–S240, 2002.

[12] T. Ito, K. Tashiro, S. Muta, R. Ozawa, T. Chiba, M. Nishizawa, K. Yamamoto, S. Kuhara, and Y. Sakaki. Toward a protein-protein interaction map of the budding yeast: A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc. Natl. Acad. Sci. USA*, 97(3):1143–1147, 2000.

[13] J. L. Johnson and J. P. Spinrad. A polynimial algorithm time recognition algorithm for probe interval graphs. In *Proceedings of the 12th Annual Symposium on Discrete Algorithm*, pages 477–486, 2001.

[14] E. M. Marcotte, M. Pellegrini, H.-L. Ng, D. W. Rice, T. O. Yeates, and D. Eisenberg. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285:751–753, 1999.

[15] F. R. McMorris, R. Wang, and P.Zhang. On probe interval graphs. *Discrete Applied Mathematics*, 88:315–324, 1998.

[16] H. W. Mewes, D. Frishman, C. Gruber, B. Geier, D. Haase, A. Kaps, K. Lemcke, G. Mannhaupt, F. Pfeiffer, C. ller, S. Stocker, and B. Weil. Mips: a database for genomes and protein sequences. *Nucleic Acids Research*, 28:37–40, 2000.

[17] P. Milkereit, H. Kuhn, N. Gas, and H. Tsochner. The preribosomal network. *Nucleic Acids Research*, 31:799–804, 2003.

[18] R. Overbeek, M. Fonstein, M. D'Souza, G. D. Pusch, and N. Maltsev. The use of gene clusters to infer functional coupling. *Proc. Natl Acad Sci, USA*, 96(6):2896–2901, 1999.

[19] F. Pazos and A. Valencia. Similarity of phylogenetic trees as indicator of protein-protein interaction. *Protein Engineering*, 14(9):609–614, 2001.

[20] S. Rao and A. W. Richa. New approximation techniques for some ordering problems. In *Proceedings of the 9th Annual Symposium on Discrete Algorithms*, pages 211–218, 1998.

[21] D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5:266–283, 1976.

[22] E. Segal, H. Wang, and D. Koller. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19(Suppl 1):I264–I272, 2003.

[23] P. Smolen, D. A. Baxter, and J. H. Byme. Mathematical modelling of gene networks. *Neuron*, 26:567–580, 2000.

[24] P. Uetz, L. Giot, G. Cagney, T. A. Mansfield, R. S. Judson, J. R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadamodar, M. Yang, M. Johnston, S. Fields, and J. M. Rothberg. A comprehensive analysis of protein-protein interactions in *saccharomyces cerevisiae*. *Nature*, 403:623–627, 2000.

[25] I. Xenarios, L. Salwinski, X. J. Duan, P. Higney, S. M. Kim, and D. Eisenberg. Dip, the database of interacting proteins: a research tool for studyingcellular networks of protein interactions. *Nuclei Acids Reseach*, 30:303–305, 2002.

[26] P. Zhang. Probe interval graph and its applications to physical mapping of dna. *technical report*, 1982.

[27] P. Zhang, E. A. Schon, S. G. Fischer, E. Cayanis, J. Weiss, and S. K. P. E. Bourne. An algorithm based on graph theory for the assembly of contigs in physical mapping of dna. *CABIOS*, 10:309–317, 1994.