BIOINFORMATICS ORIGINAL PAPER Vol. 23 no. 20 2007, pages 2775–2783
doi:10.1093/bioinformatics/btm409

Data and text mining

Mining biological networks for unknown pathways

Ali Cakmak* and Gultekin Ozsoyoglu

Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH 44106, USA

Received on May 2, 2007; revised on July 20, 2007; accepted on August 8, 2007

Advance Access publication August 30, 2007

Associate Editor: John Quackenbush

ABSTRACT

Motivation: Biological pathways provide significant insights on the interaction mechanisms of molecules. Presently, many essential pathways still remain unknown or incomplete for newly sequenced organisms. Moreover, experimental validation of enormous numbers of possible pathway candidates in a wet-lab environment is time- and effort-extensive. Thus, there is a need for comparative genomics tools that help scientists predict pathways in an organism's biological network.

Results: In this article, we propose a technique to discover unknown pathways in organisms. Our approach makes in-depth use of Gene Ontology (GO)-based functionalities of enzymes involved in metabolic pathways as follows:

(i) Model each pathway as a biological functionality graph of enzyme GO functions, which we call pathway functionality template.

(ii) Locate frequent pathway functionality patterns so as to infer previously unknown pathways through pattern matching in metabolic networks of organisms.

We have experimentally evaluated the accuracy of the presented technique for 30 bacterial organisms to predict around 1500 organism-specific versions of 50 reference pathways. Using cross-validation strategy on known pathways, we have been able to infer pathways with 86% precision and 72% recall for enzymes (i.e. nodes). The accuracy of the predicted enzyme relationships has been measured at 85% precision with 64% recall.

Availability: Code upon request.

Contact: ali.cakmak@case.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

In the course of studying organisms at a coarser, systems level, life scientists recently listed (Kelley *et al.*, 2003) the following questions: (i) to what extent are the genomic pathways conserved among different species? (ii) Is there a minimal set of pathways that are required by all organisms? (iii) How are organisms related in terms of the distance between pathways rather than at the level of DNA sequence similarity? At the core of such questions lies the identification of pathways in different organisms. However, experimental validation of an enormous number of possible candidates in a wet-lab environment requires monumental amounts of time and effort. Thus, there is a need for comparative genomics tools that help scientists predict pathways in an organism's biological network.

Due to the complex and incomplete nature of biological data, at the present time, fully automated computational pathway prediction is excessively ambitious. Hence, in this article, we propose a new technique to automatically discover *fragments* of pathways in biological networks so that biologists can proceed to extend discovered fragments into a full pathway with less effort. We consider only metabolic pathways. However, the techniques described here can be applied to other biological networks (e.g. signaling pathways) with minimal modifications. A metabolic pathway is a set of biological reactions where each reaction consumes a set of metabolites, called substrates, and produces another set of metabolites, called products. A reaction is catalyzed by an enzyme (i.e. a gene product) or a set of enzymes.

1.1 Related work

There are many web resources that provide access to curated as well as predicted collections of pathways, e.g. KEGG (Kanehisa *et al.*, 2004), EcoCyc (Keseler *et al.*, 2005), Reactome (Joshi-Tope *et al.*, 2005) and PathCase (Ozsoyoglu *et al.*, 2006). Work to date on discovering biological (sub)networks can be organized under two main titles: (i) Pathway Inference (Osterman and Overbeek, 2003; Pireddu *et al.*, 2005; Shlomi *et al.*, 2006, Yamanishi *et al.*, 2007), and (ii) Whole-Network Detection (Jansen *et al.*, 2003; Tu *et al.*, 2006; Yamanishi *et al.*, 2005). Pathway inference is to discover unknown pathways for a given specific organism. An important step of pathway inference is to mine for graph patterns that are common among existing known pathways/networks (Sharan *et al.*, 2005). Koyuturk *et al.* (2006) casts the problem of finding conserved pathway fragments among species as a frequent itemset mining problem. Tohsato *et al.* (2000) extends the multiple sequence alignment algorithm to the pathway alignment problem, and utilizes the EC (Enzyme Commission) hierarchy to relax the matching of enzymes. Hu *et al.* (2005) discovers coherent dense subgraphs in a network of genes that are constructed based on microarray data. Pinter *et al.* (2005) describes MetaPathwayHunter, a tool that allows for approximate pathway matching and alignment via the use of subtree homeomorphism. To the best of our knowledge, in these studies, the problem is not formulated as a pathway prediction

*To whom correspondence should be addressed.

problem where the hierarchical structure of functional annotations is extensively utilized.

Variations of sequence-based techniques have been the mostwidely employed pathway inference methods. Reactome computationally predicts metabolic pathways using ortholog tables in OrthoMCL database (Joshi-Tope *et al*., 2005). Kharchenko *et al*. (2004) infers missing genes in pathways based on expression data. Pathway Analyst (Pireddu *et al*., 2005) attempts to find catalyzing proteins for each reaction in the target organism via using BLAST sequence similarity, hidden markov models and support vector machines. Similarly, several other works (e.g. Bono *et al*., 1998; Dandekar and Sauerborn, 2002; Kelley *et al*., 2004; Paley and Karp, 2002; Romero *et al*., 2005; Shlomi *et al*., 2006;) heavily rely on sequence homology. Osterman and Overbeek (2003) illustrate the weaknesses of homology-based approaches for finding missing genes, and they suggest additional comparative genomic measures for the final decision on missing enzymes. Green and Karp (2004) integrate a Bayesian network approach into Pathologic (Paley and Karp, 2002) based on the measures discussed by Osterman and Overbeek (2002). Nevertheless, the first set of candidate genes that are further filtered through a Bayesian network are located through sequence-based homology, which does not consistently correspond to functional similarity. Hence, if a candidate gene is not homologous to already known genes in a pathway, sequence-based methodologies tend to ignore such candidates. In addition, such works assume a fixed reference pathway template, and attempt to find individual enzymes corresponding to the reactions in the template. However, pathways may have variations in terms of sequences of reactions (Ye *et al*., 2005). Thus, assuming a prespecified structure for an unknown pathway is another limiting aspect of these previous studies.

Yamanishi *et al*. (2007) combines gene position and phylogenetic profile information to discover missing enzymes of pathways. However, their main assumption is that functionally related genes tend to occur closely in a genome only applies to bacterial genomes, and does not commonly generalize to other organisms.

In addition to individual pathway prediction, the reconstruction of whole biological networks is also a popular research area. Yamanishi *et al*. (2005) builds kernels based on multiple kinds of genomic information such as gene expression data, co-localization data and phylogenetic profile to build kernel-based classifiers that are combined to determine if there is a metabolic interaction between two given enzyme genes. Given a set of user-provided proteins, Bio- PIXIE (Myers *et al*., 2005) predicts a localized network around input proteins using a probabilistic framework. Jansen *et al*. (2003) combines expression and co-localization data into a Bayesian network framework to reconstruct protein interaction network of yeast. Bang *et al*. (2003) adopts a similar strategy to infer the whole metabolic network of an organism. One major drawback of whole-network prediction studies is that predicted interactions cannot be associated with a particular pathway. Hence, outputs of such approaches require further processing to organize inferred interactions into pathways.

In recent years, many general graph mining and indexing methods have been proposed (Huan *et al*., 2003, Huan *et al*., 2004, Kuramochi and Karypis 2001). Most work in graph mining (Yan and Han 2002, Yan *et al*., 2005, Zaki 2005) focuses on extracting exact frequent patterns. Canonical forms are utilized (Kuramochi and Karypis 2001) to test whether two graphs are isomorphic. However, none of the existing frequent subgraph mining methods consider graph structures where nodes are part of a well-defined hierarchy. Please see Section 1 in the Supplementary Material for a more detailed discussion on related work.

1.2 Approach

Here, we propose an alternative *focus change* from enzymes and metabolites of pathways to '*enzyme GO (Gene Ontology) functionalities*' of pathways. Gene Ontology (GO Consortium, 2004) is a controlled term vocabulary containing about 20 000 hierarchically organized concepts, and attaches a new attribute for two genomic entity types, namely, genes and gene products. The *true-path rule* (GO Consortium, 2004) applies to GO, which states that a gene/protein annotated with a GO concept G is also annotated with all the ancestors of G. In particular, the explicit annotation of a gene/protein p with the GO concept G is done *at the most specific level* known in that none of the descendants of G annotate p. In this article, we use concepts from the *GO molecular function subontology* as the units of our functionality representation.

We thus model each metabolic pathway as a *functional* pathway graph of *enzyme GO functions*, which we call *pathway functionality template (PFT)*, and focus on enzyme GO functions [i.e. the *pathway GO functionality (PF) domain*]. Figure 1 shows a sample pathway where rectangles represent reactions (labeled with names of genes encoding for their catalyzing enzymes), and circles labeled with letter 'm' represent metabolites (which are not explicitly named here for simplicity) being consumed and/or produced. Figure 2 depicts the PFT of the same pathway where enzymes are replaced with their most-specific functional annotations, and, for simplicity in presentation, metabolites are omitted.

Note that due to (a) multiple GO annotations of enzymes, (b) size (20 000 concepts) and type (not a tree, but a directed acyclic graph) of the GO hierarchy and (c) the true-path rule, large numbers of PFTs are likely to exist for a given pathway. Thus, building efficient pathway prediction/functionality conservation algorithms is a challenging task.

Our motivation behind the use of pathway functionality templates is that essential cellular actions are common to a large set of organisms regardless of their complexity

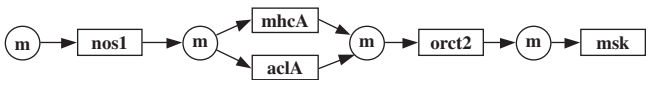

Fig. 1. A sample pathway.

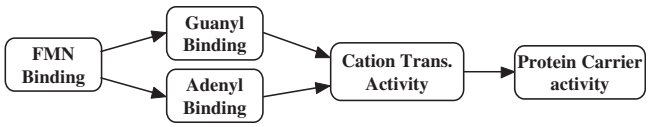

Fig. 2. The functionality template of the pathway in Figure 1.

(Kelley *et al.*, 2003). However, the same function in different organisms can be carried out by different genomic agents with similar functional annotations. Hence, to compensate for the variances in genomes of different organisms, and yet to accommodate the commonness in the blueprints of biological processes, we argue that the unit of focus may be shifted to the function carried out in each individual step of a pathway, rather than the performer of the step, i.e. the enzymes.

1.3 Contributions

Contributions of this article are as follows:

- A new GO-based *gene-function-centric* pathways paradigm which can accommodate genetic variations among organisms at the functionality level.
- A metabolic pathway inference framework tool that efficiently and effectively predicts unknown pathways of organisms.
- An effective algorithm for mining frequent PF patterns that are common in most organisms.
- Extension of generalized suffix trees (Gusfield, 1997) to index multiple PFTs.
- Evaluation of proposed model's accuracy through precision-recall analysis.

The research presented in this article is performed as part of PathCase Pathways Database System (Ozsoyoglu *et al.*, 2006), which is a web-based bioinformatics tool that allows for storing, visualizing and querying of pathways at different abstraction levels.

This article is organized as follows. In Section 2, the PF model is presented with a formal discussion of the pathway prediction problem. Section 3 elaborates on building an index structure to efficiently mine frequent PFTs. In Section 4, we discuss an algorithm for mining frequent PF patterns, which is followed by a discussion of a pattern matching algorithm described in Section 5. Section 6 presents our experimental evaluation framework and the experimental results. In Section 7, we conclude and discuss future work.

2 SYSTEM AND METHODS

2.1 Functional model of pathways

We first translate a pathway into a graph of enzymes as nodes, where the enzymes of consecutive reactions interact indirectly through shared products and substrates. Figure 3 depicts the enzyme graph for the pathway of Figure 1 with GO annotations of the enzymes. Next, we replace each enzyme with its 'most specific' annotations from GO to obtain a *PFT* for the pathway (Fig. 2). Note that, due to the true-path rule on the hierarchical organization of GO concepts (e.g. Fig. 7), a given PFT can be turned into a 'more general' PFT by replacing any annotation with any of its ancestors. Therefore, a pathway can have multiple functionality templates depending on the levels in GO hierarchy from which the annotations are selected. As an example, in the original PFT of Figure 2, the branching nodes that follow the first step, *FMN Binding*, can be replaced with their immediate parents. Similarly, the first and the last steps can be replaced with their ancestors to get the PFT in Figure 4.

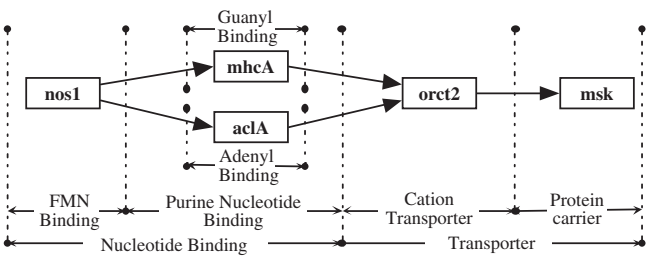

Fig. 3. Enzyme graph with GO functionality annotations.

2.2 Problem definition

Given a set of organism-specific versions of a pathway, we would like to computationally infer pathway fragments in another organism's metabolic network for which the given pathway has not yet been characterized. We give an example.

Example 1. Consider the enzyme graphs of sample pathways $P1$ through $P4$ in Figure 5 that are different versions of a given pathway P in four different organisms. Note that all the enzymes are different, and the four enzyme-only pathways graphs show no similarity to each other. Suppose (i) we have a simple ontology of functionality concepts provided in Figure 7, (ii) the *true-path rule* of GO holds in our sample annotation ontology and (iii) the graphs in Figure 6 constitute the functionality domain representations of P per organism. Then, one can locate instances of the PF pattern P_f depicted in Figure 8a in the functional views of pathways P_1, P_3, P_4. That is, P_f appears in P_1 (by replacing o with g), P_3 (by replacing k with d, and l with h, and i with g) and P_4 (by replacing m with h, and j with g, and h with d) where

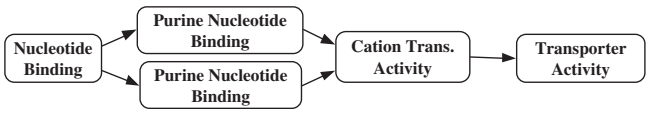

Fig. 4. Alternative functionality template

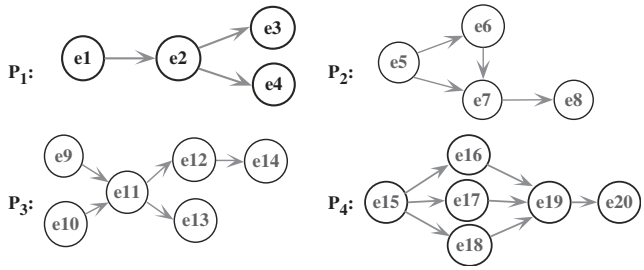

Fig. 5. Organism-specific enzyme-only versions of a single reference pathway.

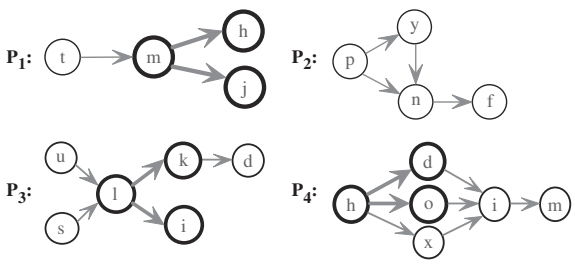

Fig. 6. PFT representations of pathways in Figure 5.

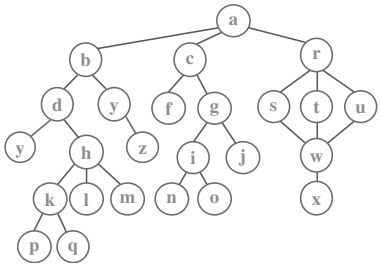

Fig. 7. A sample ontology.

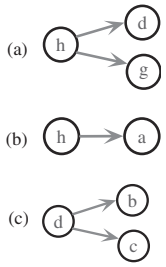

Fig. 8. Frequent PF patterns.

all node replacements are done using the true-path rule. Note that a typical graph mining process on the actual four sample pathways of Figure 5 will not locate any pattern as none of the graphs explicitly contain the nodes of the pattern. Only by (a) moving from the traditional domain of processes-metabolites of pathways into the functionality domain, and (b) utilizing the generalization/specification relationships embedded in a hierarchical organization of the functionality concepts (Fig. 7), we are able to locate the pattern P_f in Figure 8a as well as the others shown in Figure 8b and c.

Given a set of PFTs for organism-specific versions of a pathway P_R, we want to (step 1) find *PF patterns* (which are subgraphs within PFTs) that are common in most of the organisms, and (step 2) locate the discovered patterns in a given organism's functional metabolic network, for which the given pathway has not yet been characterized, and infer the pathway P_k of P_R for the given organism O_k.

Given a set H of PFTs, a subgraph of a PFT in H is called a *PF pattern* if it is contained in *sufficiently* many number of PFTs in H.

Definition. *Support of a pattern F, denoted as support(F), with respect to a set S of PFTs is the number of PFTs that contain F in S.*

As an example, the support of the pattern in Figure 8a in the four PFTs in Figure 6 is 3.

Definition. *(Closure of a PF Pattern). Given a PF pattern F, the closure F* of F is the set of all PF patterns that can be obtained by (i) replacing any node in F with any of its ancestors in GO, and/or (ii) deleting any node and its incident edges from F.*

Example 2. Given the PFT in Figure 8a as a PF pattern F, the PFTs in Figure 8b and c are both in the closure F^* of F. Note that $F_1 = F_2$ iff $F_1^* = F_2^*$.

Also, we require the discovered pattern set to be *minimal* and *complete*. For a set of patterns to be minimal, no pattern in the set should be included in (i.e. be a subgraph of) another pattern in the set, or be included in the closure of a pattern in the set. Furthermore, completeness requires a pattern set to include all possible PF patterns that satisfy the specified threshold requirements.

Definition. *(Minimality of a PF Pattern Set): A set R of PF patterns is minimal if, for any pair of patterns F_i, F_j in R, $\{F_k \mid F_k \in F_j^*$ and F_i is a subgraph of $F_k\} = \emptyset$.*

Example 3. The pattern set that is shown in Figure 8 is not minimal as the closure of the pattern of Figure 8a includes the pattern of Figure 8c.

Definition. *(Completeness of a PF Pattern Set): Let S be a set of PFTs, and $R(\varepsilon)$ be a set of patterns over the PFTs in S with support $\geq \varepsilon$ where ε, $0 < \varepsilon \leq 1$, is the support threshold. Then a set of patterns R' is complete with respect to S with support threshold ε if R' contains $R(\varepsilon)$.*

Next we specify the two steps of pathway prediction.

Step 1: Finding Frequent PF Patterns in a PFT set. Given (a) a pathway P_R, (b) the set $PO = \{(P_1, O_1), (P_2, O_2), \ldots, (P_n, O_n)\}$ of PFT-organism pairs such that P_i is the most-specific PFT for the organism-specific version of P_R in organism O_i and (c) a threshold ε, the frequent PF pattern mining problem is to find the PF pattern set $F(P_R, PO, \varepsilon)$.

Once the set $F(P_R, PO, \varepsilon)$ of frequent PF patterns is identified, next we search for the patterns in the *functional PF network* M_k of the target organism O_k, where M_k consists of all known reactions in organism O_k. The subgraphs of M_k matching the patterns of $F(P_R, PO, \varepsilon)$ are mapped to the actual enzymes, and predicted as fragments of pathway P_R in organism O_k.

Step 2: Predicting Pathways from Matched Frequent Patterns. Given the functional PF network M_k of an organism O_k, and the set $F(P_R, PO, \varepsilon)$ of PF patterns extracted (with respect to ε) from the organism-specific versions of a pathway P_R, the pathway prediction problem is to (i) search for matches in M_k to patterns of $F(P_R, PO, \varepsilon)$, and, out of the matched patterns in M_k, (ii) identify a subgraph G of M_k such that, when mapped back into the traditional pathway domain, G is the predicted pathway P_k of P_R in organism O_k.

3 ALGORITHM

Given a pathway P_R and a set $PO = \{(P_1, O_1), (P_2, O_2), \ldots, (P_n, O_n)\}$ of PFT-organism pairs such that P_i is the PFT for the organism-specific version of pathway P_R in organism O_i, we first construct canonical string representations for each P_i in PO. Next, each constructed string is inserted into *Generalized Suffix Graph (GSG)*. Then, we mine for frequent PF patterns on the GSG. Finally, we search for occurrences of the discovered PF patterns in the metabolic network of the given organism for which the organism-specific-version of P_R is not known.

(i) *Restructuring the GO and pathways:* in order to simplify the presentation, and decrease the level of the problem complexity, we transform all pathways and the GO into trees by node and edge replications. (see Section 2 in Supplementary Material.)

(ii) *Canonical string representation:* in order to facilitate the representation of all possible PFTs in a compact manner, we introduce string-based *canonical representation schemes* for both individual enzymes and PFTs. (see Section 3 in Supplementary Material.)

(iii) *Generalized suffix graph:* we extend the generalized suffix tree (GST) data structure (Gusfield, 1997) to represent multiple PFTs in a single structure, and to efficiently locate frequent PFT patterns. Due to use of non-tree *auxiliary edges*, we refer to the extended GST as

generalized suffix graph (GSG). (see Section 4 in Supplementary Material.)

3.1 Mining frequent PF patterns on a GSG

The frequent PF pattern mining task has two steps. Step 1 'grows' multiple subgraphs R of the GSG G, each via a set C of 'candidate edges'. Step 2 converts each R into a frequent PF pattern. Step 1 consists of two iterative subtasks: (a) identify a candidate (expansion) edge set C from the GSG G, and (b) using C, expand or initiate the subgraph R, for eventual frequent pattern identification. More specifically, given (i) a GSG $G(r, V, E)$ with root r, node set V and edge set E, (ii) a subgraph R of G (originally contains only root r), (iii) a candidate (expansion) edge set C, $C \subseteq E$, to visit, (iv) a set of V of already visited edges, $V \subset E$, $V \cap C = \emptyset$ and (v) a support threshold ε, the frequent pattern mining algorithm returns (step 1) a set of subgraphs R of G, which are then converted back (step 2) to tree-structured patterns. (see Section 5 in Supplementary Material for more details.)

Definition. *(Subgraph of a GSG). Consider a GSG $G(r, V, E)$, and a connected graph $R(r', V', E')$, where V, and V' are the node sets, E, and E' are the edge sets, r, and r' are the root nodes of G and R, respectively. Then, R is a subgraph of G if (a) $r = r'$, (b) $V' \subseteq V$ and (c) $E' \subseteq E$.*

(i) *Enumerating candidates:* given an edge E in a GSG G, the set of candidate edges that can be used to expand R through E contains those edges which follow E in G. In order not to consider the same edge as a candidate more than once, a set of previously visited candidates, V, is also kept track of.

(ii) *Expanding subgraphs:* given a candidate edge set C and a subgraph R to be expanded, an edge E is chosen from C, and R is expanded with E to construct a larger subgraph R'. Then, the support of R' is computed directly from the suffix sets of the edges in R. Similar to the backtracking mechanism (Zaki, 2005) during the expansion of subgraph R, whenever an edge E is chosen from the candidate edge set C, E is removed from C, and inserted into the visited edge set V. This step is taken in order to prune the duplicate subgraphs that can result from the consideration of candidate edges in different orders.

Example 4. Consider the GSG in Figure 9 that contains PFT string suffixes S_1 = be \$ c \$ r, S_2 = be \$ {cg \$ r}{rt}, S_3 = b \$ {c}{rt}. PFT graph representations P_1, P_2 and P_3 of PFT string suffixes S_1, S_2 and S_3 are given in Figure 10. With support threshold as 2, the algorithm produces two subGSGs (Fig. 11) that, after conversion, represent two distinct PF patterns.

Given a set P of GSG subgraph R's computed by step 1, in step 2, each subgraph G is converted to a frequent PF pattern string by traversing their edges recursively in depth-first order, and appending the edge labels to the constructed pattern string.

3.2 Pathway fragment prediction

Once the frequent PF pattern set $F(P_R)$ for pathway P_R is computed, the metabolic PF network M of a target organism O is searched for occurrences of PF patterns in $F(P_R)$. As part of preprocessing, on each metabolic network, enzyme nodes with multiple GO concept annotations are replicated.

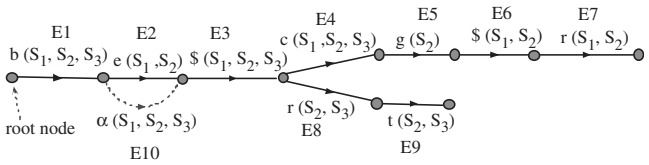

Fig. 9. Computing frequent patterns on a GSG.

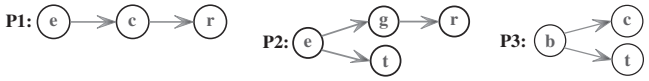

Fig. 10. PFT graph representations of the input PFT strings.

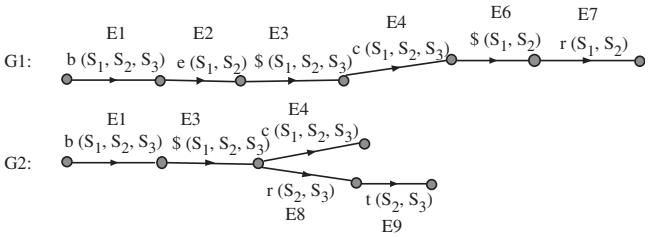

Fig. 11. GSG subgraphs representing PF patterns.

When a metabolic network fragment is more specific than a PF pattern, by, applying the true-path rule, a match occurs. As an example, PF pattern P = 'ab \$ de' matches to the more-specific metabolic network fragment 'abc \$ def'. However, if a pattern is more specific than the corresponding metabolic pathway fragment, there is no match. We choose not to allow matches to fragments that are more general than the pattern because, given a pattern, the number of matching candidate pathway fragments can easily explode, which leads to more false positives than true positives. In Section 6.4, for inferring an unknown pathway, we also perform a more relaxed matching, and discuss possible uses of external genomic information to eliminate or strengthen some of the alternatives in the result set. Finally, in the future work section, we discuss possible integration of taxonomy-based semantic similarity measures to allow 'approximate' pattern matches.

After each PF pattern in $F(P_R)$ is searched in the target network, the matched nodes and edges are added as the sum of the matching scores of all patterns which match to that edge or node. The matching score of a pattern is an aggregate of two measures:

(i) *Selectivity (Sel):* given a pattern P, the metabolic network of each organism (excluding the target organism) is searched for P. Then, in the metabolic network M, the total number of nodes and edges that are included in at least one match to P in M is recorded. Next, the fraction F of matched nodes to the size (number of nodes) of the network is computed. Finally, F is normalized by the total number of nodes in P, and recorded as the

selectivity of the pattern P in M. This process is repeated for each metabolic network, and the final selectivity of the pattern is computed as the average of its selectivity values over all metabolic networks.

(ii) *Support of a PF Pattern (Sup):* PF patterns which have higher support among the organisms for which an instance of P_R is known are indicative of the existence of an instance of P_R in the searched organism.

Final confidence of a candidate edge/node is computed by the aggregation of the selectivity and the support measures. That is, the confidence of a node or edge x in M to match to a node/edge in a pattern P with selectivity *Sel* and support *Sup* is:

$$Conf(x,M,P) = w_{sel}*Sel + w_{sup}*Sup$$

where w_{sel}, *and* w_{sup} are weights that are experimentally determined according to the accuracy that the measure provides when applied alone (i.e. independent of the other measures). (see Sections 6 and 7.1 in Supplementary Material for more details).

4 EXPERIMENTS

4.1 Data set

The experiments were performed on a set of pathways that were downloaded from KEGG (Kanehisa *et al.*, 2004) pathways database (as of June 2006). We randomly picked 50 pathways of 30 bacterial organisms, which provided us with 1500 organism-specific pathways as the core data set. Common molecules (e.g. H_2O) that appear in at least half of the pathways were eliminated from the data set. Biological processes were assumed to always proceed from substrates to products, and reversible processes are ignored. Metabolic networks of our chosen organisms were constructed from all known enzymatic reactions (processes) in these organisms. As a result, each metabolic network consisted of, on the average, 402 enzymes and 9695 enzyme relationships. In the PF domain, the average number of nodes was 1037, and the average number of edges per metabolic network was 51 713.

The original GO molecular function hierarchy (downloaded in September 2006) included 7459 GO concepts organized in a hierarchy of 15 levels with 8707 hierarchical relationships among the concepts. After applying the transformation described in Section 2.3, the transformed version of GO included 11 675 terms.

4.2 Metrics

In order to evaluate the accuracy of our pathway predictions, precision/recall measurements were employed. The prediction accuracies of enzymes and enzyme relationships were assessed separately through the following measures.

- *Enzyme Precision* is the fraction of correctly predicted enzymes in the inferred pathway.
- *Enzyme Recall* is the ratio of correctly predicted enzymes in the inferred pathway to the total number of enzymes in the actual pathway for a given organism.
- *Enzyme Relationship Precision* is the fraction of correctly predicted edges among the enzyme nodes in an inferred pathway instance for a given organism.
- *Enzyme Relationship Recall* is the ratio of correctly predicted relationships between enzymes to all known enzyme relationships in the actual pathway for a given organism.

4.3 Results

The main goal of this study is to accurately predict the organism-specific version of a pathway P for a given organism for which P is not known yet. Hence, in the first experiment, we evaluated the accuracy of the overall system on the known data using the *leave-one-out* strategy as follows: for each reference pathway P, we pick a target organism O for which P is to be predicted. PF patterns are mined from the known instances of P in organisms other than O from our chosen set of organisms. Then, the generated patterns are searched in the metabolic network of O to predict a partial instance of P in O. This procedure is repeated 30 times for each pathway, where, at each iteration, a distinct organism is selected as the target organism. Overall, 1500 distinct pathway inference tasks were run. The overall accuracy was computed as the average of all runs. Figure 12 plots the overall precision/recall values at different GO specificity levels.

Observation 1: *by switching to the GO functionality domain, and using PF patterns, test pathways are successfully predicted with the maximum precision of 88% for enzymes, and 87% for enzyme relationships at specificity levels of 8 and above. The maximum prediction recall is 74% for enzymes, and 65% for enzyme relationships, both at specificity level 3. High accuracy shows that functionalities of genomic agents that are involved in different organism-specific versions of a pathway are conserved substantially among organisms.*

Since the deepest level that contains a GO term annotating an enzyme in our data set is 14, the specificity level of 14 in Figure 12 corresponds to the case where the true-path rule of GO is ignored. Hence, the accuracy at specificity level 14 is utilized for comparison purposes against those cases where the true-path rule is employed during pattern discovery.

Observation 2: *taking the hierarchical organization of functionality terms into account increases the recall values of both enzyme and enzyme relationships by 5% while at lower specificity levels the precision decreases sharply. If high recall is the primary goal, specificity level should be set to lower levels to take advantage of GO hierarchy at maximum. Otherwise, for*

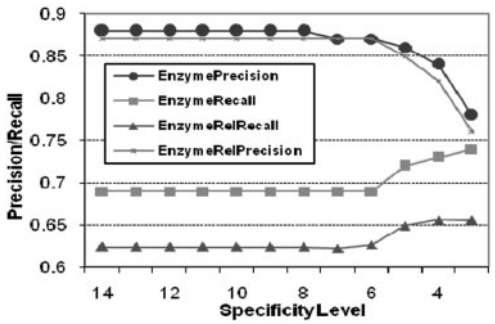

Fig. 12. Overall precision/recall at different specificity levels.

high precision, 5 is the lowest specificity level that provides a balanced rate of precision and recall.

Increase in the recall value as the specificity level decreases is expected since, as illustrated in the running example of Section 2, using more general functionality terms leads to patterns that match to larger set of enzymes with different, but closely related, functionalities. As for precision, in the best case, the precision can be the same as the case where the GO hierarchy information is not considered during frequent pattern discovery.

Figure 12 shows that considering the GO hierarchy has the most significant impact on GO concepts at levels 6 and above in the GO tree. This result correlates well with the database statistics that the average level of GO concepts that annotate at least one enzyme of a pathway is 6.72.

Observation 3: *precision and recall values remain almost the same until specificity level 7, where an increase in recall starts, and becomes flat at specificity level 3.*

Another experiment was conducted to study the contribution of using the GO hierarchy for enriching the set of PF patterns common to majority of organisms. For first run, we ignored the true-path rule of GO, and counted the number of created patterns. We then repeated this process 10 times, and at each run *i*, we replaced the nodes in the most specific PFTs with their ancestors that are *i*-level above those in the first run in the GO hierarchy. We refer the value *i* as the *relaxation level (RL)*. At each relaxation level, we computed the average pattern support. Figure 13 shows the average pattern support values at each relaxation level where the relaxation level 0 represents the most specific PFT of a pathway. The experiment was performed with min threshold of 10.

Observation 4: *For pattern creation, going up in the hierarchy increases the average pattern support until relaxation level 5, after which, a gradual decrease occurs at levels 6 and 8.*

We explain the average pattern support decrease after relaxation levels 4 and 6 based on the distributions of patterns at different sizes where size is expressed in terms of the number of nodes in the pattern. Figure 14 shows distributions for patterns of different sizes, where size-10+ refers to the class of patterns with size 10 or more. Larger patterns tend to have smaller support. Therefore, whenever the percentage of smaller size patterns increases within an RL, the average pattern support usually increases as the majority of patterns in that set have large support values due to their small sizes. For instance, in Figure 13, there is a steep increase in average pattern support from RL 3 to RL 4. This is mainly because RL 4 has a fewer number of size-2 patterns, and, instead, has more size-3 and size-4 patterns in comparison to RL 3 (Fig. 14). RL 3 has more size-10+ patterns, but since the number of such patterns is very small in the overall set of RL 3's patterns, its effect is minimal. As for the decrease in average pattern support from RL 4 to RL 5, the percentages of size-3, -4 and -10+ patterns are higher in RL 5 compared to RL 4, which leads to a gradual decrease in average pattern support.

In an additional experiment, we further studied how the predicted precision and recall change at different pattern

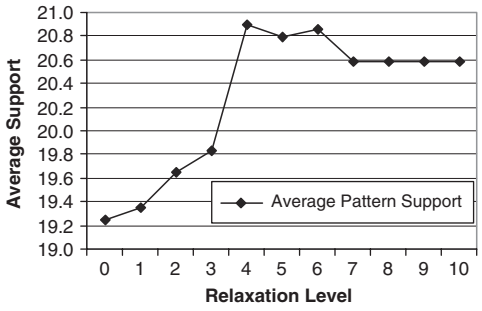

Fig. 13. Average support of patterns at different relaxation levels.

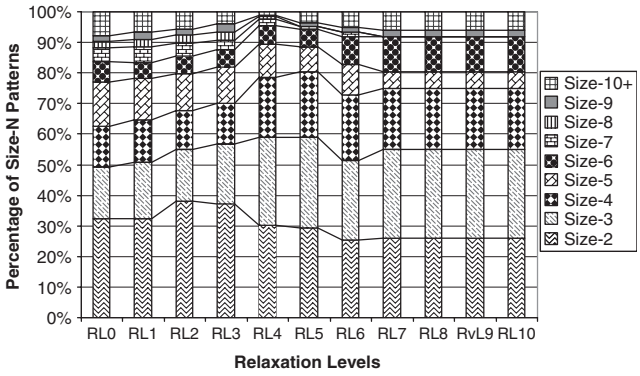

Fig. 14. Percentage distribution of patterns at distinct relaxation levels.

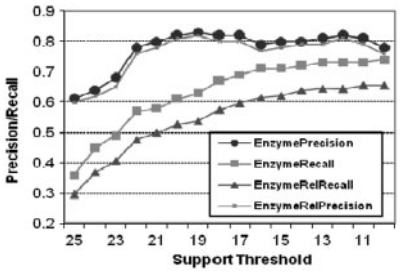

Fig. 15. Overall precision/recall at different threshold levels.

thresholds. We also counted the numbers and the average sizes of patterns produced at each threshold setting. Figure 15 plots the precision/recall against the support threshold at specificity level 3.

Observation 5: *The recall increases as the minimum support threshold gets smaller. And, the precision peaks at support threshold 19, and starts to decrease gradually at support threshold 18 and 11.*

When the support threshold is high, the number of patterns is usually low (on the average, around two patterns per pathway). Hence, most of the enzymes and enzyme relationships are missed. On the other hand, at low thresholds, the number of patterns increases (around nine per pathway on the average), and, thus, the ratios of discovered enzymes and enzyme relations also increase. Nevertheless, the precision decreases due to the relatively low quality of patterns as the threshold decreases. For high threshold values, the precision is expected

to be higher. However, during the experiments, for high threshold values, no patterns could be generated, which leaves precision/recall as 0. Since we also include cases with no results in precision computation, the overall precision is low for experiments with high support thresholds. (see Section 7.2 in Supplementary Material for results where such cases are excluded).

4.4 Candidate novel pathways

The main goal of this study is to infer novel pathways. To this end, we setup a prototype prediction framework for *Saccharomyces cerevisiae* as follows. We constructed a metabolic network out of all known enzyme genes (i.e. genes with an EC number) where nodes are genes, and an edge is created between any pair of genes based on the substrate-product relationship defined by their EC numbers. In total, there are 1196 genes with at least one EC number.

Here, we present candidate novel pathway fragments for two pathways that are not yet known for *S.cerevisiae*. Figure 16 shows the predicted pathway fragment for *Biosynthesis of siderophore group nonribosomal peptides* where round rectangles represent genes possibly catalyzing a process, and directed solid lines represent substrate/product-based biochemical relationships between gene pairs. Undirected dashed lines are not part of the pathway, and are shown just to facilitate referring to gene pairs on the pathway graph. Since the predicted pathway of Figure 16 is not known yet, in order to independently assess the correctness of the prediction, we looked up gene expression experiments available in the literature. Mizuguchi *et al.* (2004) showed that Pearson correlation values for the expression values of gene pairs represented by edges #1, #3 and #5 are very high (i.e. greater than 0.8, which is a commonly accepted threshold). In addition, Van Attikum *et al.* (2004) and Cullen *et al.* (2004) reported that expression values for gene pairs #2, #4, #5 and #6 are highly correlated (Pearson Value > 0.8). This constitutes an independent verification of our predicted pathway fragments in Figure 16.

As a second experiment, we attempted to predict another unknown pathway, *2,4-dichlorobenzoate degradation* for *S.cerevisiae*. First, we performed pattern matching as described in Section 5, but our prototype did not return any results. Then, we relaxed pattern matching by replacing each node in the pattern with its ancestor. However, the relaxed pattern matching led to multiple candidate matches with the same confidence score. In Figure 17, all linear paths of three genes from *TES1* or *ACH1* to *BNA1* or *BNA2* represent alternative predictions. In order to eliminate some of the alternatives, we first searched for transcription factors that are known to be common regulators of mRNA expressions for each pair of genes. The names in dotted rectangles attached to edges are shared transcription factors for the associated gene pairs (Teixeira *et al.*, 2006). Due to the lack of common transcription factors that support the prediction, alternative paths that start from *TES1* are removed. The remaining five genes (with bold border lines) and the numbered relationships are left, and may take part in 2,4-dichlorobenzoate degradation pathway of *S.cerevisiae*. On this remaining set, we further searched for gene expression values, and found that the gene pairs connected by edges #1, #2, #4, #6 (Cullen *et al.*, 2004), #3, #5

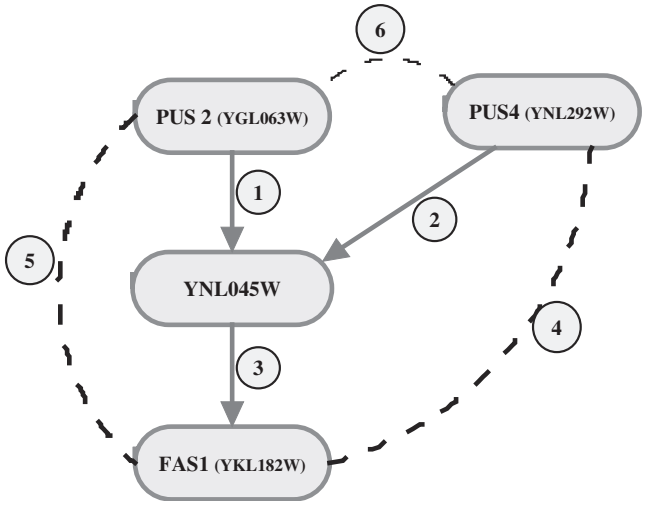

Fig. 16. Predicted *Biosynthesis of siderophore group non-ribosomal peptide* pathway.

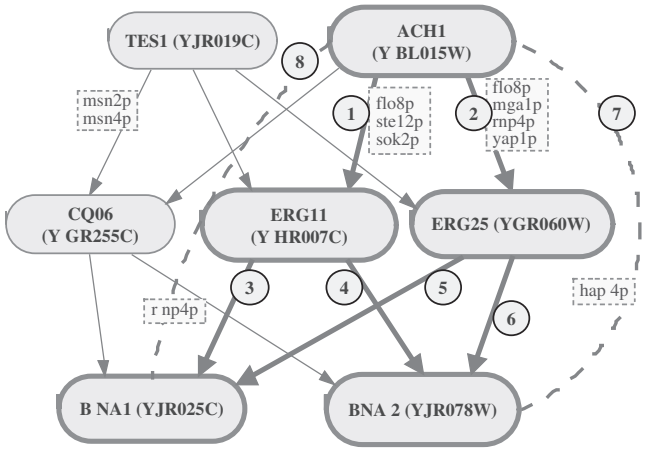

Fig. 17. Predicted 2,4-*dichlorobenzoate degradation* pathway.

(Derisi *et al.*, 1997), #7 and #8 (Wyrick *et al.*, 1999) all have well-correlated expression patterns. Therefore, this constitutes an independent verification that these five genes and the dark-colored solid edges may have a role in this pathway.

5 DISCUSSION

We have presented a pathway inference framework based on the functional annotations of enzymes participating in a pathway. Given a pathway P, we first create a pathway functionality template for each known organism-specific version of the pathway. Next, using a generalized suffix graph, frequent pathway functionality template patterns are discovered. Finally, discovered patterns are searched in the metabolic network of the organism for which P will be predicted. Matching fragments are evaluated based on the selectivity and the support of the patterns.

As part of future work, we are planning to study two distinct directions for approximate pattern matching. First, we would like to allow matches to patterns where matched fragments are more general than patterns. Taxonomy-based semantic

similarity measures (Lin 1998; Lord *et al*., 2003; Resnik, 1999) can be employed to judge the similarity between a pattern and an approximately matching metabolic network fragment. Second, due to the incomplete nature of biological data, some metabolic networks may have missing relationships which can prevent an exact match to a given pattern. One can develop a pattern matching scheme that can tolerate missing edges to some extent under certain constraints. In order to avoid/minimize the introduction of false positives into the predictions, formally defining and evaluating constraints under which such missing edges would be tolerated is a promising research direction.

In addition, employing statistical machine-learning techniques such as SVM (Vapnik, 1995) by building kernels based on external genomic information (e.g. gene expression data and co-localization) can provide an alternative assessment, and we can then choose the most promising predictions when there exist multiple candidates.

Finally, exploring the effect of taxonomic distance between a predicted organism and those organisms whose pathways are used for training (creating patterns) on accuracy is an interesting future direction. (see Section 8 in Supplementary Material for more discussion.)

ACKNOWLEDGEMENTS

This research is supported in part by the NSF award DBI-0218061, a grant from the Charles B. Wang Foundation, and Microsoft equipment grant.

Conflict of Interest: none declared.

REFERENCES

Bang,J.W. *et al*. (2003) Two-stage Bayesian networks for metabolic network prediction. In *Proceedings of the Workshop on Qualitative and Model-Based Reasoning in Biomedicine, 9th Conference on Artificial Intelligence in Medicine*.

Bono,H. *et al*. (1998) Reconstruction of amino acid biosynthesis pathways from the complete genome sequence. *Genome Res*., **8**, 203–210.

Cakmak,A. and Ozsoyoglu,G. (2007) Mining biological networks for unknown pathways. *Technical report* (*full version*). Available at http://cakmak.case.edu/PFT.

Cullen,P.J. *et al*. (2004) A signaling mucin at the head of the Cdc42- and MAPK-dependent filamentous growth pathway in yeast. *Genes Dev*., **18**, 1695–1708.

Dandekar,T. and Sauerborn,R. (2002) Comparative genome analysis and pathway reconstruction. *Pharmacogenomics*, **3**, 245–256.

Derisi,J.L. *et al*. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

Gene Ontology Consortium (2004) The GO database and informatics resource. *Nucleic Acids Res*., **32**, D258–D261.

Green,M. and Karp,P. (2004) A Bayesian method for identifying missing enzymes in predicted metabolic pathway databases. *BMC Bioinformatics*, **9**, 76.

Gusfield,D. (1997) *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, New York, NY, USA.

Hu,H. *et al*. (2005) Mining coherent dense subgraphs across massive biological networks for functional discovery. In *Proceedings of ISMB*, Ann Arbor, MI.

Huan,J. *et al*. (2003) Efficient mining of frequent subgraphs in the presence of isomorphism. In *Proceedings of International Conference on Data Mining*. IEEE Computer Society, Washington DC, p. 549.

Huan,J. *et al*. (2004) SPIN: mining maximal frequent subgraphs from graph databases. In *Proceedings of KDD*.

Jansen,R. *et al*. (2003) A Bayesian networks approach for predicting protein-protein interactions from genomic data. *Science*, **17**, 449–453.

Joshi-Tope,G. *et al*. (2005) Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res*., **33**, D428–D432.

Kanehisa,M. *et al*. (2004) The KEGG resource for deciphering the genome. *Nucleic Acids Res*., **32**, D277–D280.

Kelley,B.P. *et al*. (2004) PathBLAST: a tool for alignment of protein interaction networks. *Nucleic Acids Res*., **32**, W83–W88.

Kelley,P. *et al*. (2003) Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proc. Natl Acad. Sci*., pp. 11394–11399, USA.

Keseler,I.M. *et al*. (2005) Eco-Cyc: a comprehensive database resource for Escherichia coli. *Nucleic Acids Res*., **33**, D334–D337.

Kharchenko,P. *et al*. (2004) Filling gaps in a metabolic network using expression information. *Bioinformatics*, **20**, 449–453.

Koyuturk,M. *et al*. (2006) Detecting conserved interaction patterns in biological networks. *J. Comput. Biol*., **13**, 1299–1322.

Kuramochi,M. and Karypis,G. (2001) Frequent subgraph discovery. In *Proceedings of IEEE International Conference on Data Engineering*. IEEE Computer Society, Washington, DC, pp. 313–320.

Lin,D. (1998) An information-theoretic definition of similarity. *International Conference on Machine Learning*. Morgan Kaufmann, San Fransisco, pp. 296–304.

Lord,W. *et al*. (2003) Investigating semantic similarity measures across the Gene Ontology. *Bioinformatics*, **19**, 1275–1283.

Mizuguchi,G. *et al*. (2004) ATP-driven exchange of histone H2AZ variant catalyzed by SWR1 chromatin remodeling complex. *Science*, **303**, 343–348.

Myers,C.L. *et al*. (2005) Discovery of biological networks from diverse functional genomic data. *Genome Biol*., **6**, R114.

Osterman,A. and Overbeek,R. (2003) Missing genes in metabolic pathways: a comparative genomics approach. *Curr. Opin. Chem. Biol*., **7**, 238–251.

Ozsoyoglu,M. *et al*. (2006) Genomic pathways database and biological data management. *Animal Genet*., **37** (Suppl. 1), 41–47.

Paley,S. and Karp,P.D. (2002) Evaluation of computational metabolic-pathway predictions for Helicobacter pylori. *Bioinformatics*, **18**, 715–724.

Pinter,R. *et al*. (2005) Alignment of metabolic pathways. *Bioinformatics*, **21**, 3401–3408.

Pireddu,L. *et al*. (2005) Pathway analyst: automated metabolic Pathway prediction. In *Proceedings of the IEEE Symposium CIBCB*.

Romero,P. *et al*. (2005) Computational prediction of human metabolic pathways from the complete genome. *Genome Biol*., **6**, R2.

Resnik,P. (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res*., **11**, 95–130.

Sharan,R. *et al*. (2005) Conserved patterns of protein interaction in multiple species. *Proc. Natl Acad. Sci. USA*, **102**, 1974–1979.

Shlomi,T. *et al*. (2006) QPath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, **7**, 199.

Teixeira,M. *et al*. (2006) The YEASTRACT: a tool for the analysis of transcription regulatory associations in S. cerevisiae. *Nucleic Acids Res*., **34**, D446–D451.

Tohsato,Y. *et al*. (2000) A multiple alignment algorithm for metabolic pathway analysis using enzyme hierarchy. *Intelligent Systems for Molecular Biology (Supplement of Bioinformatics)*, 376–383.

Tu,Z. *et al*. (2006) An integrative approach for causal gene identification and gene regulatory pathway inference. *Bioinformatics*, **22**, e489–e496.

Van Attikum,H. *et al*. (2004) Recruitment of the INO80 complex by H2A phosphorylation links ATP-dependent chromatin remodeling with DNA double-strand break repair. *Cell*, **119**, 777–788.

Vapnik,V. (1995) *The Nature of Statistical Learning Theory*. Springer-Verlag, NY.

Wyrick,J. *et al*. (1999) Chromosomal landscape of nucleosome-dependent gene expression and silencing in yeast. *Nature*, **402**, 418–421.

Yamanishi,Y. *et al*. (2005) Supervised enzyme network inference from the integration of genomic data and chemical information. *Intelligent Systems for Molecular Biology* (*Supplement of Bioinformatics*), pp. 468–477.

Yamanishi,Y. *et al*. (2007) Prediction of missing enzyme genes in a bacterial metabolic network. *FEBS J*., **274**, 2262–2273.

Yan,X. and Han,J. (2002) gSpan: graph-based substructure pattern mining. In *International Conference on Data Mining*, Technical Report.

Yan,X. *et al*. (2005) Substructure similarity search in graph database. *Substructure Similarity search in Graph Database*, SIGMOD, June 14–16, 2005.

Ye,Y. *et al*. (2005) Automatic detection of subsystem/pathway variants in genome analysis. *Bioinformatics*, (Suppl. 1), i478–i486.

Zaki,M. (2005) Efficiently mining frequent trees in a forest: algorithms and applications. In *IEEE Transactions on Knowledge and Data Engineerings*.