Effective Identification of Conserved Pathways in Biological Networks Using Hidden Markov Models

Xiaoning Qian[1], **Byung-Jun Yoon**[2]*

[1] Department of Computer Science and Engineering, University of South Florida, Tampa, Florida, United States of America, [2] Department of Electrical and Computer Engineering, Texas A&M University, College Station, Texas, United States of America

Abstract

Background: The advent of various high-throughput experimental techniques for measuring molecular interactions has enabled the systematic study of biological interactions on a global scale. Since biological processes are carried out by elaborate collaborations of numerous molecules that give rise to a complex network of molecular interactions, comparative analysis of these biological networks can bring important insights into the functional organization and regulatory mechanisms of biological systems.

Methodology/Principal Findings: In this paper, we present an effective framework for identifying common interaction patterns in the biological networks of different organisms based on hidden Markov models (HMMs). Given two or more networks, our method efficiently finds the top k matching paths in the respective networks, where the matching paths may contain a flexible number of consecutive insertions and deletions.

Conclusions/Significance: Based on several protein-protein interaction (PPI) networks obtained from the Database of Interacting Proteins (DIP) and other public databases, we demonstrate that our method is able to detect biologically significant pathways that are conserved across different organisms. Our algorithm has a polynomial complexity that grows linearly with the size of the aligned paths. This enables the search for very long paths with more than 10 nodes within a few minutes on a desktop computer. The software program that implements this algorithm is available upon request from the authors.

Citation: Qian X, Yoon B-J (2009) Effective Identification of Conserved Pathways in Biological Networks Using Hidden Markov Models. PLoS ONE 4(12): e8070. doi:10.1371/journal.pone.0008070

Editor: Diego Di Bernardo, Fondazione Telethon, Italy

Received September 17, 2009; **Accepted** October 29, 2009; **Published** December 7, 2009

Funding: This work was supported in part by the National Cancer Institute under 2 R25CA090301-06. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: bjyoon@ece.tamu.edu

Introduction

Recent advances in high-throughput experimental techniques for measuring molecular interactions [1–4] have enabled the systematic study of biological interactions on a global scale for an increasing number of organisms [5]. Genome-scale interaction networks provide invaluable resources for investigating the functional organization of cells and understanding their regulatory mechanisms. Biological networks can be conveniently represented as graphs, in which the nodes represent the basic entities in a given network and the edges indicate the interactions between them. Network alignment provides an effective means for comparing the networks of different organisms by aligning these graphs and finding their common substructures. This can facilitate the discovery of conserved functional modules and ultimately help us study their functions and the detailed molecular mechanisms that contribute to these functions. For this reason, there have been growing efforts to develop efficient network alignment algorithms that can effectively detect conserved interaction patterns in various biological networks, including protein-protein interaction (PPI) networks [6–20], metabolic networks [7,12,21], gene regulatory networks [22], and signal transduction networks [23]. It has been demonstrated that network alignment algorithms can detect many known biological pathways and also make statistically significant predictions of novel pathways.

Network alignment can be broadly divided into two categories, namely, *global alignment*, which tries to find the best coherent mapping between nodes in different networks that covers all nodes; and *local alignment*, which simply tries to detect significant common substructures in the given networks. Typically, the global network alignment problem is formulated as a graph matching problem whose goal is to find the optimal alignment that maximizes a global objective function that simultaneously measures the similarity between the constituent nodes and also between their interaction patterns. This optimization problem can be solved by a number of techniques, such as integer programming [24], spectral clustering [16,17], and message passing [20]. To cope with the high complexity of the global alignment problem, many algorithms incorporate heuristic techniques, such as greedy extension of high scoring subnetwork alignments and progressive construction of multiple network alignments [9,15,17,19].

There are also many local network alignment algorithms, where examples include PathBLAST [6], NetworkBLAST [10], QPath [11], PathMatch and GraphMatch [12], just to name a few. These algorithms can effectively find conserved substructures with relatively small sizes, but many of them suffer from high computational complexity that makes it difficult to find larger substructures. Furthermore, many algorithms have limited flexibility of handling node insertions and deletions and/or rely on randomized heuristics that may not necessarily yield optimal results. In [18], we introduced an effective framework for local network alignment based on hidden Markov models (HMMs) that can effectively overcome many of these issues. The HMM framework can naturally integrate both the "node similarity" (typically estimated by sequence similarity) and the "interaction reliability" into the scoring scheme for comparing aligned paths, and it can deal with a large class of path isomorphism. Based on the HMM-based framework, we devised an efficient algorithm that can find the optimal homologous pathway for a given query pathway in a PPI network, whose complexity is linear with respect to the network size and the query length, making it applicable to search for long pathways. It was demonstrated that the algorithm can accurately detect homologous pathways that are biologically significant. However, the algorithm in [18] was mainly developed for *querying* pathways in a target network, hence it cannot be directly used for local alignment of general networks.

In this paper, we extend the HMM-based framework proposed in [18] to make it applicable for local alignment of general biological networks. Especially, we focus on the problem of identifying similar pathways that are conserved across two or more biological networks. Based on HMMs, we propose a general probabilistic framework for scoring pathway alignments and present an efficient search algorithm that can find the top k alignments of homologous pathways with the highest scores. The algorithm has polynomial complexity which increases linearly with the length of the aligned pathways as well as the number of interactions in each network. The aligned pathways in a predicted alignment may contain flexible number of consecutive insertions and/or deletions. By combining the high-scoring pathway alignments that overlap with another, we can also detect conserved subnetworks with a general structure. Note that the algorithm can be also used for network querying, by designating one network as the query and another network as the target network.

Methods

In this section, we present an algorithm for solving the local network alignment problem based on HMMs. For simplicity, we first focus on the problem of aligning two networks, which can be formally defined as follows: Given two biological networks $\mathcal{G}_1$ and $\mathcal{G}_2$ and a specified length L, find the most similar pair $(\mathbf{p},\mathbf{q})$ of linear paths, where $\mathbf{p}$ belongs to the network $\mathcal{G}_1$ and $\mathbf{q}$ belongs to $\mathcal{G}_2$, and each of them have L nodes. As we show later, the pairwise network alignment algorithm can be easily extended for aligning multiple networks in a straightforward manner.

Pairwise Network Alignment

Let $\mathcal{G}_1=(\mathcal{U},\mathcal{D})$ be a graph representing a biological network. We assume that $\mathcal{G}_1$ has a set $\mathcal{U}=\{u_1,u_2,\ldots,u_{N_1}\}$ of N_1 nodes, representing the entities in the network, and a set $\mathcal{D}=\{d_{ij}\}$ of M_1 edges, where d_{ij} represents the interaction (binding or regulation) between u_i and u_j. When the network $\mathcal{G}_1$ is undirected, we assume that both d_{ij} and d_{ji} are present in the set $\mathcal{D}$ for simplicity. For example, when $\mathcal{G}_1$ represents a PPI network, u_i corresponds to a protein, and the edge between u_i and u_j indicates that these proteins can bind to each other. For a pair (u_i,u_j) of interacting nodes such that $d_{ij}\in\mathcal{D}$, we define their interaction reliability as $w_1(u_i,u_j)$. Similarly, let $\mathcal{G}_2=(\mathcal{V},\mathcal{E})$ be another graph with N_2 nodes and M_2 edges, representing a different biological network. We denote the interaction reliability between two nodes v_i and v_j in the graph $\mathcal{G}_2$ as $w_2(v_i,v_j)$. Finally, we denote the similarity between two nodes $u_i\in\mathcal{G}_1$ and $v_j\in\mathcal{G}_2$ in the respective networks as $h(u_i,v_j)$, which may be derived using the sequence similarity between two biological entities represented by two nodes as in our experiments.

Our goal is to find the best matching pair of paths $\mathbf{p}=p_1p_2\ldots p_L$ $(p_i\in\mathcal{U})$ and $\mathbf{q}=q_1q_2\ldots q_L$ $(q_i\in\mathcal{V})$ in the respective networks that maximizes a predefined pathway alignment score $S(\mathbf{p},\mathbf{q})$. In order to obtain meaningful results, the alignment score $S(\mathbf{p},\mathbf{q})$ should sensibly integrate the similarity score $h(p_i,q_i)$ between aligned nodes p_i and q_i $(1\le i\le L)$, the interaction reliability scores $w_1(p_i,p_{i+1})$ between p_i and p_{i+1} $(1\le i\le L-1)$ and $w_2(q_j,q_{j+1})$ between q_j and q_{j+1} $(1\le j\le L-1)$, and the penalty for any gaps in the alignment.

Figure 1C illustrates an example of an alignment between two similar paths $\mathbf{p}$ and $\mathbf{q}$, where $\mathbf{p}$ belongs to $\mathcal{G}_1$ and $\mathbf{q}$ belongs to $\mathcal{G}_2$ as shown in Fig. 1A. The dashed lines in Fig. 1A that connect two nodes u_i and v_j indicate that there exist significant similarities between the connected nodes. In the example shown in Figure 1C, the optimal alignment that maximizes the alignment score $S(\mathbf{p},\mathbf{q})$ has two gaps at q_3 and p_5. Note that "insertions" and "deletions" are relative terms, and an insertion in $\mathbf{p}$ (e.g., p_5) can be viewed as a deletion in the aligned path $\mathbf{q}$, and similarly, an insertion in $\mathbf{q}$ (e.g., q_3) can be viewed as a deletion in $\mathbf{p}$.

Network Representation by HMM

To define the alignment score $S(\mathbf{p},\mathbf{q})$, we adopt the hidden Markov model (HMM) formalism. We begin by constructing two HMMs based on the network graphs $\mathcal{G}_1$ and $\mathcal{G}_2$. Let us first focus on the construction of HMM for $\mathcal{G}_1$. Each node $u_i\in\mathcal{U}$ in $\mathcal{G}_1$ corresponds to a hidden state in the HMM. For convenience, we represent this hidden state using the same notation u_i. For each edge $d_{ij}\in\mathcal{D}$ in the graph $\mathcal{G}_1$, we add an edge from state u_i to state u_j in the HMM. The resulting HMM has an identical structure as the network graph $\mathcal{G}_1$. The HMM for $\mathcal{G}_2$ can be constructed in a similar way. Figure 2A illustrates the HMMs that correspond to the network graphs shown in Fig. 1A. In order to find the best matching pairs of paths in the given networks, we define the concept of a "virtual" path $\mathbf{s}=s_1s_2\ldots s_L$ that contains L nodes, as shown in Fig. 1B. A node s_i in the virtual path can be viewed as a symbol that is emitted by a pair of hidden states p_i and q_i in the respective HMMs. From this point of view, the two HMMs can be regarded as generative models that *jointly* produce (or "emit") the virtual path $\mathbf{s}$, and the underlying state sequence for $\mathbf{s}$ will be a pair of state sequences $\mathbf{p}$ and $\mathbf{q}$ in the respective HMMs. Therefore, the concept of a virtual path can naturally couple a path in $\mathcal{G}_1$ with another in $\mathcal{G}_2$, providing a convenient framework for identifying conserved pathways in the original biological networks.

The described HMM-based network representation allows us to naturally integrate the interaction reliability scores and the node similarity scores into an effective probabilistic framework. We first define two mappings $\mathbf{f}_1: w_1(u_m,u_n)\mapsto t_1(u_n|u_m)$ and $\mathbf{f}_2: w_2(v_m,v_n)\mapsto t_2(v_n|v_m)$, which convert the interaction reliability scores $w_1(u_m,u_n)$ and $w_2(v_m,v_n)$ between two nodes in $\mathcal{G}_1$ and $\mathcal{G}_2$ to the following transition probabilities

$$P(p_i=u_n|p_{i-1}=u_m)=t_1(u_n|u_m)=\mathbf{f}_1(w_1(u_m,u_n)) \qquad (1)$$

$$P(q_i=v_n|q_{i-1}=v_m)=t_2(v_n|v_m)=\mathbf{f}_2(w_2(v_m,v_n)) \qquad (2)$$

between the corresponding hidden states in the constructed HMMs.

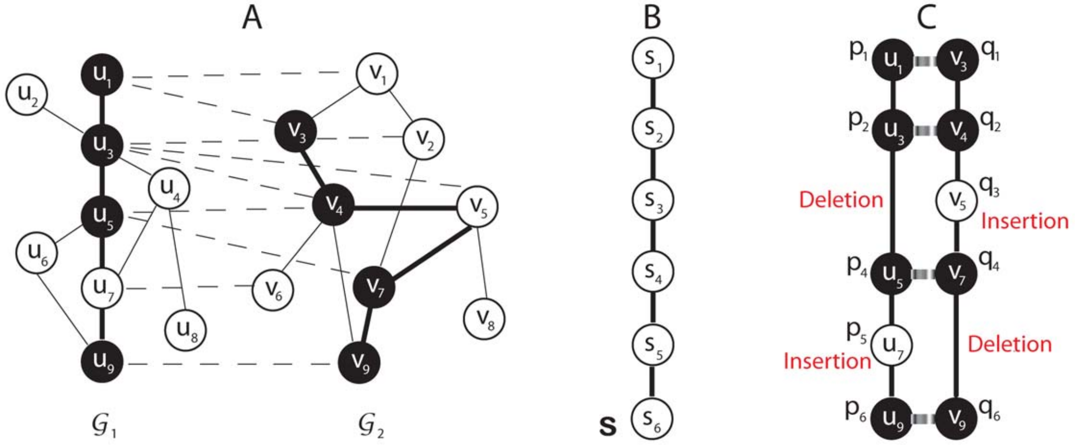

Figure 1. Network representation and alignment. (A) Example of two undirected biological networks $\mathcal{G}_1$ and $\mathcal{G}_2$. (B) A virtual path $\mathbf{s}$ that corresponds to the alignment of best matching paths. (C) The top-scoring alignment between two similar paths $\mathbf{p}$ (in $\mathcal{G}_1$) and $\mathbf{q}$ (in $\mathcal{G}_2$).
doi:10.1371/journal.pone.0008070.g001

The mapping $\mathbf{f}_1$ is defined so that (i) $t_1(u_n|u_m)=0$ for $d_{mn}\notin\mathcal{D}$, (ii) $\sum_n t_1(u_n|u_m)=1$ for all m, and (iii) $t_1(u_{n1}|u_m)>t_1(u_{n2}|u_m)$ for $w_1(u_m,u_{n1})>w_1(u_m,u_{n2})$. Similarly, the mapping $\mathbf{f}_2$ follows the same constraints: (i) $t_2(v_n|v_m)=0$ for $e_{mn}\notin\mathcal{E}$, (ii) $\sum_n t_2(v_n|v_m)=1$ for all m, and (iii) $t_2(v_{n1}|v_m)>t_2(v_{n2}|v_m)$ for $w_2(v_m,v_{n1})>w_2(v_m,v_{n2})$. To specify the emission probability of a virtual symbol s_i at a pair of hidden states in the two HMMs, we define another mapping $\mathbf{g}:h(u_m,v_n)\mapsto e(u_m,v_n)$ that converts the node similarity score $h(u_m,v_n)$ to the following "pairing" probability

$$P(p_i=u_m,q_i=v_n)=e(u_m,v_n)=\mathbf{g}(h(u_m,v_n)), \tag{3}$$

where $(p_i,q_i)=(u_m,v_n)$ is the pair of underlying hidden states for s_i. The mapping $\mathbf{g}$ is defined so that (i) $\sum_{m=1}^{N_1}\sum_{n=1}^{N_2}e(u_m,v_n)=1$ for all possible pairs of (u_m,v_n), and (ii) $e(u_{m1},v_{n1})>e(u_{m2},v_{n2})$ for $h(u_{m1},v_{n1})>h(u_{m2},v_{n2})$.

Ungapped Alignment

Based on the HMM framework, the problem of finding the best matching pair of paths is transformed into the problem of finding the optimal pair of state sequences in the two HMMs that jointly maximize the observation probability of the virtual path $\mathbf{s}$. In an ungapped pathway alignment, the underlying state pair (p_i,q_i) of a virtual symbol s_i directly corresponds to a pair of aligned nodes in the original networks. We can find the optimal pair of paths in polynomial time by using a dynamic programming algorithm defined in the following, which is conceptually similar to the Viterbi algorithm. We first define $\gamma(t,j,\ell)$ as the log-probability of the most probable pair of paths for a subsequence $\widehat{\mathbf{s}}=s_1\dots s_t$ of length $t(\leq L)$, where the underlying states for the virtual symbol s_t are $p_t=u_j$ and $q_t=v_\ell$. The log-probability $\gamma(t,j,\ell)$ can be recursively computed as follows:

$$\gamma(t,j,\ell)=\max_{i,k}\left[\gamma(t-1,i,k)+\log t_1(u_j|u_i)+\log t_2(v_\ell|v_k)+\log e(u_j,v_\ell)\right]. \tag{4}$$

We repeat the above iterations until $t=L$. At the end of the iterations, the maximum log-probability of the virtual path $\mathbf{s}$ is given by:

$$\log P(\mathbf{p}^*,\mathbf{q}^*)=\max_{\mathbf{p},\mathbf{q}}[\log P(\mathbf{p},\mathbf{q})]=\max_{j,\ell}\gamma(L,j,\ell), \tag{5}$$

where $\{\mathbf{p}^*,\mathbf{q}^*\}=\arg\max_{\mathbf{p},\mathbf{q}}[\log P(\mathbf{p},\mathbf{q})]$ is the optimal pair of state sequences that correspond to the best matching paths in the original biological networks. Once we have computed

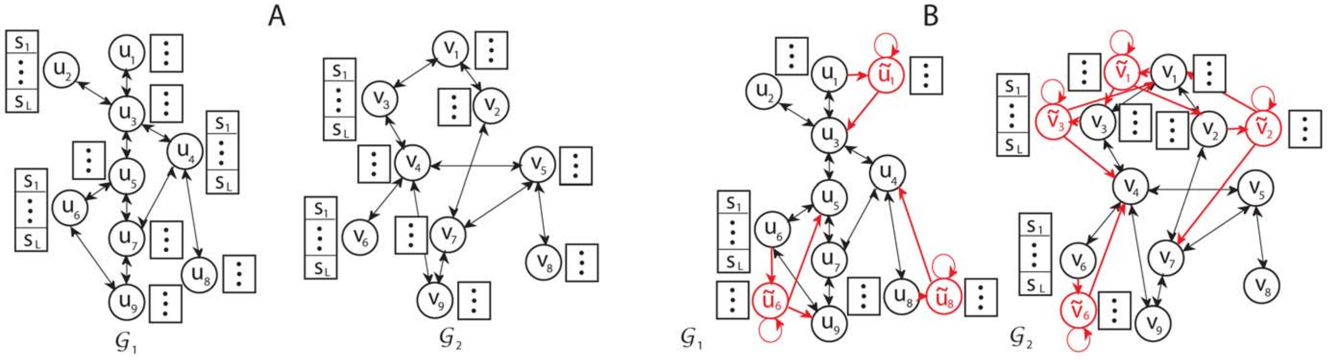

Figure 2. Hidden Markov models for network alignment. (A) Ungapped hidden Markov models (HMMs) for finding the best matching pair of paths. The dots next to the hidden states represent all possible symbols corresponding to virtual nodes in s that can be emitted. (B) Modified HMMs that allow insertions and deletions. For simplicity, changes to the HMMs are shown only for the nodes u_1, u_6, and u_8 in $\mathcal{G}_1$; v_1, v_2, v_3, and v_6 in $\mathcal{G}_2$.
doi:10.1371/journal.pone.0008070.g002

$\log P(\mathbf{p}^*,\mathbf{q}^*)$, it is straightforward to find $\{\mathbf{p}^*,\mathbf{q}^*\}$ by tracing the recursive equations that led to the maximum log-probability $\log P(\mathbf{p}^*,\mathbf{q}^*)$. Although the above algorithm only finds the top-scoring pair of paths, we can easily extend it to find the top k pairs simply by replacing the max operator by an operator that finds the k largest scores.

The computational complexity of the above algorithm is $O(kLM_1M_2)$ for finding the top k pairs of matching paths, where L is the length of the aligned paths that we want to find, M_1 is the number of edges in $\mathcal{G}_1$, and M_2 is the number of edges in $\mathcal{G}_2$. Note that the complexity is linear with respect to all the parameters k, L, M_1, and M_2.

The log-probability $S(\mathbf{p},\mathbf{q}) = \log P(\mathbf{p},\mathbf{q})$ can serve as a good alignment score for the paths $\mathbf{p}$ and $\mathbf{q}$ that effectively combines node similarity and interaction reliability. In principle, we can also use non-stochastic emission (pairing) scores $s_{em}(u_j,v_\ell)$ and transition scores $s^1_{tr}(u_j|u_i)$ and $s^2_{tr}(v_\ell|v_k)$ in the recursive equation (4), in place of the log-probabilities $\log e(u_j,v_\ell)$, $\log t_1(u_j|u_i)$, and $\log t_2(v_\ell|v_k)$, respectively. This will yield a non-stochastic pathway alignment score instead of an observation probability.

As we can see, the concept of the "virtual" path provides an intuitive way of coupling states in two different HMMs. In fact, by taking a closer look at the recursive equation (4), the proposed alignment algorithm can also be viewed as a Markovian walk on a product graph, whose nodes consist of all possible pairs of hidden states in the respective HMMs and the edges between these nodes are determined by the connectivity (or transition probability) between the corresponding states in the HMMs. The algorithm searches for the optimal path (or the top-k paths) in the product graph that yields the highest score based on the parameters of the given HMMs.

Alignment with Gaps

To accommodate gaps in the aligned paths $\mathbf{p}$ and $\mathbf{q}$, we modify the previous HMMs as follows. First, we add an accompanying state $\tilde{u}_m$ for every state u_m in $\mathcal{G}_1$, and similarly, we add an accompanying state $\tilde{v}_n$ for every state v_n in $\mathcal{G}_2$. Next, we add an outgoing edge from each state to the corresponding accompanying state. In addition to this, we also add outgoing edges from the accompanying state to all the neighboring states of the original state. To be more precise, $\tilde{u}_m$ will have an outgoing edge to every $u_k \in \mathcal{U}(m) = \{u_k | d_{mk} \in \mathcal{D}\}$, and $\tilde{v}_n$ will have an outgoing edge to every $v_\ell \in \mathcal{V}(n) = \{v_\ell | e_{n\ell} \in \mathcal{E}\}$. By varying the transition probabilities $t_1(\tilde{u}_m|u_m)$ and $t_2(\tilde{v}_n|v_n)$, we can control the probabilities of having insertions and/or deletions, and thereby control the "gap penalties" in a pathway alignment. We adjust the outgoing transition probability from u_m so that $t_1(\tilde{u}_m|u_m) + \sum_{u_k} t(u_k|u_m) = 1$; and for the outgoing transition probability from v_n so that $t_1(\tilde{v}_n|v_n) + \sum_{v_\ell} t(v_\ell|v_n) = 1$. We can also control the probabilities of having *consecutive* insertions or deletions by adjusting the probabilities $t_1(\tilde{u}_m|\tilde{u}_m)$ and $t_2(\tilde{v}_n|\tilde{v}_n)$ for making self-transitions at either $\tilde{u}_m$ or $\tilde{v}_n$. The outgoing transition probabilities $t_1(u_k|\tilde{u}_m)$ from an accompanying state $\tilde{u}_m$ are chosen so that they are proportional to $t_1(u_k|u_m)$ and satisfy $t_1(\tilde{u}_m|\tilde{u}_m) + \sum_{u_k} t_1(u_k|\tilde{u}_m) = 1$. The transition probabilities in $\mathcal{G}_2$ can be chosen in a similar manner. The structures of the modified HMMs are depicted in Fig. 2B. Note that, in a gapped alignment, the matching paths (or state sequences) $\mathbf{p}$ and $\mathbf{q}$ will still contain L nodes each, and the only difference from an ungapped alignment is that the paths may now contain one or more accompanying nodes which represent gaps. The proposed framework does not impose any restriction on the number of gaps and their locations in the pathway alignment.

In order to find the optimal pair of paths (and their alignment) that maximize the pathway alignment score, we can apply the same dynamic programming algorithm described in the previous section. The retrieved paths can contain any of the hidden states u_j $(1 \le j \le 2N_1)$ and v_ℓ $(1 \le \ell \le 2N_2)$ in the modified HMMs, where we define $u_{m+N_1} \overset{\Delta}{=} \tilde{u}_m$ and $v_{n+N_2} \overset{\Delta}{=} \tilde{v}_n$ for notational convenience. The optimal paths $\{\mathbf{p}^*,\mathbf{q}^*\} = \arg\max_{\mathbf{p},\mathbf{q}}[\log P(\mathbf{p},\mathbf{q})]$ is the best matching pair of paths from two networks, and they may now contain insertions and/or deletions. As before, if we want to find the top k pairs instead of a single top-scoring pair, we can simply replace the max operator by an operator that finds the k largest scores. Note that the computational complexity of the algorithm is $O(4kLM_1M_2)$, which is still linear with respect to all the parameters.

Extension to Multiple Networks

It is straightforward to extend the described pairwise network alignment algorithm for aligning multiple networks. Without loss of generality, we only consider the extension to the alignment of three networks. Given three network graphs $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$, we construct the corresponding HMMs based on their structures. We again use the concept of virtual paths, and now we assume that a virtual path $\mathbf{s}$ is jointly emitted by these three HMMs. The emission of a virtual symbol s_i is now governed by a pairing probability $e(u_j,v_\ell,x_n)$ of three hidden states u_j, v_ℓ, and x_n that belong to the HMMs that correspond to $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$, respectively. We can find the best matching paths based on the following recursive equation:

$$\gamma(t,j,\ell,n) = \max_{i,k,m}\left[\gamma(t-1,i,k,m) + \log t_1(u_j|u_i) + \log t_2(v_\ell|v_k) + \log t_3(x_n|x_m) + \log e(u_j,v_\ell,x_n)\right], \quad (6)$$

where $e(u_j,v_\ell,x_n) \propto e(u_j,v_\ell)e(v_\ell,x_n)e(u_j,x_n)$ is assumed for simplicity. We repeat the above iterations until we reach $t = L$ and compute the maximum log-probability as follows:

$$\log P(\mathbf{p}^*,\mathbf{q}^*,\mathbf{r}^*) = \max_{\mathbf{p},\mathbf{q},\mathbf{r}}[\log P(\mathbf{p},\mathbf{q},\mathbf{r})] = \max_{j,\ell,n} \gamma(L,j,\ell,n), \quad (7)$$

where $\{\mathbf{p}^*,\mathbf{q}^*,\mathbf{r}^*\} = \arg\max_{\mathbf{q},\mathbf{q},\mathbf{r}}[\log P(\mathbf{p},\mathbf{q},\mathbf{r})]$ corresponds to the set of best matching paths in the three networks.

Implementation of the Alignment Algorithm

It should be noted that although we fix the length of the virtual path to L, we can in fact find any top-scoring alignment with a shorter length $L' \le L$, since we store all the alignment scores for shorter alignments while running the dynamic programming algorithm. The recursive equations in (4) and (6) do not restrict multiple occurrence of the same node in the final pathway alignment. However, when it is desirable to avoid such multiple occurrence, we can easily incorporate a "look-back" step into each iteration in order to prevent adding a node that is already included in the (intermediate) alignment. As this requires tracing the intermediate optimal (or top k) alignment, the computational complexity of the recursive equations (4) and (6) with a "look-back" step will be increased in proportion to the length of the intermediate alignment.

In order to obtain more general subnetwork alignments, not just alignments of linear paths, we can combine the overlapping paths among the top k retrieved pairs of paths. The edges that are already contained in the constructed subnetwork alignment (which correspond to the conserved molecular interactions in the biological networks) are then removed from the HMMs, and we run the dynamic programming algorithm again to find another subnetwork alignment that does not overlap with the retrieved subnetworks. By repeating this "search and peel-off" process, we can effectively find diverse subnetwork regions that are conserved in the given networks.

The memory complexity of the proposed algorithm is $O(kLN_1N_2\ldots N_B)$ for finding the top k pathway alignments for B networks. Although the required amount of memory increases only linearly with respect to each parameter, it can still make the algorithm infeasible when we want to align multiple number of large networks. To overcome this problem, we may assign non-zero pairing probabilities $e(\cdot)$ to a set of nodes (in the respective networks) only if every pair in this set has considerable node similarity that exceeds a certain threshold. Assuming that there are T sets of nodes that satisfy this condition, we only need to consider these T possible node alignments, in which case the overall memory complexity reduces to $O(kLT)$. Since T is often much smaller than $N_1N_2\ldots N_B$, this scheme can save significant amount of memory, thereby making the algorithm feasible.

Results

To demonstrate the effectiveness of the HMM-based network alignment algorithm, we carried out the following experiments. First, we used our algorithm to align two pairs of small synthetic networks that were used to validate the network alignment algorithm proposed in [24]. Second, we used the proposed algorithm for finding putative pathways in the fruit fly PPI network that look similar to known human pathways. Finally, we applied the algorithm for aligning microbial PPI networks to assess its ability to find conserved functional modules.

Aligning Synthetic Networks

To illustrate the potential capability of aligning different types of molecular networks, we first tested our algorithm using two small synthetic examples, which include a pair of undirected networks and another pair of directed networks. These examples were obtained from the tutorial files in the PathBLAST plugin of software Cytopscape (version 1.1, http://www.cytoscape.org/plugins1.php) and they were used for the validation of a network alignment algorithm called MNAligner [24].

HMM parameterization. For aligning the synthetic networks, we parameterized the HMMs as follows. We set the transition scores $s_{tr}(u_n|u_m)$ directly based on the "adjacent matrices" given in [24], which contain the interaction scores between two nodes in the respective networks. Every interaction score takes a value between 0 and 1, hence we can view it as the "interaction probability". We took the logarithm of this interaction probability as the transition score $s_{tr}(u_n|u_m)$. When there is no interaction between two nodes, we have $s_{tr}(u_n|u_m) = -\infty$. This keeps the HMM from making a direct transition from a state u_m to a non-relevant state u_n, thereby preventing the inclusion of irrelevant protein interactions that do not have any biological support in the network. Similarly, we obtained the emission scores $s_{em}(u_m,v_n)$ by taking the logarithm of the similarity scores between nodes given by the "similarity matrices" in [24]. The adjacent matrices and the similarity scores for the two examples can be found in the Supporting Information S1.

Example 1: Aligning undirected networks. We first used our algorithm for aligning a pair of undirected networks. To compare the alignment results with the results obtained by MNAligner [24], we looked for the top 500 alignments without gaps, where the length of the virtual path was set to $L = 3$. By incorporating "look-back" steps into our dynamic programming algorithm, we restricted the multiple occurrence of the same node pair in the obtained pathway alignment. The top-scoring pathway alignment obtained from our algorithm was $A|QQ\leftrightarrow C|BB\leftrightarrow F|HH$, which is identical to the optimal alignment identified by both PathBLAST [6] and MNAligner [24]. Unlike PathBLAST, the proposed HMM-based algorithm and the MNAligner both keep the natural order of the nodes in the original networks. We also noticed that the paths $A\leftrightarrow C\leftrightarrow F$ and $QQ\leftrightarrow BB\leftrightarrow HH$ can be aligned with several other potential similar paths in the corresponding networks from the top 500 aligned results. After removing the interactions included in the top-scoring alignment, we searched for the next top-scoring alignment. This returned the alignment $J|WW\leftrightarrow I|DD\leftrightarrow L|OO$, which was also ranked as the second best alignment by MNAligner [24]. Repeating the experiment after removing this alignment returned $B|MM\leftrightarrow D|CC\leftrightarrow E|ZZ$ as the third best alignment. This is different from the alignment $H|AA\leftrightarrow G|NN\leftrightarrow B|CC$ that was found by MNAligner, which got a lower score in our experiment. We noted that the alignment $H|AA\leftrightarrow G|NN\leftrightarrow B|CC$ is not as significant as the three alignments that we found, as $H\leftrightarrow G\leftrightarrow B$ can be aligned with many other paths with the same alignment score. By repeating the above experiments and combining the pathway alignment results, we obtained the global network alignment illustrated in Fig. 3A, where a bold line represents that the corresponding edges in the respective networks are matched, whereas a thin line indicates a mismatch. These results show that the HMM-based method can effectively identify the top matching paths in different undirected networks, and it yields better results with higher alignment scores integrating both node similarity and interaction probability compared to PathBLAST and MNAligner for this purpose.

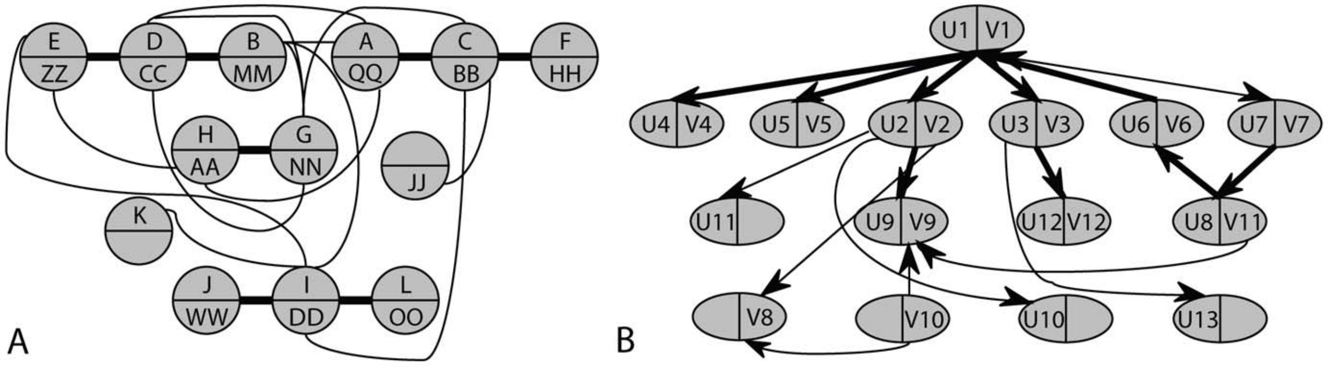

Figure 3. The alignment results for synthetic networks. (A) Undirected networks; (B) Directed networks.
doi:10.1371/journal.pone.0008070.g003

Example 2: Aligning directed networks. Without any modification, our algorithm can also be used for aligning directed networks. We demonstrate this by using the second example that contains a pair of small directed networks. In this experiment, we set the length of the virtual path to $L=6$, which is the length of the longest path in these two networks. As there are fewer legitimate paths in these networks, we only looked for the top 20 aligned pairs of paths. The obtained pathway alignments were combined to get the global network alignment shown in Fig. 3B. The alignment results were similar to those obtained by MNAligner [24], except that we found fewer aligned nodes and edges. This is natural since there exist only a few similar pairs of nodes in the given networks (see Supporting Information S1) and as our algorithm focuses on finding the best local alignments instead of a global alignment. Note that, unlike PathBLAST, which finds path alignments based on several heuristics, the proposed algorithm can find the mathematically optimal path alignment for the given networks.

Aligning Annotated Pathways with PPI Networks

HMM parameterization. The proposed algorithm can also be used for identifying putative pathways in a new biological network, which look similar to known pathways. To demonstrate this, we used our algorithm to search for human signaling pathways in the fruit fly PPI network. In order to compare the search results with those of the network querying algorithm in [18], the HMMs were parameterized according to the non-stochastic scoring scheme in [18] as we describe in the following. The transition score $s_{tr}(u_n|u_m)$ was set to $s_{tr}(u_n|u_m)=\log(1)=0$ in the presence of interaction between the proteins that correspond to u_n and u_m, and it was set to $s_{tr}(u_n|u_m)=\log(0)=-\infty$ in the absence of any interaction. To allow gaps in alignments, the transition score from a state u_m to its accompanying state $\tilde{u}_m$ was set to $s_{tr}(\tilde{u}_m|u_m)=0$, and we set the self-transition score at $\tilde{u}_m$ to $s_{tr}(\tilde{u}_m|\tilde{u}_m)=0$ to allow consecutive gaps. Furthermore, the score for making a transition from $\tilde{u}_m$ to a regular state u_n was set to $s_{tr}(u_n|\tilde{u}_m)=0$ for $u_n \in \mathcal{U}(m)=\{u_n|d_{mn}\in\mathcal{D}\}$ and $s_{tr}(u_n|\tilde{u}_m)=-\infty$ for $u_n \notin \mathcal{U}(m)$. The emission score $s_{em}(u_m,v_n)$ for two proteins u_m and v_n in different networks (where the query network is simply a linear path in this case) was computed based on their sequence similarity. For each protein pair (u_m,v_n), we computed its E-value using the PRSS routine in the FASTA package [25,26], which is known to yield more accurate E-values compared to BLASTP [27]. We regarded a protein pair (u_m,v_n) as a "match" if its E-value $E_v(u_m,v_n)$ was below a threshold λ_{th}. Otherwise, we regarded the pair as a "mismatch", which implies that the proteins do not bear significant similarity. Based on this criterion, we set the emission score $s_{em}(u_m,v_n)$ as follows:

$$s_{em}(u_m,v_n)=\begin{cases} -\log_{10} E_v(u_m,v_n), & \text{if } E_v(u_m,v_n)\le\lambda_{th} \\ -\Delta, & \text{otherwise.} \end{cases} \tag{8}$$

The value Δ can be viewed as the mismatch penalty, and is selected so that $-\Delta \ll -\log_{10}\lambda_{th}$. We set the insertion and deletion penalty also to $-\Delta$. Finally, since two accompanying states cannot be paired with each other, we set $s_{em}(\tilde{u}_m,\tilde{v}_n)=-\infty$.

Querying human pathways in the fruit fly PPI network. We first obtained the PPI network of *Drosophila melanogaster* from the Database of Interacting Proteins (DIP) [28] and constructed the "target HMM". Then we constructed a "query HMM" for the human hedgehog signaling pathway and another query HMM based on the human MAP kinase pathway. When constructing the query HMMs, we regarded each signaling pathway as a "directed network" with a linear structure, instead of a "sequence of proteins" as in [18]. The similarity threshold was set to $\lambda_{th}=0.5$ and the gap penalty was set to $\Delta=12$, as in [18]. The constructed query HMMs were then used to search for matching paths in the target HMM. Despite the generality and the different implementation of the proposed algorithm, the top pathways retrieved by the proposed algorithm agree with the predictions in [18], which is the direct consequence of the mathematical optimality of both methods. For the human hedgehog signaling pathway lhh–Ptch–Smo–Stk36–Gli, the top-scoring pathway in the *D. melanogaster* network agreed well with the putative *D. melanogaster* hedgehog signaling pathway reported in the KEGG database [29]. In fact, the best aligned path in the fruit fly network contained shh–ptc–Smo–fu–ci, which is identical to the core portion of the putative fly hedgehog signaling pathway (http: //www. genome.jp /dbget-bin /get_ pathway?org_name = dme&mapno = 04340) in the KEGG database [29]. The query result of the human MAP kinase pathway Egfr–drk–Sos–Ras85D–ph1–Mekk1–ERKA was also biologically significant, and the seven proteins in the retrieved pathway matched exactly with the proteins in the putative fruit fly MAP kinase pathway (http://www.genome.jp/dbget-bin/get_pathway?org_name = map&mapno = 04010) reported in KEGG. These results compare favorably to the results obtained by one of the state-of-the-art algorithms [11], where they found two identical proteins in the putative fly hedgehog signaling pathway and five proteins in the putative fly MAPK pathway.

Aligning Microbial PPI Networks

In order to validate the accuracy of our algorithm for predicting functional modules that are conserved in different organisms, we performed additional experiments using three microbial PPI networks obtained from [9]. In our experiments, we performed a pairwise alignment between the *E. coli* and the *C. crescentus* networks as well as a pairwise alignment between the *E. coli* and the *S. typhimurium* networks. We assessed the accuracy of our algorithm based on the consistency of the *KEGG ortholog (KO) group* annotations [29] of the aligned proteins. In order to measure the consistency of KO group annotations, we computed the specificity of the predictions based on a similar methodology that was used in [14]. To compute this measure, we first remove all the aligned protein pairs that do not have complete KO annotations, and then compute the total number of annotated protein pairs. An annotated protein pair is regarded as being *correct* if both proteins have the same KO group annotations, and *incorrect* if the annotations do not agree. The specificity is defined as the ratio of the number of "correct" protein pairs among all annotated protein pairs.

For this experiment, the parameters of the HMMs have been chosen as follows. First, the transition scores $s_{tr}(u_n|u_m)$ have been obtained by taking the logarithm of the protein interaction probabilities in the microbial networks, which had been assigned by the SRINI algorithm [30]. The emission scores $s_{em}(u_m,v_n)$ have been computed based on the sequence similarity between the proteins u_m and v_n, as in the previous section, where the protein similarities have been estimated based on the BLASTP hit scores between protein pairs provided in [9].

Based on the constructed HMMs, we used our algorithm to find the top-scoring pathway alignment with gaps. At each iteration, we looked for the top aligned pair of paths, stored the alignment, and removed the interactions included in the alignment from the respective networks for the next iteration. By repeating this iteration, we found 200 high-scoring path alignments. This experiment has been repeated with varying virtual path length: $L=6$, 12, 18, 24, and 30. In all our experiments, we disallowed multiple occurrence of identical protein pairs and set the gap/

mismatch penalty to $\Delta = 1.0$. For each experiment, we computed the cumulative specificity for the top k alignments, which is given by

$$cs_k = \frac{\sum_{i=1}^{k} c_i^c}{\sum_{i=1}^{k} c_i^a}, \tag{9}$$

where c_i^c is the total number of correctly aligned protein pairs in the top i alignments, and c_i^a is the total number of annotated protein pairs also in the top i alignments. The result from the pairwise alignment of the *E. coli* and the *C. crescentus* networks is shown in Fig. 4A, and the result from the alignment of the *E. coli* and the *S. typhimurium* networks is shown in Fig. 4B. As we can see in both Fig. 4A and Fig. 4B, the cumulative specificity cs_k generally decreases when we increase the alignment length L. This is expected since the algorithm tends to recruit more protein pairs in the alignment if we increase L. Furthermore, cs_k generally decreases if we increase k. This is natural, since alignments with lower scores correspond to less conserved pathways with larger variations. Although it is difficult to directly compare our results with those reported in [14], it is still worth to note that the cumulative specificity (for the top 200 alignments) of the proposed HMM-based algorithm is higher than the specificity of the alignment algorithm Græmlin 2.0 [14], for both pairwise network alignments. These results clearly indicate that our HMM-based algorithm can produce accurate network alignments that are biologically meaningful.

Further analysis of the predicted alignments led to a number of interesting observations. For example, the alignment of *E. coli* and *C. crescentus* networks and the alignment of *E. coli* and *S. typhimurium* networks both detected conserved DNA replication modules. The module contained components of the primosome (dnaA, gyrA, gyrB), subunits of topoisomerase IV (parC, parE), and a subunit of DNA polymerase III (dnaN). These protein families are all known to be involved in DNA replication. We also found other interesting conserved modules, which include both large and small subunits of ribosomal protein complexes (rplA, rplB, rplC, rplE, rplK, rplP; and rpsA, rpsB, rpsC, rpsE, rpsG, rpsK); DNA-directed RNA polymerase complex containing rpoA, rpoB, rpoC, and other subunits; the citrate cycle (TCA cycle) containing 2-oxoglutarate dehydrogenase E1 component (sucA, sucB) and succinyl-CoA synthetase (sucC, sucD); NADH dehydrogenase I (nuoA, nuoB, nuoC, nuoF, nuoH, nuoI, nuoL, nuoM), which is a part of the oxidative phosphorylation pathway; nitrate reductase 1 (with narG, narH, narI, and narJ); and a portion of the bacterial secretion system (with secA, secD, secY).

Discussion

In this paper, we proposed an HMM-based network alignment algorithm that can be used for finding conserved pathways in two or more biological networks. The HMM framework and the proposed alignment algorithm has a number of important advantages compared to other existing local network alignment algorithms. First of all, despite its generality, the proposed algorithm is very simple and efficient. In fact, the alignment algorithm based on the proposed HMM framework is a variant of the Viterbi algorithm. As a result, it has a very low polynomial computational complexity, which grows only linearly with respect to the length of the identified pathways and the number of edges in each network. This makes it possible to find conserved pathways with more than 10 nodes in networks with thousands of nodes and tens of thousands of interactions within a few minutes on a personal computer. Furthermore, the HMM-based framework can handle a large class of path isomorphism, which allows us to find pathway alignments with any number of gaps (node insertions and deletions) at arbitrary locations. In addition to this, the proposed framework is very flexible in choosing the scoring scheme for pathway alignments, where different penalties can be used for mismatches, insertions and deletions. We can also assign different penalties for gap opening and gap extension, which can be convenient when comparing networks that are remotely related to each other. Another important advantage of the proposed framework is that it allows us to use an efficient dynamic programming algorithm for finding the mathematically optimal alignment. Considering that many available algorithms rely on heuristics that cannot guarantee the optimality of the obtained solutions, this is certainly a significant merit of the HMM-based approach. Although the mathematical optimality does not

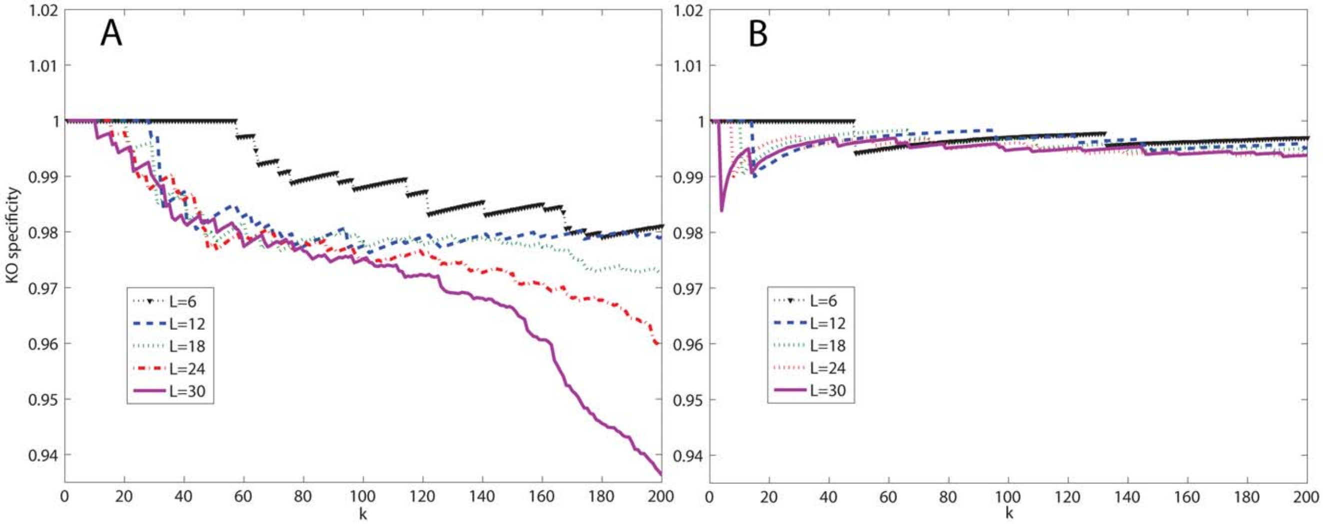

Figure 4. Functional specificity for microbial network alignment. The cumulative specificity of the top 200 aligned pathways obtained from (A) the pairwise alignment between *E. coli* and *C. crescentus* networks; and (B) the pairwise alignment between *E. coli* and *S. typhimurium* networks.
doi:10.1371/journal.pone.0008070.g004

guarantee the biological significance of the obtained solution, it can certainly lead to more accurate predictions if combined with a realistic scoring scheme for assessing pathway homology. As demonstrated in our experiments, the proposed algorithm yields accurate and biologically meaningful results both for querying known pathways in the network of another organism and also for finding conserved functional modules in the networks of different organisms. Finally, the HMM-based framework presented in this paper can be extended for aligning multiple networks. While many current multiple network alignment algorithms adopt a progressive approach for comparing multiple networks [9,14–17], our HMM-based framework provides a potential way to simultaneously align multiple networks to find the optimal set of conserved pathways with maximum alignment score.

For future research, we plan to evaluate the performance of our HMM-based algorithm more extensively by investigating the consistency of the predicted alignments based on other available functional annotations, including the gene ontology (GO) annotations [31]. It would be also beneficial to develop a more elaborate scoring scheme that integrates additional information, such as the GO annotations and the KO group annotations, to obtain more reliable alignment results. Finally, we are currently working on simultaneous multiple network alignment based on the HMM framework, where the goal is to construct a scalable multiple alignment algorithm that yields network alignments with higher fidelity.

Supporting Information

Supporting Information S1
Found at: doi:10.1371/journal.pone.0008070.s001 (0.06 MB PDF)

Acknowledgments

The authors would also like to thank Maxim Kalaev, Wenhong Tian, as well as Jason Flannick for sharing the datasets and for the helpful communication.

Author Contributions

Conceived and designed the experiments: XQ BJY. Performed the experiments: XQ. Analyzed the data: XQ BJY. Wrote the paper: XQ BJY.

References

1. Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, et al. (2001) A comprehensive two-hybrid analysis to explore the yeast protein interactome. Proc Natl Acad Sci USA 98: 4569–4574.
2. Mann M, Hendrickson R, Pandey A (2001) Analysis of proteins and proteomes by mass spectrometry. Annu Rev Biochem 70: 437–473.
3. Uetz P, Rajagopala S, Dong Y, Haas J (2004) From orfeomes to protein interaction maps in viruses. Genome Res 14: 2029–2033.
4. Krogan N, et al. (2006) Global landscape of protein complexes in the yeast saccharomyces cerevisiae. Nature 440: 4412–4415.
5. von Mering C, Krause R, Snel B, Cornell M, Oliver S, et al. (2002) Comparative assessment of large-scale data sets of protein-protein interactions. Nature 417: 399–403.
6. Kelley B, Sharan R, Karp R, Sittler T, Root D, et al. (2003) Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proc Natl Acad Sci USA 100: 11394–11399.
7. Koyutürk M, Grama A, Szpankowski W (2004) An efficient algorithm for detecting frequent subgraphs in biological networks. Bioinformatics 20: SI200–207.
8. Sharan R, Suthram S, Kelley R, Kuhn T, McCuine S, et al. (2005) Conserved patterns of protein interaction in multiple species. Proc Natl Acad Sci USA 102: 1974–1979.
9. Flannick J, Novak A, Srinivasan B, McAdams H, Batzoglou S (2006) Græmlin: general and robust alignment of multiple large interaction networks. Genome Res 16: 1169–1181.
10. Scott J, Ideker T, Karp R, Sharan R (2006) Efficient algorithms for detecting signaling pathways in protein interaction networks. J Comput Biol 13: 133–144.
11. Shlomi T, Segal D, Ruppin E, Sharan R (2006) QPath: a method for querying pathways in a protein-protein interaction network. BMC Bioinformatics 7.
12. Yang Q, Sze S (2007) Path matching and graph matching in biological networks. J Comput Biol 14: 56–67.
13. Dost B, Shlomi T, Gupta N, Ruppin E, Bafna V, et al. (2008) QNet: a tool for querying protein interaction networks. J Comput Biol 15: 913–925.
14. Flannick J, Novak A, Dol C, Srinivasan B, Batzoglou S (2008) Automatic parameter learning for multiple network alignment. In: Proc of the 10th Annu Int Conf Res Comput Mol Bio (RECOMB 2008).
15. Kalaev M, Bafna V, Sharan R (2008) Fast and accurate alignment of multiple protein networks. In: Proc of the 10th Annu Int Conf Res Comput Mol Bio (RECOMB 2008).
16. Singh R, Xu J, Berger B (2008) Global alignment of multiple protein interaction networks with application to functional orthology detection. Proc Natl Acad Sci USA 105: 12763–12768.
17. Liao C, Lu K, Baym M, Singh R, Berger B (2009) IsoRankN: spectral methods for global alignment of multiple protein networks. Bioinformatics 25: 253–258.
18. Qian X, Sze S, Yoon B (2009) Querying pathways in protein interaction networks based on hidden markov models. J Comput Biol 16: 145–157.
19. Tian W, Samatova N (2009) Pairwise alignment of interaction networks by fast identification of maximal conserved patterns. In: Pac Symp Biocomput. volume 14. pp 99–110.
20. Zaslavskiy M, Bach F, Vert J (2009) Global alignment of protein-protein interaction entworks by graph matching methods. Bioinformatics 25: 259–267.
21. Pinter R, Rokhlenko O, Yeger-Lotem E, Ziv-Ukelson M (2005) Alignment of metabolic pathways. Bioinformatics 21: 3401–3408.
22. Akutsu T, Kuhara S, Maruyama O, Miyano S (1998) Identification of gene regulatory networks by strategic gene disruptions and gene overexpressions. In: Proc. 9th Annu. ACM-SIAM Symp. Discrete Alg. pp 695–706.
23. Steffen M, Petti A, Aach J, D'haeseleer P, Church G (2002) Automated modelling of signal transduction networks. BMC Bioinformatics 3: 34.
24. Li Z, Zhang S, Wang Y, Zhang X, Chen L (2007) Alignment of molecular networks by integer quadratic programming. Bioinformatics 23: 1631–1639.
25. Pearson W, Lipman D (1988) Improved tools for biological sequence comparison. Proc Natl Acad Sci USA 85: 2444–2448.
26. Pearson W (1996) Effective protein sequence comparison. Methods Enzymol 266: 227–258.
27. Pagni M, Jongeneel C (2001) Making sense of score statistics for sequence alignments. Brief Bioinform 2: 51–67.
28. Xenarios I, Salwinski L, Duan X, Higney P, Kim S, et al. (2002) DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions. Nucleic Acids Res 30: 303–305.
29. Kanehisa M, Goto S (2000) KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res 28: 27–30.
30. Srinivasan B, Novak A, Flannick J, Batzoglou S, McAdams H (2006) Integrated protein interaction networks for 11 microbes. In: Proc of the 10th Annu Int Conf Res Comput Mol Bio (RECOMB 2006).
31. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, et al. (2000) Gene ontology: tool for the unification of biology. the gene ontology consortium. Nat Genet 25: 25–29.