BIOINFORMATICS

Vol. 00 no. 00 2012
Pages 1–9

RESQUE: Network Reduction Using Semi-Markov Random Walk Scores for Efficient Querying of Biological Networks

Sayed Mohammad Ebrahim Sahraeian [1] and Byung-Jun Yoon, [1,*]

[1] Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843-3128, USA

ABSTRACT

Motivation: Recent technological advances in measuring molecular interactions have resulted in an increasing number of large-scale biological networks. Translation of these enormous network data into meaningful biological insights requires efficient computational techniques that can unearth the biological information that is encoded in the networks. One such example is network querying, which aims to identify similar subnetwork regions in a large target network that are similar to a given query network. Network querying tools can be used to identify novel biological pathways that are homologous to known pathways, thereby enabling knowledge transfer across different organisms.

Results: In this paper, we introduce an efficient algorithm for querying large-scale biological networks, called RESQUE. The proposed algorithm adopts a semi-Markov random walk model to probabilistically estimate the correspondence scores between nodes that belong to different networks. The target network is iteratively reduced based on the estimated correspondence scores, which are also iteratively re-estimated to improve accuracy until the best matching subnetwork emerges. We demonstrate that the proposed network querying scheme is computationally efficient, can handle any network query with an arbitrary topology, and yields accurate querying results.

Availability: The source code of RESQUE is freely available at http://www.ece.tamu.edu/~bjyoon/RESQUE/

Contact: bjyoon@ece.tamu.edu

1 INTRODUCTION

Biological functions in cells are carried out through complicated interactions among various cellular constituents. For instance, protein-protein interactions (PPI) lie at the core of various transcriptional, signaling, and metabolic processes in cells (Zhang, 2009). Quantitative genome-scale description of such interactions by graphical representation of biological networks can facilitate the study of the cell as an integrated system (Barabasi and Oltvai, 2004; Cusick *et al.*, 2005) and help us better understand the structure and dynamics of diverse biological mechanisms. Recent technological advances have enabled high-throughput global measurement of protein-protein interactions (Uetz *et al.*, 2000; Ho *et al.*, 2002; Ge, 2000), resulting in large-scale PPI networks. Furthermore, many text-mining tools have been developed to search the vast amount of biomedical research literature to collect known molecular interactions that have been reported before (Huang *et al.*, 2008). As a result, genome-scale biological networks are available for a number of model organisms, and biological network databases are still in rapid expansion. In order to translate these large-scale network data into meaningful biological insights, we need efficient computational techniques that can be used to unearth the important information that is buried in these networks.

As comparative methods have played crucial roles in the analysis of biological sequences, comparative network analysis can also serve as an effective way of analyzing the available network data (Sharan and Ideker, 2006; Yoon *et al.*, 2012). One such example is network querying. *Network querying* aims to identify similar subnetwork regions in a large biological network (referred as the "target network") that are similar to a given query network. This technique can be used to search for novel potential pathways in a given biological network that are similar to known biological pathways, thereby enabling "knowledge transfer" across different organisms, from well-studied ones to others that have been studied less. To obtain biologically meaningful results, the network querying algorithm needs to incorporate the similarity between the individual nodes (i.e., biomolecules in the networks)–in terms of their composition, structure, or function– as well as the similarity between their interactions patterns (i.e., topological similarity). However, the optimal network querying problem has been shown to be NP-complete by reduction to the graph isomorphism problem (Dost *et al.*, 2008), and until now, various approaches have been proposed to make this problem computationally feasible (Kelley *et al.*, 2004; Shlomi *et al.*, 2006; Dost *et al.*, 2008; Blin *et al.*, 2010b; Yang and Sze, 2007; Qian *et al.*, 2009; Mongiovi *et al.*, 2010; Gulsoy and Kahveci, 2011; Ferraro *et al.*, 2011; Bruckner *et al.*, 2010; Blin *et al.*, 2010a; Pinter *et al.*, 2005; Durand *et al.*, 2006; Tian *et al.*, 2007; Wernicke and Rasche, 2007; Ferro *et al.*, 2007; Fionda *et al.*, 2008; Ay *et al.*, 2011; Singh *et al.*, 2008; Liao *et al.*, 2009; Sahraeian and Yoon, 2011a).

PathBLAST (Kelley *et al.*, 2004) is one of the pioneering network querying schemes that identifies conserved linear pathways in a pair of networks using a greedy "seed-and-extend" approach. Due to the high computational burden, this algorithm is restricted to relatively short pathways. To reduce computational complexity, QPath (Shlomi *et al.*, 2006) and QNet (Dost *et al.*, 2008) adopted a color-coding based approach. QPath (Shlomi *et al.*, 2006) is

[*] to whom correspondence should be addressed

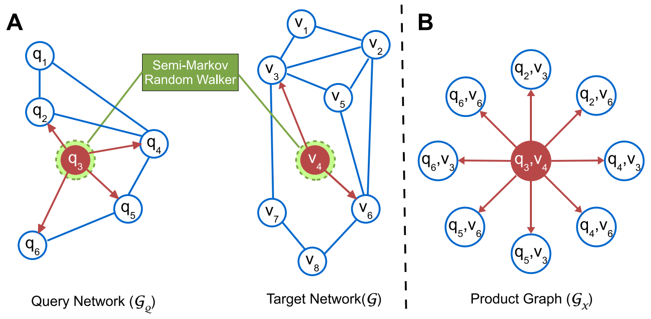

Fig. 1. (A) Simultaneous random walk on the query network $\mathcal{G}_\mathcal{Q}$ and the target network $\mathcal{G}$. (B) This is equivalent to a random walk on the product graph $\mathcal{G}_\mathcal{X} = \mathcal{G}_\mathcal{Q} \times \mathcal{G}$.

restricted to querying linear paths, while QNet (Dost *et al.*, 2008) allows searching for trees and bounded tree-width graphs. Another recent algorithm, called PADA1 (Blin *et al.*, 2010b), also used the color-coding scheme and dynamic programming for network querying. Yang and Sze (Yang and Sze, 2007) proposed two algorithms called PathMatch, which searches for the longest weighted path, and GraphMatch, which enumerates all possible solutions to find the highest scoring subgraphs in a directed acyclic graph. Qian and Yoon (Qian *et al.*, 2009) proposed a hidden Markov model (HMM) based scheme for querying linear paths. SIGMA (Mongiovi *et al.*, 2010) and RINQ (Gulsoy and Kahveci, 2011) consider the problem of efficient querying in biological network databases using indexing schemes. Ferraro *et al.* (2011) proposed an asymmetric approach that uses a master-slave scheme to extract, via the Viterbi algorithm, matching subgraphs in the master network. Another recent algorithm, called Torque (Bruckner *et al.*, 2010), proposed a topology-free querying scheme that looks for a connected set of matching proteins in the target network that are sequence-similar to proteins in the query network, where the underlying motivation for taking a topology-free approach was the incompleteness of the currently available protein interaction data. Another algorithm called GraMoFoNe (Blin *et al.*, 2010a) also‘ adopted this topology-free approach, based on a color coded motif matching scheme.

Currently, most network querying algorithms cannot handle queries with general network structures. Many algorithms either restrict the query to have a relatively simple structure (e.g., linear path or tree) (Kelley *et al.*, 2004; Shlomi *et al.*, 2006; Dost *et al.*, 2008; Blin *et al.*, 2010b; Qian *et al.*, 2009) or simply view it as a collection of nodes by ignoring the underlying topology (Bruckner *et al.*, 2010; Blin *et al.*, 2010a). Furthermore, many algorithms strongly rely on sequence similarity to identify homologous node pairs, where candidate pairs are typically identified by thresholding the similarity scores (e.g., BLAST score) (Ferraro *et al.*, 2011; Bruckner *et al.*, 2010; Blin *et al.*, 2010a; Kelley *et al.*, 2004). Besides, many network querying algorithms still suffer from high computational cost, which often increases exponentially with the query size, rendering them impractical for large queries (Fionda and Palopoli, 2011).

In this paper, we propose a novel network querying algorithm, called RESQUE (REduction-based scheme using Semi-Markov scores for network QUErying), that can effectively address the shortcomings of existing algorithms. For fast and accurate network querying, the proposed algorithm takes an efficient reduction-based approach, where the target network is iteratively reduced based on the so-called node correspondence scores that are computed using a semi-Markov random walk model. The node correspondence score provides a probabilistic similarity measure between nodes that belong to different networks (i.e., query and target networks), which can be efficiently computed using a closed-form formula. At each iteration, the estimated scores are used to remove the nodes in the target network that have minimal correspondence to the nodes in the query network, thereby shrinking the search space. The node correspondence scores are then re-estimated based on the reduced network and the aforementioned reduction process is repeated until the best matching subnetwork emerges. Based on real as well as synthetic examples, we demonstrate that RESQUE outperforms state-of-the-art network querying techniques, in terms of both computational efficiency and querying accuracy.

2 MATERIALS AND METHODS

Suppose we have a target protein-protein interaction (PPI) network, represented by an undirected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$, where $\mathcal{V}$ is the set of N nodes that correspond to the proteins in the network, $\mathcal{E}$ is the set of m edges that represent the protein interactions, and w is a weight function $w : \mathcal{E} \to \mathbb{R}$, representing the strength or reliability of an interaction. Let $\mathcal{G}_\mathcal{Q} = (\mathcal{V}_\mathcal{Q}, \mathcal{E}_\mathcal{Q}, w_\mathcal{Q})$ be the query network with a set $\mathcal{V}_\mathcal{Q}$ of $N_\mathcal{Q}$ nodes, edge set $\mathcal{E}_\mathcal{Q}$, and interaction reliability $w_\mathcal{Q}$. In case the interaction reliability data are unavailable, we may simply assign uniform reliability score to all edges. For a pair of proteins (q_i, v_j), where $q_i \in \mathcal{V}_\mathcal{Q}$ and $v_j \in \mathcal{V}$, the node similarity is denoted as $h(q_i, v_j)$. Typically, sequence similarity scores are used to measure this node similarity, although we may also use other measures based on structural or functional similarity between the proteins. The overall goal of network querying is to identify the subnetwork in the target network $\mathcal{G}$ that is maximally similar to the given query network $\mathcal{G}_\mathcal{Q}$.

2.1 Estimation of probabilistic node correspondence scores through semi-Markov random walk

The network querying problem can be viewed as a mapping problem, in which we assign each query node $q_i \in \mathcal{V}_\mathcal{Q}$ to one or more target nodes $v_j \in \mathcal{V}$, based on their overall biological similarity. This biological similarity should be measured by integrating the node similarity between the matching proteins themselves as well as the similarity between their interaction patterns with other neighboring proteins. The semi-Markov random walk (SMRW) model can provide an effective means of obtaining such integrated similarity scores (Sahraeian and Yoon, 2011a,b).

Markov random walk is a process that takes successive random steps (on a graph or a path) according to the Markov assumption. In an ordinary Markov random walk, the random walker always spends a fixed amount of time at a given position before making the next move. On the other hand, in a semi-Markov random walk, the walker may spend a random amount of time between each transition. Here, we consider a *simultaneous* semi-Markov random walk on both query and target networks, as shown in Figure 1A. The position of the walker is given by a pair of nodes (q_i, v_j), where $q_i \in \mathcal{V}_\mathcal{Q}$ and $v_j \in \mathcal{V}$. At each time step, the walker takes simultaneous random steps on both networks, by moving to one of the neighboring nodes in each network, where a neighbor with a higher interaction reliability score is more likely to be selected. As illustrated in Figure 1B, this simultaneous random walk on two graphs $\mathcal{G}_\mathcal{Q}$ and $\mathcal{G}$ is equivalent to a random walk on their product graph $\mathcal{G}_\mathcal{X} = (\mathcal{V}_\mathcal{X}, \mathcal{E}_\mathcal{X})$ (Vishwanathan *et al.*, 2010). In this product graph $\mathcal{G}_\mathcal{X}$, every node in $\mathcal{V}_\mathcal{X}$ corresponds to a node pair $x = (q_i, v_j)$, and an edge exists between two node pairs $x = (q_i, v_j)$ and $y = (q_k, v_l)$ if and only if the edges $(q_i, q_k) \in \mathcal{E}_\mathcal{Q}$ and $(v_j, v_l) \in \mathcal{E}$ are present in the original networks. As shown in Vishwanathan *et al.* (2010), the transition probability matrix of the underlying Markov chain for the random walk on $\mathcal{G}_\mathcal{X}$ can be computed as $\mathcal{A}_\mathcal{X} = \mathcal{A}_\mathcal{Q} \otimes \mathcal{A}$, where $\otimes$ denotes the Kronecker product, and $\mathcal{A}_\mathcal{Q} = [a_\mathcal{Q}(i,k)]$ and $\mathcal{A} = [a(j,l)]$ are the transition probability matrices for the random walks on $\mathcal{G}_\mathcal{Q}$ and $\mathcal{G}$, respectively. The transition probability $a(j,l)$ measures the normalized contribution from the neighboring node v_l to the node v_j based on the interaction reliability $w(v_j, v_l)$ as follows:

$$a(j,l) = \frac{w(v_j, v_l)}{\sum_{v_{l'} \in \mathcal{N}(v_j)} w(v_j, v_{l'})},$$

where $\mathcal{N}(v_j)$ is the set of neighbors of node v_i. The transition probability $a_\mathcal{Q}(i,k)$ is also defined in a similar way.

We model the semi-Markov random walk on $\mathcal{G}_\mathcal{X}$ such that $\mu(x)$, the expected amount of time that the random walker spends at a node pair $x = (q_i, v_j)$, is proportional to the node similarity $h(q_i, v_j)$. Using this model, we can measure the *global correspondence score* between q_i and v_j based on the long-run proportion of time that the random walker stays at the node pair $x = (q_i, v_j)$. According to this model, both higher interaction similarity as well as higher node similarity will increase the long-run proportion of time that the random walker stays simultaneously at q_i and v_j, making it a good measure for assessing the global similarity between nodes in different networks. As shown in Sahraeian and Yoon (2011a,b), the global correspondence score described above can be computed as follows:

$$\begin{aligned} s(q_i, v_j) &= \frac{\pi_\mathcal{X}(x)\mu(x)}{\sum_{x' \in \mathcal{V}_\mathcal{X}} \pi_\mathcal{X}(x')\mu(x')} \\ &= \frac{\pi_\mathcal{Q}(q_i)\pi(v_j)h(q_i, v_j)}{\sum_{i'=1}^{N_\mathcal{Q}} \sum_{j'=1}^{N} \pi_\mathcal{Q}(q_{i'})\pi(v_{j'})h(q_{i'}, v_{j'})}, \end{aligned} \tag{1}$$

where $\pi_\mathcal{X}$ is the steady state distribution of the Markov random walk on $\mathcal{G}_\mathcal{X}$, whose transition probability matrix is $\mathcal{A}_\mathcal{X}$. According to a property of the product graph (Vishwanathan *et al.*, 2010), we have $\pi_\mathcal{X} = \pi_\mathcal{Q} \otimes \pi$, where $\pi_\mathcal{Q}$ and π are the steady state distributions of the random walks on $\mathcal{G}_\mathcal{Q}$ and $\mathcal{G}$, respectively. We can compute the distributions $\pi_\mathcal{Q}$ and π by respectively finding the eigenvectors of the transition matrices $\mathcal{A}_\mathcal{Q}$ and $\mathcal{A}$ with unit eigenvalue. For graphs that consist of multiple disconnected subgraphs, we find the steady state distributions for each connected subgraph separately. Besides, we also add a self-transition edge to every isolated node in $\mathcal{G}$ and $\mathcal{G}_\mathcal{Q}$. We can conveniently rewrite (1) as:

$$\mathbf{S} = \frac{\mathbf{P} \circ \mathbf{H}}{\text{trace}(\mathbf{P}\mathbf{H}^T)}, \tag{2}$$

where $\mathbf{S}$, $\mathbf{H}$, and $\mathbf{P}$ are $N_\mathcal{Q} \times N$-dimensional matrices such that $\mathbf{S}[i,j] = s(q_i, v_j)$, $\mathbf{H}[i,j] = h(q_i, v_j)$, and $\mathbf{P}[i,j] = \pi_\mathcal{Q}(q_i)\pi(v_j)$, and $\circ$ denotes the Hadamard (or element-wise) product.

As we have discussed, the SMRW model provides us with a computationally efficient method for computing probabilistic node correspondence scores that sensibly integrate node similarities and topological similarities. The estimated scores are used for iterative network reduction, which will be described in the next subsection. It is worth noting that a similar random walk based scheme was recently proposed by Singh *et al.* (2008) in the context of global network alignment, which served as the basis of two popular network alignment algorithms called IsoRank (Singh *et al.*, 2008) and IsoRankN (Liao *et al.*, 2009). In this approach, the similarity between nodes in two different PPI networks are measure by solving a matrix equation, where the resulting similarity scores can be viewed as the stationary probabilities of a random walk with restart on the product graph (Yoon *et al.*, 2012). The transitions in this random walk are governed by the transition probability matrix $\mathcal{A}_\mathcal{X}$ while the restart probability is governed by the node similarity score matrix $\mathbf{H}$. One practical limitation of this scheme is the high computational complexity for computing the stationary probabilities, which requires us to find the eigenvector of a huge $N_\mathcal{X}$-dimensional square matrix, where $N_\mathcal{X}$ is the number of nodes in the product graph $\mathcal{G}_\mathcal{X}$. On the other hand, the SMRW scheme decouples the problem of finding stationary probabilities of the random walk on the product graph $\mathcal{G}_\mathcal{X}$ into two subproblems, each of which requires the computation of the stationary probabilities of the random walk on a single graph, as shown in (1) and (2). This remarkably reduces the overall cost for computing the global node correspondence scores. For instance, for a query complex with $N_\mathcal{Q}$ nodes and $m_\mathcal{Q}$ edges, and the target network with N nodes and m edges, the time complexity of the random walk scheme proposed in Singh *et al.* (2008) is $O(m \cdot m_\mathcal{Q})$, while that of the proposed SMRW scheme is $O(m + m_\mathcal{Q} + z)$, where z is the number of non-zero elements in $\mathbf{H}$ (which corresponds to the number of potential homologues across the two networks). $\mathbf{H}$ is a $N_\mathcal{Q} \times N$ matrix, but in practice, it is highly sparse, hence $z \ll NN_\mathcal{Q}$. The difference between the two schemes, in terms of complexity, will become more prominent for larger networks.

2.2 Iterative network reduction

To search for the best matching subnetwork region in the target network, we take a reduction-based approach, in which we shrink the search space through iterative network reduction. In this scheme, we repeatedly reduce the size of the target network by discarding the nodes with the lowest affinity to the query nodes, which is reflected in the global node correspondence scores measured by the SMRW model. In each iteration, we update the correspondence scores based on the reduced target network to re-estimate the affinity between nodes across the two networks. The main motivation underlying this iterative "reduction and re-estimation" technique is that the estimated node correspondence scores in the $\mathbf{S}$ matrix tend to be less reliable when there exist a large number of irrelevant nodes in the target network. To tackle this problem, we iteratively filter out the least relevant nodes in the target network and re-estimate the correspondence scores so that the reliability of the estimated scores are successively refined through the iterations. In general, the proposed technique improves the reliability of the estimated node correspondence scores and lead to more accurate querying results, as will be discussed in this subsection and also demonstrated based on actual querying experiments in the results section.

Before performing the iterative network reduction, we discard any potential non-homologous nodes from the target network, such that $v_j \in \mathcal{V}$ and all its edges are removed if $\forall q_i \in \mathcal{V}_\mathcal{Q} : \mathbf{S}[i,j] < T$, where T is a given threshold. We denote the remaining network as $\mathcal{G}'$. This pruning step can reduce false positive errors and improve the overall speed of the querying algorithm. At the beginning of the reduction process, we let $\mathcal{G}^{(0)} = \mathcal{G}'$ to be the initial network. Then, in the k^{th} iteration, we update $\mathcal{G}^{(k-1)} = (\mathcal{V}^{(k-1)}, \mathcal{E}^{(k-1)})$ to $\mathcal{G}^{(k)} = (\mathcal{V}^{(k)}, \mathcal{E}^{(k)})$ as follows. First of all, we (re)estimate the correspondence scores $\mathbf{S}^{(k)}$using (2) based on the network $\mathcal{G}^{(k-1)}$. Next, we select the node $v_d \in \mathcal{V}^{(k-1)}$ in $\mathcal{G}^{(k-1)}$ with the lowest correspondence score to the query network as described in the following. Let a^* be the (unknown) optimal mapping between the query network $\mathcal{G}_\mathcal{Q}$ and the best-matching subnetwork in the target $\mathcal{G}$. We define the minimal correspondence node v_d as:

$$\begin{aligned} d &= \underset{j}{\text{argmin}}\, P(v_j \in a^*) = \underset{j}{\text{argmin}}\, P(\exists q_i \in \mathcal{V}_\mathcal{Q}, q_i \sim v_j \in a^*) \\ &= \underset{j}{\text{argmin}} \left(1 - \prod_{q_i \in \mathcal{V}_\mathcal{Q}} [1 - P(q_i \sim v_j \in a^*)]\right) \\ &\simeq \underset{j}{\text{argmin}} \left(1 - \prod_{q_i \in \mathcal{V}_\mathcal{Q}} [1 - \mathbf{S}^{(k)}[i,j]]\right), \end{aligned}$$

where we view the correspondence score $\mathbf{S}^{(k)}[i,j]$ (estimated in the current iteration k) as an estimator of $P(q_i \sim v_j \in a^*)$, which is the posterior probability that the query node q_i will be mapped to the target node v_j in the optimal mapping a^*. According to (3), v_d will be the node that is least likely to be included in the best matching subnetwork region. Thus, we discard v_d and its edges from $\mathcal{G}^{(k-1)}$ to obtain $\mathcal{G}^{(k)}$, unless v_d is the only homologous node (in the current target network) for some query node q_i. More precisely, if there exists a $q_i \in \mathcal{V}_\mathcal{Q}$ such that $\mathbf{S}^{(k)}[i,d] > 0$

and $\mathbf{S}^{(k)}[i,j] = 0$ for all $j \neq d$, we keep v_d in the network and check the next candidate node. This reduces the risk of discarding a true match with a relatively low correspondence score. After the node removal at each iteration, we also discard all isolated nodes $\mathcal{V}_{\mathcal{O}} = \{v_{d'_1}, v_{d'_2} \cdots\}$, unless the isolated node is the only homologous node of a query node. The set $\mathcal{V}_{\mathcal{O}}$ can be easily identified, as it simply consists of nodes in $\mathcal{V}^{(k-1)}$ whose only neighbor is v_d. Following this reduction step, we recompute the correspondence scores $\mathbf{S}^{(k+1)}$ based on the reduced network, to be used in the next iteration. We repeat the network reduction and score re-estimation process until we have $|\mathcal{V}^{(k)}| \leq 2N_{\mathcal{Q}}$. Figure S1 (see Supplementary Data) illustrates the network reduction process. In this example, the original target network $\mathcal{G}^{(0)}$ undergoes two reduction cycles until the matching subnetwork $\mathcal{G}^{(2)}$ is identified.

As mentioned earlier, the proposed network reduction scheme can improve the accuracy of the estimated node correspondence scores and thereby enhance the expected accuracy of the final querying result. To define the querying accuracy, let A represent the space of all possible mappings between the query and the target networks, where a mapping a uniquely maps the query nodes $\mathcal{V}_{\mathcal{Q}}$ to a subnetwork region in the target network. As before, let $a^* \in A$ be the true (unknown) mapping between $\mathcal{G}_{\mathcal{Q}}$ and $\mathcal{G}$. We can define the *accuracy* of a mapping a (obtained as a result of network querying) with respect to the true mapping a^* as follows:

$$\text{accuracy}(a, a^*) = \frac{1}{N_{\mathcal{Q}}} \sum_{q_i \sim v_j \in a} \mathbf{1}\{q_i \sim v_j \in a^*\},$$

where $q_i \sim v_j$ indicates the alignment between $q_i \in \mathcal{V}_{\mathcal{Q}}$ and $v_j \in \mathcal{V}$, and $\mathbf{1}\{\cdot\}$ is the indicator function whose value is 1 if the argument is true, and 0 otherwise. In practice, the true mapping a^* is unknown. Therefore, instead of assessing the accuracy defined in (2.2), we estimate the *expected accuracy* as follows:

$$\begin{aligned}\mathbf{E}_{a^*}[\text{accuracy}(a, a^*)] &= \frac{1}{N_{\mathcal{Q}}} \sum_{q_i \sim v_j \in a} \mathbf{E}_{a^*}[\mathbf{1}\{q_i \sim v_j \in a^*\}] \\ &= \frac{1}{N_{\mathcal{Q}}} \sum_{q_i \sim v_j \in a} P(q_i \sim v_j \in a^*), \qquad (3)\end{aligned}$$

where $P(q_i \sim v_j \in a^*)$ is the *posterior probability* of that q_i will be mapped to v_j in the true mapping a^*. Based on this setting, the goal of network querying would be to find the optimal mapping $a \in A$ that has the maximum expected accuracy. Note that the concept of maximum expected accuracy was previously adopted by a number of multiple sequence alignment algorithms (Sahraeian and Yoon, 2010; Do *et al.*, 2005), where it has been shown to be an effective framework for predicting accurate alignments.

Now, let us consider a bipartite graph with the set of nodes $\mathcal{V}_{\mathcal{Q}} \bigcup \mathcal{V}$ and a set of weighted edges that link all node pairs (q_i, v_j), where $q_i \in \mathcal{V}_{\mathcal{Q}}$ and $v_j \in \mathcal{V}$. Suppose that the weight of an edge that connects the node pair (q_i, v_j) is assigned as $\omega_{i,j} = P(q_i \sim v_j \in a^*)$. Under this setting, the problem of finding the maximum expected accuracy mapping can be translated into a maximum weight bipartite matching (MWM) problem. For each query node q_i, let us denote its (unknown) true matching node in the target network as v_{i^*} (i.e., $q_i \sim v_{i^*} \in a^*$). Consider an optimization problem where the goal is to minimize the objective function $f(\mathbf{S}) = \sum_{i=1}^{N_{\mathcal{Q}}} \mathbf{S}[i, i^*]$, constrained on the constant L_1 norm of $\mathbf{S}$ (i.e., $||\mathbf{S}||_1 = 1$). As the global node correspondence score $\mathbf{S}[i,j]$ can serve as a good estimate of the posterior probability $P(q_i \sim v_j \in a^*)$, the objective function $f(\mathbf{S})$ is proportional to the expected accuracy defined in (3). Therefore, the maximum weighted matching on the bipartite graph with edge weights $\omega_{i,j} = \mathbf{S}^*[i,j]$, where $\mathbf{S}^* = \underset{\mathbf{S}}{\text{argmax}}\, f(\mathbf{S})$, will lead us to the maximum expected accuracy solution for the querying problem. Note that the MWM problem can be efficiently solved in polynomial time, using the well-known *Hungarian* algorithm (Kuhn, 1955). As shown in the Supplementary Data, the proposed network reduction scheme improves $f(\mathbf{S})$ at each iteration, which demonstrates that the iterative reduction–re-estimation process can improve the reliability of the node correspondence scores and thereby lead to more accurate querying results.

2.3 Identification of the best matching subnetwork

After the network reduction process, we are left with a reduced target network $\mathcal{G}^{(\infty)}$ with a relatively small size (with no more than $2N_{\mathcal{Q}}$ nodes). The remaining step is to detect the subnetwork of $\mathcal{G}^{(\infty)}$ that consists of the nodes that best match the query nodes. To identify the best matching subnetwork, we propose two different strategies, where each strategy has its own advantages. In both strategies, we consider a bipartite graph $\mathcal{G}_{\mathcal{B}}$ with the node set $\mathcal{V}_{\mathcal{Q}} \bigcup \mathcal{V}^{(\infty)}$ and a set of weighted edges between $q_i \in \mathcal{V}_{\mathcal{Q}}$ and $v_j \in \mathcal{V}^{(\infty)}$, whose weight is assigned as $\omega_{i,j} = \mathbf{S}^{(\infty)}[i,j]$. In the first strategy, we seek the maximum expected accuracy querying result. As discussed in the previous section, this is found by maximizing the objective function $f(\mathbf{S}^{(\infty)}) = \sum_{i=1}^{N_{\mathcal{Q}}} \mathbf{S}^{(\infty)}[i, i^*]$. As mentioned earlier, this corresponds to finding the maximum weighted matching of the bipartite graph $\mathcal{G}_{\mathcal{B}}$, which can be solved using the *Hungarian* algorithm (Kuhn, 1955). One disadvantage of this approach is that it does not guarantee that the querying result will be a *connected* subgraph of the target network. However, considering the incompleteness of the current PPI data, this strategy may be still desirable if we care more about maximizing the expected accuracy of the querying result rather than retrieving a connected subnetwork. We refer to this network querying scheme as RESQUE-M (for Maximum expected accuracy matching). In the second strategy, we search for the largest *connected* component $\mathcal{G}_c$ in the reduced target network $\mathcal{G}^{(\infty)}$ and report this network $\mathcal{G}_c$ as the final querying result. Since all remaining nodes in the reduced network $\mathcal{G}^{(\infty)}$ have high correspondence to the query nodes, $\mathcal{G}_c$ is guaranteed to be highly similar to the query network $\mathcal{G}_{\mathcal{Q}}$. We refer to this second network querying scheme as RESQUE-C (for largest Connected component). In case the target network contains two duplicated copies of the query network, both copies will be present in the reduced target network $\mathcal{G}^{(\infty)}$ at the end of the iterative reduction process, with high probability. RESQUE-C will return both copies in the final result, assuming they are connected. On the other hand, RESQUE-M will generally report only the node with a higher correspondence score, for each pair of duplicated nodes.

2.4 Materials

We assess the performance of RESQUE based on the PPI networks of *D. melanogaster*, *H. sapiens*, and *S. cerevisiae*, which are the three largest PPI networks that are currently available. We obtained the PPI data from IsoBase (Park *et al.*, 2011), a recently published database of functionally related proteins across PPI networks. IsoBase consists of several PPI networks that belong to different species, along with the homology scores between all pairs of proteins across species (measured in terms of the BLAST bit-value similarity between the protein sequences). The PPI networks in IsoBase have been constructed by integrating the PPI data from three different public databases: DIP (Salwinski *et al.*, 2004), BioGRID (Stark *et al.*, 2011), and HPRD (Keshava Prasad *et al.*, 2009). Currently, the *D. melanogaster* PPI network in IsoBase contains 14,098 proteins and 26,726 interactions, the *H. sapiens* network contains 22,369 proteins and 43,757 interactions, and the *S. cerevisiae* network includes 6,659 proteins and 38,109 interactions. The query networks were obtained by taking protein complexes of size 4∼25 in the *D. melanogaster*, *H. sapiens*, and *S. cerevisiae* networks, as in (Bruckner *et al.*, 2010), and projecting each complex to the corresponding PPI network to find its induced subnetwork.

3 RESULTS

To investigate the performance of the proposed network querying algorithm, we conducted a set of querying experiments based on real PPI datasets as well as a number of simulated examples.

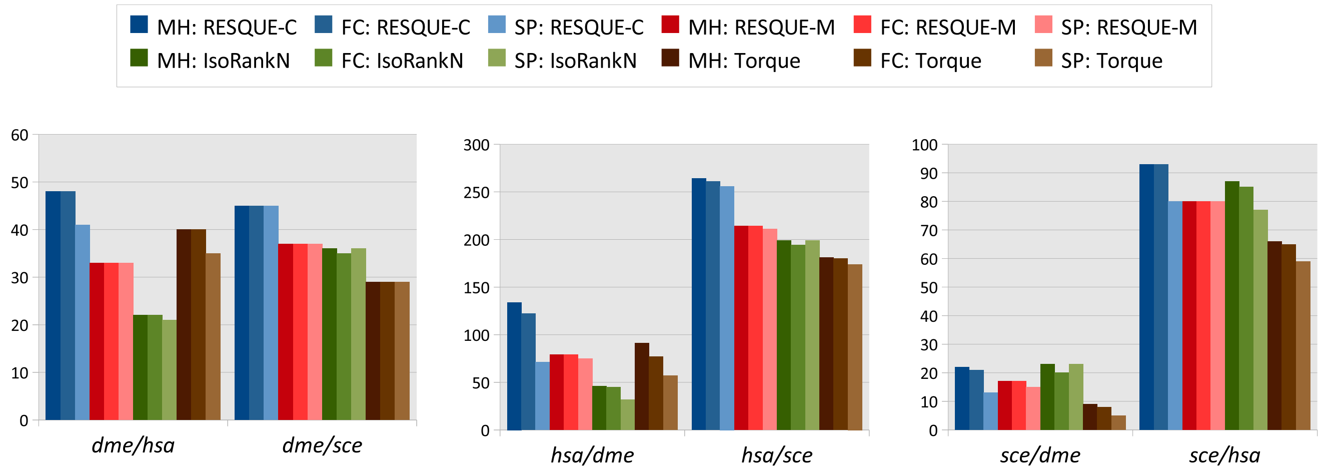

Fig. 3. Performance of different querying algorithms. For each query/target pair, we report the number of meaningful hits (MH), the number of functionally coherent hits (FC), and the number of specific hits (SP). (dme: *D. melanogaster*, hsa: *H. sapiens*, sce: *S. cerevisiae*).

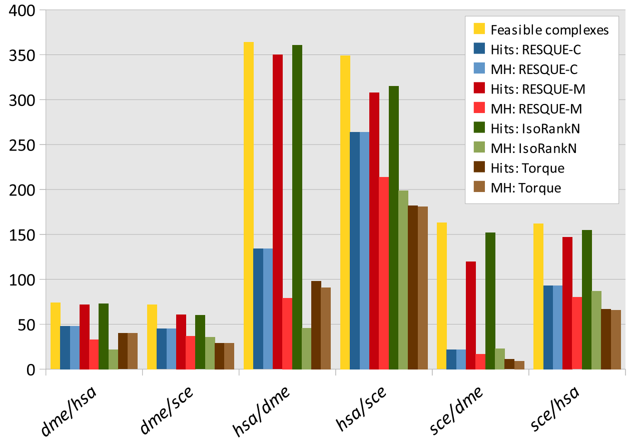

Fig. 2. Performance of different querying algorithms. For each query/target pair, we report the number of feasible hits, the number of hits, and the number of meaningful hits (MH). (dme: *D. melanogaster*, hsa: *H. sapiens*, sce: *S. cerevisiae*)

3.1 Querying performance on real PPI datasets

We first evaluated the performance of RESQUE by performing queries in the PPI networks of *D. melanogaster*, *H. sapiens*, and *S. cerevisiae*. We compared RESQUE against two state-of-the-art network querying algorithms: Torque (Bruckner *et al.*, 2010) and IsoRankN (Liao *et al.*, 2009). Torque adopts a topology-free approach, in which it searches for a connected set of proteins (in the target network) that are homologous to the query proteins, without taking the topology of the query network into account. We chose to compare RESQUE with Torque, as Torque has been shown (Bruckner *et al.*, 2010) to outperform other existing algorithms, such as QNet (Dost *et al.*, 2008). IsoRankN is a popular multiple network alignment algorithm, which uses spectral graph theory to evaluate the overall similarity between nodes across different networks and employs a spectral clustering scheme to find the best mapping between nodes. Although IsoRankN was mainly developed for network alignment, it can also be used for network querying (which can be viewed as a special case of network alignment). As the SMRW model adopted by RESQUE bears conceptual similarity to the spectral graph theory based approach used in IsoRankN, we decided to compare the querying performance of the two algorithms.

Note that both RESQUE-C and Torque always report a set of connected nodes as the query result. However, RESQUE-M and IsoRankN reports the set of best matching nodes that do not necessarily induce a connected subgraph in the target network. itmeaningful hits (MH) as the total number hits that include a connected subgraph with at least $N_Q/2$ nodes. Considering that the goal of network querying is to identify the target subnetwork that is most similar to the query, in terms of similarity between the constituent nodes as well as the network topology, it would make more sense to count such meaningful hits rather than counting all hits regardless of their size and topology (e.g., connectivity). Figure 2 shows the number of feasible hits (i.e., the total number of query complexes of size 4$\sim$25 that were used in our querying experiments), the number of hits, and the number of meaningful hits, for all pairs of query and target species. As we see from this result, almost all hits returned by RESQUE-C and Torque are strongly connected, while IsoRankN and RESQUE-M yield a larger number of hits that are loosely connected. Figure 2 also shows that, on average, RESQUE-C can identify meaningful hits for 51% query complexes, while RESQUE-M returns meaningful hits for 39% complexes, and both IsoRankN and Torque for 35% complexes. Also note that Torque typically has a smaller number of hits, as it can handle only a pre-specified number of indels (insertions/deletions).

To evaluate the accuracy of the querying algorithms, we measured the *functional coherence* and *specificity* of the querying results, as in (Bruckner *et al.*, 2010). Functional coherence measures the relative number of hits with significant functional coherence, which is assessed based on the Gene Ontology (GO) (Ashburner *et al.*, 2000) annotation. We used the GO TermFinder (Boyle *et al.*, 2004)

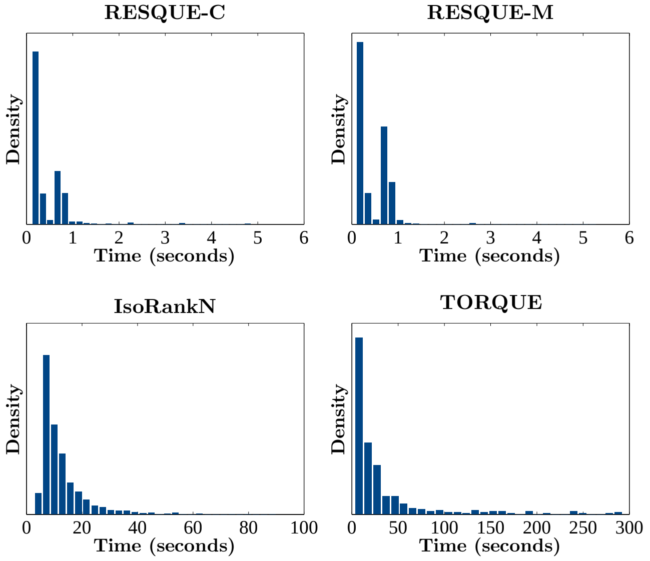

Fig. 4. Distribution of the amount of computational time that is needed by different network querying algorithms to complete a network query.

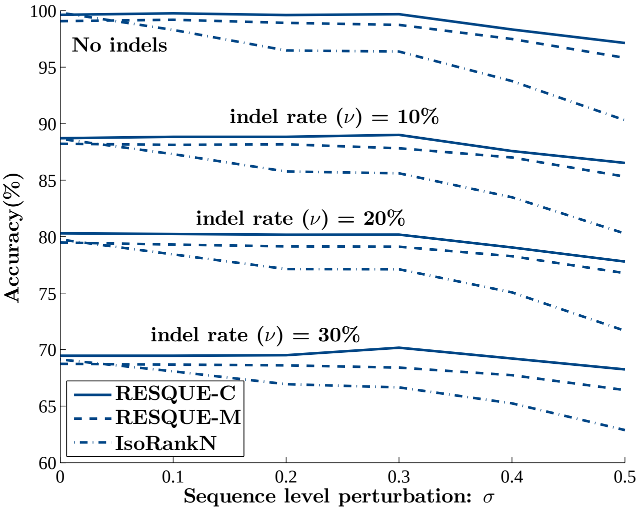

Fig. 5. Performance on simulated data. The querying accuracy is measured in terms of the average percentage of correctly predicted node matches, at four different indel rates and different sequence perturbation levels.

to compute the false discovery rate (FDR) corrected p-value of the functional coherence of the proteins in the retrieved target subnetwork. The specificity is computed based on the relative number of hits that significantly overlap with a known protein complex. We used a similar statistical procedure as in Bruckner *et al.* (2010) to measure the significance of the overlap. Figure 3 shows the performance of the respective querying algorithms for different pairs of query and target species, in terms of the number of meaningful hits (MH), the number of functionally coherent hits (FC), and the number of specific hits (SP). As we can observe, all four algorithms have similar levels of specificity (SP/MH > 0.8) and functional coherence (FC/MH > 0.9). However, the results clearly show that RESQUE-C yields the largest FC and SP. Considering the high functional coherence of the hits reported by RESQUE, the meaningful hits that do not overlap with known protein complexes may correspond to novel complexes. RESQUE-C and RESQUE-M respectively reported 100 and 58 novel MH with no overlap with known complexes, whereas IsoRank and Torque reported 26 and 57 novel hits, respectively.

We also carried out a similar performance comparison between RESQUE and a preliminary implementation of the reduction-based network querying algorithm presented in (Sahraeian and Yoon, 2011a), whose results can be found in the Supplementary Data (see Section S2).

3.2 Computational complexity

The proposed network querying scheme is highly efficient and has only a polynomial computational complexity in terms of the size of both networks. As shown in the Supplementary Data (see Section S3), the computational complexity is $O(mN + zN + N_Q^3)$ for RESQUE-C and $O(mN + zN + N_Q^2 \log(N_Q))$ for RESQUE-M, where N_Q is the number of query nodes, N is the number of nodes and m is the number of edges in the target network, and z is the number of homologues across the two networks (i.e., number of non-zero elements in similarity matrix $\mathbf{H}$). In practice, $\mathbf{H}$ is highly sparse, thus $z \ll N \times N_Q$. Considering that the computational complexity of many existing network querying algorithms increases exponentially with the size of the query network, RESQUE can provide a practical solution for searching networks with large queries. Figure 4 shows the distribution of the computational time needed by the respective network querying algorithms for performing the queries described in Section 3.1. As we can see in Figure 4, both RESQUE-C and RESQUE-M need only a few seconds to complete a network query. In contrast, IsoRankN often needs tens of seconds to complete a single query, and Torque also needs tens to hundreds of seconds for many queries. In fact, for some large protein complexes, it took even longer than an hour for Torque to obtain the querying result (see Figure S5 in the Supplementary Data).

3.3 Querying performance on simulated data

In order to evaluate the robustness of the proposed network querying scheme to changes in the node similarity scores as well as the topology of the query network, we conducted further querying experiments based on a simulated examples. In these experiments, we randomly extracted 100 connected subnetworks with general topology from the *S. cerevisiae* PPI network, where the size of each subnetwork ranged between 5 and 20. We applied two different types of perturbations to each query network. First, we considered a topological perturbation, where ν% of the query nodes were randomly deleted and inserted. Second, we perturbed the BLAST bit-value similarity scores by scaling the original similarity score by a random number drawn from a Gaussian distribution with mean $\mu = 1$ and variance σ^2.

Figure 5 shows the querying accuracy of different algorithms, for four different indel rates ($\nu = 0, 10\%, 20\%,$ and 30%) and various sequence perturbation levels ($\sigma \in [0, 0.5]$). As we can observe, RESQUE is highly robust against changes in the node similarity scores. Furthermore, Figure 5 also shows that RESQUE is robust to topological changes that involve node deletions and insertions. The results clearly demonstrate the effectiveness of the SMRW model and the reduction-based approach in identifying the true matching nodes across networks. In all cases, RESQUE outperformed IsoRankN, in terms of both accuracy and robustness. For comparison, we also performed similar experiments using Torque (results not shown). These experiments showed that Torque is almost invariant to changes in sequences similarity scores, which is expected as Torque uses the similarity score just to determine the presence (or absence) of homology between proteins. However, Torque typically showed lower accuracy compared to RESQUE and IsoRank, which was 88.8% for $\nu = 0\%$, 79.13% for $\nu = 10\%$, 72.26% for $\nu = 20\%$, and 60.75% for $\nu = 30\%$.

3.4 Example querying results for known pathways and molecular complexes

Here, we present several querying examples for known signaling pathways and molecular complexes. Figure 6A illustrates the result of querying the MAPK signaling pathway of *S. cerevisiae* (obtained from the KEGG database (Kanehisa and Goto, 2000)) in the PPI network of *H. sapiens*. The querying result is in good agreement with the MAPK signaling pathway of *H. sapiens*, and the identified subnetwork shows high functional coherence (p-value of 1.91e-11), measured based on GO annotations. The next example in Figure 6B represents the result of querying the proteasome complex of *H. sapiens* in the *S. cerevisiae* PPI network. The querying result closely matches the proteasome core complex in *S. cerevisiae* with high GO enrichment (p-value of 3.65e-42). The next three examples shown in Figures 6C, D,E are three instances where RESQUE could successfully predict the matching subnetwork, whereas Torque was not able to find a hit for the given query. Figure 6C illustrates the result of querying the prefoldin complex of *S. cerevisiae* in the *D. melanogaster* PPI network. The matching proteins reported by RESQUE belong to the same complex in *D. melanogaster* with high GO enrichment (p-value of 2.87e-12). Next, Figure 6D shows the querying result of the transcription factor TFIID complex of *D. melanogaster* in the *S. cerevisiae* PPI network. RESQUE was able to identify the matching TFIID proteins with high GO enrichment with a p-value of 1.41e-15. Finally, Figure 6E illustrates the result of querying the RNA polymerase complex of *S. cerevisiae* in the PPI network of *H. sapiens*, where the retrieved subnetwork closely matched the query network and contained proteins that are functionally coherent (p-value of 8.30e-29). These examples clearly show that RESQUE can efficiently search biological networks based on queries that have general topology, effectively handle node insertions and deletions, and yield biologically significant results. In Tables S1-S6 (see Supplementary Data), we also report a list of novel querying results obtained by RESQUE-C for queries for which IsoRankN and Tourque failed to identify meaningful hits.

4 CONCLUSION

In this paper, we proposed RESQUE, a novel network querying algorithm that can efficiently query biological pathways and molecular complexes in large-scale biological networks. The algorithm uses a semi-Markov random walk (SMRW) model to estimate probabilistic correspondence scores between nodes across different networks, which are used to identify the best match to the given query according to the maximum expected accuracy principle. RESQUE adopts an iterative network reduction and score re-estimation technique to improve the expected accuracy of the final querying result. As discussed in this paper, RESQUE does not restrict the topology of the query network, and it can handle paths, trees, and loopy graphs. The algorithm supports both connected and partially connected query networks. In the extreme case, RESQUE can also be used with a query pathway (or complex) with an unknown topology, in which case we can simply treat the query as a collection of isolated nodes. In this case, the steady state distribution of the random walk on the query network, π_Q, will be uniform over all the query nodes. Furthermore, RESQUE can effectively deal with node insertions and deletions at arbitrary locations. Despite its generality, RESQUE has very low computational complexity, which allows us to use the algorithm for querying large pathways/complexes in genome-scale networks. Experimental results based on real and synthetic examples show that RESQUE outperforms other popular algorithms, in terms of speed, accuracy, and robustness.

Although we have not explored the application of RESQUE to directed networks, it is worth noting that such extension would be relatively straightforward. In fact, in order to incorporate the edge directions when estimating the node correspondence scores, we only have to require the semi-Markov random walker to follow these directions during the random walk (i.e., by choosing the next node among the neighbors that are connected via outgoing edges).

ACKNOWLEDGMENTS

The authors would like to thank the authors of (Bruckner *et al.*, 2010) for providing the source code of Torque and the data files used in their paper.

Funding: This work was supported in part by the National Science Foundation, through NSF Award CCF-1149544.

REFERENCES

Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H. *et al.* (2000) Gene Ontology: Tool for the unification of biology. the gene ontology consortium. *Nat Genet*, **25**, 25–29.

Ay, F., Kellis, M. and Kahveci, T. (2011) SubMAP: aligning metabolic pathways with subnetwork mappings. *J. Comput. Biol.*, **18**, 219–235.

Barabasi, A. L. and Oltvai, Z. N. (2004) Network biology: understanding the cell's functional organization. *Nat. Rev. Genet.*, **5**, 101–113.

Blin, G., Sikora, F. and Vialette, S. (2010a) GraMoFoNe: a Cytoscape plugin for querying motifs without topology in Protein-Protein Interactions networks. In Al-Mubaid, H. (ed.), *2nd International Conference on Bioinformatics and Computational Biology (BICoB'10)*, International Society for Computers and their Applications (ISCA), pp. 38–43. Honolulu, États-Unis.

Blin, G., Sikora, F. and Vialette, S. (2010b) Querying graphs in protein-protein interactions networks using feedback vertex set. *IEEE/ACM Trans Comput Biol Bioinform*, **7**, 628–635.

Boyle, E. I., Weng, S., Gollub, J., Jin, H., Botstein, D., Cherry, J. M. and Sherlock, G. (2004) GO::TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics*, **20**, 3710–3715.

Bruckner, S., Huffner, F., Karp, R. M., Shamir, R. and Sharan, R. (2010) Topology-free querying of protein interaction networks. *J. Comput. Biol.*, **17**, 237–252.

Cusick, M. E., Klitgord, N., Vidal, M. and Hill, D. E. (2005) Interactome: gateway into systems biology. *Hum. Mol. Genet.*, **14 Spec No. 2**, R171–181.

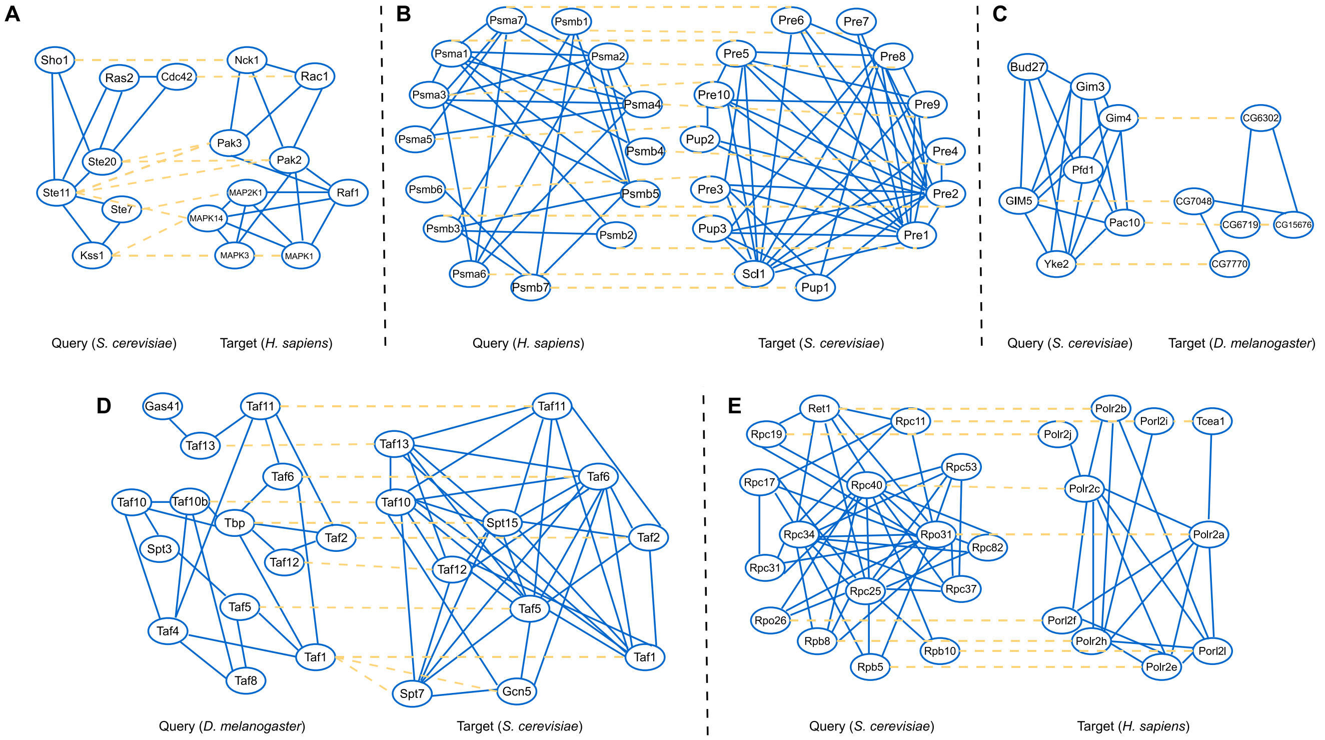

Fig. 6. Querying examples. (A) Querying the MAPK signaling pathway of *S. cerevisiae* in *H. sapiens*. Matching nodes are connected by dashed lines. (B) Querying the proteasome complex of *H. sapiens* in *S. cerevisiae*. (C) Querying the prefoldin complex of *S. cerevisiae* in *D. melanogaster*. (D) Querying the transcription factor TFIID complex of *D. melanogaster* in *S. cerevisiae*. (E) Querying the RNA polymerase complex of *S. cerevisiae* in *H. sapiens*.

Do, C. B., Mahabhashyam, M. S., Brudno, M. and Batzoglou, S. (2005) ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Res.*, **15**, 330–340.

Dost, B., Shlomi, T., Gupta, N., Ruppin, E., Bafna, V. and Sharan, R. (2008) QNet: A tool for querying protein interaction networks. *J Comput Biol*, **15**, 913–925.

Durand, P., Labarre, L., Meil, A., Divo, J. L., Vandenbrouck, Y., Viari, A. and Wojcik, J. (2006) GenoLink: a graph-based querying and browsing system for investigating the function of genes and proteins. *BMC Bioinformatics*, **7**, 21.

Ferraro, N., Palopoli, L., Panni, S. and Rombo, S. E. (2011) Asymmetric comparison and querying of biological networks. *IEEE/ACM Trans Comput Biol Bioinform*, **8**, 876–889.

Ferro, A., Giugno, R., Pigola, G., Pulvirenti, A., Skripin, D., Bader, G. D. and Shasha, D. (2007) NetMatch: a Cytoscape plugin for searching biological networks. *Bioinformatics*, **23**, 910–912.

Fionda, V. and Palopoli, L. (2011) Biological network querying techniques: analysis and comparison. *J. Comput. Biol.*, **18**, 595–625.

Fionda, V., Palopoli, L., Panni, S. and Rombo, S. E. (2008) Protein-protein interaction network querying by a "focus and zoom" approach. In Elloumi, M., Kung, J., Linial, M., Murphy, R. F., Schneider, K. and Toma, C. (eds.), *Bioinformatics Research and Development*, volume 13 of *Communications in Computer and Information Science*, pp. 331–346. Springer Berlin Heidelberg.

Ge, H. (2000) UPA, a universal protein array system for quantitative detection of protein-protein, protein-DNA, protein-RNA and protein-ligand interactions. *Nucleic Acids Res.*, **28**, e3.

Gulsoy, G. and Kahveci, T. (2011) RINQ: Reference-based Indexing for Network Queries. *Bioinformatics*, **27**, i149–i158.

Ho, Y., Gruhler, A., Heilbut, A., Bader, G. D. *et al.* (2002) Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature*, **415**, 180–183.

Huang, M., Ding, S., Wang, H. and Zhu, X. (2008) Mining physical protein-protein interactions from the literature. *Genome Biol.*, **9 Suppl 2**, S12.

Kanehisa, M. and Goto, S. (2000) KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, **28**, 27–30.

Kelley, B. P., Yuan, B., Lewitter, F., Sharan, R., Stockwell, B. R. and Ideker, T. (2004) PathBLAST: a tool for alignment of protein interaction networks. *Nucleic Acids Res.*, **32**, W83–88.

Keshava Prasad, T. S., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S. *et al.* (2009) Human Protein Reference Database–2009 update. *Nucleic Acids Res.*, **37**, D767–772.

Kuhn, H. W. (1955) The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, **2**, 83–97.

Liao, C. S., Lu, K., Baym, M., Singh, R. and Berger, B. (2009) IsoRankN: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, **25**, i253–258.

Mongiovi, M., Di Natale, R., Giugno, R., Pulvirenti, A., Ferro, A. and Sharan, R. (2010) SIGMA: a set-cover-based inexact graph matching algorithm. *J Bioinform Comput Biol*, **8**, 199–218.

Park, D., Singh, R., Baym, M., Liao, C. S. and Berger, B. (2011) IsoBase: a database of functionally related proteins across PPI networks. *Nucleic Acids Res.*, **39**, 295–300.

Pinter, R., Rokhlenko, O., Yeger-Lotem, E. and Ziv-Ukelson, M. (2005) Alignment of metabolic pathways. *Bioinformatics*, **21**, 3401–3408.

Qian, X., Sze, S. H. and Yoon, B.-J. (2009) Querying pathways in protein interaction networks based on hidden Markov models. *Journal of Computational Biology*, **16**, 145–157.

Sahraeian, S. and Yoon, B.-J. (2011a) Fast network querying algorithm for searching large-scale biological networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 6008 –6011.

Sahraeian, S. and Yoon, B.-J. (2011b) A novel low-complexity hmm similarity measure. *Signal Processing Letters, IEEE*, **18**, 87 –90.

Sahraeian, S. M. and Yoon, B. J. (2010) PicXAA: greedy probabilistic construction of maximum expected accuracy alignment of multiple sequences. *Nucleic Acids Res.*, **38**, 4917–4928.

Salwinski, L., Miller, C. S., Smith, A. J., Pettit, F. K., Bowie, J. U. and Eisenberg, D. (2004) The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.*, **32**, D449–451.

Sharan, R. and Ideker, T. (2006) Modeling cellular machinery through biological network comparison. *Nat. Biotechnol.*, **24**, 427–433.

Shlomi, T., Segal, D., Ruppin, E. and Sharan, R. (2006) QPath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, **7**.

Singh, R., Xu, J. and Berger, B. (2008) Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proc. Natl. Acad. Sci. U.S.A.*, **105**, 12763–12768.

Stark, C., Breitkreutz, B. J., Chatr-Aryamontri, A., Boucher, L., Oughtred, R. *et al.* (2011) The BioGRID Interaction Database: 2011 update. *Nucleic Acids Res.*, **39**, 698–704.

Tian, Y., McEachin, R. C., Santos, C., States, D. J. and Patel, J. M. (2007) SAGA: a subgraph matching tool for biological graphs. *Bioinformatics*, **23**, 232–239.

Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S. *et al.* (2000) A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. *Nature*, **403**, 623–627.

Vishwanathan, S., Schraudolph, N. N., Kondor, R. and Borgwardt, K. M. (2010) Graph Kernels. *Journal of Machine Learning Research*, **11**, 1201–1242.

Wernicke, S. and Rasche, F. (2007) Simple and fast alignment of metabolic pathways by exploiting local diversity. *Bioinformatics*, **23**, 1978–1985.

Yang, Q. and Sze, S. (2007) Path matching and graph matching in biological networks. *J Comput Biol*, **14**, 56–67.

Yoon, B.-J., Qian, X. and Sahraeian, S. (2012) Comparative analysis of biological networks: Hidden markov model and markov chain-based approach. *Signal Processing Magazine, IEEE*, **29**, 22 –34.

Zhang, A. (2009) *Protein Interaction Networks: Computational Analysis*. Cambridge University Press, New York, NY, USA, 1st edition.