Systems biology

RESQUE: Network reduction using semi-Markov random walk scores for efficient querying of biological networks
Sayed Mohammad Ebrahim Sahraeian and Byung-Jun Yoon
Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843-3128, USA
Associate Editor: Mario Albrecht

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 article, we introduce an efficient algorithm for querying large-scale biological networks, called RESQUE. The proposed algorithm adopts a semi-Markov random walk (SMRW) 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

Supplementary Information: Supplementary data are available at Bioinformatics online.

Received on December 9, 2011; revised on June 7, 2012; accepted on June 8, 2012

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 et al. 2005). 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; Zhang et al. 2009). 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.

Motivation

The rapid expansion of large-scale biological networks has resulted in an increasing number of large-scale PPI networks. For instance, 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. 2009; PathBLAST (Kelley et al. 2006)) and help us better understand the structure and dynamics of different biological mechanisms. Recent technological advances have enabled high-throughput global measurement of protein–protein interactions (Ge et al. 2000; Ho et al. 2002; Uetz et al. 2001), 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. 2009). 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.

Motivation

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. 2004), and until now, various approaches have been proposed to make this problem computationally feasible (Kolipaka et al. 2005; Huang et al. 2009; Shlomi et al. 2009). Among them, QPath (Shlomi et al. 2009) and PathBLAST (Kelley et al. 2006) are some of the pioneering network querying schemes that identify 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. 2009) and QNet (Dost et al. 2008) adopted a color-coding-based approach. QPath (Shlomi et al. 2009) is restricted to querying linear pathways, while QNet (Dost et al. 2008) allows searching for trees and bounded tree-width graphs. Another recent algorithm, called PADAM (Hill et al. 2010), also used the color-coding scheme and
dynamic programming for network querying. 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 et al. (2009) proposed a hidden Markov model (HMM)-based scheme for querying linear paths. SIGMA (Mongiovì et al., 2011) and RINQ (Litisey and Kahveci, 2013) consider the problem of efficient querying in biological network databases using indexing schemes. Ferraro et al. (2013) 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., 2011), 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 GraMoToNC (Blin et al., 2010b) 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) (Blin et al. (2010); Host et al., 2009; Kelley et al., 2010; Qian et al., 2009; Shlomi et al., 2008) or simply view it as a collection of nodes by ignoring the underlying topology (Blin et al. (2010a); Bruckner et al. (2011)). 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) (Riccardi et al. (2010); Ferraro et al., 2011; 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 (Fiorida and Palopoli, 2011).

In this article, 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 (SMRW) 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 PPI network, represented by an undirected weighted graph $G = (V, E, w)$, where $V$ is the set of $N$ nodes that correspond to the proteins in the network, $E$ is the set of $M$ edges that represent the protein interactions and $w$ is a weight function $w : E \rightarrow \mathbb{R}$, representing the strength or reliability of an interaction. Let $G_Q = (V_Q, E_Q, w_Q)$ be the query network with a set $V_Q$ of $N_Q$ nodes, edge set $E_Q$ and interaction reliability $w_Q$. 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 $q_i \in V_Q$ and $v_j \in V$, 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 $G$ that is maximally similar to the given query network $G_Q$.

2.1 Estimation of probabilistic node correspondence scores through SMRW

The network querying problem can be viewed as a mapping problem, in which we assign each query node $q_i \in V_Q$ to one or more target nodes $v_j \in V$, 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. SMRW model can provide an effective means of obtaining such integrated similarity scores (Sahraei and Yoon, 2011).

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 SMRW, the walker may spend a random amount of time between each transition. Hence, we consider a ‘simultaneous’ SMRW on both query and target networks, as shown in Figure 2. The position of the walker is given by a pair of nodes $(q_i, v_j)$, where $q_i \in V_Q$ and $v_j \in V$. 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 2, this simultaneous random walk on two graphs $G_Q$ and $G$ is equivalent to a random walk on their product graph $G_X = (V_X, E_X)$ (Vishwanathan et al., 2011). In this product graph $G_X$, every node in $V_X$ corresponds to a node pair $(q_i, v_j)$, 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 $(q_i, q_k) \in E_Q$ and $(v_j, v_l) \in E$ are present in the original networks. As shown in Vishwanathan et al. (2011), the transition probability matrix of the underlying Markov chain for the random walk on $G_X$ can be computed as $A_X = A_Q \odot A$, where $\odot$ denotes the Kronecker product, and $A_Q = [\log(c_{ij})]$ and $A = [a(i,j)]$ are the transition probability matrices for the random walks on $G_Q$ and $G$, respectively. The transition probability $a(i,j)$ measures the normalized contribution from the neighboring node $v_j$ to the node $v_i$ based on the
interaction reliability \( w(i, j) \) as follows:

\[
\alpha(i, j) = \frac{w(i, j)}{\sum_{i'=1}^{N} w(i, i')},
\]

where \( N \) is the set of neighbors of node \( i \). The transition probability \( \alpha(i, j) \) is also defined in a similar way.

We model the SMRW on \( G_X \) such that \( \mu(x) \), the expected amount of time that the random walker spends at a node pair \( x = (i, j) \), is proportional to the node similarity \( \beta(i, j) \). 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 \( x = (i, j) \). 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 Saharabum and Vriend, 2010, the global correspondence score described above can be computed as follows:

\[
\pi G(i, j) = \frac{\mu(x) \pi G(x)}{\sum_{i'=1}^{N} \mu(x') \pi G(x')},
\]

where \( \pi G(x) \) is the steady-state distribution of the Markov random walk on \( G_X \), whose transition probability matrix is \( A_X \). According to a property of the product graph (Coifman et al., 2005), we have \( \pi G = \pi G Q \pi \), where \( \pi G \) and \( \pi \) are the steady-state distributions of the random walks on \( G_X \) and \( G \), respectively. We can compute the distributions \( \pi G \) and \( \pi \) by, respectively, finding the eigenvectors of the transition matrices \( A_Q \) and \( A \) 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 \( G \) and \( G_Q \). We can conveniently rewrite \( G \) as

\[
S = P \circ \text{Hadamard}(P H),
\]

where \( S, P, H \) and \( Q \) are \( N_Q \times N \)-dimensional matrices such that \( j S(i, j) = \pi G(i, j) \), \( H(i, j) = \pi G(i, j) \), and \( P(j, j) = \pi G(i, j) \pi G(j, j) \) 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 is related to the stationary probability of a 'random walk with restart' on the product graph (Coifman et al., 2005). In this way, the similarity between nodes in two different PPI networks is measured 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 (Coifman et al., 2005). The transitions in this random walk are governed by the transition probability matrix \( A_Q \), while the restart probability is governed by the node similarity score matrix \( H \). 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 \( N_Q \times N_Q \) dimensional square matrix, where \( N_Q \) is the number of nodes in the product graph \( G_Q \). On the other hand, the SMRW scheme decouples the problem of finding stationary probabilities of the random walk on the product graph \( G_Q \) into two subproblems, each of which requires the computation of the stationary probabilities of the random walk on a single graph, as shown in B and C. This remarkably reduces the overall cost for computing the global node correspondence scores. For instance, for a query complex with \( N_Q \) nodes and \( m_Q \) edges, the time complexity of the random walk scheme proposed in Singh et al., 2008, is \( O(m_Q + N_Q \log N_Q) \), while that of the proposed SMRW scheme is \( O(m + m_Q + N_Q \log N_Q, \zeta) \), where \( \zeta \) is the number of non-zero elements in \( H \) (which corresponds to the number of potential homologues across the two networks). \( H \) is a \( N_Q \times N \) matrix, but in practice, it is highly sparse, hence \( \zeta \ll N_Q \). 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 underpinning this iterative ‘reduction and re-estimation’ technique is that the estimated node correspondence scores in the \( S \) 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 is successively refined through the iterations. In general, the proposed technique improves the reliability of the estimated node correspondence scores and leads to more accurate querying results, as will be discussed in this subsection and also demonstrated based on actual querying experiments in Section 3.

Before performing the iterative network reduction, we discard any potential non-homologous nodes from the target network, such that \( v_j \in V \) and all its edges are removed if \( Q(v_j) \in Q(V) \) or \( v_j \in T \), where \( T \) is a given threshold. We denote the remaining network as \( G' \). 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 \( G' = G_Q \) to be the initial network. Then, in the 6th iteration, we update \( O^{(6-1)} \) using (2) based on the network \( G^{(6-1)} \). Next, we select the node \( v_j \in V^{(6-1)} \) 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 \( G_Q \) and the best-matching subnetwork in the target \( G' \). We define the minimal correspondence node \( v_j \) as

\[
d = \arg \min_{v_j} P(v_j \in a^*) = \arg \min_{v_j} P(v_j \in Q(V), v_j \sim a^*),
\]

\[
= \arg \min_{v_j} \left( 1 - \prod_{\forall v_j \in Q(V)} \left( P(Q(v_j) \sim a^*) \right) \right)
\]

where we view the correspondence score \( S^{(6)}(i, j) \) (estimated in the current iteration \( k \)) as an estimator of \( P(Q(v_j) \sim a^*) \), 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 (2), \( v_j \) will be the node that is least likely to be included in the best matching subnetwork region. Thus, we discard \( v_j \) and its edges from \( G^{(6-1)} \) to obtain \( G^{(6)} \), unless \( v_j \) is the only homologous node (in the current target network) for some query node \( q_i \). More precisely, if there exists a \( q_i \in Q(V) \) such that \( S^{(6)}(i, j) = 0 \) and \( S^{(6)}(i, j) = 0 \) for all \( j \neq a^* \), we keep \( v_j \) 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 \( V = \{v_j | v_j \sim a^* \} \), unless the isolated node is the only homologous node of a query node. The set \( V = \{v_j \} \) can be easily identified, as it simply consists of nodes in \( V^{(6-1)} \) whose only neighbor is \( v_j \). Following this reduction step, we recompute the correspondence scores on the reduced network, to be used in the next iteration. We repeat the network reduction and score re-estimation process until we have
After the network reduction process, we are left with a reduced target network $G$, which demonstrates that the iterative reduction–re-estimation process can improve the reliability of the node correspondence scores and thereby lead to more accurate alignments.

We assess the performance of RESQUE based on the PPI networks of *Drosophila melanogaster*, *Homo sapiens*, and *Saccharomyces 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., 2006), and HPRD (Keshava Prasad et al., 2004). Currently, the *D. melanogaster* PPI network in IsoBase contains 14 058 proteins and 26 776 interactions, the *H. sapiens* network contains 22 369 proteins and 43 757 interactions, and the *S. cerevisiae* network includes 6 459 proteins and 38 199 interactions. The query networks were obtained by taking protein complexes of size 4–25 in the *D. melanogaster* network 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.

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., 2004). 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 to outperform other existing algorithms, such as QNet. IsoRankN is a popular multiple network alignment algorithm, which uses spectral graph theory to evaluate the overall similarity between nodes across different networks and uses 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 report the set of best matching nodes that do not necessarily induce a connected subgraph in the target network. For a fair and meaningful comparison among the different querying schemes, we define meaningful hits (MH) as the total number of 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 MH 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 4–25 that were used in our querying experiments), the number of hits and the number of MH, 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 MH for 51% query complexes, while RESQUE-M returns MH 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 (i.e. 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) annotation. We used the GO TermFinder 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 MH, the number of 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 MH that do not overlap with known protein complexes may correspond to novel complexes. RESQUE-C and
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 Section S3, the computational complexity is $O(mN + zN + NQ)$ for RESQUE-C and $O(mN + zN + NQ \log(NQ))$ for RESQUE-M, where $NQ$ 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 $H$). In practice, $H$ is highly sparse, thus $z \ll NQ$. 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 3 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 3, 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 Supplementary Figure S5).

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 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 $\%$ 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 $\sigma^2$.

Figure 4 shows the querying accuracy of different algorithms, for four different indel rates ($\nu = 0, 10, 20$ and $30\%$) and various sequence perturbation levels ($\sigma \in [0.05, 0.5]$). As we can observe, RESQUE is highly robust against changes in the node similarity scores. Furthermore, Figure 4 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 with 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

Herein, we present several querying examples for known signaling pathways and molecular complexes. Figure 5 illustrates the result of querying the Mitogen-activated protein kinase (MAPK) signaling pathway of $S. cerevisiae$ (obtained from the KEGG database...
RESQUE: Efficient Querying of Biological Networks

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

Also report a list of novel querying results obtained by RESQUE-C for queries for which IsoRankN and Torque failed to identify MH.

4 CONCLUSION

In this article, 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 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 article, 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, $\pi_Q$, 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 also show the effectiveness of RESQUE-C in handling node insertions and deletions.
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 SMRW to follow these directions during the random walk (i.e. by choosing the next node among the neighbors that are connected through outgoing edges).

ACKNOWLEDGMENTS

The authors thank the authors of Bruckner et al., Ferraro, N., et al., Bruckner, S. Conflict of Interest

Funding

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

Conflict of Interest: none declared.

REFERENCES