Asymmetric Global Alignment of Protein-Protein Interaction Graph Databases (Extended Abstract)

Nicola Ferraro¹, Luigi Palopoli¹, Simona Panni², and Simona E. Rombo¹

¹ DEIS, Università della Calabria
² Dept. of Cellular Biology, Università della Calabria

Abstract. In the last few years a large amount of protein interaction data has been collected and stored in public databases. The automatic analysis and management of such data can provide valuable information on the evolution of different organisms. To this aim, interaction data can be modelled as graphs where nodes represent cellular components and edges are associated to interactions. Such graphs are called protein-protein interaction (PPI) networks. A number of methods have been proposed to perform PPI-network alignment. Such methods operate symmetrically, that is to say, they do not assign a distinct role to the input PPI networks. However, in most cases, the input networks are indeed distinguishable on the basis of how well the corresponding organism is biologically well-characterized.

We propose a method for global alignment of PPI networks that exploits differences in the characterization of organisms at hand. We assume that the PPI network (called Master) of the best characterized is used as a fingerprint to guide the alignment process to the second input network (called Slave), so that generated results preferably retain the structural characteristics of the Master (and using the Slave) network. We tested our method showing that the results it returns are biologically relevant.

1 Introduction

A large amount of information on the interaction of cellular components has been recently stored and made available in public databases, such as BIOGRID [2] and DIP [14]. Such interaction data have been collected by both high-throughput technologies, including genome sequencing, expression profiling, cellular localization and other methods for large-scale protein-protein interactions, providing a deep characterization of few model organisms such as, for instance, the yeast Saccharomyces cerevisiae [7, 11]. A specific case of biological interaction data are the interactions among the proteins of a given organism, that can be modeled by a network, called protein-protein interaction (PPI) network, highlighting the mutual interactions between pairs of proteins. By comparing the PPI networks of different organisms the complex mechanisms at the basis of evolutionary conservations can be uncovered and the biological meaning of groups of interacting proteins belonging to not yet well characterized organisms can be thus inferred. As a result, a number of approaches have been recently presented in the literature for local [13, 4] and global [15, 16, 8, 10, 12] alignment of PPI networks.

In this context, the research presented here deals with global alignment of PPI networks. Global network alignment aims at finding a unique (possibly, the best) overall
alignment of the input networks, in such a way that all the nodes of the networks are mapped. Unfortunately, exact algorithms for PPI network global alignment cannot be afforded, inasmuch as the PPI network alignment problem can be reduced to subgraph isomorphism checking, that is known to be NP-complete [6] and, therefore, heuristic approaches are to be adopted.

A common characteristics of known methods for global alignment handle their input PPI networks symmetrically, that is to say, without considering if one organism is better characterized than another one. Indeed, while for well-characterized organisms, the associated PPI networks supposedly encode in a sound manner all the information about their proteins and associated interactions, this is far from being the case for not well characterized ones. Therefore, it seems sensible to devise methods for global alignment that in fact exploit differences in the characterization of the organisms at hand, which is precisely the main idea underlying this paper. In particular, in our approach, the PPI network (called Master) of the best characterized organism is used as a fingerprint to guide the alignment process to the second input network (called Slave), so that generated results preferably retain the structural characteristics of the Master network. This is obtained by generating from the Master (and using the Slave) a finite automaton, called alignment model, which is then fed with a (linearization of) the Slave network for the purpose of generating, via the Viterbi algorithm, matching subgraphs. In this way most of the structural information of the Master is kept, while only the Slave information useful to understand how much of the Master has been conserved in the Slave is exploited in the alignment process.

In more detail, the technique presented here aims to iteratively extract similar connected subgraphs from the input networks. The algorithm starts by searching for an initial seed, that is, a best pair of proteins \((p, q)\) (one from the first network and one from the second) to be matched. To this end, information about both protein sequence similarity and network topology are used\(^3\). Then, the seed is expanded to a pair of matching subgraphs of the two input networks by exploring the nodes adjacent to \(p\) and \(q\). When a new pair of connected subgraphs is eventually discovered, the two subgraphs are deleted from the input networks and the subgraph extraction procedure is started again. The process is iterated until no further solutions can be generated. The set of all the protein pairings resulting from the discovered subgraph matchings makes the global alignment between the input networks.

In order to assess the effectiveness of the approach, several experiments have been conducted over the PPI networks of \(\text{Saccharomyces cerevisiae}\) (yeast) and \(\text{Drosophila melanogaster}\) (fly). Experimental evaluations on these two networks demonstrate that our technique is able to find biologically significant subgraph pairings, some of which are not generated by other global alignment methods. Due to space constraints, we briefly discuss here only the main results obtained in the experimental campaign, while a detailed description of the validation and comparison is provided in the extended version of this work [3].

\(^3\) Other kinds of information about protein structure might be taken advantage of as well.
2 Preliminaries

A PPI network can be modeled as an indirect graph \( N = (P, I) \), where \( P \) is a set of nodes, each denoting a specific protein in the considered organism, and \( I \) is the set of edges representing protein-protein interactions. Nodes can be labeled by protein names or by database ids. Now, let us denote with \( a \circ b \) the concatenation of elements (or pairs) \( a \) and \( b \). Analogously, for elements (or pairs) \( a_1, a_2, \ldots, a_n \), \( a_1 \circ a_2 \circ \cdots \circ a_n \) denotes \((a_1 \circ (a_2 \circ ( \cdots \circ (a_{n-1} \circ a_n)))))\), and, for an ordered set \( A \), \( a_{\alpha \in A} \) denotes \((a_1 \circ (a_2 \circ ( \cdots \circ (a_{n-1} \circ a_n)))))\) where \( A = \langle a_1, a_2, \ldots, a_n \rangle \).

Furthermore, given a PPI network \( N = (P, I) \) and a node \( p \in P \), the adjacency set of \( p \) is the set \( \text{adj}(p) = \{q \in P | [p, q] \in I\} \) of nodes adjacent to \( p \).

Next we introduce the technical machinery useful to our purposes. We begin by modeling the Master network by defining its associated automata, called the alignment model, which is defined below.

Definition 1. (Alignment model) Let \( N_M = (P_M, I_M) \) and \( N_S = (P_S, I_S) \) be two PPI networks that we call Master and Slave, resp., and let \( k \) be an integer such that \( k \geq 1 \). Furthermore, let \( D \) be a set of triplets \( \langle p, q, s_{pq} \rangle \) and \( s_{ih} \) be a real value such that for \( p \in P_M \) and \( q \in P_S \), \( s_{pq} \) is the similarity value for \( p \) and \( q \) and \( s_{ih} \) is a threshold value.

Finally, let \( v \) and \( v' \) be two values such that \( v < v' \).

An alignment model \( M \) of order \( k \) for \( N_M \) w.r.t. \( N_S \) is a finite state automaton such that:

- the states of the automaton include one state for each protein in \( P_M \cup P_S \) and, moreover, states \( \beta \), \( \tau \), and a set of states \( \epsilon_i \) defined as follows;
- \( \beta \) is the initial state and it is linked to itself by a transition with value \( v \);
- the state \( \tau \) is linked to itself by a transition with value \( v \);
- each node of \( P_M \) corresponds to a state of level 0, presenting an input transition from the node \( \beta \) with value \( v' \), and an output transition towards the node \( \tau \) with value \( v \);
- for each state of level \( i = 0, \ldots, k - 2 \) corresponding to a node \( p \in P_M \), there is a set of states of level \( i + 1 \) linked in input and in output to the state of level \( i \) by transitions with value \( v' \). Each state of level \( i + 1 \) corresponds to a node \( p' \in \text{adj}(p) \);
- each state of level \( i = 0, \ldots, k - 2 \) corresponding to a node \( p \in P_M \), is linked to a state \( \epsilon_{i+1} \) by a transition with value \( v \). The state \( \epsilon_{i+1} \), in its turn, is linked to itself and to the node \( p \) by transitions with value \( v \);
- states \( \beta \) and \( \tau \) emit any symbol with emission value equal to 1;
- each state of level \( i \) corresponding to a node \( p \in P_M \) emits symbols of the type \((q, i) (q \in P_S) \) whose emission value is equal to 1 if \( s_{pq} \geq s_{ih} \), while it is equal to 0 otherwise;
- each state \( \epsilon_i \) emits symbols of the type \((q, j) (q \in P_S, j \geq i) \) with emission value 1, and all the other symbols with emission value 0.

Let \( \pi \) be a path of the alignment model and \( w(\pi) \) be its weight, that is, the sum of the values of the transitions in \( \pi \). Intuitively, we point out that paths scoring high weights will correspond to good pairings between Master and Slave nodes, as will be more clear...
Let $N = \langle P, I \rangle$ be a PPI network, $p \in P$ and $k$ be an integer, $k \geq 1$. A $k$-tour for $p$, is defined as $\text{tour}_k(p) = \langle T_k(p, 0) \rangle$ where, for a generic node $a$:

$T_k(a, k - 1) = (a, k - 1),$  
$T_k(a, i) = (a, i) \circ (\circ_{\text{adj-fil}} T_k(b, i + 1) \circ (a, i)), \forall i < k - 1.$

**Example 1.** Consider the graph illustrated in Figure 1.

![A sample graph](image)

**Fig. 1.** A sample graph

For the node $p_5$, we have the following $k$-tours (for $k = 1, 2, 3$):

$\text{tour}_1(p_5) = \{(p_5, 0)\}$  
$\text{tour}_2(p_5) = \{(p_5, 0), (p_4, 1), (p_5, 0), (p_7, 1), (p_5, 0), (p_1, 1), (p_5, 0)\}$  
$\text{tour}_3(p_5) = \{(p_5, 0), (p_4, 1), (p_3, 2), (p_4, 1), (p_5, 2), (p_4, 1), (p_5, 0), (p_7, 1), (p_3, 2), (p_7, 1), (p_5, 2), (p_7, 1), (p_5, 0), (p_1, 1), (p_5, 2), (p_1, 1), (p_5, 2), (p_2, 2), (p_1, 1), (p_5, 0)\}$

The following definition extends previous Definition 2 to leave out a specific group of nodes from the adjacent sets under consideration.

**Definition 3.** (partial $k$-tour) Let $N = \langle P, I \rangle$ be a PPI network, $p \in P$, $k$ be an integer, $k \geq 1$ and $Q$ be a subset of $P$. A partial $k$-tour for $p$ is defined as $\text{ctour}_k(p, Q) = \langle T_k(p, 0, Q) \rangle$ where, for a generic node $a$:

$T_k(a, k - 1, Q) = (a, k - 1),$  
$T_k(a, i, Q) = (a, i) \circ (\circ_{\text{adj-fil}} T_k(b, i + 1, Q) \circ (a, i)), \forall i < k - 1.$

Both a $k$-tour and a partial $k$-tour can be referred to a specific set of nodes $Q' \subseteq P$. In such a case, they are denoted by $\text{tour}_k(Q') = \{\circ_{\text{ord}}(Q') \circ \text{tour}_k(p)\}$ and $\text{ctour}_k(Q', Q) = \{\circ_{\text{ord}}(Q') \circ \text{ctour}_k(p, Q)\}$, respectively, where $\text{ord}(Q')$ is any given permutation of the elements of $Q'$.

$^4$ Depending on the chosen permutation, different tours are generated, but this choice is immaterial for our purposes.
3 Methods

In the following, we assume that, for each pair of proteins belonging to distinct networks, a basic similarity value (e.g., protein sequence similarity [1]) is known and stored in a suitable dictionary.

Let $N_M = (P_M, I_M)$ and $N_S = (P_S, I_S)$ be the two input PPI networks, where $N_M$ is the Master and $N_S$ is the Slave. Let $D$ be a dictionary of basic similarities, that is, a set of triplets $(p_x, p_y, s_b)$ such that $p_x \in P_M$, $p_y \in P_S$ and $s_b$ is the basic similarity between $p_x$ and $p_y$. Finally, let $k$ be an integer such that $k \geq 1$. The procedure Connected-subgraphs Extraction includes two main steps:

1. find the pair of nodes $(p_0, q_0)$, such that $p_0 \in P_M$ and $q_0 \in P_S$, to be set as best-pair, that is, the seed pair of nodes making the starting solution $S_0$;
2. expand $S_0$ to obtain the solution $S_f$ corresponding to a pair of similar connected subgraphs $C_L$ and $C_F$ of the two input networks.

Step 1 and Step 2 are performed by two algorithms, called Best-pair Finder and Expander, that are described in detail in the following sections.

3.1 Best-pair Finder

Given the two networks $N_M$ and $N_S$, the integer $k$ and the dictionary $D$ of basic similarity in input, Best-pair Finder returns in output the best-pair $(p_0, q_0)$ as follows.

An alignment model $M$ of order $k$ for $N_M$ w.r.t. $N_S$ is generated, and a $k$-tour $T_F$ for the set of nodes in $N_S$ is considered as the output sequence of $M$. Here, high weights of the paths on $M$ correspond to good pairings between Master and Slave nodes. In fact, the value $w(\pi)$ of a path $\pi$ gives a measure of how much the Master node corresponding to the state of level 0 in the path “matches” with the emitted symbol, that corresponds to a Slave node. The notion of “good matching” we adopt is referred to the basic similarity associated with both $p_0$ and $q_0$, and their correspondent adjacent nodes.

To obtain the best match between a node $p_0$ of the Master and a node $q_0$ of the Slave, the path $\pi$ scoring the maximum weight has to be chosen, and the Viterbi algorithm [5, 9] is exploited to this aim.

3.2 Expander

Once that the best-pair of proteins composing the starting solution $S_0 = \{(p_0, q_0)\}$ is computed by Best-pair Finder, $S_0$ has to be expanded until no more proteins belonging to connected sub-graphs we are generating can be paired.

The Expander takes in input two networks $N_M$ and $N_S$, an integer $k$, the current solution $S_0$ and the basic similarity dictionary $D$, and returns in output the solution $S_f$ corresponding to matching two connected subgraphs in the input networks.

To expand $S_0$, the Expander algorithm first analyzes the adjacent sets $adj(p_0)$ and $adj(q_0)$ to find a suitable pair $(p_1, q_1)$, such that $p_1 \in adj(p_0)$ and $q_1 \in adj(q_0)$, to be added to $S_0$. This process leads to the generation of a new partial solution $S_1 =$
The algorithm works analogously to expand $S_1$ until the final solution $S_f$ is generated.

At the generic step $i$, the pair $(p_i,q_i)$ is computed according to the following procedure. Let $S_{i-1} = \{(p_0,q_0),(p_1,q_1),(p_2,q_2),\ldots,(p_{i-1},q_{i-1})\}$ be the solution at the step $i-1$. A partial $k$-tour $T_{p_i}$ for the set of nodes in $N_S$ on the set $Q = \{q_0,q_1,\ldots,q_{i-1}\}$ is generated, as well as a special alignment model $M_p$ for $N_M$. This model is obtained accordingly to the following variant of Definition 1:

- Nodes in the set $P = \{p_0,p_1,p_2,\ldots,p_{i-1}\}$ can not be associated to states of level greater than 0, and only nodes in $P$ are states of level 0. Furthermore, states of level 0 are not linked to any state $\epsilon$, nor to the state $\tau$;
- each state of level 0 emits symbols of the type $(p,q)$, such that $p \in P$ and $q \in Q$ with value 1, and any other symbol with value 0;
- there is a transition with value $v$ from each node of level 1 to the node $\tau$;
- there are no transitions from nodes of level 1 to nodes of level 0;

We call partial alignment model the alignment model $M_p$ generated as described above. Differently from the alignment model of Definition 1, $M_p$ allows to select pairs of proteins belonging to the adjacent sets of already chosen proteins, obtaining the correspondence between connected subgraphs as a final solution.

The partial tour $T_{p_i}$ is used as the output sequence of $M_p$, and the Viterbi Algorithm is applied again to find the path $\pi$ scoring maximum weight. Then, the pair $(p_i,q_i)$ corresponding to $\pi$ is added to $S_{i-1}$, generating this way the new solution $S_i$.

Note that, in the partial alignment model, only nodes of level 1 concur to generate the solution, while nodes of level 0 guarantee that, if the subgraphs generated at the previous step are connected, the new ones will be connected as well, and sharing the same spanning tree.

### 3.3 Global alignment

To perform a global alignment between two networks $N_M = \langle P_M,I_M \rangle$ and $N_S = \langle P_S,I_S \rangle$, the procedure Connected-subgraphs Extraction, illustrated in Section 3, is called iteratively on the two input networks, at each iteration discarding from the analysis protein nodes belonging to the current solution. The process stops when no further correspondence between pairs of subgraphs is returned. Discarding nodes means eliminating them and all the associated edges from the input networks. This way, a one-to-one correspondence between pairs of nodes in the two networks is constructed.

In more detail, the two networks $N_M$ and $N_S$ and an integer $k$ are provided in input, and the output solution $S$ is set equal to the empty-set at the beginning. Then, the procedure Connected-subgraphs Extraction is called on $N_M$, $N_S$ and $k$, and the solution $S_i$ it returns is added to $S$. At this point, nodes included in $S_i$ and all the associated edges are eliminated from the two networks, and Connected-subgraphs Extraction is called again until it does not return any further solution. The final $S$ returned in output will consist in a set of correspondences between pairs of (non-overlapping) connected subgraphs of $N_M$ and $N_S$. 
4 Experimental Results

We tested our technique on the two PPI networks of *Saccharomyces cerevisiae* (yeast) and *Drosophila melanogaster* (fly). We exploited interaction data collected from BI-OGRID [2] and DIP [14]. In particular, the resulting yeast network has 5,443 nodes and 31,898 interactions, while the fly network has 7,404 nodes and 25,830 interactions. The size of the two interaction datasets highlights that the yeast is better characterized than the fly, since a smaller number of fly interactions has been discovered although *D. melanogaster* has a larger number of proteins than *S. cerevisiae*. We performed two different series of experiments, in both cases comparing our results with those returned by one of the most successful tools for global alignment, that is, *IsoRank* [12].

In the first series of tests, we set the yeast network as the Master and the fly network as the Slave. Then, analyzed things the other way around. When the yeast PPI network has been set as the Master, our system returned a global alignment involving 945 protein pairings, while when the fly has been exploited as the Master and the yeast as the Slave, the system returned a global alignment involving 707 pairings. In the first case, the proteins aligned by our system include enzymes involved in carbohydrate metabolism, mitochondrial enzymes involved in various metabolic pathways, glycosyl trasferase and other enzymes, as well as chaperonin proteins and proteins involved in endocytosis. In the second case, our system correctly pairs most conserved proteins that include, again, mainly metabolic enzymes, aminoacil-tRNA- synthetase that are crucial enzymes for protein synthesis, and RNA polymerase subunits.

On the same PPI networks, *IsoRank* returned a global alignment involving 5,499 protein pairings. Although when, as in the discussed case, the global alignment returned by our system involves a smaller set of pairings than *IsoRank*, our system returned pairings that *IsoRank* did not (764 for yeast and 589 for fly). This is due to the threshold value that we forced on the sequence similarity, relaxing which the number of returned protein pairs becomes larger. On the other hand, all those pairings returned by *IsoRank* but not by our system have sequence similarity lower then the threshold value. These results point out that aligning the two networks from a different point of view, where the approximation plays different roles on the two sides and only what of the Master is conserved in the Slave is searched for, leads to different and still biologically meaningful results.

Furthermore, when the focus is turned on the fly network, and most of its structural information is kept, the resulting alignment is smaller than in the previous case, possibly because the yeast is better characterized than the fly, thus presenting a larger number of interactions. When the yeast is the Slave, most of its structural information gets lost, and thus some of the associations found in the previous case are no longer recognized.

A second key to explain the results is the following. When a PPI network is exploited as the Master, this makes the search process to follow a precise direction, that is, searching for *those regions of the Master which have been conserved in the Slave*. Our analysis showed that, according to the available interaction data, there are more yeast regions that have been conserved in the fly than vice versa, which is reasonable observing that the fly is a more complex organism than the yeast.
5 Conclusion

The method we proposed is general enough to be applied to other types of networks. Furthermore, the approach can be extended to handle multiple network alignment by iteratively aligning pairs of networks and taking, at any iteration, the set of already aligned networks, encoded as a suitable finite state automaton, as the Master. We argue that, when more reliable and accurate interaction data will be available, our approach can effectively support the discovery and prediction of unknown protein functions for the less characterized organisms, providing a new direction of investigation that is orthogonal to those of the other techniques.

References