A heuristic for biological network alignment.

Alex Radu
Michael Charleston
School of Information Technologies
FACULTY OF ENGINEERING & INFORMATION TECHNOLOGIES

MOTIVATION

Modern sequencing techniques have generated a large influx of biological data. Current analysis techniques are becoming infeasible due the scope of the data being generated. Reducing raw data into networks has shown itself to be a viable technique for data mining in other fields and is applied to the field of systems biology.

While techniques to reduce the vast raw data into accurate networks are not yet consistent; they are improving quickly and the need for methods to analyze these inferred networks has brought about many new algorithms for discovering motifs - commonly occurring patterns - within networks.

Network querying has been a successful method for discovering already known motifs, but it lacks the capability of discovering new motifs within one or more organism.

Network alignment[1,4,6,7,8] can be used to discover similarities and differences between the networks of two or more organisms. Existing methods require complex pre-processing or careful calibration of key parameters to produce an accurate alignment.

AIM

- To propose a simple, flexible and efficient method for pairwise and multiple network alignment that does not require complex pre-processing or parameter calibration.

SIGNIFICANCE

The literature shows that current network analysis methods have been used to identify gene expression signatures of cancer[5], hypothesized to identify and validate drug targets[2] and identify condition-specific topological changes in organisms[3].

The biological networks that are being focused on within this study are gene regulatory networks. These networks model the regulatory interaction governing the expression of genes.

CHALLENGES

- Mutations pose a large challenge in any network alignment algorithm. Organisms may have many small mutations within the course of their existence. These pose a great challenge as small mutations may cause large changes in the network. Any method devised must be able to flexibly align networks.
- Large datasets are also a concern as the algorithms devised for this problem must be able to execute within a reasonable time frame.

METHOD

Core Heuristic

The core heuristic compares the nodes in a graph based on the degree distribution of the parent and child nodes. This process is used to "fingerprint" a node. The more similar the degree distribution of the parents and children of two nodes, the more likely they are to correspond in two networks. My heuristic calculates a score of the mapping of two nodes and is calculated as follows:

$$s(u,v) = \left(\beta|I_v - I_u| \times \prod_{a\in p(v),b\in p(u)} \delta_{a,b}\right) + \left(\beta|O_v - O_u| \times \prod_{a\in c(v),b\in c(u)} \delta_{a,b}\right)$$

where

$$\delta_{a,b} = \begin{cases} \alpha \text{ if } a \cong b \\ \dfrac{\min(I_{a,}, I_{b,})}{\max(I_{a,}, I_{b,})} + \dfrac{\min(O_{a,}, O_{b,})}{\max(O_{a,}, O_{b,})} \text{ otherwise} \end{cases}$$

and α and β are tunable parameters(in practice 4, 0.8 respectively). They represent the bonus score for an already mapped pairing and the penalty score for a different number of vertices in the set. *p(v)* and *p(u)* are functions to get the parents of nodes *v* and *u.* Similarly *c(v)* and *c(u)* are functions to get the children of nodes *v* and *u*.

Network Alignment System

The remainder of the system uses progressive alignment to increase the accuracy and performance. Progressive alignment is performed by using the adjacent nodes of already mapped nodes as candidates for alignment.

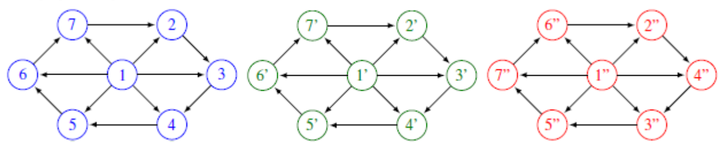
If there are no nodes adjacent to already mapped nodes that have not yet been mapped, the algorithm chooses the top 10 remaining nodes decreasing in order of degree. These nodes are chosen as the score increases with the degree of the node being calculated and such, nodes with high degrees will generally have higher scores.

Nodes are then mapped if they are above the average score.

It is possible to create a pathological example.

Pathological example

Networks with high similarity may create a poor mapping.



First Pass

	1	2	3	4	5	6	7
1'	64	0.41943	0.41943	0.41943	0.41943	0.41943	0.41943
2'	0.41943	8	8	8	8	8	8
3'	0.41943	8	8	8	8	8	8
4'	0.41943	8	8	8	8	8	8
5'	0.41943	8	8	8	8	8	8
6'	0.41943	8	8	8	8	8	8
7'	0.41943	8	8	8	8	8	8

Second Pass

	2	3	4	5	6	7
2'	16	16	16	16	16	16
3'	16	16	16	16	16	16
4'	16	16	16	16	16	16
5'	16	16	16	16	16	16
6'	16	16	16	16	16	16
7'	16	16	16	16	16	16

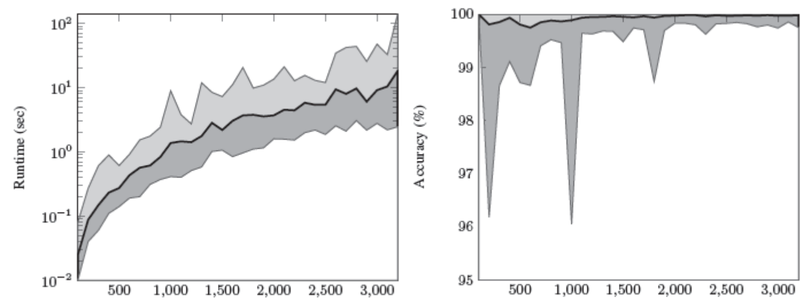
In this example, the first pass maps 1 to 1' since the score is much larger than other scores. During the second pass, all of the scores are identical due to the structure of the network and therefore may result in a poor mapping.

Real networks will not have this degree of symmetry and in practice is not a problem.

PERFORMANCE

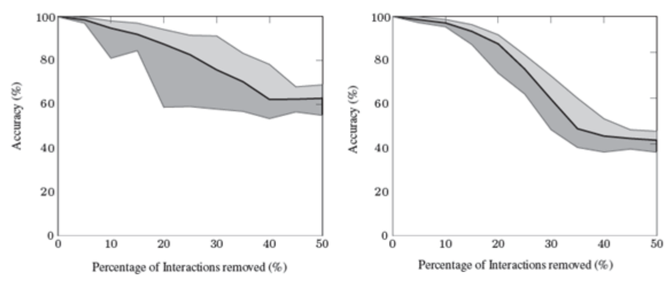
Identical Graphs

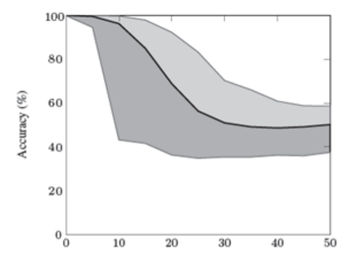
- Runtime & Accuracy:



Modified Graphs

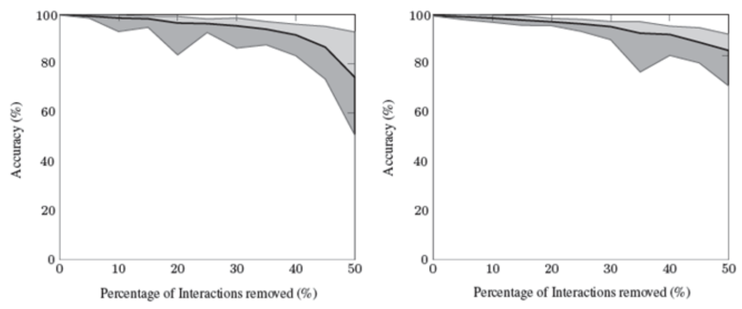
- Interaction deletion

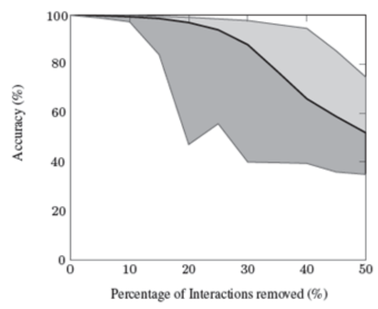




(c) 3200 node simulated GRN

- Node Deletion





(c) 3200 node simulated GRN

Conclusion

I have described a novel approach to the network alignment problem. This approach aligns networks with no parameter calibration. It is able to complete alignments of realistically sized networks quickly and accurately. We believe it is the first practical method designed to work in bioinformatics.

References

1. Jiefeng Cheng *et al*. IEEE Transactions on Knowledge and Data Engineering, 23(10), 2010.
2. S. Imoto *et al.* J Bioinform Comput Biol, 1(3):459–474, 2003.
3. I King Jordan *et al.* Molecular biology and evolution, 21(11):2058–70, November 2004.
4. Maxim Kalaev *et al.* Bioinformatics (Oxford, England), 24(4): 594–6, February 2008.
5. Nikolai Slavov *et al.* Proceedings of the National Academy of Sciences, 106(11):4079, 2009.
6. Roded Sharan *et al.* Nature biotechnology, 24(4):427–33, April 2006
7. Roded Sharan *et al.* Proceedings of the National Academy of Sciences of the United States of America, 102(6): 1974–9, February 2005.
8. S. Zampelli *et al.* Principles and Practice of Constraint Programming-CP 2005.