Connecting Seed Lists of Mammalian Proteins Using Steiner Trees

Amelia G. White
Department of Computational Biology and Molecular Biophysics
Rutgers, The State University of New Jersey
Hill Center, 110 Frelinghuysen Road
Piscataway, NJ 08854
white.amelia@gmail.com

Avi Ma'ayan
Department of Pharmacology and Systems Therapeutics
Mount Sinai School of Medicine
1425 Madison Avenue
New York, NY 10029
avi.maayan@mssm.edu

***Abstract*-Multivariate experiments and genomics studies applied to mammalian cells often produce lists of genes or proteins altered under treatment/disease vs. control/normal conditions. Such lists can be identified in known protein-protein interaction networks to produce subnetworks that "connect" the genes or proteins from the lists. Such subnetworks are valuable for biologists since they can suggest regulatory mechanisms that are altered under different conditions. Often such subnetworks are overloaded with links and nodes resulting in connectivity diagrams that are illegible due to edge overlap. In this study, we attempt to address this problem by implementing an approximation to the Steiner Tree problem to connect seed lists of mammalian proteins/genes using literature-based protein-protein interaction networks. To avoid over-representation of hubs in the resultant Steiner Trees we assign a cost to Steiner Vertices based on their connectivity degree. We applied the algorithm to lists of genes commonly mutated in colorectal cancer to demonstrate the usefulness of this approach.**

I. INTRODUCTION

Advanced experimental technologies that measure many cellular components at once often produce lists of genes or proteins that were identified as statistically altered under treatment vs. control conditions. The size of such lists ranges between handfuls to thousands. To assist in interpreting such experimental result it is often desired to place such lists in functional context. For this, Gene Ontology [1] analysis is commonly used. Alternatively, protein-protein interaction networks can also be used to "connect" seed lists of genes or proteins based on known protein interactions. Several graph-theory algorithms can be used to "connect" seed lists of proteins using known intracellular interactions networks. Formally, the input to such algorithms is an undirected graph $G = (V, E)$ and a set of vertices (the seed list) $N \subset V$. The task is to find a subgraph G' that links the vertices from N. Applications of this concept to analyze biological networks is an active area of research: For example, this approach was used for expanding metabolic networks [2]; enriching classical pathways [3]; linking disease genes with disease phenotype using information from protein interaction networks, microarrays, and *in-silico* predicted interactions [4]; Asthana et al. [5] used interactions from multiple sources to fill-in additional proteins in a complex, using seed lists of proteins already known to be in a complex; Ideker et al. [6] used protein interaction networks to identify clusters of differentially expressed genes; Scott et al. [7] used a similar approach to detect signaling pathways in yeast. Connecting seed lists is also used to predict protein function [8].

We implemented Genes2Networks [9], a web-based tool that uses ten mammalian protein interaction databases to connect seed lists of gene symbols. The algorithm implemented for Genes2Networks finds all paths within a certain path-length threshold between pairs of nodes from seed lists. The output subgraphs are visualized using AVIS [10], a web-based pathway viewer. Efforts to improve visualization of biochemical pathways is also an active field of research [11]. For biologists, having an output subgraph created from seed lists visualized as connectivity diagram is informative. When seed lists are relatively large (>30) such maps are overloaded with information because visualization algorithms need to project high-dimensional networks into two or three dimensions having to "draw" overlapping nodes and tangled links. Hence, it is often desired to find minimal subgraphs that could be used to "connect" seed lists in the context of large-scale interaction networks. Such minimal subgraphs are useful because they are visually manageable, and can present the intermediate vertices that are most relevant to the seed list.

Given a seed list of vertices, Steiner Trees (STs) are used to find such minimal "skeleton" subgraphs [12, 13]. STs are similar to minimal spanning trees [14], except that STs include intermediate vertices and consider weights of edges [12, 13] or vertices [15]. Intermediate vertices, called Steiner Vertices (SVs), are vertices that are not

present in the seed lists but exist in the background network and appear in the output ST. Finding the ST is NP-hard [13, 16] and a dynamic programming solution was first suggested by Dreyfus and Wagner [13]. Many approaches since then have improve the performance of the Dreyfus-Wagner algorithm [17, 18]. Approximations are required for practical applications, for example, when the seed list is ~>10 and the background network is large [19]. Applications of ST in biology so far include reconstructing phylogenetic trees [20, 21] and predicting protein folds [22]. ST were also implemented to analyze seed lists of genes and proteins in yeast using a large protein interaction network [15, 23]. The ST application to connect seed lists of genes or proteins as it was implemented so far has a major drawback. The SVs that are used to connect seed lists are often highly connected vertices (hubs). Hence, the inclusion of hubs in the output ST has a strong bias because regardless of the seed list hubs would reappear in many output STs. Here, we developed an approach to overcome this drawback by assigning a cost to edges based on the connectivity degree of the vertex at the head of the edge. We implemented an algorithm that quickly approximates a ST using measures such as the shortest path length [24]. We use Genes2Networks to create a subgraph from a large mammalian protein-protein interaction network developed from multiple sources, and then find the ST in this subgraph. To demonstrate its usefulness, we applied this algorithm to seed lists of genes identified as mutated in colorectal cancer [25].

II. METHODS

A. The Steiner Tree problem

The ST problem [26] in graphs is described as: Given an undirected graph $G = (V, E, c)$, where $c : E \to R$ is an edge length function, and a non-empty set of seed vertices N, $N \subseteq V$, called terminals, find a subgraph $T_G(N)$ of G such that there is a path between every pair of terminals, and the total length is minimized:

$$| T_G(N) | = \sum_{e_i \in T_G(N)} c(e_i) \qquad (1)$$

The vertices $V \setminus N$ are called non-terminals; non-terminals that end up in $T_G(N)$ are called Steiner Vertices (SVs).

B. Distance Network Heuristic

Since the ST problem is NP-hard, many approximation algorithms are available. We chose to use an approximation algorithm called the Distance Network Heuristic (DNH) [27]. This approximation uses the Distance Network, which is defined as a fully connected subgraph $D_G = (N, E_D, c_D)$, where $c_D : E_D \to R$ is an edge length function, where $c_D(e_D ij)$ is the length of the shortest path between the terminals n_i and n_j.

Algorithm steps:

Step 1: Construct the distance network $D_G(N)$ for N

Step 2: Determine a minimum spanning tree of $D_G(N)$

Step 3: Replace each edge in the minimum spanning tree by the corresponding shortest path. Let T_D denote this network.

Step 4: Determine a minimum spanning tree T_{DNH} of the subgraph of G induced by the vertices of T_D.

Step 5: Delete from T_{DNH} non-terminals vertices with degree k=1

The DNH has complexity of $O(nv^2)$, finding the Distance Network contributes mostly to this complexity, using a Fibonacci heap can improve the complexity to $O(n(e + v\log v))$ in sparse networks [28]. In this work we chose to use a Binary heap to compute the distance network, this gives a complexity of $O(n(v\log v + e\log v))$. We chose to implement a binary heap because it is simpler to implement and known to outperform better than the Fibonacci heap for moderate sized graphs [26].

The worst case error ratio for the DNH is:

$$\frac{| T_{DNH} |}{| T_G(N) |} \leq 2 - \frac{2}{n} \qquad (2)$$

for any network G and any set of N terminals [26]. Although in practice the DNH usually performs much better than the worst case error ratio, it is easy to find examples of STs that will be missed by DNH. Even though the DNH may not always produce the true ST, the other approximation algorithms that exist do not perform significantly better and have similar error ratios [26]. From now on we will refer to the approximate ST from the DNH as the ST solution.

C. Assigning Cost to Edges

To find the ST we replaced each undirected edge with two directed edges. In the first version $G_1 = (V_1, E_1, c_1)$, the weight of each edge is set to be 1, $c_1(e_{1ij}) = 1$ for all i, j. In the second version $G_2 = (V_2, E_2, c_2)$, the length of each edge was set as the degree of the vertex at the head of the edge $c_2(e_{2ij}) = \text{degree}(v_{2j})$.

The ST found in graph G_1 is likely to include major hubs as SVs. Vertices that are hubs have high degree and

therefore appear in more possible paths, a node appearing in many paths is more likely to be included as an SV. Hub SVs might not be as interesting because they would appear in STs found using many random seed lists. We would like to isolate SVs specific to specific seed lists, the ST found using G_2 will accomplish this because the costs of the edges will favor paths that do not go through hubs. For example, see Fig. 1.

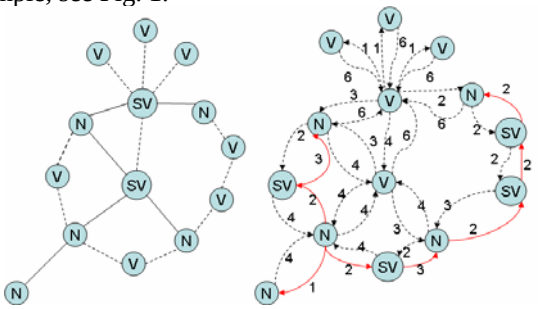

Figure 1. Comparing a graph before and after being converted to G_2. V-Vertices, N-terminals, SV-SV, edge costs are shown, edges included in ST are solid; edges from the graph not included in the ST are dashed. The ST on the right is different from the ST on the left since it does not go through two vertices with highest degree.

D. Mammalian Protein-Protein Interaction Network

We merged the following available protein-protein interaction datasets: The human and mouse interactions from the BioGRID [29], DIP [30], HPRD [31], IntAct [32], MINT [33], Ma'ayan et al. [34], BIND [35], PPID [36] and Reactome [37]. All interactions from these databases report protein-protein and signaling interactions determined experimentally, and include the PubMed reference of the research article that describes the experiments used to identify the interaction. This network was filtered by excluding articles that contributed more than five interactions to reduce the content of interactions determined using high-throughput methods and interactions extracted from review articles.

III. Results

In order to verify that our algorithm reduces the probability for the presence of hubs in resultant STs, or in other words, the ST found using G_2 did not isolate as many hubs as the ST found using G_1, we generated 100 sets of 35 randomly chosen terminals from the mammalian protein-protein interaction network, N_1-N_{100}, $|N| = 35$. Using each set of terminals, N_i, as input to Genes2Networks we generated 100 subgraphs, $G(N_i)$ for $i=1..100$. We created $G_1(N_i)$ and $G_2(N_i)$ from these subgraphs, for each set of terminals N_i, using the methods described previously. We found STs, $T_{DNHG1}(N_i)$ and $T_{DNHG2}(N_i)$, for each i, and compared the results by counting the number of times each vertex appeared as a SV in $T_{DNHG1}(N_i)$, for $i=1..100$, and comparing it to the number of times the vertex appeared as SV in $T_{DNHG2}(N_i)$, for $i=1..100$.

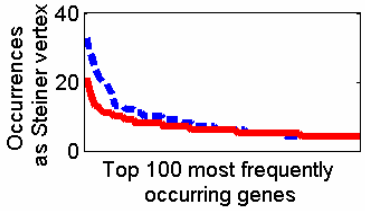

Figure 2. Frequency of SVs found using 100 randomly generated sets of terminals. The Plot shows the 100 most frequently appearing SVs and the number of times they appeared as SVs using G_1 (blue, dashed) and G_2 (red, solid).

Fig. 2 shows that when we used G_2, the frequency of the vertices appearing as SVs is reduced, suggesting that our algorithm reduces the probability for SVs to reappear in many STs regardless of the content of seed lists. Since we are using Genes2Networks [9] with a limited path length threshold, paths that do not go through hubs may not be included in the initial subgraph, and as such, will not be found by the DNH algorithm. Therefore, we expect that if we would have used the protein-protein interaction network directly as the input to the DNH algorithm, the difference between the frequencies of vertices appearing as SVs would be magnified.

To illustrate the usefulness of our approach for the application of analyzing lists of mammalian genes, we used a recent publication by Sjoblom et al. [25]. In their study, the authors identified many mutations in human genes in breast and colorectal cancers. The study reports lists of genes that are heavily mutated in those cancers. The authors speculated that some of those genes may function in the same pathways and the genetic alteration may share related functional outcomes. To further examine this we used a list of 69 genes that were identified as being highly mutated in colorectal cancer. Fig. 3a shows the output subgraph created with Genes2Networks [9]. This subgraph contains many tangled edges. In contrast, in Fig. 3b, we use the ST algorithm without assigning weights. The connectivity diagram is more informative. Fig. 3c improves the specificity of the output ST by reducing the influence of hubs, using costs on the edges, G_2.

Encouragingly, some of the SVs appearing to connect the mutated genes in the STs are classic colorectal cancer genes, often used as biomarkers to determine disease progression. Some SVs function within well-studied signaling pathways. For example, AXIN2 [38] and β-catenin [38] were identified in proximity to APC and TCF7L2 all known to participate in the same pathway. It is notable that although β-catenin appeared in both STs, AXIN2 only appeared when we used G_2. The output STs

can also visually suggest clustering and hierarchical organization of mutated genes. TP53 is a root vertex in both STs, and it becomes more central in the G_2 map. This happens because TP53 is a hub. The output trees from G_1 have less distinct roots but are also very informative. For example, one of the branches from MAP2 connects Ephrins to adapter proteins such as GRB2, CRK and CRKL all are known to function in the MAPK pathway. Another example is the terminal braches from TP53 which are nuclear proteins connected through regulators of transcription such the histone deacetylase HDAC1 and nuclear kinase CDK2.

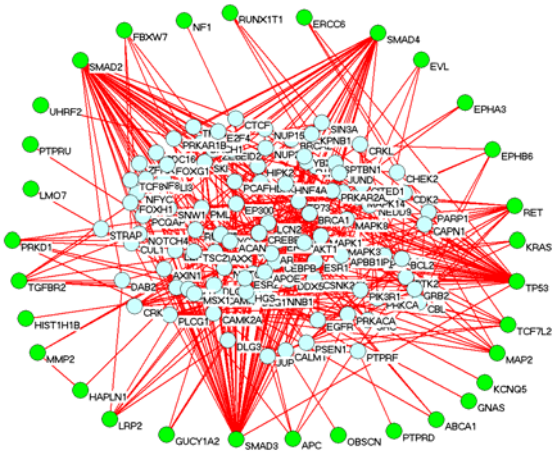

Figure 3a. Subnetwork output for the colorectal cancer seed list using Genes2Networks.

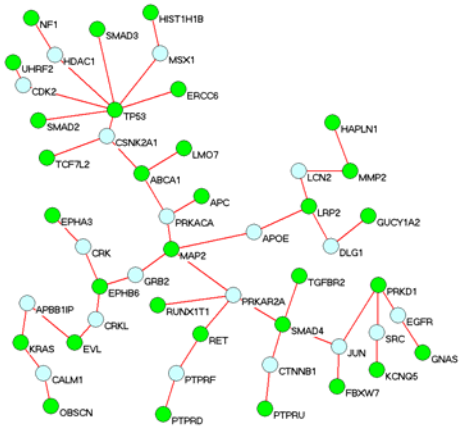

Figure 3b. ST output for the colorectal cancer seed list without assigning cost to edges (G_1), seed nodes are green Steiner vertices are light cyan.

Figure 3c. ST output for the colorectal cancer seed list assigning cost to the edges (G_2), seed nodes are geen, Steiner vertices are light cyan.

These functional modules, identified automatically, are already known to become mutated in colorectal cancer and are known to play important functional roles in the different stages of cancer progression. Other less studied "connectors" can provide novel hypotheses for directed experimental exploration.

IV. Conclusions

The ST approach to connect seed lists of genes can be used to extract knowledge from complex biological networks by visually producing sizable connectivity diagrams that can suggest SVs as additional potential players, cluster genes or proteins based on their relations in network connectivity space, and trim dense subgraphs to keep the most important vertices and edges. ST algorithms can produce many correct or correct approximations for the solution output ST. additionally, finding STs requires that there exists a path between every gene in the seed lists. Hence, it might be more appropriate to output the union of all possible STs (Steiner Forest). The methods presented here can be used for the analysis of a variety of experimental results, for example, microarrays, proteomics, or protein/DNA arrays. Data from such studies can be incorporated into the algorithm to assign weights to terminals and SVs accordingly based on quantifications observed experimentally. The output STs can also incorporate gene classification such as Gene Ontology, linkage to known pathways, and clustering analysis; all would improve interpretations of the ST output.

Acknowledgements

This research was supported by NIH Grant No. 1P50GM071558-01A27398 and start-up fund from Mount Sinai School of Medicine to AM. We thank Eduardo

Sontag and Paola Vera-Licona from Rutgers University for useful discussions.

REFERENCES

[1] M. Ashburner, et al. "Gene Ontology: tool for the unification of biology," *Nat Genet*, vol. 25, pp. 25-29, 2000.

[2] T. Handorf and O. Ebenhoh, "MetaPath Online: a web server implementation of the network expansion algorithm," *Nucl. Acids Res.*, p. gkm287, May 5, 2007 2007.

[3] L. J. Lu, et al. "Comparing classical pathways and modern networks: towards the development of an edge ontology," *Trends in Biochemical Sciences*, vol. In Press, Corrected Proof.

[4] K. Lage, et al. "A human phenome-interactome network of protein complexes implicated in genetic disorders," *Nat Biotech*, vol. 25, pp. 309-316, 2007.

[5] S. Asthana, O. D. King, F. D. Gibbons, and F. P. Roth, "Predicting Protein Complex Membership Using Probabilistic Network Reliability," *Genome Res.*, vol. 14, pp. 1170-1175, June 1, 2004 2004.

[6] T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel, "Discovering regulatory and signalling circuits in molecular interaction networks," *Bioinformatics*, vol. 18, pp. S233 – 240, 2002.

[7] J. Scott, T. Ideker, R. M. Karp, and R. Sharan, "Efficient Algorithms for Detecting Signaling Pathways in Protein Interaction Networks," *Journal of Computational Biology*, vol. 13, pp. 133-144, 2006.

[8] R. Sharan, I. Ulitsky, and R. Shamir, "Network-based prediction of protein function," *Molecular Systems Biology*, vol. 3, p. 88, 2007.

[9] S. I. Berger, J. M. Posner, and A. Ma'ayan, "Genes2Networks: connecting lists of gene symbols using mammalian protein interactions databases," *BMC Bioinformatics* vol. 8, p. 372, 2007.

[10] S. I. Berger, R. Iyengar, and A. Ma'ayan, "AVIS: AJAX Viewer of Interactive Signaling Networks," *Bioinformatics*, p. btm444, September 12, 2007 2007.

[11] M. Suderman and M. Hallett, "Tools for Visually Exploring Biological Networks," *Bioinformatics*, p. btm401, August 25, 2007 2007.

[12] S. L. Hakimi, "Steiner's problem in graphs and its implications," *Networks*, vol. 1, 1971.

[13] S. E. Dreyfus and R. A. Wagner "The Steiner problem in graphs," *Networks*, vol. 1, p. 111, 1972.

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Minimum Spanning Trees," in *Introduction to Algorithms*, Second Edition ed: MIT Press and McGraw-Hill, 2001, pp. 561-579.

[15] M. S. Scott, et al. "Identifying Regulatory Subnetworks for a Set of Genes," *Mol Cell Proteomics*, vol. 4, pp. 683-692, May 1, 2005 2005.

[16] M. R. Garey, Johnson, D. S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W.H. Freeman, 1979.

[17] E. Althaus, T. Polzin, and S. V. Daneshmand, "Improving Linear Programming Approaches for the Steiner Tree Problem," in *Experimental and Efficient Algorithms: Second International Workshop, WEA 2003, Ascona, Switzerland, May 26-28, 2003. Proceedings*, 2003, pp. 620-621.

[18] B. Fuchs, W. Kern, and X. Wang, "Speeding up the Dreyfus–Wagner algorithm for minimum Steiner trees," *Mathematical Methods of Operations Research*, vol. 66, pp. 117-125, 2007.

[19] R. M. Karp, *Reducibility Among Combinatorial Problems*. New York: Plenum Press, 1972.

[20] Z. Adam, M. Turmel, C. Lemieux, and D. Sankoff, "Common Intervals and Symmetric Difference in a Model-Free Phylogenomics, with an Application to Streptophyte Evolution," *Journal of Computational Biology*, vol. 14, pp. 436-445, 2007.

[21] L. R. Foulds, M. D. Hendy, and D. Penny, "A graph theoretic approach to the development of minimal phylogenetic trees.," *J Mol Evol.* , vol. 13, pp. 127-149, 1979

[22] J. M. Smith, Y. Jang, and M. K. Kim, "Steiner minimal trees, twist angles, and the protein folding problem," *Proteins: Structure, Function, and Bioinformatics*, vol. 66, pp. 889-902, 2007.

[23] N. Betzler, "Thesis: Steiner Tree Problems in the Analysis of Biological Networks," University of Tubingen, Tübingen, Baden-Württemberg, Germany 2006.

[24] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[25] T. Sjoblom, et al. "The Consensus Coding Sequences of Human Breast and Colorectal Cancers," *Science*, vol. 314, pp. 268-274, October 13, 2006 2006.

[26] F. K. Hwang, D. S. Richards, and P.Winter, *The Steiner Tree Problem* vol. 53. Amsterdam: Elsevier Science Publishers B.V, 1992.

[27] K. Melhorn, "A faster approximation algorithm for the Steiner problem in graphs," *Inf. Processs. Lett.*, vol. 27, pp. 125-128, 1988.

[28] M. L. F. a. R.E.Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms for the Steiner problem in graphs," *J. Assoc.Comput. Mach.*, vol. 34, pp. 596-615, 1987.

[29] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, "BioGRID: a general repository for interaction datasets," *Nucl. Acids Res.*, vol. 34, pp. D535-539, January 1, 2006 2006.

[30] I. Xenarios, E. Fernandez, L. Salwinski, X. J. Duan, M. J. Thompson, E. M. Marcotte, and D. Eisenberg, "DIP: The Database of Interacting Proteins: 2001 update," *Nucl. Acids Res.*, vol. 29, pp. 239-241, January 1, 2001 2001.

[31] G. R. Mishra, et al. "Human protein reference database--2006 update," *Nucl. Acids Res.*, vol. 34, pp. D411-414, January 1, 2006 2006.

[32] S. Kerrien, et al. "IntAct--open source resource for molecular interaction data," *Nucl. Acids Res.*, vol. 35, pp. D561-565, January 12, 2007 2007.

[33] A. Chatr-aryamontri, et al., "MINT: the Molecular INTeraction database," *Nucl. Acids Res.*, vol. 35, pp. D572-574, January 12, 2007 2007.

[34] A. Ma'ayan, et al. "Formation of regulatory patterns during signal propagation in a Mammalian cellular network," *Science*, vol. 309, pp. 1078-83, 2005.

[35] G. D. Bader, D. Betel, and C. W. V. Hogue, "BIND: the Biomolecular Interaction Network Database," *Nucl. Acids Res.*, vol. 31, pp. 248-250, January 1, 2003 2003.

[36] H. Husi, M. A. Ward, J. S. Choudhary, W. P. Blackstock, and S. G. N. Grant, "Proteomic analysis of NMDA receptor-adhesion protein signaling complexes," *Nat Neurosci*, vol. 3, pp. 661-669, 2000.

[37] G. Joshi-Tope, et al. "Reactome: a knowledgebase of biological pathways," *Nucl. Acids Res.*, vol. 33, pp. D428-432, January 1, 2005 2005.

[38] M. D. Castellone, et al. "Prostaglandin E2 Promotes Colon Cancer Cell Growth Through a Gs-Axin-{beta}-Catenin Signaling Axis," *Science*, vol. 310, pp. 1504-1510, December 2, 2005 2005.