2012

Function-preserving filters for sampling in biological networks

Kathryn Dempsey Cooper
University of Nebraska at Omaha, kdempsey@unomaha.edu

Sanjukta Bhowmick
University of Nebraska at Omaha, sbhowmick@unomaha.edu

Hesham Ali
University of Nebraska at Omaha, hali@unomaha.edu

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 9 (2012) 587 – 595

Procedia Computer Science

International Conference on Computational Science, ICCS 2012

Function-preserving filters for sampling in biological networks

Kathryn Dempsey[a,b], Sanjukta Bhowmick[a], Hesham Ali[a,b]*

[a]College of Information Science & Technology, University of Nebraska Omaha, Omaha, NE, 68198
[b]Department of Pathology & Microbiology, University of Nebraska Medical Center, Omaha, NE, 68182

Email: hali@mail.unomaha.edu

Abstract

Assays created to study systems of disease and aging can offer a whole new set of therapeutic targets. However, with experiments of this immense volume, data becomes unmanageable for many traditional analyses. Enter the biological network, a tool for modeling relationships among high-throughput data that is quickly rising in popularity. Small networks (in the order of hundreds to few thousands of nodes) use relationships between network structure to infer biological function; this relationship has been confirmed and used in many studies to advance the study of model organisms. Networks built for assessing entire genomes, or entire protein repertoires, however, tend to be very large and complex, having tens of thousands of nodes and in some cases upwards of millions of edges. Thus, network sampling techniques take an appropriate step to reduce complexity while modeling data. Here we present a new type of network sampling applied to biological correlations network, the spanning tree, designed to identify critical hub nodes in the model. We compare this filter to others previously used to identify structures in complex networks, chordal-based filters. The results of this work highlight the applicability for multiple filters based upon the graphic structure and biological result desired.

Keywords: spanning tree; chordal graphs; biological networks

1. Introduction

Biological networks are quickly becoming a popular tool for modeling high-throughput data, as they represent relationships among elements that are otherwise ignored or assumed, especially in traditional analyses such as Gene Set Enrichment Analysis [10]. Although many studies have been performed using biological networks, typically large complex networks are avoided due to a lack of computational resources for proper analysis of complex networks. While work to create algorithms that are capable of handling large networks is ongoing [15], typically these networks are filtered to become usable for analysis. Networks can be reduced in size by removal of "non-important" nodes and edges; however, in noisy and incomplete biological data, no edge or node can truly be considered "non-important." Enter the concept of a network sampling, or filter, which is designed to maintain a specific structure of the network. Only recently has the concept of an intelligent filter been applied to work in biological networks, whereas the concept of finding high-priority nodes or substructures has been a constant since the inception of graph theory.

Our previous work on intelligent network filtering [3, 11] has found that within biological networks, identifying the chordal graph of a larger graph G as a sampling technique is able to consistently filter networks while enhancing clusters within the original network. A chordal graph is one that contains no cycles larger than C_3, and is called triangulated because of the repetition of the triangle motif, or the subset of edges $E = \{(node_a, node_b), (node_a,$

doi:10.1016/j.procs.2012.04.063

$node_c$), ($node_b$, $node_c$)}. This motif has been found to exist in co-expression networks as an identifier of true co-regulated expression. In biological networks, and specifically correlation networks, clusters of these tightly regulated genes have been found to correspond with genes working together towards a common cellular function [5]. To identify these clusters accurately we have to identifying all genes involved with a certain function, including some that were likely previously not associated with that function. In this vein, the approach of modeling relationships graphically becomes appealing especially to systems biologists, who seek to gain insights on biological systems from a "bird's eye" point of view.

However, clusters are not the only desirable structure from which to draw biological ties in a network. Another highly researched structure within the network is the high-degree node, informally known as a "hub," which has been found to be critical for network robustness and communication in biological networks [7, 8]. Hubs can represent essential genes, multi-functional proteins, or elements whose function is so critical that without it, the network, and in turn, the organism, perishes. Thus, it becomes advantageous to identify hubs within a network especially when determining the critical genes for a specific time or state of the organism (such as age, disease stage, etc.). It is known that hubs are not *always* representative of essential genes; for example, the complete *Saccharomyces cerevisiae* (yeast) protein-protein interaction network has been constructed by multiple laboratories (in the network, nodes represent proteins and edges represent some physical interaction between two proteins). Essential gene information has been verified for all ~6,000 genes within this network. Using these tools, it has been determined that within the yeast protein-protein interaction network, ~60% of hub genes represent essential genes, compared to ~20% of genes randomly chosen from the network.

However, the *S. cerevisiae* network is currently one of the only networks with complete essential and interaction information and relative to most model organisms, it is very small. Studies on both types of data in other model organisms are ongoing but incomplete, and a major bottleneck to essential gene studies in higher-order organisms is that as the complexity of organisms increases, so does the number of genes. Human and mouse genomes both are estimated to contain around 23,000 genes each, which is an impossibly high number to test *in vivo* even with large labor and monetary resources at one's disposal. As such, essential gene prediction based on approximated network models remains a hot topic of study in systems biology. Networks built from high-throughput data that are rapidly becoming available via open-access data publishing are often large (40,000 nodes or more with billions of edges possible) and so, filtering the network becomes a mainstay of network analysis unless large parallel computing resources and algorithms are at hand. Thus, applying filters that best are able to maintain the desired network structure is an appropriate next step in identifying potential essential genes.

1.1. Hypothesis

Our current research in this area reveals that network filtering based on maintaining specific structures has a definite impact on that filters ability to identify potential essential genes. The prediction of hub nodes is a simpler task than identifying chordal subgraphs, and as such, we propose the following hypothesis, H_O*: A tree-based network filter will identify essential high-degree nodes in some network G while reducing the number of edges comparably to a chordal-based filtered network.* With this hypothesis we propose a spanning tree network filter that will drastically reduce the number of edges in the network, but nodes with high degree will maintain their high degree in the spanning tree. We can assume that by using a spanning tree filter with unweighted edges, that edges chosen for the filter will connect to nodes that are more central in the original network due to the "friendship paradox" described by [16] for modular networks. We use two controls, the original unfiltered network, and multiple chordal graph filters, to verify our findings.

1.2. Complexity

Our proposed hypothesis pits the spanning tree filter against chordal-based filters and proposes that the filter will perform comparably in tests of lethality based on degree. The inherent benefit of identifying high-degree nodes with comparable accuracy in this case is the change in complexity between filters. With a complexity of $O(V+E)$ (where V = number of vertices, E = number of edges), the spanning tree takes much less computational resources to compute compared to chordal subgraphs, with a complexity of $O(E*d)$ (where E = number of edges and d = maximum degree). Further, many previous works have implemented parallel versions of chordal subgraph filtering

[3, 11] to reduce computational burden especially in the case of large, sparse networks. Reproducibility and/or implementation of chordal-based filtering in high-throughput derived biological networks assumes access to distributed computing resources and ability to implement chordal sampling algorithms. This is not always the case especially for biological laboratories using the network model in their own research; as such the spanning tree is available for implementation without massive computing power and ability to use graph library functionalities in scripting languages such as R. As such, a confirmation of our hypothesis implies that the spanning tree network is available as a low complexity algorithm that removes considerable amount of edges (~50% in our work) while maintaining essential genes.

2. Methodology

In this section we describe our method for creating and testing the spanning tree network filter. The four networks we describe in this work are created from datasets GSE5078 (YNG and MID) and GSE5140 (UNT, CRE) in NCBI's Gene Expression Omnibus (GEO) website [9, 12, 13]. The YNG and MID networks represent expression data created from mice at 2 months and 18 months, respectively. The UNT and CRE networks represent 20 month old mice controls (not treated with any drug) and mice at 20 months which had been fed creatine daily since birth to enhance longevity. Both datasets were designed to identify mechanisms behind the processes of aging.

2.1. Network Creation

Networks were created by representing genes as nodes and edges were drawn between two genes if they have a high level of correlation of expression (Pearson correlation, $0.91 \leq \rho \leq 1.00$, p-value <0.0005) [6]. The resulting networks were found to be modular and following a power-law degree distribution, indicative of the presence of clusters and hub nodes within the network [1].

2.2. Metrics of Validation

For each network, original or filtered, we identified lethal nodes as those genes that had previously been tested *in vivo* by knockout mutation. This type of mutation renders the gene unusable by the organism, such that organism growth must continue without the gene. If the knockout is determined to be fatal for the organism, this gene is essential. Information on these mutations for each gene were downloaded from the Mouse Genome Informatics (MGI) database [2] and each gene has been classified as untested (no knockout performed), tested (knockout performed) and lethal (knockout performed was lethal). *It is important to note that untested genes are not included in this analysis.* Untested genes cannot be assumed to be lethal or non-lethal, and as such they are not included in this analysis. We elaborate on the rate of untested genes in each dataset under Results for Figure 6. Using these metrics, we identify the lethality rates of each network as described in Equation 1:

$$Lethality_T = \frac{\#\ lethal\ genes\ in\ T}{\#\ tested\ genes\ in\ T} \qquad (1)$$

In this equation, *t* represents the node set from the network described by the top *t%* of nodes according to degree. For example, in a network of 1,000 nodes, the node set *T=10%* would represent the top 100 nodes according to ranking via degree. For our work, we examined $T = 5$, 10, 15, 20, and 25%. Multiple percentages of *T* were selected because previous studies of centrality in correlation networks [4] found that due to high levels of noise within the network, one level of *t* can misrepresent actual lethality levels of nodes in networks where all essential genes are not known, such is the case in the MGI database.

While the rate of lethality is representative of the essentialness of the chosen set, we also want to determine how lethal the network is compared to the background, or the non-high-degree nodes within the network. To examine this measure, we identify the test set T and the background set B (nodes in network G not included in T) to define the lethality enrichment of the entire network as the following Equation 2:

$$Enrichment = \frac{\#\ lethal\ genes\ in\ T \Big/ \#\ tested\ genes\ in\ T}{\#\ lethal\ genes\ in\ B \Big/ \#\ tested\ genes\ in\ B} \qquad (2)$$

This equation can be simplified to Equation 3, which translates to:

$$Enrichment = \frac{Lethality_T}{Lethality_B} \qquad (3)$$

Using these metrics, we can identify how well two filtered networks compare to each other, or how well two networks compare to the original in terms of ability to identify or enhance lethal hub genes

The filters as described in Table 1 were used in this research. Each filter represents some variation of network structure being retained after filtering. Typically, chordal filters are used to identify clusters or sets of nodes with a very dense community structure. The Spanning Tree is our testing filter, chosen as potential candidate for identifying essential hub nodes. The random walk filter was used in preliminary results as a random control with which to compare highly clustered network chordal filters.

Table 1: Description of Chordal-Based Filters (at left) and Tree-based filters (at right).

Chordal-based filters			**Tree-based filters**		
Filter	***Name***	***Description***	***Filter***	***Name***	***Description***
HD	High Degree	Traversal based on ascending order of vertices	ST	Spanning Tree	Tree determined by Prims Algorithm
LD	Low Degree	Traversal based on descending order of vertices			

3. Results

The results of our analysis highlight our original hypotheses, H_O. We restate our hypotheses now, that *tree-based network filters will identify high-degree nodes in some network G while reducing the number of edges within that network compared to a chordal-based graph filter.* We compare the spanning tree filter to HD and LD in terms of lethality and lethality enrichment to compare performance in a real network setting. We elaborate on the results of studies designed to prove or disprove this hypothesis below.

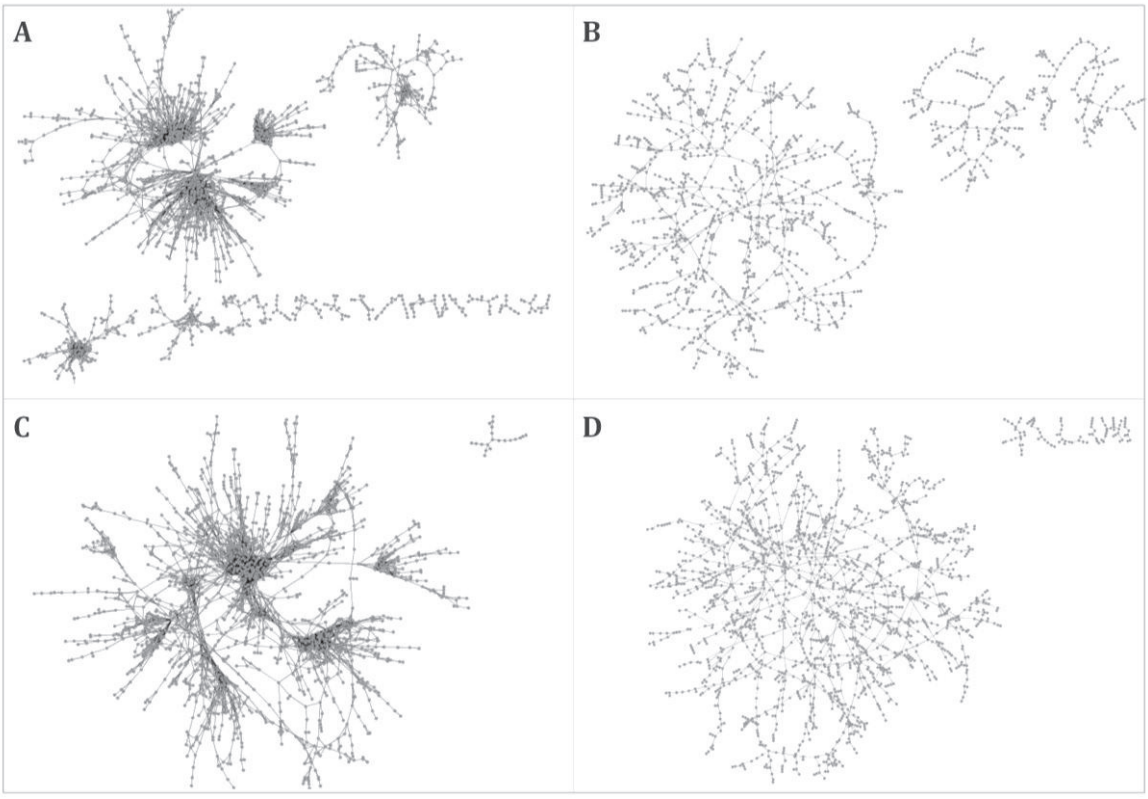

Figure 1: Original networks versus minimum spanning tree. (A) The YNG network created as described in Methods with 5,348 nodes and 7,277 edges. (B). The YNG MST filtered network with 5,348 nodes and 3,885 edges. (C) The MID network created as described in Methods with 5,549 nodes and 7,178 edges. (D). The MID MST filtered network with 5,549 nodes and 4,154 edges.

3.1. Edge Reduction

The Spanning Tree filter results in networks with the least amount of edges compared to original, HD, and LD. High degree and low degree chordal filters were found in our previous work [3, 11] to be good candidates for consistently identifying clusters that were found in the original network over finding new or hidden clusters. For their ability to maintain key network structures as chordal filters, they were chosen to compare to our spanning tree filter and the original networks. We first find that in terms of network size, all filters correctly maintain all nodes from the original networks as shown in Figure 2 (5,348 in YNG and 5,549 in MID), but that the spanning tree filtered network has the least amount of edges, with 0.027% edge densities for both YNG and MID networks, compared to ~0.030-0.040% edge densities for the HD and LD networks, and 0.04-0.05% edge densities in the original networks. YNG and MID original graph representations are shown in Figure 1A and 1C with their respective spanning tree filtered versions represented in Figure 1B and 1D. Both spanning tree filtered networks have just over *half* the number of edges found in the original networks. However, it should be noted than chordal filtered networks have their own niche in biological network applications. It has been previously shown that chordal graph-based filters are well-suited for identifying clusters present in the original network while enhancing their functional understanding, and also these types of filters are able to identify new functions previously hidden by noise (Figure 7).

3.2. Lethality and Enrichment

The spanning tree filter performs better than HD and LD on lethality and lethality enrichment studies, and sometimes even original networks in terms of finding lethal genes. Figure 3 represents the lethality for ORIG,

spanning tree, HD and LD filters for YNG and MID networks. We find that the spanning tree performs better on lethality tests than the original network in the YNG network, and better than or equal to the original network in 3/5 tests for the MID network. In a few instances, the LD filter identifies lethal genes as well as or better than the spanning tree filter, but this result is not consistent among all networks (2/10 tests). The same is true for the HD filter, performing better than the spanning tree filter in 2/10 tests. Lethality enrichment, where the top *T* nodes are compared to background, is shown in Figure 4. For the YNG network results, all measures were within less than 1.0 enrichment of each other, and spanning tree enrichment p erforming equally with other filters in all tests. The MID network found the spanning tree filter performing better than the original network in all 5 tests, and only being outperformed by the LD filter at the *T= 25%* measurement, with the spanning tree filter having an enrichment of 0.971 and the LD filter having an enrichment score of 0.971. These two sets of results confirm our hypothesis, that the spanning tree filter can identify lethal genes *as well as* the original network, and in 12/20 of the cases, spanning tree can identify lethal genes based on degree better than the original network. In 4/20 of thes e studies, it performs almost as well as the original network, and only in 4/20 cases is the spanning tree filter outperformed by the original network. In terms of comparison to the HD and LD networks, the spanning tree filter outperforms the LD and HD filters in 31/40 tests.

Network	Filter	Nodes	Edges	Possible Edges	Edge Density
YNG	ORIG	5,348	7,277	14,297,878	0.0509%
MID	ORIG	5,549	7,178	15,392,926	0.0466%
YNG	ST	5,348	3,885	14,297,878	0.0272%
MID	St	5,549	4,154	15,392,926	0.0270%
YNG	HD	5,348	4,518	14,297,878	0.0316%
MID	HD	5,549	4,757	15,392,926	0.0309%
YNG	LD	5,348	5,811	14,297,878	0.0406%
MID	LD	5,549	5,886	15,392,926	0.0382%

Figure 2: Network sizes for ORIG, HD, LD, and MST filters.

3.3. Avera ge Lethality and Enrichment

Averages of lethality and enrichment reflect these same patterns over multiple measures of T. Dempsey and Ali 2012 [4] noted that lethality in correlation networks should be measured over multiple thresholds to approximate lethality and better obtains a vision of true lethal counts in networks where the entire set of essential genes is not known. Such is the case in this data, where all essential genes are not known in the mouse genome. As such, we present the average of the lethality and enrichment scores for each filter in Figure 5. On average, the spanning tree performs better than the HD and LD filters in both lethality and enrichment, and better than the original in half the cases tested. Considering the fact that edge density for these networks is almost half the original and ~ 25% less than the other chordal filters, this presents compelling evidence that the tree-based filters are better for identifying and/or predicting lethality in unknown networks.

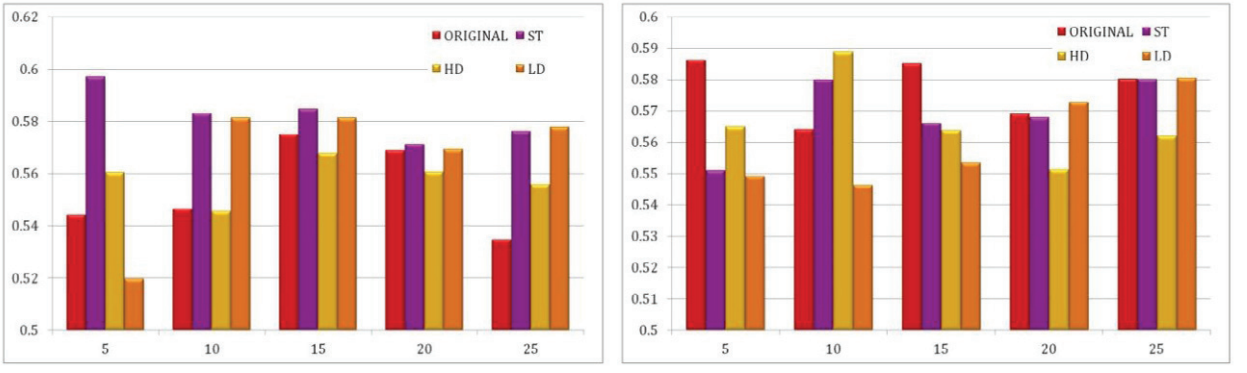

Figure 3: Lethality results for YNG (left) and MID (right).

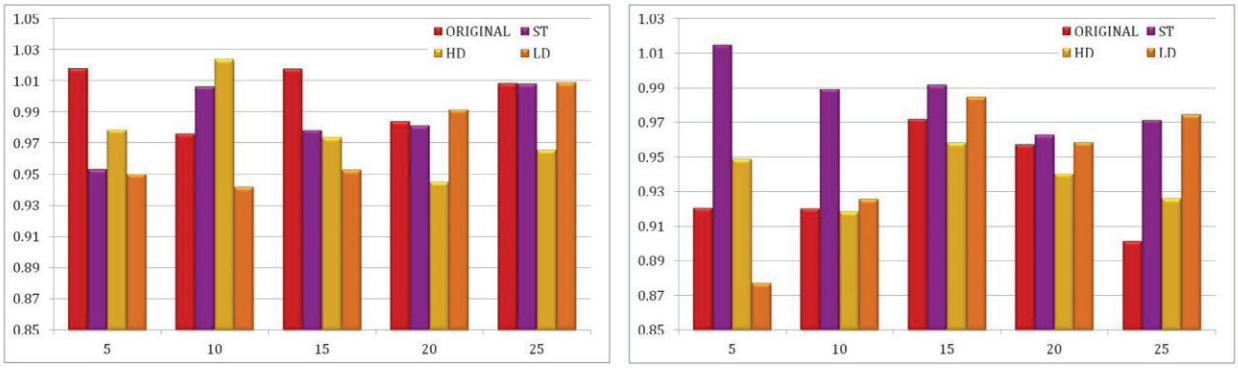

Figure 4: Lethality Enrichment results for YNG (left) and MID (right).

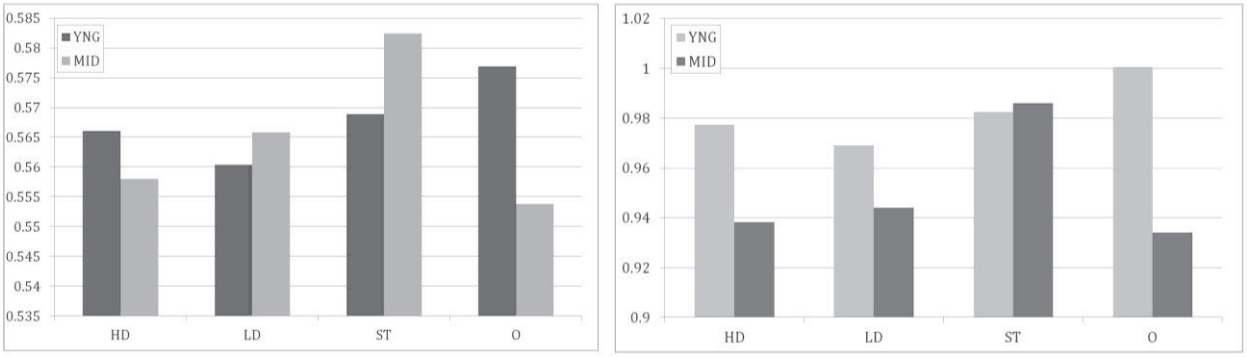

Figure 5: Average lethality (left) and average enrichment (right) for all four filters.

3.4. Untested genes

Over half of the genes in the original result remain untested for in vivo knockout. It should again be noted that the results for these networks can be skewed due to the incompleteness of the essentiality studies in mouse; indeed, it is likely that enrichment scores will improve with database maturation. It has been found [4] that essential genes in the mouse genome are slightly more likely to be tested for essentiality due to their blatantly large roles in critical model function, so it can be speculated that many of the remaining genes may actually not be mostly essential. In our study, we found that 56.325% of genes remained untested for *in vivo* knockout, over 22,000 genes in the mouse genome. It is implausible to expect all these genes to be tested within a reasonable amount of time, but we *can narrow the scope of genes to the top nodes via degree based on spanning tree degree distribution*. If we take predicted but untested nodes from the 5% thresholds, we find that there are on average 136-146 untested genes, representing less than 0.3% of the original dataset size. Even if the largest threshold of 25% *T* is used, we see in Figure 6 that this suggests about 750 genes for testing (less than 2% of the original dataset). While this set is a large one, it is not insurmountable by current testing standards in a matter of years.

	T	**Average Untested**		***T***	**Average Untested**
YNG	5	146.6	**MID**	5	136.4
	10	293		10	259.4
	15	444.7		15	419.7
	20	597.6		20	563.3
	25	742.1		25	709.9

Figure 6: Average number of untested nodes in the top *T* of genes ranked by degree for all filters, including original.

4. Discussion

In this work we have presented our evidence for different network filters relaying different network structures. These studies are critical for research focused on complex systems biology, and can be applied within reason to other types of networks built from high-throughput data. It has

	GO Term	Original	1p	2p	4p	8p	16p	32p
Cluster 1	cellular amino acid, derivative metabolic process		0.15		0.15	0.15	0.15	0.15
	cellular component organization		0.60		0.60	0.60	0.60	0.60
	nitrogen compound metabolic process		0.03		0.03	0.03	0.03	0.03
Cluster2	segment specification				0.24	0.24	0.24	
	cell surface receptor linked signal transduction				2.58	2.58	2.58	
	nervous system development				1.32	1.32	1.32	
Cluster3	nucleo -base/-side/-tide, acid metabolic process	2.06	2.06	2.06	2.06	2.06	2.06	2.06
	cell motion		0.49	0.49	0.49	0.49	0.49	0.49
	system development	1.10	1.10	1.10	1.10	1.10	1.10	1.10
	nervous system development	0.68	0.68	0.68	0.68	0.68	0.68	0.68
	ectoderm development		0.77	0.77	0.77	0.77	0.77	0.77
	nuclear transport		0.05	0.05	0.05	0.05	0.05	0.05
	primary metabolic process		4.53	4.53	4.53	4.53	4.53	4.53
	protein transport		0.84	0.84	0.84	0.84	0.84	0.84
	intracellular protein transport		0.84	0.84	0.84	0.84	0.84	0.84
	mesoderm development	0.84	0.84	0.84	0.84	0.84	0.84	0.84
	intracellular signaling cascade		0.85	0.85	0.85	0.85	0.85	0.85
Cluster4	ion transport			0.21				0.21
	oxygen, reactive oxygen species metabolic process			0.02				0.02
	response to toxin			0.03				0.03
	transport			0.80				0.80
	anion transport			0.04				0.04
Cluster5	vitamin transport	0.04	0.14					
	transport	1.38	4.48					

Figure 7: Reprinted from Dempsey *et al*. 2011 [3] with permission: An example of how chordal based-filter BFS performs on YNG network at different processors. A number in a column represents a cluster with the function of its respective row present at that specific filter; for example, the original network had 2 clusters enriched with 4 functions (C3) and 2 functions (C5). The BFS filter also found those clusters in sequential (1P) and parallel (2-32p) states, as well as discovered new functions within the cluster based on noise removal.

been shown previously that chordal-based filters are well-suited f or identifying clusters in a network, as they are able to retain groups of nodes whose edges are tightly connected while removing density, and in many cases, noisy edges. We have also shown that tree-based filters are better suited than chordal-based filters for identifying hub genes, and in turn, identifying essential genes within the network. While tree-based filters outperform the chordal-based filters in this fashion, tree-based methods are in return poor samplers of clusters within a network, as trees do not contain clusters. However, we find that the tree-based methods retain and can even improve upon the identification of known lethal genes, and further, these filters offer a new, smaller set of prime targets for *in vivo* knockout testing. The concept is that to approximate a large complex network, one must use a variety of filters to explore the implications of that structure on a sampled network. Future work for this type of study includes performing tree-based filtering on multiple datasets taken from mouse data, and computing the most common untested but highly degree ranked genes in spanning tree based filters to further identify possible essential genes; it would be appropriate to suggest that genes that frequently and consistently occur as hub nodes in datasets would become the best targets for essential gene testing, thus narrowing the scope even further.

Acknowledgements

The authors would like to acknowledge Kanimathi Duraisamy for her help with preparation of parallel chordal results. The authors also would like to thank the Nebraska EPSCoR First Award, the Nebraska INBRE Program and the College of IS&T, University of Nebraska at Omaha for supporting this research.

References

1. Barabasi, A. L., & Oltvai, Z. N. (2004). Network biology: Understanding the cell's functional organization. Nature Reviews.Genetics, 5(2), 101-113.
2. Bult, Cj, Eppig JT, Kadin JA, Richardson JE, Blake JA; and the members of the Mouse Genome Database Group. 2008. The Mouse Genome Database (MGD): mouse biology and model systems. Nucleic Acids Res 36(Database Issue):D724-8.

3. Dempsey, K., Duraisamy, K., Ali, H., & Bhowmick, S. (2011). A Parallel Graph Sampling Algorithm for Analyzing Gene Correlation Networks. Proceedings of the 2011 International Conference on Computational Science.
4. Dempsey, K. and Ali, H. On the discovery of Cellular subsystems in correlation networks using centrality measures (2012). Current Bioinformatics. Publication pending.
5. Dong, J., & Horvath, S. (2007). Understanding network concepts in modules. BMC Systems Biology, 1, 24.
6. Ewens, W. J., & Grant, G. R. (2005). Statistical methods in bioinformatics (Second Edition ed.). New York, NY: Springer.
7. Jeong, H., Mason, S. P., Barabasi, A. L., & Oltvai, Z. N. (2001). Lethality and centrality in protein networks. Nature, 411(6833), 41-42.
8. Opgen-Rhein, R., & Strimmer, K. (2007). From correlation to causation networks: A simple approximate learning algorithm and its application to high-dimensional plant gene expression data. BMC Systems Biology, 1, 37.
9. Verbitsky, M., Yonan, A. L., Malleret, G., Kandel, E. R., Gilliam, T. C., & Pavlidis, P. (2004). Altered hippocampal transcript profile accompanies an age-related spatial memory deficit in mice. Learning & Memory (Cold Spring Harbor, N.Y.), 11(3), 253-260.
10. Subramanian, A., Tamayo, P., Mootha, VK., Mukherjee, S., Ebert, BL., Gilette, MA., Paulovich, A.,Pomeroy, SL., Golub, TR., Lander, ES., and JP Mesirov (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wise expression profiles. Proc Natl Acad Sci 102(43):15545-15550.
11. Duraisamy, K., Dempsey, K., Ali, H., and S. Bhowmick (2011). A noise reducing sampling approach for uncovering critical properties in large scale biological networks. High Performance Computing and Simulation 2011 International Conference (HPCS): July 4-8. Istanbul, Turkey.
12. Bender, A., Beckers, J., Schneider, I., Holter, SM. *et al*. (2008). Creatine improves health and survival of mice. Neurobiol Aging 29(9):1404-11.
13. Edgar, R., Domrachev, M., and AE. Lash (2002). Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nuc Acid Res 30(1):207-10.
14. Gleich, D. 16 May 2009. Gaimc: Graph Algorithms in Matlab Code. *Mathworks.com*. Obtained on 01.17.2012, from http://www.mathworks.com/matlabcentral/fileexchange/24134
15. Yoon, JS and WH Jung (2011). A GPU-accelerated bioinformatics application for large-scale protein interaction networks. APBC poster presentation.
16. Christakis, NA., *et al*. (2007). The spread of obesity in a large social network over 32 years. New England Journal of Medicine.357 no 4:370-379.