BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 8 2006, pages 974–980
doi:10.1093/bioinformatics/btl030

Systems biology

Efficient estimation of graphlet frequency distributions in protein–protein interaction networks

N. Pržulj[1,3], D. G. Corneil[1] and I. Jurisica[1,2,*]

[1]Department of Computer Science, University of Toronto, Toronto M5S 3G4, Canada, [2]Ontario Cancer Institute, Division of Signaling Biology, Toronto M5G 1L7, Canada and [3]Department of Computer Science, UC Irvine, Irvine, CA 92697-3435, USA

Received on May 19, 2005; revised on January 25, 2006; accepted on January 26, 2006
Advance Access publication February 1, 2006
Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: Algorithmic and modeling advances in the area of protein–protein interaction (PPI) network analysis could contribute to the understanding of biological processes. Local structure of networks can be measured by the frequency distribution of graphlets, small connected non-isomorphic induced subgraphs. This measure of local structure has been used to show that high-confidence PPI networks have local structure of geometric random graphs. Finding graphlets exhaustively in a large network is computationally intensive. More complete PPI networks, as well as PPI networks of higher organisms, will thus require efficient heuristic approaches.

Results: We propose two efficient and scalable heuristics for finding graphlets in high-confidence PPI networks. We show that both PPI and their model geometric random networks, have defined boundaries that are sparser than the 'inner parts' of the networks. In addition, these networks exhibit 'uniformity' of local structure inside the networks. Our first heuristic exploits these two structural properties of PPI and geometric random networks to find good estimates of graphlet frequency distributions in these networks up to 690 times faster than the exhaustive searches. Our second heuristic is a variant of a more standard sampling technique and it produces accurate approximate results up to 377 times faster than the exhaustive searches. We indicate how the combination of these approaches may result in an even better heuristic.

Availability: Supplementary information is available at http://www.cs.toronto.edu/~natasha/BIOINF-2005-0946/Supplementary.pdf
Software implementing the algorithms is available at http://www.cs.toronto.edu/~natasha/BIOINF-2005-0946/estimate_grap-hlets.html

Contact: juris@cs.toronto.edu; natasha@igor.ics.uci.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Protein–protein interactions (PPIs) are commonly modeled by graphs, where nodes represent proteins and edges represent physical interactions between the corresponding proteins. To date several large PPI networks have been accumulated for multiple organisms (Ito *et al*., 2000; Uetz *et al*., 2000; Gavin *et al*., 2002; Ho *et al*., 2002; Giot *et al*., 2003; Li *et al*., 2004). Modeling and understanding the structure of these large networks is an important problem with profound biological implications (Lappe and Holm, 2004), which requires new mathematical and computational advances (de Aguiar and Bar-Yam, 2005; Itzkovitz *et al*., 2005; Pržulj *et al*., 2004; Song *et al*., 2005; Stumpf *et al*., 2005). Global properties of these networks, such as their degree distribution, have been extensively studied and PPI networks have been shown to have scale-free degree distributions. However, since current PPI networks are still incomplete, and contain localized and biased biological experiments, local approaches to analyzing the structure of these networks have recently been proposed (Milo *et al*., 2002; Shen-Orr *et al*., 2002; Pržulj *et al*., 2004). Small subgraphs that appear in a biological network at significantly higher frequencies than expected in randomized networks are called network motifs and they are believed to represent significant evolutionary conserved modules (Milo *et al*., 2002; Shen-Orr *et al*., 2002). However, this approach has recently been criticized, since it is sensitive to the choice of network randomization as a null hypothesis in testing statistical significance (Artzy-Randrup *et al*., 2004). Also, it has been argued that global structural features of networks, such as the clustering coefficient, are intertwined with local structural properties (Vazquez *et al*., 2004). Furthermore, there is a growing body of literature showing that scale-free (Barabási and Albert, 1999) and hierarchical (Ravasz *et al*., 2002) network models may be inadequate for PPI networks (de Aguiar and Bar-Yam, 2005; Pržulj *et al*., 2004; Stumpf *et al*., 2005; Han *et al*., 2005; Keller, 2005) and a new, geometric random graph model, has been proposed for these networks (Pržulj *et al*., 2005). Clearly, analyzing and modeling PPI networks has lately become an active and a controversial research topic.

A different approach to studying local structural properties of large networks, based on the assumption that it is equally important to understand infrequent as it is to understand frequent network subpatterns, has recently been proposed (Pržulj *et al*., 2004). Small induced subgraphs (see Section 2.1) of a large network, regardless of whether or not they appear in the network at significantly higher frequencies than expected in randomized networks, are called graphlets; their frequency distribution in networks has been used to define a new measure of similarity between large networks and introduce a new, better fitting, geometric random graph model for high-confidence PPI networks (Pržulj *et al*., 2004). Just as it was shown that the scale-free model is inferior to the geometric random graph model for high-confidence PPI networks, it is possible that

*To whom correspondence should be addressed.

some other model will be shown to be superior to the geometric random graph model. Regardless, graphlet frequency is a key measure of networks (note that the scale-free model is based on the distribution of the graphlet with two nodes and one edge) and it is fully expected that graphlet information will probably play a major role in subsequent PPI network modeling. Finding the occurrences of small subgraphs in large networks is computationally intensive and exhaustive searches become computationaly infeasible even when applied to small, currently available PPI data. Thus, regardless of the specific model, heuristic algorithms must be available to analyze large instances of observed data and one can only develop such algorithms based on the currently accepted (perhaps not universally) models. In this paper, we show that a sampling technique restricted to a specific part of the graph gives a very good insight into the global structure, at least as seen for the existing high-confidence data and the geometric random model. Hopefully, this technique will be adaptable to other, future, more sophisticated models than the geometric random model.

Currently available PPI networks are largely incomplete and thus represent just a small fraction of real, complete PPI networks. In addition, PPI networks of higher organisms will be much larger. For example, humans have less than 30 000 genes, each of which can have 4–6 splice variants, and therefore, including >200 possible post-translational protein modifications, humans are expected to have at least hundreds of thousands of proteins and millions of interactions between them. In addition, plant genome sizes are much larger (IRGSP, 2005; Arumuganathan and Earle, 1991). Anticipating the arrival of these large PPI networks, we need to make sure that our algorithmic techniques are scalable and ready for processing them. Since exhaustive searches are already computationally infeasible, it is important to find well fitting network models for high-confidence PPI data and use these models to generate large realistic networks on which we can develop and test new algorithms.

A heuristic random sampling approach for detection of network motifs has been proposed by Kashtan *et al*. (2004). Their algorithm efficiently estimates subgraph concentrations in networks with hubs; however, this algorithm is slower than the exhaustive search algorithm for networks without hubs.

We propose two heuristic approaches for estimating graphlet frequency distributions in high-confidence PPI networks and the corresponding model networks. Since random, scale-free and hierarchical network models have been shown to inadequately model PPI networks (de Aguiar and Bar-Yam, 2005; Pržulj *et al*., 2004; Stumpf *et al*., 2005), we focus on a better fitting, geometric random graph model of high-confidence PPI networks (Pržulj *et al*., 2004). The first heuristic approach, called Targeted Node Processing (TNP), uses the structure of PPI and geometric random graphs to achieve accurate graphlet frequency distribution estimates, 300–690 times faster than the exhaustive searches. The second approach, called Neighborhood Local Search (NLS), using a more standard random sampling technique, produces accurate graphlet frequency distribution estimates, 95–377 times faster than the exhaustive searches with surprisingly few samples. Importantly, both of our heuristic approaches work well for high-confidence PPI networks, which have scale-free degree distributions and contain hubs, as well as for geometric random networks, which have Poisson degree distributions and lack hubs. Thus, it is not the presence or absence of hubs that dictates the behavior of these heuristics, as was the case in the Kashtan *et al*. algorithm (Kashtan *et al*., 2004), but the local structure of the networks. This feature of our algorithms is important because of an increasing evidence that degree distributions of biological networks may not be scale-free (Pržulj, 2005; Tanaka, 2005; Tanaka *et al*., 2005; Keller, 2005).

1.1 Background

Given graphs G and H, determining whether G contains a subgraph isomorphic to H is NP-complete, since it includes problems such as Hamiltonian path/cycle, and the maximum clique as special cases (Garey and Johnson, 1979). If graph G on n_G nodes is input and graph H on n_H is fixed, then the subgraph isomorphism can be tested in polynomial time. $O(n_H! \cdot n_H^2 \cdot \binom{n_G}{n_H})$, by iterating through all subsets of n_H nodes of G. However, such exhaustive searches are computationally infeasible for large networks and thus heuristics are needed.

Examples of efficient approximate subgraph counting algorithms include sampling algorithms for counting classical graph structures such as Hamiltonian cycles and spanning trees in graphs (Dyer *et al*., 1994; Jerrum, 2003). An $O[(n\,1/\ln\ln n) \cdot M(n)]$ algorithm for finding an approximate number of induced copies of H in G for a given undirected labeled n-node graph G and each graph H in a list of labeled k-node graphs was developed (Duke *et al*., 1995), where $M(n)$ is the time needed to square an $n \times n$ matrix with 0, 1-entries. This algorithm has strong constraints on the subgraph's size for a given size of G and it is limited to 3-node subgraphs on a network with hundreds of thousands of nodes. Running times of the above algorithms asymptotically depend on the network size, which is impractical for large networks. A probabilistic random sampling algorithm for estimating subgraph counts for small subgraphs, whose runtime does not asymptotically depend on network size, has recently appeared (Kashtan *et al*., 2004).

2 METHODS

2.1 Definitions

A graph is denoted by G, or $G(V, E)$, where V is the set of nodes and $E \subseteq V \times V$ is the set of edges of G. We use n to represent the number of nodes, $|V|$, and m to represent the number of edges, $|E|$. We use $V(G)$ to represent the set of nodes and $E(G)$ to represent the set of edges of a graph G. Nodes joined by an edge are called adjacent. A neighbor of a node v is a node adjacent to v. We denote by $N(v)$ the set of neighbors of node v (called the neighborhood of v). The degree of a node is the number of edges incident with the node.

A path in a graph is a sequence of nodes and edges, such that a node belongs to the edges before and after it (except for the first and last node, which only belong to the first and last edge, respectively) and no nodes are repeated. A path with k nodes is denoted by P_k and its length is the number of edges in the path. The shortest path length between nodes u and v is the distance between u and v, denoted $d(u, v)$. The diameter of a graph is the maximum of $d(u, v)$ over all nodes u and v, denoted $\mathrm{diam}(G) = \max_{u,v \in G} d(u, v)$. A graph is connected if there exists a path between each pair of its nodes; otherwise, it is disconnected, and its diameter is equal to the maximum diameter of its connected components.

The eccentricity of a node v in G is the maximum distance from v to other nodes of G, i.e. $\mathcal{E}_G(v) = \max_{u \in V(G)} d(u, v)$. The radius of G, is the minimum of node eccentricities of G, i.e. $\mathrm{rad}(G) = \min_{u \in V(G)} \mathcal{E}(u)$. For all graphs G, $\mathrm{rad}(G) \leq \mathrm{diam}(G) \leq 2 \cdot \mathrm{rad}(G)$. Note that for all the networks that we study here, the diameter is significantly larger than the radius. The center of graph G is the subgraph of G induced by the nodes of minimum eccentricity. Thus, if the eccentricity of a node is close to the radius of the network, the node is close to the center of the network; if the eccentricity

is close to the diameter of the network, the node is close to the fringe of the network.

A subgraph of G is a graph whose nodes and edges belong to G. An induced subgraph H of G, is a subgraph of G on $V(H)$ nodes, such that $E(H)$ consists of all edges of G that connect nodes of $V(H)$. A graphlet is a small connected induced subgraph of a network (Pržulj *et al*., 2004). All 3-, 4- and 5-node graphlets are presented in Supplementary Figure 1. We focus on analyzing frequency distribution of these 29 graphlets.

2.2 Data and model networks

Our heuristic algorithms have been designed to work well on the high-confidence PPI and geometric random networks. We used the two yeast *Saccharomyces cerevisiae* PPI networks described in Pržulj *et al*. (2004b) and Pržulj (2005): the high-confidence PPI network (von Mering *et al*., 2002) and the PPI network comprising the top 11 000 interactions (von Mering *et al*., 2002). It has been shown that both these networks are accurately modeled by geometric random graphs (Pržulj *et al*., 2004). In addition, we tested the performance of our heuristic on more noisy data (King *et al*., 2004; Pržulj *et al*., 2004b): the higher-confidence and the entire currently available fruitfly *Drosophila melanogaster* PPI networks (Giot *et al*., 2003).

We used a variant of geometric random graphs as in (Pržulj *et al*., 2004) (denoted by GEO): nodes correspond to uniformly randomly distributed points in bounded two-, three- and four-dimensional Euclidean space (denoted by GEO-2D, GEO-3D and GEO-4D, respectively) and two nodes in the graph are adjacent if the corresponding points are close enough in the metric space, where closeness is measured by the Euclidean distance norm. The geometric random graph model networks corresponding to the PPI networks described in (Pržulj *et al*., 2004) were used in this study. In addition to geometric random graphs, we tested the performance of our algorithms on the following model networks: (1) Erdös-Rényi random graphs (Erdös and Rényi, 1959, 1960) with the same scale-free degree distributions as the PPI networks (denoted by ER-DD; these networks are also called random scale-free networks) and (2) scale-free Barabási–Albert networks (Barabási and Albert, 1999) (denoted by SF).

2.3 Algorithms

2.3.1 Targeted node processing (TNP) This heuristic approach identifies a small part of the network in which graphlets can be quickly found exhaustively, and then uses the obtained graphlet frequency distribution to estimate the graphlet frequency distribution in the entire network.

Geometric random networks used to model PPI networks have a boundary that is sparser than the rest of the network. Apart from the boundary, the rest of a geometric random network has a uniform structure, since it corresponds to uniformly randomly distributed points in a bounded space. Also, the diameters of these networks are almost twice their radii, indicating that these networks are 'stretched' as far as possible (see Supplementary Tables 10 and 15). The same stretched structure with a boundary is observed for the yeast and fruitfly PPI networks that we studied (see Supplementary Table 5). Therefore, we hypothesized that graphlets on the sparse boundary of these networks could be quickly found exhaustively and, owing to the uniformity inside these networks, the graphlet frequency distributions obtained in this way would be representative of graphlet frequency distributions of the entire networks.

To test this, we performed the following experiments. We started by 'processing' nodes one at a time, as in the exhaustive search (Algorithm 1 in Supplementary information), i.e. looking for all induced subgraphs of size 3, 4 and 5 containing the node. However, in order to separate nodes that are 'easy to process' (i.e. for which we can find all graphlets that contain that particular node in a reasonable amount of time) from the nodes that are 'hard to process,' we started with limiting the processing time given to each node to get 'finished,' or 'processed'; by a 'finished node', or a 'processed node' we mean that for that node, it was possible to exhaustively find all induced 3-, 4- and 5-node subgraphs containing it in the alloted amount of time. The basics of a single node processing to detect all 3- and 4-node graphlets containing the node are presented in Algorithm 2 in Supplementary information (see also Sections 1 and 3.1 in Supplementary information).

If all subgraphs containing a node cannot be exhaustively found in the alloted time, we declare the node 'unfinished,' and discard all induced subgraphs that were found by the partial processing of that node. In the end, we correct for over-counting, as in the exhaustive search algorithm (see Section 1 in Supplementary information). After we processed all nodes of a large network in this way, we compared the properties of the finished and unfinished nodes. As expected, the finished nodes have low degree and are on the periphery of the network, i.e. high-degree nodes deeper in the network are harder to process (see Supplementary Tables 5, 6 and 10–16). Since this approach is based on processing only the nodes that can be processed fast, we call this heuristic approach the 'Time-Limited Node Processing' (TLNP) (see Section 3.1 in Supplementary information).

Since low-degree nodes on the 'edge' of a network can be processed quickly, we first sort the nodes by increasing degree and then by decreasing eccentricity. The top nodes in this list are of lowest degree and on the fringe of the network. Examples showing degrees and eccentricities of the top 2% of nodes sorted in this way are presented in Supplementary Table 1. We process the top x% of nodes sorted in this way (i.e. we exhaustively search for all 3-, 4- and 5-node graphlets initiating the search at these nodes as described in Algorithm 2 of Supplementary information, without bounding the search time) and add up the resulting graphlet frequencies correcting for over-counting. The larger x is, the closer the estimated graphlet frequency distribution is to the exact graphlet frequency distribution. Thus, the estimated graphlet frequency distribution converges to the fully enumerated one. The resulting estimated graphlet frequency distribution patterns are surprisingly close to the exact graphlet frequency distribution patterns for PPI and geometric random networks even when x is very small, such as $x < 1$% (detailed results are presented in Section 3.1 below and in Section 3.1 of Supplementary information). Notice that this is not the case for SF and ER networks (see Section 3.1 below and Section 3.1 of Supplementary information). Since for every heuristic, an example could be constructed on which it would perform poorly. one has to focus on designing a heuristic for a particular application. In our case, since the problem is computationally expensive in general, we focused on finding a heuristic that works well for PPI and geometric random graphs. We exploit the structure of our data and geometric random graph model networks to design such a heuristic.

This heuristic results only in the under-counting of graphlets as a type of deviation from the exact number of graphlets. Since full, time unlimited processing of selected nodes is performed by this heuristic approach, we call this heuristic 'Targeted Node Processing' (TNP).

2.3.2 Neighborhood local search (NLS) NLS randomly chooses a seed node in a network and searches in its neighborhood for a specific graphlet. While the TNP approach processes only the fringe of the network, NLS randomly samples the network and each part of the network has the same probability to be sampled [similar to the Kashtan *et al*. (2004) algorithm]. However, unlike in the Kashtan *et al*. (2004) algorithm, we do not correct for non-uniform sampling caused by the existence of hubs. Thus, our approach works well for PPI and geometric random networks, but not for SF and ER-DD networks (details are below). We can adjust how 'hard' we search for a graphlet. Since we do not just randomly pick a subgraph, but rather search in the neighborhood of a seed node for a specific graphlet, if we choose an extensive search for a graphlet that rarely or never occurs in the network, our NLS algorithm will have a large running time. However, our NLS algorithm resulted in two interesting observations. First, correction for non-uniform sampling is not needed in PPI and geometric random networks, but it is needed in Erdös–Rényi and Scale-Free networks (details are presented in Section 3.2 below and Section 3.2 of Supplementary information). Second, taking as few as 100 samples per n-node, m-edge subgraph was enough to get graphlet frequency distributions in PPI and the corresponding geometric random networks that are very close to the exact graphlet frequency distributions (the definition of 'close' is described in Section 2.4).

The basics of the NLS approach are presented in Algorithm 3 of Supplementary information. NLS starts with a randomly chosen seed node v and puts into a set of nodes called Neighbors the node v and the set of all nodes at distance at most $n-1$ from v. Then it randomly selects a set called Subnodes of n connected nodes in Neighbors and checks if the subgraph G_s of G induced on Subnodes has m edges. If it does, it returns it and stops: otherwise, it searches in the neighborhood of G_s for a subgraph with n nodes that has closer to m edges than G_s. It does this by executing a sequence of NUM-MOVES moves. A move consists of swapping a random node in the set $N(G_s)$ of nodes in the neighborhood of G_s and a node in G_s if by doing so the number of edges in G_s gets closer to m. In this way, we are doing a local search for an n-node, m-edge subgraph of G. The total number of moves is bounded by NUM-MOVES. To prevent local minima, NLS executes diversification every DIV-FREQth move: it swaps a node in G_s with a node in $N(G_s)$ without asking for an improvement in the number of edges.

The whole procedure of searching for an n-node, m-edge induced subgraph is repeated NUM-EXP times for each of the n-node, m-edge subgraphs, where $m \in \{n-1, \ldots, (n(n-1)/2)\}$. If an n-node, m-edge graphlet is found in an experiment, we determine which graphlet it is isomorphic to (as in Algorithms 1 and 2 of Supplementary information) and increase the number of found instances of that particular graphlet. If an n-node, m-edge graphlet is not found, we proceed to the next experiment in the sequence. Note that in this way, we search NUM-EXP times for all n-node, m-edge graphlets. For example, there is one 3-node, 2-edge graphlet (a P_3), one 3-node, 3-edge graphlet (a triangle), but there are two 4-node, 3-edge graphlets (graphlets 3 and 4 in Supplementary Figure 1), three 5-node, 4-edge graphlets (graphlets 9–11 in Supplementary Figure 1), five 5-node, 6-edge graphlets (graphlets 17–21 in Supplementary Figure 1), etc. Thus, we do NUM-EXP experiments to sample all of the five 5-node, 6-edge graphlets. This heuristic works well for estimating graphlet frequency distributions in PPI and geometric random networks. The description of the results and their dependence on the choice of search parameters is presented in Section 3.2.

2.4 Distance measure

We computed the distances between the results of the exhaustive and heuristic graphlet searches using the relative graphlet frequency distance measure as in (Pržulj *et al*., 2004), $D(G,H) = \sum_{i=1}^{29} |F_i(G) - F_i(H)|$, where $F_i(G) = -\log(N_i(G)/T(G))$, $N_i(G)$ is the exact number of graphlets of type i ($i \in \{1, \ldots, 29\}$ in network G (all 29 graphlets are presented in Supplementary Figure 1), $T(G) = \sum_{i=1}^{29} N_i(G)$ is the total number of graphlets in G, $F_i(H) = -\log(H_i(G)/T_H(G))$, $H_i(G)$ is the number of graphlets of type i ($i \in \{1, \ldots, 29\}$) in network G found by the heuristic search algorithm and $T_H(G) = \sum_{i=1}^{29} H_i(G)$ is the total number of graphlets of G found by the heuristic search algorithm.

When we say that a distance is 'low' or 'high', or that graphlet frequency distributions are 'close', we use the following rule of thumb: graphlet frequency distances of 50 or less are considered low (i.e. graphlet frequency distributions are close) and those higher than 50 are considered high. The motivation for this convention was the observed distances between the PPI and the corresponding model networks (Pržulj *et al*., 2004). A more complicated metric for evaluating the distance could be designed, perhaps as a function of the percentage of processed nodes. For example, if a percentage of unprocessed nodes is very low, such as <1%, even otherwise low distances of 25–50 indicate that the heuristic estimates are of low accuracy and that a different heuristic approach needs to be sought (this happens for SF and ER-DD networks; details are below and in Supplementary Table 19 and Supplementary Figures 12–16). However, the above rule of thumb works well and therefore we leave the design of a new metric for future research.

3 RESULTS AND DISCUSSION

We present in-depth results and discussion of the TLNP and TNP heuristics. Since NLS is based on a more standard random sampling technique, we present its results in less detail for comparing it with the TNP approach. In Section 4. we compare TNP and NLS approaches and indicate how the two techniques could be merged into an even better hybrid heuristic.

3.1 Time limited and targeted node processing

3.1.1 PPI networks Supplementary Figure 2 A–C presents the graphlet frequency distributions resulting from the TLNP experiments with different cut-off times applied to two high-confidence yeast and a noisy fruitfly PPI networks (see also Section 3.1 of Supplementary information). Most graphlets in high-confidence yeast PPI networks are uniformly under-counted by this heuristic. The graphlets that are more severely under-counted in both yeast and fruitfly PPI networks are graphlets 4, 10, 11 and 14 [The graphlet numbering scheme is defined in (Pržulj *et al*., 2004) and presented in Supplementary Figure 1]. However, the under-counting of these graphlets is more prominent in the fruitfly PPI networks, which are noisier and thus of scale-free-like structure (Pržulj *et al*., 2004), than in the yeast PPI networks.

All of these highly under-counted graphlets contain graphlet 4 (a graphlet with a 'central' node linked to 3 nodes of degree 1) as an induced subgraph. This is expected, since, as we previously observed, high degree nodes, as well as nodes in dense neighborhoods, get under-counted by this heuristic. Thus, this heuristic graphlet search does not work well on network models with pronounced hub nodes, such as scale-free networks (see Section 3.1.2 of Supplementary information). However, despite the presence of hubs in PPI networks, it works surprisingly well for these networks. This further supports previous observations that PPI networks have a different local structure than scale-free networks (Pržulj *et al*., 2004).

All nodes unfinished by the TLNP experiments with all tested cut-off times belong to the largest connected component of the corresponding PPI network. Most of them are of high degree and deep inside, i.e. close to the center of the network (see Supplementary Tables 5 and 6). Thus, in the TNP approach, we processed the top 10, 20, 30, 40 and 50% of the nodes of the yeast high-confidence PPI network ordered as described above, by a stable sort first in increasing degree and then in decreasing eccentricity order. That is, we did not initiate a search at 90, 80, 70, 60, and 50% of the nodes in this PPI network, respectively. The resulting graphlet frequency distributions and the CPU times taken to process the selected nodes are presented in Supplementary Figure 2 D and Supplementary Table 7, respectively. For the graphlet frequency distribution estimate obtained by TNP processing of 20% of the nodes in the yeast high-confidence PPI network, the ratio of the exhaustive and heuristic search times is $r = (T_E/T_H) \approx 690$ and the distance between the estimated and the exact graphlet counts is low at 45.91. The large speedup factor of 690 of our algorithm is comparable with the Kashtan *et al*. algorithm speedup factor of around 500 (Kashtan *et al*., 2004), although the two algorithms are designed to work on different networks.

3.1.2 Geometric random graphs We tested the TLNP approach on geometric random graphs with sizes and densities comparable with the sizes and densities of PPI networks (Pržulj *et al*., 2004) (see Section 3.1.2 in Supplementary information). In geometric random networks, even when a very large percentage of nodes remains unprocessed, the resulting graphlet distribution pattern is very

close to the exact one (Supplementary Figure 2 E). For example, even when <98% of the nodes in these networks remain unprocessed, the distance between the heuristic and exact graphlet distributions is only between 32.76 and 67.22 (Supplementary Figure 9 and Supplementary Table 8: 3 second TLNP for GEO-3D-6x graphs). Note that the distance of 67.22 happens when we process only 2 out of 988 nodes! The remaining four distances were between 32.76 and 40.60 and they resulted from processing only between 3 and 11 out of 988 nodes of a network. If we process only 11.91–20.24% of the nodes in these networks, the distance falls to 14.80–17.84 (Supplementary Figure 9 and Supplementary Table 8: 30 second TLNP for GEO-3D-6x graphs).

Owing to uniform under-counting of graphlets in geometric random networks, it is sufficient to process a very small fraction of nodes to get a good estimate of the relative graphlet frequency distribution patterns for these networks. That is, the graphlet frequency distribution obtained by this heuristic multiplied by a constant gives a good estimate of the exact graphlet frequency distribution in these networks (see Section 3.1.2 in Supplementary information).

Similar to PPI networks, the nodes that do not get processed in the tested geometric random networks and with the tested TLNP cut-off times are deeper in the network and of higher degree than the nodes that get processed: as we increase the processing cut-off time and allow more and more nodes to get processed, the average degree of both processed nodes and unprocessed nodes grows, while the average eccentricities fall (Supplementary Tables 10, 13 and 14).

A possible explanation of why TLNP works so well on geometric random networks is the following. In this heuristic, we are starting from the nodes on the fringe of the network and 'grabbing' a sample of graphlets that are up to depth 5 from the fringe of the network. Since the structure of these networks is uniform inside the network (note that the boundary has a different structure), it is enough to sample the graphlets that are about 5-deep from the fringe of the network to get the estimate of the distribution of graphlets in the whole network. It can be argued that these networks are of small diameter; thus going 5-deep into the network, we may be reaching its center. However, sampling the center may not even be needed, since the structure of these networks looks the same in all inner parts of the network. This is further supported by the observation that this approach approximates well the graphlet distributions of geometric random networks with diameters of 52–53 (Supplementary Tables 9 and 15 and Supplementary Figure 6: GEO-2D networks).

The TNP accurately approximates the graphlet frequency distributions of geometric random networks. For example, the TNP approach applied to the five three-dimensional geometric random networks corresponding to the yeast high-confidence PPI network with six times as many edges as the PPI network (denoted by GEO-3D 6x) (Pržulj *et al*., 2004) gave the following results. We selected the top 1 and 2% of nodes of these networks ordered by increasing degree and decreasing eccentricity and fully processed them. The resulting heuristic graphlet frequency distributions accurately estimated the results of the exhaustive search (see Supplementary Figure 2 F, Supplementary Figure 11 and Supplementary Tables 17 and 18). Also, the heuristic running times are orders of magnitude lower than the running times of the exhaustive searches. For example, the ratio of exhaustive and TNP heuristic CPU times for the GEO-3D-6x 1 network with 1% processed nodes was $r = (T_E/T_H) \approx 300$ and the distance of the heuristic from the exhaustive graphlet frequency distribution was only 33.

3.1.3 SF and ER-DD networks The under-counting of graphlets in the SF and ER-DD model networks is not uniform and it results in higher graphlet distances between the exact and the estimated graphlet counts, despite the small number of unprocessed nodes (see Supplementary information Table 19 and Figures 12–16). This is caused by the highly frequent 'hub-specific' graphlets, i.e. graphlets with induced graphlet 4, in SF and ER-DD networks, that get severely under-counted by the heuristic (see Section 3.1.2 in Supplementary information).

3.1.4 Limitations More experimentation with a larger number of networks is needed to determine better node selection criteria that would further decrease the processing time and possibly increase the quality of the estimated graphlet distributions. Also, the dependence of graph density, node selection and processing time needs to be understood. Further investigation of the dependence of the 'translation' of the estimated graphlet distributions (and their 'alignments' with the exact ones) on network properties is needed as well.

3.1.5 Conclusions and future directions We have observed that the TNP heuristic approach for estimating the graphlet frequency distribution in a network works well for geometric random graphs and not well for network models with hubs. However, it works surprisingly well for PPI networks despite the fact that they have hubs. Thus, if the true structure of PPI networks, once we obtain more complete data on them, happens to be similar to the structure of a geometric random graph as we expect, this heuristic approach will be adequate for estimating the graphlet distribution patterns in PPI networks and will result in uniform underestimation of the number of graphlets in these networks. In addition, with a decreased fraction of nodes that get processed and thus decreased processing time, the accuracy of the graphlet distribution estimates hardly decreases, which makes this approach very appealing.

As mentioned in Section 1, PPI networks for higher organisms will be much larger than the current yeast and fruitfly ones. Since exhaustive processing of these network and finding their graphlet frequency distributions will not be tractable, we need to use heuristics. We applied the TLNP approach with various cut-off times to a three-dimensional geometric random graph with 100 000 nodes and 750 000 edges (this networks has three times as many edges as the two yeast PPI networks that we analyzed). The resulting estimated graphlet frequency distribution patterns were very close to those obtained by exhaustive searches for other, smaller three-dimensional geometric random networks with similar edge densities (see Section 3.1.4 in Supplementary information). As before, the nodes that got finished by the TLNP experiments were of low degree and on the fringe of the network. Also, as before, the running times of these TLNP experiments were reasonably low even when we randomly rewired as many as 30% of the edges in this network. For the networks with added noise, resulting from random rewiring of edges of this large geometric random network, with the increased amount of noise, TLNP experiments yielded graphlet frequency distribution patterns which were between the graphlet frequency distributions observed for geometric random graphs and Erdös–Rényi networks, as expected.

3.2 Neighborhood local search (NLS)

We analyzed the yeast high-confidence PPI network and the corresponding model networks using the NLS approach (described in Section 2.3.2) with the following choice of search parameters: maximum number of experiments is two, maximum number of moves per experiment is five, diversification frequency is three and diversification duration is one (i.e. every third move is random in the neighborhood of the selected subgraph). We experimented with different numbers of seed nodes: for each graph $G(V, E)$ processed by this heuristic, we performed experiments using $|V|/8$, $|V|/4$, $|V|/2$, $|V|$ and $2|V|$ seed nodes per n-node, m-edge graphlet (as described in Section 2.3.2), respectively. We performed 10 distinct runs of the algorithm for each choice of the number of seed nodes for each graph. The averages and standard deviations of estimated graphlet frequencies were obtained for the 10 runs for the same graph and the same number of seed nodes; the standard deviations were several orders of magnitude smaller than the corresponding averages.

The resulting pattern of averages of graphlet frequency distribution estimates for the PPI network and the corresponding geometric random model networks is close to the pattern of the exact graphlet frequency distributions for these networks (see Supplementary Figure 21 A–D and Supplementary Table 27). However, this is not the case for the ER-DD and SF model networks (Supplementary Figure 22 E and F and Supplementary Table 27).

This heuristic approach works much better for the PPI and GEO networks than for the ER-DD and SF networks because in PPI and GEO networks the frequencies of different graphlets are much more evenly distributed than in the ER-DD and SF networks. That is, in ER-DD and SF networks, the number of sparse graphlets is several orders of magnitude larger than the number of dense graphlets. Thus, since the algorithm is always trying to sample the same number of n-node, m-edge graphlets, the disproportionality of graphlet counts in ER-DD and SF networks cannot be fully detected by this heuristic algorithm. This is also why the (Kashtan *et al*., 2004) algorithm had to analytically account for similar non-uniform sampling.

Determining the number of required samples is well explored in random sampling from databases (Chaudhuri *et al*., 1998; Flajolet and Martins, 1985; Gibbons, 2001; Olken and Rotem, 1995) and estimating statistics on sampled populations (Bunge and Fitzpatrick, 1993). It has been shown that there does not exist an estimator $\hat{d}$ of the number d of distinct values in a value set V based on random sampling, which can guarantee a reasonably small error with any reasonable probability unless the sample size is very close to the size of the database (Chaudhuri *et al*., 1998). This explains why all known estimators give exceedingly large errors on at least some of the datasets (Olken and Rotem, 1995). Note that we obtained accurate graphlet frequency distribution estimates with as few as $(|V|/8)$ samples per n-node, m-edge graphlet for the yeast high-confidence PPI and the corresponding model networks with around 1000 nodes and 2400 edges. This is a several orders of magnitude smaller number of samples than the 10^5 samples that were required by the (Kashtan *et al*., 2004) algorithm for much smaller *Escherichia coli* transcriptional and *Caenorhabditis elegans* neural networks. We are doing a limited, 5-move search in the neighborhood of a random graphlet rather than selecting a random graphlet as in the Kashtan *et al*. (2004) algorithm; the Kashtan *et al*. (2004) algorithm corrects for non-uniform sampling by calculating probabilities to sample a random graphlet instead. Currently, we are estimating only the graphlet frequencies relative to one another; an analytical 'translation' of the resulting estimate should be easy to determine experimentally and is left for future research.

The average processing times taken by these experiments are presented in Supplementary Tables 29 and 30. They are much smaller than the exhaustive search processing times for PPI and geometric random networks. For example, the ratio of the exhaustive search time, T_E, and the heuristic search time, T_H, for yeast high-confidence PPI network and $(|V|/8)$ seed nodes is $r = (T_E/T_H) \approx 95$ while the distance is low at 46.46 (Supplementary Tables 28 and 30). Similarly, this ratio for the tested geometric random networks is as high as 377 and the distances are low (Supplementary Tables 27–30). However, the processing times of these experiments are much higher for ER-DD and SF networks when compared with the results of the exhaustive searches (Supplementary Tables 24, 25, 29 and 30). This owes to the algorithm's extensive searches for graphlets that are very infrequent, or do not exist at all, in these networks; since only the sparse graphlets are frequent in these networks, this results in much wasted time as most of the graphlets, i.e. all of the denser ones, are infrequent, or non-existent, in these networks. Thus, this approach should not be used for ER-DD and SF networks.

As expected, the processing times increase with increased numbers of samples. However, it is interesting that by taking fewer samples we do not lose accuracy of estimated graphlet frequency distribution patterns for PPI and GEO networks. (The results of the TNP heuristic approach behaved this way as well.) Also, with increased dimensionality and density of PPI networks, the processing time grows as a result of larger local neighborhoods having to be explored (the same is true for the TNP heuristic). Regardless, this approach scales to large networks (see Section 3.2 of Supplementary information).

More details about the NLS heuristic are given in Section 3.2 of Supplementary information.

4 CONCLUSION

We have described two heuristic graphlet frequency estimation approaches that work well for high-confidence PPI and geometric random networks. They do not work well for ER-DD and SF networks both in terms of the resulting estimates and running times. Note that both of these approaches work well for high-confidence PPI networks, which have scale-free degree distributions and contain hubs. They also work well for geometric random networks, which have Poisson degree distributions and lack hubs. Thus, it is not the presence or absence of hubs that dictates the behavior of these heuristics, as was the case in the Kashtan *et al*. (2004) algorithm, but the local structure of the networks. Surprisingly few samples were needed to produce very good estimates of graphlet frequency distribution patterns in PPI and geometric random networks. However, unlike the Kashtan *et al*. (2004) algorithm, for both of our approaches, the processing time grows with the density of the network as a result of larger local neighborhoods having to be explored.

A sample comparison of the TNP and NLS performances for PPI and geometric random networks is presented in Supplementary Table 31. The TNP and NLS experiments with approximately the same number of processed and seed nodes were chosen for the comparison. In this comparison, a slightly larger number of PPI network nodes was processed by the TNP than by the NLS approach (Supplementary Table 31). Also, slightly better distances

were obtained by the TNP than by the NLS approach for the PPI network. In addition, much better running time ratios were obtained by the TNP heuristic despite a larger number of nodes of the PPI network being processed by it than by the NLS approach.

A similar situation was observed for the geometric random model network. For this network, the graphlet frequency distribution estimates obtained by the TNP approach were much better than those obtained by the NLS approach, despite a much smaller number of nodes being processed by the TNP than by the NLS approach (Supplementary Table 31). The running time ratios in these two approaches applied to this model network were comparable, with the NLS approach achieving a slightly better ratio.

From these comparisons it seems that the TNP approach performs better than the NLS approach for estimating graphlet frequency distributions in PPI and geometric random networks. Also, these results indicate that a combined TNP-NLS-based approach may give the best performance: rather than sampling everywhere in the network as we do in the NLS heuristic, we should only sample the fringe. That is, rather than processing nodes on the fringe of a network exhaustively, as we do in the TNP approach, we should sample this part of the network as in the NLS approach. In this way, fast and good estimates of graphlet frequency distributions in PPI and geometric random networks will likely be obtained.

Although we have obtained accurate relative graphlet frequency estimates, more experiments are needed to determine approaches that would 'translate' the estimated graphlet frequency distributions closer to the exact one in absolute values. Also, a more detailed theoretical explanation of the relationship between the structure of the networks and the success of the heuristic approaches would be beneficial. Our results give hope that similar approaches may be used to distinguish between types of networks, or to elucidate the structure-function relationship in PPI networks (Milo *et al*., 2002; Pržulj *et al*., 2004). Implementation of these and the development of other approaches that would efficiently detect larger graphlets is a topic for future research.

ACKNOWLEDGEMENTS

The authors thank R. Mathon, G. Prive, W. Hayes and J. Wrana for helpful comments and discussions. The research was supported by the Natural Sciences and Engineering Research Council of Canada, the Ontario Graduate Scholarship Program and IBM Canada.

Conflict of Interest: none declared.

REFERENCES

Artzy-Randrup,Y. *et al*. (2004) Comment on ''Network motifs: simple building blocks of complex networks'' and ''Superfamilies of evolved and designed networks''. *Science*, **305**, 1107.

Arumuganathan,K. and Earle,E.D. (1991) Nuclear DNA content of some important plant species. *Plant Mol. Biol. Rep*., **9**, 208–218.

Barabási,A.L. and Albert,R. (1999) Emergence of scaling in random networks. *Science*, **286**, 509–512.

Bunge,J. and Fitzpatrick,M. (1993) Estimating the number of species: a review. *J. Am. Stat. Assoc*., **88**, 364–373.

Chaudhuri,S., Motwani,R. and Narasayya,V. (1998) Random sampling for histogram construction: how much is enough? In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data,* Seattle, Washington, June 2–4. 1998, pp. 436–447.

de Aguiar,M.A.M. and Bar-Yam,Y. (2005) Spectral analysis and the dynamic response of complex networks. *Phys. Rev. E. Stat. Nonlin. Soft Matter Phys*., 71.016106.

Duke,R.A. *et al*. (1995) A fast approimation algorithm for computing the frequencies of subgraphs in a given graph. *SIAM J. Comput*., **24**, 598–620.

Dyer,M., Frieze,A. and Jerrum,M. (1994) Approximately counting Hamilton cycles in dense graphs. In *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms,* Society for Industrial and Applied Mathematics. Arlington, Virginia, January 23–25, 1994, pp. 336–343.

Erdös,P. and Rényi,A. (1959) On random graphs. *Publ. Math*., **6**, 290–297.

Erdös,P. and Rényi,A. (1960) On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*., **5**, 17–61.

Flajolet,P. and Martin,G.N. (1985) Probabilistic counting algorithms for data base applications. *Comput. Sys. Sci*., **31**, 182–209.

Garey,M.R. and Johnson,D.S. (1979) *Computers and Intractability—A Guide to the Theory of NP-Completeness*. Freeman, W. H. and Company, New York.

Gavin,A.C. *et al*. (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, **415**, 141–147.

Gibbons,P.B. (2001) Distinct sampling for highly-accurate answers to distinct values queries and event reports. In *Proceedings of the 27th VLDB Conference,* Roma, Italy, September 11–14, 2001, pp. 541–550.

Giot,L. *et al*. (2003) A protein interaction map of *Drosophila melanogaster*. *Science*, **302**, 1727–1736.

Han,J.D.H. *et al*. (2005) Effects of sampling on topology predictions of protein-protein interaction networks. *Nat. Biotechnol*., **23**, 839–844.

Ho,Y. *et al*. (2002) Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry. *Nature*, **415**, 180–183.

IRGSP International Rice Genome Sequencing Project, (2005) The map-based sequence of the rice genome. *Nature*, **436**, 793–800.

Ito,T. *et al*. (2000) Toward a protein–protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc. Natl Acad. Sci. USA*, **97**, 1143–1147.

Itzkovitz,S. *et al*. (2005) Coarse graining and self-dissimilarity of complex networks. *Phys. Rev*., 71.016127.

Jerrum,M. (2003) *Counting, Sampling and Integrating: Algorithms and Complexity*. Springer Verlag.

Kashtan,N. *et al*. (2004) Series: Lecture Notes in Mathematics, ETH Zurich. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, **20**, 1746–1758.

Keller,E.F. (2005) Revisiting ''scale-free'' networks. *BioEssays*, **27**, 11060–11068.

King,A.D. *et al*. (2004) Protein complex prediction via cost-based clustering. *Bioinformatics*, **20**, 3013–3020.

Lappe,M. and Holm,L. (2004) Unraveling protein interaction networks with near-optimal efficiency. *Nat. Biotechnol*., **22**, 98–103.

Li,S. *et al*. (2004) A map of the interactome network of the metazoan *caenorhabditis elegans*. *Science*, **303**, 540–543.

Milo,R. *et al*. (2002) Network motifs: simple building blocks of complex networks. *Science*, **298**, 824–827.

Olkon,F. and Rotem,D. (1995) Random sampling from databases—a survey. *Stat. Comput*., **5**, 25–41.

Pržulj,N. (2005) Analyzing large biological networks: protein–protein interactions example, Ph. D. thesis. University of Toronto, Canada.

Pržulj,N. *et al*. (2004a) Modeling interactome: scale-free or geometric?. *Bioinformatics*, **20**, 3508–3515.

Pržulj,N. *et al*. (2004b) Functional topology in a network of protein interactions. *Bioinformatics*, **20**, 340–348.

Ravasz,E. *et al*. (2002) Hierarchical organization of modularity in metabolic networks. *Science*, **297**, 1551–1555.

Shen-Orr,S.S. *et al*. (2002) Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nat. Genet*., **31**, 64–68.

Song,C. *et al*. (2005) Self-similarity of complex networks. *Nature*, **433**, 392–395.

Stumpf,M.P. *et al*. (2005) Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proc. Natl Acad. Sci. USA*, **102**, 4221–4224.

Tanaka,R. (2005) Scale-rich metabolic networks. *Phys. Rev. Lett*., **94**, 168101.

Tanaka,R. *et al*. (2005) Some protein interaction data do not exhibit power law statistics. *FEBS lett*., **579**, 5140–5144.

Uetz,P. *et al*. (2000) A comprehensive analysis of protein–protein interactions in *Saccharomyces cerevisiae*. *Nature*, **403**, 623–624.

Vazquez,A. *et al*. (2004) The topological relationship between the large-scale attributes and local interaction patterns of complex networks. *Proc. Natl Acad. Sci. USA*, **101**, 17940–17945.

von Mering,C. *et al*. (2002) Comparative assessment of large-scale datasets of protein–protein interactions. *Nature*, **417**, 399–403.