*Bioinformatics*, 33(9), 2017, 1354–1361
doi: 10.1093/bioinformatics/btw733
Advance Access Publication Date: 18 January 2017
Original Paper

Systems biology

# Linearity of network proximity measures: implications for set-based queries and significance testing

**Sean Maxwell[1,*], Mark R. Chance[1,2] and Mehmet Koyutürk[1,3,*]**

[1]Center for Proteomics and Bioinformatics, [2]Department of Nutrition and [3]Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH 44106, USA

*To whom correspondence should be addressed.

Associate Editor: Cenk Sahinalp

Received on July 16, 2016; revised on October 21, 2016; editorial decision on November 14, 2016; accepted on November 17, 2016

## Abstract

**Motivation:** In recent years, various network proximity measures have been proposed to facilitate the use of biomolecular interaction data in a broad range of applications. These applications include functional annotation, disease gene prioritization, comparative analysis of biological systems and prediction of new interactions. In such applications, a major task is the scoring or ranking of the nodes in the network in terms of their proximity to a given set of 'seed' nodes (e.g. a group of proteins that are identified to be associated with a disease, or are deferentially expressed in a certain condition). Many different network proximity measures are utilized for this purpose, and these measures are quite diverse in terms of the benefits they offer.

**Results:** We propose a unifying framework for characterizing network proximity measures for set-based queries. We observe that many existing measures are linear, in that the proximity of a node to a set of nodes can be represented as an aggregation of its proximity to the individual nodes in the set. Based on this observation, we propose methods for processing of set-based proximity queries that take advantage of sparse local proximity information. In addition, we provide an analytical framework for characterizing the distribution of proximity scores based on reference models that accurately capture the characteristics of the seed set (e.g. degree distribution and biological function). The resulting framework facilitates computation of exact figures for the statistical significance of network proximity scores, enabling assessment of the accuracy of Monte Carlo simulation based estimation methods.

**Availability and Implementation:** Implementations of the methods in this paper are available at https://bioengine.case.edu/crosstalker which includes a robust visualization for results viewing.

**Contact:** stm@case.edu or mxk331@case.edu

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

## 1 Introduction

The use of molecular interaction networks for inferring biological function is well-established in systems biology (Legrain and Rain, 2014). In earlier applications, computational methods treated networks as 'bags of pairwise interactions', where inference on individual molecules was based on direct interacting partners of the molecule of interest (Lage *et al.*, 2007). Later developments suggested that incorporating network context drastically improves functional inference. In other words, the paths in molecular interaction networks, as well as their length and multiplicity, provide information on the functional relationships among biomolecules. Motivated by these observations, a number of measures for quantifying the topological relationship of the nodes in a network have been developed (Halldórsson and Sharan, 2013). While the focus of these methods is not limited to quantifying proximity *per se*, we here refer to these measures as *network proximity measures* for brevity.

## 1.1 Network proximity measures and their applications

Common network proximity measures shown to be successful in systems biology applications include random walk with restarts (Macropol *et al.*, 2009; Hofree *et al.*, 2013), network propagation (Vanunu *et al.*, 2010), network diffusion (Vandin *et al.*, 2010), topological similarity (Erten *et al.*, 2011b; Lei and Ruan, 2013) and diffusion state distance (Cao *et al.*, 2013). Applications of these measures include functional annotation of individual biomolecules, prioritization of candidate disease genes (Erten *et al.*, 2011b; Vanunu *et al.*, 2010), network-guided discovery of disease associated processes and pathways (Patel *et al.*, 2013; Zhang *et al.*, 2013) and the interpretation of cancer mutations in a broader context (Hofree *et al.*, 2013; Kim *et al.*, 2011; Vandin *et al.*, 2010). These methods usually simulate a form of information flow in the network and quantify the proximity between two nodes based on the amount of information that flows between the two nodes. Topological similarity and diffusion state take this information one step further and capture the topological relationship between two nodes by comparing them to other nodes in the network (Cao *et al.*, 2013; Erten *et al.*, 2011b).

## 1.2 Quantifying proximity to a set of 'seed' nodes

While network proximity measures are designed to assess the topological relationship between a pair of nodes in a network, their applications reach beyond pairwise relationships. In most applications, researchers are interested in quantifying the topological relationship between sets of nodes, or ranking the nodes in the network based on their proximity to a given 'seed' set of nodes. For example, in the context of candidate disease gene prioritization, the objective is to rank a set of candidate genes based on the proximity of their products to the products of genes implicated in similar diseases (Navlakha and Kingsford, 2010). Similarly, in the context of functional annotation, a specific function is usually associated with multiple proteins (Pritykin *et al.*, 2015). In this case, the potential association of a protein with a given function is assessed based on its proximity to the set of proteins that are associated with the function. Furthermore, in many cases, when researchers identify a set of molecules with altered expression or activity in a certain experiment, an important follow-up question is to identify functional commonalities among these molecules (Nibbe *et al.*, 2009). While enrichment analysis is commonly applied in these applications, some experimental technologies generate information with low coverage, and therefore identification of molecules that are functionally related to the identified set of molecules can be useful (Nibbe *et al.*, 2010).

Consider the example shown in Figure 1. In the figure, an interaction network is shown with seed nodes filled in dark grey. The seed nodes may be proteins or genes identified by experimental or computational methods such as differential RNA/protein expression, differential phosphorylation or other PTMs, frequently mutated genes in a specific condition, or genes containing SNPs with potential associations. In Figure 1A, we observe that seeds are uniformly distributed throughout the network in general, but are dense in the area expanded in Figure 1B. Under the guilt-by-association principle in interaction networks, the seeds in and around the dense area of Figure 1B are likely functionally related, whereas the seeds spread throughout the network are unrelated or not known to be related. A proximity measure such as Random Walk with Restarts (RWR) can identify the genes/proteins in the dense area of Figure 1B because walks restarting and proceeding in the dense region spend a significant amount of time traversing the proteins in this area. Thus, the RWR score for these genes/proteins is higher than other areas in which there is a lower density of the seeds. A caveat of this approach is highlighted in Figure 1C., in which some seeds are in proximity to high degree nodes frequently because high degree nodes are in proximity to a large portion of the network. This inflates the RWR scores of high degree nodes compared to other nodes even when no functional association exits. Additional steps can be taken to normalize scores to account for this bias, such as using Monte-Carlo simulations with random seed sets to adjust the scores for each gene/protein by its score distribution.

## 1.3 Linearity of network proximity measures

The focus of this study is on quantifying the network proximity between each node in the network and a given 'seed set' of nodes. We investigate a common property of network proximity measures, namely linearity. We observe that multiple network proximity measures are linear in that the proximity to a set of nodes (seed set) can be written as a linear combination of proximity to the individual nodes in the seed set. We argue that linearity can be exploited for indexing and efficient computation of network proximity for set-based queries, as well as analytical characterization of the distribution of network proximity. We also generalize *betweenness centrality* to be constrained to a set of nodes such that it exhibits a similar property, in that betweenness centrality with respect to a set of

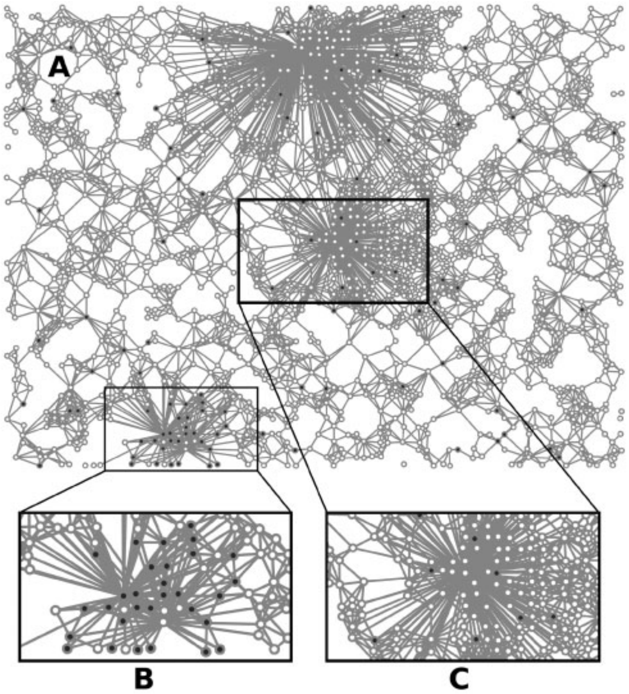

**Fig. 1. (A)** An example interaction network where nodes represent molecules such as genes or gene products and edges represent interactions such as protein–protein interactions or co-expression. Dark nodes represent 'seed' nodes to be analyzed in the context of the interaction network, e.g. the query aims to find additional molecules that are functionally related to the molecules in the seed set. **(B)** Functionally related seeds appear near one another in the interaction network under the 'guilt-by-association' principle. Non-seed nodes in this area rank high because they are in proximity to many seeds. **(C)** Seeds randomly distributed across the network appear around high degree 'hub' nodes by chance more than they appear around low degree nodes. It is important to correct for this trend statistically to avoid scoring hub nodes high when they are in proximity to a large portion of molecules in the network, and thus likely to score high by chance

nodes (seed set) can be written as a linear combination of the betweenness centrality with respect to pairs of nodes in the seed set.

### 1.4 Applications of linearity

We discuss two potential applications of linearity: (i) efficient computation of set-based queries using sparse indexes constructed from proximity profiles of individual nodes, (ii) precise characterization of the distribution of network proximity scores. First, we discuss how linearity can be exploited to index 'proximity to nodes' *a priori* and efficiently process queries that involve proximity to node sets. Subsequently, we provide a general framework for computing the mean and standard deviation of proximity of a given node to a subset of nodes with given properties. The proposed framework applies to any measure that is linear, and can be useful in accurately characterizing the significance of proximity to a set of nodes.

In various studies, it has been shown that network proximity measures can be biased by such factors as node degree (Erten *et al.*, 2011b), and statistical correction based on a background distribution of these scores can improve the predictive ability of these measures (Nibbe *et al.*, 2010). However, in systems biology, the common practice is to estimate the statistics of these distributions using Monte-Carlo simulations (also called permutation tests) (Guo *et al.*, 2015; Krämer *et al.*, 2014; Garcia-Alonso *et al.*, 2012; Ideker *et al.*, 2002). The framework we present here provides the ability to construct reference models that take into account important characteristics of the network and the seed set, such as the degree distribution and the types or functional background of molecules in the seed set. It also provides a theoretically grounded method for exact characterization of the distributions of network proximity based on these flexible reference models. In this study, we also use these analytical results to assess the accuracy of Monte Carlo simulations in capturing the basic characteristics of the distribution of network proximity scores.

### 1.5 Experimental results

In our experiments, we use random walk with restarts (RWR) as the benchmark proximity measure and perform systematic experiments to assess whether node-based indexing can improve efficiency of computing RWR-based proximity to a 'seed set' of nodes in protein-protein interaction (PPI) networks. Since the proposed sparse indexing scheme stores partial information, we also assess the accuracy of proximity scores computed via sparse indexing. Our results show that sparse indexing drastically improves the efficiency of computing set-based network proximity queries without compromising accuracy. We also perform systematic experiments to assess the accuracy of Monte Carlo simulations in estimating the mean and variance of RWR-based network proximity. Our results suggest that the choice of the number of simulations has a significant effect on the accuracy of figures computed via Monte Carlo simulations. Specifically, we observe that estimates of mean and variance based on a small number of simulations diverge significantly from actual figures; however, Monte Carlo simulations produce robust estimates when a sufficient number of simulations is used.

### 1.6 Organization of the paper

In the next section, we introduce the notion of linearity of network proximity measures. Subsequently, we prove that several network proximity measures are linear. We then discuss how linearity can be exploited to index node-based proximity for efficient computation of proximity to seed sets. Subsequently, we present our framework for characterizing the distribution of network proximity scores. In Section 3, we report the findings from our computational experiments. We conclude our discussion and outline avenues for future research in Section 4.

## 2 Materials and Methods

In this section, we define linearity of network proximity measures and show that commonly used network proximity measures such as random walk with restarts, network propagation, effective importance, network diffusion and betweenness centrality satisfy this property. We then discuss how this property can be used for indexing and efficient computation of network proximity scores for sets of nodes, and to analytically characterize their distribution.

### 2.1 Linearity of network proximity measures

Let $G=(V,E)$ denote a graph with a set of vertices $V$ and a set of edges $E \subseteq V \times V$. Let $S \subseteq V$ be a subset of vertices in this graph.

Let $\mathbf{h}$ denote a $|V|$-dimensional vector function that represents the 'network proximity' between the vertices in $S$ and any other vertex in the graph. For example, $\mathbf{h}$ can be the network proximity between $S$ and any other vertex in the graph based on random walk with restarts, computed by setting the restart vector to have non-zero elements in the entries that correspond to vertices in $S$.

The primary motivation for this work is the observation that a class of common network-proximity measures can be written as a linear combination of the proximities to the vertices or pairs of vertices in the set $S$. Namely, we say a *network proximity measure* $\mathbf{h}_S$ is *linear* if it can be written as follows:

$$\mathbf{h}_S = \sum_{v \in S} \mathbf{f}_S(v)\mathbf{h}_v \tag{1}$$

Here, $\mathbf{f}_S$ weights the score vector each $v$ contributes to the solution (e.g. $\mathbf{f}_S$ assigns different restart probabilities or confidence levels to each $v \in S$). In many cases the entries in $\mathbf{f}_S$ are all identical. In this case, we use the scalar form $f_S$ outside the summation to simplify notation.

To further generalize this notion, we also define linearity over pairs of nodes in $S$. Namely, we say a *network proximity measure* $\mathbf{p}_S$ is *pairwise linear* if it can be written as follows:

$$\mathbf{p}_S = \sum_{s,t \in S \times S} \mathbf{f}_S(st)\mathbf{p}_{\{st\}} \tag{2}$$

In other words, a network proximity measure $\mathbf{p}_S$ is pairwise linear if it can be written as a weighted sum of the vectors $\mathbf{p}_{\{st\}}$ over all pairs of vertices in $S$.

Next, we show that multiple network proximity measures are linear, i.e. the measures satisfy Equation (1). Subsequently, we discuss how a common network centrality measure, betweenness centrality, can be formulated as a proximity measure that is pairwise linear.

#### 2.1.1 Linearity of random walk with restarts

For a given graph $G=(V,E)$ represented by adjacency matrix $\mathcal{W}$ and a subset $S \subseteq V$ of vertices, random walk with restarts based proximity is defined as:

$$\mathbf{h}_S^{(rw)} = (1-r)\mathcal{W}'\mathbf{h}_S^{(rw)} + r\mathbf{e}_S, \tag{3}$$

where $\mathbf{e}_S$ is an $n$-dimensional restart vector with $\mathbf{e}_S(v) = 1/|S|$ if $v \in S$, $\mathbf{e}_S(v) = 0$ otherwise, $r$ is the restart probability parameter, and

$\mathcal{W}'$ is the stochastic matrix derived from the adjacency matrix $\mathcal{W}$. That is, the rows of $\mathcal{W}'$ are normalized by vertex degree such that $\mathcal{W}'_{i,j} = \mathcal{W}_{i,j}/\sum_k \mathcal{W}_{i,k}$.

THEOREM 1. Random walk with restart based network proximity, $\mathbf{h}_S^{(rw)}$, is a linear network proximity measure.

**PROOF.** Note that random walk with restart based proximity to a single vertex $v \in V$, $\mathbf{h}_v^{(rw)}$, is defined as in Equation (3) with restart vector $\mathbf{e}_v$. Now observe that we can write Equation (3) as:

$$\mathbf{h}_S^{(rw)} - (1-r)\mathcal{W}'\mathbf{h}_S^{(rw)} = r\mathbf{e}_S \rightarrow (I-(1-r)\mathcal{W}')\mathbf{h}_S^{(rw)} = \mathbf{e}_S$$

and hence the solution to the linear system is given by

$$\mathbf{h}_S^{(rw)} = r(I-(1-r)\mathcal{W}\prime)^{-1}\mathbf{e}_S$$

$\mathbf{h}_v^{(rw)}$ can be similarly rearranged. Consequently, letting $X = r(I-(1-r)\mathcal{W}\prime)^{-1}$, we have $\mathbf{h}_S^{(rw)} = X\mathbf{e}_S$ and $\mathbf{h}_v^{(rw)} = X\mathbf{e}_v$. Furthermore, since $\mathbf{e}_S = \frac{1}{|S|}\sum_{v\in S}\mathbf{e}_v$, we obtain:

$$\mathbf{h}_S^{(rw)} = X\mathbf{e}_S = X\frac{1}{|S|}\sum_{v\in S}\mathbf{e}_v = \frac{1}{|S|}\sum_{v\in S}X\mathbf{e}_v = \frac{1}{|S|}\sum_{v\in S}\mathbf{h}_v^{(rw)}.$$

Therefore, $\mathbf{h}_S^{(rw)}$ is linear with $f_S = 1/|S|$.

### 2.1.2 Linearity of network propagation

Another commonly used network proximity measure is network propagation (Vanunu *et al.*, 2010), defined as:

$$\mathbf{h}_S^{(np)} = (1-r)\mathcal{W}''\mathbf{h}_S^{(np)} + r\mathbf{y}_S \tag{4}$$

where $\mathbf{y}_S$ is an $n$-dimensional vector similar to $\mathbf{e}_S$ representing prior knowledge of disease associations of nodes in $S$, $r$ is a parameter that balances prior knowledge with propagated information and $\mathcal{W}''$ is derived from the adjacency matrix $\mathcal{W}$. Namely, the entries of $\mathcal{W}''$ are the entries of $\mathcal{W}$ normalized by the geometric mean of the degrees of the edge end-points, i.e. $\mathcal{W}''_{i,j} = \mathcal{W}_{i,j}/\sqrt{D(i,i)D(j,j)}$ where $D$ is a diagonal matrix such that $D(i,i) = \sum_j \mathcal{W}_{i,j}$.

$\mathbf{y}_S(v) = 1$ for $v \in S$, and 0 otherwise, so $\mathbf{y}_S = \sum_{v\in S}\mathbf{y}_v$. The proof for the linearity of $\mathbf{h}_S^{(rw)}$ immediately generalizes to $\mathbf{h}_S^{(np)}$ because the formulations of the two methods are identical with the exception that they use different methods to normalize the adjacency matrix. Therefore, $\mathbf{h}_S^{(np)}$ is linear with $f_S = 1$.

### 2.1.3 Linearity of effective importance

Effective importance (Bogdanov and Singh, 2013) is defined as:

$$\mathbf{h}_S^{(ei)}(v) = \mathbf{h}_S^{(rw)}(v)/d(v) \tag{5}$$

where $d(v)$ is the degree of vertex $v$ or the total edge weight of $v$. This is simply $\mathbf{h}_S^{(rw)}$ with a vector weight $\mathbf{f}_S$, i.e. $\mathbf{f}_S(v) = 1/(|S|d(v))$. Therefore, $\mathbf{h}_S^{(ei)}$ is $\mathbf{h}_S^{(rw)}$ multiplied by a constant, so $\mathbf{h}_S^{(ei)}$ is linear.

### 2.1.4 Linearity of network diffusion

Network diffusion (Qi *et al.*, 2008) of a set $S$ is defined as:

$$\mathbf{h}_S^{(nd)} = G\mathbf{b}_S \tag{6}$$

Where $\mathbf{b}_S(v) = 1$ if $v \in S$ and 0 otherwise, and $G$ is a constant matrix derived from a column normalized adjacency matrix and a diffusion parameter $\gamma$ similar to $r$ in $\mathbf{h}_S^{(rw)}$. Because $\mathbf{b}_S = \sum_{v\in S}\mathbf{b}_v$, it immediately follows that $\mathbf{h}_S^{(nd)} = \sum_{v\in S}G\mathbf{b}_v$. Therefore, $\mathbf{h}_S^{(nd)}$ is linear with $f_S = 1$.

### 2.1.5 Linearity of betweenness centrality

Betweenness centrality is a measure of the network centrality of a node, which is based on the number of shortest paths that go through the node. In a graph $G = (V, E)$, the betweenness centrality $C_B(v)$ of a vertex $v \in V$ is usually defined as follows:

$$C_B(v) = \sum_{s,t\in V'\times V'}\frac{\sigma_{st}(v)}{\sigma_{st}}. \tag{7}$$

Here, $V' = V\backslash\{v\}$, $\sigma_{st}$ denotes the total number of shortest paths connecting $s$ and $t$, and $\sigma_{st}(v)$ denotes the number of shortest paths connecting $s$ and $t$ that pass through $v$.

While betweenness centrality is not a measure of network proximity, it can be formulated to assess the 'betweenness' of a node with respect to a subset of vertices, thereby providing an alternate measure of being interconnected to a subset of vertices. For this purpose, we define a vector function similar to $C_B$ that operates on a vertex pair and is defined for all $v \in V$:

$$\mathbf{p}_{\{st\}}(v) = \begin{cases} 0 & \text{if } v \in \{s,t\} \\ \dfrac{\sigma_{st}(v)}{\sigma_{st}} & \text{otherwise} \end{cases} \tag{8}$$

We can now define $\mathbf{p}_S^{(bc)}$ for a subset of vertices $S$, which measures the proportion of shortest paths each $v \in V$ appears on between all pairs $s, t \in S$, i.e.:

$$\mathbf{p}_S^{(bc)} = \sum_{s,t\in S\times S}\mathbf{p}_{\{st\}} \tag{9}$$

Observe that $\mathbf{p}_S^{(bc)}$ is not linear (i.e. it does not satisfy Equation (1)), but it is pairwise linear by definition [i.e. it satisfies Equation (2) with $f_S = 1$]. Note also that $\mathbf{p}_V^{(bc)}(v) = C_B(v)$.

## 2.2 Applications of linearity

### 2.2.1 Indexing and efficient computation

If a network proximity measure is linear (or pairwise linear), the score of a node with respect to a set $S$ of nodes can be computed from the scores of individual elements of $S$. For this reason, for a given query 'seed set' $S$, the proximity of other nodes in the network to $S$ can be computed more efficiently than by using standard methods. For this purpose, $\mathbf{h}_v$ for all nodes in the network or $\mathbf{p}_{\{st\}}$ for each pair of nodes in the network can be pre-computed, or 'indexed', and used later in Equations 1 or 2 to compute the solution for any $S$.

The time required to build the index depends on the complexity of the proximity measure. For example, for random walk with restarts, the proximity vector for each node can be computed via iterative sparse matrix-vector multiplications in $\Theta(t|E|)$ time, where $t$ denotes the number of iterations until convergence. Thus the index can be built in $\Theta(t|V||E|)$. Note that index construction is highly parallelizable since the computation is performed independently for each node. The space requirement of this index is $\Theta(|V|^2)$.

Once the index is available, $\mathbf{h}_S^{(rw)}, \mathbf{h}_S^{(np)}, \mathbf{h}_S^{(ei)}$ and $\mathbf{h}_S^{(nd)}$ can be computed with the index in $\Theta(|S||V|)$ time, where we usually have $|S| \ll |V|$. On the other hand, for $\mathbf{h}_S^{(rw)}, \mathbf{h}_S^{(np)}$ and $\mathbf{h}_S^{(ei)}$ the computation without the index requires $\Theta(t|E|)$ time. The definition of $\mathbf{h}_S^{(nd)}$ is the same as our index formulation because $G$ is constant, so it has no alternative runtime.

Similarly, $\mathbf{p}_S^{(bc)}$ can be computed $\Theta(|S|^2|V|)$ with the use of an index. The most efficient algorithm for computing betweenness centrality in sparse graphs is Brandes' algorithm (Brandes, 2001), which requires $O(|V||E|)$ time. While this algorithm can be modified to take advantage of indexing as well, brute force computation with indexing would also outperform Brandes algorithm when $|S|^2 = o(|E|)$, which is often the case since seed sets are rather small.

### 2.2.2 Sparse indexing

If set-based proximity queries are repeatedly processed in an application, then the indexing scheme described in the previous section can be useful. However, the storage requirement of the index can be limiting for very large networks, since the number of values that needs to be stored is quadratic in the number of nodes. To alleviate this problem, the index can rather be constructed using local search methods that identify the closest $K$ neighbors (with respect to the proximity measure) of each node $v$, and the corresponding $K$ proximity values can be used to build a 'sparse' index. Local search methods leverage the fact that nodes distant from $v$ have low proximity scores with negligible local effects that can be treated as zero (Wu *et al.*, 2014). At sparsity level $K$, each row of the index contains only the $K$ closest neighbors of each node in the network, and a method such as K-dash is used (Fujiwara *et al.*, 2012) to compute the full sparse index in time $O(|V|^2 + |V||E|)$. Once the sparse index is constructed, the proximity vector for a given seed set $S$ can be computed by performing the summation in Equations 1 or 2 for only those entries that are present in the index. This computation can be performed in $\Theta(|S||K|)$ time, and storage of the index requires only $\Theta(|V||K|)$ time. Similar strategies can be applied to indexes for pairwise linear measures, which would be significantly sparse for measures such as betweenness centrality.

The sparseness of the index balances accuracy with space complexity, where a full index computes exact solutions and a sparse index computes an approximate solution. For this reason, in the 'Results' section, we compare runtimes and solutions from a sparse index method to iterative and analytic methods to assess the tradeoff between the savings in storage and accuracy.

### 2.2.3 Characteristic distributions of $\mathbf{h}_S$ and $\mathbf{p}_S$

In computational biology, an important consideration is the distribution of scores generated by scoring functions, since this distribution influences the interpretation of scores. In the context of network proximity to a subset of nodes, multiple studies have shown that correction of node scores based on a background distribution improves the accuracy of predictions that are based on these scores (Erten *et al.*, 2011a; Nibbe *et al.*, 2010; Krämer *et al.*, 2014). Furthermore, while characterizing the distribution of network proximity scores, the reference model needs to accurately capture background information. Since molecular interaction networks are characteristic in their topological properties, including their degree distribution, a background model that takes into account these topological properties is desirable. Similarly, while assessing proximity to a subset of nodes, it is desirable to take into account the properties of the nodes in the subset of interest, e.g. if the subset is composed of phosphoproteins, then an accurate reference model can be 'proximity to a fixed-sized subset of phosphoproteins', rather than 'proximity to a fixed-sized subset of nodes'. The common practice today is to characterize the distribution of network proximity scores using permutation tests (Erten *et al.*, 2011a; Krämer *et al.*, 2014; Nibbe *et al.*, 2010).

Here, we show that linearity of network proximity measures can be exploited to derive exact forms for mean and variance of network proximity scores. The approach we propose allows the incorporation of a broad range of reference models. Namely, we assume that the network is fixed and the set of query nodes $S$ is variable. We are interested in the distribution of $\mathbf{h}_S$ based on a reference model that generates $S$. The properties of the nodes in $S$ are determined in the reference model; for example, the degree distribution of the nodes in $S$ is known or the functional classification of some of the nodes in $S$ is known. Assume that there are $m$ such properties. We represent this reference model by partitioning $V$ into $m$ subsets of nodes, $\mathcal{Q}_1, \mathcal{Q}_2, \ldots, \mathcal{Q}_m$, where $\mathcal{Q}_i$ represents the subset of nodes in the network possessing property $i$, and $\mathcal{Q}_i \cap \mathcal{Q}_j = \emptyset$ for $i \neq j$. Note, the union of all $\mathcal{Q}_i$ is not required to be $V$. Since the set $S$ is given, we know the distribution of the nodes in $S$ to bins $\mathcal{Q}$, i.e. $\mathcal{L}_i = S \cap \mathcal{Q}_i$. In this setting, the number of nodes in the network that possess property $i$ is given by $|\mathcal{Q}_i|$, whereas the number of nodes in $S$ that possess the same property is given by $|\mathcal{L}_i|$.

The purpose of the partitioning-based model is to accurately capture the 'background' characteristics of the seed set, thereby providing an accurate reference model for the hypothesis being tested. Specifically, the null hypothesis here is as follows: the individual proximity of each node to the seed set stems from a specific characteristic (e.g. the degrees of the nodes) of the seed set, as opposed to the specific biological process(es) represented by the seed set. This partitioning based model allows representation of different reference models by choosing different criteria for partitioning. Based on this reference model, the linearity property defined in Equation (1) can be used to derive an expression for the expected value of network proximity scores for all nodes.

Theorem 2. Let $G(V, E)$ be a network, $\mathbf{h}$ be a linear network proximity measure defined on this network, and $\mathcal{Q}$ and $\mathcal{L}$ denote a reference model for a query set $S$. For a subset T of size $|S|$ that is generated by uniformly sampling $|\mathcal{L}_i|$ nodes from $\mathcal{Q}_i$ for $1 \leq i \leq m$, the expected value of $\mathbf{h}_T$ is given by:

$$E(\mathbf{h}_T) = f_S \sum_{i=1}^{m} \sum_{v \in \mathcal{Q}_i} \frac{|\mathcal{L}_i|}{|\mathcal{Q}_i|} \mathbf{h}_v \qquad (10)$$

Due to space considerations, the proof of Theorem 1 is provided in the Supplementary Material. The main idea behind the proof of this theorem is the observation that the proximity to each vertex in the graph contributes to the expected value in proportion to the relative number of vertices that share the same property. By counting the number of subsets that contain a given number of vertices with a given property, the contribution of each vertex to the expected value can be precisely computed. Using Equation (10), the expected value of proximity to a set with given properties can be computed in $O(|V|^2)$ time for linear proximity measures.

Using the reference model specified above, we also derive the variance of the proximity scores for a seed set with given properties.

Theorem 3. For the reference model specified in Theorem 1, the expected value of $\mathbf{h}_T^2$ is given by the following expression:

$$\begin{aligned} E(\mathbf{h}_T^2) = \; & f_S^2 \sum_{i=1}^{m} \left( \frac{|\mathcal{L}_i|}{|\mathcal{Q}_i|} \sum_{v \in \mathcal{Q}_i} \mathbf{h}(v)^2 \right) \\ & + 2 f_S^2 \sum_{|\mathcal{Q}_i| > 1} \left( \frac{|\mathcal{L}_i|(|\mathcal{L}_i| - 1)}{|\mathcal{Q}_i|(|\mathcal{Q}_i| - 1)} \sum_{u,v \in \mathcal{Q}_i} \mathbf{h}_u \mathbf{h}_v \right) \qquad (11) \\ & + 2 f_S^2 \sum_{\mathcal{Q}_i \neq \mathcal{Q}_j} \left( \frac{|\mathcal{L}_i||\mathcal{L}_j|}{|\mathcal{Q}_i||\mathcal{Q}_j|} \sum_{u \in \mathcal{Q}_i, v \in \mathcal{Q}_j} \mathbf{h}_u \mathbf{h}_v \right) \end{aligned}$$

The proof of this theorem is provided in the Supplementary Material. This proof follows the lines of the proof for the previous theorem. Using Equation (11), the variance of proximity to a set with given properties can be computed in $O(|V|^3)$ time for linear proximity measures.Finally, using the same reference model, we derive an expression for the expected value of a pairwise linear proximity measure.

THEOREM 4. For the reference model specified in Theorem 1, the expected value of $\mathbf{p}_T$ is given by the following expression:

$$E(\mathbf{p}_T) = f_S \sum_{|\mathcal{Q}_i|>1} \left( \frac{|\mathcal{L}_i|(|\mathcal{L}_i|-1)}{|\mathcal{Q}_i|(|\mathcal{Q}_i|-1)} \sum_{u,v\in\mathcal{Q}_i} \mathbf{p}_{uv} \right) + \sum_{\mathcal{Q}_i \neq \mathcal{Q}_j} \left( \frac{|\mathcal{L}_i||\mathcal{L}_j|}{|\mathcal{Q}_i||\mathcal{Q}_j|} \sum_{u\in\mathcal{Q}_i, v\in\mathcal{Q}_j} \mathbf{p}_{uv} \right) \tag{12}$$

The proof of this theorem, which is based on a generalization of Theorem 1, is also provided in the Supplementary Material. Using this result, the expected value of proximity to a set with given properties can be computed in $O(|V|^3)$ time for pairwise linear proximity measures.An important requirement of our reference model is that the nodes of the network be partitioned to the bins of $\mathcal{Q}$. For example, each $v \in V$ appears in zero or one $\mathcal{Q}_i$. It is not straightforward to generalize the proofs for the preceding analytical forms for $E(X)$ and $E(X^2)$ to the case when bins overlap, since sampling of $S$ will result in dependencies between different bins (a node that is assigned to multiple bins cannot be selected more than once). It is possible to bypass this problem by constructing bins for each intersection. The problem with this approach, however, is that it will greatly increase the number of bins, limiting the descriptiveness of the reference model for a given seed set. Therefore, the generalization of our analytical results to the case with overlapping bins is an open problem of interest.

## 3 Results

### 3.1 Datasets

For the input networks, we use the STRING (Szklarczyk *et al.*, 2015) and BioGRID (Chatr-aryamontri *et al.*, 2015) networks. We include only high confidence(confidence score $\geq 0.7$), human interactions from STRING, which results in a network of 15 524 nodes and 320 462 edges. BioGRID is more sparse than STRING and contains limited interaction confidence information, so we include all human interactions, which results in a network of 14 638 nodes and 144 708 interactions.

### 3.2 Experimental setup

We perform all experiments using random walk with restarts (RWR) as the network proximity measure with the restart probability $r = 0.5$. We sample random seed sets $S$ of sizes 20, 50, 100, 500 and 1000 from the set of all proteins $V$ of each network. For each $S$, we analytically compute the mean and standard deviation of the distribution of scores for a random seed set that mirrors the degree distribution of $S$ using the equations in Section 2.2.3 and a full index. We also estimate these figures using Monte Carlo simulations based on $10^k$ random instances, where $k$ ranges from 2 to 4, for both an iterative method and sparse indexes. We vary the sparsity of the indexes to include the closest $10^j$ neighbors, where $j$ ranges from 1 to 3, but this is a soft limit that includes all nodes with the $j$th highest proximity score. This process is replicated 50 times for each size $|S|$ and the results are averaged to generalize the overall response. Figures apply only to the STRING network. The BioGRID results exhibit the same trends and are available in the Supplementary Material.

### 3.3 Runtime with sparse indexing

We first evaluate the differences in runtime between the index methods and conventional iterative methods for computing random walk scores. We include the runtime for a full index along with sparse indexes here for comparison and the results are shown in Figure 2. As seen in the figure, the index methods are more stable and efficient in terms of computation time, and runtime decreases at higher levels of sparsity.

### 3.4 Accuracy in the estimation of statistics

In Figure 3, the 1-norm of the difference between the analytic distribution vectors and estimated vectors, normalized by the 1-norm of the analytic vectors, are shown respectively for mean and standard deviation. These figures assess the deviation of sampling from the correct statistics as a fraction of the correct statistics. The mean and standard deviation vectors estimated by the simulation methods are relatively close to the true analytically computed values when a sufficient number of simulations are used. The vectors computed from indexes containing fewer than $10^3$ nearest nodes begin to deviate significantly from the other methods.

### 3.5 Ranking of nodes based on proximity

Next, we investigate how estimation of mean and standard deviation affects ranking of nodes. For each seed set $S$, we compute adjusted proximity scores for each node as $(\mathbf{h}_S(v) - \mu_v)/\sigma_v$, where $\mu_v$ and $\sigma_v$ respectively denote mean and standard deviation of the proximity of $v$ to a seed set with 'similar' degree distribution to the input seed set $S$. For example, sets with similar degree distribution to the input set $S$ can be generated by binning nodes from the network to the seed node with the smallest absolute difference in degree out of all $v \in S$, and then sampling 1 node randomly from each bin to generate sets of the same size as $S$ (Erten *et al.*, 2011a; Nibbe *et al.*, 2010).

We use our analytical method and estimation methods to compute this mean and standard deviation, obtain adjusted vertex scores

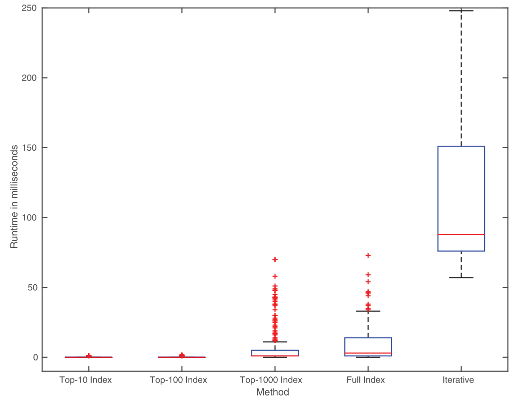

**Fig. 2.** Runtime comparison of index methods to iterative methods for computing random walk with restarts. The full index includes proximity of each node to all other nodes in the network, while the Top-*K* sparse indexes include the proximity of each node to the closest *K* nodes. Runtimes for all sizes $|S|$ are included for each plot (Color version of this figure is available at *Bioinformatics* online.)

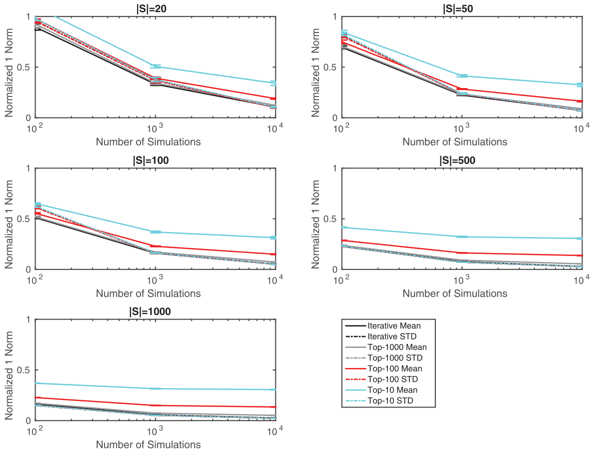

**Fig. 3.** The normalized 1-norm of the mean and standard deviation vectors computed analytically versus vectors estimated by iterative and sparse index simulation methods. Each data point is the average over 50 randomly drawn inputs *S* (Color version of this figure is available at *Bioinformatics* online.)

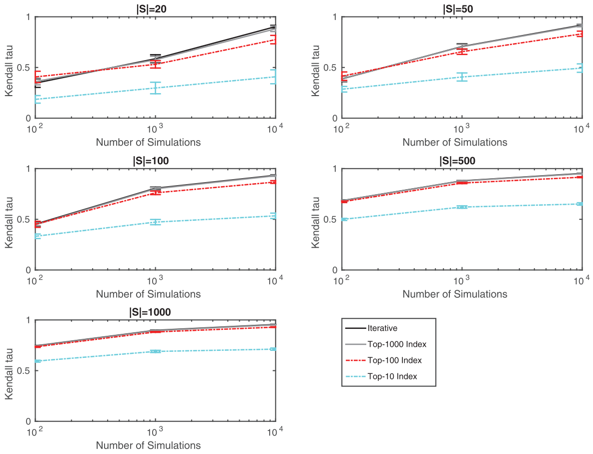

**Fig. 4.** The Kendall rank correlation of the node rankings computed by the analytic method versus Monte Carlo simulations of the sparse index and iterative methods. Each data point is the average over 50 randomly drawn inputs *S* (Color version of this figure is available at *Bioinformatics* online.)

**Table 1.** Accuracy of the iterative and sparse index methods when classifying vertices as significant/non-significant at $P < 0.001$ significance level

| | Iterative | | Top-1000 | | Top-100 | | Top-10 | |
|---|---|---|---|---|---|---|---|---|
| \|S\| | Prec | Rec | Prec | Rec | Prec | Rec | Prec | Rec |
| 20 | 0.97 | 0.98 | 0.97 | 0.97 | 0.95 | 0.95 | 0.92 | 0.72 |
| 50 | 0.98 | 0.99 | 0.98 | 0.98 | 0.97 | 0.98 | 0.96 | 0.94 |
| 100 | 0.99 | 1.00 | 0.99 | 0.99 | 0.99 | 1.00 | 0.98 | 0.99 |
| 500 | 0.99 | 1.00 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 | 0.99 |
| 1000 | 0.97 | 0.96 | 0.97 | 0.96 | 0.97 | 0.96 | 0.96 | 0.96 |

The distributions are estimated using $10^4$ Monte-Carlo simulations.

for each method and use the adjusted scores to rank the nodes according to their adjusted proximity to the seed set. Note if $\sigma_v = 0$, which occurs in the Monte Carlo methods primarily when using low numbers of simulations, the adjusted score takes an undefined or infinite value. We then determine the concordance in the rankings obtained by the simulation methods versus our analytic method using the Kendall rank correlation coefficient $\tau$, which is the number of concordantly ranked pairs minus the number of discordantly ranked pairs divided by the total number of ranked pairs (Kendall, 1938). $\tau$ values range from $-1$ to 1 where values closer to 1 indicate greater agreement between the estimation method and analytic method rankings. We observe that the estimation methods rank the nodes very similarly to the analytic method when $10^4$ Monte Carlo simulations are used. We also observe that the ranking becomes less accurate for sparser indexes, but is equivalent to the iterative method when $10^3$ nearest nodes are used (Fig. 4).

### 3.6 Identification of statistically significant nodes

Finally, we explore how variation in the ranking of nodes by simulation methods affects classification of statistically significant nodes. We classify nodes as significant or non-significant by our analytic solution, where significant nodes have adjusted scores > 3.45, corresponding to a *P*-value < 0.001 under the assumption that the adjusted scores are normally distributed (Nibbe *et al.*, 2010). We then compute the recall and precision with which the estimation methods classify nodes as significant/non-significant using the same significance criteria. In Table 1, we list the precision and recall of the iterative and sparse index estimation methods respectively, when using $10^4$ Monte-Carlo simulations. The sparse index method performs nearly equivalent to the iterative method across all sizes $|S|$ for the index of $10^3$ nearest nodes. The effects of index sparseness on accuracy are most apparent for the index of 10 nearest neighbors, which shows significantly lower recall than the other methods for $|S| = 20$.

## 4 Discussion and future work

### 4.1 Sparsity and simulation accuracy

Our computational experiments suggest that $10^4$ Monte-Carlo simulations are adequate to ensure accurate score distribution estimates for protein and gene interaction networks, in which the number of nodes and edges are respectively less than $2 \times 10^4$ and $4 \times 10^5$. In our experiments, sparse computations were accurate for indexes of the 10 nearest neighbors of each node when the seed set size was over 100, but more neighbors were required for smaller seed sets. While generalizing these results to other networks, it would be best practice to identify the response for different sparsity levels with the analytical methods, particularly if a network has significantly different characteristics from our test networks (e.g. the network is not scale free or is significantly larger).

### 4.2 Relation to non-linear measures

It is important to note that the results presented in this paper are limited to network proximity measures that satisfy our definition of linearity(i.e. proximity to a set is a weighted sum of proximity to each element of the set). There exist proximity measures that are not linear by our definition (e.g. average topological similarity; Erten *et al.*, 2011b) and such measures can be more useful than linear measures in various contexts. It is not straightforward to extend the theoretical framework presented here to non-linear measures, but the results presented here may provide a stepping stone toward deriving exact analytical solutions for non-linear measures as well.

### 4.3 Biological relevance

In this work, we focus on the numerical accuracy of the significance figures computed by simulation studies. An interesting follow-up question is the impact of accurate assessment of statistical significance on the biological relevance of results. While it has been already shown that adjustment of random walk scores based on node degree improves the biological relevance of results in specific contexts (Nibbe *et al.*, 2010; Erten *et al.*, 2011b), comprehensive analysis that involves functional annotations can lead to further insights.

## 5 Conclusion

We have proposed a common framework for linear network proximity measures that facilitates set-based queries from sparse local search proximity data, as well as exact figures for the mean and standard deviation of proximity scores relative to a query set. Using our analytic methods, we show that sparse local search data can be used to compute proximity to a query set with little impact on accuracy compared to iterative methods. In addition, our analytic methods are tractable and can be used as the basis for deterministic algorithms that utilize proximity score distributions.

## Funding

Clinical and Translational Science Collaborative of Cleveland, UL1TR000439 from the National Center for Advancing Translational Sciences (NCATS) component of the National Institutes of Health and NIH roadmap for Medical Research; National Institutes of Health (NIH) grants R01-LM011247 and U01-CA198941.

*Conflict of Interest*: none declared.

## References

Bogdanov,P. and Singh,A. (2013). Accurate and scalable nearest neighbors in large networks based on effective importance. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management, CIKM '13*, pp. 1009–1018. ACM, New York, NY.

Brandes,U. (2001) A faster algorithm for betweenness centrality. *J. Math. Sociol.*, **25**, 163–177.

Cao,M. *et al.* (2013). Going the Distance for Protein Function Prediction: A New Distance Metric for Protein Interaction Networks. *PLoS One*, **8**, e76339.

Chatr-Aryamontri,A. *et al.* (2015) The biogrid interaction database: 2015 update. *Nucleic Acids Res.*, **43**, 470–478.

Erten,S. *et al.* (2011a) Dada: degree-aware algorithms for network-based disease gene prioritization. *BioData Min.*, **4**, 1–20.

Erten,S. *et al.* (2011b) Vavien: an algorithm for prioritizing candidate disease genes based on topological similarity of proteins in interaction networks. *J. Comput. Biol.*, **18**, 1561–1574.

Fujiwara,Y. *et al.* (2012) Fast and exact top-k search for random walk with restart. *Proc. VLDB Endow.*, **5**, 442–453.

Garcia-Alonso,L. *et al.* (2012) Discovering the hidden sub-network component in a ranked list of genes or proteins derived from genomic experiments. *Nucleic Acids Res.*, **40**, e158.

Guo,H. *et al.* (2015) Biased random walk model for the prioritization of drug resistance associated proteins. *Sci. Rep.*, **5**:10857

Halldórsson,B.V. and Sharan,R. (2013) Network-based interpretation of genomic variation data. *J. Mol. Biol.*, **425**, 3964–3969. [Understanding Molecular Effects of Naturally Occurring Genetic Differences].

Hofree,M. *et al.* (2013) Network-based stratification of tumor mutations. *Nat Methods*, **10**, 1108–1115.

Ideker,T. *et al.* (2002) Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, **18(suppl 1)**, S233–S240.

Kendall,M. G. (1938) A new measure of rank correlation. *Biometrika*, **30**, 81–93.

Kim,Y.A. *et al.* (2011) Identifying causal genes and dysregulated pathways in complex diseases. *PLoS Comput. Biol.*, **7**, e1001095.

Krämer,A. *et al.* (2014) Causal analysis approaches in ingenuity pathway analysis. *Bioinformatics*, **30**, 523–530.

Lage,K. *et al.* (2007) A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nat. Biotechnol.*, **25**, 309–316.

Legrain,P. and Rain,J.C.C. (2014) Twenty years of protein interaction studies for biological function deciphering. *J. Proteomics*, **107**, 93–97

Lei,C. and Ruan,J. (2013) A novel link prediction algorithm for reconstructing protein–protein interaction networks by topological similarity. *Bioinformatics*, **29**, 355–364.

Macropol,K. *et al.* (2009) Rrw: repeated random walks on genome-scale protein networks for local cluster discovery. *BMC Bioinformatics*, **10**, 283.

Navlakha,S. and Kingsford,C. (2010) The power of protein interaction networks for associating genes with diseases. *Bioinformatics*, **26**, 1057–1063.

Nibbe,R. *et al.* (2009) Discovery and scoring of protein interaction subnetworks discriminative of late stage human colon cancer. *Mol. Cell Proteomics*, **8**, 827–845.

Nibbe,R.K. *et al.* (2010) An integrative -omics approach to identify functional sub-networks in human colorectal cancer. *PLOS Comput. Biol.*, **6**, e1000639.

Patel,V. *et al.* (2013) Network signatures of survival in glioblastoma multiforme. *PLOS Comput Biol.*, **9**, e1003237.

Pritykin,Y. *et al.* (2015) Genome-wide detection and analysis of multifunctional genes. *PLoS Comput. Biol.*, **11**, e1004467 .

Qi,Y. *et al.* (2008) Finding friends and enemies in an enemies-only network: A graph diffusion kernel for predicting novel genetic interactions and co-complex membership from yeast genetic interactions. *Genome Res.*, **18**, 1991–2004.

Szklarczyk,D. *et al.* (2015) String v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic Acids Res.*, **43**, 447–452.

Vandin,F. *et al.* (2010). Algorithms for detecting significantly mutated pathways in cancer. In: Berger, B. (ed.), *Research in Computational Molecular Biology: 14th Annual International Conference, RECOMB 2010, Lisbon, Portugal, April 25-28, 2010*, pp. 506–521. Springer, Berlin, Heidelberg.

Vanunu,O. *et al.* (2010) Associating genes and protein complexes with disease via network propagation. *PLOS Comput. Biol.*, **6**, e1000641.

Wu,Y. *et al.* (2014). Fast and unified local search for random walk based k-nearest-neighbor query in large graphs. In: *Proceedings of the 2014 ACM SIGMOD international conference on Management of Data*, pp. 1139–1150. ACM, New York.

Zhang,W. *et al.* (2013) Network-based survival analysis reveals subnetwork signatures for predicting outcomes of ovarian cancer treatment. *PLoS Comput. Biol.*, **9**, e1002975.