An empirical comparison of connectivity-based distances on a graph and their computational scalability

Pierre Miasnikof*[1,2], Alexander Y. Shestopaloff[3,4], Leonidas Pitsoulis[5], and Alexander Ponomarenko[6,7]

[1]Dept. of Electrical & Computer Engineering, University of Toronto, Canada
[2]The University of Toronto Data Sciences Institute (DSI), Canada
[3]School of Mathematical Sciences, Queen Mary University of London, United Kingdom
[4]The Alan Turing Institute, United Kingdom
[5]Dept. of Electrical & Computer Engineering, Aristotle University of Thessaloniki, Greece
[6]Laboratory of Algorithms and Technologies for Networks Analysis, National Research University Higher School of Economics, Nizhny Novgorod, Russian Federation
[7]Institute of Applied Physics of The Russian Academy of Sciences, Nizhny Novgorod, Russian Federation

Abstract

In this study, we compare distance measures with respect to their ability to capture vertex community structure and the scalability of their computation. Our goal is to find a distance measure which can be used in an aggregate pairwise minimization clustering scheme. The minimization should lead to subsets of vertices with high induced subgraph density. Our definition of distance is rooted in the notion that vertices sharing more connections are closer to each other than vertices which share fewer connections. This definition differs from that of the geodesic distance typically used in graphs. It is based on neighborhood overlap, not shortest path. We compare four

*corresponding author: p.miasnikof@mail.utoronto.ca

distance measures from the literature and evaluate their accuracy in reflecting intra-cluster density, when aggregated (averaged) at the cluster level. Our tests are conducted on synthetic graphs, where clusters and intra-cluster densities are known in advance. We find that amplified commute, Otsuka-Ochiai and Jaccard distances display a consistent inverse relation to intra-cluster density. We also conclude that the computation of amplified commute distance does not scale as well to large graphs as that of the other two distances.

1 Introduction

The goal of this work is to find a suitable distance measure, to be used in clustering (community detection). Given a graph $G(V,E)$, we seek a transformation of a graph's adjacency matrix into a symmetric $|V| \times |V|$ distance matrix whose elements are the pairwise distances between vertices, $d_{ij}\,(\geq 0, d_{ii} = 0)$. This transformation allows us to cluster vertices using distance minimization (similarity maximization) techniques from the literature and obtain densely connected clusters. The quadratic clustering problem presented by Fan and Pardalos [17, 16] and Fan et al. [18] and the recent quadratic unconstrained binary optimization (QUBO) K-medoids technique of Bauckhage et al. [5] are examples of such clustering techniques.

We compare various node to node distances and assess their suitability as a quantity to be minimized for the purpose of graph clustering. We seek a quantity whose aggregate pairwise minimization through a graph clustering algorithm will yield dense clusters. When averaged over a cluster, our chosen distance should display a consistent inverse relation to intra-cluster density. It is important to note that our definition of distance differs from that of the usual shortest path distance. It measures the similarity in the neighborhoods of a pair of vertices, not the length of the shortest path between them.

Clustering is an unsupervised learning task which consists of grouping elements deemed somehow similar. Although a formal definition of graph clustering remains an unsettled matter, there is broad consensus that clusters (communities) should form densely connected subsets of vertices (e.g., [46, 19, 44, 45]). Consequently, graph clustering algorithms aim to group nodes into dense subsets. In this context, distance (similarity) is measured by shared connections. Nodes sharing many neighbors are considered more similar to each other than nodes which share fewer neighbors. In the case of a representative clustering, the resulting subsets of vertices (clusters) form dense induced subgraphs. These clusters should form subsets of vertices that exhibit a high level of interconnection and a lower level of connection to vertices in the rest of the graph. In terms of distance, this density pattern should translate into shorter node to node distances within clusters than between clusters, on average.

Ease of computation is a very valuable feature for any distance used in graph clustering. Most graph clustering formulations are known to be NP-hard. For example, binary optimization-based graph clustering formulations fall into this category of difficult problems [25]. In order for clustering to remain feasible, computing distances should not impose additional onerous costs. For this reason, we focus our examination on three distances whose computation does not require the entire adjacency or Laplacian matrix.

However, because of its importance in graph theory (e.g., [10]) and to highlight the computational challenges of all distances requiring matrix-based computations, we also examine a recent correction of commute (resistance) distance known as amplified commute distance [48, 43]. Nevertheless, we call the reader's attention to the fact that its computation requires matrix inversion. It cannot be computed on a pairwise basis. In the context of the massive data sets being analyzed today, most of which do not fit in the memories of even the most advanced computers, this requirement is a major impediment to its use in practical applications.

This article expands the scope of our previous work on the topic of vertex to vertex distance [39]. It includes a broader set of experiments and an examination of weighted graphs, amplified commute distance, robustness to noise, statistical interpretations and computational issues. It follows the groundwork of Fouss et al. [22], Kivimäki et al. [32] and Sommer et al. [47], who all compared various distances.

The remainder of this article is organized as follows. First, we end this section by illustrating the need for vertex to vertex distances. Then, after a brief overview of the literature, we present the distances under consideration. Next, we examine their relationship to intra-cluster density. To perform this examination, we use synthetic graphs where cluster membership and intra-cluster densities are known, by construction. Finally, to illustrate and validate our distances, we compute each of them for the vertex pairs of two well-know benchmark graph data sets, the Zachary karate club graph [51] and the United States college football graph [24].

Throughout this document, we only consider connectundirected graphs with no self-loops, which are either unweighted or weighted with non-negative edge weights. We denote the set of clusters by $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$. Individual clusters are identified by the subscript $k \in \{1, \ldots, K\}$. The number of nodes in an arbitrary cluster k is represented by n_k and the total number of nodes on the graph by $N = \sum_k n_k = |V|$. The set of edges connecting two nodes in this same arbitrary cluster is denoted as e_k. Finally, graph density is denoted as $\mathcal{K}$ (capital kappa),

$$\mathcal{K} = \frac{|E|}{0.5 \times N \times (N-1)} \, .$$

Similarly, for an arbitrary cluster k, intra cluster density is denoted by $\mathcal{K}_{\text{intra}}^{(k)}$,

$$\mathcal{K}_{\text{intra}}^{(k)} = \frac{|e_k|}{0.5 \times n_k \times (n_k - 1)} .$$

In the unweighted graph case, these densities can be interpreted as an empirical estimate of edge probability. The graph's overall density is the estimate of the unconditional probability that two arbitrary nodes are joined by an edge. Intra-cluster density is the estimate of the conditional probability that two nodes in a given cluster k are connected. In weighted graphs, these quantities become mean edge weights.

1.1 Motivation

In his article, Chebotarev [10] identified the need for a dissimilarity measure in data analysis applications. More recently, Granata et al. [27] also made a similar observation. Graph clustering, sometimes referred to as community detection, is one such application. To illustrate this need, we examine two optimization-based graph clustering formulations. Both formulations require a distance whose pairwise minimization will lead to densely connected clusters, for all vertex pairs.

As mentioned earlier, it is widely understood that clusters are subsets of vertices that form dense induced subgraphs with sparser connections to the remaining vertices (e.g., [46, 19, 21, 44]). In accordance with this broadly accepted understanding, an intuitive optimization-based clustering formulation consists of assigning vertices into K (non-overlapping) clusters such that intra-cluster density is maximized. This problem formulation consists of maximizing the objective function f_o shown below.

$$\begin{aligned} f_o &= \sum_{k=1}^{K} \frac{|e_k|}{0.5 \times n_k(n_k - 1)} \\ &= \sum_{k=1}^{K} \frac{\sum_i \sum_j x_{ik} x_{jk} w_{ij}}{0.5 \times \left(\sum_i x_{ik}\right)\left(\left(\sum_i x_{ik}\right) - 1\right)} , \end{aligned}$$

where,

- $x_{ik} \in \{0, 1\}$: the (binary) decision variable which takes the value of 1 if node i is assigned to cluster k (0 otherwise) and
- w_{ij}: the weight of the edge joining nodes i and j.

Unfortunately, this formulation is fractional. Worse, the objective function f_o is degenerate and may not be defined in many cases (e.g., empty clusters, single node clusters), making its optimization impossible. Fortunately, some alternative formulations exist. For example, Fan and Pardalos [17, 16] and Fan et al. [18] present a formulation that is based on a binary quadratic distance minimization. Their objective function (f_o) is optimized by assigning vertices that are more similar or separated by shorter distances (d_{ij}) to the same clusters,

$$
\begin{aligned}
f_o &= \sum_{i=1}^{N} \sum_{j=i+1}^{N} \sum_{k=1}^{K} d_{ij} x_{ik} x_{jk} \\
& \qquad x_{ik} \in \{0,1\}, \quad \sum_{k=1}^{K} x_{ik} = 1, \quad \forall i \in V \,.
\end{aligned}
$$

Another viable option is offered by the recently introduced quadratic K-medoids formulation [5, 29],

$$
\begin{aligned}
\underset{z}{min} \quad & \left\{ f_o = \beta z^T \Delta \mathbf{1} - \alpha \frac{1}{2} z^T \Delta z + \gamma \left(z^T \mathbf{1} - K \right)^2 \right\} \\
& z_i \in \{0,1\}, \quad \forall i \in V \,,
\end{aligned}
$$

where,

- α, β, γ: are the (input) trade-off parameters,
- $z \, (\in \mathbb{R}^N)$ is the vector of decision variables,
- $z_i \in \{0,1\}$: are the (binary) decision variables which take the value of 1 if node i is selected as an exemplar (0 otherwise),
- $\mathbf{1} \, (\in \mathbb{R}^N)$ is a vector of ones and
- Δ is an $N \times N$ matrix of node-node distances.

Unlike the binary quadratic distance minimization formulation, the K-medoids algorithm does not directly assign nodes to clusters. Instead, the algorithm identifies K exemplars around which the clusters will be built.

While discussing K-medoids, in its original or QUBO formulations, it is important to note that this algorithm was not initially designed for graphs. Indeed, it requires some metric space, in which distances between all covariates (nodes) is available. It is through the work discussed here that this clustering technique can be applied to graphs. Similarly, the binary quadratic distance minimization also requires distances between all node pairs. While that formulation was presented in the context of graphs, it assumes the availability of all-pairs distances.

The topic of graph clustering algorithms is beyond the scope of this article. We only describe two optmization-based techniques to illustrate the need for a quantity whose minimization will lead to dense clusters. We also want to call the reader's attention to two distinguishing features of these formulations. First, they are or can be cast as QUBO problems [25]. The QUBO formulation allows the use of newly available purpose-built computational architectures which help circumvent the NP-hardness of the clustering problem and of binary optimization in general. Through the use of these novel architectures, we obtain very good heuristic solutions with only short computation times [23, 46, 19, 36, 3, 29].

Another very appealing feature of these distance-based formulations is that they do not depend on modularity and its well-documented shortcomings (e.g., [20, 1, 26, 31, 40, 41]), unlike the very popular Louvain technique [6]. In fact, recent work has demonstrated these distance-based optimization techniques to be superior to the Louvain method [38].

2 Previous Work

Typically, when distance is discussed in the context of graphs and complex networks, it refers to shortest paths between nodes (e.g., [14, 10, 4, 2]). Another approach in the literature consists of borrowing distance measures used in Euclidian space, like the Euclidian or cosine distances (e.g., [46, 19]) for example. Unfortunately, none of these interpretations is well-suited to clustering. They fail to capture the similarity in the connectivity of vertices and community structure or the discrete nature of graphs.

On the topic of shortest path distance, Akara-pipattana et al. [2] stated the following: *"While intuitive and visual, this notion of distance is limited in that it does not fully capture the ease or difficulty of reaching point j from point i by navigating the graph edges. It does not say whether there is only one path of minimal length or many such paths, whether these paths can be straightforwardly located, or whether alternative paths are considerably or only slightly longer."* A few years earlier, in the course of their distance comparisons, Fouss et al. [22] also made a similar statement. Several years earlier, while presenting their theorems, Chebotarev and Shamis also highlighted the unsuitability of shortest-path distance as a similarity measure [13]. We agree with these authors' statements and also illustrate them through a concrete example, in Section 3.

In the case of Euclidean space distance measures, like cosine or Euclidean distances, we must note that graphs are not geometric objects. In fact, the "fundamental problem of mapping graphs to vectors" remains a topic of discussion in the current literature (e.g., [35]). This representation mismatch between graphs and distances raises questions of interpretation. Of course, we can always treat rows (or columns) of a graph's adjacency or Laplacian matrix as vectors representing vertices, but proceeding in such a manner distorts the meaning and definitions of these measures. However, Burt's [8] and Otsuka-Ochiai [42] distances, which we explore in the article, can respectively be interpreted as set-theoretic versions of Euclidean and cosine distances.

In an effort to consider all paths between two vertices, Chebotarev and Shamis [13, 12, 11] introduced the "matrix-forest" theorems and "forest-accessibility" distance. Later, Chebotarev [10] established the link between a logarithmic transformation of the "forest-accessibility", shortest path and resistance distances. Unfortunately, computing forest-accessibility distance requires matrix inversions, a very costly operation which can quickly become infeasible, when data sets grow too large.

Several authors (e.g., [50, 37]) have also introduced and made use of a hybrid distance that combines both shortest path and random walk distance, known as "randomized shortest path" (RSP). In fact, Kivimäki et al. [32] not only reported good clustering results with the use of RSP, they also offered an efficient computation technique for it. As well, these same authors found "free energy distance" (FE) to offer a good reflection of graph structure. Sommer et al. [47] compared a total of five distances, including RSP and FE, in the context of graph clustering, and came to similar conclusions. In this discussion of RSP and FE, it must be highlighted that to obtain these quantities, it is also necessary to perform matrix inversions.

Another approach to vertex distances and graph geometry more generally, hyperbolic network geometry, was proposed by Krioukov et al. [33] and also discussed more recently by Boguñá et al. [7]. It consists of embedding graphs into hyperbolic space. Although it allows a latent-space view of the graph, this method is not immediately or easily applicable without significant further work.

Yet another distance measure was introduced by Estrada [15]. The author begins by introducing the concept of "communicability" between nodes, a concept rooted in random walks. Its computation requires the eigendecomposition of the adjacency matrix. He then goes on to derive "communicability distance", a metric.

The common challenge posed by these alternatives to shortest path distance is the difficulty of applying them to larger graphs. As mentioned earlier, to obtain RSP and FE, it is necessary to perform matrix inversions. To obtain communicability, it is necessary to obtain the eigendecomposition of the adjacency matrix. Neither of these matrix operations are easily scaled to large data sets. In today's context of "big data", very large-scale graphs are not just hypothetical (e.g., [34]), which makes computational overhead a fundamental topic. A more detailed discussion of computation is included in Section 5.

Meanwhile, Fouss et al. [22] also compared several distances in the context of semi-supervised learning and recommenders. While they reported good results with regularized commute-time, Markov diffusion and regularized Laplacian kernels, computing any of these distances also requires several matrix-based operations.

Fortunately, some long-standing techniques borrowed from other areas of science seem to capture vertex similarity and seem better suited to large data sets due to their lower computational overhead. Burt's distance [8], borrowed from sociology and Jaccard distance [30], borrowed from botany, have both been suggested as valid measures of vertex similarity [46, 19, 9]. Otsuka-Ochiai distance [42], borrowed from zoology, can be understood as a set-theoretic analogue of cosine distance. The operations involved in its computation are similar to those of Jaccard. A more in depth description of these distances and their computation is provided in Sections 4 and 5.

Finally, commute (resistance) distance also seems to capture community structure. In fact, this distance, with roots in both probability theory and electrical engineering, is ubiquitous in graph theory (e.g., [10]). However, it should be noted that commute distance becomes inaccurate in the case of large graphs, which is why some authors have suggested amplified commute distance instead [48, 49, 43]. Unfortunately, its computation, as in the cases of RSP, FE, communicability and the kernels compared by Fouss et al. [22], is not well-suited to large graphs. We include its examination in our experiments because of its importance in graph theory and to highlight the computational advantage of the other distances we study.

3 Defining Distance

Clusters are sets of items which can be considered similar in some way. This similarity is represented by shorter distances. In the case of graphs, this similarity is captured by the number of shared neighbors. Consequently, in such a model, vertices sharing a large number of neighbors should also be separated by smaller distances than vertices sharing fewer connections. Accordingly, distance measures similarity (overlap) in neighborhoods, not shortest path distance. For example, two adjacent vertices with high degrees that do not share any neighbors have a shortest path distance of one, but are dissimilar on the basis of their connectivity. At the cluster level, shorter mean intra-cluster distances are indicative of more densely connected vertices.

Figure 1 shows an instance of a graph that is arguably composed of two clusters, one containing vertices $v_1, \ldots, v_4$ (in blue) and one containing vertices v_5, v_6, v_7 (in red). We observe that each cluster forms a dense induced subgraph. Most notably, we also see that the shortest-path distance separating vertices v_1 and v_3 (i.e., $d(1,3) = 2$, two blue edges) is greater than the distance separating v_1 and v_5

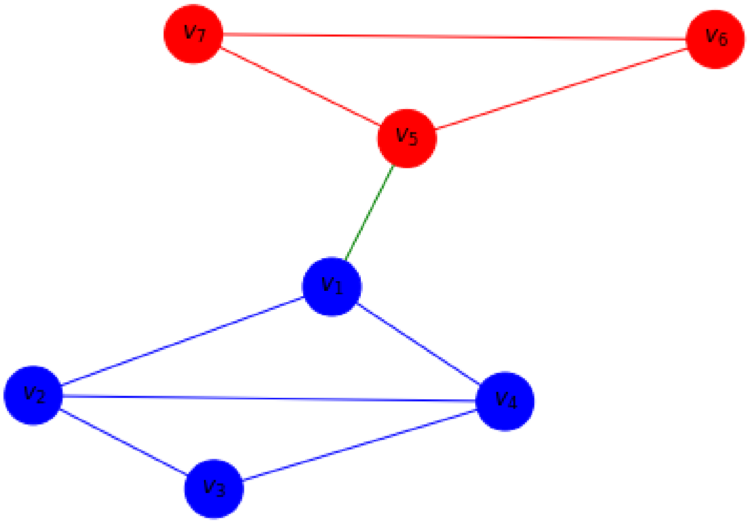


Figure 1: Graph with two clusters

(i.e., $d(1,5) = 1$, single green edge). Nevertheless, in the context of clustering, we argue that v_3 is closer, more similar, to v_1 than to v_5. As mentioned earlier, this similarity or closeness is measured by the number of shared neighbors. Vertices v_1 and v_3 share two neighbors, while vertices v_1 and v_5 have no common neighbors.

4 Distance Measurements Under Study

We compare four different distance measurements from the literature and examine how consistently they reflect connectivity patterns. Under our definition of distance, mean node to node distance within a cluster should consistently vary in step with intra-cluster density, but move in an opposite direction. Densely connected clusters should display low mean node to node distances.

We then examine the relationship between mean Jaccard [30], Otsuka-Ochiai [42], Burt's [8, 19] and amplified commute [48] distances, on one hand, and intra-cluster density, on the other. Because these distances are pairwise measures, we compare their mean value for a given cluster to the cluster's internal density.

4.1 Burt's Distance

Burt's distance (Burt) between two vertices i and j, denoted as b_{ij}, is computed using the adjacency matrix (A):

$$b_{ij} = \sqrt{\sum_{h \neq i,j} (A_{ih} - A_{jh})^2}\,.$$

At the cluster level, we denote the mean Burt distance as $\mathcal{B}$. For an arbitrary cluster k with n_k vertices, it is expressed as

$$\mathcal{B}_k = \frac{1}{0.5 \times n_k \times (n_k - 1)} \sum_{i,j=i+1} b_{ij}\,.$$

4.2 Jaccard Distance

For unweighted graphs, the Jaccard distance (Jacc) separating two vertices i and j is defined as

$$\zeta_{ij} = 1 - \underbrace{\frac{|a_i \cap a_j|}{|a_i \cup a_j|}}_{s_{ij}} \in [0, 1]\,.$$

Here, $a_i\,(a_j)$ represents the set of all vertices with which vertex $i\,(j)$ shares an edge. The ratio s_{ij} is the well known Jaccard similarity. The Jaccard distance (ζ_{ij}) is its complement.

In the weighted case, the numerator $|a_i \cap a_j|$ is replaced by

$$\sum_{h=1}^{N} \min\{e_{ih}, e_{jh}\}\,.$$

The denominator is replaced by

$$\sum_{h=1}^{N} \max\{e_{ih}, e_{jh}\}\,.$$

For both quantities, N denotes the total number of vertices ($N = |E|$) and e_{ih} the weight of the edge between nodes i and h.

At the cluster level, we compute the mean distance separating all pairs of vertices within the cluster, which we denote as $\mathcal{J}$. Here too, for an arbitrary cluster 'k' with n_k vertices, we have

$$\mathcal{J}_k = \frac{1}{0.5 \times n_k \times (n_k - 1)} \sum_{i,j=i+1} \zeta_{ij}\,.$$

4.2.1 Probabilistic and Statistical Interpretation of the Jaccard Distance

We begin by noting that Jaccard distance has a clear intuitive interpretation. It is the complement of Jaccard similarity (s_{ij}). In the context of graph node pairs, the quantity s_{ij} is the amount of shared connections expressed as a fraction of the total number of connections of both nodes forming the pair.

For unweighted graphs, the expected value of the ratio s_{ij} can also be understood to be a function of edge probabilities. The quantity ζ_{ij} is the complement of this expected value. In the weighed case (non-negative weights), the ratio s_{ij} represents the proportion of edge weight (degree) that connects shared neighbors. Again, the quantity ζ_{ij} is the complement of this proportion.

Von Luxburg et al. [48] identified the breakdown in commute distance that occurs in the case of larger graphs. They identified the problem of loss of interpretability and even noted its occurrence in graphs with as little as 1,000 nodes. Fortunately, Jaccard distance does not suffer from this limitation, although it can also be affected by graph sizes in some cases. Empirically, we observe that it remains interpretable with much larger graphs. The ones in our tests have 3,400 vertices.

Large graph behavior of the Jaccard distance can also be analyzed more closely, by assuming a very simple probabilistic model, without loss of generality. Indeed, the phenomena we observe under this simple model would also hold under more complex edge-generation models as well. For clarity, we focus on the Jaccard similarity portion (s_{ij}) and unweighted graphs. Our conclusions apply equally to weighted graphs.

Let's begin with a recall of the following definition:

$$\zeta_{ij} = 1 - \underbrace{\frac{|a_i \cap a_j|}{|a_i \cup a_j|}}_{s_{ij}} .$$

We then define the following model:

- Let $\mathbf{1}_{ik}$ be an indicator function that takes the value of 1 if i is connected to a node k ($\neq i$), among the remaining $N-1$ vertices ($\mathbf{1}_{ik} = 0$, if $i = k$, by definition);

- Here again, $N = |V|$ denotes the total number of nodes in the graph.

Under this model we can express the expected value of node-node similarity, $E\left(s_{ij}\right)$, as

$$\begin{aligned} E\left(s_{ij}\right) &= E\left(\frac{|a_i \cap a_j|}{|a_i \cup a_j|}\right) \\ &= E\left(\frac{\sum_k \left(\mathbf{1}_{ik} \times \mathbf{1}_{jk}\right)}{\sum_k \left(\mathbf{1}_{ik} + \mathbf{1}_{jk} - \mathbf{1}_{ik} \times \mathbf{1}_{jk}\right)}\right) . \end{aligned}$$

As a verification, we observe that under the $G(n,p)$ Erdős-Rényi model all distances are equal in expectation, $E\left(\mathbf{1}_{ik}\right) = E\left(\mathbf{1}_{jk}\right) = p,\ \forall i, k \neq i, \forall j, k \neq j$. Under this model, edge probabilities are all equal to the fixed generative model edge probability parameter, p. So we obtain uniform similarities (and distances) between all node pairs:

$$E\left(s_{ij}\right) \approx \frac{p}{2-p} \Rightarrow E\left(\zeta_{ij}\right) \approx 1 - \frac{p}{2-p} \quad \forall i, j\,.$$

This modeling exercise highlights a scale-based effect in Jaccard distance. We observe that, in cases where $E(\mathbf{1}_{ik}) >> E(\mathbf{1}_{jk})\,(\forall i, k \neq i, \forall j, k \neq j)$, the denominator of the ratio s_{ij} increases at a much higher rate than the numerator, pushing the pairwise similarity s_{ij} to zero and, consequently, distance ζ_{ij} to one. While this convergence to one is completely intuitive and reflective of disparate connectivity patterns, it is amplified by graph sizes. Fortunately, however, nodes belonging to the same cluster are expected to have $E(\mathbf{1}_{ik}) \approx E(\mathbf{1}_{jk})\,(\forall i, k \neq i, \forall j, k \neq j)$. In summary, it should be noted that pairwise distances are not comparable across graphs, they are idiosyncratic to the graph to which they belong. Also, very large distances, even within the same graph, may be difficult to interpret and compare.

4.3 Otsuka-Ochiai Distance

Otsuka-Ochiai distance (OtOc) can be interpreted as the extension of cosine distance to the discrete case. In the case of unweighted graphs, the distance separating two vertices i and j is defined as (the ratio's numerator is the same as the Jaccard distance's)

$$o_{ij} = 1 - \frac{|a_i \cap a_j|}{\sqrt{|a_i| \times |a_j|}} \in [0,1]\,.$$

In the weighted case, the numerator is modified in the same way as in the case of the Jaccard distance. In the denominator, the quantities $|a_i|$ are replaced by the weighted degree:

$$|a_i| = \sum_{h=1}^{N} w_{ih}\,.$$

Here too, we obtain a cluster level measure of similarity by taking the mean over each pair of nodes within a cluster. We denote this mean as $\mathcal{O}$. Again, for an arbitrary cluster 'k' with n_k vertices, we have

$$\mathcal{O}_k = \frac{1}{0.5 \times n_k \times (n_k - 1)} \sum_{i,j=i+1} o_{ij} \, .$$

4.4 Amplified Commute Distance

Von Luxburg et al. [48] describe commute distance between two vertices as *"the expected time it takes a random walk to travel from the first to the second vertex and back"*. Unfortunately, it has been shown to be inaccurate in the case of larger graphs [48, 49]. Amplified commute distance is a correction that seeks to address its known shortcomings [48, 43].

The amplified commute distance between two vertices i and j, denoted as a_{ij}, is computed as

$$\begin{aligned} a_{ij} &= S_{ij} + u_{ij} \\ \text{where,} & \\ S_{ij} &= R_{ij} - \frac{1}{d_i} - \frac{1}{d_j} \, , \\ R_{ij} &= \Gamma^{+}_{ii}\Gamma^{+}_{jj} - 2\Gamma^{+}_{ij} \, , \\ \Gamma &= L + \frac{1}{|V|}\mathbf{1} \\ u_{ij} &= \frac{2w_{ij}}{d_i d_j} - \frac{w_{ii}}{d_i^2} - \frac{w_{jj}}{d_j^2} \, . \end{aligned}$$

In the equations above,

- d_i is the degree of vertex i,
- Γ^{+} is the Moore-Penrose inverse of the matrix Γ,
- Γ^{+}_{ij} is the element at the intersection of the i-th row and j-th column in the Moore-Penrose inverse of Γ,
- L is the graph's Laplacian matrix,
- $\mathbf{1}$ is a $|V| \times |V|$ matrix of ones and
- w_{ij} is the weight of the edge connecting nodes i and j ($\in \{0, 1\}$ in the unweighted case).

The cluster-level figure is denoted by $\mathcal{A}$. As with the other distances, for an arbitrary cluster k with n_k vertices, it is computed as

$$\mathcal{A}_k = \frac{1}{0.5 \times n_k \times (n_k - 1)} \sum_{i,j=i+1} a_{ij} \, .$$

4.4.1 Other Interpretations of the (Amplified) Commute Distance

As mentioned earlier, amplified commute distance is a recently introduced correction to the well-known commute distance. Commute distance has roots in Markovian processes and the study of electrical flows. It is the sum of two hitting times, $H_{ij} + H_{ji}$. This sum corresponds to the expected time for a random walk to travel from vertex i to vertex j (H_{ij}) and return to its starting point i (H_{ji}).

Commute distance is also equal (to a constant) to the resistance distance between two nodes. It can be expressed as $\mathtt{C}_{ij} = vol(G)\Omega_{ij}$, where Ω_{ij} represents the resistance distance between nodes i and j and $vol(G)$ is the volume of the graph containing these nodes.

Amplified commute distance is a first-order correction. It reduces the dominance of the local effect of node degrees on the pairwise distance and provides a more macroscopic measure. Indeed, von Luxburg et al. [48] have shown that, in larger graphs ($|V| > \sim 1,000$), commute distance $\mathtt{C}_{ij}$ is dominated by each nodes' degree.

4.4.2 Observed Degeneracies

In our initial experiments with amplified commute distance, we noted instances of counter-intuitive results. It appears amplified commute distance breaks down in at least one case, the case of path graphs.

5 Computational Complexity

As mentioned earlier, our comparison is also based on an examination of the computational cost of each of the distances under study. Typically, graphs of interest tend to be composed of a very large number of vertices. In addition, most graph clustering formulations are known to be NP-hard. For these reasons, if we hope to use a vertex to vertex distance in conjunction with an already costly distance minimization routine, it must be easy to compute. In our analysis, we make the simplifying assumptions that all arithmetic operations have constant (unit) cost and look-ups have a cost of zero. Given that we are only interested in comparing time complexity as a function of graph size (number of vertices), these assumptions are justified.

The most important element of this assessment is that all four of our distances have the same asymptotic (big-O) upper bound of $O(N^3)$. Although in theory computing any of these distances falls into the same complexity class, their true computational costs vary greatly.

In the cases of Burt, Otsuka-Ochiai and Jaccard distances all computations are pairwise computations that can be parallelized or processed in increments. They do not require the storage of the entire adjacency or Laplacian matrix in memory or any costly computations, like inversion or eigendecomposition. Distances between each pair of vertices are computed independently by performing elementary arithmetic on the elements of a pair of rows (columns) of the adjacency matrix. This computation can be further broken down by using only portions of these rows (columns) and proceeding iteratively until the entirety of the rows (columns) has been processed. To speed computation this entire process can also very easily be parallelized.

In contrast, the bulk of the cost in computing amplified commute distance does not come from pair-wise computations but from computations requiring the entire graph (inverting Γ). This computational challenge is akin to the costs of computing RSP, FE or communicability distance.

5.1 Burt's Distance

For each vertex pair, we must compute $N-2$ subtractions, $N-2$ additions, take the power of two and then the square root. Under our assumptions, the total computational cost is $2(N-2)+2 = 2N-2$.

At the graph level, we have $0.5 \times N\,(N-1)$ distances to compute. Therefore, the total computational cost to convert an adjacency matrix into an all-pairs Burt's distance matrix is $O(N^3)$:

$$\frac{1}{2}N(N-1)(2N-2) = N^3 - 4N^2 + 2N\,.$$

5.2 Jaccard Distance

The numerator of the Jaccard distance, $|a_i \cap a_j|$ or $\sum_{h=1}^{N} \min\{e_{ih}, e_{jh}\}$ in the weighted case requires N comparisons and N additions, for a total of $2N$ operations. Similarly, the denominator, $|a_i \cup a_j|$ or $\sum_k \max\{e_{ih}, e_{jh}\}$ requires N comparison and N additions. Combining those two computations, taking their ratio and subtracting it from 1, we obtain a total cost of $2N + 2N + 2 = 4N + 2$ for each pair. At the graph level, the total computational cost to convert an adjacency matrix into an all-pairs Jaccard distance matrix is also $O(N^3)$, since

$$\frac{1}{2}N(N-1)(4N+2) = 2N^3 - N^2 - N\,.$$

5.3 Otsuka-Ochiai Distance

The numerator of the Otsuka-Ochiai distance is the same as in the Jaccard distance's. It requires $2N$ operations. The denominator, $\sqrt{|a_i| \times |a_j|}$ or $\sqrt{\sum_h w_{ih} \times \sum_h w_{jh}}$, carries a cost of $N + N + 2$.

Adding the two costs plus the cost of taking the ratio and subtracting it from 1, we obtain a total cost per vertex pair of $4N + 4$. At the graph level, the total computational cost to convert an adjacency matrix into an all-pairs Otsuka-Ochiai distance matrix is, once again, $O(N^3)$:

$$\frac{1}{2}N(N-1)(4N+4) = 2N^3 - 2N\,.$$

5.4 Amplified Commute Distance

Although it is not immediately apparent when examining the worst-case complexity, amplified commute distance is by far the most costly to compute. To obtain it, we must first obtain the Laplacian, which requires $N(N-1) + N^2$ operations. Then, to obtain the matrix Γ, we must perform $N^2 + (N^2 + 1)$ operations. We must add to that the cost of obtaining the Moore-Penrose inverse, which requires $O(N^3)$ operations. These operations require the entire graph or Laplacian. Unlike with the other distances, the large bulk of the computational cost does not come from easily parallelizable node or node-pair operations.

Finally, for each pair of vertices, we must perform 17 additional arthimetic operations. The total computational cost to obtain all-pairs amplified commute distances is, here too, $O(N^3)$:

$$17\times\frac{1}{2}N(N-1)+N^3+N^2+(N^2+1)+N(N-1)+N^2 = N^3+\frac{25}{2}N^2-\frac{19}{2}N+1\,.$$

6 Numerical Comparisons

To compare the distance measures and assess the accuracy of each measure as a reflection of intra-cluster density, we generate synthetic graphs with known cluster membership, using the Python NetworkX library's [28] stochastic block model generator. In our experiments, we generate several graphs with varying graph and cluster sizes and inter and intra-cluster edge probabilities.

While the stochastic block model may, arguably, be described as simplistic or not reflective of complex networks, our goal is to examine the effect of intra-cluster edge probability (mean intra-cluster density) on mean intra-cluster distances. The stochastic block model provides an excellent tool to study this relationship.

For each test graph in the experiments below, we compute each of our four vertex to vertex distances, for every pair of vertices (generically denoted as d_{ij}). We then compute mean distances within each cluster k:

$$\bar{\mathcal{D}}_k = \frac{1}{0.5 \times n_k \times (n_k - 1)} \sum_{i,j} d_{ij} \, .$$

We also compute intra-cluster density for each cluster, as shown in Section 1.

To obtain a graph-wide assessment, we then take the mean of all cluster quantities (distances and densities) over the entire graph:

$$\begin{aligned} \bar{\mathcal{D}} &= \frac{1}{K} \sum_{k} \bar{\mathcal{D}}_k \text{ and} \\ \bar{\mathcal{K}}_{\text{intra}} &= \frac{1}{K} \sum_{k} \mathcal{K}_{\text{intra}}^{(k)} \, . \end{aligned}$$

Although our clusters vary in size, we ensure our conclusions are not skewed by disproportionately-sized clusters. As shown, both $\bar{D}$ and $\bar{\mathcal{K}}_{\text{intra}}$ are simple unweighted means. They remain unaffected by individual cluster sizes.

6.1 Test Data: Synthetic Graphs with Known Clusters

We use the stochastic block model to generate two sets of 18 graphs. For each graph, we vary the probability of an intra-cluster (P_{in}) and inter-cluster (P_{out}) edge. Details are shown in Table 1.

In the first set of 18 experiments, the graphs are unweighted. In the second set of experiments, the graphs have the same characteristics but their edges are weighted. Edges within clusters have a weight corresponding to the intra-cluster edge probability, while edges across clusters carry a weight equal to the inter-cluster edge probability.

Although our cluster sizes vary within the graph, they are kept constant in each graph. Clusters $c_1, \ldots, c_K$ in graphs $G_1, \ldots, G_{18}$ all have $n_1, \ldots, n_K$ nodes. All our graphs have $N = 3,400$ nodes, $K = 45$ clusters which contain $n_k \in [50, 88]$ nodes.

Table 1: Graph Characteristics

Graph	P_{in}	P_{out}
G_1	0	0
G_2	0.2	0
G_3	0.4	0
G_4	0.6	0
G_5	0.8	0
G_6	1	0
G_7	0	0.1
G_8	0.2	0.1
G_9	0.4	0.1
G_{10}	0.6	0.1
G_{11}	0.8	0.1
G_{12}	1	0.1
G_{13}	0	0.2
G_{14}	0.2	0.2
G_{15}	0.4	0.2
G_{16}	0.6	0.2
G_{17}	0.8	0.2
G_{18}	1	0.2

Table 2: Burt unweighted

P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	4.82	24.94	32.97
0.4	5.91	25.17	33.14
0.6	5.91	25.17	33.14
0.8	4.82	24.93	32.96
1	0.00	24.46	32.61

Table 3: Jaccard unweighted

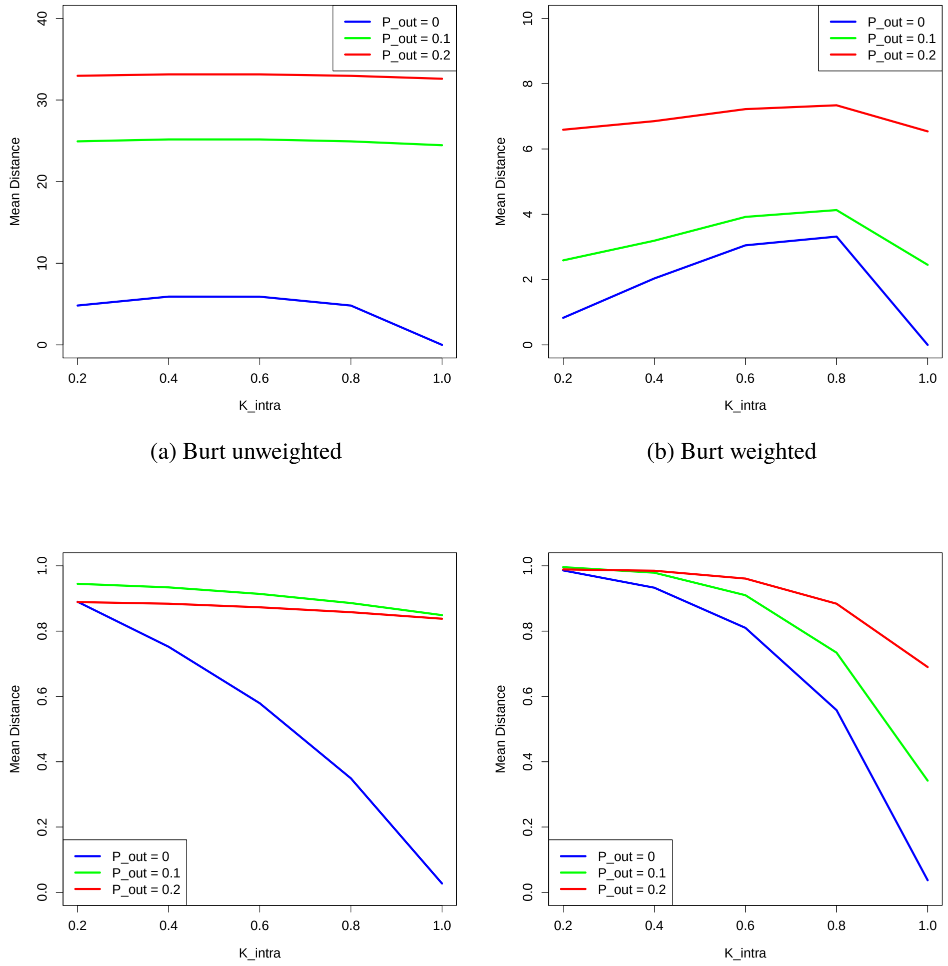
P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	0.89	0.95	0.89
0.4	0.75	0.93	0.88
0.6	0.58	0.91	0.87
0.8	0.35	0.89	0.86
1	0.03	0.85	0.84

6.2 Variations in Density and Distances

As mentioned earlier, we conduct several experiments with varying inter and intra-cluster edge probabilities. Naturally, these variations are directly reflected in the mean inter and intra-cluster densities. After all, these densities are an empirical estimate of (inter/intra-cluster) edge probabilities.

We immediately note that Jaccard, Otsuka-Ochiai and amplified commute distances offer a good reflection of intra-cluster density and that their change under variations in intra-cluster edge probability are in reverse lock-step with intra-cluster density. Under all scenarios, these (mean) distances decrease monotonically, as we increase intra-cluster edge probability (P_{in}) while keeping inter-cluster edge probability (P_{out}) constant. However, it must also be mentioned that amplified commute distance is on a very different scale than the other two distances. Finally, we observe that Burt's distance offers a very poor reflection of intra-cluster density. In fact, Burt's distance even increases with intra-cluster edge probability, under some scenarios. These results are consistent with our previous work [39], in which we also investigate the counter-intuitive behavior of Burt's distance.

A visual summary is presented in Figures 2 and 3. Results for unweighted graphs are shown in the plots on the left. Weighted graph results appear on the right. Full numerical details are shown in Tables 2 to 5 for unweighted graphs and in Tables 6 to 9 for weighted ones.



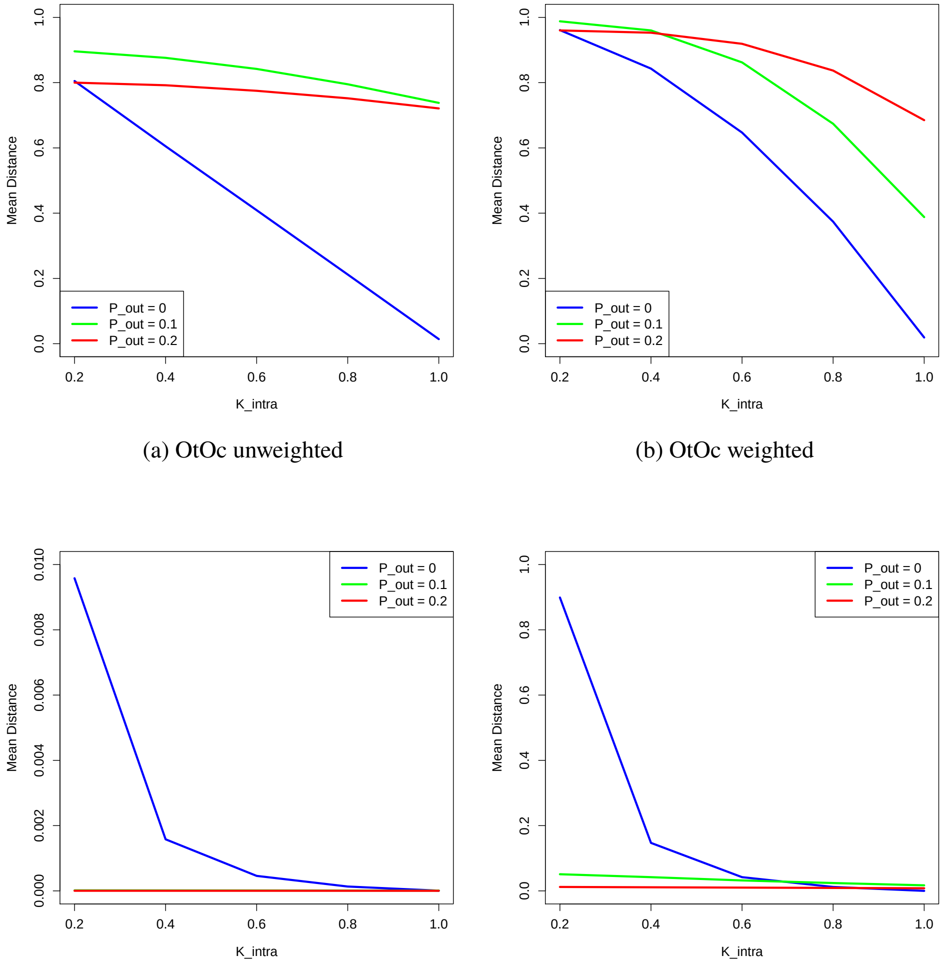
(a) Burt unweighted

(b) Burt weighted

(c) Jaccard unweighted

(d) Jaccard weighted

Figure 2: Distance variations, Burt and Jaccard (unweighted on left, weighted on right)



(a) OtOc unweighted

(b) OtOc weighted

(c) Amp. commute unweighted

(d) Amp. commute weighted

Figure 3: Distance variations, OtOc and Amplified commute (unweighted on left, weighted on right)

Table 4: OtOc unweighted

P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	0.80	0.90	0.80
0.4	0.60	0.88	0.79
0.6	0.41	0.84	0.78
0.8	0.21	0.80	0.75
1	0.01	0.74	0.72

Table 5: Amplified commute unweighted

P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	9.58E-03	1.49E-05	3.48E-06
0.4	1.58E-03	1.34E-05	3.29E-06
0.6	4.59E-04	1.19E-05	3.09E-06
0.8	1.33E-04	1.03E-05	2.87E-06
1	5.85E-06	8.89E-06	2.64E-06

Table 6: Burt weighted

P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	0.83	2.59	6.59
0.4	2.04	3.19	6.85
0.6	3.05	3.92	7.22
0.8	3.32	4.13	7.34
1	0.00	2.45	6.54

Table 7: Jaccard weighted

P_{in}	$P_{\text{out}} = 0$	$P_{\text{out}} = 0.1$	$P_{\text{out}} = 0.2$
0.2	0.99	1.00	0.99
0.4	0.93	0.98	0.99
0.6	0.81	0.91	0.96
0.8	0.56	0.73	0.88
1	0.04	0.34	0.69

Table 8: OtOc weighted

P_{in}	P_{out} = 0	P_{out} = 0.1	P_{out} = 0.2
0.2	0.96	0.99	0.96
0.4	0.84	0.96	0.95
0.6	0.65	0.86	0.92
0.8	0.37	0.67	0.84
1	0.02	0.39	0.69

Table 9: Amplified commute weighted

P_{in}	P_{out} = 0	P_{out} = 0.1	P_{out} = 0.2
0.2	0.899	0.051	0.012
0.4	0.147	0.042	0.011
0.6	0.042	0.032	0.010
0.8	0.012	0.024	0.009
1	0.000	0.017	0.008

6.3 Relative Variations and Inter-cluster Edge Probabilities

In order to compare all distances on an equal footing, regardless of their scale, we compute their relative variations under the effect of increases in intra-cluster and inter-cluster edge probabilities. These relative variations (δ) are computed as follows:

$$\delta = \frac{\text{dist after increase} - \text{dist before increase}}{\text{dist before increase}} .$$

In columns two, three and four of Table 10 and Table 11, we show the variations in distance as a percentage of its value in the previous example. For example, the first set of percentages are the variations of each quantity (K_{intra}, Burt,...) observed when P_{in} is increased from 0.2 to 0.4, expressed as a percentage of those same quantities when P_{in} was 0.2.

The last column (“Inter-cluster Effect”) contains the change in each quantity when P_{out} is doubled, when it is increased from 0.1 to 0.2, for an equal level of P_{in}. Again, this variation is expressed as a percentage of the same quantity when P_{out} is 0.1. These percentages quantify the effect of added noise (higher inter-cluster probability) on each distance and on our estimate of intra-cluster density (K_{intra}).

Predictably, we observe that noise has a negligible effect on intra-cluster densities ($\bar{K}_{\text{intra}}$) and that variations are probably due to the random edge generation process. On the other hand, noise seems to have a non-trivial effect on the distances. Thankfully, graphs that are clusterable, those displaying strong community structure, are also typically expected to be sparse and have few inter-cluster edges.

Table 10: Unweighted Graphs

	$\mathbf{P}_{\text{out}}$			**Inter-cluster Effect**
Mean	0	0.1	0.2	($P_{\text{out}}0.1 \rightarrow 0.2$)
	$\mathbf{P}_{\text{in}}\mathbf{0.2} \rightarrow \mathbf{0.4}$			
$\bar{\mathcal{K}}_{\text{intra}}$	100.53%	100.75%	98.69%	-2.04%
Burt	22.63%	0.94%	0.53%	-43.61%
Jacc	-15.50%	-1.20%	-0.59%	-50.54%
OtOc	-24.85%	-2.25%	-1.06%	-52.73%
Amplified	-83.53%	-10.37%	-5.37%	-48.16%
	$\mathbf{P}_{\text{in}}\mathbf{0.4} \rightarrow \mathbf{0.6}$			
$\bar{\mathcal{K}}_{\text{intra}}$	49.16%	50.02%	49.57%	-0.90%
Burt	-0.13%	0.00%	0.00%	30.98%
Jacc	-23.02%	-2.12%	-1.16%	-45.08%
OtOc	-32.41%	-3.89%	-2.06%	-47.02%
Amplified	-70.92%	-11.54%	-6.23%	-46.00%
	$\mathbf{P}_{\text{in}}\mathbf{0.6} \rightarrow \mathbf{0.8}$			
$\bar{\mathcal{K}}_{\text{intra}}$	33.06%	33.42%	33.30%	-0.34%
Burt	-18.41%	-0.94%	-0.54%	-42.81%
Jacc	-39.71%	-3.09%	-1.75%	-43.25%
OtOc	-48.10%	-5.53%	-3.07%	-44.58%
Amplified	-71.03%	-12.76%	-7.13%	-44.17%
	$\mathbf{P}_{\text{in}}\mathbf{0.8} \rightarrow \mathbf{1}$			
$\bar{\mathcal{K}}_{\text{intra}}$	25.05%	24.76%	24.92%	0.65%
Burt	-100.00%	-1.90%	-1.08%	-42.94%
Jacc	-92.16%	-4.13%	-2.37%	-42.59%
OtOc	-93.46%	-7.17%	-4.08%	-43.09%
Amplified	-95.60%	-13.98%	-8.02%	-42.66%

Table 11: Weighted Graphs

	$\mathbf{P}_{\text{out}}$			Noise Effect
Mean	0	0.1	0.2	($P_{\text{out}}0.1 \rightarrow 0.2$)
	$\mathbf{P}_{\text{in}}\mathbf{0.2 \rightarrow 0.4}$			
$\bar{\mathcal{K}}_{\text{intra}}$	300.70%	300.64%	298.99%	-0.55%
Burt	145.44%	23.25%	4.00%	-82.81%
Jacc	-5.39%	-1.70%	-0.44%	-74.14%
OtOc	-12.30%	-2.89%	-0.77%	-73.30%
Amplified	-83.63%	-17.43%	-5.65%	-67.60%
	$\mathbf{P}_{\text{in}}\mathbf{0.4 \rightarrow 0.6}$			
$\bar{\mathcal{K}}_{\text{intra}}$	123.71%	126.95%	124.57%	-1.87%
Burt	49.79%	22.85%	5.36%	-76.54%
Jacc	-13.16%	-7.05%	-2.45%	-65.31%
OtOc	-23.21%	-10.17%	-3.52%	-65.41%
Amplified	-71.30%	-23.04%	-8.55%	-62.89%
	$\mathbf{P}_{\text{in}}\mathbf{0.6 \rightarrow 0.8}$			
$\bar{\mathcal{K}}_{\text{intra}}$	77.31%	77.78%	78.05%	0.35%
Burt	8.80%	5.34%	1.62%	-69.76%
Jacc	-31.20%	-19.34%	-7.97%	-58.82%
OtOc	-42.30%	-21.77%	-8.97%	-58.79%
Amplified	-72.35%	-25.95%	-11.66%	-55.05%
	$\mathbf{P}_{\text{in}}\mathbf{0.8 \rightarrow 1}$			
$\bar{\mathcal{K}}_{\text{intra}}$	56.46%	56.35%	56.21%	-0.25%
Burt	-100.00%	-40.63%	-10.89%	-73.20%
Jacc	-93.41%	-53.49%	-21.95%	-58.96%
OtOc	-94.99%	-42.41%	-18.13%	-57.25%
Amplified	-99.87%	-26.73%	-13.47%	-49.59%

Table 12: Zachary Karate Club

		Mean Distance			
		Amp	Burt	Jacc	OtOc
Data	All	0.18	2.50	0.85	0.76
	Across	0.24	2.75	0.94	0.90
	Within	0.11	2.24	0.75	0.61

Nevertheless, precautions should be taken when using any distance in a clustering algorithm, especially when the graph under consideration is dense.

6.4 Illustrative Case Studies: Zachary's Karate Club and US College Football

Although our numerical tests using synthetic graphs were designed to show the relationship between density and distance, we find it useful to illustrate, interpret and validate our distances using real-world networks with known community (cluster) structures. We compute our distances for the node pairs of two famous graph data sets with known community labels for each node, Zachary's karate club [51] and the United States College Football Division IA games for the 2000 season [24]. We then examine the mean distances separating nodes sharing the same community (cluster) labeling and those separating nodes in different clusters.

The first network is a representation of the friendship links (edges) between members (nodes) of a karate club which dissolved due to a conflict between the instructor and the club's administrator. The club then gave rise to two new (sub)clubs named "Mr. Hi" and "John A" (also sometimes labeled as "Officer"). These two new (sub)clubs are considered node communities (clusters) and are often used as "ground truth" in tests of clustering algorithms.

We use a version of the Zachary data set generated with the NetworkX library [28]. Using this graph data set and its node-cluster membership labeling, we compute the mean distances between members of the same (sub)clubs, between members of the two different (sub)clubs and the overall mean between all nodes of the graph. Results are reported in Table 12.

In Table 12, we clearly note that distances are greatest for nodes in different (sub)clubs (Across) and that they are lowest for nodes in the same (sub)clubs (Within). Naturally, the mean distance between all nodes (All), regardless of their (sub)club membership, is in between these two extremes. These results are consistent with our own earlier observations and with previous investigations into this data set in the literature. They reflect the fact that members of the original club tended to regroup into the new (sub)clubs formed along friendship lines. Our

Table 13: Division IA College Football Games

		Mean Distances			
		Amp	Burt	Jacc	OtOc
Data	All	0.04	4.37	0.95	0.91
	Across	0.04	4.45	0.97	0.94
	Within	0.01	3.09	0.69	0.55

computations show that nodes within each of the two new (sub)clubs share a greater number of connections (friendships) and are separated by shorter distances, than nodes in the other (sub)club.

The second data set, the United States College Football Division IA games for the 2000 season is a network representation of the regular season encounters between 115 college football teams. These teams are represented by vertices and grouped into 12 conferences (clusters). Edges connect teams that faced each other at least once during the regular season. Teams within a conference all face each other during regular season, while they do not necessarily face teams outside their conference during the regular season. Therefore, there are more shared connections between teams of the same conference than between teams in different conferences.

We perform the same computations as in the case of the Zachary data set, using the football data set. We compute the mean distances separating teams within the same conference, in different conferences and between all nodes on the graph. Results are reported in Table 13. Here again, we note that nodes within clusters (teams within conferences) are separated by noticeably shorter distances than nodes in different clusters (conferences).

7 Our Chosen Distance

While our tests on real-world benchmark graphs in Section 6.4 show that all distances capture the community structure of the data, the remainder of our computational results in Section 6 reveal that Burt distance is an unreliable gauge of it. This conclusion is also consistent with our prior work [39]. Meanwhile, Jaccard, Otsuka-Ochiai and amplified distances all offer adequate reflections of intra-cluster density. Clustering by minimizing any one should result in dense clusters. However, in practice, amplified commute distance is more costly to compute than the other two distances.

Jaccard similarity and its complement, the Jaccard distance, are used widely in a variety of different fields, including complex networks [9]. Because of this widespread use, clear interpretability and availability of pre-built computational functions, we recommend the Jaccard distance as a vertex to vertex distance measure for graph clustering. For example, the NetworkX library offers a Jaccard similarity function, which we use in this work [28].

8 Conclusion

We show that Jaccard, Otsuka-Ochiai and amplified commute distances, when averaged over clusters, follow the evolution of intra-cluster density monotonically. They are all shown to vary in an opposite direction to intra-cluster density. However, amplified commute distance, unlike the other two distances, cannot easily be decomposed into pairwise computations. Meanwhile, Burt's distance displays a very counter-intuitive behavior with respect to edge probability and intra-cluster density variations.

Unfortunately, Jaccard, Otsuka-Ochiai and amplified commute distances all seem to be sensitive to noise. Our future work will focus on a study of Jaccard and Otsuka-Ochiai distances under the effect of noise and on larger scale graphs. We will also investigate possible corrections that could render them more robust.

List of Abbreviations

N: total number of vertices in the entire graph
n_k: number of vertices in cluster k
V: set of all vertices in the graph
$|V|$: cardinality of the set V (same as N)
K: number of (vertex) clusters in the graph
$\mathcal{K}$: graph's overall density
$\mathcal{K}_{\text{intra}}^{(k)}$: intra-cluster density in cluster k, density of the induced subgraph formed by the vertices in cluster k
e_k: set of edges connecting nodes in cluster k
$|e_k|$: cardinality of the set e_k
Burt: Burt's distance
Jacc: Jaccard distance
OtOc: Otsuka-Ochiai distance
amp: Amplified commute distance
G_i: graph i, where $i \in \{1, \ldots, 18\}$

Funding

PM thanks Fujitsu Limited and Fujitsu Consulting (Canada) Inc. for providing financial support.

Acknowledgements

- The work of Alexander Ponomarenko was conducted within the framework of the Basic Research Program at the National Research University Higher School of Economics (HSE).
- The authors thank the organizers and attendees of The 10th International Conference on Network Analysis at the Laboratory of Algorithms and Technologies for Networks Analysis in Nizhny Novgorod.
- PM wishes to thank Prof. Ali Sheikholeslami of The Dept. of Electrical & Computer Engineering at the University of Toronto for his support and guidance.

References

[1] M. Ackerman and S. Ben-David. Measures of clustering quality: A working set of axioms for clustering. *Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference*, pages 121–128, 01 2008.

[2] P. Akara-pipattana, T. Chotibut, and O. Evnin. Resistance distance distribution in large sparse random graphs. *arXiv e-prints*, 2021.

[3] M. Aramon, G. Rosenberg, E. Valiante, T. Miyazawa, H. Tamura, and H.G. Katzgraber. Physics-Inspired Optimization for Quadratic Unconstrained Problems Using a Digital Annealer. *Frontiers in Physics*, 7, Apr 2019.

[4] K. Avrachenkov, P. Chebotarev, and D. Rubanov. Similarities on graphs: Kernels versus proximity measures. *European Journal of Combinatorics*, 80:47–56, 2019. Special Issue in Memory of Michel Marie Deza.

[5] Christian Bauckhage, Nico Piatkowski, Rafet Sifa, Dirk Hecker, and Stefan Wrobel. A QUBO formulation of the k-medoids problem. In Robert Jäschke and Matthias Weidlich, editors, *Proceedings of the Conference on "Lernen, Wissen, Daten, Analysen", Berlin, Germany, September 30 - October 2, 2019*, volume 2454 of *CEUR Workshop Proceedings*, pages 54–63. CEUR-WS.org, 2019.

[6] V. D. Blondel, J-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, oct 2008.

[7] M. Boguñá, I. Bonamassa, Manlio M. De Domenico, Shlomo S. Havlin, D. Krioukov, and M. Á. Serrano. Network geometry. *Nature Reviews Physics*, 3(2):114–135, January 2021.

[8] R.S. Burt. Positions in Networks*. *Social Forces*, 55(1):93–122, 09 1976.

[9] E. Camby and G. Caporossi. The extended Jaccard distance in complex networks. *Les Cahiers du GERAD*, G-2017-77, 09 2017.

[10] P. Chebotarev. A class of graph-geodetic distances generalizing the shortest-path and the resistance distances. *Discrete Applied Mathematics*, 159(5):295–302, 2011.

[11] P. Chebotarev and E. Shamis. Matrix-Forest Theorems. *arXiv Mathematics e-prints*, 2006.

[12] P. Chebotarev and E. Shamis. The Forest Metrics for Graph Vertices. *arXiv Mathematics e-prints*, 2006.

[13] P. Chebotarev and E. Shamis. The Matrix-Forest Theorem and Measuring Relations in Small Social Groups. *arXiv Mathematics e-prints*, 2006.

[14] F. Chung and L. Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.

[15] E. Estrada. The communicability distance in graphs. *Linear Algebra and its Applications*, 436(11):4317–4328, 2012.

[16] N. Fan and P.M. Pardalos. Linear and Quadratic Programming Approaches for the General Graph Partitioning Problem. *J. of Global Optimization*, 48(1):57–71, September 2010.

[17] N. Fan and P.M. Pardalos. Robust Optimization of Graph Partitioning and Critical Node Detection in Analyzing Networks. In *Proceedings of the 4th International Conference on Combinatorial Optimization and Applications - Volume Part I*, COCOA'10, pages 170–183, Berlin, Heidelberg, 2010. Springer-Verlag.

[18] N. Fan, Q.P. Zheng, and P.M. Pardalos. Robust optimization of graph partitioning involving interval uncertainty. *Theoretical Computer Science*, 447:53–61, 2012. Combinational Algorithms and Applications (COCOA 2010).

[19] S. Fortunato. Community detection in graphs. *Physics Reports*, 486:75–174, February 2010.

[20] S. Fortunato and M. Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.

[21] S. Fortunato and D. Hric. Community detection in networks: A user guide. *ArXiv e-prints*, November 2016.

[22] F. Fouss, K. Francoisse, L. Yen, A. Pirotte, and M. Saerens. An experimental investigation of kernels on graphs for collaborative recommendation and semisupervised classification. *Neural Networks*, 31:53–72, 2012.

[23] Y. Fu and P.W. Anderson. Application of statistical mechanics to NP-complete problems in combinatorial optimisation. *Journal of Physics A: Mathematical and General*, 19(9):1605–1620, June 1986.

[24] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[25] F. Glover, G. Kochenberger, and Y. Du. A Tutorial on Formulating and Using QUBO Models. *arXiv e-prints*, page arXiv:1811.11538, June 2018.

[26] B. H. Good, Y.-A. de Montjoye, and A. Clauset. Performance of modularity maximization in practical contexts. *preprint*, 81(4):046106, April 2010.

[27] I. Granata, M. R. Guarracino, L. Maddalena, and I. Manipur. Network distances for weighted digraphs. In Yury Kochetov, Igor Bykadorov, and Tatiana Gruzdeva, editors, *Mathematical Optimization Theory and Operations Research*, pages 389–408, Cham, 2020. Springer International Publishing.

[28] A.A. Hagberg, D.A. Schult, and P.J. Swart. Exploring Network Structure, Dynamics, and Function using NetworkX. In G. Varoquaux, T. Vaught, and J. Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11–15, Pasadena, CA USA, 2008.

[29] S.W. Hong, P. Miasnikof, R. Kwon, and Y. Lawryshyn. Market graph clustering via qubo and digital annealing. *Journal of Risk and Financial Management*, 14(1), 2021.

[30] P. Jaccard. Étude de la distribution florale dans une portion des Alpes et du Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:547–579, 01 1901.

[31] A. Kehagias and L. Pitsoulis. Bad communities with high modularity. *The European Physical Journal B*, 86(7):330, Jul 2013.

[32] I. Kivimäki, M. Shimbo, and M. Saerens. Developments in the theory of randomized shortest paths with a comparison of graph node distances. *Physica A: Statistical Mechanics and its Applications*, 393:600–616, 2014.

[33] D. Krioukov, F. Papadopoulos, M. Kitsak, Amin A. Vahdat, and M. Boguñá. Hyperbolic geometry of complex networks. *arXiv e-prints*, 82(3):036106, September 2010.

[34] J. Leskovec and A. Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[35] L. Liberti. Distance geometry and data science. *TOP*, 28:271–339, 2020.

[36] A. Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2, Feb 2014.

[37] Marco M. Saerens, Y. Achbany, F. Fouss, and L. Yen. Randomized shortest-path problems: Two related models. *Neural Computation*, 21(8):2363–2404, 2009.

[38] P. Miasnikof, M. Bagherbeik, and A. Sheikholeslami. Graph clustering with Boltzmann machines. *submitted*, 2021.

[39] P. Miasnikof, A. Y. Shestopaloff, L. Pitsoulis, A. Ponomarenko, and Y. Lawryshyn. Distances on a graph. In Rosa M. Benito, Chantal Cherifi, Hocine Cherifi, Esteban Moro, Luis Mateus Rocha, and Marta Sales-Pardo, editors, *Complex Networks & Their Applications IX*, pages 189–199, Cham, 2021. Springer International Publishing.

[40] P. Miasnikof, A.Y. Shestopaloff, A.J. Bonner, and Y. Lawryshyn. *A Statistical Performance Analysis of Graph Clustering Algorithms*, chapter 11. Lecture Notes in Computer Science. Springer Nature, 6 2018.

[41] P. Miasnikof, A.Y. Shestopaloff, A.J. Bonner, Y. Lawryshyn, and P.M. Pardalos. A density-based statistical analysis of graph clustering algorithm performance. *Journal of Complex Networks*, 8(3), 08 2020. cnaa012.

[42] A. Ochiai. Zoogeographical studies on the soleoid fishes found in japan and its neighbouring regions-i. *NIPPON SUISAN GAKKAISHI*, 22(9):522–525, 1957.

[43] A. Ponomarenko, L. Pitsoulis, and M. Shamshetdinov. Overlapping community detection in networks based on link partitioning and partitioning around medoids. *PLOS ONE*, 16(8):1–43, 08 2021.

[44] L. Ostroumova Prokhorenkova, P. Prałat, and A. Raigorodskii. Modularity of Complex Networks Models. In A. Bonato, F.C. Graham, and P. Prałat, editors, *Algorithms and Models for the Web Graph*, pages 115–126, Cham, 2016. Springer International Publishing.

[45] L. Ostroumova Prokhorenkova, P. Prałat, and A. Raigorodskii. Modularity in several random graph models. *Electronic Notes in Discrete Mathematics*, 61:947–953, 2017. The European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB'17).

[46] S.E. Schaeffer. Survey: Graph clustering. *Comput. Sci. Rev.*, 1(1):27–64, August 2007.

[47] F. Sommer, F. Fouss, and M. Saerens. Comparison of graph node distances on clustering tasks. In Alessandro E.P. Villa, Paolo Masulli, and Antonio Javier Pons Rivero, editors, *Artificial Neural Networks and Machine Learning – ICANN 2016*, pages 192–201, Cham, 2016. Springer International Publishing.

[48] U. von Luxburg, A. Radl, and M. Hein. Getting lost in space: Large sample analysis of the resistance distance. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 2622–2630. Curran Associates, Inc., 2010.

[49] U. von Luxburg, A. Radl, and M. Hein. Hitting and commute times in large random neighborhood graphs. *Journal of Machine Learning Research*, 15(52):1751–1798, 2014.

[50] L. Yen, M. Saerens, A. Mantrach, and M. Shimbo. A family of dissimilarity measures between nodes generalizing both the shortest-path and the commute-time distances. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 785–793, New York, NY, USA, 2008. Association for Computing Machinery.

[51] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.