Multidimensional Scaling: Approximation and Complexity

Erik Demaine [1] **Adam Hesterberg** [2] **Frederic Koehler** [3] **Jayson Lynch** [4] **John Urschel** [3]

Abstract

Metric Multidimensional scaling (MDS) is a classical method for generating meaningful (non-linear) low-dimensional embeddings of high-dimensional data. MDS has a long history in the statistics, machine learning, and graph drawing communities. In particular, the Kamada-Kawai force-directed graph drawing method is equivalent to MDS and is one of the most popular ways in practice to embed graphs into low dimensions. Despite its ubiquity, our theoretical understanding of MDS remains limited as its objective function is highly non-convex. In this paper, we prove that minimizing the Kamada-Kawai objective is NP-hard and give a provable approximation algorithm for optimizing it, which in particular is a PTAS on low-diameter graphs. We supplement this result with experiments suggesting possible connections between our greedy approximation algorithm and gradient-based methods.

1. Introduction

Given the distances between data points living in a high dimensional space, how can we meaningfully visualize their relationships? This is a fundamental task in exploratory data analysis for which a variety of different approaches have been proposed. Many of these approaches seek to visualize high-dimensional data by embedding it into lower dimensional, e.g. two or three-dimensional, space.

Metric multidimensional scaling (MDS or mMDS) (Kruskal, 1964a; 1978) is a classical approach to this problem which attempts to find a low-dimensional embedding that accurately represents the *distances between points*. Originally motivated by applications in psychometrics, MDS has now been recognized as a fundamental tool for data analysis across a broad range of disciplines. See the texts (Kruskal, 1978; Borg & Groenen, 2005) for more details, including a discussion of applications to data from scientific, economic, political, and other domains. Compared to other classical visualization tools like PCA[1], metric multidimensional scaling has the advantage that it 1) is not restricted to linear projections of the data, i.e. it is nonlinear, and 2) is applicable to data from an arbitrary *metric space*, rather than just Euclidean space. Because of this versatility, MDS has also become one of the most popular algorithms in the field of *graph drawing*, where the goal is to visualize relationships between nodes (e.g. people in a social network). In this context, MDS was independently proposed by Kamada and Kawai (Kamada et al., 1989) as a *force-directed graph drawing* method.

In this paper, we consider the algorithmic problem of computing the optimal embedding under the MDS/Kamada-Kawai objective. The Kamada-Kawai objective is to minimize the following energy/stress functional $E : \mathbb{R}^{rn} \to \mathbb{R}$

$$E(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i<j} \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{d(i,j)} - 1 \right)^2, \qquad (1)$$

which corresponds to the physical situation where $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$ are particles and for each $i \neq j$, particles $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected by an idealized spring with equilibrium length $d(i,j)$ following Hooke's law with spring constant $k_{ij} = \frac{1}{d(i,j)^2}$. In applications to visualization, the choice of dimension is often small, i.e. $r = 1, 2, 3$. We also note that in (1) the terms in the sum are sometimes re-weighted with vertex or edge weights, which we discuss in more detail later.

In practice, the MDS/Kamada-Kawai objective (1) is optimized via a heuristic procedure like gradient descent (Kruskal, 1964b; Zheng et al., 2018) or stress majorization (De Leeuw et al., 1977; Gansner et al., 2004). Because the objective is non-convex, these algorithms may not reach the global minimum, but instead may terminate at approximate critical points of the objective function. Heuristics such as restarting an algorithm from different initializations

[1]Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA, USA [2]John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA [3]Department of Mathematics, MIT, Cambridge, MA, USA [4]Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada. Correspondence to: Frederic Koehler <fkoehler@mit.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

[1]In the literature, PCA is sometimes referred to as *classical multidimensional scaling*, in contrast to *metric multidimensional scaling*, which we study in this work.

and using modified step size schedules have been proposed to improve the quality of results. In practice, these heuristic methods do seem to work well for the Kamada-Kawai objective and are implemented in popular packages like GRAPHVIZ (Ellson et al., 2001) and the SMACOF package in R.

1.1. Our Results

In this work, we revisit this problem from an approximation algorithms perspective. First, we resolve the computational complexity of minimizing (1) by proving that finding the global minimum is NP-hard, even for graph metrics (where the metric is the shortest path distance on a graph). Consider the decision version of stress minimization over graph metrics, which we formally define below:

STRESS MINIMIZATION
Input: Graph $G = ([n], E)$, $r \in \mathbb{N}$, $L \geq 0$.
Output: TRUE if there exists $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{nr}$ such that $E(\mathbf{x}) \leq L$; FALSE otherwise.

Theorem 1. *There exists a polynomial $p(n)$ such that the following gap version of* **STRESS MINIMIZATION** *in dimension $r = 1$ is NP-hard: given an input graph G with n vertices and $L > 0$, return TRUE if there exists $\mathbf{x}$ such that $E(\mathbf{x}) \leq L$ and return FALSE if for every $\mathbf{x}$, $E(\mathbf{x}) \geq L + 1/p(n)$. Furthermore, the problem is hard even restricted to input graphs with diameter bounded by an absolute constant.*

As a gap problem, the output is allowed to be arbitrary if neither case holds; the hardness of the gap formulation shows that there cannot exist a *Fully-Polynomial Randomized Approximation Scheme* (FPRAS) for this problem if P $\neq$ NP, i.e. the runtime cannot be polynomial in the desired approximation guarantee. Our reduction shows this problem is hard even when the input graph has *low diameter* (even bounded by an absolute constant): this is a natural setting to consider since many real world graphs (for example, social networks (Dodds et al., 2003)) and random graph models (Watts & Strogatz, 1998) indeed have low diameter due to the "small-world phenomena". Other key aspects of this hardness proof are: 1) we show the problem is hard even when the input d is a graph metric, and 2) we show it is hard even in its canonical unweighted formulation (1).

Given that computing the minimizer is NP-hard, a natural question is whether there exist polynomial time approximation algorithms for minimizing (1). We show that if the input graph has bounded diameter $D = O(1)$, then there indeed exists a *Polynomial-Time Approximation Scheme* (PTAS) to minimize (1), i.e. for fixed $\epsilon > 0$ and fixed D there exists an algorithm to approximate the global minimum of a n vertex diameter D graph up to multiplicative error $(1 + \epsilon)$ in time $f(\epsilon, D) \cdot poly(n)$. More generally, we show:

Theorem 2 (Informal version of Theorem 4)**.** *Let $R > \epsilon > 0$ be arbitrary. Algorithm* KKSCHEME *runs in time $n^2(R/\epsilon)^{O(rR^4/\epsilon^2)}$ and outputs $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$ with $\|\mathbf{x}_i\| \leq R$ such that*

$$\mathbb{E}\left[E(\mathbf{x}_1, \ldots, \mathbf{x}_n)\right] \leq E(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*) + \epsilon n^2$$

for any $\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*$ with $\|\mathbf{x}_i^*\| \leq R$ for all i, where $\mathbb{E}$ is the expectation over the randomness of the algorithm.

where KKSCHEME is a simple greedy algorithm described in Section 4 below. The fact that this result is a PTAS for bounded diameter graphs follows from combining it with the two structural results regarding optimal Kamada-Kawai embeddings, which are of independent interest. The first (Lemma 4) shows that the optimal objective value for low diameter graphs must be of order $\Omega(n^2)$ and the second (Lemma 5) shows that the optimal KK embedding is "contractive" in the sense that the diameter of the output is never much larger than the diameter of the input.

Lemma 1 (Informal version of Lemma 4)**.** *For any target dimension $r \geq 1$, all graphs of diameter $D = O(n^{1/r})$ satisfy $E(\mathbf{x}) = \Omega(n^2/D^r)$ for all $\mathbf{x}$.*

Lemma 2 (Informal version of Lemma 5)**.** *For any graph of diameter D and any target dimension $r \geq 1$, any global minimizer of $E(\mathbf{x})$ satisfies*

$$\max_{i,j} \|\mathbf{x}_i - \mathbf{x}_j\| = O(D \log\log D),$$

i.e. the diameter of the embedding is $O(D \log\log D)$.

1.2. Related Work

Other Approaches to Nonlinear Dimensionality Reduction and Visualization. Recently, there has been renewed interest in force-directed graph layouts due to new applications in machine learning and data science. MDS itself is a popular technique for dimension reduction. Newer techniques, such as t-SNE (Maaten & Hinton, 2008) and UMAP (McInnes et al., 2018), can be viewed as similar type of force-directed weighted graph drawing with more complex objectives than Kamada-Kawai (see the discussion in (McInnes et al., 2018)); in comparison, some other dimensionality reduction methods, e.g. Laplacian eigenmaps (Belkin & Niyogi, 2003), are based on spectral embeddings of graphs.

In practice, methods like t-SNE and UMAP appear to work quite well, even though they are based on optimizing non-convex objectives with gradient descent, which in general comes with no guarantee of success. Towards explaining this phenomena, t-SNE has been mathematically analyzed

in a fairly specific setting where the data is split into well-separated clusters (e.g. generated by well-separated Gaussian mixtures); in this case, the works (Arora et al., 2018; Linderman & Steinerberger, 2019) prove that the visualization recovers the corresponding cluster structure. A difficulty when proving more general guarantees is that the t-SNE and UMAP objectives are fairly complex, and hence not so easy to mathematically analyze.

Partially for this reason, in this work we focus on the simpler metric MDS/Kamada-Kawai objective. Experimentally, it has been observed that, using this objective, it is easy to find high quality minima in many different situations (see e.g. (Zheng et al., 2018)), but to our knowledge there has not been a mathematical explanation of this phenomena.

Other related work. In the multidimensional scaling literature, there has been some study of the *local convergence* of algorithms like stress majorization, see for example (De Leeuw, 1988), which shows that stress majorization will converge quickly if in a sufficiently small neighborhood of a local minimum. This work seems to propose the first provable guarantees for global optimization. The closest previous hardness result is the work of (Cayton & Dasgupta, 2006) where they showed that a similar problem is hard. In their problem: 1) the terms in (1) are weighted by $d(i,j)$ and absolute value loss replaces the squared loss and 2) the input is an arbitrary pseudometric where nodes in the input are allowed to be at distance zero from each other. The second assumption makes the diameter (ratio of max to min distance in the input) infinite, and this is a major obstruction to modifying their approach to show Theorem 1. See Remark 1 for further discussion. A much earlier hardness result is the work of (Saxe, 1979), in the easier (for proving hardness) case where distortion is only measured with respect to edges of the graph.

In the approximation algorithms literature, there has been a great deal of interest in optimizing the worst-case distortion of metric embeddings into various spaces, see e.g. (Badoiu et al., 2005) for approximation algorithms for embeddings into one dimension, and (Deza & Laurent, 2009; Naor, 2012) for more general surveys of low distortion metric embeddings. Though conceptually related, the techniques used in this literature are not generally targeted for minimizing a measure of average pairwise distortion like (1).

In the graph drawing literature, there are a number of competing methods for drawing a graph, with the best approach depending on application (Battista et al., 1998). Tutte's spring embedding theorem is often considered the seminal work in the force-directed layout community, and provides a method for producing a planar drawing of a three-connected planar graph (Tutte, 1963). Though the problem under consideration in this work does indeed belong to the class of force-directed layouts, we stress the layouts under consideration do not minimize edge crossings in any sense.

Notation. In the remainder of the paper, we will generally assume the input is given as an unweighted graph to simplify notation; however, for the upper bounds (e.g. Theorem 2) we do handle the general case of arbitrary metrics with distances in $[1, D]$ — note that the lower bound of 1 is without loss of generality after re-scaling. In the lower bound (i.e. Theorem 1), we prove the (stronger) result that the problem is hard when restricted to graph metrics, instead of just for arbitrary metrics. We use standard asymptotic notation: $f(n) = O(g(n))$ means that $\limsup_{n\to\infty} \frac{f(n)}{g(n)} < \infty$, $f(n) = \Omega(g(n))$ means that $\liminf_{n\to\infty} \frac{f(n)}{g(n)} > 0$, and $f(n) = \Theta(g(n))$ means that $f(n) = \Omega(g(n))$ and $f(n) = O(g(n))$. The notation $[n]$ denotes $\{1, \ldots, n\}$. Unless otherwise noted, $\|\cdot\|$ denotes the Euclidean norm.

We also recall that a *metric* $d : V \times V \to \mathbb{R}_{\geq 0}$ on a set V is formally defined to be any function satisfying 1) $d(v,w) = 0$ iff $v = w$, 2) $d(v,w) = d(w,v)$ for all $v, w \in V$ and 3) $d(v,w) \leq d(v,u) + d(u,w)$ for any $u, v, w \in V$. A *pseudometric* relaxes 1) to the requirement that $d(v,v) = 0$ for all v.

2. Structural Results for Optimal Embeddings

In this section, we present two results regarding optimal layouts of a given graph. In particular, we provide a lower bound for the energy of a graph layout and an upper bound for the diameter of an optimal layout. The techniques used primarily involve estimating different components of the objective function $E(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ given by (1) (written as $E(\mathbf{x})$ in this section for convenience). For this reason, we introduce the notation

$$E_{i,j}(\mathbf{x}) := \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{d(i,j)} - 1\right)^2 \quad \text{for } i,j \in [n],$$

$$E_S(\mathbf{x}) := \sum_{\substack{i,j\in S \\ i<j}} E_{i,j}(\mathbf{x}) \quad \text{for } S \subset [n],$$

$$E_{S,T}(\mathbf{x}) := \sum_{i\in S}\sum_{j\in T} E_{i,j}(\mathbf{x}) \quad \text{for } S, T \subset [n],\ S \cap T = \emptyset.$$

We also make use of this notation in Appendices A and B.

First, we recall the following standard ϵ-net estimate.

Lemma 3 (Corollary 4.2.13 of (Vershynin, 2018))**.** *Let $B_R = \{x : \|x\| \leq R\} \subset \mathbb{R}^r$ be the origin-centered radius R ball in r dimensions. For any $\epsilon \in (0, R)$ there exists a subset $S_\epsilon \subset B_R$ with $|S_\epsilon| \leq (3R/\epsilon)^r$ such that*

$$\max_{\|x\|\leq R} \min_{y\in S_\epsilon} \|x - y\| \leq \epsilon,$$

i.e. S_ϵ is an ϵ-net of B_R.

Using this result, we prove the following lower bound for the objective value of any layout of a diameter D graph in $\mathbb{R}^r$.

Lemma 4. *Let $G = ([n], E)$ have diameter*

$$D \leq \frac{(n/2)^{1/r}}{10}.$$

Then any layout $\mathbf{x} \in \mathbb{R}^{rn}$ has energy

$$E(\mathbf{x}) \geq \frac{n^2}{81(10D)^r}.$$

Proof. Let $G = ([n], E)$ have diameter $D \leq (n/2)^{1/r}/10$, and suppose that there exists a layout $\mathbf{x} \subset \mathbb{R}^r$ of G in dimension r with energy $E(\mathbf{x}) = cn^2$ for some $c \leq 1/810$. If no such layout exists, then we are done. We aim to lower bound the possible values of c. For each vertex $i \in [n]$, we consider the quantity $E_{i,V\setminus i}(\mathbf{x})$. The sum

$$\sum_{i\in[n]} E_{i,V\setminus i}(\mathbf{x}) = 2cn^2,$$

and so there exists some $i' \in [n]$ such that $E_{i',V\setminus i'}(\mathbf{x}) \leq 2cn$. By Markov's inequality,

$$\left|\{j \in [n] \,|\, E_{i',j}(\mathbf{x}) > 10c\}\right| < n/5,$$

and so at least $4n/5$ vertices (including i') in $[n]$ satisfy

$$\left(\frac{\|\mathbf{x}_{i'} - \mathbf{x}_j\|}{d(i',j)} - 1\right)^2 \leq 10c,$$

and also

$$\|\mathbf{x}_{i'} - \mathbf{x}_j\| \leq d(i',j)(1+\sqrt{10c}) \leq \frac{10}{9}D.$$

The remainder of the proof consists of taking the d-dimensional ball with center $\mathbf{x}_{i^*}$ and radius $10D/9$ (which contains $\geq 4n/5$ vertices), partitioning it into smaller sub-regions, and then lower bounding the energy resulting from the interactions between vertices within each sub-region.

By applying Lemma 3 with $R := 10D/9$ and $\epsilon := 1/3$, we may partition the r dimensional ball with center $\mathbf{x}_{i'}$ and radius $10D/9$ into $(10D)^r$ disjoint regions, each of diameter at most $2/3$. For each of these regions, we denote by $S_j \subset [n]$, $j \in [(10D)^r]$, the subset of vertices whose corresponding point lies in the corresponding region. As each region is of diameter at most $2/3$ and the graph distance between any two distinct vertices is at least one, either

$$E_{S_j}(\mathbf{x}) \geq \binom{|S_j|}{2}(2/3-1)^2 = \frac{|S_j|(|S_j|-1)}{18}$$

or $|S_j| = 0$. Empty intervals provide no benefit and can be safely ignored. The optimization problem

$$\min \sum_{k=1}^{\ell} m_k(m_k-1) \quad \text{s.t.} \sum_{k=1}^{\ell} m_k = m,\ m_k \geq 1,\ k \in [\ell],$$

has a non-empty feasible region for $m \geq \ell$, and the solution is given by $m(m/\ell - 1)$ (achieved when $m_k = m/\ell$ for all k). In our situation, $m := 4n/5$ and $\ell := (10D)^r$, and, by assumption, $m \geq \ell$. This leads to the lower bound

$$cn^2 = E(\mathbf{x}) \geq \sum_{j=1}^{\ell} E_{S_j}(\mathbf{x}) \geq \frac{4n}{90}\left[\frac{4n}{5(10D)^r} - 1\right],$$

which implies that

$$c \geq \frac{16}{450(10D)^r}\left(1 - \frac{5(10D)^r}{4n}\right) \geq \frac{1}{75(10D)^r}$$

for $D \leq (n/2)^{1/r}/10$. This completes the proof. □

The above estimate has the correct dependence for $r = 1$. For instance, consider the lexicographical product of a path P_D and a clique $K_{n/D}$: i.e. a graph with D cliques in a line, and complete bipartite graphs between neighboring cliques. This graph has diameter D, and the layout in which the "vertices" (each corresponding to a copy of $K_{n/D}$) of P_D lie exactly at the integer values $[D]$ has objective value $\frac{n}{2}(n/D - 1)$. This estimate is almost certainly not tight for dimensions $r > 1$, as there is no higher dimensional analogue of the path (i.e., a graph with $O(D^r)$ vertices and diameter D that embeds isometrically in $\mathbb{R}^r$).

Next, we provide an upper bound for the diameter of any optimal layout of a diameter D graph. For the sake of space, the proof of this result is reserved for Appendix A.

Lemma 5 (Proved in Appendix A)**.** *Let $G = ([n], E)$ have diameter D. Then, for any optimal layout $\mathbf{x} \in \mathbb{R}^{rn}$, i.e., $\mathbf{x}$ such that $E(\mathbf{x}) \leq E(\mathbf{y})$ for all $\mathbf{y} \in \mathbb{R}^{rn}$,*

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2 \lesssim D \log\log D$$

for all $i, j \in [n]$.

While the above estimate is sufficient for our purposes, we conjecture that this is not tight, and that the diameter of an optimal layout of a diameter D graph is always at most $2D$.

3. Algorithmic Lower Bounds

In this section, we discuss algorithmic lower bounds for multidimensional scaling. In particular, we provide a sketch of the reduction used in the proof of Theorem 1. The formal proof itself is quite involved, and is therefore reserved for Appendix B.

To show that minimizing (1) is NP-hard in dimension $r = 1$, we use a reduction from a version of Max All-Equal 3SAT. The Max All-Equal 3SAT decision problem asks whether, given variables $t_1, \ldots, t_\ell$, clauses $C_1, \ldots, C_m \subset \{t_1, \ldots, t_\ell, \bar{t}_1, \ldots, \bar{t}_\ell\}$ each consisting of at most three literals (variables or their negation), and some value L, there exists an assignment of variables such that at least L clauses have all literals equal. The Max All-Equal 3SAT decision problem is known to be APX-hard, as it does not satisfy the conditions of the Max CSP classification theorem for a polynomial time optimizable Max CSP (Khanna et al., 2001). More precisely, this is because of the following properties: 1) setting all variables true or all variables false does not satisfy all clauses, and 2) all clauses cannot be written in disjunctive normal form as two terms, one with all unnegated variables and one with all negated variables.

We require a much more restrictive version of this problem. In particular, we require a version in which all clauses have exactly three literals, no literal appears in a clause more than once, the number of copies of a clause is equal to the number of copies of its complement (defined as the negation of all its elements), and each literal appears in exactly k clauses. This more restricted version is shown to still be APX-hard in Appendix B.

Suppose we have an instance of the aforementioned version of Max All-Equal 3SAT with variables $t_1, \ldots, t_\ell$ and clauses $C_1, \ldots, C_{2m}$. Let $\mathcal{L} = \{t_1, \ldots, t_\ell, \bar{t}_1, \ldots, \bar{t}_\ell\}$ be the set of literals and $\mathcal{C} = \{C_1, \ldots, C_{2m}\}$ be the multiset of clauses. Consider the graph $G = (V, E)$, with $V = V_0 \sqcup V_1 \sqcup V_2$, where

$$V_0 = \{v^i : i \in [N_v]\},$$
$$V_1 = \{t^i : t \in \mathcal{L}, i \in [N_t]\},$$
$$V_2 = \{C^i : C \in \mathcal{C}, i \in [N_c]\},$$

and $E = V^{(2)} \setminus (\overline{E}_1 \cup \overline{E}_2)$, where

$$\overline{E}_1 = \{(t^i, \bar{t}^j) : t \in \mathcal{L}, i, j \in [N_t]\}\},$$
$$\overline{E}_2 = \{(t^i, C^j) : t \in C, C \in \mathcal{C}, i \in [N_t], j \in [N_c]\},$$

$\sqcup$ denotes disjoint union, parameters $N_v \gg N_t \gg N_c \gg m$, and $V^{(2)} := \{U \subset V : |U| = 2\}$.

For simplicity, in the following description we assume that cliques (other than V_0) in the original graph generally embed together as one collection of nearby points, so we can treat them as single objects in the embedding. In Appendix B, this intuition is rigorously justified.

The clique on vertices V_0 serves as an "anchor" that forces all other vertices to be almost exactly at the correct distance from its center. Without loss of generality, assume this anchor clique is centered at 0. In this graph, the cliques corresponding to literals and clauses, given by $\{t^i\}_{i\in[N_t]}$ and $\{C^i\}_{i\in[N_C]}$ respectively, are all at distance one from the anchor clique. Literal cliques are at distance one from each other, except negations of each other, which are at distance two. Clause cliques are distance two from the literal cliques corresponding to literals in the clause and distance one from literal cliques corresponding to literals not in the clause. Clause cliques are all distance one from each other. The main idea of the reduction is that the location of the center of the anchor clique at 0 forces each literal to roughly be at either -1 or $+1$, and the distance between negations forces negations to be on opposite sides, i.e., $\mathbf{x}_{t^i} \approx -\mathbf{x}_{\bar{t}^i}$. Clause cliques are also roughly at either -1 or $+1$ and the distance to literals forces clauses to be opposite the side with the majority of its literals, i.e., clause $C = \{t_1, t_2, t_3\}$ lies at $\mathbf{x}_{C^i} \approx -\chi\{\mathbf{x}_{t_1^i} + \mathbf{x}_{t_2^i} + \mathbf{x}_{t_3^i} \geq 0\}$, where χ is the indicator variable. The optimal embedding of G, i.e. the location of variable cliques at either $+1$ or -1, corresponds to an optimal assignment for the Max All-Equal 3SAT instance.

Remark 1 (Comparison to (Cayton & Dasgupta, 2006))**.** As mentioned in the Introduction, the reduction here is significantly more involved than the hardness proof for a related problem in (Cayton & Dasgupta, 2006). At a high level, the key difference is that in (Cayton & Dasgupta, 2006) they were able to use a large number of distance-zero vertices to create a simple structure around the origin. This is no longer possible in our setting (in particular, with bounded diameter graph metrics), which results in graph layouts with much less structure. For this reason, we require a graph that exhibits as much structure as possible. To this end, a reduction from Max All-Equal 3SAT using both literals and clauses in the graph is a much more suitable technique than a reduction from NAE 3SAT using only literals. In fact, it is not at all obvious that the same approach in (Cayton & Dasgupta, 2006), applied to unweighted graphs, would lead to a computationally hard instance.

4. Approximation Algorithm

In this section, we formally describe an approximation algorithm using tools from the Dense CSP literature, and prove theoretical guarantees for the algorithm.

4.1. Preliminaries: Greedy Algorithms for Max-CSP

A long line of work studies the feasibility of solving the Max-CSP problem under various related pseudorandomness and density assumptions. In our case, an algorithm with mild dependence on the alphabet size is extremely important. A very simple greedy approach, proposed and analyzed by Mathieu and Schudy (Mathieu & Schudy, 2008; Schudy, 2012) (see also (Yaroslavtsev, 2014)), satisfies this requirement.

Theorem 3 ((Mathieu & Schudy, 2008; Schudy, 2012))**.** *Suppose that Σ is a finite alphabet, $n \geq 1$ is a posi-*

Algorithm 1 Greedy Algorithm for Dense CSPs (Mathieu & Schudy, 2008; Schudy, 2012)

function GreedyCSP($\Sigma, n, t_0, \{f_{ij}\}$)
  Shuffle the order of variables $x_1, \ldots, x_n$ by a random permutation.
  for all assignments $x_1, \ldots, x_{t_0} \in \Sigma^{t_0}$ **do**
    for $(t_0+1) \le i \le n$ **do**
      Choose $x_i \in \Sigma$ to maximize

$$\sum_{j<i} f_{ji}(x_j, x_i)$$

    end for
    Record x and objective value $\sum_{i \neq j} f_{ij}(x_i, x_j)$.
  end for
  Return the assignment x found with maximum objective value.
end function

tive integer, and for every $i, j \in \binom{n}{2}$ we have a function $f_{ij} : \Sigma \times \Sigma \to [-M, M]$. Then for any $\epsilon > 0$, Algorithm GREEDYCSP *with $t_0 = O(1/\epsilon^2)$ runs in time $n^2|\Sigma|^{O(1/\epsilon^2)}$ and returns $x_1, \ldots, x_n \in \Sigma$ such that*

$$\mathbb{E} \sum_{i \neq j} f_{ij}(x_i, x_j) \ge \sum_{i \neq j} f_{ij}(x_i^*, x_j^*) - \epsilon M n^2$$

for any $x_1^*, \ldots, x_n^* \in \Sigma$, where $\mathbb{E}$ denotes the expectation over the randomness of the algorithm.

In comparison, we note that computing the maximizer using brute force would run in time $|\Sigma|^n$, i.e. exponentially slower in terms of n. This guarantee is stated in expectation but, if desired, can be converted to a high probability guarantee by using Markov's inequality and repeating the algorithm multiple times (as in Remark 2). We use GREEDYCSP to solve a minimization problem instead of maximization, which corresponds to negating all of the functions f_{ij}.

4.2. Discretization Argument

Lemma 6. *For $c, R > 0$, the function $x \mapsto (x/c - 1)^2$ is $\frac{2}{c}\max(1, R/c)$-Lipschitz on the interval $[0, R]$.*

Proof. Because the derivative of the function is $\frac{2}{c}(x/c - 1)$ and $\left|\frac{2}{c}(x/c-1)\right| \le \frac{2}{c}\max(1, R/c)$ on $[0, R]$, the result follows from the mean value theorem. □

Lemma 7. *For $c, R > 0$ and $y \in \mathbb{R}^r$ with $\|y\| \le R$, the function $x \mapsto (\|x-y\|/c-1)^2$ is $\frac{2}{c}\max(1, 2R/c)$-Lipschitz on $B_R = \{x : \|x\| \le R\}$.*

Proof. Because the function $\|x - y\|$ is 1-Lipschitz and $\|x - y\| \le \|x\| + \|y\| \le 2R$ by the triangle inequality, the result follows from Lemma 6 and the fact that a composition of Lipschitz functions is Lipschitz. □

Lemma 8. *Let $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$ be arbitrary vectors such that $\|\mathbf{x}_i\| \le R$ for all i and $\epsilon > 0$ be arbitrary. Define S_ϵ to be an ϵ-net of B_R as in Lemma 3, so $|S_\epsilon| \le (3R/\epsilon)^r$. For any input metric over $[n]$ with $\min_{i,j \in [n]} d(i, j) = 1$, there exists $\mathbf{x}'_1, \ldots, \mathbf{x}'_n \in S_\epsilon$ such that*

$$E(\mathbf{x}'_1, \ldots, \mathbf{x}'_n) \le E(\mathbf{x}_1, \ldots, \mathbf{x}_n) + 4\epsilon R n^2$$

where E is (1) defined with respect to an arbitrary graph with n vertices.

Proof. By Lemma 7 the energy $E(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is the sum of $\binom{n}{2} \le n^2/2$ many terms, which, for each i and j, are individually $4R$-Lipschitz in $\mathbf{x}_i$ and $\mathbf{x}_j$. Therefore, defining $\mathbf{x}'_i$ to be the closest point in S_ϵ for all i gives the desired result. □

4.3. Approximation Algorithm

Algorithm 2 Approximation Algorithm KKSCHEME

function KKSCHEME($\epsilon_1, \epsilon_2, R$):
  Build an ϵ_1-net S_{ϵ_1} of $B_R = \{x : \|x\| \le R\} \subset \mathbb{R}^r$ as in Lemma 3.
  Apply the GREEDYCSP algorithm of Theorem 3 with $\epsilon = \epsilon_2$ to approximately minimize $E(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ over $\mathbf{x}_1, \ldots, \mathbf{x}_n \in S_{\epsilon_1}^n$.
  Return $\mathbf{x}_1, \ldots, \mathbf{x}_n$.
end function

Theorem 4 (Formal Statement of Theorem 2)**.** *Let $R > \epsilon > 0$ be arbitrary. For any input metric over $[n]$ with $\min_{i,j \in [n]} d(i, j) = 1$, Algorithm* KKSCHEME *with $\epsilon_1 = O(\epsilon/R)$ and $\epsilon_2 = O(\epsilon/R^2)$ runs in time $n^2(R/\epsilon)^{O(rR^4/\epsilon^2)}$ and outputs $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$ with $\|\mathbf{x}_i\| \le R$ such that*

$$\mathbb{E}\left[E(\mathbf{x}_1, \ldots, \mathbf{x}_n)\right] \le E(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*) + \epsilon n^2$$

for any $\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*$ with $\|\mathbf{x}_i^*\| \le R$ for all i, where $\mathbb{E}$ is the expectation over the randomness of the algorithm.

Proof. By combining Lemma 8 with Theorem 3 (used as a minimization instead of maximization algorithm), the output $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of KKSCHEME satisfies

$$E(\mathbf{x}_1, \ldots, \mathbf{x}_n) \le E(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*) + 4\epsilon_1 R n^2 + \epsilon_2 R^2 n^2$$

and runs in time $n^2 2^{O(1/\epsilon_2^2) r \log(3R/\epsilon_1)}$. Taking $\epsilon_2 = O(\epsilon/R^2)$ and $\epsilon_1 = O(\epsilon/R)$ gives the desired result. □

Remark 2. The runtime can be improved to $n^2 + (R/\epsilon)^{O(dR^4/\epsilon^2)}$ using a slightly more complex greedy CSP algorithm (Mathieu & Schudy, 2008). Also, by the usual argument, a high probability guarantee can be derived by repeating the algorithm $O(\log(2/\delta))$ times, where $\delta > 0$ is the desired failure probability.

4.4. Extension to Vertex-Weighted Setting

In this section, we generalize the approximation algorithm to handle vertex weights. This generalization is useful if vertices have associated *importance weights*, e.g. each vertex represents a different number of people, and larger/more important vertices should be embedded more accurately. Given a probability measure μ over $[n]$, the weighted Kamada-Kawai objective is

$$E^{\mu}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = n^2 \sum_{i<j} \mu(i)\mu(j) \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{d(i,j)} - 1 \right)^2. \tag{2}$$

Note that when μ is the uniform measure on $[n]$, this reduces to (1).

Theorem 5. *Let $R > \epsilon > 0$ be arbitrary. Algorithm* KKSCHEME *with $\epsilon_1 = O(\epsilon/R)$ and $\epsilon_2 = O(\epsilon/R^2)$ runs in time $n^{O(rR^4 \log(R/\epsilon)/\epsilon^2)}$ and outputs $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$ with $\|\mathbf{x}_i\| \leq R$ such that*

$$\mathbb{E}\left[E(\mathbf{x}_1, \ldots, \mathbf{x}_n)\right] \leq E(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*) + \epsilon n^2$$

for any $\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*$ with $\|\mathbf{x}_i^*\| \leq R$ for all i, where $\mathbb{E}$ is the expectation over the randomness of the algorithm.

Proof. The proof is the same as Theorem 4, except that we require a different dense CSP algorithm. More precisely, we can directly verify that the discretization Lemma, Lemma 8, holds with the same guarantee for the weighted Kamada-Kawai objective. This reduces the problem to approximating a dense CSP with vertex weights, for which we use Theorem 6. □

The following Theorem formally describes the guarantee we use for approximately optimizing dense CSPs with vertex/variable weights. This result can be proved by slightly modifying the algorithm and analysis in (Yoshida & Zhou, 2014). For completeness, we provide a proof in Appendix C.

Theorem 6 (Proved in Appendix C)**.** *Suppose that Σ is a finite alphabet, $n \geq 1$ is a positive integer, and for every $i, j \in \binom{n}{2}$ we have a function $f_{ij} : \Sigma \times \Sigma \to [-M, M]$. Then for any $\epsilon > 0$, there exists an algorithm which runs in time $n^{O(\log|\Sigma|/\epsilon^2)}$ and returns $x_1, \ldots, x_n \in \Sigma$ such that*

$$\mathbb{E}\left[\mathbb{E}_{i,j\sim\mu} f_{ij}(x_i, x_j)\right] \geq \mathbb{E}_{i,j\sim\mu} f_{ij}(x_i^*, x_j^*) - \epsilon M$$

for any $x_1^*, \ldots, x_n^* \in \Sigma$, where the outer $\mathbb{E}$ denotes the expectation over the randomness of the algorithm.

5. Experiments

We implemented the GREEDYCSP-based algorithm described above as well as a standard gradient descent approach to minimizing the Kamada-Kawai objective. In this section we compare the behavior of these algorithms in a few interesting instances.

In addition to gradient descent, a couple of other local search heuristics are popular for minimizing the Kamada-Kawai objective: 1) the original algorithm proposed by Kamada and Kawai (Kamada et al., 1989), which updates single points at a time using a Newton-Raphson scheme, and 2) a variational approach known as *majorization*, which optimizes a sequence of upper bounds on the KK objective (De Leeuw et al., 1977; Gansner et al., 2004), where each step reduces to solving a Laplacian system. The recent work of (Zheng et al., 2018) compared these local search heuristics and argued that (stochastic) gradient descent, proposed in the early work of (Kruskal, 1964a), is one of the best performing methods in practice. For this reason, we focus on comparing with gradient descent.

Some Graph Drawing Examples. In Figure 1 we show the result of embedding a random Watts-Strogatz "small world" graph (Watts & Strogatz, 1998), a model of random graph intended to reflect some properties of real world networks. In Figure 2 we show an embedding of the "3elt" graph from (Diekmann & Preis); in this case, it's interesting that all of the methods optimizing (1) seem to find the same solution, except Greedy suffers a small loss due to discretization. This suggests that this solution may be the global optimum.

Note that in all figures, the MDS/Kamada-Kawai objective value achieved (normalized by $1/n^2$, where n is the number of vertices) is included in the subtitle of each plot. For comparison, in the bottom right of each Figure we display the standard spectral embedding given by embedding each vertex according to the entries of the bottom two nontrivial eigenvectors of the graph Laplacian.

Experiment with restarts. The algorithm we propose in Theorem 4 is randomized, which leaves open the possibility that better results are obtained by running the algorithm multiple times and taking the best result. In Figure 3, we show the result of embedding a well-known social network graph, the Davis Southern Women Network (Davis et al., 2009), by running all methods 10 times and taking the result with best objective value. This graph has a total of 32 nodes and records the attendance of 18 Southern women at 14 social events during the 1930s. To compare with the minimum, the average objective value achieved in the run is 0.0588, 0.0498, and 0.0515 for Greedy, Greedy and Grad, and Grad respectively so all methods did improved slightly by running multiple times. Finally, we note that running gradient descent with 30 restarts (as opposed to 10) improved its best score to 0.0478, essentially the same as the Greedy and Grad result.

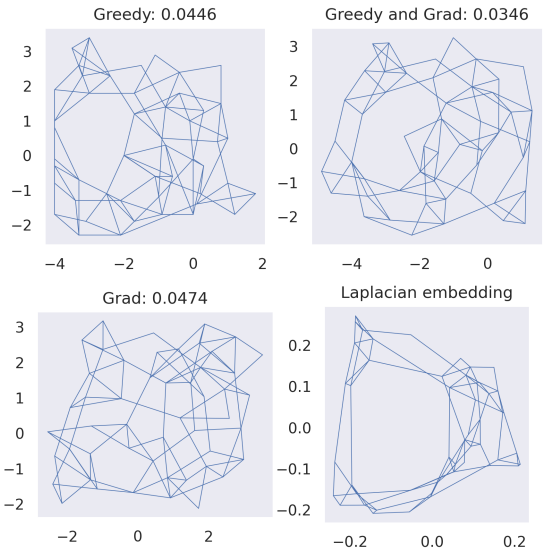

Figure 1. Embeddings of Watts Strogatz graph on 50 nodes with graph parameters $K = 4$ and $\beta = 0.3$ and $t_0 = 3$ for GREEDYCSP.

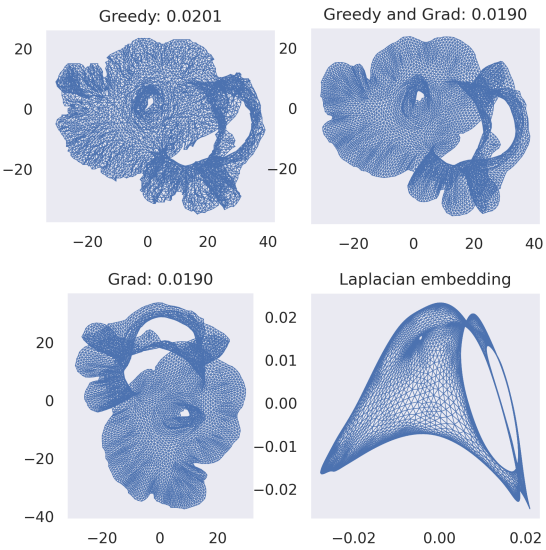

Figure 2. 3elt graph from AG Monien collection (Diekmann & Preis); GREEDYCSP run with parameter $t_0 = 2$.

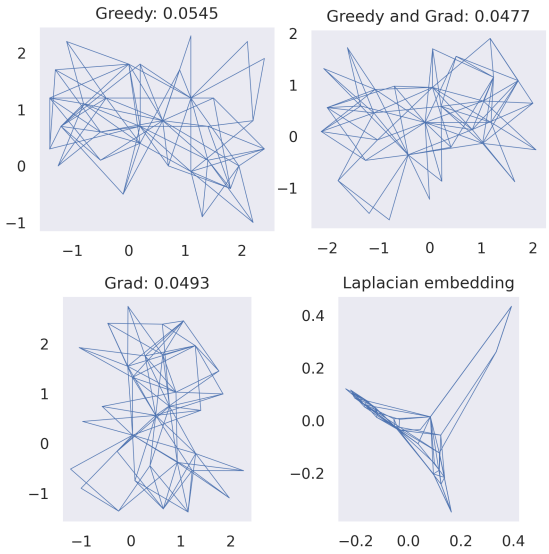

Figure 3. Embedding of Davis Southern Women Network graph. The top left figure was generated using GREEDYCSP with $t_0 = 3$.

Runtime	Greedy	Grad	Laplacian
Davis	5.6 s	4 s	4 ms
Watts-Strogatz	453 s	4 s	20 ms

Table 1. Runtimes for methods with parameters used in figures.

Community Detection Experiment. A lot of the recent interest in force-directed graph drawing algorithms has been in their ability to discover interesting latent structure in data and with a view towards applications like non-linear dimensionality reduction. As a test of this concept on synthetic data, we tested the algorithms on a celebrated model of latent community structure in graphs, the stochastic block model. The results are shown in Figure 4, along with the results of a standard spectral embedding using the bottom two nontrivial eigenvectors of the Laplacian. We did not draw the edges in this case as they make the Figure difficult to read; more importantly, the location of points in the embedding show that nontrivial community structure was recovered; for example, the green and blue communities are roughly linearly separable in all of the embeddings. Note that the spectral embedding approach admits strong provable guarantees for community recovery (see the survey (Abbe, 2017)), and so the interesting thing to observe here is that the force-directed drawing methods also recover nontrivial information about the latent structure.

Implementation details. All experiments were performed on a standard Kaggle GPU Kernel with a V80 GPU. Gradient descent was run with learning rate 0.005 for 4000 steps on all instances. We seeded the RNG with zero before each simulation for reproducibility. For the greedy method,

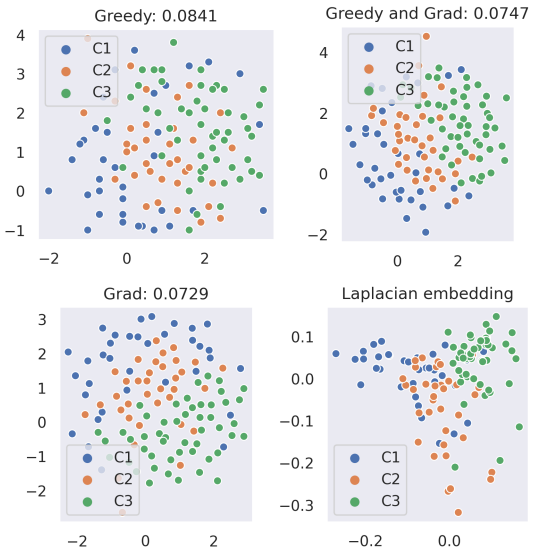

Figure 4. Embeddings of a 3-community Stochastic Block Model (SBM) with connection probabilities $\begin{bmatrix} 0.09 & 0.03 & 0.02 \\ 0.03 & 0.15 & 0.04 \\ 0.02 & 0.04 & 0.1 \end{bmatrix}$ and community sizes $35, 35, 50$. Colors correspond to latent community assignments. The top left is constructed using the GREEDYCSP algorithm with $t_0 = 3$. For this experiment only, we used the degree-normalized Laplacian since it is generally preferred in the context of the SBM.

we eliminated the rotation and translation degrees of freedom when implementing the initial brute force step; the parameter R was set to 2.5 for the Davis experiment, and set to 4 for all others — informally, the tuning rule for this parameter is to increase its value until the plot does not hit the boundary of the region. We compare runtimes in Table 5; the runtime for Greedy in Watts-Strogatz is much larger due to the larger value of n and of R used; the latter roughly corresponds to the larger diameter of the underlying graph (cf. Lemma 4).

6. Conclusions

Our theory and experimental results suggest the following natural question: does gradient descent, with enough random restarts, have a similar provable guarantee to Theorem 3? As noted in our experiments and in the experiments of (Zheng et al., 2018), gradient-based optimization often seems to find high quality (albeit not global) minima of the Kamada-Kawai objective, even though the loss is highly non-convex. In fact, combining our analysis with a different theorem from (Schudy, 2012) proves that running a variant of GREEDYCSP without the initial brute force step (i.e. with $t_0 = 0$), achieves an additive $O(\epsilon n^2)$ approximation if we repeat the algorithm $2^{2^{1/\epsilon^2}}$ many times. A similar guarantee for gradient descent, a different sort of greedy procedure, sounds plausible.

Acknowledgements Frederic Koehler was supported in part by NSF CAREER Award CCF-1453261, NSF Large CCF-1565235, Ankur Moitra's ONR Young Investigator Award, and E. Mossel's Vannevar Bush Fellowship ONR-N00014-20-1-2826. The work of J. Urschel was supported in part by ONR Research Contract N00014-17-1-2177. The authors would like to thank Michel Goemans for valuable conversations on the subject. The authors are grateful to Louisa Thomas for greatly improving the style of presentation.

References

Abbe, E. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

Arora, S., Hu, W., and Kothari, P. K. An analysis of the t-sne algorithm for data visualization. *arXiv preprint arXiv:1803.01768*, 2018.

Badoiu, M., Dhamdhere, K., Gupta, A., Rabinovich, Y., Räcke, H., Ravi, R., and Sidiropoulos, A. Approximation algorithms for low-distortion embeddings into low-dimensional spaces. In *SODA*, volume 5, pp. 119–128. Citeseer, 2005.

Barak, B., Raghavendra, P., and Steurer, D. Rounding semidefinite programming hierarchies via global correlation. In *2011 ieee 52nd annual symposium on foundations of computer science*, pp. 472–481. IEEE, 2011.

Battista, G. D., Eades, P., Tamassia, R., and Tollis, I. G. *Graph drawing: algorithms for the visualization of graphs*. Prentice Hall PTR, 1998.

Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Borg, I. and Groenen, P. J. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005.

Cayton, L. and Dasgupta, S. Robust euclidean embedding. In *Proceedings of the 23rd international conference on machine learning*, pp. 169–176, 2006.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Davis, A., Gardner, B. B., and Gardner, M. R. *Deep South: A social anthropological study of caste and class*. Univ of South Carolina Press, 2009.

De Leeuw, J. Convergence of the majorization method for multidimensional scaling. *Journal of classification*, 5(2): 163–180, 1988.

De Leeuw, J., Barra, I. J., Brodeau, F., Romier, G., Van Cutsem, B., et al. Applications of convex analysis to multidimensional scaling. In *Recent Developments in Statistics*. Citeseer, 1977.

Deza, M. M. and Laurent, M. *Geometry of cuts and metrics*, volume 15. Springer, 2009.

Diekmann, R. and Preis, R. Ag-monien graph collectionn. https://www.cise.ufl.edu/research/sparse/mat/AG-Monien/README.txt. Accessed: 2020-02-01.

Dodds, P. S., Muhamad, R., and Watts, D. J. An experimental study of search in global social networks. *science*, 301 (5634):827–829, 2003.

Ellson, J., Gansner, E., Koutsofios, L., North, S. C., and Woodhull, G. Graphviz—open source graph drawing tools. In *International Symposium on Graph Drawing*, pp. 483–484. Springer, 2001.

Feige, U. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45(4):634–652, 1998. doi: 10.1145/285055.285059. URL https://doi.org/10.1145/285055.285059.

Filho, I. T. F. A. *Characterizing Boolean Satisfiability Variants*. PhD thesis, Massachusetts Institute of Technology, 2019.

Gansner, E. R., Koren, Y., and North, S. Graph drawing by stress majorization. In *International Symposium on Graph Drawing*, pp. 239–250. Springer, 2004.

Kamada, T., Kawai, S., et al. An algorithm for drawing general undirected graphs. *Information processing letters*, 31(1):7–15, 1989.

Khanna, S., Sudan, M., Trevisan, L., and Williamson, D. P. The approximability of constraint satisfaction problems. *SIAM Journal on Computing*, 30(6):1863–1920, 2001.

Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964a.

Kruskal, J. B. Nonmetric multidimensional scaling: a numerical method. *Psychometrika*, 29(2):115–129, 1964b.

Kruskal, J. B. *Multidimensional scaling*. Number 11. Sage, 1978.

Linderman, G. C. and Steinerberger, S. Clustering with t-sne, provably. *SIAM Journal on Mathematics of Data Science*, 1(2):313–332, 2019.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov): 2579–2605, 2008.

Mathieu, C. and Schudy, W. Yet another algorithm for dense max cut: go greedy. In *SODA*, pp. 176–182, 2008.

McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Montanari, A. Estimating random variables from random sparse observations. *European Transactions on Telecommunications*, 19(4):385–403, 2008.

Naor, A. An introduction to the ribe program. *Japanese Journal of Mathematics*, 7(2):167–233, 2012.

Papadimitriou, C. H. and Yannakakis, M. Optimization, approximation, and complexity classes. *J. Comput. Syst. Sci.*, 43(3):425–440, 1991. doi: 10.1016/0022-0000(91)90023-X. URL https://doi.org/10.1016/0022-0000(91)90023-X.

Raghavendra, P. and Tan, N. Approximating csps with global cardinality constraints using sdp hierarchies. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pp. 373–387. SIAM, 2012.

Saxe, J. B. Embeddability of weighted graphs in k-space is strongly np-hard. In *Proc. of 17th Allerton Conference in Communications, Control and Computing, Monticello, IL*, pp. 480–489, 1979.

Schudy, W. *Approximation Schemes for Inferring Rankings and Clusterings from Pairwise Data*. PhD thesis, Brown University, 2012.

Tutte, W. T. How to draw a graph. *Proceedings of the London Mathematical Society*, 3(1):743–767, 1963.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Watts, D. J. and Strogatz, S. H. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

Yaroslavtsev, G. Going for speed: Sublinear algorithms for dense r-csps. *arXiv preprint arXiv:1407.7887*, 2014.

Yoshida, Y. and Zhou, Y. Approximation schemes via sherali-adams hierarchy for dense constraint satisfaction problems and assignment problems. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pp. 423–438, 2014.

Zheng, J. X., Pawar, S., and Goodman, D. F. Graph drawing by stochastic gradient descent. *IEEE transactions on visualization and computer graphics*, 25(9):2738–2748, 2018.