Metric Violation Distance: Hardness and Approximation

Chenglin Fan* Benjamin Raichel† Gregory Van Buskirk‡

Abstract

Metric data plays an important role in various settings, for example, in metric-based indexing, clustering, classification, and approximation algorithms in general. Due to measurement error, noise, or an inability to completely gather all the data, a collection of distances may not satisfy the basic metric requirements, most notably the triangle inequality. In this paper we initiate the study of the *metric violation distance* problem: given a set of pairwise distances, modify the minimum number of distances such that the resulting set forms a metric. Three variants of the problem are considered, based on whether distances are allowed to only decrease, only increase, or the general case which allows both decreases and increases. We show that while the decrease only variant is polynomial time solvable, the increase only and general variants are NP-Complete, and moreover cannot in polynomial time be approximated to any ratio better than the minimum vertex cover problem. We then provide approximation algorithms for the increase only and general variants of the problem, by proving interesting necessary and sufficient conditions on the optimal solution, which are used to approximately reduce to a purely combinatorial problem for which we provide matching asymptotic upper and lower bounds.

1 Introduction

Suppose you are given a collection of data points, along with a corresponding set of distances (i.e. dissimilarity scores) between every pair of points. The ability to perform various computational tasks over this collection of data points depends highly on what structural properties these distances obey. Perhaps the most often considered and desired are the basic metric requirements, most notably the triangle inequality (as the other metric requirements are often trivially satisfied). Metric data plays a critical role in various areas such as metric-based indexing, clustering, classification, and approximation algorithms in general. Moreover, given our points come from a metric space, we can potentially obtain additional properties by applying tools such as metric embeddings. As a simple example of the power of metric properties, consider the famous NP-Complete traveling salesperson problem (TSP). For general positively weighted graphs, it is a standard exercise to show TSP cannot be approximated within *any* constant factor unless P=NP. On the other hand if the graph satisfies the triangle inequality, Christofides' algorithm gives a 1.5-approximation [Chr76].

Here we consider the *metric violation distance* problem, denoted MVD, where given a collection of distances between data points (forming a semi-metric), we seek the smallest sized set of distance values that can be (arbitrarily) modified to produce a metric space overall. One can consider the resulting metric as the nearest neighbor from the space of all metrics, and thus one interpretation of the MVD problem is as a measurement of how close the input is to representing a metric space. A solution to this problem is thus desirable, since when the distance is small, one could potentially use the nearest metric as a proxy for the original set of distances, unlocking all the above benefits of metric spaces. Alternatively, the MVD problem arises naturally in the following setting. Suppose you are collecting experimental data by measuring distances between a collection of objects, where you know the distances should form a metric space. The collected data might be non-metric for a variety of reasons, such as measurement error, or incomplete or corrupted data entries. In this setting the MVD problem can thus be used as a means to recover the true underlying data set.

In this paper we initiate the study of the MVD problem. We prove the problem is APX-hard, and provide approximation algorithms for different variants.

Related Work. Brickell *et al.* [BDST08] studied the metric nearness problem. Similar to MVD, the input is a semi-metric (i.e. triangle inequalities may be violated) and the goal is to find the closest metric space. The difference is that for MVD a closest metric space is defined by minimizing the number of changed

*Department of Computer Science; University of Texas at Dallas; cxf160130@utdallas.edu. Work on this paper was partially supported by NSF CRII Award 1566137.

†Department of Computer Science; University of Texas at Dallas; benjamin.raichel@utdallas.edu; http://utdallas.edu/~benjamin.raichel. Work on this paper was partially supported by NSF CRII Award 1566137.

‡Department of Computer Science; University of Texas at Dallas; greg.vanbuskirk@utdallas.edu. Work on this paper was partially supported by NSF CRII Award 1566137.

values (irrespective of how much they are changed), and in their case it is de ned by minimizing the sum of the changes in value (irrespective of how many are changed). In other words, metric nearness is the linear programming relaxation of MVD, and thus is not NP-hard like MVD. So while metric nearness is semantically similar, the challenges and approach are di erent, and thus [BDST08] focus on other aspects such as varying the norm of the objective and experimental results.

The MVD problem also bears a clear resemblance to metric embeddings, though one level removed. In a standard metric embedding problem there are two collections of metric spaces of interest, call them S (for source) and T (for target). Given a metric space $X \in S$ the goal is then to injectively map the points from X into the metric space $Y \in T$ which preserves distances as best as possible, when measured by the distortion (roughly the maximum any distance is scaled). Typically the collection T has some desirable structural property that in general is lacking from metrics in S, and thus metric embeddings are a tool to gain these structural bene ts. For example, the famous result of Bourgain [Bou85] shows that any arbitrary n-point metric embeds into $\ell_2^{O(\log n)}$ with $O(\log n)$ distortion, and more generally into $\ell_p^{O(\log^2 n)}$ for any p, due to Linial *et al.* [LLR95]. (As metric embeddings are too broad to fully cover here, we refer the reader to the surveys [IM04, Mat13].) Thus MVD can be viewed as a type of \embedding" problem, where S is the collection of all semi-metrics and T is the collection of all metrics.

The larger di erence between metric embedding and MVD, is that for metric embedding the typical goal is to minimize distortion, whereas in our case the goal is to minimize the number of (arbitrarily) changed distances. Thus our work is more akin to isometrically embedding with outliers. Recently Sidiropoulos *et al.* [SWW17] considered isometrically embedding metrics into ultrametrics, trees, and Euclidean space, while minimizing the number of outlier points. Despite their input starting out as a metric, the problems they consider similarly involve satisfying inequalities with a small number of distances (for example, ultrametrics replace the sum with max in the triangle inequality). The larger di erence is that since their notion of outliers is point based, when a violated inequality is identi ed the points can simply be thrown out, whereas in our case the values need to be corrected, and it is not possible to locally determine how to do so (see Figure 1.1, discussed in detail below). Moving towards distance based notions of outliers, the problem of embeddings with *slack* was previously considered. Rather than isometrically embedding, the goal moves back to nding minimum distortion embeddings, but subject to allowing an ε fraction of the distances to be arbitrarily distorted. For example, Abraham *et al.* [ABC+05, CDG+09] showed that any metric space can be embedded into $\ell_p^{O(\log^2(1/\varepsilon))}$ for any $p \geq 1$ with $O(\log(1/\varepsilon))$ distortion, i.e. a slack version of Bourgain's theorem.

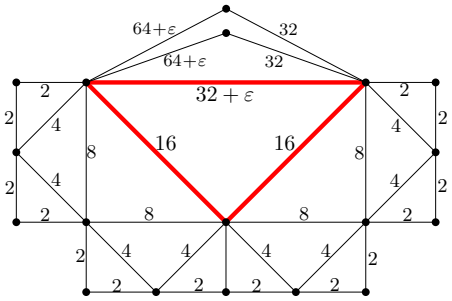

Figure 1.1: A single red thickened violated triangle. Increasing a 16 edge causes a chain of violations, which more generally can be logarithmic in size. Many edges are omitted for simplicity.

There are a number of other more loosely related problems. In the matrix completion problem, from the machine learning community, one is given a distance matrix with many missing entries, and the goal is to ll in the missing entries such that the resulting matrix has as low a rank as possible [CR12]. In the distance realization problem, from the networking community, given a distance matrix the goal is to nd a corresponding weighted graph (possibly restricted to be a tree) which minimizes the sum of edge lengths [CGGS01]. While related, these problems do not capture the speci c challenges of MVD.

Contribution and Results. From a rst glance it may not be apparent why the MVD problem is so challenging. Violated triangle inequalities can trivially be identi ed, so why not x them one at a time. The rst issue is that xing a triangle may require violating another, as shown in Figure 1.1. In this example, the larger issue is that xing the inequality requires deciding whether to increase smaller weight edges or decrease the larger one (or both), as well as deciding how much to change them. Increasing one lower weight edge by ε creates only one new violation, and thus locally appears preferable, as decreasing the large edge by ε creates two violations. Unfortunately, decreasing the lower weight edge creates a chaining e ect ultimately requiring a logarithmic number of xes. Alternatively one can take a more global approach, treating all the violated inequalities as a system of constraints and then trying to solve that system. The issue with this approach is that the objective is to minimize the number of changes rather than the sum of the changed amounts, which

is a red ag as roughly speaking this is the di erence between integer programming formulations of many NP-Complete problems and their linear programming relaxations.

Here we take a systematic approach to understanding and handling the challenges of the MVD problem. We consider three variants: 1) MVD where one is only allowed to decrease values, 2) MVD where one is only allowed to increase values, and 3) the general MVD problem. When only decreases are allowed we provide a polynomial time solution, making a connection to the all pairs shortest paths problem. For both the increase only and general cases, we show that the problem is NP-Complete, and moreover is as hard to approximate as vertex cover. We also give a polynomial time $O(OPT^{1=3})$-approximation for both cases, where OPT denotes the optimal solution size. We rst present the approximation algorithm for the increase only case, as it is conceptually simpler, and its results can be used to help understand the approach for the general case. That said, the increase only case is still quite challenging, and requires proving necessary and su cient conditions on the solution, reducing the problem to a purely combinatorial one with connections to fundamental problems such as block design. Several interesting open problems remain such as whether there is a provable hardness gap between the increase only and general versions, and for either problem whether the gap between approximation and hardness can be narrowed.

2 Preliminaries

The MVD problem can be equivalently formulated as either a problem on symmetric matrices or on weighted graphs. For the majority of the paper the graph formulation is used, however the matrix formulation is also presented as this was the original terminology used by Brickel *et al.* [BDST08] for the similar metric nearness problem, and also more naturally connects to the linear program feasibility check we ultimately use to verify our solution.

2.1 Problem Definition. Let a *dissimilarity matrix* be any square symmetric matrix, where diagonal entries are all zero, and all other entries are positive. For an $n \times n$ dissimilarity matrix M, we refer to each entry M_{ij} as the distance from point i to point j. M is said to be a *metric matrix* if for every triple of distinct indices $(i; j; k)$ it holds that $M_{ik} \leq M_{ij} + M_{jk}$, that is the distances in the matrix represent the interpoint distances of an n point metric space. A *symmetric modification* of a distance M_{ij} to a value , sets $M_{ij} = M_{ji} =$.

Problem 2.1. (Matrix Metric Violation Distance (MMVD)) *Given a dissimilarity matrix M, compute a minimum size set S of distances which can be symmetrically modified to convert M into a metric matrix.*

Alternatively, de ne a *dissimilarity graph* as any complete, undirected, and positively-weighted graph, and de ne a *metric graph* to be any dissimilarity graph which is its own metric completion.

Problem 2.2. (Graph Metric Violation Distance (GMVD)) *Given a dissimilarity graph G, compute a minimum size set S of edges whose weights can be modified to convert G into a metric graph.*

Note that GMVD and MMVD are equivalent problems and an instance of one can be trivially converted into an instance of the other. We thus freely interchange between the two in the text when needed, though outside this section we mainly defer to the graph formulation as it avoids the confusion of symmetric modi cation.

Two variants of the GMVD problem will also be considered in the paper, one where weights are only allowed to decrease, and the other where weights are only allowed to increase.

Problem 2.3. (Graph Metric Violation Decrease Distance (GMVDD)) *Given a dissimilarity graph G, compute a minimum size set S of edges whose weights can be decreased to covert G into a metric graph.*

Problem 2.4. (Graph Metric Violation Increase Distance (GMVID)) *Given a dissimilarity graph G, compute a minimum size set S of edges whose weights can be increased to covert G into a metric graph.*

2.2 Feasibility Checking. Let M be an $n \times n$ dissimilarity matrix. Let S be a set of entries from M, namely a set of pairs of distinct integers $\{i; j\}$, with $1 \leq i; j \leq n$. Here we show that checking whether there is a solution to MMVD for M which symmetrically modi es only the entries in S, is polynomial time solvable by writing the problem as an instance of linear programming feasibility. Speci cally, for each entry M_{ij} we de ne a new matrix entry $_{ij} = M_{ij} + x_{ij}$, where x_{ij} is a variable representing the deviation from the original entry. If there is a solution for the MMVD problem that only symmetric modi es entries from S, then the $_{ij}$'s must satisfy all triangle inequalities, and for all $\{i; j\} \in S$ we must have $x_{ij} = x_{ji} = 0$. Thus the problem is equivalent to the feasibility of the following linear program.

$$
\begin{aligned}
_{ij} &= M_{ij} + x_{ij} > 0 && \forall \text{ distinct pairs } \{i;j\} \\
_{ik} &\leq \ _{ij} + \ _{jk} && \forall \text{ distinct triples } \{i;j;k\} \\
x_{ij} &= x_{ji} = 0 && \forall \{i;j\} \in S \\
x_{ij} &= x_{ji} && \forall \{i;j\} \in S
\end{aligned}
$$

Later in the paper we consider variants of MMVD where entries are only allowed to be increased. Note that the above linear program can trivially be modi ed to handle this case. Namely modify the last constraint to be $x_{ij} = x_{ji} \geq 0 \ \ \forall \{i;j\} \in S$.

2.3 Notation and an Observation. We now list some notation which will be used for our approximation algorithms. Given a dissimilarity graph $G = (V;E)$, a subgraph $C = (V';E')$ is called a k-cycle if $|V'| = |E'| = k$ and the subgraph is connected with every vertex having degree exactly 2. We often overload this notation and use C to denote either the cyclically ordered list of vertices or edges from this subgraph. Given a k-cycle in G, if the weight of a single edge is strictly larger than sum of the weights of the other edges in the cycle, we say it is an *unbalanced k-cycle*. For a given unbalanced k-cycle, call the largest edge of the cycle the *top edge* and the other edges of the cycle the non-top edges. Call any edge from G connecting two vertices which are non-adjacent in a given cycle, a *chord* of that cycle.

We de ne three notions of covering unbalanced cycles, which correspond to our three problem variants. Speci cally, let $\mathbb{C}$ be any collection of unbalanced cycles from a dissimilarity graph $G = (V;E)$. We say an edge subset $F \subseteq E$ is a (i) *regular cover*, (ii) *non-top cover*, or (iii) *top cover* of $\mathbb{C}$, if F contains at least one (i) edge, (ii) non-top edge, or (iii) top edge of every unbalanced cycle in $\mathbb{C}$. In particular, if $\mathbb{C}$ is the set of *all* unbalanced cycles in G then we say F (i) regular covers, (ii) non-top covers, or (iii) top covers G, respectively.

The following simple lemma implies that all unbalanced cycles must be regular covered for GMVD, non-top covered for GMVID, and top covered for GMVDD.

LEMMA 2.1. *For any dissimilarity graph G, if there exists an unbalanced k-cycle for $k \geq 4$, then there exists an unbalanced 3-cycle, i.e. an unsatisfied triangle inequality.*

Proof. Let k be the smallest value such that there is an unbalanced k-cycle, and suppose for contradiction that $k \geq 4$. Let C be an unbalanced k-cycle, and let x and y be any two non-top edges of C which are adjacent in C. These two edge form a unique triangle in the complete input graph G with some third edge z (note as $k \geq 4$, z is not in C). Let C' be the cycle obtained from C by removing x and y and adding the edge z. Since there are no unbalanced 3-cycles, $w(z) \leq w(x) + w(y)$. This implies C' is an unbalanced $k-1$ cycle, which is a contradiction with the de nition of k.

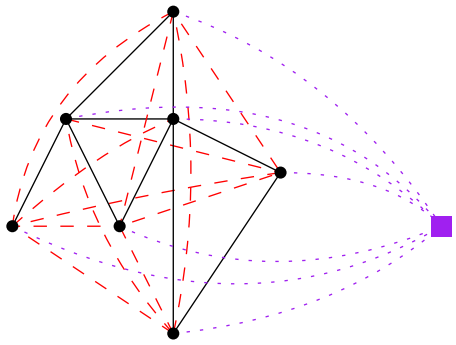

Figure 3.1: Vertex cover reduction: E edges are solid black, red are dashed, and purple are dotted.

3 Metric Violation Distance Complexity

In this section we prove the following decision version of GMVD (and hence MMVD) is NP-hard, and moreover the optimization version is APX-hard. The hardness of GMVID will then follow by essentially the same proof. Finally, we show that conversely GMVDD is polynomial time solvable.

PROBLEM 3.1. *Given a dissimilarity graph G, can G be converted into a metric graph by modifying the weights of at most k edges.*

The proof of hardness will be by reduction from vertex cover.

PROBLEM 3.2. (VERTEX COVER) *Given an undirected graph $G = (V;E)$, is there a subset $V' \subseteq V$ of size $|V'| \leq k$ such that each edge $e \in E$ is incident to at least one vertex in V'.*

As our reduction will be approximation preserving, it actually implies GMVD is APX-hard, as minimum vertex cover is APX-hard. Moreover, assuming the Unique Games Conjecture, minimum vertex cover is hard to approximate within $2 - "$ for any $" > 0$ [KR08], and thus the same is true for GMVD.

THEOREM 3.1. *Problem 3.1 is NP-Complete. Moreover, GMVD is APX-hard, and assuming the Unique Games Conjecture, is hard to approximate within a factor of $2 - "$ for any $" > 0$.*

Proof. Let $G = (V;E)$ be an instance of Problem 3.2. We construct a corresponding instance $H = (V \cup$

$\{v_0\}; E'$) of Problem 3.1 with weight function $w : E' \to \mathbb{R}^+$, where v_0 is a newly added \apex" vertex. Note by de nition H is a complete graph. Partition the edges in E' into three named groups. The set E of edges from G, the set P consisting of all edges adjacent to v_0 called *purple edges*, and the set R of all other edges (i.e. edges in the complement of G) called *red edges*. See Figure 3.1 for an illustration. The idea is to set edge weights such that modifying the weight of a given purple edge to satisfy the triangle inequalities it is involved in, corresponds to selecting the non-apex endpoint to cover its adjacent edges from E. Speci cally, for $e \in E$ set $w(e) = 2 +$, for $e \in P$ set $w(e) = 1$, and for $e \in R$ set $w(e) = 2$, where is a su ciently small constant.

It is easy to verify that any unsatis ed triangle inequality in H must involve both an edge from E and its two adjacent purple edges, that is it must be of the form $w(x; y) = 2 + > 1 + 1 = w(v_0; x) + w(v_0; y)$ where the edge $\{x; y\} \in E$. Thus there is a one-to-one correspondence between E and the unsatis ed triangle inequalities. So let F be the set of modi ed edges from a solution to Problem 3.1 on H. We now show how to construct a vertex cover V' of G such that $|V'| \leq |F|$. For $e \in F$, if e is purple, i.e. $e = \{v_0; x\}$, then add x to V'. Otherwise, if $e = \{x; y\} \in E$ then add either x or y to V' (and if it is red ignore it). Clearly $|V'| \leq |F|$ and moreover it is a valid vertex cover, since for the unsatis ed triangle inequality corresponding to a given edge in $e \in E$, the solution F must contain either e or one of its adjacent purple edges (and in either case we add an endpoint of e to V').

Conversely, observe that for each unsatis ed inequality corresponding to an edge $\{x; y\} \in E$, if we change the weight of either the purple edge $w(v_0; x)$ or $w(v_0; y)$ (or both) to be $1 +$ then the inequality will be satis ed. Moreover, observe that if for a given subset of purple edges, if we increase all their weights to be $1 +$, and not modify any other edge weights, then we will not create any new unsatis ed inequalities. So let V' be any solution to the vertex cover problem on G. The above implies that if for every $x \in V'$ we set $w(v_0; x) = 1 +$, and leave all other weights unchanged, then we have a valid solution to Problem 3.1 on H, such that $|F| = |V'|$ where F is the set of modi ed edges.

The above reduction shows that Problem 3.1 is NP-Complete. In terms of approximating the GMVD (or equivalent MMVD) optimization problem, observe that the above reduction actually implies that a minimum sized solution to the described instance of GMVD corresponds to a minimum vertex cover from G (which we can even be read o). Thus any approximation to GMVD yields the same approximation to the minimum vertex cover problem.

It is not di cult to argue that the hardness proof above for GMVD also applies to GMVID, since in the proof weights only ever needed to be increased (we omit the details as they are nearly identical to the above). Thus we have the following corollary of Theorem 3.1.

COROLLARY 3.1. *The decision version of GMVID is NP-Complete. Moreover, GMVID is APX-hard, and assuming the Unique Games Conjecture, is hard to approximate within a factor of* $2 -$ " *for any* " > 0.

REMARK 3.1. *Consider the standard fixed parameter tractable algorithm for vertex cover. For a given edge in the graph, we know it must be covered by one of its endpoints, thus we try both and in each case recursively solve the subproblem on the graph obtained by throwing out all the edges adjacent to the selected vertex (stopping when the depth is larger than* k*). Now consider the GMVD problem. The analogous strategy would be to find some unsatisfied triangle inequality and try all possibilities for selecting each one of its three edges. The issue is that here this does not define a nice recursive subproblem since whether the selected edge covers other unsatisfied triangles it is involved in depends on how we set its weight, and these different triangles can have competing constraints on how this weight must be set. Thus at least naively it is not clear how to obtain a fixed parameter tractable algorithm for GMVD, and this difference may suggest it is strictly harder than vertex cover to approximate.*

3.1 Metric Violation Decrease Distance is Polynomial Time Solvable.

In this section we show that GMVDD is actually polynomial time solvable by making a connection to the all pairs shortest paths problem.

LEMMA 3.1. *For an instance* $G = (V; E)$ *of GMVDD (Problem 2.3), a minimum size subset* $S \subseteq E$ *such that only decreasing edges from* S *turns* G *into a metric graph can be found in polynomial time. More precisely, the running time is the same as computing the set of all pairs shortest path distances.*

Proof. The algorithm is simple. Compute the all pairs shortest paths distances, and for every edge $e = \{u; v\} \in E$, set $w(e)$ equal to the shortest path distance between u and v. Thus S is the set of edges whose weight is larger than the shortest path distance between its end points, as these were the only weights which were changed.

First observe that it is necessary to change the weights of the edges in S. Speci cally, let $e = \{u; v\}$, and suppose $w(e)$ is larger than the shortest path distance between u and v. In this case the shortest

$u;v$ path together with the edge e forms an unbalanced cycle, which by Lemma 2.1 we must balance. In order to balance this cycle, $w(e)$ needs to be less than or equal to the sum of the weights of the other edges in the cycle, and since edge weights can only decrease, this implies balancing this cycle requires decreasing $w(e)$.

Now we argue that setting the edge weights in S to their shortest path distance su ces to solve this instance of Problem 2.3. First observe that for any edge $e = \{u;v\}$ from any positively weighted graph if we set $w(e)$ equal to its shortest path distance, then the shortest path distance between every pair of vertices in the graph remains the same. (This is because the only path lengths which can change are those which used the edge e, and for any such path if we replace e with the shortest $u;v$ path from the original graph, we will get a walk in the original graph of the same total length.) Now let H be the graph obtained from G after setting the weights in S to their shortest path distance from G. The above observation implies that shortest path distances in H are the same as those in G, as one can imagine obtaining H from G by modifying one edge from S at a time. As H was obtained from G by setting weights of edges to their shortest path distance in G, this implies that in H the weight of every edge is equal to the shortest path distance between its endpoints, implying there can be no unsatis ed triangle inequalities and so we are done.

4 Approximation Algorithm for GMVID

In this section we rst provide the details for our approximation to the GMVID problem. In the next section, our approximation for the general GMVD problem will then follow by a more intricate version of the same argument.

4.1 Unbalanced Cycles. Before providing our approximation algorithm for GMVID, in this subsection we rst show that solutions to GMVID can be characterized in terms of unbalanced cycles. Recall Lemma 2.1, which implies that any solution to a GMVID instance must non-top cover all unbalanced cycles. We argue the more surprising fact that any such non-top cover is also su cient.

LEMMA 4.1. *If G is an instance of GMVID and S is a non-top cover of all unbalanced cycles, then G can be converted into a metric graph by only increasing weights of edges in S.*

Proof. For now assume all edge weights are integers and let L denote the largest edge weight. We describe a procedure which only modi es edges from S, producing a new instance G', such that (i) S remains a non-top cover of G', (ii) the weight of at least one edge strictly increases and none decrease, and (iii) no edge weight is ever increased above L. For any instance G and non-top cover S, if we prove such a procedure exists whenever not all the edges in S have weight L, then this will imply the lemma. Speci cally, after applying the procedure at most $|S| \cdot L$ times, all edge weights will be equal to L, and hence no unbalanced cycle (and so no unsatis ed triangle inequality) can remain, since otherwise S would cover that unbalanced cycle with an edge of weight L, which is at least the weight of that cycle's top edge (i.e. the cycle is not actually unbalanced). Moreover, this procedure only increases weights of edges in S, as desired. Note also that as the number of steps in this existential argument was irrelevant, so long as it was nite, this procedure will imply the claim for rational input weights as well.

We now prove the above described procedure exists. Let G and S be as in the lemma statement, and x any unbalanced triangle, which must exist otherwise all triangle inequalities are already satis ed. Let a, b, and t be the edges of this triangle, where t is the top edge, i.e. $w(t) > w(a) + w(b)$. Note at least one of either a or b must be in the set S. We break the analysis into two cases based on whether just one or both are in S. In the following, for any cycle $\mathbb{C}$, let $w(\mathbb{C})$ denote the sum of the weights of the edges in $\mathbb{C}$.

Case 1: only one of a or b is in S. Without loss of generality suppose a is in S. We then increase $w(a)$ to be $w(t) - w(b)$. If no new unbalanced cycles are created after the increase, then we are done. Otherwise, if some new unbalanced cycle $\mathbb{C}$ was created after increasing $w(a)$, then a must be the top edge of $\mathbb{C}$. Let $(\mathbb{C} \setminus a)$ denote the sub-path of cycle $\mathbb{C}$ starting at the common vertex of a and t, and ending at the common vertex of a and b. Consider the closed walk $= (\mathbb{C} \setminus a) \circ b \circ t$, where `$\circ$' denotes walk concatenation. Note may not be a (simple) cycle, however must contain a cycle C which includes the edge t. Note that C is unbalanced with top edge t, since it only contains edges from $(\mathbb{C} \setminus a) \cup \{b;t\}$, and so $w(C \setminus t) = w(C) - w(t) \leq w(\mathbb{C} \setminus a) + w(b) = w(\mathbb{C} \setminus a) - w(a) + w(t) < w(t)$, where the last equality follows since we set $w(a) = w(t) - w(b)$. Since C is unbalanced, it must be non-top covered by S, which implies $\mathbb{C}$ is non-top cover by S. This is because C only contains edges from $(\mathbb{C} \setminus a) \cup \{b;t\}$, has top edge t, and by assumption $b \in S$.

Case 2: both a and b are in S. First, increase $w(a)$ to the largest value possible that does not create any new cycles that are both unbalanced and not non-top covered. (Note the largest such value is well de ned as equality satis es the triangle inequality.) If afterwards $w(a) \geq w(t) - w(b)$, then we are done as the weight of a has strictly increased since the previously violated

triangle is now satis ed. Otherwise, $w(a) < w(t) - w(b)$ and there is some just balanced cycle $\mathbb{C}_1$ such that $(\mathbb{C}_1 \setminus a) \cap S = \emptyset$, which would become unbalanced if $w(a)$ were any larger. Now increase $w(b)$ to $w(b) = w(t) - w(a)$. If this does not create any non-top uncovered unbalanced cycle then we are done. Otherwise, there is some non-top uncovered unbalanced cycle $\mathbb{C}_2$ created by setting $w(b) = w(t) - w(a)$. Observe for later that after the change to $w(b)$, it still holds that $w(\mathbb{C}_1 \setminus a) = w(a)$ as $(\mathbb{C}_1 \setminus a) \cap S = \emptyset$ and $b \in S$. Let $(\mathbb{C}_1 \setminus a)$ denote the sub-path of cycle $\mathbb{C}_1$ starting at the common vertex of a and t, and ending at the common vertex of a and b, and let $(\mathbb{C}_2 \setminus b)$ denote the sub-path of cycle $\mathbb{C}_2$ starting at the common vertex of a and b, and ending at the common vertex of b and t. Then consider the closed walk $= (\mathbb{C}_1 \setminus a) \circ (\mathbb{C}_2 \setminus b) \circ t$. As is closed it must contain some cycle C which includes t. Note that C is unbalanced with top edge t, since it only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup \{t\}$, and so $w(C \setminus t) = w(C) - w(t) \leq w(\mathbb{C}_1 \setminus a) + w(\mathbb{C}_2 \setminus b) = w(a) + w(\mathbb{C}_2 \setminus b) < w(a) + w(b) = w(t)$. Since C is unbalanced, it must be non-top covered by S, which implies $\mathbb{C}_2$ is non-top covered by S. This is because C only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup \{t\}$, $(\mathbb{C}_1 \setminus a) \cap S = \emptyset$, and t is the top edge of C.

In either case, the weight of at least one edge from S was strictly increased, no edges were decreased, and any newly created unbalanced cycle is still covered by S, and thus the conditions of the above described procedure are satis ed.

Lemma 2.1 and Lemma 4.1 immediately imply the following.

COROLLARY 4.1. *Let G be an instance of GMVID. If $S \subset E$ is a minimum size set which non-top covers all the unbalanced cycles in G, then S is an optimal solution to GMVID.*

4.2 Small Cycle Covers are Almost Enough. Corollary 4.1 showed the equivalence between minimum non-top covers of all unbalanced cycles and solutions to GMVID. The question is how hard is it to nd or approximate such a cover. We have already argued GMVID is at least as hard as vertex cover, however, covering all unbalanced cycles appears a step closer to the more general set cover problem, which is logarithmically hard to approximate. Worse still, we cannot (at least explicitly) write down our non-top cover problem as a set cover instance as there are potentially an exponential number of unbalanced cycles.

One natural question is whether explicitly covering larger cycles is necessary, i.e. perhaps non-top covering all unbalanced 3 cycles implies non-top covering all larger unbalanced cycles, however the example in Figure 4.1 quickly dispels this idea. This example consists of single unbalanced 4-cycle, whose vertices in cyclic order are $(a; b; c; d)$, whose top edge $\{a; d\}$ has weight $3+"$, and all other edges have weight 1. Here there are are two unbalanced 3-cycles, $(a; b; d)$ and $(a; c; d)$. Thus all unbalanced 3-cycles can be covered by the edges, $\{a; c\}$ and $\{b; d\}$, however this does not cover the unbalanced 4-cycle. Unfortunately, this can be generalized for any integer n, by creating a cycle of length n where all edges in the graph have weight 1 except for a single cycle edge which has weight $n - 1 + "$. Thus for no $k < n$ does a non-top cover of all unbalanced cycles of size $\leq k$ imply a non-top cover of all unbalanced cycles.

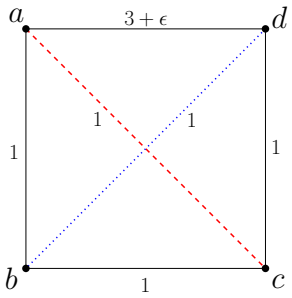

Figure 4.1: Unbalanced 4-cycle not covered by the dashed 3-cycle cover.

So we must consider larger unbalanced cycles, which motivates the following de nitions.

DEFINITION 4.1. *Let C be an unbalanced cycle of length $k \geq 4$, and let $(v_1; v_2; \ldots; v_k)$ be the cyclic ordering of the vertices in C, where $t = \{v_k; v_1\}$ is the top edge of C. For any chord $e = \{v_i; v_j\}$ of C (i.e. $i < j$ and differ by more than 1 mod k), we define two cycles. The* top cycle, *$top(C; e) = (v_1; \ldots; v_i; v_j; \ldots; v_k)$ containing the top edge t, and the* bottom cycle, *$bot(C; e) = (v_i; v_{i+1} \ldots; v_j)$.*

For an unbalanced cycle C, if there exists a chord e of C such that $bot(C; e)$ is unbalanced and *e is the top edge of $bot(C; e)$, then C is called a* non-unit *cycle, otherwise C is called a* unit *cycle.*

The following simple observation implies we can limit attention to unit cycles.

OBSERVATION 4.1. *Let $G = (V; E)$ be an instance of GMVID. If $S \subset E$ non-top covers all unit cycles in G, then S must non-top cover all unbalanced cycles, and hence is a solution to GMVID. This is because by definition any non-unit cycle C must have a chord e whose bottom cycle is unbalanced with top edge e, and in particular the smallest such bottom cycle C' must be a unit cycle. Thus as the non-top edges of C' are a subset*

of those of C, if S non-top covers C' it must non-top cover C.

In general there can still be large unit cycles, however, we have the following useful property.

LEMMA 4.2. *For any chord e of a unit cycle C, $top(C;e)$ is unbalanced, with the same top edge as C.*

Proof. First, observe that since C is a unit cycle, either the bottom cycle of e is balanced, or it is unbalanced but the top edge is not e. In either case, $w(e) \leq w(bot(C;e)) - w(e)$. Let t be the top edge of C. We thus have $w(t) > w(C) - w(t) = w(top(C;e)) + w(bot(C;e)) - 2w(e) - w(t) \geq w(top(C;e)) - w(t)$.

To see why the above lemma is useful for handling larger unit cycles, we must first get rid of smaller unbalanced cycles, which can be easily done.

LEMMA 4.3. *Let $G = (V;E)$ be an instance of GMVID, let $C_{\leq k}$ be the set of all unbalanced cycles with at most k edges for some constant k, and let opt be a minimum size non-top cover of $C_{\leq k}$. Then in polynomial time one can compute a constant factor approximation to opt.*

More precisely, in $O(|V|^k)$ time, one can compute a set $S_{\leq k}$ which non-top covers $C_{\leq k}$, and such that $|S_{\leq k}| \leq (k-1)|opt|$.

Proof. The proof follows by applying the standard hitting set approximation for bounded size sets. Specifically, each unbalanced cycle in $C_{\leq k}$ defines a set of $\leq k-1$ non-top edges. Let H denote the collection of all such sets, and observe a non-top cover of $C_{\leq k}$ corresponds to a hitting set for H. So initially let $S_{\leq k}$ be the empty set, and consider the sets in H one at a time. If when considering a set $h \in H$, if h has not been hit, then add all edges of h to $S_{\leq k}$, and otherwise do nothing. Clearly the final set $S_{\leq k}$ is a hitting set, and has size at most $(k-1)|opt|$ since each time we add all edges of a set, the set was uncovered, and so any solution had to add at least one of those edges. The running time follows as we make a single linear pass over H, and note that $C_{\leq k}$ and hence H can be enumerated in $O(|V|^k)$ time.

Now for larger unit cycles we have the following useful corollary of Lemma 4.2.

COROLLARY 4.2. *Let $S_{\leq k}$ be a non-top cover of all unbalanced cycles of length $\leq k$. Then for any unit cycle C with strictly more than k edges, if C is* not *non-top covered by $S_{\leq k}$, then for any chord e such that $top(C;e)$ has $\leq k$ edges, $e \in S_{\leq k}$.*

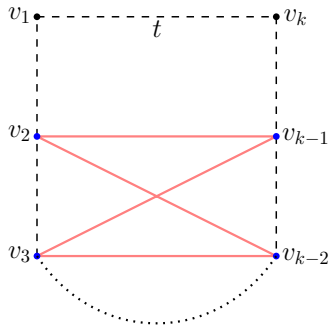

Figure 4.2: Cycle $C = (v_1; v_2 : : : ; v_k)$, and edges of $embed4(C)$ in solid red.

Proof. Let e be any chord such that $top(C;e)$ has $\leq k$ edges. As C is a unit cycle, by Lemma 4.2, $top(C;e)$ is unbalanced, with the same top edge as C. Thus $S_{\leq k}$ must non-top cover $top(C;e)$. Note however, that with the exception of the edge e, the non-top edges of $top(C;e)$ are a subset of those of C. Thus since C is not non-top covered by $S_{\leq k}$, e must be the edge in $S_{\leq k}$ which non-top covers $top(C;e)$.

By Observation 4.1 we know it suffices to non-top cover all unit cycles, however, the issue is that in general unit cycles may be large. On the other hand, Corollary 4.2 tells us that if we first cover smaller unbalanced cycles, then while we may not cover larger unit cycles, we do cover all chords near their top edges. We first describe a procedure which adds a carefully chosen set of edges to cover larger unit cycles, and then in the next section argue how to bound the size of the added set by charging to these chords.

DEFINITION 4.2. *Let $S_{\leq 6}$ be a non-top cover of all unbalanced cycles with ≤ 6 edges. Consider any unbalanced cycle $C = (v_1; v_2; v_3 : : : ; v_{k-2}; v_{k-1}; v_k)$, with $k > 6$, where $t = \{v_k; v_1\}$ is the top edge of C. For any such cycle C, we define the* embedded 4-cycle *of C to be the cycle defined by the cyclically ordered edges $embed4(C) = (\{v_2; v_{k-2}\}; \{v_{k-2}; v_3\}; \{v_3; v_{k-1}\}; \{v_{k-1}; v_2\})$. See Figure 4.2.*

Observe that for a unit cycle C with more than 6 edges, Corollary 4.2 implies that if C is not covered by $S_{\leq 6}$, then $S_{\leq 6}$ must contain all edges of $embed4(C)$. Next observe that the chords of $embed4(C)$, namely the edges $\{v_2; v_3\}$ and $\{v_{k-2}; v_{k-1}\}$ are actually non-top edges of the cycle C. In other words, if we add to $S_{\leq 6}$ any set of edges containing at least one chord from the embedded 4 cycle of each unit cycle, then we will have a non-top cover of all unit cycles, and hence a solution to GMVID by Observation 4.1. Recall we cannot list all unit cycles, and hence it is not clear how

to precisely list all their embedded 4 cycles. On the other hand, as all edges of an embedded 4 cycle of a unit cycle are contained in $S_{\leq 6}$, we can instead consider the potentially larger set of all 4 cycles in the graph de ned by the edge set $S_{\leq 6}$. This implies the following lemma.

LEMMA 4.4. *Let G be an instance of GMVID, and let $S_{\leq 6}$ be a non-top cover of all unbalanced cycles with ≤ 6 edges. Let S_c be a set of edges containing at least one chord from all possible 4 cycles defined by the edges of $S_{\leq 6}$. Then $S = S_{\leq 6} \cup S_c$ is a valid solution to the given instance of GMVID, that is the edges in S can be increased to convert G into a metric graph.*

4.3 Chording Cycles. The goal of this section is to compute a set S_c as described in Lemma 4.4 which contains at least one chord of every induced 4-cycle of a subset of edges. So let $G = (V; E)$ be a complete graph, and for any edge subset $F \subseteq E$, consider the induced subgraph $H = (V; F)$. Let $Cycle(F)$ denote the set of all cycles in H with with exactly 4 edges. Let $chord4(F)$ denote the following iterative procedure. Initially let $S = \emptyset$, and consider each cycle $C \in Cycle(F)$ one at time. If either chord of C appears in S ignore C, and otherwise add both chords to S. (Note chords are considered from the complete graph G, i.e. regardless of whether they appear in H:) After all cycles have been considered output S.

Clearly $chord4(F)$ outputs a set S containing at least one chord from each cycle in $Cycle(F)$ and does so in polynomial time. The question is how big is $|S|$ relative to $|F|$. So let $\mathbb{C}$ denote the set of cycles for which $chord4(F)$ added its chords to S. Observe that no two cycles in $\mathbb{C}$ can share a chord (as otherwise $chord4(F)$ would not have added the latter of these two cycles to $\mathbb{C}$). So we have $|S|{=}2 = |\mathbb{C}|$, and also note that $|edges(\mathbb{C})| \leq |F|$, where $edges(\mathbb{C})$ is the set of all edges from cycles in $\mathbb{C}$. Thus to bound the size of $|S|$ in terms of $|F|$, it su ces to bound $|\mathbb{C}|$ in terms of $|edges(\mathbb{C})|$. This gives the following purely combinatorial problem.

LEMMA 4.5. *Let $G = (V; E)$ be a graph whose edge set E is the union of the edges of a collection of 4-cycles, $\mathbb{C}$, such that no two 4-cycles in $\mathbb{C}$ share a chord. (Note this applies to all chords in the complete graph on V, i.e. regardless of whether they appear in E.) Then $|\mathbb{C}| = O(|E|^{4=3})$.*

Proof. We partition the vertex set V into two groups based on degree. Speci cally, let V_s be the set of all vertices of degree $\leq |E|^{1=3}$ and let V_l be the vertices with degree $> |E|^{1=3}$. Now we separately bound the number of cycles containing at least one vertex from V_s and the number of cycles only containing vertices from V_l, and hence the total number of cycles is the sum of these two numbers. First, we further partition V_s into groups $g_1; :::; g_k$ such that for all $1 \leq i \leq k$, $|E|^{1=3} \leq \sum_{v \in g_i} degree(v) \leq 2|E|^{1=3}$. (Such a partition can be computed by iterating over the vertices in V_s with a running degree sum total, setting aside a group and reseting the total to zero each time the sum is $\geq |E|^{1=3}$.) Note that as the total degree of each g_i is at least $|E|^{1=3}$, by the degree sum formula $k|E|^{1=3} \leq 2|E|$, and so $k \leq 2|E|^{2=3}$. Now any two edges adjacent to the same vertex can both appear together in at most one 4-cycle, as otherwise this would imply the two 4-cycles share a chord. Thus each distinct pair of edges adjacent to a vertex v can correspond to at most one cycle, and hence the total number of cycles involving v is bounded by $\binom{degree(v)}{2} \leq (degree(v))^2{=}2$. We thus have that number of cycles involving vertices from V_s is

$$\leq \frac{1}{2} \sum_{i=1}^{k} \sum_{v \in g_i} (degree(v))^2 \leq \frac{1}{2} \sum_{i=1}^{k} \left(\sum_{v \in g_i} degree(v) \right)^2$$
$$\leq \frac{1}{2} \sum_{i=1}^{k} 4|E|^{2=3} \leq |E|^{2=3} \cdot 4|E|^{2=3} = 4|E|^{4=3};$$

where the second inequality follows from the fact that $a^2 + b^2 \leq (a + b)^2$ for $a; b \geq 0$.

Now consider the vertices in V_l. As each vertex in V_l has degree $> |E|^{1=3}$, by the degree sum formula, $2|E| \geq \sum_{v \in V_l} degree(v) \geq |V_l| \cdot |E|^{1=3}$. Therefore $|V_l| \leq 2|E|^{2=3}$. Now as discussed above, we only need to consider cycles composed entirely of vertices from V_l. Now V_l can de ne at most $\binom{|V_l|}{2} \leq |V_l|^2{=}2 \leq 2|E|^{4=3}$ chords, and since no two cycles can share a chord, this implies V_l can de ne at most $|E|^{4=3}$ cycles, and thus the lemma statement follows.

By the discussion before the lemma, we thus have the following.

COROLLARY 4.3. *Let F be any subset of edges from a complete graph $G = (V; E)$. Then in polynomial time $chord4(F)$ outputs a set of edges S such that (i) S contains at least one chord of every 4-cycle induced by the edges of F, and (ii) $|S| = O(|F|^{4=3})$.*

The reader may wonder whether the analysis of Lemma 4.5 is tight. Thus in Appendix A we provide a corresponding lower bound showing that indeed the 4=3 exponent is tight in the worst case for this combinatorial problem. This lower bound uses an interesting and non-trivial probabilistic method argument, though as it does not directly relate to upper bounding the approximation quality of our algorithm, it is left to the appendix.

4.4 The Result. The approach outlined in Section 4.2 (particularly Lemma 4.4), together with the bound from Corollary 4.3 readily give the following.

Input : An instance G of GMVID
Output: Valid solution S to the given instance.
1 Compute a non-top cover $S_{\leq 6}$ of all unbalanced cycles with ≤ 6 edges using Lemma 4.3
2 Compute a chord cover $S_c = chord4(S_{\leq 6})$ using Corollary 4.3
3 Formulate and return any feasible solution to the LP described in Section 2.2 for the edge set $S = S_{\leq 6} \cup S_c$.

Algorithm 1: Finds a valid solution for GMVID.

THEOREM 4.1. *Algorithm 1 gives a polynomial time $O(OPT^{1=3})$-approximation to any instance G of GMVID (Problem 2.4), where OPT denotes the size of an optimal solution.*

Proof. Algorithm 1 returns a set $S = S_{\leq 6} \cup S_c$. By Lemma 4.3, $S_{\leq 6}$ is a non-top cover of all unbalanced cycles with ≤ 6 edges, and by Corollary 4.3, S_c contains at least one chord from every 4-cycle induced by the edges of $S_{\leq 6}$. Thus by Lemma 4.4, S is a valid solution to the given instance of GMVID.

The running time is clearly polynomial as the procedures of Lemma 4.3, Corollary 4.3, and the feasibility checking of Section 2.2, were all already argued to run in polynomial time. As for the approximation quality, note that Lemma 4.1 implies OPT is the size of a minimum sized set of edges which non-top covers all unbalanced cycles. Moreover, Lemma 4.3 tells us $|S_{\leq 6}| \leq 5|opt_6| = O(OPT)$, where opt_6 denotes any minimum size non-top cover of all unbalanced cycles of length ≤ 6. Finally, Corollary 4.3 tells us $|S_c| = O(|S_{\leq 6}|^{4=3})$, and therefore $|S| = O(OPT^{4=3})$.

5 General Metric Violation Distance

Now we consider the general GMVD problem, where both increasing and decreasing edge weights are allowed. The high level argument will be similar to that used for GMVID.

5.1 Unbalanced Cycles. Recall our di erent notions of covering from Section 2.3. Lemma 2.1 implies any solution to a GMVD instance must regular cover all unbalanced cycles. We now prove the more surprising fact that any such regular cover is also su cient, that is the analog of Lemma 4.1 but for GMVD. However, the proof here is far more intricate, and in particular requires the following helper lemma.

LEMMA 5.1. *Let $G = (V; E)$ be an instance of GMVD. If S is a regular cover of all unbalanced cycles, then S can be partitioned into two disjoint sets S^+ and S^- such that each unbalanced cycle is either non-top covered by S^+ or top covered by S^-.*

Proof. Let S be a regular cover of all unbalanced cycles and let S^+ and S^- initially be empty sets. We now de ne an interative procedure, which in each iteration removes one edge from S and adds it to either S^+ or S^-. We maintain the invariant that each unbalanced cycle is either top covered by $S^- \cup S$ or non-top covered by $S^+ \cup S$. Thus, after a nite number of iterations, S will be empty, and S^+ and S^- will be two disjoint sets such that each unbalanced cycle is either non-top covered by S^+ or top covered by S^-.

Suppose at step i, we pick an edge b from set S. There are two cases. Case 1: if each unbalanced cycle is either top covered by $S^- \cup (S \setminus b)$ or non-top covered by $(S^+ \cup b) \cup (S \setminus b)$, then we add b to S^+. Case 2: if each unbalanced cycle is either top covered by $(S^- \cup b) \cup (S \setminus b)$ or non-top covered by $S^+ \cup (S \setminus b)$, then we add b to S^-. We now argue by contradiction that these are the only possible cases.

If Case 1 does not hold, then there must be an unbalanced cycle $\mathbb{C}_1$ which is neither top covered by $S^- \cup (S \setminus b)$ nor non-top covered by $(S^+ \cup b) \cup (S \setminus b)$. As $\mathbb{C}_1$ must be top covered by $S^- \cup S$ or non-top covered by $S^+ \cup S$ (by induction), this implies that b must top cover $\mathbb{C}_1$. If Case 2 does not hold then there is an unbalanced cycle $\mathbb{C}_2$ which is neither top covered $(S^- \cup b) \cup (S \setminus b)$ nor non-top covered by $S^+ \cup (S \setminus b)$, and similarly this implies b must non-top cover $\mathbb{C}_2$. Now consider the closed walk $= (\mathbb{C}_1 \setminus b) \circ (\mathbb{C}_2 \setminus b)$ (see Figure 5.1). Let t be the top edge of $\mathbb{C}_2$. There exists a cycle C only containing edges from whose top edge is t, and is unbalanced because $w(t) > w(\mathbb{C}_2 \setminus t) = w(\mathbb{C}_2 \setminus \{t; b\}) + w(b) > w(\mathbb{C}_2 \setminus \{t; b\}) + w(\mathbb{C}_1 \setminus b) \geq w(C) - w(t)$. Observe that $(\mathbb{C}_1 \setminus b) \cap (S^+ \cup S) = \emptyset$ and $(\mathbb{C}_2 \setminus t) \cap (S^+ \cup (S \setminus b)) = \emptyset$ which implies $(C \setminus t) \cap (S^+ \cup S) = \emptyset$ because $b \in C$. Additionally, we have that $t \notin (S^- \cup S)$ as otherwise $\mathbb{C}_2$ would be top covered by $(S^- \cup b) \cup (S \setminus b)$. As C must be top covered by $S^- \cup S$ or non-top covered by $S^+ \cup S$ (again by induction), this give a contradiction as we showed $t \notin (S^- \cup S)$ and $(C \setminus t) \cap (S^+ \cup S) = \emptyset$.

Therefore, we can add b to S^+ or S^- (according to case 1 or 2) and remove it from S such that each unbalanced cycle remains either top covered by $S^- \cup S$ or non-top covered by $S^+ \cup S$. Thus, after at most $|S_0|$ rounds, where S_0 is the initial state of S, S^+ and S^- will be as in the lemma statement.

LEMMA 5.2. *If G is an instance of GMVD and S is a regular cover of all unbalanced cycles, then G can be*

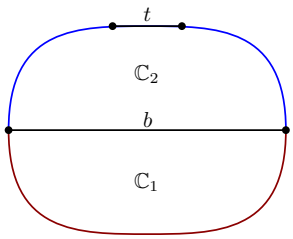

Figure 5.1: Edge b top covers $\mathbb{C}_1$ and non-top covers $\mathbb{C}_2$.

converted into a metric graph by only changing weights of edges in S.

Proof. For now assume all edge weights are integers and let L denote the largest edge weight. First use Lemma 5.1 to partition S into two disjoint sets S^+ and S^-, such that every unbalanced cycle is either non-top covered by S^+ or top covered by S^-. We describe a procedure, producing a new instance G', such that (i) every unbalanced cycle of G' is either non-top covered by S^+ or top covered by S^-, (ii) the weight of either at least one S^+ edge strictly increases or at least one S^- edge strictly decreases (and no other edge weights are modified), and (iii) no edge weight is ever increased above L or below 0. Proving such a procedure exists for any instance G and regular cover S, will imply the lemma. Specifically, after applying the procedure at most $|S| \cdot L$ times, all edge weights in S^+ will be equal to L, and all edge weights in S^- will be equal to 0. This implies there are no remaining unbalanced cycles, since otherwise the unbalanced cycle is either non-top covered by an edge of weight L or top covered by an edge of weight 0, in either case implying the cycle is not actually unbalanced. Moreover, this procedure only modifies edge weights from S as desired. As the number of steps in this existential argument was irrelevant, so long as it was finite, this procedure will imply the claim for rational input weights as well.

We now prove the above described procedure exists. Let G and S be as in the lemma statement, and x any unbalanced triangle, which must exist otherwise all triangle inequalities are already satisfied. Let a, b, and t be the edges of this triangle, where t is the top edge. Note that either $\{a; b\} \cap S^+ \neq \emptyset$ or $t \in S^-$. We break the analysis into several cases based on whether just one or both of a,b are in S^+, or neither are in S^+ and t is in S^-. If a cycle is not non-top covered by S^+ or not top covered by S^-, we say that cycle is uncovered.

Case 1: $t \in S^-$, and exactly one of a or b is in S^+. Without loss of generality suppose $a \in S^+$, $b \in S^+; t \in S^-$. We then increase $w(a)$ to be $w(t) - w(b)$. If no new unbalanced cycles are created after the increase, then we are done. Otherwise, if some new unbalanced cycle $\mathbb{C}$ was created after increasing $w(a)$, then a must be the top edge of $\mathbb{C}$. Let $(\mathbb{C} \setminus a)$ denote the sub-path of cycle $\mathbb{C}$ which starts at the common vertex of a and t, and ends at the common vertex of a and b. Consider the closed walk $= (\mathbb{C} \setminus a) \circ b \circ t$. Note may not be a (simple) cycle, but must contain a cycle C which includes the edge t. Note that C is unbalanced with top edge t, since it only contains edges from $(\mathbb{C} \setminus a) \cup \{b; t\}$, and so $w(C \setminus t) = w(C) - w(t) \leq w(\mathbb{C} \setminus a) + w(b) = w(\mathbb{C} \setminus a) - w(a) + w(t) < w(t)$, where the last equality follows since we set $w(a) = w(t) - w(b)$. Since C is unbalanced, it must be non-top covered by S^+ as $t \in S^-$, which implies $\mathbb{C}$ is non-top covered by S^+, because C only contains edges from $(\mathbb{C} \setminus a) \cup \{b; t\}$, has top edge t, and by assumption $b \in S^+$.

Case 2: $t \in S^-$ and $a; b \in S^+$. First, increase $w(a)$ to the largest value possible that does not create any uncovered unbalanced cycle. If afterwards $w(a) \geq w(t) - w(b)$, then we are done as the weight of a has strictly increased since the previously violated triangle is now satisfied. Otherwise, $w(a) < w(t) - w(b)$ and there is some just balanced cycle $\mathbb{C}_1$ that $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$, which would become unbalanced if $w(a)$ were any larger. Now increase $w(b)$ to $w(t) - w(a)$. If this does not create any uncovered unbalanced cycle then we are done; otherwise, some unbalanced cycle $\mathbb{C}_2$ was created. Observe for later that after the change to $w(b)$, it still holds that $w(\mathbb{C}_1 \setminus a) = w(a)$ as $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$ and $b \in S^+$. Let $(\mathbb{C}_1 \setminus a)$ denote the sub-path of $\mathbb{C}_1$ starting at the common vertex of a and t, and ending at the common vertex of a and b, and let $(\mathbb{C}_2 \setminus b)$ denote the sub-path of $\mathbb{C}_2$ starting at the common vertex of a and b, and ending at the common vertex of b and t. Then consider the closed walk $= (\mathbb{C}_1 \setminus a) \circ (\mathbb{C}_2 \setminus b) \circ t$. As is closed it must contain some cycle C which includes the edge t. Note that C is unbalanced with top edge t, since it only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup \{t\}$, and so $w(C \setminus t) = w(C) - w(t) \leq w(\mathbb{C}_1 \setminus a) + w(\mathbb{C}_2 \setminus b) = w(a) + w(\mathbb{C}_2 \setminus b) < w(a) + w(b) = w(t)$. Since C is unbalanced, it must be non-top covered by S^+ as $t \in S^-$, which implies $\mathbb{C}_2$ is non-top covered by S^+. This is because C only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup \{t\}$, and $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$, and t is the top edge of C.

Case 3: $t \in S^-$ and $a; b \in S^+$. First, decrease $w(t)$ to be $w(a) + w(b)$. If no new unbalanced cycles are created, then we are done. Otherwise, some new unbalanced cycle $\mathbb{C}$ was created, which includes t and has top edge $t' \neq t$. Let $(\mathbb{C} \setminus t)$ denote the sub-path $\mathbb{C}$ starting at the common vertex of t and a, and ending at the common vertex of t and b. Consider the closed walk $= (\mathbb{C} \setminus t) \circ a \circ b$, and let C be a cycle contained in the which includes t'. Note that C is unbalanced with top edge t', since it only contains edges from $(\mathbb{C} \setminus t) \cup \{a; b\}$,

and so it holds that $w(C \setminus t') = w(C) - w(t') \leq w(\mathbb{C}) + w(a) + w(b) - w(t) - w(t') = w(\mathbb{C} \setminus t') < w(t')$, where the last equality follows since we set $w(t) = w(a) + w(b)$. Since C is unbalanced, it must be either top covered by S^- or non-top covered by S^+, which implies $\mathbb{C}$ is either top covered by S^- or non-top covered by S^+, because C only contains edges from $(\mathbb{C} \setminus t) \cup \{a, b\}$, and by assumption $a, b \in S^+$.

Case 4: $t \in S^-$ and exactly one of a or b is in S^+. Without loss of generality suppose $a \in S^+$, $b \notin S^+$. First, increase $w(a)$ to the largest value possible that does not create any new uncovered unbalanced cycle. If afterwards $w(a) \geq w(t) - w(b)$, then we are done. Otherwise, $w(a) < w(t) - w(b)$ and there is some just balanced cycle $\mathbb{C}_1$ with $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$, which would be unbalanced if $w(a)$ were any larger. Now decrease $w(t)$ to $w(t) = w(b) + w(a)$. If this does not create an unbalance cycle, then we are done. Otherwise, some unbalanced cycle $\mathbb{C}_2$ containing t was created, with top edge $t' \neq t$. Let $(\mathbb{C}_1 \setminus a)$ denote the sub-path of $\mathbb{C}_1$ starting at the common vertex of a and t and ending at the common vertex of a and b, and let $(\mathbb{C}_2 \setminus b)$ denote the sub-path of $\mathbb{C}_2$ starting at the common vertex of b and t and ending at the common vertex of a and t. Consider the closed walk $= (\mathbb{C}_1 \setminus a) \circ b \circ (\mathbb{C}_2 \setminus t)$, and let C be a cycle in including t'. Note C is unbalanced with top edge t', since it only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus t) \cup \{b\}$, and so $w(C \setminus t') \leq w(C) - w(t') \leq w(\mathbb{C}_1 \setminus a) + w(\mathbb{C}_2 \setminus t) + w(b) - w(t') = w(a) + w(b) + w(\mathbb{C}_2) - w(t) - w(t') = w(\mathbb{C}_2) - w(t') = w(\mathbb{C}_2 \setminus t') < w(t')$. Since C is unbalanced, it is either non-top covered by S^+ or top covered by S^-, implying $\mathbb{C}_2$ is either non-top covered by S^+ or top covered by $t' \in S^-$, since C only has edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus t) \cup \{b\}$, and $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$, $b \notin S^+$, and t' is the top edge of C.

Case 5: $t \in S^-$ and $a, b \in S^+$. First, increase $w(a)$ to the largest value possible that does not create any uncovered unbalanced cycles. If afterwards $w(a) \geq w(t) - w(b)$, then we are done. Otherwise, $w(a) < w(t) - w(b)$ and there is some just balanced cycle $\mathbb{C}_1$ with $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset$, which would be unbalanced if $w(a)$ were any larger. Now increase $w(b)$ to the largest value possible that does not create any new unbalanced cycles that are not non-top covered. If afterwards $w(b) \geq w(t) - w(a)$, then we are done. Otherwise, $w(b) < w(t) - w(a)$ and there is some just balanced cycle $\mathbb{C}_2$ with $(\mathbb{C}_2 \setminus b) \cap S^+ = \emptyset$, which would be unbalanced if $w(b)$ were any larger. Now decrease $w(t)$ to $w(a) + w(b)$. If this does not create any uncovered unbalanced cycle then we are done. Otherwise, some uncovered unbalanced cycle $\mathbb{C}_3$ is created which contains t and whose top edge is $t' \neq t$. Let $(\mathbb{C}_1 \setminus a)$ denote the sub-path of $\mathbb{C}_1$ starting at the common vertex of a and t, and ending at the common vertex of a and b, let $(\mathbb{C}_2 \setminus b)$ denote the sub-path of $\mathbb{C}_2$ starting at the common vertex of a and b, and ending at the common vertex of b and t, and let $(\mathbb{C}_3 \setminus t)$ denote the sub-path of $\mathbb{C}_3$ starting at the common vertex of b and t, and ending at the common vertex of a and t. Consider the closed walk $= (\mathbb{C}_1 \setminus a) \circ (\mathbb{C}_2 \setminus b) \circ (\mathbb{C}_3 \setminus t)$. As is closed it must contain some cycle C which includes the edge t'. Note that C is unbalanced with top edge t', since it only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup (\mathbb{C}_3 \setminus t)$, and so it holds that $w(C \setminus t') = w(C) - w(t') \leq w(\mathbb{C}_1 \setminus a) + w(\mathbb{C}_2 \setminus b) + w(\mathbb{C}_3 \setminus t) - w(t') = w(a) + w(b) - w(t) + w(\mathbb{C}_3 \setminus t') = w(\mathbb{C}_3 \setminus t') < w(t')$. Since C is unbalanced, it must be either non-top covered by S^+ or top covered by t', which implies $\mathbb{C}_3$ is either non-top covered by S^+ or top covered by t'. This is because C only contains edges from $(\mathbb{C}_1 \setminus a) \cup (\mathbb{C}_2 \setminus b) \cup (\mathbb{C}_3 \setminus t)$, and $(\mathbb{C}_1 \setminus a) \cap S^+ = \emptyset; (\mathbb{C}_2 \setminus b) \cap S^+ = \emptyset$, and t' is the top edge of C.

In every case, the weight of at least one edge from S^+ was strictly increased or from S^- was strictly decreased, and any newly created unbalanced cycle is either top covered by S^- or non-top covered by S^+, and thus the conditions of the above described procedure are satis ed.

5.2 The Result. By the previous subsection, we know that optimal solutions to GMVD are minimum regular covers of all unbalanced cycles. Surprisingly we can approximate such regular covers using the exact same algorithm used to approximate non-top covers, namely output a set $S = S_{\leq 6} \cup S_c$ where $S_{\leq 6}$ is a regular cover of all unbalanced cycles with ≤ 6 edges, and S_c is a set containing at least one chord from every induced 4-cycle of $S_{\leq 6}$. While this still produces a valid solution, the reason somewhat di ers from before, as outlined in proof below. First, we need the following lemma, whose proof is omitted as it is identical to Lemma 4.3, except for the change from non-top to regular covers.

LEMMA 5.3. *Let $G = (V, E)$ be an instance of GMVD, let $C_{\leq k}$ be the set of all unbalanced cycles with at most k edges for some constant k, and let opt be a minimum size regular cover of $C_{\leq k}$. Then in polynomial time one can compute a constant factor approximation to opt.*

More precisely, in $O(|V|^k)$ time, one can compute a set $S_{\leq k}$ which regular covers $C_{\leq k}$, and such that $|S_{\leq k}| \leq k|opt|$.

THEOREM 5.1. *Algorithm 2 gives a polynomial time $O(OPT^{1=3})$-approximation to any instance G of GMVD (Problem 2.2), where OPT denotes the size of an optimal solution.*

Input : An instance G of GMVD
Output: Valid solution S to the given instance.
1 Compute a regular cover $S_{\leq 6}$ of all unbalanced cycles with ≤ 6 edges using Lemma 5.3
2 Compute a cover $S_c = \mathit{chord}4(S_{\leq 6})$ using Corollary 4.3
3 Formulate and return any feasible solution to the LP described in Section 2.2 for the edge set $S = S_{\leq 6} \cup S_c$.

Algorithm 2: Finds a valid solution for GMVD.

Proof. First we argue that Algorithm 2 returns a regular cover of all unbalanced cycles and hence is a valid solution by Lemma 5.2. So consider some unbalanced cycle C'. If it is regular covered by $S_{\leq 6}$ then we are done, so assume otherwise. We now argue C' must be regular covered by S_c. Among all unbalanced bottom cycles de ned by a chord of C' such that the chord is the top edge of the bottom cycle and the bottom cycle is not covered by $S_{\leq 6}$, let $C = \mathit{bot}(C'; e')$ be the one with the minimum number of edges. If C' has no such unbalanced non-regular covered bottom cycle set $C = C'$. Let t denote the top edge of C.

As C is not regular covered by $S_{\leq 6}$, clearly it has > 6 edges, and thus $\mathit{embed}4(C)$ is well de ned (see De nition 4.2). Fix any edge $e \in \mathit{embed}4(C)$, and let $C_1 = \mathit{top}(C; e)$ and $C_2 = \mathit{bot}(C; e)$. We now argue $e \in S_{\leq 6}$. Note that C_1 has at most 6 edges, thus if C_1 is unbalanced, then it must be regular covered by $S_{\leq 6}$, and in particular this implies $e \in S_{\leq 6}$ since e is the only edge of C_1 not in C. Otherwise, C_1 is balanced, in which case $w(e) \geq w(t) - w(C_1 \setminus \{t; e\})$, which implies C_2 is unbalanced as $w(e) \geq w(t) - (w(C_1) - w(t) - w(e)) = 2w(t) - (w(C_1) - w(e)) = 2w(t) - (w(C) - w(C_2 \setminus e)) > w(C_2 \setminus e)$ since $w(t) > w(C) - w(t)$. Thus $C_2 = \mathit{bot}(C; e) = \mathit{bot}(C'; e)$ is an unbalanced bottom cycle of C' with length less than C and with top edge e, and so must be regular covered by $S_{\leq 6}$. As C is not regular covered, and e is the only edge of C_2 not in C, this implies $e \in S_{\leq 6}$. Thus in either case, for any $e \in \mathit{embed}4(C)$, $e \in S_{\leq 6}$. This implies S_c will contain a chord of the cycle $\mathit{embed}4(C)$, and therefore C' will be regular covered by S_c.

The above argues we return a valid solution. The approximation ratio and time complexity analysis are nearly identical to that in the proof of Theorem 4.1, and so are omitted here.

Acknowledgements. The authors thank Sariel Har-Peled for helping us understand the nature of the combinatorial problem arising from our chording procedure in Section 4.3. The authors also thank Hsien-Chih Chang, K. Alex Mills, and Amir Nayyeri for helpful discussions. Finally, the authors thank the reviewers for their valuable comments.

References

[ABC+05] I. Abraham, Y. Bartal, T.-H. Chan, K. Dhamdhere, A. Gupta, J. Kleinberg, O. Neiman, and A. Slivkins. Metric embeddings with relaxed guarantees. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 83–100, 2005.

[BDST08] J. Brickell, I. Dhillon, S. Sra, and J. Tropp. The metric nearness problem. *SIAM J. Matrix Analysis Applications*, 30(1):375–396, 2008.

[Bou85] J. Bourgain. On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52(1-2):46–52, 1985.

[CDG+09] T.-H. Chan, K. Dhamdhere, A. Gupta, J. Kleinberg, and A. Slivkins. Metric embeddings with relaxed guarantees. *SIAM J. Comput.*, 38(6):2303–2329, 2009.

[CGGS01] F. Chung, M. Garrett, R. Graham, and D. Shallcross. Distance realization problems with applications to internet tomography. *J. Comput. Syst. Sci.*, 63(3):432–448, 2001.

[Chr76] N. Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie Mellon University, 1976.

[CR12] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Commun. ACM*, 55(6):111–119, June 2012.

[IM04] Piotr Indyk and Jiří Matoušek. Low-distortion embeddings of finite metric spaces. In *Handbook of Discrete and Computational Geometry*, pages 177–196. CRC Press, 2004.

[KR08] S. Khot and O. Regev. Vertex cover might be hard to approximate to within 2-epsilon. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[LLR95] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.

[Mat13] J. Matoušek. *Lecture notes on metric embeddings*, 2013. Available at: http://kam.mff.cuni.cz/~matousek/ba-a4.pdf.

[SWW17] A. Sidiropoulos, D. Wang, and Y. Wang. Metric embeddings with outliers. In *Proc. Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 670–689, 2017.

A Matching Lower Bound

The following is a matching lower bound to the combinatorial problem from Lemma 4.5.

LEMMA A.1. *Let $G = (V; E)$ be a graph whose edge set E is the union of the edges of a collection of 4-cycles, $\mathbb{C}$, such that no two 4-cycles in $\mathbb{C}$ can share a chord. (Note this applies to all chords in the complete graph on V, i.e. regardless of whether they appear in E.) Then in the worst case $|\mathbb{C}| = \quad (m^{4=3})$, where $m = |E|$.*

Proof. Construct a graph $G = (V; E)$, where V is the disjoint union of four sets of vertices X_1, X_2, X_3, and X_4, each containing exactly t vertices, where t is a value to be determined shortly. The edge set E is sampled as follows. For $1 \le i \le 4$, for each pair $(u; v) \in X_i \times X_{i+1}$ (where $X_5 = X_1$), the edge $(u; v)$ is sampled into E independently with probability

$$p = \frac{m}{8t^2} \ll 1:$$

Let $\mathbb{C}$ be the set of 4-cycles de ned by E. Any cycle $C \in \mathbb{C}$ must contain exactly one vertex from each of X_1, X_2, X_3, and X_4. The probability that any quadruple of vertices $(i_1; i_2; i_3; i_4) \in X_1 \times X_2 \times X_3 \times X_4$ de nes a cycle in $\mathbb{C}$ is p^4. As such, the expected size of $\mathbb{C}$ is

$$= p^4 t^4 = \left(\frac{m}{8t^2}\right)^4 t^4 = \left(\frac{m}{8t}\right)^4:$$

Consider such a cycle $C = (i_1; i_2; i_3; i_4)$ that we know exists in the graph. Any cycle which shares the chord $\{i_1; i_3\}$ with C clearly shares the vertices i_1 and i_3. Now such a cycle either shares a third vertex or not. The expected number of cycles which share the chord $\{i_1; i_3\}$ and no other vertex is at most $t^2 p^4$ The expected number of cycles which share the chord $\{i_1; i_3\}$ and one other vertex is at most $2tp^2$. Let X_C be a random variable denoting the number of cycles sharing either chord (i.e., $\{i_1; i_3\}$ or $\{i_2; i_4\}$) with C. Assuming $tp^2 \le 1$ we have,

$$\mathbf{E}[X_C \mid C \text{ exists}] \le 2(2tp^2 + t^2p^4) \le 2(2 + tp^2)tp^2 \le 6tp^2:$$

Assume further that $6tp^2 \le 1=10$, then by Markov's inequality we have

$$(C) = \mathbf{Pr}[\text{no cycle shares a chord with } C \mid C \text{ exists}]$$
$$= 1 - \mathbf{Pr}[X_C \ge 1 \mid C \text{ exists}] \ge \frac{9}{10}:$$

Let Y be a random variable denoting the number of cycles that exists in the graph and don't share a chord with any other cycle that exists in the graph. We have that

$$= \mathbf{E}[Y]$$
$$= \sum_C \mathbf{Pr}[(\text{no cycle shares chord with } C) \cap (C \text{ exists})]$$
$$= \sum_C (C) \cdot \mathbf{Pr}[C \text{ exists}] \ge \frac{9}{10} \quad :$$

Note that as was the expected number of cycles overall, this implies $= \mathbf{E}[Y] = ( )$.

Recall that we assumed $6tp^2 \le 1=10$, which plugging in for p becomes,

$$\frac{1}{10} \ge 6t \left(\frac{m}{8t^2}\right)^2 \ge \frac{3}{32}\frac{m^2}{t^3} \quad \Longrightarrow \quad t \ge (30=32)^{1=3} m^{2=3}:$$

Thus setting $t = m^{2=3}$ (which up to constants minimizes) implies the expected number of cycles that do not share a diagonal is

$$= \quad (p^4 t^4) = \quad \frac{m}{t^2}^{\ 4} t^4 \quad = \quad \frac{m^4}{t^4}$$

$$= \quad \frac{m^4}{m^{8=3}} \quad = \quad m^{4=3} \ :$$

On the other hand, the expected number of edges is $4t^2 p = m=2$, and moreover by the Cherno bound with high probability is at most m. Thus by the probabilistic method there exists a graph where $|E| \leq m$ and the number of 4-cycles which don't share a chord is $(m^{4=3})$. (Note to match the lemma statement, in the above construction one should only keep edges which were in cycles that did not share a chord with any other cycle.)