

Handling Correlated Rounding Error via Preclustering: A 1.73-approximation for Correlation Clustering

Vincent Cohen-Addad
Google Research
cohenaddad@google.com

Euiwoong Lee*
University of Michigan
euiwoong@umich.edu

Shi Li
Nanjing University
shili@nju.edu.cn

Alantha Newman
Université Grenoble Alpes
alantha.newman@grenoble-inp.fr

Abstract

We consider the classic Correlation Clustering problem: Given a complete graph where edges are labelled either + or −, the goal is to find a partition of the vertices that minimizes the number of the +edges across parts plus the number of the −edges within parts. Recently, Cohen-Addad, Lee and Newman [CLN22] presented a 1.994-approximation algorithm for the problem using the Sherali-Adams hierarchy, hence breaking through the integrality gap of 2 for the classic linear program and improving upon the 2.06-approximation of Chawla, Makarychev, Schramm and Yaroslavtsev [CMSY15].

We significantly improve the state-of-the-art by providing a 1.73-approximation for the problem. Our approach introduces a preclustering of Correlation Clustering instances that allows us to essentially ignore the error arising from the *correlated rounding* used by [CLN22]. This additional power simplifies the previous algorithm and analysis. More importantly, it enables a new *set-based rounding* that complements the previous roundings. A combination of these two rounding algorithms yields the improved bound.

*Supported in part by NSF grant CCF-2236669 and Google.

1 Introduction

Clustering is a classic problem in unsupervised machine learning and data mining. Given a set of data elements and pairwise similarity information between the elements, the goal of clustering is to find a partition of the data elements such that elements in the same clusters are pairwise similar while elements in different clusters are pairwise dissimilar. Introduced by Bansal, Blum and Chawla [BBC04], Correlation Clustering has become one of the most widely studied formulations for clustering. The input of the problem consists of a complete graph $(V, E^+ \uplus E^-)$, where $E^+ \uplus E^- = \binom{V}{2}$, E^+ representing the so-called *positive* edges and E^- the so-called *negative* edges. The goal is to find a partition of the vertex set so as to minimize the number of *unsatisfied* edges, namely the number of the negative edges uv where u and v are in the same cluster plus the number of the positive edges uv where u and v are in different clusters. Here, the vertex set represents the elements to cluster while positive edges represent pairs of similar elements and negative edges pairs of dissimilar elements. The above formulation is very basic and has thus led the Correlation Clustering problem to encompass a variety of applications from finding clustering ensembles [BGU13], duplicate detection [ARS09], community mining [CSX12], disambiguation tasks [KCMNT08], to automated labelling [AHK⁺09, CKP08] and many more.

The problem is known to be NP-hard, and so the focus has been on designing approximation algorithms for the problem. In their seminal paper, Bansal, Blum and Chawla [BBC04] gave an $O(1)$ -approximation algorithm for the problem, which was later improved by Charikar, Guruswami and Wirth [CGW05] to a 4-approximation, obtained by rounding the natural linear program (LP) relaxation for the problem. Charikar, Guruswami and Wirth [CGW05] also showed that the problem is APX-Hard. Soon after, Ailon, Charikar and Newman [ACN08] introduced an influential *pivot-based* algorithm, that leads to a combinatorial 3-approximation and a LP-based 2.5-approximation. The pivot-based algorithms continue to inspire new results for Correlation Clustering in different settings (see Section 1.3 for more details). Ten years after the results of [ACN08], Chawla, Makarychev, Schramm and Yaroslavtsev [CMSY15] further improved the LP rounding scheme and obtained a 2.06-approximation, nearly matching the LP integrality gap of 2. Since the best known approximation results have been obtained through LP rounding techniques and the LP has an integrality gap of 2, this bound seemed to be an important roadblock in the direction of getting better approximation bounds. Recently, Cohen-Addad, Lee and Newman [CLN22] broke through this barrier using $O(1/\varepsilon^2)$ rounds of the Sherali-Adams hierarchy on top of the standard LP and rounding the resulting fractional solution to obtain a $(1.994 + \varepsilon)$ -approximation algorithm.

The result of [CLN22] shows the importance of hierarchies to decrease the approximation ratio for the problem. In fact, for a large constant number of rounds of the Sherali-Adams hierarchy, we do not know a lower bound on the integrality gap of the resulting LP, which may even perhaps yield optimal results under $P \neq NP$ or the Unique Games Conjecture. Presumably, the approximation ratio of $(1.994 + \varepsilon)$ obtained through rounding the Sherali-Adams relaxation does not reflect the actual power of the hierarchies, but rather the limitations of the current rounding approaches and of their analysis. Thus, to improve over the $(1.994 + \varepsilon)$ -approximation one might have to go beyond the pivot-based rounding framework introduced by [ACN08], further developed by [CMSY15] and [CLN22], which has been the only known way of obtaining a better than 3-approximation for the problem for the last 15 years. This is what we propose in this paper.

1.1 Our Results

We present a drastic improvement over the result of Cohen-Addad, Lee and Newman [CLN22] by showing a $(1.73 + \varepsilon)$ -approximation to the problem using the Sherali-Adams hierarchy.

Theorem 1. (Main Result) *For any $\varepsilon > 0$, there exists a $(1.73 + \varepsilon)$ -approximation algorithm for Correlation Clustering with running time $n^{O(1/\text{poly}(\varepsilon))}$.*

In addition to the above bound, our contributions are the following. Our new result departs from previous work in several ways:

1. We provide a preclustering step that identifies pairs of vertices that “clearly” belong to the same cluster/do not belong to the same cluster in an optimum solution. This is achieved by computing a

group of vertices whose contribution to the objective function in an optimum solution is tiny compared to the number of incident +edges, and results in a preclustering where the number of “undecided” pairs can be upper bounded in terms of the cost of the optimal clustering; we hence get a useful lower bound on the cost contribution of the remaining instance.

2. Equipped with this, we provide a new *set-based rounding* approach inspired by the work of Kleinberg and Tardos [KT02] for the Uniform Metric Labeling problem. To the best of our knowledge, it is the first application of the ideas from [KT02] to Correlation Clustering. The original work [KT02], when adapted to Correlation Clustering, creates a variable y_S for each $S \subseteq V$ that can possibly become a cluster and sample a set with probability proportional to y_S 's (hence the name *set-based*).

Of course, it is impossible to create a variable for each subset, and we therefore need to use the power of the Sherali-Adams hierarchy and the *correlated rounding* technique introduced by Raghavendra and Tan [RT12] to extend this approach to output arbitrary-size subsets when needed. This comes at a price: there is an additive rounding error proportional to the total number of vertices squared, which is prohibitive when the optimal value is $o(n^2)$. However, this is where the preclustering step comes to the rescue; we use the correlated rounding only for the “undecided pairs” whose number is small compared to opt, so that this additive rounding error is negligible.

3. This way of handling error from the correlated rounding also provides a refined analysis of the Cohen-Addad, Lee and Newman [CLN22] rounding, which combines the classical *pivot-based rounding* with the correlated rounding. As their main technical complications arise from the rounding error, our preclustering step considerably simplifies their analysis. While following the same triangle-based analysis of [CLN22], we introduce a new edge-by-edge charging argument that helps this algorithm nicely complement the set-based rounding above.
4. In the final step, we combine the two above rounding approaches to show that the bad cases for one type of edges (E^+ or E^-) are the good cases of the other; our combined rounding gives the desired bound of $1.73 + \varepsilon$.

We next present a more detailed overview of the techniques.

1.2 Our Techniques

To understand in more details our contribution and the new techniques introduced, we need to provide a brief summary of the previous approaches. We first recall the classic LP relaxation (whose integrality gap is known to be in $[2, 2.06]$). There is a variable x_{uv} for each pair of vertices whose intended value is 1 if u, v are not in the same cluster and 0 otherwise. The goal is thus to minimize $\sum_{uv \in E^+} x_{uv} + \sum_{uv \in E^-} (1 - x_{uv})$, under the classic triangle inequality constraint: $\forall u, v, w, x_{uv} \leq x_{uw} + x_{wv}$.

To obtain the 2.5-approximation algorithm, Ailon, Charikar and Newman [ACN08] designed a pivot-based rounding method that proceeds as follows: (1) Pick a random vertex p , called the *pivot*, and create the cluster containing p ; and (2) Recurse on the rest of the instance (the vertices not in the cluster). This hence builds the clustering in a sequential manner. Once we commit to this scheme, the main design question that remains is how to construct the cluster containing the pivot. The solution of [ACN08] was to go over all the other vertices and for each other vertex u place it in the cluster of the pivot p with probability $(1 - x_{pu})$, independently of the other random decisions made for the other vertices.

This was further improved by Chawla, Makarychev, Schramm and Yaroslavtsev [CMSY15] who kept the same scheme but provided a better rounding approach: For +edges, they replaced the probability $(1 - x_{pu})$ with a new rounding function f^+ to apply to the quantity $(1 - x_{pu})$ yielding the probability of incorporating u in p 's cluster. Similar to the analysis of [ACN08], the analysis of the rounding scheme was triangle-based: The analysis is a charging scheme that charges the cost paid by each pair u, v to the triangle p, u, v where p is the pivot which decided edge u, v . Then the crux of the analysis is to show that for any triangle p, u, v , the charge is bounded compared to the LP cost.

The approach of [CLN22], while using the same pivot-based rounding and triangle-based analysis, introduces two twists. The first twist is the usage of *correlated rounding* based on the Sherali-Adams hierarchy.

Namely, given a pivot p , the set of +neighbors of p that join the cluster of p is chosen in a correlated manner, using the techniques of Raghavendra and Tan [RT12]. Concretely, the Sherali-Adams hierarchy provides variables of the form y_S for any constant-sized set of vertices S that indicates the probability that all the vertices in S are in the same cluster.¹ Given a pivot p , the correlated rounding then allows to sample in such a way that the probability that u, v join the cluster of p is $y_{puv} \pm \varepsilon$, where ε is an arbitrarily small constant. An important issue is that this additive ε error is in some cases not a relative error (think of $(u, v) \in E^+$ and $x_{uv} = 0$). Dealing with these additive errors is one of the most technical parts of the contribution of [CLN22] and perhaps a limitation to getting an improved approximation ratio (e.g., special attention must be paid to edges with values close to 0 and 1 in both the algorithm and in the analysis). This first twist allows to bring the approximation ratio down to 2 but not further; the ratio of some of the triangles is 2. To bypass this bound, the second twist in the approach is to use the properties of the Sherali-Adams hierarchy to argue that one cannot pack too many bad triangles without creating some good triangles (for which the ratio is smaller than 2), hence bringing down the ratio to $(1.994 + \varepsilon)$ via a charging argument. We next describe how we go beyond previous work.

The High-Level Approach The governing high-level intuition for what we are trying to achieve is this. Given a pivot vertex p , the key property of the correlated rounding (see Lemma 22 for a restatement) ensures that we can sample a set C from a set of vertices S both being of arbitrary size (importantly not necessarily constant) such that pairs u, v ends up in S with probability y_{puv} on average, up to losing an additive $\varepsilon|S|^2$ error in the cost. This is one of the key properties of the Sherali-Adams hierarchy that can be used to obtain approximation schemes for dense CSPs, in particular Dense Max-Cut [dIVKM07, YZ14]. In light of the previous approaches, this is a very desirable property: We could repeatedly construct a cluster using this tool and the error would only be coming from the fact that the clusters are constructed sequentially.

The main issue we need to face with the above plan of attack is that the additive error of $\varepsilon|S|^2$, which we pay when we create a cluster C , may end up being much larger than the cost of the optimum solution. In fact, an optimum solution may have cost zero. So our goal is the following: (1) precluster the instance so that the cost of the optimal solution becomes relatively large compared to the size of the “unsure part”, and (2) use correlated rounding only for the unsure part of the instance so that the error from the correlated rounding is negligible.

A New Preclustering Our first conceptual contribution is a preclustering of the instance that aims at identifying “clear clusters”. More concretely, our goal is to make sure that if we apply the correlated rounding to create a cluster of size s , then we can accommodate an additive cost of εs^2 . Thus, consider the clusters of the optimum (integral) solution; when is the cost contribution of such a cluster S significantly smaller than s^2 , say εs^2 ? For this to happen, it must be that the number of $-$ edges internal to S is small compared to $\binom{s}{2}$ and that the number $+$ edges with exactly one endpoint in S is also much smaller than $\binom{s}{2}$. This cluster is thus almost a clique with tiny $+$ edge-expansion; we formalize this through the notion of an *atom* in Section 3.

This means that for a very large fraction of the pairs u, v of vertices in an atomic cluster S , the $+$ neighborhood of u and v are nearly identical (up to a tiny fraction). This is exactly what the notion of *agreement* introduced by Cohen-Addad, Lattanzi, Mitrovic, Norouzi-Fard, Parotsidis and Tarnawski [CLM+21] (see Definition 10) was designed for. Indeed, in this work, the authors presented a massively-parallel constant factor approximation algorithm. The constant obtained there is larger than 500, but the key idea is that it is enough to identify the atoms, since all the other clusters C pay a cost of at least $\varepsilon \sum_{u \in C} d_u$, where d_u is the number of $+$ neighbors of u , and therefore making all the remaining vertices singleton clusters would be enough to get an $O(1/\varepsilon)$ -approximation.

We thus re-use the Algorithm 1 of [CLM+21] and show that it correctly identifies the atoms. [CLM+21] then showed that there exists a $(1 + \varepsilon)$ -approximate solution \mathcal{C}^* , such that each non-singleton cluster output by the Algorithm 1 of [CLM+21], now referred to as an atom, is fully contained within one cluster of \mathcal{C}^* . This

¹In which case, $y_{uv} = 1 - x_{uv}$.

immediately allows us to treat each atom as a single (weighted) vertex, or in other words enforce $x_{uv} = 0$ for all pairs u, v in the same atom.

The next key step is that for each vertex u , we can identify a set of so-called *admissible* edges, which induce a neighborhood N'_u of size at most $d_u/\varepsilon^{O(1)}$. We then show that there exists a near-optimum solution such that if u, v is not an atomic or an admissible pair, then u and v are not in the same cluster. For such pairs u, v , we can immediately set $x_{uv} = 1$ in the LP, because we know in advance that they must be placed in separate clusters. This immediately reduces the uncertainty of the fractional LP solution: for any vertex u , the number of pairs u, v for which the LP can have a fractional value (i.e., a value not in $\{0, 1\}$) is at most the number of admissible pairs which we show through a delicate argument is at most $\sum_{u \in V} d_u/\varepsilon^{O(1)}$.

The final step of the preclustering is then to show that the cost of an integral solution under the above constraints is at least $\varepsilon^{O(1)}$ times the number of admissible pairs. Combined with the above argument, we can show that this enables the use of correlated rounding while only losing an overall $(1 + \varepsilon)$ multiplicative approximation, which is crucial because both of our rounding algorithms below use the correlated rounding.

We believe that this preclustering will be of particular interest to future work on improving the approximation ratio given that it provides a much more structured instance and it remains completely independent from the rounding process.

A New Set-Based Rounding Our other major contribution is the following *set-based* rounding, whose performance complements the previous ones [ACN08, CMSY15, CLN22]. Namely, give a solution $x \in [0, 1]^{(V)}$ to some LP relaxation, our goal is to obtain (as stated in Theorem 7) a clustering \mathcal{C} with

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \frac{2x_{vw}}{1+x_{vw}} + \sum_{vw \in E^-} \frac{1-x_{vw}}{1+x_{vw}} + O(\varepsilon) \cdot |E_{\text{adm}}|,$$

where $\text{cost}(\mathcal{C})$ is the cost of the clustering \mathcal{C} . The set E_{adm} refers to the set of admissible edges defined in the preclustering.

To achieve this we use the ideas from the 2-approximation algorithm for the Uniform Metric Labeling problem by Kleinberg and Tardos [KT02]. Consider a special case of the Correlation Clustering problem where we are promised that the optimum clustering has maximum cluster size $t = O(1)$. Then, in our LP, we could simply have a variable z_S for every set S of size at most t , indicating if S is a cluster or not. For every $v \in V$, we have $\sum_{S \ni v} z_S = 1$. In each iteration of the rounding algorithm, we randomly choose a set S with probability z_S/Z , where $Z := \sum_S z_S$. So an edge vw is decided with probability $\frac{1+x_{vw}}{Z}$. For an edge $vw \in E^+$, it incurs a cost with probability $\frac{2x_{vw}}{Z}$; for an edge $vw \in E^-$, the probability is $\frac{1-x_{vw}}{Z}$. This gives expected cost of $\frac{2x_{vw}}{1+x_{vw}}$ and $\frac{1-x_{vw}}{1+x_{vw}}$ for + and -edges vw respectively, proving the inequality.

The most prominent challenge is thus to enable this approach to work beyond the assumption that the optimum clusters have constant size. This may seem impossible at first since if such a technique existed in general graph partitioning, this would provide an approximation bound that would yield a result believed to be unlikely (e.g., a 2-approximation for Minimum Bisection). Thus, we need to exploit the particularities of the Correlation Clustering problem. To achieve this, we first enrich the program with variables of the form y_S^s , for all $s \in [n]$, and subset S of $1/\varepsilon^{O(1)}$ vertices. Here y_S^s aims at representing the number of clusters of size s containing S as a subset, and should be in $\{0, 1\}$ in an integral solution when $S \neq \emptyset$. Then, y_S now represents the number of clusters (of various sizes) containing S as a subset, and should also be in $\{0, 1\}$ in an integral solution. It thus indicates whether S is a subset of a cluster.

We next show that this is enough to achieve the above bound. To do so, we provide the following rounding approach: We first sample a size s based on the y_0^s probabilities, then a pivot a that will go in a cluster of size s based on the y_a^s probabilities (correcting for the fact that we are aiming to build a cluster of size s), and then use the correlated rounding to complete the cluster. We show, perhaps surprisingly, that this is enough to achieve the bounds offered by the approach of [KT02], at the price of an extra additive error incurred by the correlated rounding, which, as we show, results in an additive $O(\varepsilon) \cdot |E_{\text{adm}}|$ additive error. This error is tolerable thanks to the preclustering step.

A Refined and Cleaner Analysis of the Pivot-Based Rounding This preclustering immediately makes the analysis of [CLN22] rounding much simpler. All the corner cases that required a lot of work to handle the additive error can be removed: The concepts of short and long edges (edges uv for which x_{uv} was close to 0 or 1) can be completely forgotten and the rounding can be simplified. In fact, [CLN22] proposed a simpler and better so-called “ideal” analysis of their rounding if the additive error ε can be assumed to be 0 (which of course cannot be achieved with $o(n)$ rounds of Sherali-Adams) and our preclustering allows to immediately recover this ideal analysis, modulo a simple cleaning step to handle the atoms. This yields a very simple 2-approximation rounding scheme.

We then refine the bound obtained by [CLN22] to express it as a per-edge approximation. Concretely, we show that the output clustering \mathcal{C} satisfies the following guarantees (as stated in Theorem 8):

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \min\{1.515 + x_{vw}, 2\} \cdot x_{vw} + 2 \sum_{vw \in E^-} (1 - x_{vw}) + O(\varepsilon) \cdot |E_{\text{adm}}|.$$

Combining Two Roundings The two above bounds are simply combined as follows: We round the LP with the pivot-based rounding of [CLN22] with some carefully chosen probability and the new rounding approach with the remaining probability; for intuition, note that $-$ edges have a 1-approximation in the first rounding and 2 in the second rounding, while $+$ edges with small x value have a 2-approximation in the first rounding but close to a 1.5-approximation in the second. Therefore, by appropriately combining the two roundings, we can show that every edge has a 1.73-approximation in expectation. The preclustering ensures that the additive $O(\varepsilon) \cdot |E_{\text{adm}}|$ terms are only an $O(\text{poly}(\varepsilon))$ fraction of the optimum cost and can thus be ignored.

Round-or-Cut Framework In order to perform the set-based rounding and pivot-based rounding *simultaneously*, our overall algorithmic framework after the preclustering uses the round-or-cut paradigm. In a typical rounding algorithm, we are given a metric $x \in [0, 1]^{\binom{V}{2}}$ over V along with some auxiliary variables, obtained from solving an LP relaxation. We then round x into some integral clustering with a small cost. This structure works for the pivot based rounding algorithm: we randomly generate a cluster C according to the algorithm, remove C , focus on the LP relaxation restricted to $V \setminus C$ (i.e., we reduce the LP solution), and repeat until we clustered all vertices in V . However, the adaptive feature of our set-based rounding algorithm, which solves an LP extending x in every iteration, makes this one-shot strategy hard to implement.

Instead, we design a rounding algorithm with violation detection (Theorem 5). In the algorithm, we are given the core vector $x \in [0, 1]^{\binom{V}{2}}$ only. The algorithm proceeds in iterations. In each iteration, we try extend x to a vector (x, y) with the auxiliary variables y , which depend on the remaining set V' of vertices. If the extension is successful, then we can proceed with the iteration by constructing the cluster C and removing C . Otherwise, we find a hyperplane that separates x from the convex hull of all integral clusterings, and return it. The well-known property of the ellipsoid algorithm [GLS12] will ensure that we will eventually find a desired clustering.

1.3 Further Related Work

The Correlation Clustering problem has also been studied in the weighted case, where each pair of vertices has an associated weight and unsatisfied edges contribute a cost proportional to their weight to the objective. Unfortunately, this version of the problem is equivalent to the Multicut problem in terms of approximation guarantee and so an $O(\log n)$ -approximation is known [DEFI06] and improving this bound significantly would be a major breakthrough. Moreover, no polynomial-time constant factor approximation algorithm exists assuming the Unique Game Conjecture [CKK⁺06].

The maximization version of the problem, where the goal is to maximize the number of satisfied edges, has also been studied. A PTAS for the problem was given by Bansal, Blum and Chawla [BBC04], and a .77-approximation for the weighted case was given by Charikar, Guruswami and Wirth [CGW05] and Swamy [Swa04].

In the unweighted case, a PTAS exists when the number of clusters is a fixed constant [GG06, KS09]. A lot of work has also been devoted to the minimization version of Correlation Clustering in other computation models: online [MSS10, LMV+21, CLMP22], more practical settings, and in particular distributed or parallel [CDK14, ACG+15, CLM+21, PPO+15, CCMU21, Vel22, VGW18], differential-privacy [BEK21, Liu22, CFL+22]. Related to our results is the work of Cohen-Addad, Lattanzi, Mitrovic, Norouzi-Fard, Parotsidis and Tarnawski [CLM+21] whose approach is useful to our preprocessing step, see also the work of Assadi and Wang [AW22]. The above works have been improved by Behnezhad, Charikar, Ma, and Tan [BCMT22, BCMT23] by showing how to extend the combinatorial pivot approach to the distributed or streaming settings. Note that recently, connections between metric embeddings into ultrametric have been established and Correlation Clustering plays a central role in the current best known approximation algorithms [CDK+21, CFLM22]. In fact, our new bound (marginally) improves the constant factor approximation of Cohen-Addad, Das, Kipouridis, Parotsidis and Thorup [CDK+21].

2 Overall Framework

In this section, we present our overall framework to achieve a $(1.73 + \varepsilon)$ -approximation algorithm. On the way, we will also introduce our main technical results for the preclustering (Theorem 4), the set-based rounding (Theorem 7), and the pivot-based rounding (Theorem 8), which will be proved in Section 3, 4, and 5 respectively. We begin with some definitions related to our preclustering step.

Definition 2. Given a Correlation Clustering instance $(V, E^+ \cup E^-)$, a preclustered instance is defined by a pair $(\mathcal{K}, E_{\text{adm}})$, where \mathcal{K} is a family of disjoint subsets of V (not necessarily a partition), and $E_{\text{adm}} \subseteq \binom{V}{2}$ is a set of pairs such that for every $uv \in E_{\text{adm}}$, at least one of u and v is not in $\bigcup_{K \in \mathcal{K}} K$.

Each set $K \in \mathcal{K}$ is called an atom. We use $V_{\mathcal{K}} := \bigcup_{K \in \mathcal{K}} K$ to denote the set of all vertices in atoms. A pair (u, v) between two vertices u, v in a same $K \in \mathcal{K}$ is called an atomic edge. A pair that is neither an atomic nor an admissible edge is called a non-admissible edge.

Therefore, in a preclustered instance, the set $\binom{V}{2}$ is partitioned into atomic, admissible and non-admissible edges. By the definition of E_{adm} , a pair (u, v) between two different atoms is non-admissible.

Definition 3. Given a preclustered instance $(\mathcal{K}, E_{\text{adm}})$ for some Correlation Clustering instance $(V, E^+ \cup E^-)$, a partition \mathcal{C} of V (also called a clustering) is called good with respect to $(\mathcal{K}, E_{\text{adm}})$ if

- u and v are in the same set (or cluster) in \mathcal{C} for an atomic edge (u, v) , and
- u and v are not in the same set (or cluster) in \mathcal{C} for a non-admissible edge (u, v) .

That is, a good clustering can not break an atom, or join a non-admissible pair. As a result, two atoms can not be in the same cluster as the edges between them are non-admissible. The main theorem we prove for our preclustering step in Section 3 is the following:

Theorem 4. (Preclustering) For any sufficiently small $\delta > 0$, there exists a polynomial-time algorithm that, given a Correlation Clustering instance $(V, E^+ \cup E^-)$ with the optimal value opt_0 , produces a preclustered instance $(\mathcal{K}, E_{\text{adm}})$ such that

- there exists a good clustering whose cost is at most $(1 + \delta)\text{opt}_0$, and
- $|E_{\text{adm}}| \leq O(\text{opt}_0/\delta^{12})$.

We remark that, after the preclustering, the benchmark clustering we are comparing to is a good clustering, but our algorithm does not need to find a good clustering. For example, the clustering we construct might join a non-admissible pair, or two atoms. But we guarantee that an atom will not be broken by the clustering.

Given the desired error parameter $\varepsilon_0 > 0$ for the overall approximation factor in Theorem 1, let $\delta := \varepsilon_0/4$, $\varepsilon := \Theta(\delta^{12}\varepsilon_0) = \Theta(\varepsilon_0^{13})$ and perform the preclustering with parameter δ (let opt be the cost of the best good

clustering with respect to the preclustering). Then, producing a clustering \mathcal{C} whose cost (denoted as $\text{cost}(\mathcal{C})$) is at most $1.73\text{opt} + \varepsilon|E_{\text{adm}}|$ guarantees that

$$\text{cost}(\mathcal{C}) \leq 1.73\text{opt} + \varepsilon|E_{\text{adm}}| \leq 1.73(1 + \delta)\text{opt} + O(\varepsilon/\delta^{12})\text{opt}_0 \leq (1.73 + \varepsilon_0)\text{opt}_0.$$

Therefore, given a preclustered instance $(V, E^+ \cup E^-)$ together with $(\mathcal{K}, E_{\text{adm}})$, it suffices to compute a clustering \mathcal{C} with $\text{cost}(\mathcal{C}) \leq 1.73\text{opt} + \varepsilon|E_{\text{adm}}|$.

Our algorithm proceeds via the round-or-cut framework. Every clustering corresponds to a 0/1-valued metric over V , where the distance between u and v indicates if u and v are cut in the clustering or not. We let $\mathcal{P} \subseteq [0, 1]^{\binom{V}{2}}$ denote the convex hull of the metrics for all good clusterings. The central piece of the algorithm for the proof of Theorem 1 is a rounding algorithm \mathcal{A} with violation detection. Formally, we prove the following theorem:

Theorem 5. *Given a metric $x \in [0, 1]^{\binom{V}{2}}$ over V , for which any two vertices in a same atom have distance 0, and the two end points of any non-admissible edge have distance 1, and $\varepsilon > 0$, there is an $n^{O(1/\varepsilon^4)}$ time algorithm \mathcal{A} that outputs one of the following two things:*

- a clustering for the instance whose cost is at most $1.73 \cdot \text{cost}(x) + \varepsilon|E_{\text{adm}}|$,
- a hyperplane separating x and \mathcal{P} (i.e., a vector $w \in \mathbb{R}^{\binom{V}{2}}$ and $b \in \mathbb{R}$ such that $w^T x' \geq b$ for every $x' \in \mathcal{P}$, but $w^T x < b$). Sometimes we also simply call the (w, b) pair separating x and \mathcal{P} a separation plane for x .

It is well known [GLS12] that once we have the algorithm \mathcal{A} in Theorem 5, we can combine it with the ellipsoid method to find the desired approximate solution for the preclustered instance in polynomial time, which proves Theorem 1, with a final running time of $n^{O(1/\varepsilon^4)} = n^{O(1/\varepsilon_0^2)}$. By enumeration or binary search, we assume we are given the value of opt , and our goal is to find a clustering with cost at most $1.73\text{opt} + \varepsilon|E_{\text{adm}}|$. Let $\mathcal{P}' := \mathcal{P} \cap \{x : \text{cost}(x) \leq \text{opt}\}$ be the convex hull of good clusterings of cost at most opt . \mathcal{P}' is non-empty since there exists a good clustering of cost at most opt . Then we take an ellipsoid containing the \mathcal{P}' . In every iteration, we take the center x of the ellipsoid, and run the algorithm in Theorem 5 over this x . If the algorithm returns a clustering, then the cost of the clustering is at most $1.73\text{opt} + \varepsilon|E_{\text{adm}}|$ and we are done. Otherwise, it returns a hyperplane separating x and \mathcal{P}' , which breaks the ellipsoid into two parts, one containing x , and the other containing \mathcal{P}' . We define a new ellipsoid that contains the latter part, and repeat. Since \mathcal{P}' is nonempty, the algorithm will successfully output a desired clustering in polynomial number of iterations. Thus our goal in this section becomes to prove Theorem 5.

A useful tool is the following lemma that shows we can try to extend x to any domain. Suppose D is any domain of variables, and $\mathcal{Q} \subseteq [0, 1]^{\binom{V}{2}} \times \mathbb{R}^D$ is a polytope such that the projection of \mathcal{Q} to coordinates in $\binom{V}{2}$ contains \mathcal{P} . If x is also in the projection, namely if we have $(x, y) \in \mathcal{Q}$ for some $y \in [0, 1]^D$, then we can successfully extend x to (x, y) and use y in our rounding algorithm. Otherwise, we can output a separation plane for x using the following lemma:

Lemma 6. *Suppose x is not in the projection of \mathcal{Q} ; namely for every $y \in \mathbb{R}^D$, we have $(x, y) \notin \mathcal{Q}$. Then in time polynomial in the description of \mathcal{Q} , we can find a separation plane (w, b) for x .*

Proof. Consider the LP $(A, A') \begin{pmatrix} x \\ y \end{pmatrix} \leq c$ that defines the polytope \mathcal{Q} , where x corresponds to variables in $\binom{V}{2}$ and y corresponds to variables in D , and the number of columns of A is the same as the number of rows of x . Now suppose for a fixed x , the LP is infeasible. That is, the LP $A'y \leq c - Ax$ (with variables y) is infeasible. By LP duality, we can find a vector u with non-negative entries such that $u^T A' = 0$ and $u^T (c - Ax) = -1$. On the other hand, for every x' that is in the projection of \mathcal{Q} , we have $u^T (c - Ax') \geq 0$: There exists some y with $(A, A') \begin{pmatrix} x' \\ y \end{pmatrix} \leq c$, which is equivalent to $A'y \leq c - Ax'$, which implies $0 = u^T A'y \leq u^T (c - Ax')$. This gives us a separation plane between x and \mathcal{P} . Moreover, the plane can be found in time polynomial in the size of the LP for \mathcal{Q} . \square

In the algorithm that proves Theorem 5, we generate two clusterings using two different procedures (*set-based rounding* and *pivot-based rounding*) and output the better of the two clusterings. The properties of the two procedures are described in the following theorems. In both theorems, the input is the same as that from Theorem 5.

Theorem 7. (Set-based Rounding) *There is an $n^{O(1/\varepsilon^4)}$ -time procedure \mathcal{A}_{set} that either outputs a separation plane for x , or a clustering \mathcal{C} such that*

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \frac{2x_{vw}}{1+x_{vw}} + \sum_{vw \in E^-} \frac{1-x_{vw}}{1+x_{vw}} + \varepsilon \cdot |E_{\text{adm}}|.$$

Theorem 8. (Pivot-based Rounding) *There is an $n^{O(1/\varepsilon^2)}$ -time procedure $\mathcal{A}_{\text{pivot}}$ that either outputs a separation plane for x or a clustering \mathcal{C} such that*

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \min\{1.515 + x_{vw}, 2\} \cdot x_{vw} + 2 \sum_{vw \in E^-} (1 - x_{vw}) + \varepsilon \cdot |E_{\text{adm}}|.$$

It is simple to show that combining the two procedures proves Theorem 5.

Proof of Theorem 5. Suppose that given $x \in [0, 1]^{\binom{V}{2}}$, both \mathcal{A}_{set} and $\mathcal{A}_{\text{pivot}}$ return a solution \mathcal{C}_1 and \mathcal{C}_2 respectively; otherwise a separation plane is found.

Consider the cost of \mathcal{C}_1 times 0.42 plus the cost of \mathcal{C}_2 times 0.58. Then the approximation ratio for any +edge is at most

$$\max_{x \in [0, 1]} \left(0.42 \frac{2}{1+x} + 0.58 \min(1.515 + x, 2) \right) \leq 1.7257, \quad (1)$$

and the ratio for -edge is at most $0.42 \cdot 1 + 0.58 \cdot 2 = 1.58$. Therefore, the cost of the better one is at most $1.73 \text{cost}(x) + \varepsilon |E_{\text{adm}}| \leq 1.73 \text{opt} + \varepsilon |E_{\text{adm}}|$.

To see the ratio for +edges, notice that we only need to consider two x values: $x = 0$ and $x = 2 - 1.515 = 0.485$. The function inside $\max(\cdot)$ in (1) is decreasing for $x \in [0.485, 1]$ and convex for $x \in [0, 0.485]$. \square

Theorem 7 and 8 will be proved in Section 4 and 5 respectively.

3 Preclustering

In this section, we show how to find a preclustered instance and prove Theorem 4. Given a Correlation Clustering instance $I = (V, E^+ \cup E^-)$ on a set of vertices V , we let $G = (V, E^+)$ be the graph of +edges, and d_v and N_v be respectively the degree and neighbor set of v in G . For reasons that we will note later on, we assume that each vertex has a self-loop +edge. (Note that this does not affect the cost of clustering, as a self-loop is never cut by a clustering.) This assumption implies that $v \in N_v$, and we have $d_v = |N_v|$. Moreover, the number of +neighbors of v equals d_v . Notice that without loss of generality, we can assume that each vertex v in the input instance has at least one proper +neighbor (i.e., $d_v \geq 2$); otherwise, we would clearly put that vertex in its own (singleton) cluster and solve the remaining instance. Let $\varepsilon_q < 10^{-8}$ be a fixed constant, and let $\varepsilon = \sqrt{\varepsilon_q}$.

The goal of this section is to prove the following theorem.

Theorem 9. *Given a Correlation Clustering instance G , we can in polynomial time construct a preclustered instance $(\mathcal{K}, E_{\text{adm}})$ with the following two properties:*

- $\text{cost}(\mathcal{C}_{(\mathcal{K}, E_{\text{adm}})}^*)$ is at most $(1 + \varepsilon)$ times the cost of the optimum clustering for G ,
- $\text{cost}(\mathcal{C}_{(\mathcal{K}, E_{\text{adm}})}^*)$ is at least $(\varepsilon_q^6/2) \cdot |E_{\text{adm}}| = \varepsilon_a |E_{\text{adm}}|$,

where $\mathcal{C}_{(\mathcal{K}, E_{\text{adm}})}^*$ denotes a good (but unknown) clustering for $(\mathcal{K}, E_{\text{adm}})$ of minimum cost.

Note that Theorem 4 immediately follows from Theorem 9 by letting $\varepsilon = \sqrt{\varepsilon_q} \leftarrow \delta$. For the rest of the section, we prove Theorem 9 by describing a procedure that takes a Correlation Clustering instance and outputs a preclustered instance.

Atomic Subpartitioning Our algorithm is similar to the algorithm of [CLM⁺21] and relies on the notion of weak agreement. Informally, we say that u and v are in agreement when their neighborhoods are almost identical, up to a tiny fraction of the neighbors. We expect u and v to be treated similarly in a target optimal solution: either u and v are in the same cluster, or both form singleton clusters.

Our algorithms are parameterized by two constants β, λ that will be determined later.

Definition 10 (Weak Agreement [CLM⁺21]). *Two vertices u and v are in i -weak agreement if $|N_u \Delta N_v| < i\beta \cdot \max\{|N_u|, |N_v|\}$, where $N_u \Delta N_v$ denotes the symmetric difference between N_u and N_v . If u and v are in 1-weak agreement, we also say that u and v are in agreement.*

Remark 1. *As noted earlier, we assume that each vertex has a self-loop +edge, because we follow the assumptions of [CLM⁺21].*

[CLM⁺21] then provide Algorithm 1, which we apply with $\beta := \varepsilon_q$ and $\lambda := \varepsilon_q$.

Algorithm 1 Atomic-Preclustering(G) – Algorithm 1 in [CLM⁺21]

- 1: Discard all +edges whose endpoints are not in agreement.
 - 2: Call a vertex *light* if it has lost more than a λ -fraction of its +neighbors in the previous step. Otherwise call it *heavy*.
 - 3: Discard all +edges between two light vertices.
 - 4: Let \tilde{G} be the *sparsified graph* on the remaining +edges. Let C_1, \dots, C_k be the connected components of \tilde{G} ; each C_i with $|C_i| \geq 2$ is defined as an *atom*. Any pair u, v such that u and v are in the same atom is an atomic pair (or edge).
-

We will make use of the following structural lemmas.

Lemma 11 (Lemma 3.4 in full version (arXiv) of [CLM⁺21]). *Let $\beta = \lambda = \varepsilon_q < 1/100$ in Algorithm 1. Let C be a connected component of \tilde{G} of size at least 2, then for each vertex u in C , we have that the number of +neighbors of u in C is at least $(1 - 9\varepsilon_q)|C|$.*

Lemma 12 (Lemma 3.3 in full version (arXiv) of [CLM⁺21]). *Let $\beta = \lambda = \varepsilon_q < 1/100$ in Algorithm 1. Then, for any u, v in the same atom, we have that if u or v is heavy, then u and v are in 4-weak agreement. Moreover, for any atom K , the hop-distance in K between any two vertices of K is at most 4.*

Lemma 13 (Fact 3.2 of [CLM⁺21], second bullet). *Let $k \in \{2, 3, 4, 5\}$ and $v_1, \dots, v_k \in V$ be a sequence of vertices such that v_i is in agreement with v_{i+1} for $i \in \{1, \dots, k-1\}$. Then v_1 and v_k are in k -weak agreement.*

One of our key lemmas is based on Lemma 3.5 of the full arXiv version of [CLM⁺21].

Lemma 14 (Atom Structure in Preclustering). *There exists an optimum solution \mathcal{C}_1 for I such that for any atom K , there exists a cluster $C \in \mathcal{C}_1$ containing K .*

Proof. Consider an atom K . By the description of Algorithm 1, there is a heavy vertex v in K , because K contains at least one edge and edges with two light endpoints have been removed. Since v is heavy, a $(1 - \varepsilon_q)$ -fraction of its +neighbors are also in K . Moreover by Lemma 12, every other vertex $u \in K$ is in 4-agreement with v . The vertex v has at most $\varepsilon_q|K|$ +neighbors outgoing from K . By 4-agreement, for every other vertex $u \in K$, u has at most $36\varepsilon_q|K|$ +neighbors outgoing from K (by the choice of ε_q). Moreover, by Lemma 11, u has at most $9\varepsilon_q|K|$ –neighbors in K .

Now, fix an optimum solution \mathcal{C} and assume toward contradiction that there exists $C_1, \dots, C_\ell \in \mathcal{C}$, where $\ell > 1$, such that $C_i \cap K \neq \emptyset$ for all i . Recall that each vertex $u \in K$ has at most $36\varepsilon_q|K|$ + neighbors outside

of K and so for each C_i , $|C_i \setminus K| \leq 73\varepsilon_q|K|$ since otherwise the number of $-$ edges between $C_i \setminus K$ and $C_i \cap K$ outnumbers the number of $+$ edges and the clustering where C_i is replaced with $C_i \setminus K$ and $C_i \cap K$ has cheaper cost.

Similar to the proof of Lemma 3.5 in [CLM⁺21], we consider two different cases, either there exists a cluster C^* of size larger than $(1 - 200\varepsilon_q)|K|$ or all the clusters have size at most $(1 - 200\varepsilon_q)|K|$.

In the first case, observe that by the above discussion, C^* contains at least $(1 - 273\varepsilon_q)|K|$ vertices of K and so for each $C_i \neq C^*$, we have that $|C_i \cap K| \leq 273\varepsilon_q|K|$. Since each vertex $v \in C_i \cap K$ has at least $(1 - 9\varepsilon_q)|K| +$ neighbors in K , we have that it has at least $(1 - 282\varepsilon_q)|K| +$ neighbors in C^* . Moreover, since $|C^*| \leq (1 + 73\varepsilon_q)|K|$ by the above discussion, the cost of moving v from C_i to C^* is at most $619\varepsilon_q|K|$ while the saving is at least $(1 - 282\varepsilon_q)|K|$ and since $\varepsilon_q < 1/1000$, the change induces a positive saving and so a contradiction to the fact that the solution is optimal.

We thus turn to the second case where all the clusters C_1, \dots, C_ℓ have size at most $(1 - 200\varepsilon_q)|K|$. By Lemma 11, each vertex $v \in C_i \cap K$ has at least $(1 - 9\varepsilon_q)|K| - |C_i| \geq (1 - 9\varepsilon_q)|K| - (1 - 200\varepsilon_q)|K| = 191\varepsilon_q|K| +$ neighbors in $K \setminus C_i$. Hence \mathcal{C} separates at least $191|K|^2\varepsilon_q/2$ $+$ edges. On the other hand, consider modifying \mathcal{C} by removing all the vertices in K from their current clusters and creating a cluster consisting only of K . The new clustering will be saving at least $191|K|^2\varepsilon_q/2$ (by the above computation) while paying an additional $10|K|^2\varepsilon_q$ cost for the internal $-$ edges and an additional $36\varepsilon_q|K|^2$ cost for the outgoing $+$ edges. Since $46\varepsilon_q|K|^2 < 191|K|^2\varepsilon_q/2$, we have that the resulting clustering is of cheaper cost, a contradiction that concludes the proof. \square

Admissibility A pair (u, v) is *degree-similar* if $\varepsilon_q d_v \leq d_u \leq d_v/\varepsilon_q$. Let \mathcal{K} be the set of atoms output by Algorithm 1. A pair (u, v) is *admissible* if (1) either u or v belongs to $V \setminus V_{\mathcal{K}}$, and (2) it is degree-similar, and (3) the number of common neighbors that are degree-similar to both u and v is at least $\varepsilon_q \cdot \min\{d_u, d_v\}$. We let E_{adm} be the set of all admissible pairs in $\binom{V}{2}$. This finishes the description of our preclustered instance $(\mathcal{K}, E_{\text{adm}})$. To this end, we let $G' = (V, E')$ be the graph containing the set of atomic and admissible edges, and for every $v \in V$, let N'_v and d'_v be the neighbor set and degree of v in G' .

Lemma 15. *The following two properties hold for $(\mathcal{K}, E_{\text{adm}})$:*

- (i) *for every $v \in V$, we have $d'_v \leq 2\varepsilon_q^{-3} \cdot d_v$, and*
- (ii) *for every $uv \in E'$, we have $d_u \leq 2\varepsilon_q^{-1} \cdot d_v$.*

Proof. We consider an arbitrary vertex u . Let us first analyse the number of edges u, v that are admissible edges. By the definition of an admissible pair, we have that $d_u \leq \varepsilon_q^{-1}d_v$ for an admissible edge $uv \in E'$. Next we show that for any u , $d'_u \leq d_u/\varepsilon_q^3$. Let D_u be the set of neighbors of u in G that are degree similar to u . Each vertex v such that (u, v) is an admissible pair is connected to at least $\varepsilon_q \cdot \min\{d_u, d_v\} \geq \varepsilon_q^2 d_u$ vertices in D_u . Moreover, each vertex $w \in D_u$ has degree $d_w \leq d_u/\varepsilon_q$ by definition of degree-similar. Hence, by using a counting argument, we conclude that the total number of vertices v such that uv is in E_{adm} is at most $|D_u|(d_u/\varepsilon_q)/(\varepsilon_q^2 d_u) \leq d_u/\varepsilon_q^3$. This yields the desired bounds for any vertex u that does not belong to an atom.

To finish the analysis, we need to analyze the number of atomic edges attached to any vertex u in an atom and show that a pair of vertices u, v in the same atom satisfies $d_u \leq 2\varepsilon_q^{-1} \cdot d_v$. By Lemma 12 and Lemma 13, we have that u is in 8-weak agreement with any element of K so we have that $d_u \leq 2d_v \cdot \varepsilon_q^{-1}$ by our choice of ε_q , as desired. Moreover, consider a heavy vertex v' in K (there must exist one by definition). The fact v remained heavy implies that K contains at least a $(1 - \varepsilon_q)$ fraction of the $+$ neighbors of v' , and Lemma 11 implies that the degree of v' is at least $(1 - O(\varepsilon_q))|K|$. Since any other vertex is in 4-weak agreement with v' , we have that the degree of any vertex $u \in K$ is $(1 \pm O(\varepsilon_q))|K|$ and so $d'_u \leq 2\varepsilon_q^{-3} \cdot d_u$. \square

We now want to prove Theorem 9 for an preclustered instance $(\mathcal{K}, E_{\text{adm}})$. We start with the following lemma.

Lemma 16. *Let \mathcal{K} be the set of atoms output by Algorithm 1. There exists a good clustering \mathcal{C}_2 of I with cost at most $(1 + \varepsilon)$ times the optimum solution of I such that*

1. For any vertex v that belongs to a non-singleton cluster $C \in \mathcal{C}_2$, we have that v has at least $(1 + \varepsilon)|C|/2$ +neighbors in C .
2. For any vertex v that belongs to a non-singleton cluster $C \in \mathcal{C}_2$, we have that $|C| \geq \varepsilon d_v$.

Proof. Start from \mathcal{C}_1 as per Lemma 14 and modify it as follows. For any cluster $C \in \mathcal{C}_1$, let $\mathcal{K}(C)$ be the set of atoms K such that $K \cap C \neq \emptyset$ (note that by Lemma 14, if $K \cap C \neq \emptyset$, then $K \subseteq C$). For any cluster $C \in \mathcal{C}_1$, if the number of $-$ edges within C is at least $(1 - 10\varepsilon)\binom{|C|}{2}/2$, then make each $K \in \mathcal{K}(C)$ a cluster by itself and each non-atom vertex $v \in C$ a singleton cluster. This increases the cost by a factor of at most $(1 + 10\varepsilon)/(1 - 10\varepsilon)$. Secondly, if the number of $+$ edges outgoing from the cluster is at least $|C|^2/\varepsilon$, then make all the non-atom vertices of C singleton clusters and each $K \in \mathcal{K}(C)$ a cluster by itself, and charge the cost of $|C|^2$ for the operation to the outgoing $+$ edges; each edge across clusters of \mathcal{C}_1 each gets a charge of $O(\varepsilon)$ (ε for each endpoint). Let \mathcal{C}'_1 be the resulting solution which has cost at most $(1 + \varepsilon)$ times the cost of \mathcal{C}_1 by the above argument.

We next apply the following iterative procedure to \mathcal{C}'_1 : As long as there exists a vertex v in a non-singleton cluster C such that either Event (1): v has less than $(1 + \varepsilon)|C|/2$ neighbors in C , or Event (2): $|C| < \varepsilon d_v$, we proceed as follows:

- **Individual case** If v is not in an atom, we make v a singleton.
- **Atom case** Otherwise v belongs to an atom, we remove the whole atom K containing v from C and make it a cluster on its own.

We first argue that the procedure outputs a clustering that satisfies the conditions of the lemma. The only way the two items can be violated is by an atom (since otherwise, the problematic vertex would be made a singleton). We thus argue that atoms satisfy the two properties. By Lemma 11 and $\varepsilon < 1 - 18\varepsilon_q$, each atom satisfies the first bullet. We then argue that each atom K is such that for any vertex $v \in K$, $\varepsilon d_v < |K|$. This follows from the fact that each atom K contains a heavy vertex u that is thus in agreement with a $(1 - \varepsilon_q)$ fraction of its neighbors that hence join the atom; it follows that $|K| > (1 - \varepsilon_q)d_u > \varepsilon d_u$. For the other vertices, v is in 4-weak agreement with u , and so the property holds. It follows that when the procedure stops, the solution obtained satisfies the constraints of the theorem.

We then need to account for the cost increase. We first make the following observation: Any non-singleton cluster $C \in \mathcal{C}'_1$ (and so of \mathcal{C}_1) cannot lose more than $(1 - \varepsilon)|C|$ vertices because of Event (1) and $|C|/2$ vertices because of Event (2). Indeed, if it loses more than $(1 - \varepsilon)|C|$ vertices because of Event (1), then that implies that the total number of $-$ edges internal to C was already larger than $(1 - 10\varepsilon)|C|(|C| - 1)/4$ and should not have been in \mathcal{C}'_1 . Moreover, if it loses more than $|C|/2$ vertices because of Event (2), that means that the number of $+$ edges outgoing C was at least $|C|^2/(2\varepsilon)$ and should not have been in \mathcal{C}'_1 .

We can now analyse the cost increase due to Event (2). In the individual case, namely when a single vertex v is placed in a singleton cluster, we have that v has at least $|C|(1 - \varepsilon)/(2\varepsilon)$ +neighbors outside its cluster in \mathcal{C}'_1 and so the cost can be charged (with an individual edge charge of 2ε) to these +edges. In the atom case, namely when an entire atom K is placed in a single cluster, we have that there exists a vertex v in K with at least $(|C| - 1)/(2\varepsilon)$ +neighbors outside its cluster in \mathcal{C}'_1 . Moreover, each vertex $u \in K$ is in at most 5-weak-agreement (by Lemmas 12 and 13 and so has at least $(1 - 2\varepsilon)(|C| - 1)/(2\varepsilon)$ +neighbors outside its cluster in \mathcal{C}'_1 and so the cost of moving the atom outside of C can be charged to the +neighbors of the atom that are not in the cluster of the atom in \mathcal{C}'_1 , with a charge of $2\varepsilon/(1 - 2\varepsilon)$ per individual edge. Therefore, the cost of Event (2) can be charged to the +edges paid in \mathcal{C}'_1 , each edge receiving a charge of $O(\varepsilon)$.

It thus remains to bound the cost incurred by Event (1). Consider the atom case, namely the case where an entire atom K is moved out of a cluster and creates a cluster on its own. In this case, we have that there is a vertex $u \in K$ which is adjacent to less than a $(1 + \varepsilon)|C|/2$ fraction of its current cluster C and assume that $\varepsilon d_u \leq |C|$ since otherwise this is an Event (2) case. Note that in this case, since $\varepsilon d_u \leq |C|$ and by Lemma 12 that for any vertex $u \in K$, $|N_u \Delta N_v| < 4\varepsilon_q \cdot \max\{|N_u|, |N_v|\}$ and so each vertex $u \in K$ is adjacent to less than $(1 + 5\varepsilon)|C|/2$ vertices in C . This implies that whenever a vertex v is moved out of its cluster in the procedure, it is adjacent to at most $(1 + 5\varepsilon)|C|/2$ vertices in C and so has at least $(1 - 5\varepsilon)|C|/2$ $-$ neighbors in C . We thus charge the cost of moving v outside of C , which is at most $(1 + 5\varepsilon)|C|/2$, to the

cost v was paying in \mathcal{C}'_1 by placing a charge of $O(\varepsilon)$ on each $-$ neighbor of v in C . This account for the change in cost since each $-$ neighbor of v in C was contributing one to the objective and is now contributing 0. Note that in this way, each $-$ edge that contributes to the objective in \mathcal{C}'_1 is charged at most once: when one of its endpoints is moved out of the cluster and cannot be charged later in the procedure since the endpoints remain in different clusters. \square

We can now turn to the proof of Theorem 9.

Proof of Theorem 9. We consider the solution \mathcal{C}_2 from Lemma 16 and the pair $(\mathcal{K}, E_{\text{adm}})$. It is easy to see that $(\mathcal{K}, E_{\text{adm}})$ can be computed in polynomial time.

Lemma 16 already shows that the cost of \mathcal{C}_2 is within a $(1 + \varepsilon)$ factor of the optimum cost and so it only remains to show that \mathcal{C}_2 is a good clustering for $(\mathcal{K}, E_{\text{adm}})$ and that its cost is at least $\varepsilon_a \cdot |E_{\text{adm}}|$.

Let us first argue that \mathcal{C}_2 is a good clustering for $(\mathcal{K}, E_{\text{adm}})$, namely that

- For every atom $K \in \mathcal{K}$, we have $K \subseteq C$ for some $C \in \mathcal{C}_2$.
- For every non-admissible edge uv , u and v are not in the same cluster in \mathcal{C}_2 .

Note that the first bullet is satisfied by \mathcal{C}_2 by Lemma 16. It thus remains to show that the second bullet is satisfied. By Lemma 16, pairs of vertices in the same cluster are degree-similar and moreover, for any vertex v that belongs to a non-singleton cluster $C \in \mathcal{C}_2$, we have that v has at least $(1 + \varepsilon)|C|/2$ $+$ neighbors in C that are degree-similar. This implies that any pair of vertices u, v that are in the same cluster $C \in \mathcal{C}_2$ has at least $\varepsilon|C|$ common neighbors that are degree similar, and by Lemma 16, $\varepsilon|C| \geq \varepsilon_q \cdot \min\{d_u, d_v\}$. Thus, either u, v is a pair of vertices in the same atom (hence $(u, v) \in E'$) or it is admissible (and so $(u, v) \in E'$ too) as desired.

We now turn to providing a lower bound on the cost. We say that a vertex v is a *high-contributor* if its contribution to the cost is at least $\varepsilon_q^2 d_v$. Thus, any vertex that is not a high-contributor must be adjacent to a $(1 - \varepsilon_q^2)$ -fraction of its cluster and have at most $\varepsilon_q^2 d_v$ $+$ neighbors outside of its cluster. It follows that $+$ neighbors that are not high-contributors and that are in the same cluster are in agreement. Next, define a cluster C to be a high-contributor cluster if the total number of outgoing $+$ edges (i.e., $+$ edges with exactly one endpoint in C) is at least $\varepsilon_q^4 |C|^2$ or the total number of internal $-$ edges ($-$ edges with both endpoints in C) is at least $\varepsilon_q^4 |C|^2$.

Next, consider a non-high-contributor cluster C ; it must be that the number of non-high-contributor vertices in C is at least $(1 - \varepsilon_q^2)|C|$. Thus, since each non-high-contributor vertex in C is adjacent to at least $(1 - \varepsilon_q^2)|C|$ vertices in C and in agreement with the other non-high-contributor vertices, we have that each non-high-contributor vertex is heavy, and so all the non-high-contributor vertices of C are in the same atom.

We can thus rewrite a lower bound on the cost of the solution \mathcal{C}_3 as

$$\begin{aligned} & \sum_{\text{high-contributor cluster } C} \varepsilon_q^2 |C|^2 + \sum_{\text{non high-contributor cluster } C} \sum_{\text{high-contributor vertex } v \in C} \varepsilon_q^2 d_v \\ & \sum_{\text{high-contributor cluster } C} \varepsilon_q^2 |C|^2 \geq \sum_{\text{high-contributor cluster } C} \varepsilon_q^2 \sum_{v \in C} |C| \\ & \geq \sum_{\text{high-contributor cluster } C} \varepsilon_q^2 \sum_{v \in C} \varepsilon d_v \\ & \geq \sum_{\text{high-contributor cluster } C} \sum_{v \in C} \varepsilon_q^3 d_v, \end{aligned}$$

where the second inequality follows from the second bullet of Lemma 16.

So we have that the total contributions of vertices is at least:

$$\sum_{\text{non high-contributor cluster } C} \sum_{v \in C \setminus K} \varepsilon_q^3 d_v + \sum_{\text{high-contributor cluster } C} \sum_{v \in C} \varepsilon_q^3 d_v + \sum_{\text{Singletons}} d_v / 2 \geq \sum_{v \in V \setminus V_{\mathcal{K}}} \varepsilon_q^3 d_v$$

It follows that the cost of solution \mathcal{C}_3 is at least $\varepsilon_q^3 \sum_{v \in V \setminus V_{\mathcal{K}}} d_v$, as claimed (where $\varepsilon_q = \varepsilon^2$). By Lemma 15, we have

$$\sum_{v \in V \setminus V_{\mathcal{K}}} \varepsilon_q^3 d_v \geq \sum_{v \in V \setminus V_{\mathcal{K}}} \varepsilon_q^3 (d'_v \varepsilon_q^3) / 2 \geq \sum_{v \in V \setminus V_{\mathcal{K}}} \varepsilon_q^6 d'_v / 2 \geq \varepsilon_q^6 / 2 \cdot |E_{\text{adm}}|. \quad \square$$

4 Set-Based Rounding Procedure

In this section, we prove Theorem 7, by giving the set-based rounding algorithm \mathcal{A}_{set} . We repeat the theorem below:

Theorem 7. (Set-based Rounding) *There is an $n^{O(1/\varepsilon^4)}$ -time procedure \mathcal{A}_{set} that either outputs a separation plane for x , or a clustering \mathcal{C} such that*

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \frac{2x_{vw}}{1+x_{vw}} + \sum_{vw \in E^-} \frac{1-x_{vw}}{1+x_{vw}} + \varepsilon \cdot |E_{\text{adm}}|.$$

Recall that we are given a Correlation Clustering instance $(V, E^+ \cup E^-)$, a preclustered instance $(\mathcal{K}, E_{\text{adm}})$, and a metric $x \in [0, 1]^{\binom{V}{2}}$ satisfying the properties in Theorem 5. The output is either a separation plane between x and \mathcal{P} , or a clustering \mathcal{C} . Recall that \mathcal{P} is the convex hull of the metrics of all good-clusterings for $(\mathcal{K}, E_{\text{adm}})$.

Throughout the section, we use $N_{\text{adm}}(u)$ to denote the set of admissible edges incident to u , $d_{\text{adm}}(v)$ to denote its size, and $E_{\text{adm}}(A, B)$ for two disjoint subsets A and B to denote the set of admissible edges between A and B . It is good to keep in mind the algorithm for the $O(1)$ -sized cluster case from the introduction, as it will serve as a baseline for our more general algorithm.

4.1 Linear Program Relaxation

During the algorithm, we maintain the set V' of vertices that are not clustered yet, and let $n' = |V'|$; initially we have $V' = V$. For $(\mathcal{K}, E_{\text{adm}})$ and for each $u \in V$, we shall let K_u denote the atom containing u , if there is one; otherwise let $K_u = \{u\}$. Let \mathcal{C}^* be any good clustering for $(\mathcal{K}, E_{\text{adm}})$; in other words, the metric for \mathcal{C}^* is a vertex-point of \mathcal{P} . Let \mathcal{C}'^* be the clustering \mathcal{C}^* , restricted to V' and with empty clusters removed. We shall define a clustering $\tilde{\mathcal{C}}$ which will serve as the target clustering for our LP. The cluster $\tilde{\mathcal{C}}$ is constructed as follows:

- 1: let $\tilde{\mathcal{C}} \leftarrow \mathcal{C}'^*$
- 2: **while** there exists some K_u in a cluster $C \in \tilde{\mathcal{C}}$ with $|K_u| < |C| < \varepsilon d_{\text{adm}}(u) + |K_u|$ **do**
- 3: $\tilde{\mathcal{C}} \leftarrow \tilde{\mathcal{C}} \setminus \{C\} \cup \{K_u, C \setminus K_u\}$

Note this procedure is only for analysis purpose and is not a part of our algorithm, as we do not know \mathcal{C}^* and \mathcal{C}'^* .

Claim 17. *The following statements are true for the clustering $\tilde{\mathcal{C}}$:*

- (17a) *For every $u \in V'$, K_u is either a cluster, or in a cluster of size more than $|K_u| + \varepsilon \cdot d_{\text{adm}}(u)$.*
- (17b) *The number of pairs in $\binom{V'}{2}$ cut in $\tilde{\mathcal{C}}$ is at most that in \mathcal{C}'^* plus $\varepsilon \cdot \sum_{u \in V'} d_{\text{adm}}(u)$.*

Proof. The first statement is straightforward. Whenever we break C into K_u and $C \setminus K_u$ in the procedure, the cost increase is at most $|K_u| \cdot |C \setminus K_u| \leq |K_u| \cdot \varepsilon d_{\text{adm}}(u) \leq \varepsilon \sum_{v \in K_u} d_{\text{adm}}(v)$. We separate each K_u at most once. Therefore, the second statement holds. \square

We then describe the LP relaxation, which depends on V' . Suppose we have a good clustering \mathcal{C}^* , where \mathcal{C}'^* is the clustering \mathcal{C}^* restricted to V' , and let $\tilde{\mathcal{C}}$ be obtained as above. Let $r = \Theta(\frac{1}{\varepsilon^4})$, with a large enough hidden constant inside $\Theta(\cdot)$. In the LP, we have a variable y_s^* , for every $s \in [n']$, and $S \subseteq V'$ of size at most r , that denotes the number of clusters of size s in $\tilde{\mathcal{C}}$ containing S as a subset. When $S \neq \emptyset$, there is at

most one such cluster and thus $y_S^s \in \{0, 1\}$ in an integral solution. For every S , let $y_S := \sum_s y_S^s$ denote the number of clusters (of any size) in $\tilde{\mathcal{C}}$ containing S as a subset. If $S \neq \emptyset$, then y_S indicate if S is a subset of a cluster in $\tilde{\mathcal{C}}$ or not. For every $uv \in \binom{V'}{2}$, we have a variable \tilde{x}_{uv} indicating if u and v are separated or not in $\tilde{\mathcal{C}}$.

For convenience, we shall use the following type of shorthand: y_u^s for $y_{\{u\}}^s$, y_{uv}^s for $y_{\{u,v\}}^s$, and y_{Su}^s for $y_{S \cup \{u\}}^s$. The LP is defined as in LP(2-11). In the description of the LP, we always have $s \in [n']$, $u \in V'$ and $uv \in \binom{V'}{2}$. For convenience, we omit the restrictions. By default, any variable of the form y_S or y_S^s has $|S| \leq r$; if not, we do not have the variable and the constraint involving it.

$$\begin{aligned} \sum_{s=1}^{n'} y_S^s &= y_S & \forall S & \quad (2) & \tilde{x}_{uv} &= 0 & \forall u, v \text{ in a same } K \in \mathcal{K} & \quad (7) \\ y_u &= 1 & \forall u & \quad (3) & \sum_{uv} (\tilde{x}_{uv} - x_{uv}) &\leq \varepsilon \sum_u d_{\text{adm}}(u) & & \quad (8) \\ y_{uv} &= 1 - \tilde{x}_{uv} & \forall uv & \quad (4) & y_S^s &= 0 & \text{if implied by (17a)} & \quad (9) \\ \tilde{x}_{uv} &\geq x_{uv} & \forall uv & \quad (5) & \sum_{T' \subseteq T} (-1)^{|T'|} y_{S \cup T'}^s &\in [0, y_S^s] & \forall s, S \cap T = \emptyset & \quad (10) \\ \frac{1}{s} \sum_u y_{Su}^s &= y_S^s & \forall s, S & \quad (6) & & & \text{all variables are non-negative} & \quad (11) \end{aligned}$$

\tilde{x} and y variables are the LP variables, and x variables are given as the input by Theorem 7. (2) gives the definition of y_S , (3) requires u to be contained in some cluster, and (4) gives the definition of \tilde{x}_{uv} . (5) holds as $\tilde{\mathcal{C}}$ is a refinement of \mathcal{C}'^* . (6) says if $y_S^s = 1$, then there are exactly s elements $u \in V$ with $y_{Su}^s = 1$. (An exception is when $S = \emptyset$; but the equality also holds). (8) follows from (17b) and (9) is by (17a). The left side of (10) is the number of clusters of size s in $\tilde{\mathcal{C}}$ containing S but does not contain any vertex in T . So the inequality holds. This corresponds to a Sherali-Adams relaxation needed for the correlated rounding [RT12], see Lemma 18. (11) is the non-negativity constraint.

If $x \in \mathcal{P}$, then the above LP is feasible, as this holds for every vertex point x of \mathcal{P} . On the other hand, if the LP is infeasible, then we can use Lemma 6 to return a separation plane between x and \mathcal{P} .

4.2 Rounding Algorithm

With the LP defined, we can then describe the rounding algorithm for Theorem 7. The pseudo-code is given in Algorithm 2, and it calls Algorithm 3.

Algorithm 2 Set-Based Rounding Procedure

Input: A Correlation Clustering instance G , a preclustered instance $(\mathcal{K}, E_{\text{adm}})$ for G , and a metric $x \in [0, 1]^{\binom{V'}{2}}$ satisfying properties in Theorem 5, and $\varepsilon > 0$

Output: Either a separation plane for x , or an integral clustering \mathcal{C}

- 1: $V' \leftarrow V, \mathcal{C} \leftarrow \emptyset$
 - 2: **while** $V' \neq \emptyset$ **do**
 - 3: try to solve LP(2-11) for the V' to obtain a vector (\tilde{x}, y)
 - 4: **if** the LP is infeasible **then**
 - 5: **return** a separation plane for x using Lemma 6
 - 6: **else**
 - 7: $C \leftarrow \text{set-based-cstr-clst}(V', y), \mathcal{C} \leftarrow \mathcal{C} \cup \{C\}, V' \leftarrow V' \setminus C$
 - 8: **return** \mathcal{C}
-

Step 4 of Algorithm 3 uses the following *correlated rounding* from Raghavendra and Tan [RT12]. Note that vertex v with $y'_v \in \{0, 1\}$ is trivially decided and we incur errors only for v with $y'_v \in (0, 1)$.

Algorithm 3 set-based-cstr-clst(V', y)

- 1: randomly choose a cardinality s , so that s is chosen with probability $\frac{y_0^s}{y_0}$
 - 2: randomly choose a vertex $u \in V'$, such that u is chosen with probability $\frac{y_u^s}{s y_0^s}$
 - 3: define a vector y' such that $y'_S = \frac{y_{S_u}^s}{y_u^s}$ for every $S \subseteq V$ of size at most $r - 1$
 - 4: apply the Raghavendra-Tan correlated rounding technique over the fractional set y' to construct the cluster $C \subseteq V'$ that does not break any atom, and **return** C
-

Lemma 18 ([RT12]). *In Step 4 of Algorithm 3, let $V'_u := \{v : y'_v \in (0, 1)\}$. One can sample $C \subseteq V'$ in time $n^{O(r)}$ such that*

- For each $v \in V'$, $\Pr[v \in C] = y'_{uv}$.
- $\mathbb{E}_{u, v \in V'_u} [|\Pr[v, w \in C] - y'_{uvw}|] \leq \varepsilon_r$, where $\varepsilon_r = O(1/\sqrt{r})$.

Recall that $r = \Theta(\frac{1}{\varepsilon^4})$. So by setting the constant appropriately, we can have $\varepsilon_r = \varepsilon^2$.

4.3 Analysis of Error-Free Version of Algorithm 3

In this section, we analyze Algorithm 3 by ignoring the errors incurred by the RT procedure. We focus on one execution of the algorithm with input V' and y . To do this formally, we define $\text{err}_{vw|u}^s$ to be the error generated by the procedure when we choose s as the size and u as the pivot:

$$\text{err}_{vw|u}^s := \left| \Pr[v, w \in C | s, u] - \frac{y_{uvw}^s}{y_u^s} \right|, \forall vw \in \binom{V'}{2},$$

and

$$\text{err}_{vw}^s := \frac{1}{s y_0^s} \sum_{u \in V'} y_u^s \cdot \text{err}_{vw|u}^s \quad \text{and} \quad \text{err}_{vw} := \sum_s \frac{y_0^s}{y_0} \cdot \text{err}_{vw}^s.$$

as the error for vw conditioned on s , and the unconditioned error. Notice that all these quantities are expectations, and thus deterministic. In our analysis, we shall leave the error terms in all inequalities, and bound them in the next section.

Lemma 19. *Given a vertex $v \in V'$, the probability that v is clustered in the execution of set-based-cstr-clst is exactly $\frac{1}{y_0}$.*

Proof. The probability is

$$\sum_s \frac{y_0^s}{y_0} \sum_{u \in V'} \frac{y_u^s}{s y_0^s} \cdot \frac{y_{uv}^s}{y_u^s} = \frac{1}{y_0} \sum_s \frac{1}{s} \sum_{u \in V'} y_{uv}^s = \frac{1}{y_0} \sum_s y_v^s = \frac{1}{y_0} y_v = \frac{1}{y_0}.$$

The second equality is by (6). The third and the last inequalities are by (2) and (3) respectively. □

Lemma 20. *Focus on an edge $vw \in \binom{V'}{2}$.*

1. *The probability that vw is decided (i.e., one of v and w is in C) by the execution of set-based-cstr-clst is at least $\frac{1}{y_0}(1 + \tilde{x}_{vw}) - \text{err}_{vw}$.*
2. *If $vw \in E^+$, then the probability that vw is decided wrongly is at most $\frac{2}{y_0} \cdot \tilde{x}_{vw} + \text{err}_{vw}$.*
3. *If $vw \in E^-$, then the probability that vw is decided wrongly is at most $\frac{1}{y_0} \cdot (1 - \tilde{x}_{vw}) + \text{err}_{vw}$.*

Proof. We focus on the first statement. The probability that vw is decided conditioned on s is at least

$$\begin{aligned} & \sum_{u \in V'} \frac{y_u^s}{s y_\emptyset^s} \cdot \left(\frac{1}{y_u^s} \cdot (y_{uv}^s + y_{uw}^s - y_{uvw}^s) - \text{err}_{vw|u}^s \right) = \sum_{u \in V'} \left(\frac{1}{s y_\emptyset^s} \cdot (y_{uv}^s + y_{uw}^s - y_{uvw}^s) - \frac{y_u^s}{s y_\emptyset^s} \cdot \text{err}_{vw|u}^s \right) \\ & = \frac{1}{y_\emptyset^s} (y_v^s + y_w^s - y_{vw}^s) - \text{err}_{vw}^s. \end{aligned}$$

To see the second equality, we apply (6) with $S = \{v\}, \{w\}$ and $\{v, w\}$ respectively, and use the definition of err_{vw}^s .

Deconditioning on s , we have that the probability vw is decided is at least

$$\begin{aligned} & \sum_s \frac{y_\emptyset^s}{y_\emptyset} \cdot \left(\frac{1}{y_\emptyset^s} (y_v^s + y_w^s - y_{vw}^s) - \text{err}_{vw}^s \right) = \frac{1}{y_\emptyset} \sum_s (y_v^s + y_w^s - y_{vw}^s) - \text{err}_{vw} \\ & = \frac{1}{y_\emptyset} (1 + 1 - y_{vw}) - \text{err}_{vw} = \frac{1}{y_\emptyset} (1 + \tilde{x}_{vw}) - \text{err}_{vw}. \end{aligned}$$

The first equality used the definition of err_{vw}^s . The third equality used that $\sum_s y_v^s = y_v = 1, \sum_s y_w^s = y_w = 1$ and $\sum_s y_{vw}^s = y_{vw} = 1 - \tilde{x}_{vw}$.

The second and third statements can be proved similarly. When $vw \in E^+$, the probability that vw is wrongly decided conditioned on s and u is at most $\frac{1}{y_u^s} (y_{uv}^s + y_{uw}^s - y_{uvw}^s) + \text{err}_{vw|u}^s$; when $vw \in E^-$, this is at most $\frac{y_{uvw}^s}{y_u^s} + \text{err}_{vw|u}^s$. Following the calculations, the two unconditioned probabilities are at most $\frac{1}{y_\emptyset} (1 + 1 - 2y_{vw}) + \text{err}_{vw} = \frac{1}{y_\emptyset} \cdot 2\tilde{x}_{vw} + \text{err}_{vw}$ and $\frac{1}{y_\emptyset} \cdot y_{vw} + \text{err}_{vw} = \frac{1}{y_\emptyset} \cdot (1 - \tilde{x}_{vw}) + \text{err}_{vw}$. \square

Therefore, if we ignore the error terms, and the difference between \tilde{x}_{vw} 's and x_{vw} 's, the algorithm exactly simulates the KT-rounding algorithm for the bounded-cluster size case. By (8), the difference between \tilde{x} and x is small. In the next section, we shall bound the errors.

4.4 Handling the Errors

In this section, we handle the errors. Again recall that we focus on one execution of the set-based-cstr-clst procedure, with input V' and vector y , which defines the vector \tilde{x} . The key lemma we prove is the following:

Lemma 21.

$$\sum_{vw \in \binom{V'}{2}} \text{err}_{vw} \leq \varepsilon \cdot \mathbb{E} \left[\left| \{uw \in \binom{V'}{2} \cap E_{\text{adm}} : uw \text{ decided}\} \right| \right]. \quad (12)$$

Proof. Through the proof, we assume u, v, w are all in V' , vw and uw are in $\binom{V'}{2}$. We bound the sum of errors conditioned on s :

$$\sum_{vw} \text{err}_{vw}^s = \sum_{vw} \frac{1}{s y_\emptyset^s} \cdot \sum_{u \in V'} y_u^s \cdot \text{err}_{vw|u}^s = \frac{1}{s y_\emptyset^s} \sum_{u \in V'} y_u^s \cdot \sum_{vw} \text{err}_{vw|u}^s.$$

Fix some $s \in [n'], u \in V'$, and we now bound $\sum_{vw} \text{err}_{vw|u}^s$. If $s = |K_u|$, then C will be K_u and no errors will be created. (Notice that in this case the LP constraints will imply that $y_{uv}^s = 0$ for every $v \notin K_u$.) So, we assume $s > |K_u|$. By (9), we have that $s > |K_u| + \varepsilon \cdot d_{\text{adm}}(u)$, since otherwise we shall $y_u^s = 0$. Finally, by

Lemma 18, $\sum_{vw} \text{err}_{vw|u}^s \leq \varepsilon_r |N_{\text{adm}}(u) \cap V'|^2$, and recall that $\varepsilon_r = \varepsilon^2$. Therefore,

$$\begin{aligned} \sum_{vw} \text{err}_{vw|u}^s &\leq \varepsilon_r \cdot |N_{\text{adm}}(u) \cap V'|^2 \leq \frac{\varepsilon_r}{\varepsilon} \cdot |N_{\text{adm}}(u) \cap V'| \cdot (s - |K_u|) \\ &= \varepsilon \cdot |N_{\text{adm}}(u) \cap V'| \cdot \sum_{v \in N_{\text{adm}}(u) \cap V'} \frac{y_{uv}^s}{y_u^s} \\ &= \varepsilon \cdot \sum_{v, w \in N_{\text{adm}}(u) \cap V'} \frac{y_{uv}^s}{y_u^s}. \end{aligned}$$

So we have

$$\sum_{vw} \text{err}_{vw}^s \leq \frac{1}{sy_\emptyset^s} \sum_{u \in V'} y_u^s \cdot \sum_{v, w \in N_{\text{adm}}(u) \cap V'} \varepsilon \cdot \frac{y_{uv}^s}{y_u^s} \leq \varepsilon \cdot \frac{1}{sy_\emptyset^s} \cdot \sum_{u \in V', v, w \in N_{\text{adm}}(u) \cap V'} y_{uv}^s.$$

Now we consider the right side of (12). The expectation of the quantity conditioned on s is at least

$$\varepsilon \cdot \sum_{v \in V'} \frac{y_v^s}{sy_\emptyset^s} \sum_{u \in N_{\text{adm}}(v) \cap V', w \in N_{\text{adm}}(u)} \frac{y_{uv}^s}{y_v^s} = \varepsilon \cdot \frac{1}{sy_\emptyset^s} \cdot \sum_{u \in V', v, w \in N_{\text{adm}}(u) \cap V'} y_{uv}^s.$$

This is at least $\sum_{vw} \text{err}_{vw}^s$. Taking all s into consideration gives us (12). \square

Wrapping Up Now we finish the proof of Theorem 7. Focus on one execution of set-based-cstr-clst. We define the following three types of *budgets*:

- When an edge $vw \in \binom{V'}{2}$ is decided in the procedure, we get an *LP budget* of $\frac{2x_{vw}}{1+x_{vw}}$ from vw if vw is a +edge, and $\frac{1-x_{vw}}{1+x_{vw}}$ if it is a -edge. This is the budget coming from the cost of x .
- If additionally vw is an admissible edge, we get an *error budget* of ε from vw . This will be used to cover the errors incurred by the RT rounding procedure.
- Finally a vertex v is clustered in the procedure, we get a *difference budget* of $2\varepsilon \cdot d_{\text{adm}}(v)$. This will be used to cover the difference between \tilde{x} and x .

If an edge $vw \in \binom{V'}{2}$ is wrongly decided, we pay a cost of 1. For a +edge $vw \in \binom{V'}{2}$,

$$\begin{aligned} \Pr[vw \text{ wrongly decided}] &\leq \frac{1}{y_\emptyset} \cdot 2\tilde{x}_{vw} + \text{err}_{vw} = \frac{1}{y_\emptyset} \cdot (2x_{vw} + 2(\tilde{x}_{vw} - x_{vw})) + \text{err}_{vw} \\ &= \mathbb{E}[\text{LP budget from } vw] + \frac{2(\tilde{x}_{vw} - x_{vw})}{y_\emptyset} + \text{err}_{vw}. \end{aligned}$$

Similarly, for a -edge $vw \in \binom{V'}{2}$, we have

$$\Pr[vw \text{ wrongly decided}] \leq \mathbb{E}[\text{LP budget from } vw] + \text{err}_{vw}.$$

Summing up the inequalities over all edges $vw \in \binom{V'}{2}$, we have that in the execution of set-based-cstr-clst,

$$\begin{aligned} \mathbb{E}[\text{cost incurred}] &\leq \mathbb{E} \left[\text{LP budget from } \binom{V'}{2} \right] + \frac{2}{y_\emptyset} \sum_{vw} (\tilde{x}_{vw} - x_{vw}) + \sum_{vw} \text{err}_{vw} \\ &\leq \mathbb{E} \left[\text{LP budget from } \binom{V'}{2} \right] + \frac{2}{y_\emptyset} \cdot \varepsilon \sum_{u \in V'} d_{\text{adm}}(u) + \mathbb{E} \left[\text{error budget from } \binom{V'}{2} \cap E_{\text{adm}} \right] \\ &= \mathbb{E} [\text{all 3 types of budget we get}]. \end{aligned}$$

The second inequality is by (8), and Lemma 21. To see the last equality, notice that every u is clustered with probability exactly $\frac{1}{y_0}$ by Lemma 19.

The procedure can be derandomized by enumerating s, u and the random seeds used in Raghavendra-Tan rounding procedure.² So, we can guarantee that in every execution of set-based-ctr-clst, the cost incurred is at most the budget we get, including the LP, error and difference budgets. So, considering the whole Algorithm 2, the cost of the clustering is at most the total budget we get. For every edge $vw \in \binom{V}{2}$, we only get the LP and error budget once from vw . For every vertex $v \in V$, we only get the error budget once from v . Therefore, the total budget we get is

$$\sum_{vw \in E^+} \frac{2x_{vw}}{1+x_{vw}} + \sum_{vw \in E^-} \frac{1-x_{vw}}{1+x_{vw}} + \varepsilon|E_{\text{adm}}| + 2\varepsilon \sum_{u \in V} d_{\text{adm}}(u) \leq \sum_{vw \in E^+} \frac{2x_{vw}}{1+x_{vw}} + \sum_{vw \in E^-} \frac{1-x_{vw}}{1+x_{vw}} + O(\varepsilon) \cdot |E_{\text{adm}}|.$$

We rescale ε so that the additive term becomes $\varepsilon|E_{\text{adm}}|$. This finishes the proof of Theorem 7.

5 Pivot-Based Rounding Procedure

In this section, we present our pivot-based rounding algorithm $\mathcal{A}_{\text{pivot}}$ and prove its guarantee restated below.

Theorem 8. (Pivot-based Rounding) *There is an $n^{O(1/\varepsilon^2)}$ -time procedure $\mathcal{A}_{\text{pivot}}$ that either outputs a separation plane for x or a clustering \mathcal{C} such that*

$$\text{cost}(\mathcal{C}) \leq \sum_{vw \in E^+} \min\{1.515 + x_{vw}, 2\} \cdot x_{vw} + 2 \sum_{vw \in E^-} (1 - x_{vw}) + \varepsilon \cdot |E_{\text{adm}}|.$$

In this section, a singleton vertex in $V \setminus V_{\mathcal{K}}$ will be treated almost like an atom, so the term atom will also refer to (the singleton set of) such a vertex. Let $\mathcal{K}' := \mathcal{K} \cup (\cup_{v \in V \setminus V_{\mathcal{K}}} \{\{v\}\})$ be the set of atoms. We will also assume that two vertices in the same atom have exactly the same neighbors in E_{adm} . This can be ensured by, for every triple uvw with u, v in the same atom and $vw \in E_{\text{adm}}$, $uw \notin E_{\text{adm}}$, removing vw from E_{adm} ; any good clustering cannot put v and w in the same cluster, so it is safe to remove it.

5.1 Relaxation and Algorithm

Given $x \in [0, 1]^{\binom{V}{2}}$, for some constant $r = O(1/\varepsilon^2)$, we use the following LP. For a set $S \subseteq V$, y_S indicates whether S is a subset of a cluster in an optimal solution. We also add the constraint that $x_{uv} = 1 - y_{uv}$, which will ensure that $y_{uv} = 1$ if u and v are in the same atom and $y_{uv} = 0$ if uv is non-admissible.

Apart from the constraints imposed by x , this is a weaker version of r -rounds of the Sherali-Adams relaxation introduced by [CLN22]. We present this relaxation to present a smaller set of constraints used in our algorithm (e.g., this relaxation also works with the rounding algorithm of [CLN22]) and be consistent with the LP introduced in Section 4. In particular, apart from the constraint (16) used in the triangle analysis (its LHS indicates the event all u, v, w are in different clusters; see Section 5.4), it is a simpler version of the LP used in Section 4 that does not distinguish sets of different sizes and does not involve additional variables \tilde{x} .

²We remark that the derandomization step is necessary due to our round-or-cut paradigm. Since we can not control whether the future iterations will succeed or fail (i.e., return a separation plane), we could not guarantee that the expected cost we pay is the expected budget we get, conditioned on that the algorithm succeeds.

$$y_{uv} = 1 - x_{uv} \quad \forall uv \in \binom{V}{2} \quad (13)$$

$$y_u = 1 \quad \forall u \in V \quad (14)$$

$$\sum_{T' \subseteq T} (-1)^{|T'|} y_{S \cup T'} \in [0, y_S] \quad \forall S \cap T = \emptyset, |S \cup T| \leq r \quad (15)$$

$$1 - (y_{vw} + y_{wu} + y_{uv} - 2y_{uvw}) \geq 0 \quad \forall u, v, w \in V \quad (16)$$

$$y \geq 0. \quad (17)$$

Any integral good clustering with respect to $(\mathcal{K}, E_{\text{adm}})$ yields a feasible solution to the above LP. Therefore, if the above LP is not feasible for the given x , it implies that x is not a convex combination of good clustering, and by Lemma 6, the algorithm can yield a hyperplane separating x from the convex hull of good clusterings.

For the rest of the section, we assume that y is an optimal solution for the above relaxation in the initial graph. The rounding algorithm is given in Algorithm 4. Let $N^+(v)$ and $N_{\text{adm}}^+(v)$ denote the neighbors of v with respect to E^+ and $E^+ \cap E_{\text{adm}}$ respectively. $N^-(v)$ and $N_{\text{adm}}^-(v)$ are defined similarly.

Algorithm 4 Pivot-based Rounding

Input: A Correlation Clustering instance $G = (V, E^+ \cup E^-)$, a preclustered instance $(\mathcal{K}, E_{\text{adm}})$ for G , and a metric $x \in [0, 1]^{\binom{V}{2}}$ satisfying properties in Theorem 5

Output: Either a separation plane for x , or an integral clustering \mathcal{C}

- 1: Run the above LP to obtain a feasible solution y . If not feasible, return a separation plane for x
 - 2: $\mathcal{C} \leftarrow \emptyset$
 - 3: **while** $V \neq \emptyset$ **do**
 - 4: $S \leftarrow \text{Cleanup}(V)$ (Algorithm 5)
 - 5: **if** $S \neq \emptyset$ **then**
 - 6: $\mathcal{C} \leftarrow \mathcal{C} \cup \{S\}$, $V \leftarrow V \setminus S$, **continue**
 - 7: Pick a pivot $p \in V$ uniformly at random
 - 8: **for** $v \in N^-(p)$ **do**
 - 9: Add v independently to S^- with probability y_{pv}
 - 10: Let K be the atom containing p
 - 11: Sample $S^+ \subseteq N_{\text{adm}}^+(p)$ from the correlated rounding procedure (Lemma 22)
 - 12: $S \leftarrow S^- \cup S^+ \cup K$, $\mathcal{C} \leftarrow \mathcal{C} \cup \{S\}$, $V \leftarrow V \setminus S$
 - 13: **return** \mathcal{C}
-

The cleanup subroutine is described in Algorithm 5. Let $E^+(V, K)$ be the set of +edges incident on K . Its meaning will be more clear after we introduce the setup for the analysis using budgets in Section 5.2.

Algorithm 5 Cleanup(V)

- 1: **for** each atom $K \in \mathcal{K}'$ **do**
 - 2: Let ALG_K be the cost of removing K as a single cluster, which is the number of $-$ edges inside K plus the number of $+$ edges between K and $V \setminus K$.
 - 3: Let Δ_K be the decrease in the budget if we remove K from V ; formally, $\Delta' := (\sum_{uv \in E^+(K, V)} \min(1.515 + x_{uv}, 2)x_{uv}) + (\sum_{uv \in E^-(K, V)} 2(1 - x_{uv})) + \varepsilon \cdot |\{uv \in E_{\text{adm}} : u \in K \text{ or } v \in K\}|$.
 - 4: **If** $\Delta_K \geq ALG_K$, **return** K
 - 5: **return** \emptyset
-

Algorithm 4 also uses the correlated rounding procedure of Raghavendra and Tan [RT12], adapted for Correlation Clustering in [CLN22]. The guarantee of the correlated rounding is as follows.

Lemma 22. *In Step 11 of Algorithm 4, one can sample $S^+ \subseteq N_{\text{adm}}^+(p)$ in time $n^{O(r)}$ such that*

- For each $v \in N_{\text{adm}}^+(p)$, $\Pr[v \in S^+] = y_{pv}$.
- $\mathbb{E}_{u,v \in N_{\text{adm}}^+(p)} [|\Pr[u, v \in S^+] - y_{puv}|] \leq \varepsilon_r$, where $\varepsilon_r = O(1/\sqrt{r})$.

Even though we present the algorithm as a randomized algorithm, a standard derandomization using conditional expectations will deterministically yield a clustering whose cost is as good as the expected value [CLN22].

5.2 Setup for Analysis

Our high-level setup of the analysis follows from [CLN22], which in turn is based on [ACN08] and [CMSY15], with a slight change that instead of the total LP value remaining in the current instance, we consider a more general *budget* remaining in the current instance, which already incorporates the approximation ratio. For each $uv \in E^+$, give the budget of $\min(1.515 + x_{uv}, 2)x_{uv}$, and for each $uv \in E^-$, give the budget of $2(1 - x_{uv})$ (call them the *LP budget*). Furthermore, for each $uv \in E_{\text{adm}}$, give an additional budget of ε (call it the *error budget*). Then it is clear that the total initial budget is exactly the desired upper bound on the (expected) cost of the clustering (the RHS of Theorem 8).

Consider the t -th iteration of Algorithm 4 with the vertex set V_t and suppose that the algorithm obtains the cluster S in this iteration by building it from a random pivot p (instead of using the cleanup). Let $\text{cost}_p^r(u, v)$ be the probability that uv is violated in the rounding algorithm when p is the pivot, and $\Delta_p^r(u, v)$ be the total budget of uv (LP and error budget combined) times the probability that uv disappears (i.e., $\Pr[S \cap \{u, v\} \neq \emptyset]$). The superscript r stands for (actual) rounding.

We call a set of three distinct vertices a *triangle*. A set of two vertices is called a *degenerate triangle*. For triangle uvw , let $\text{cost}^r(u, v, w) = \text{cost}_u^r(v, w) + \text{cost}_v^r(u, w) + \text{cost}_w^r(u, v)$ and $\Delta^r(u, v, w) = \Delta_u^r(v, w) + \Delta_v^r(u, w) + \Delta_w^r(u, v)$. For degenerate triangle uv , let $\text{cost}^r(u, v) = \text{cost}_u^r(u, v) + \text{cost}_v^r(u, v)$ and $\Delta^r(u, v) = \Delta_u^r(u, v) + \Delta_v^r(u, v)$. Let

$$ALG_t := \mathbb{E}_{u \in V} \sum_{vw \in \binom{V_t}{2}} \text{cost}_u^r(v, w)$$

be the expected cost incurred by this iteration, and

$$\Delta_t := \mathbb{E}_{u \in V} \sum_{vw \in \binom{V_t}{2}} \Delta_u^r(v, w)$$

be the expected amount of the budget removed by this iteration. If we could show that $ALG_t \leq \Delta_t$ for all t , we will get an upper bound on the total cost \mathbf{ALG} as

$$\mathbb{E}[\mathbf{ALG}] = \mathbb{E}\left[\sum_{t=0}^R ALG_t\right] \leq \mathbb{E}\left[\sum_{t=0}^R \Delta_t\right] = \Phi$$

where Φ is the initial budget and R is the number of the iterations.

Notice that, even when the cluster S is chosen from the cleanup step, ALG_t and Δ_t can be still defined as the incurred cost and removed budget respectively, and the design of the cleanup step (Algorithm 5) exactly ensures that we remove S when $ALG_t \leq \Delta_t$ deterministically.

Therefore, in order to prove Theorem 8, it suffices to consider one iteration where S is built from a random pivot p . For the rest of Section 5, let us omit the subscript t denoting the iteration. We prove $ALG \leq \Delta$, which is equivalent to showing

$$\sum_{uvw \in \binom{V}{3}} \text{cost}^r(u, v, w) + \sum_{uv \in \binom{V}{2}} \text{cost}^r(u, v) \leq \sum_{uvw \in \binom{V}{3}} \Delta^r(u, v, w) + \sum_{uv \in \binom{V}{2}} \Delta^r(u, v). \quad (18)$$

Recall that a triangle is $+++$ if it has three $+$ -edges and $+-$, $+-$, $---$ triangles are defined similarly. For a degenerate triangle uv , $\text{cost}_u^r(u, v)$ and $\Delta_u^r(u, v)$ depend only on x_{uv} and the sign of uv . Even for a triangle uvw , the values of $\text{cost}_u^r(v, w)$ and $\Delta_u^r(v, w)$ only depend on x_{uv}, x_{uw}, x_{vw} and the signs and admissibilities of the edges unless both uv and uw are admissible $+$ -edges; v and w are added to S independently with the probabilities depending on x_{uv} and x_{uw} respectively. When both uv and uw are admissible $+$ -edges, then they are rounded using Lemma 22, and $\Pr[v, w \in S^+ | u \text{ is pivot}]$ must be, ideally, exactly equal to y_{uvw} , but Lemma 22 only gives an approximate guarantee amortized over the vertices in $N_{\text{adm}}^+(u)$.

We define the following idealized quantities $\text{cost}^i(\cdot)$ and $\Delta^i(\cdot)$. (The superscript i stands for ideal.) Intuitively, $\text{cost}^i(\cdot)$ and $\Delta^i(\cdot)$ are defined assuming that the correlated rounding for admissible $+$ -edges are perfect, and we do not consider the error budget for Δ . Formally, $\text{cost}_u^i(\cdot), \text{cost}^i(\cdot), \Delta_u^i(\cdot), \Delta^i(\cdot)$ are defined identically to $\text{cost}_u^r(\cdot), \text{cost}^r(\cdot), \Delta_u^r(\cdot), \Delta^r(\cdot)$ respectively, assuming that in Step 11 of Algorithm 4, the condition (2) is replaced by $\Pr[u, v \in S^+ | p \text{ is pivot}] = y_{puv}$ for every $p \in V, u, v \in N_{\text{adm}}^+(p)$ and the error budget of ε is not accounted in Δ 's. With this assumption, note that for every triangle uvw both $\text{cost}^i(u, v, w)$ and $\Delta^i(u, v, w)$ depend only on the signs of the edges and the Sherali-Adams solution induced by uvw (i.e., $y_{uvw}, y_{uv}, y_{vw}, y_{wu}$). Then one can show that $\text{cost}^i(T) \leq \Delta^i(T)$ for any triangle T . The proof of the following lemma appears in Section 5.4.

Lemma 23. *For any triangle T , $\text{cost}^i(T) \leq \Delta^i(T)$.*

5.3 Incorporating Errors

This subsection shows how to incorporate errors and finishes the proof of Theorem 8 assuming Lemma 23. To prove the theorem, as explained in Section 5.2, it suffices to show that $\mathbb{E}[ALG] \leq \mathbb{E}[\Delta]$ in one iteration where **Cleanup**(V) returns \emptyset and Algorithm 4 proceeds by choosing a random pivot p .

When p is chosen as a pivot, let $N_{\text{adm}}^+(p)$ be the set of vertices connected to p via an admissible $+$ -edge. By Lemma 22, compared to the ideal case, ALG is increased by at most $\varepsilon_r \cdot (|N_{\text{adm}}^+(p)|)$. The main challenge is to show that this can be compensated by the increase in Δ due to the additional error budget ε for each admissible edge.

Let $\text{err}_{vw|u}$ be ε_r when both uv and uw are admissible $+$ edges and 0 otherwise. Recall that $\varepsilon_r = O(1/\sqrt{r})$ be the error parameter from Lemma 22. Similarly, $\Delta'_u(v, w) := (1 - \min(x_{uv}, x_{uw}))\varepsilon$ is an *lower bound* on the expected error budget decrease from if vw if it is admissible, and 0 otherwise. (It is an lower bound because when u is the pivot, since our rounding algorithm satisfies $\Pr[v \in S] = y_{uv}$ for all v, vw will be removed with probability at least $1 - \min(x_{uv}, x_{uw})$.) By letting

$$ALG^i := \sum_{u \in V, vw \in \binom{V}{2}} \text{cost}_u^i(v, w) \quad \text{and} \quad ALG' := \sum_{u \in V, vw \in \binom{V}{2}} \text{err}_{vw|u}$$

and

$$\Delta^i := \sum_{u \in V, vw \in \binom{V}{2}} \Delta_u^i(v, w) \quad \text{and} \quad \Delta' := \sum_{u \in V, vw \in \binom{V}{2}} \Delta'_u(v, w).$$

we have that

$$|V| \cdot ALG \leq ALG^i + ALG'$$

and

$$|V| \cdot \Delta \geq \Delta^i + \Delta'.$$

Since Lemma 23 shows that $ALG^i \leq \Delta^i$, it suffices to show that $ALG' \leq \Delta'$.

We prove it on an atom-by-atom basis. Fix an atom $K \in \mathcal{K}'$ (again, it might be a singleton outside $V_{\mathcal{K}}$). Let N be the set of their admissible neighbors. Recall that if every vertex in K has exactly the same set of

neighbors with respect to E_{adm} . Let $E' = K \times N$ be the set of pairs between them (all admissible), which is partitioned into E'_+ and E'_- . Then K 's contribution to ALG' is

$$ALG'_K := \sum_{u \in K, vw \in \binom{N}{2}: uv, uw \in E'_+} \varepsilon_r \leq |K||N|^2 \varepsilon_r,$$

such that $\sum_{K \in \mathcal{K}'} ALG'_K = ALG'$. We can also define K 's contribution to Δ' as

$$\Delta'_K := \sum_{u \in N, vw \in E'} \Delta'_u(v, w) = \sum_{u \in N, vw \in E'} (1 - \min(x_{uv}, x_{uw})) \varepsilon,$$

so that $\sum_{K \in \mathcal{K}'} \Delta'_K \leq 2\Delta'$. (For each triple uvw , $\Delta'_u(v, w)$ is counted at most twice, for atoms containing v or w .) Therefore, it suffices to argue that $2ALG'_K \leq \Delta'_K$.

We use the fact that K was not removed as a single cluster from the cleanup stage, which means that

$$\begin{aligned} ALG_K &= |E^- \cap \binom{K}{2}| + |E^+ \cap (K \times (V \setminus K))| \\ &\geq \Delta_K = \sum_{uv \in (E^+ \cap (K \times V))} \min\{1.515 + x_{uv}, 2\} \cdot x_{uv} + \sum_{uv \in (E^- \cap (K \times V))} 2(1 - x_{uv}) + \varepsilon|E'|. \end{aligned}$$

Note that for each $e \in E^- \cap \binom{K}{2}$, it is an atomic edge with $x_e = 0$, so it contributes exactly 1 to ALG_K and at least 1 to Δ_K . Similarly, for each $e \in (E^+ \cap (K \times V)) \setminus E'_+$, it is a non-admissible edge with $x_e = 1$, so it contributes exactly 1 to ALG_K and at least 1 to Δ_K . Then, by only considering edges in E' , one can conclude that

$$|E'_+| \geq \sum_{uv \in E'_+} \min\{1.515 + x_{uv}, 2\} \cdot x_{uv} + \sum_{uv \in E'_-} 2(1 - x_{uv}) + \varepsilon|E'|,$$

which implies that $|E'_+| \geq \varepsilon|E'|$ and $|E'_+| \geq \sum_{uv \in E'_+} 1.5x_{uv}$, so that $\mathbb{E}_{uv \in E'_+} [x_{uv}] \leq 2/3$ and $\mathbb{E}_{uv \in E'} [x_{uv}] \leq 1 - \varepsilon/3$. Then one can lower bound Δ'_K as

$$\Delta'_K = \sum_{u \in N, vw \in E'} \varepsilon(1 - \min(x_{uv}, x_{uw})) \geq \sum_{w \in N, u \in N, v \in K} \varepsilon(1 - x_{uw}) = |N| \sum_{uv \in E'} \varepsilon(1 - x_{uv}) \geq |N|^2 |K| (\varepsilon/3).$$

Therefore, by ensuring that $\varepsilon_r \leq \varepsilon/6$, we can ensure that $ALG'_K \leq \Delta'_K/2$ for every K and $ALG' \leq K'$ eventually.

5.4 Analysis of Error-Free Version

This subsection proves Lemma 23. Specifically, for every triangle abc , we prove

$$\text{cost}_a^i(b, c) + \text{cost}_b^i(c, a) + \text{cost}_c^i(a, b) \leq \Delta_a^i(b, c) + \Delta_b^i(c, a) + \Delta_c^i(a, b). \quad (19)$$

When two of a, b, c are the same vertex, it is easy to see that (19) is true [CLN22]. Therefore, we assume that a, b, c are distinct vertices. We consider the four types of triangles according to the signs of their edges. Recall the coefficient of the LP budget for a +edge e is $f(x_e)$ defined as $f(x) = \min(1.515 + x, 2)$, whereas the coefficient for a -edge e is always 2. Also, note that given $y_{ab}, y_{bc}, y_{ca}, y_{abc}$, we can define $y_{a|b} = y_a - y_{ab}$ indicating the event a and b are separate (similarly for $y_{b|c}, y_{c|a}$), and $y_{a|bc} = y_{bc} - y_{abc}$ indicating the event b, c are together but a is separate, and $y_{a|b|c} = 1 - (y_{ab} + y_{bc} + y_{ca} - 2y_{abc})$ indicating the event a, b, c are all separate. By the constraint (16), all of them are nonnegative with $y_{abc} + y_{a|bc} + y_{b|ca} + y_{c|ab} + y_{a|b|c} = 1$, so they exactly define the probability distribution over partitions on $\{a, b, c\}$. (These variables were explicitly defined in the Sherali-Adams relaxation of [CLN22].)

+++ **triangles** [CLN22] already proved that in their setting, the approximation ratio for +++ triangles is at most 1.5. In our setting, as we have $f(x) \geq 1.5$ for every $x \in [0, 1]$, (19) holds.

--- **triangles** For a --- triangle abc , (19) holds even if we set the coefficients for -edges to 1 (instead of 2):

$$\begin{aligned} \text{left side of (19)} &= y_{ab}y_{ac} + y_{ab}y_{bc} + y_{ac}y_{bc}. \\ \text{right side of (19)} &\geq (y_{ab} + y_{ac} - y_{ab}y_{ac})y_{bc} + (y_{ab} + y_{bc} - y_{ab}y_{bc})y_{ac} + (y_{ac} + y_{bc} - y_{ac}y_{bc})y_{ab} \\ &= 2(y_{ab}y_{ac} + y_{ab}y_{bc} + y_{ac}y_{bc}) - 3y_{ab}y_{ac}y_{bc} \geq y_{ab}y_{ac} + y_{ab}y_{bc} + y_{ac}y_{bc}. \end{aligned}$$

+-- **triangles** For a +-- triangle abc , we assume ab and ac are -edges and bc is the +edge. We show that (19) holds even if we set the coefficient for bc to 1:

$$\begin{aligned} \text{left side of (19)} &= y_{ab} + y_{ac} - 2y_{ab}y_{ac} + y_{ab}y_{bc} + y_{ac}y_{bc} = (2 - x_{bc})(y_{ab} + y_{ac}) - 2y_{ab}y_{ac}. \\ \text{right side of (19)} &\geq (y_{ab} + y_{ac} - y_{ab}y_{ac})x_{bc} + 2(y_{ab} + y_{bc} - y_{ab}y_{bc})y_{ac} + 2(y_{ac} + y_{bc} - y_{ac}y_{bc})y_{ab} \\ &= (y_{ab} + y_{ac} - y_{ab}y_{ac})x_{bc} + 2(1 - x_{bc} + y_{ab}x_{bc})y_{ac} + 2(1 - x_{bc} + y_{ac}x_{bc})y_{ab} \\ &= (2 - x_{bc})(y_{ab} + y_{ac}) + y_{ab}y_{ac}x_{bc}. \end{aligned}$$

So the left side is at most the right side.

++- **triangles** It remains to consider the ++- triangle case, which contributes to the bulk of the analysis. Focus on a ++- triangle abc . We assume ab and ac are +edges, and bc is the -edge. The left side of (19) is

$$y_{abc} + y_{ab} + y_{bc} - 2y_{ab}y_{bc} + y_{ac} + y_{bc} - 2y_{ac}y_{bc} = y_{abc} + y_{ab} + y_{ac} + 2y_{bc} - 2(y_{ab} + y_{ac})y_{bc}.$$

The right side of (19) is

$$\begin{aligned} &2(y_{ab} + y_{ac} - y_{abc})y_{bc} + f(x_{ac})(y_{ab} + y_{bc} - y_{ab}y_{bc})(1 - y_{ac}) + f(x_{ab})(y_{ac} + y_{bc} - y_{ac}y_{bc})(1 - y_{ab}) \\ &= f(x_{ac})y_{ab} + f(x_{ab})y_{ac} + (f(x_{ab}) + f(x_{a,c}))y_{bc} + (2 - f(x_{ab}) - f(x_{ac}))(y_{ab} + y_{ac})y_{bc} \\ &\quad - (f(x_{ab}) + f(x_{ac}))y_{ab}y_{ac}(1 - y_{bc}) - 2y_{abc}y_{bc}. \end{aligned}$$

The right side of (19) minus the left side is

$$\begin{aligned} &(f(x_{ac}) - 1)y_{ab} + (f(x_{ab}) - 1)y_{ac} + (f(x_{ab}) + f(x_{a,c}) - 2)y_{bc} + (4 - f(x_{ab}) - f(x_{ac}))(y_{ab} + y_{ac})y_{bc} \\ &\quad - (f(x_{ab}) + f(x_{ac}))y_{ab}y_{ac}(1 - y_{bc}) - 2y_{abc}y_{bc} - y_{abc}. \end{aligned} \tag{20}$$

We need to prove that (20) is non-negative. We define $y_{ab|c} = y_{ab} - y_{abc}$ to indicate if ab are in the same cluster that does not contain c ; define $y_{a|bc}$ and $y_{b|ac}$ similarly. Define $y_{a|b|c} = 1 - (y_{ab} + y_{ac} + y_{bc}) + 2y_{abc}$ to indicate if a, b, c are in three different clusters; this is at least 0 by (16). So $y_{ab|c}, y_{ac|b}, y_{a|bc}, y_{abc}$ and $y_{a|b|c}$ indicate the 5 cases for the clustering status of a, b, c . They are non-negative reals summing up to 1. Moreover, $y_{ab} = y_{ab|c} + y_{abc}$, $y_{ac} = y_{ac|b} + y_{abc}$ and $y_{bc} = y_{a|bc} + y_{abc}$.

First we show that moving mass from $y_{a|bc}$ to $y_{a|b|c}$ can only decrease (20). Notice that this operation does not change $y_{abc}, y_{ab}, x_{ab}, y_{ac}$ and x_{ac} , and it decreases y_{bc} . The derivative of (20) w.r.t y_{bc} is at least $f(x_{ab}) + f(x_{ac}) - 2 + (f(x_{ab}) + f(x_{ac}))y_{ab}y_{ac} - 2y_{abc} \geq 1 + 3y_{abc}^2 - 2y_{abc} \geq 0$. Therefore, we can assume $y_{a|bc} = 0$, and thus $y_{bc} = y_{abc} \leq \min\{y_{ab}, y_{ac}\}$. (20) becomes

$$\begin{aligned} &(f(x_{ac}) - 1)y_{ab} + (f(x_{ab}) - 1)y_{ac} + (f(x_{ab}) + f(x_{a,c}) - 3)y_{bc} + (4 - f(x_{ab}) - f(x_{ac}))(y_{ab} + y_{ac})y_{bc} \\ &\quad - (f(x_{ab}) + f(x_{ac}))y_{ab}y_{ac}(1 - y_{bc}) - 2y_{bc}^2. \end{aligned} \tag{21}$$

It remains to prove that (21) is non-negative, under the condition that $y_{ab}, y_{ac}, y_{bc} \in [0, 1]$ and $(y_{ab} + y_{ac} - 1)_+ \leq y_{bc} \leq \min\{y_{ab}, y_{ac}\}$.

Fixing y_{ab} and y_{ac} , (21) is a quadratic function of y_{bc} . So it is minimized when $y_{bc} = (y_{ab} + y_{ac} - 1)_+$ or $y_{bc} = \min\{y_{ab}, y_{ac}\}$, as the coefficient for the quadratic term y_{bc}^2 is -2 . We consider three cases.

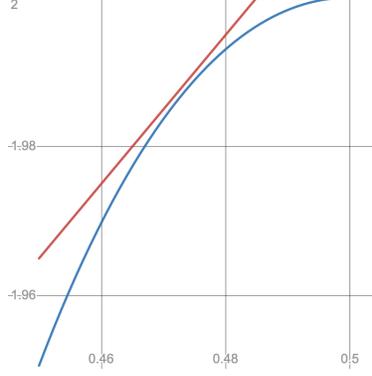


Figure 1: The function $f(x) = \min\{1.515 + x, 2\}$ (the red line) and $\frac{-1+4x-2x^2}{x^2}$ (the blue line) over the interval $[0.45, 0.5]$.

Case 1: $y_{ab} + y_{ac} \leq 1$ and $y_{bc} = 0$. In this case, (21) becomes

$$(f(x_{ac}) - 1)y_{ab} + (f(x_{ab}) - 1)y_{ac} - (f(x_{ab}) + f(x_{ac}))y_{ab}y_{ac}. \quad (22)$$

If both of y_{ab} and y_{ac} are at most $1/2$, then $x_{ab} \geq 1/2$ and $x_{ac} \geq 1/2$. Thus $f(x_{ab}) = f(x_{ac}) = 2$. (22) is $y_{ab} + y_{ac} - 4y_{ab}y_{ac} \geq 0$, as $y_{ab} \geq 2y_{ab}y_{ac}$ and $y_{ac} \geq 2y_{ab}y_{ac}$.

So, we assume exactly one of y_{ab} and y_{ac} is at least $1/2$. Wlog, we assume $y_{ab} < 1/2 \leq y_{ac}$. Thus $x_{ac} \leq 1/2 < x_{ab}$ and $f(x_{ab}) = 2$. (22) becomes $(f(x_{ac}) - 1)y_{ab} + y_{ac} - (2 + f(x_{ac}))y_{ab}y_{ac}$. Fixing $y_{ac} \geq 1/2$ and $x_{ac} = 1 - y_{ac}$, the function is linear in y_{ab} . Thus it is minimized when $y_{ab} = 0$ or $y_{ab} = 1 - y_{ac}$. In the former case, (22) is $y_{ac} > 0$. In the latter case, it is

$$\begin{aligned} & (f(x) - 1)x + 1 - x - (2 + f(x))x(1 - x) = (x - x(1 - x))f(x) + 1 - 2x - 2x(1 - x) \\ & = x^2 f(x) + 1 - 4x + 2x^2, \end{aligned}$$

where $x := x_{ac} = 1 - y_{ac} = y_{ab} \in [0, 1/2]$. The number 1.515 is chosen so that we have $\min\{1.515 + x, 2\} \geq \frac{-1+4x-2x^2}{x^2}$ for every $x \in [0, 1/2]$. See Figure 1. So in this case (21) is at least 0.

Case 2: $y_{ab} + y_{ac} \geq 1$ and $y_{bc} = y_{ab} + y_{ac} - 1$. We further divide this case into many sub-cases, which by the symmetry between y_{ab} and y_{ac} , cover the whole Case 2.

- Case 2a: $y_{ab} \leq 0.515, y_{ac} \in (0.515, 0.69]$. Then $x_{ab} \geq 0.485$ and $x_{ac} < 0.485$. So, $f(x_{ab}) = 2$ and $f(x_{ac}) = 1.515 + x_{ac} = 2.515 - y_{ac}$. (21) equals

$$\begin{aligned} & (1.515 - y_{ac})y_{ab} + y_{ac} + (1.515 - y_{ac})y_{bc} + (y_{ac} - 0.485)(y_{ab} + y_{ac})y_{bc} \\ & - (4.515 - y_{ac})y_{ab}y_{ac}(1 - y_{bc}) - 2y_{bc}^2. \end{aligned} \quad (23)$$

We fix y_{ac} . Consider the operation of increasing y_{ab} and y_{bc} at rate 1 so as to keep $y_{bc} = y_{ab} + y_{ac} - 1$. (23) will increase at a ratio of

$$\begin{aligned} & 1.515 - y_{ac} + 1.515 - y_{ac} + (y_{ac} - 0.485)(y_{ab} + y_{ac} + y_{bc}) - (4.515 - y_{ac})y_{ac}(1 - y_{bc} - y_{ab}) - 4y_{bc} \\ & = (3.03 - 2y_{ac}) + (y_{ac} - 0.485)(2y_{ab} + 2y_{ac} - 1) - (4.515 - y_{ac})y_{ac}(2 - 2y_{ab} - y_{ac}) - 4(y_{ab} + y_{ac} - 1) \\ & = 7.515 - 4.97y_{ab} - 17y_{ac} + 11.03y_{ab}y_{ac} + 8.515y_{ac}^2 - 2y_{ab}y_{ac}^2 - y_{ac}^3. \end{aligned} \quad (24)$$

We then show (24) is positive for $y_{ac} \in (1/2, 0.69]$. For a fixed y_{ac} , the quantity is a linear function of y_{ab} , and the coefficient for the y_{ab} -term is $-4.97 + 11.03y_{ac} - 2y_{ac}^2$. This function is at least 0 for

$y_{ac} \in [1/2, 1]$ as it is monotone increasing from ∞ to $11.03/4$, and its value is positive when $y_{ac} = 1/2$. Therefore (24) is minimized when y_{ab} is minimized; that is, $y_{bc} = 1 - y_{ac}$. In this case, (24) becomes

$$\begin{aligned} & 7.515 - 4.97(1 - y_{ac}) - 17y_{ac} + 11.03(1 - y_{ac})y_{ac} + 8.515y_{ac}^2 - 2y_{ac}^2(1 - y_{ac}) - y_{ac}^3 \\ &= 2.545 - y_{ac} - 4.515y_{ac}^2 + y_{ac}^3. \end{aligned}$$

For $y_{ac} \in [1/2, 0.69]$, the function is at least 0, which implies that (24) is at least 0.

So, the operation of decreasing y_{ab} and y_{bc} at the same rate can only decrease the (23). Thus (23), which is equal to (21), is minimized when $y_{ab} = 1 - y_{ac}$ and $y_{bc} = 0$. This is already considered in Case 1.

- Case 2b: $y_{ab} \leq 0.515, y_{ac} \in [0.69, 1]$. In this case, we shall simply use 1.5 for $f(x_{a,c})$ and 2 for $f(x_{ab})$. That is, (21) is lower bounded by

$$0.5y_{ab} + y_{ac} + 0.5y_{bc} + 0.5(y_{ab} + y_{ac})y_{bc} - 3.5y_{ab}y_{ac}(1 - y_{bc}) - 2y_{bc}^2. \quad (25)$$

Consider the operation of increasing y_{ab} and decreasing y_{ac} at the same rate. This does not change $y_{ab} + y_{ac}$ and $y_{bc} = y_{ab} + y_{ac} - 1$. It increases $y_{ab}y_{ac}$. Then it is easy to see that the operation will decrease the above quantity. So, the above quantity is minimized either when $y_{ab} = 1/2$ and $y_{ac} \geq 0.69$, or when $y_{ab} < 1/2$ and $y_{ac} = 0.69$ (we shall have $y_{ab} \geq 0.31$ as we have $y_{ab} + y_{ac} \geq 1$).

When $y_{ab} = 1/2$ and $y_{ac} \geq 0.69$, we consider the operation of increasing y_{ac} and y_{bc} at the same rate to maintain $y_{bc} = y_{ab} + y_{ac} - 1 = y_{ac} - 1/2$. (25) will increase at a rate of $1 + 0.5 + 0.5(y_{ab} + y_{ac} + y_{bc}) - 3.5y_{ab}(1 - y_{bc} - y_{ac}) - 4y_{bc} = 1.5 + 0.5 \cdot 2y_{ac} - 1.75 \cdot (1.5 - 2y_{ac}) - 4(y_{ac} - 1/2) = 0.5y_{ac} + 0.875 > 0$. So, the operation of decreasing y_{ac} and y_{bc} at the same rate will decrease (25). So, (25) is minimized when $y_{ab} = 1/2$ and $y_{ac} = 0.69$. This will be covered by the second case.

When $y_{ab} < 1/2$ and $y_{ac} = 0.69$, we consider the operation of increasing y_{ab} and y_{bc} at the same rate, so as to maintain $y_{bc} = y_{ab} + y_{ac} - 1 = y_{ab} - 0.31$. (25) will increase at a rate of $0.5 + 0.5 + 0.5(y_{ab} + y_{ac} + y_{bc}) - 3.5y_{ac}(1 - y_{bc} - y_{ab}) - 4y_{bc} = 1 + 0.5(2y_{ab} + 0.69 - 0.31) - 3.5 \cdot 0.69(1.31 - 2y_{ab}) - 4(y_{ab} - 0.31) = 1.83y_{ab} - 0.73365$. This is negative when $y_{ab} < \frac{0.73365}{1.83}$ and positive when $y_{ab} > \frac{0.73365}{1.83}$. So, (25) is minimized when $y_{ab} = \frac{0.73365}{1.83} \in [0.4, 0.401]$ and $y_{bc} = y_{ab} - 0.31 \in [0.09, 0.091]$. In this case, (25) is at least

$$0.5 \times 0.4 + 0.69 + 0.5 \cdot 0.09 + 0.5 \cdot 1.09 \cdot 0.09 - 3.5 \cdot 0.401 \cdot 0.69 \cdot 0.91 - 2 \cdot 0.091^2 \geq 0.086 > 0.$$

So, we have proved that (25) is at least 0 in this case, which implies (21) is at least 0.

- Case 2c: $y_{ab} \geq 0.515, y_{ac} \geq 0.515$. Then $x_{ab} \leq 0.485$ and $x_{ac} \leq 0.485$. $f(x_{a,b}) \geq 1.5 + x_{ab} = 2.5 - y_{ab}$, and $f(x_{ac}) \geq 2.5 - y_{ac}$. (21) is lower bounded by

$$\begin{aligned} & (1.5 - y_{ac})y_{ab} + (1.5 - y_{ab})y_{ac} + (2 - y_{ab} - y_{ac})y_{bc} + (y_{ab} + y_{ac} - 1)(y_{ab} + y_{ac})y_{bc} \\ & - (5 - y_{ab} - y_{ac})y_{ab}y_{ac}(1 - y_{bc}) - 2y_{bc}^2. \end{aligned} \quad (26)$$

Notice that $(1.5 - y_{ac})y_{ab} + (1.5 - y_{ab})y_{ac} = 1.5(y_{ab} + y_{ac}) - 2y_{ab}y_{ac}$. We fix the sum $y_{ab} + y_{ac}$ and $y_{bc} = y_{ab} + y_{ac} - 1$ is fixed, while changing $y_{ab}y_{ac}$. The coefficient for $y_{ab}y_{ac}$ is

$$-2 - (5 - y_{ab} - y_{ac})(1 - y_{bc}) \leq 0.$$

So (26) is minimized when $y_{ab} = y_{ac}$. Letting $y = y_{ab} = y_{ac} \in [1/2, 1]$, and $y_{bc} = 2y - 1$, (26) becomes

$$\begin{aligned} & 2(1.5 - y)y + (2 - 2y)(2y - 1) + (2y - 1)^2 \cdot 2y - (5 - 2y) \cdot y^2 \cdot (2 - 2y) - 2(2y - 1)^2 \\ &= -4y^4 + 22y^3 - 32y^2 + 19y - 4. \end{aligned}$$

This is monotone over $y \in [1/2, 1]$ and so its minimum is 0, achieved at $y = 1/2$. So in this case, (21) is at least 0.

Case 3: $y_{bc} = \min\{y_{ab}, y_{ac}\}$. We assume $y_{bc} = y_{ab} \leq y_{ac}$ wlog. In this case, we can use the lower bound 1.5 for both $f(x_{ab})$ and $f(x_{ac})$. That is, (21) is at least

$$\begin{aligned} & 0.5y_{ab} + 0.5y_{ac} + (y_{ab} + y_{ac})y_{bc} - 3y_{ab}y_{ac}(1 - y_{bc}) - 2y_{bc}^2 \\ &= 0.5y_{ab} + 0.5y_{ac} + y_{ab}^2 + y_{ab}y_{ac} - 3y_{ab}y_{ac} + 3y_{ab}^2y_{ac} - 2y_{ab}^2 \\ &= 0.5y_{ab} + 0.5y_{ac} - y_{ab}^2 - 2y_{ab}y_{ac} + 3y_{ab}^2y_{ac}. \end{aligned} \tag{27}$$

Fix y_{ab} in (27). The coefficient for y_{ac} is $0.5 - 2y_{ab} + 3y_{ab}^2$, which is always non-negative. So the quantity is minimized when $y_{ab} = y_{ac}$. Under this condition (27) becomes

$$y_{ab} - 3y_{ab}^2 + 3y_{ab}^3 = y_{ab}(1 - 3y_{ab} + 3y_{ab}^2) \geq 0, \text{ for every } y_{ab} \geq 0.$$

Therefore, (21) is non-negative in this case.

So, we have proved that (19) holds for all triangles abc .

Acknowledgements

Vincent Cohen-Addad, Euiwoong Lee, and Alantha Newman are grateful to Claire Mathieu and Farid Arthaud for valuable discussions at an early stage of this project.

References

- [ACG⁺15] Kook Jin Ahn, Graham Cormode, Sudipto Guha, Andrew McGregor, and Anthony Wirth. Correlation clustering in data streams. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pages 2237–2246, 2015. 6
- [ACN08] Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: Ranking and clustering. *Journal of the ACM*, 55(5):1–27, 2008. 1, 2, 4, 20
- [AHK⁺09] Rakesh Agrawal, Alan Halverson, Krishnaram Kenthapadi, Nina Mishra, and Panayiotis Tsaparas. Generating labels from clicks. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 172–181, 2009. 1
- [ARS09] Arvind Arasu, Christopher Ré, and Dan Suciu. Large-scale deduplication with constraints using dedupalog. In *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE)*, pages 952–963, 2009. 1
- [AW22] Sepehr Assadi and Chen Wang. Sublinear time and space algorithms for correlation clustering via sparse-dense decompositions. In *Proceedings of the 13th Conference on Innovations in Theoretical Computer Science (ITCS)*, volume 215 of *LIPIcs*, pages 10:1–10:20, 2022. 6
- [BBC04] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine learning*, 56(1):89–113, 2004. 1, 5
- [BCMT22] Soheil Behnezhad, Moses Charikar, Weiyun Ma, and Li-Yang Tan. Almost 3-approximate correlation clustering in constant rounds. In *Proceedings of 63rd Annual IEEE Symposium on Foundations of Computer Science, (FOCS)*, pages 720–731, 2022. 6
- [BCMT23] Soheil Behnezhad, Moses Charikar, Weiyun Ma, and Li-Yang Tan. Single-pass streaming algorithms for correlation clustering. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 819–849, 2023. 6
- [BEK21] Mark Bun, Marek Elias, and Janardhan Kulkarni. Differentially private correlation clustering. In *International Conference on Machine Learning (ICML)*, pages 1136–1146, 2021. 6

- [BGU13] Francesco Bonchi, Aristides Gionis, and Antti Ukkonen. Overlapping correlation clustering. *Knowledge and Information Systems*, 35(1):1–32, 2013. [1](#)
- [CCMU21] Mélanie Cambus, Davin Choo, Havu Mäkiönen, and Jara Uitto. Massively parallel correlation clustering in bounded arboricity graphs. In *35th International Symposium on Distributed Computing (DISC)*, volume 209 of *LIPICs*, pages 15:1–15:18, 2021. [6](#)
- [CDK14] Flavio Chierichetti, Nilesch Dalvi, and Ravi Kumar. Correlation clustering in Mapreduce. In *Proceedings of the 20th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 641–650, 2014. [6](#)
- [CDK+21] Vincent Cohen-Addad, Debarati Das, Evangelos Kipouridis, Nikos Parotsidis, and Mikkel Thorup. Fitting distances by tree metrics minimizing the total error within a constant factor. In *Proceedings of 62nd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 468–479, 2021. [6](#)
- [CFL+22] Vincent Cohen-Addad, Chenglin Fan, Silvio Lattanzi, Slobodan Mitrovic, Ashkan Norouzi-Fard, Nikos Parotsidis, and Jakub Tarnawski. Near-optimal correlation clustering with privacy. In *Advances in Neural Information Processing Systems (Neurips)*, 2022. [6](#)
- [CFLM22] Vincent Cohen-Addad, Chenglin Fan, Euiwoong Lee, and Arnaud de Mesmay. Fitting metrics and ultrametrics with minimum disagreements. In *Proceedings of 63rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 301–311, 2022. [6](#)
- [CGW05] Moses Charikar, Venkatesan Guruswami, and Anthony Wirth. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71(3):360–383, 2005. [1](#), [5](#)
- [CKK+06] Shuchi Chawla, Robert Krauthgamer, Ravi Kumar, Yuval Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. *Computational Complexity*, 15(2):94–114, 2006. [5](#)
- [CKP08] Deepayan Chakrabarti, Ravi Kumar, and Kunal Punera. A graph-theoretic approach to webpage segmentation. In *Proceedings of the 17th International conference on World Wide Web (WWW)*, pages 377–386, 2008. [1](#)
- [CLM+21] Vincent Cohen-Addad, Silvio Lattanzi, Slobodan Mitrovic, Ashkan Norouzi-Fard, Nikos Parotsidis, and Jakub Tarnawski. Correlation clustering in constant many parallel rounds. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 2069–2078, 2021. [3](#), [6](#), [9](#), [10](#)
- [CLMP22] Vincent Cohen-Addad, Silvio Lattanzi, Andreas Maggiori, and Nikos Parotsidis. Online and consistent correlation clustering. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 4157–4179, 2022. [6](#)
- [CLN22] Vincent Cohen-Addad, Euiwoong Lee, and Alanthia Newman. Correlation clustering with Sherali-Adams. In *Proceedings of 63rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 651–661, 2022. [1](#), [2](#), [3](#), [4](#), [5](#), [18](#), [19](#), [20](#), [22](#)
- [CMSY15] Shuchi Chawla, Konstantin Makarychev, Tselil Schramm, and Grigory Yaroslavtsev. Near optimal LP rounding algorithm for correlation clustering on complete and complete k -partite graphs. In *Proceedings of the 47th annual ACM Symposium on Theory of Computing (STOC)*, pages 219–228, 2015. [1](#), [2](#), [4](#), [20](#)
- [CSX12] Yudong Chen, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. In *Advances in Neural Information Processing Systems (Neurips)*, pages 2204–2212, 2012. [1](#)

- [DEFI06] Erik D. Demaine, Dotan Emanuel, Amos Fiat, and Nicole Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2-3):172–187, 2006. [5](#)
- [dlVKM07] Wenceslas Fernandez de la Vega and Claire Kenyon-Mathieu. Linear programming relaxations of maxcut. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 53–61, 2007. [3](#)
- [GG06] Ioannis Giotis and Venkatesan Guruswami. Correlation clustering with a fixed number of clusters. *Theory of Computing*, 2:249–266, 2006. [6](#)
- [GLS12] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer Science & Business Media, 2012. [5](#), [7](#)
- [KCMNT08] Dmitri V. Kalashnikov, Zhaoyi Chen, Sharad Mehrotra, and Rabia Nuray-Turan. Web people search via connection analysis. *IEEE Transactions on Knowledge and Data Engineering*, 20(11):1550–1565, 2008. [1](#)
- [KS09] Marek Karpinski and Warren Schudy. Linear time approximation schemes for the Gale-Berlekamp game and related minimization problems. In *Proceedings of the forty-first annual ACM symposium on Theory of computing (STOC)*, pages 313–322, 2009. [6](#)
- [KT02] Jon M. Kleinberg and Éva Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields. *Journal of the ACM*, 49(5):616–639, 2002. [2](#), [4](#)
- [Liu22] Daogao Liu. Better private algorithms for correlation clustering. *CoRR*, arXiv abs/2202.10747, 2022. [6](#)
- [LMV⁺21] Silvio Lattanzi, Benjamin Moseley, Sergei Vassilvitskii, Yuyan Wang, and Rudy Zhou. Robust online correlation clustering. In *Advances in Neural Information Processing Systems (Neurips)*, pages 4688–4698, 2021. [6](#)
- [MSS10] Claire Mathieu, Ocan Sankur, and Warren Schudy. Online correlation clustering. In *Proceedings of 27th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 573–584, 2010. [6](#)
- [PPO⁺15] Xinghao Pan, Dimitris S. Papailiopoulos, Samet Oymak, Benjamin Recht, Kannan Ramchandran, and Michael I. Jordan. Parallel correlation clustering on big graphs. In *Advances in Neural Information Processing Systems (Neurips)*, pages 82–90, 2015. [6](#)
- [RT12] Prasad Raghavendra and Ning Tan. Approximating CSPs with global cardinality constraints using SDP hierarchies. In *Proceedings of the 23d Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 373–387, 2012. [2](#), [3](#), [14](#), [15](#), [19](#)
- [Swa04] Chaitanya Swamy. Correlation clustering: Maximizing agreements via semidefinite programming. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 526–527, 2004. [5](#)
- [Vel22] Nate Veldt. Correlation clustering via strong triadic closure labeling: Fast approximation algorithms and practical lower bounds. In *International Conference on Machine Learning (ICML)*, pages 22060–22083, 2022. [6](#)
- [VGW18] Nate Veldt, David F Gleich, and Anthony Wirth. A correlation clustering framework for community detection. In *Proceedings of the 2018 ACM World Wide Web Conference (WWW)*, pages 439–448, 2018. [6](#)

- [YZ14] Yuichi Yoshida and Yuan Zhou. 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 (ITCS)*, pages 423–438, 2014. [3](#)