Finding the Closest Ultrametric

Marco Di Summa[a,*], David Pritchard[b], Laura Sanità[c]

[a] *Dipartimento di Matematica, Università degli Studi di Padova, Italy*
[b] *Department of Computer Science, Princeton University, USA*
[c] *Department of Combinatorics and Optimization, University of Waterloo, Canada*

Abstract

Ultrametrics model the pairwise distances between living species, where the distance is measured by hereditary time. Reconstructing the tree from the ultrametric distance data is easy, but only if our data is exact. We consider the NP-complete problem of finding the closest ultrametric to noisy data, as modeled by multiplicative or additive total distortion, with or without a monotonicity assumption on the noise.

We obtain approximation ratio $O(\log n)$ for multiplicative distortion where n is the number of species, and $O(1 + (\rho - 1)^{-1})$ for additive distortion where ρ is the minimum ratio of any two distinct input distances. As part of proving our approximation bound for additive distortion, we give the first constant-factor approximation algorithm for a previously-studied problem called Cluster Deletion.

Keywords: ultrametric, approximation algorithm, cluster deletion, phylogenetic reconstruction

---

1. Introduction

Given a set of points V, a *metric* on V is a non-negative symmetric function m satisfying the triangle inequality $m_{ij} \le m_{ik} + m_{kj}$ for every triple i, j, k of points in V, with $m_{ii} = 0$ for each $i \in V$. (Note that we do not require that $m_{ij} = 0$ if and only if $i = j$.) An *ultrametric* u is defined similarly, except that we strengthen the triangle inequality requirement to $u_{ij} \le \max\{u_{ik}, u_{kj}\}$. This latter condition is equivalent to requiring that in every triangle induced by any triple of points, the two longest sides have the same length.

The most natural motivation for ultrametrics comes from biology, specifically from the study of phylogenetic trees. Specifically, the phylogenetic "tree of life" forms an ultrametric on all living species, where the distance between species is the time elapsed since they had a common ancestor (see [1, 2] for more). In particular, one of the most prominent problems in computational biology is reconstructing the evolutionary tree [2], using DNA and RNA distances as an estimate of the evolutionary time separating two species. It is easy to reconstruct the evolutionary tree if the input distances exactly form an ultrametric. Unfortunately, the input data are generally only close to an ultrametric, rather than exactly so, since several factors interfere with molecular dissimilarity as a measure of hereditary time: the randomness of the evolutionary process, sampling particular individuals as representatives of a species, different speeds of gene change in different species, and

*Corresponding author
Email address: `disumma@math.unipd.it` (Marco Di Summa)

horizontal transfer [1] which disrupts the tree-based model. Hence, it makes sense to find an ultrametric that is *closest* to the observed data, which models the true underlying phylogenetic tree.

1.1. Problem description

We consider several variants of the closest ultrametric problem. In each we are given as input a non-negative function d on a finite set of points $V = \{1, \ldots, n\}$, such that d is symmetric ($d_{ij} = d_{ji}$ for all points $i, j \in V$) and $d_{ii} = 0$ for each i. We do *not* assume that d is a metric. The goal is to compute an ultrametric u that is "closest" to d, under a certain measure of dissimilarity.

We focus on two different measures (i.e., objective functions) of the dissimilarity between d and u, both of which have been commonly used in the literature:

- net additive distortion ("ℓ_1"): the dissimilarity of d and u is $\sum_{i<j} |d_{ij} - u_{ij}|$;
- net multiplicative distortion: the dissimilarity of d and u is $\sum_{i<j} \max\{d_{ij}/u_{ij}, u_{ij}/d_{ij}\}$.

In the multiplicative case, we require that the input satisfies $d_{ij} \neq 0$ for each $i \neq j$ in order that the objective is well-defined.

We model the set of feasible output ultrametrics in two ways. We consider:

- the non-contractive model, (or *min-increment* [3, 4]) where we require $u \geq d$;
- the unrestricted model, where the above condition is not required.

For comparison, the *contractive model* requires $u \leq d$. However, this is a solved case because every d has a unique *subdominant ultrametric* u^d which is as close as possible to d. Namely, define u^d_{ij} as the maximum value of x^d_{ij} taken over all ultrametrics x with $x \leq d$. Then it can be shown that u^d is an ultrametric. So in the contractive model, u^d is simultaneously as close as possible in each coordinate to d, and a simple algorithm based on spanning trees can compute u^d efficiently (e.g., see [5, 6]). For this reason, we only consider the non-contractive and the unrestricted models.

From the perspective of biological applications, one motivation for the non-contractive model is the fact that a DNA mutation at some position can mask or cancel out an earlier mutation at the same position, causing the genetic distance to be less than what the evolutionary time would predict.

Throughout the paper, we use $n := |V|$ as the number of points/species, and $\rho := \min\{d_{ij}/d_{k\ell} \mid d_{ij} > d_{k\ell}\}$ as the smallest ratio between two distinct input distances. Furthermore, we assume d to be integer valued: this can be done without loss of generality whenever the input data is rational.

1.2. Our results and techniques

All the problems under consideration cannot be exactly solved by a polynomial time algorithm, assuming $\mathsf{P} \neq \mathsf{NP}$ [7, 6]. Therefore, it makes sense to look for *approximation algorithms*. In particular, an α-approximation algorithm (also said to have *approximation ratio* α) is one that runs in polynomial time and outputs an ultrametric u^{ALG} so that the distortion of d and u^{ALG} is at most a factor α times the optimal distortion between d and the optimal ultrametric u^{OPT}. We also investigate hardness-of-approximation. In this respect, we recall that if a problem is APX-hard, then for some fixed $\varepsilon > 0$, it is NP-hard to compute a $(1 + \varepsilon)$-approximately optimal solution.

We first deal with net multiplicative distortion. We give an $O(\log n)$-approximation algorithm for finding the closest ultrametric, in both the non-contractive and unrestricted models. This improves over the previous known best approximation bound of $O\big(\log \frac{d_{\max}}{d_{\min}} \log n\big)$ given by [8], where $d_{\max}$ and $d_{\min}$ are the maximum and the minimum distance values, respectively.

In conjunction with our algorithmic results, we show that finding the closest ultrametric under net multiplicative distortion in the non-contractive model is APX-hard. In fact, our positive and negative results for this problem establish that the best possible approximation factor is the same as for Multicut, up to constant factors. Our techniques are based on linear programming (LP) relaxations of integer linear programming (ILP) formulations, and we show that the integrality gap for the multiplicative version is $\Omega(\log n)$, which rules out any approximation ratio better than $O(\log n)$ for algorithms based on this relaxation. (The integrality gap is the worst-case ratio between the ILP's optimal value and the optimal value of the LP obtained by removing all integrality constraints.)

For net additive distortion, we give an $O(1 + (\rho - 1)^{-1})$-approximation algorithm for finding the closest ultrametric, in both the non-contractive and unrestricted models. Shamir et al. [9] proved that finding the closest ultrametric to d in the non-contractive model is APX-hard, even if $d_{ij} \in \{0, 1\}$ for every i, j. In fact this restricted problem has quite a natural combinatorial description: given a graph, delete the minimum number of edges so that the remaining graph is a disjoint union of cliques. Shamir et al. [9] call this problem Cluster Deletion; it is called Min-ECP in [10]. As part of proving our approximation bound for net additive distortion, we will give a 4-approximation algorithm for Cluster Deletion (Section 4.3); to the best of our knowledge, no constant-factor approximation algorithm for it was known. We also extend the result on net additive distortion to ℓ_p distortion (Section 4.4).

All of our approximation algorithms have the same framework. First, we take advantage of structural theorems which assert that there is an optimal (or near-optimal) solution where the possible lengths comprising u come from a discrete set. Then, we write an integer linear programming formulation which assigns each edge a value from this set, while respecting the strengthened triangle inequality. The possible lengths give rise to "layers" in the ILP. We first tackle the problem of approximately solving each individual layer. Then we use a simple algorithm to combine all of the layers. Where necessary, we also use scaling and rounding techniques.

1.3. Related work

Several papers deal with finding the closest ultrametric under multiplicative distortion. In both the non-contractive and unrestricted cases, there is a $O\big(\log \frac{d_{\max}}{d_{\min}} \log n\big)$-approximation algorithm given in [8]. For the non-contractive case [11] proves that any metric space embeds into an ultrametric with small "scaling distortion", which is $O(1)$-approximately optimal.

Cavalli-Sforza and Edwards [12] proposed to minimize the ℓ_p-distortion for some $p \geq 1$, where the ℓ_p-distortion is $(\sum_{i,j \in V} |u_{ij} - d_{ij}|^p)^{1/p}$. Note that ℓ_1 distortion is additive distortion. The unrestricted case was proven to be NP-hard for $p = 1$ by Křivánek and Morávek [7], and APX-hard in [8], which extends to any fixed $p \geq 1$. The NP-hardness of the non-contractive version was proven in [6]. On the positive side, $O(\min\{\log n \log\log n, k \log n\}^{1/p})$-approximations are known for both the unrestricted and non-contractive versions of ℓ_p distortion [8, 13] (here k is the number of distinct values in the range of d). For the special case of the additive unconstrained version in which d takes only integer values in $\{0, \ldots, M\}$ (which is of interest because of hierarchical clustering problems), $O(M + 1)$-approximation algorithms are known — a deterministic one in

[13], and a simpler randomized one in [14]. The ℓ_∞ distortion, defined as $\max_{i,j} |u_{ij} - d_{ij}|$, admits polynomial time exact algorithms for both the unrestricted and the non-contractive cases [6, 4].

In developing our algorithms, we will use several graph optimization problems as subroutines. We already mentioned the MULTICUT problem. The best known approximation factor for MULTICUT is $O(\log k)$, where k is the number of terminal pairs [15]; improving this is a long-standing open problem. The lower bounds on its approximability are as follows: it is APX-hard [16]; it is NP-hard to approximate to *any* constant if we assume the Unique Games Conjecture [17]; and it is NP-hard to $\Omega(\sqrt{\log \log n})$-approximate under a slightly stronger assumption [17].

Recall CLUSTER DELETION, mentioned earlier, for which we give the first constant-factor approximation algorithm. A problem related to weighted CLUSTER DELETION that we will utilize is *minimum correlation clustering* (MINCC). We are here given n points, and for each unordered pair $\{i, j\}$ of those points, a weight w_{ij} and a single bit of information: either there is an edge between i and j (positive correlation), or not (negative correlation). Call this graph $G = (V, E)$. The cost of editing G to obtain another graph $H = (V, F)$ is defined as the sum of the weights of the edges which appear in $E \triangle F$. Then the weighted MINCC problem is to find an H with minimum edit cost from G such that H is a disjoint union of cliques. Even if all of the weights are 1, MINCC is APX-hard [18]. In this special setting, a deterministic 4-approximation algorithm based on linear programming was given in [18], and in [19] a very simple randomized 3-approximation algorithm was obtained. If the weights are 0-1, the problem is at least as hard to approximate as MULTICUT [18]. An $O(\log n)$-approximation algorithm is the best known [18, 20, 21, 22] for this case and as well for arbitrary weights. The maximization version of correlation clustering, where the objective is to maximize the weights of the edges not in $E \triangle F$, admits a constant-factor approximation; see [23, 18, 24].

Dessmark et al. [10] give a $O(\log n)$-approximation algorithm for weighted CLUSTER DELETION by reducing it to weighted MINCC. They also give an oracle-2-approximation for unweighted CLUSTER DELETION where the oracle allows exact solution of the NP-hard maximum clique problem.

Ultrametrics are a special case of *tree metrics*. A distance function on a finite set V is a tree metric if there exists a weighted tree, whose vertex set contains V, such that the distance between any two points in V is equal to the length of the unique path connecting them in the tree. There is a rich mathematical literature about tree metric embeddings. There are commonalities to some of the most famous results, such as the Johnson-Lindenstrauss lemma, Bourgain's theorem, and FRT (Fakcharoenphol-Rao-Talwar) trees. Those results deal with a "distortion" equal to maximum multiplicative distortion (usually contractive or non-contractive by convention). Furthermore, they focus on the worst possible distortion over all possible inputs. In contrast, what we seek in the present paper is an as-close-as-possible ultrametric embedding on each instance; the ratio that we are concerned with is an algorithmic approximation ratio measuring how close we can get to this per-instance goal. For example, in the contractive case, even though some d need arbitrarily high distortion in order to become ultrametrics, we consider this problem well-solved (as having an approximation ratio of 1) because we can find exactly the *closest* ultrametric to each input instance. This per-instance approximation question is relatively recent; see Dhamdere's thesis [25] for an early survey of work along this line.

2. LP formulation and layer combination

We now describe a natural linear program for ultrametric approximation, following previous work such as [13]. For all versions of ultrametric approximation, we will fix the possible values of

u in advance. In particular, we will construct a set $\delta_0 < \delta_1 < \cdots < \delta_k$ of values and force u to only take values amongst the δ_i's. We will describe later how to choose the δ_i's; at this stage, let us assume that these values are given.

Let $\binom{V}{2}$ represent the set of all unordered pairs of nodes, i.e., the set of edges, which we write as $\{i, j\}$ or ij. Define a binary variable x_e^t whose value is 1 if and only if $u_e \geq \delta_t$. Then the set of feasible ultrametrics with range in $\{\delta_i\}_i$ can be modeled as follows:

$$x_e^t \geq x_e^{t+1}, \quad \forall e \in \tbinom{V}{2},\ 1 \leq t < k, \tag{1}$$

$$x_{ij}^t + x_{jk}^t \geq x_{ik}^t, \quad \forall i, j, k \in V,\ 1 \leq t \leq k, \tag{2}$$

$$x_e^t \in \{0, 1\}, \quad \forall e \in \tbinom{V}{2}, 1 \leq t \leq k. \tag{3}$$

Inequality (2) ensures that no triangle has a unique maximum length edge. The index t starts at 1 and not 0, since $x_e^0 = 1$ would hold for all $e \in \binom{V}{2}$ given that $u_e \in \{\delta_i\}_i \Rightarrow u_e \geq \delta_0$, and instead we eliminate these constant variables.

The non-contractive case is modeled by adding the constraints

$$x_e^t = 1, \quad \forall e \in \tbinom{V}{2}, t : d_e > \delta_{t-1}. \tag{4}$$

The objective function of the model depends on the choice of the distortion, but in all cases treated here it will be a linear (or more precisely, affine) function. Therefore the resulting model is an integer linear program.

The variables and constraints associated with a particular value of the index t are referred to as *layer* t. More precisely, we think of layer t as consisting of the constraints (2) along with integrality (3) and, when appropriate, non-contractivity (4). Our approach with this ILP is to solve each single layer problem in isolation, and then to combine the layers in such a way that the monotonicity (1) is satisfied. The single layer problems will in general be too hard to solve exactly but it will instead be enough to obtain approximation algorithms for them.

We first show that the following simple method gives a feasible output: *give each edge the longest length assigned to it by any layer.*

Lemma 1 (Layer combination)**.** *Suppose that x satisfies* (2) *and* (3) *but not* (1)*. Define y_e^t to be* 1 *if any x_e^s with $s \geq t$ has $x_e^s = 1$, and* 0 *otherwise. Then y satisfies* (1)*–*(3)*. Moreover, if x satisfied* (4) *then so does y.*

Proof. The only nontrivial part to prove is that y satisfies (2). Since y is 0-1, all we need to show is that if i, j, k, t satisfy $y_{ik}^t = 1$, then $y_{ij}^t + y_{jk}^t \geq 1$. But there must have been some $s \geq t$ such that $x_{ik}^s = 1$; by the definition of y we have $y_{ij}^t + y_{jk}^t \geq x_{ij}^s + x_{jk}^s \geq x_{ik}^s = 1$ as needed, since x satisfied (2). □

In the rest of the paper we will provide analysis and additional techniques to show that the cost of this approach is not too large, for some particular choice of δ_i's. For conciseness, in the rest of the paper, we use Closest-Ultrametric as a general name for the four problems that we study; we attach a superscript $^\times$ or $^+$ depending on whether we consider multiplicative or additive distortion; and we attach a subscript $_\geqslant$ when we deal with the non-contractive case.

3. $O(\log n)$-approximation for multiplicative distortion

A useful fact for the multiplicative problem is the following. It was used, for example, implicitly in [8].

Observation 2. *Let* $\operatorname{dist}(a, b)$ *denote the net multiplicative distortion between two distance functions* a *and* b*, and* $\lambda > 1$*. If* a *and* a' *are such that each pair* a'_{ij}, a_{ij} *differ in ratio by at most a factor of* λ*, then for any* b*,* $\operatorname{dist}(a', b)$ *and* $\operatorname{dist}(a, b)$ *differ in ratio by at most a factor of* λ*.*

Now let d be the original input and obtain d' by rounding each distance up to the next power of 2. It follows from the observation above that an α-approximately optimal ultrametric u^A for d' will be a 8α-approximately optimal ultrametric for d. Indeed, for any ultrametric u (with $u \geq d$ if we are considering the non-contractive case),

$$\operatorname{dist}(d, u^A) \leq 2 \operatorname{dist}(d', u^A) \leq 2\alpha \operatorname{dist}(d', 2u) \leq 8\alpha \operatorname{dist}(d, u),$$

where (i) the first inequality follows from Observation 2; (ii) the second inequality holds because the distance between u^A and d' is at most α times the distance between d' and any ultrametric (as long as, in the non-contractive case, such ultrametric dominates d': this is ensured as in this case $u \geq d$ and therefore $2u \geq d'$); (iii) the third inequality follows by applying Observation 2 first with respect to u and $2u$, and then with respect to d and d'. Moreover, if u^A is a non-contractive ultrametric for d', it is also a non-contractive ultrametric for d. Therefore u^A is 8α-approximately optimal for d. (With little more effort, one proves that u^A is indeed 4α-approximately optimal for d.) Hence, since we seek only an $O(\log n)$ approximation ratio, we will assume without loss of generality that all d_{ij}'s are powers of 2.

*3.1. Non-contractive approximation (*Closest-Ultrametric$^{\times}_{\geqslant}$*)*

For the non-contractive version, the next lemma will provide a way to define the δ_i's values.

Lemma 3. *Let* u *be an ultrametric with* $u \geq d$*. Obtain* u' *by rounding down each* u_{ij} *to the closest value in the range of* d*, i.e.* $u'_{ij} = \max\{d_{k\ell} \mid d_{k\ell} \leq u_{ij}\}$*. Then* u' *is an ultrametric with* $u' \geq d$*, and* $\operatorname{dist}(u', d) \leq \operatorname{dist}(u, d)$*.*

Proof. It is clear that $u' \geq d$ and $\operatorname{dist}(u', d) \leq \operatorname{dist}(u, d)$. Assume by contradiction that u' is not an ultrametric, i.e., there are $i, j, k \in V$ such that $u'_{ij} > \max\{u'_{ik}, u'_{jk}\}$. Then we would also have $u_{ij} > \max\{u_{ik}, u_{jk}\}$, contradicting the fact that u is an ultrametric. □

Corollary 4. *There is an optimal non-contractive ultrametric* u *such that the range of values of* u *is a subset of the range of values of* d*.*

In accordance with Lemma 3, we set the δ_i values equal to the values in the range of d, which we assume to be powers of 2. In other words, $\{\delta_0, \delta_1, \ldots, \delta_k\} = \{d_{ij} \mid i \neq j\}$. We attach the following objective function to the constraints (1)–(4), yielding an integer linear program that models multiplicative error in the non-contractive case:

$$\min \sum_e \left(1 + \sum_{t:\delta_t > d_e} x_e^t \cdot (\delta_t - \delta_{t-1})/d_e \right). \tag{5}$$

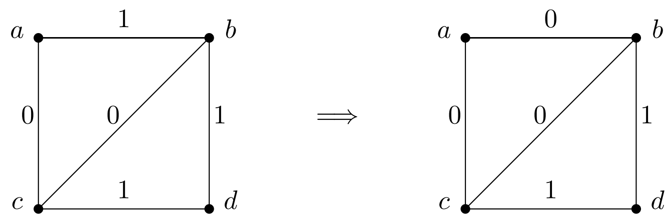

Figure 1: Converting a non-minimal multicut to a minimal one. Let $G = (V, E)$ be the graph shown, where T has only one terminal pair ad. Left: the edges labelled 1 form a multicut, but taking these labels for x^t does not satisfy (7) in triangle a, c, b. Right: replacement by a minimal multicut, which satisfies (7).

There is a constant term in the objective which we will usually ignore, since from the perspective of minimization, adding a non-negative constant cost can only make the approximation ratio of a given algorithm better. We can view the non-constant portion of the objective as the sum of an objective function for each individual layer, where layer t gets the terms that are linear in some x^t variable. Let us temporarily ignore the monotonicity constraint (1) and investigate the problem that each layer t gives rise to. We obtain the following *layer-t-problem:*

$$\min \sum_{e:\delta_t > d_e} x_e^t \cdot (\delta_t - \delta_{t-1})/d_e \tag{6}$$

$$x_{ij}^t + x_{jk}^t \geq x_{ik}^t, \quad \forall i, j, k \in V, \tag{7}$$

$$x_e^t = 1, \quad \forall e : d_e > \delta_{t-1}, \tag{8}$$

$$x_e^t \in \{0, 1\}, \quad \forall e \in \tbinom{V}{2}. \tag{9}$$

There are variables for each $\{i, j\}$; some are fixed to 1 by (8) and the rest can be either 0 or 1. The objective function is a weighted non-negative linear combination of the non-fixed variables. The most important task is to find a natural interpretation for the constraint (7). Recall that weighted Multicut is the following problem: given a weighted graph $G = (V, E)$ and a set T of terminal pairs, delete a minimum-weight set $M \subseteq E$ of edges so that for every $ij \in T$, there is no i-j path in $E \setminus M$.

Lemma 5. *Consider an instance of weighted* Multicut *on graph $G = (V, E)$ where the edge set E is the set of edges corresponding to non-fixed variables, i.e., $\left\{e \in \binom{V}{2} \mid d_e < \delta_t\right\}$, and the set of terminal pairs is $T = \binom{V}{2} \setminus E$. Then the optimal value for the layer-t-problem equals the minimum multicut cost.*

Proof. We claim that if x^t is feasible for the layer-t-problem, then for each ij with $x_{ij}^t = 1$, every i-j path includes at least one edge set to 1. This follows directly from the triangle inequality (7) for paths with exactly two edges. Now assume that a path has $k \geq 3$ edges and denote by $i_0, i_1, \ldots, i_k$ its nodes. If $x_{i_0 i_1}^t = x_{i_1 i_2}^t = 0$, then by (7) also $x_{i_0 i_2}^t = 0$; we can then apply induction on the path $i_0, i_2, i_3, \ldots, i_k$ and conclude that at least one variable $x_{i_j i_{j+1}}^t$ with $j \geq 2$ is set to 1.

Therefore if x^t is feasible for the layer-t-problem, then for each ij with $x_{ij}^t = 1$, every i-j path contains an edge set to 1. This implies that every feasible x^t yields a feasible multicut.

It is false that every multicut is feasible for the ILP; see the left side of Figure 1. However, this is only because of non-minimal multicuts. Given a multicut M, let π be the partition of V induced by the connected components of $E \setminus M$. Then taking only those edges in M which span two different parts of π yields a multicut $M' \subseteq M$. By setting the non-fixed variables equal to the characteristic vector of M', i.e., by setting $x_e^t = 1$ for all $e \in M'$, we obtain a feasible ILP solution, as it is not possible that a triangle has exactly one edge in M'. See the right side of Figure 1 for an illustration. □

An $O(\log n)$-approximation algorithm for Weighted Multicut is known [15]. Thus, we can find an $O(\log n)$-approximate solution to each of the k layer-t-problems. Let us now show that, because all δ_i's are powers of 2, this gives an $O(\log n)$-approximation algorithm for our main goal.

Proposition 6. *For* Closest-Ultrametric$^{\times}_{\geqslant}$, *using an α-approximation algorithm for* Multicut *on each layer, and combining these solutions with the layer combination approach of Lemma 1, we obtain an $O(\alpha)$-approximately closest ultrametric.*

Proof. Let L_t denote the α-approximately optimal solution found for layer t. The sum of the single-layer solutions costs at most α times the cost of an optimal ultrametric. When we combine the layers, although the cost goes up, it is not by much: the cost associated with a single edge goes up from

$$1 + \sum_{t:\delta_t > d_e,\, e \in L_t} \frac{\delta_t - \delta_{t-1}}{d_e} \tag{10}$$

to

$$1 + \sum_{t:\delta_t > d_e,\, t \leq T(e)} \frac{\delta_t - \delta_{t-1}}{d_e} \tag{11}$$

where $T(e) = \max\{t : \delta_t > d_e,\, e \in L_t\}$. Now, writing T instead of $T(e)$, (11) simplifies to δ_T/d_e while the term $(\delta_T - \delta_{T-1})/d_e$ in (10) is at least half as big, since the δ_i's are powers of 2. Consequently, adding up over all edges, combining the layers in this way only costs an extra factor of 2 in the approximation ratio. □

From the above discussion we obtain the following theorem.

Theorem 7. *For net multiplicative distortion, there is an $O(\log n)$-approximation algorithm for finding the closest ultrametric in the non-contractive model.*

3.2. Inapproximability

We now prove that the non-contractive multiplicative case is at least as hard to approximate as Multicut.

Proposition 8. *If there is an α-approximation algorithm for* Closest-Ultrametric$^{\times}_{\geqslant}$, *then there is an $(\alpha + \varepsilon)$-approximation algorithm for* Multicut, *for any $\varepsilon > 0$.*

Proof. Let an instance of Multicut on a n-node graph $H = (V, E)$, with terminals T, be given. We assume that T and E are disjoint, since otherwise this adds a constant term to the objective function, as all edges in $T \cap E$ would have to be cut. We define a Closest-Ultrametric$^{\times}_{\geqslant}$ instance on the set of nodes V by setting

$$d_e = \begin{cases} 1, & e \in E, \\ \lambda, & e \notin E \cup T, \\ \lambda^2, & e \in T. \end{cases}$$

where $\lambda > 1$ is a parameter we will fix later. Using Lemma 3, we restrict our attention to ultrametrics $u \geq d$ with $u_e \in \{\delta_1 = 1, \delta_2 = \lambda, \delta_3 = \lambda^2\}$ for all e. Notice the following important facts:

1. The multiplicative distortion of such an ultrametric u is λ^2 times $|\{e : d_e = \delta_1, u_e = \delta_3\}|$ plus other non-negative terms of value at most λn^2.
2. Writing $M = E \cap \{e : u_e = \delta_3\}$, we have that M is a multicut. Otherwise, if a path in $E \setminus M$ were to connect some terminal pair $\{i, j\}$, then by repeated application of the ultrametric inequality we would have $u_{ij} < \delta_3 = d_e$, contradicting non-contractivity.
3. Conversely, we can transform multicuts to ultrametrics. Given an arbitrary multicut M, define an ultrametric u whose values are δ_2 on the cliques defined by the connected components of $G \setminus M$, and δ_3 on edges spanning two such connected components. The resulting set $E \cap \{e : u_e = \delta_3\}$ is precisely equal to M.

Let M^* be the optimal multicut. This implies that the optimal ultrametric u^* has distortion at most $\lambda^2|M^*| + \lambda n^2$, and the approximation algorithm yields an ultrametric u^A of distortion at most $\alpha\lambda^2|M^*| + \alpha\lambda n^2$. When we convert this back to a multicut M^A, its size $|M^A|$ is at most

$$(\alpha\lambda^2|M^*| + \alpha\lambda n^2)/\lambda^2 = |M^*|(\alpha + \alpha n^2/(\lambda|M^*|)).$$

So assuming that $|M^*| \geq 1$ without loss of generality, and taking λ to be $\varepsilon^{-1}\alpha n^2$, we are done. □

3.3. Integrality gap lower bound

The approximation algorithm for Multicut in [15] is known to provide an *LP-relative* $O(\log n)$-approximation — that is to say, an integral solution of cost at most $O(\log n)$ times the fractional optimum. Using this it is easy to verify that the algorithm in Section 3.1 gives an $O(\log n)$ upper bound on the integrality gap of the LP formulation (1)–(5). We now show a matching lower bound of $\Omega(\log n)$ on its integrality gap. This rules out that any better LP-relative approximation algorithm can be developed using the same LP.

Proposition 9. *The integrality gap of* (1)–(5) *is* $\Omega(\log n)$.

Proof. We use the standard linear programming relaxation for Multicut (see, e.g., [26]). Let the graph be $G = (V, E)$ and let $\mathcal{P}$ denote the set of all paths that connect some terminal pair from T. Then the LP is

$$\min \sum_{e \in E} y_e \tag{12}$$

$$\sum_{e \in P} y_e \geq 1, \quad P \in \mathcal{P}, \tag{13}$$

$$y_e \geq 0, \quad e \in E. \tag{14}$$

The integrality gap of formulation (12)–(14) is known to be $\Theta(\log n)$ (see, e.g., [26]). Denote by $c(x)$ the objective function (5), and by $f(y)$ the objective function (12).

Given an instance of Multicut with $\Omega(\log n)$ integrality gap, construct the same instance of Closest-Ultrametric$^{\times}_{\gtrless}$ as in the proof of Proposition 8, with $\lambda = n^2$. As argued in point 2 of that proof, the integer optimum of (1)–(5) is at least n^4 times the integer multicut optimum. To finish the proof we will show a fractional analogue of the rest of that proof, namely that the fractional optimum of (1)–(5) is at most $2n^4$ times the fractional multicut optimum. This gives the desired result, since with MC short for multicut and CU short for closest ultrametric, we will have

$$\frac{\text{CU-ILP-OPT}}{\text{CU-LP-OPT}} \geq \frac{n^4\text{MC-ILP-OPT}}{2n^4\text{MC-LP-OPT}} = \Omega(\log n).$$

Fix an optimal solution y to the linear program (12)–(14). Let ℓ be the metric completion of y, i.e., ℓ_{ij} is the shortest path length between i and j w.r.t. the y distances. We construct a feasible fractional solution x to (12)–(14) by defining $x_e^1 = x_e^2 = 1$ and $x_e^3 = \min\{\ell_e, 1\}$ for all e. It is easy to check that x is feasible for (1)–(5). Analogous to the proof of Proposition 8, we have that $c(x)$ equals $\lambda^2 f(y)$ plus other non-negative terms of value at most λn^2. As a technicality we may assume $f(y) \geq 1$ since only when $\varnothing$ is a valid multicut could $f(y)$ be less than 1, but such instances have integrality gap 1 and we are considering one with a large integrality gap. So

$$\text{CU-LP-OPT} \leq c(x) \leq \lambda^2 f(y) + \lambda n^2 \leq (\lambda^2 + \lambda n^2) f(y) = 2n^4\text{MC-LP-OPT}. \qquad \square$$

*3.4. Unrestricted approximation (*Closest-Ultrametric$^\times$*)*

We now consider the unrestricted problem Closest-Ultrametric$^\times$. While there is no analogue of Lemma 3, we can get around this by setting the pre-defined values δ_i to consist of all of the powers of 2 between $d_{\min}$ and $d_{\max}$; the number k of such values is polynomial in the input size. Then, using Observation 2, we can restrict our attention to ultrametrics u whose values come from this set of δ_i's. (See [8, Lemma 1(c)] for a similar approach.)

Now, for this problem, the formulation has constraints (1)–(3). The objective function to measure multiplicative distortion in the unrestricted case can be expressed using telescoping sums as

$$\min \sum_e \left(1 + \sum_{t\geq 1:\delta_t \leq d_e} \left(\frac{d_e}{\delta_{t-1}} - \frac{d_e}{\delta_t}\right)(1 - x_e^t) + \sum_{t:\delta_t > d_e} \frac{\delta_t - \delta_{t-1}}{d_e} x_e^t\right). \tag{15}$$

As before, discarding the monotonicity constraint (1) gives k distinct layer-t-problems; the objective for the tth layer consists of that part of (15) corresponding to x^t variables, plus a constant term. The lack of the non-contractivity constraint (4) means there are no fixed edges. For this reason, the layer-t-problem is not Multicut, but rather weighted MinCC. Specifically, the layer-t-problem corresponds to the MinCC instance on graph $(V, E_t := \{e : d_e < \delta_t\})$, where each $e \in E_t$ has weight $c_e^t := (\delta_t - \delta_{t-1})/d_e$ and each $e \notin E_t$ has weight $c_e^t := d_e/\delta_{t-1} - d_e/\delta_t$. The proof is analogous to that of Lemma 5, except that the minimality argument is not needed since all feasible MinCC solutions satisfy (2), unlike all multicut solutions.

Now, using the known $O(\log n)$-approximation algorithms for weighted MinCC [18, 20, 21, 22], we will derive an $O(\log n)$-approximation algorithm for Closest-Ultrametric$^\times$.

Proposition 10. *For* Closest-Ultrametric$^\times$*, using an α-approximation algorithm for* MinCC *on each layer, and combining these solutions with the layer combination approach of Lemma 1, we obtain an $O(\alpha)$-approximately closest ultrametric.*

Proof. Analogous to Proposition 6, the main step is to show that the combination does not increase the cost contributed by any edge too much. Let L_t be the disjoint union of cliques found in the

α-approximate MinCC solution for layer t. For a fixed edge e let $T := T(e) = \max\{t : e \in L_t\}$. Before combining the layers, the cost associated with edge e is

$$1 + \sum_{t \geq 1 : \delta_t \leq d_e,\, e \notin L_t} c_e^t + \sum_{t : \delta_t > d_e,\, e \in L_t} c_e^t.$$

It is instructive to explicitly note that the sequence of costs for e is $\ldots, 4, 2, 1, 1, 2, 4 \ldots$, because of how we set the δ_i's; more specifically, if we let τ be such that $d_e = \delta_\tau$, then we have $c_e^{\tau-k} = 2^k$ for $k \geq 0$, and $c_e^{\tau+1+k} = 2^k$ for $k \geq 0$. Hence, the layer combination step has one of two effects.

- If $T > \tau$, then after combination, we pay a cost of $1+1+2+\cdots+2^k$ for e where $k = T-\tau-1$, whereas before we paid at least 2^k. So the layer combination step costs at most a factor of 2.
- If $T \leq \tau$, then after combination, we pay a cost of $1+1+2+\cdots+2^k$ for e where $k = \tau - T$, whereas before we paid at least this much. So the layer combination step does not cost anything extra. □

This gives us the following theorem.

Theorem 11. *For net multiplicative distortion, there is an $O(\log n)$-approximation algorithm for finding the closest ultrametric in the unrestricted model.*

4. Additive distortion

We are not able to assume that the input d consists of powers of 2 in the additive case. However, there is still a lot of useful structure. In particular, it was observed in [8, Lemma 1(a)] that there is an optimal ultrametric u with all of its values in the range $\left\{d_e \mid e \in \binom{V}{2}\right\}$ of d. To see this, iteratively select any value v in the range of u but not in the range of d, then either increase or decrease all u_e's with value v (at least one of these operations does not increase the additive distortion) until they hit some other value in the range of u or d. Each step decreases the number of distinct values that are in the range of u but not in the range of d, and so the procedure eventually terminates. Hence, in this section we take the layer thresholds δ_i to be equal to the values in the range of d.

*4.1. Unrestricted approximation (*Closest-Ultrametric$^+$*)*

To obtain an ILP formulation for Closest-Ultrametric$^+$, following [13], we attach the objective function

$$\min \sum_e \left(\sum_{t \geq 1 : \delta_t \leq d_e} (\delta_t - \delta_{t-1})(1 - x_e^t) + \sum_{t : \delta_t > d_e} (\delta_t - \delta_{t-1}) x_e^t \right)$$

to (1)–(3). The layer-t-problem is similar to the one for Closest-Ultrametric$^\times$ considered in Section 3.4, except that now c_e^t has the same value for all $e \in \binom{V}{2}$. Therefore the single-layer formulation is an *unweighted* MinCC problem, which admits an $O(1)$-approximation [18, 19]. We proceed to analyze what happens when we combine these layers.

From now on, let $\rho = \min_i \delta_{i+1}/\delta_i$, i.e., the minimum ratio between any two distinct input d values.

Proposition 12. *For* Closest-Ultrametric$^+$*, using an* $O(1)$*-approximation algorithm for cardinality* MinCC *on each layer, and combining these solutions with the layer combination approach of Lemma 1, we obtain an* $O(1 + (\rho - 1)^{-1})$*-approximately closest ultrametric.*

Proof. We follow the approach of Proposition 10. Recall that we defined L_t as the disjoint union of cliques found in the α-approximate MinCC solution for layer t; also, for a fixed edge e, we defined $T := \max\{t : e \in L_t\}$ and τ as the index such that $d_e = \delta_\tau$. The combination step of Lemma 1 does not cost us anything extra for edges with $T \leq \tau$. For an edge with $T > \tau$, the cost paid for it is at most d_T after the combination step, and at least $d_T - d_{T-1}$ before the combination step. In other words the cost increased by a factor of at most

$$\frac{d_T}{d_T - d_{T-1}} = \left(1 - \frac{d_{T-1}}{d_T}\right)^{-1} \leq (1 - \rho^{-1})^{-1} = 1 + (\rho - 1)^{-1}. \qquad \square$$

*4.2. Non-contractive approximation (*Closest-Ultrametric$^+_{\geqslant}$*)*

For the non-contractive version Closest-Ultrametric$^+_{\geqslant}$, the ILP formulation is (1)–(4) with objective function

$$\min \sum_e \sum_{t:\delta_t > d_e} (\delta_t - \delta_{t-1}) x_e^t.$$

Let us now examine the layer-t-problem. Let E denote the set $\{e : d_e < \delta_t\}$. Then the layer-t-problem ILP is equivalent to the following (after dividing the objective function by $\delta_t - \delta_{t-1}$):

$$\min \textstyle\sum_{e \in E} x_e \tag{16}$$
$$x_{ij} + x_{jk} \geq x_{ik}, \quad i, j, k \in V, \tag{17}$$
$$x_e = 1, \quad e \in \tbinom{V}{2} \setminus E, \tag{18}$$
$$x_e \in \{0, 1\}, \quad e \in E. \tag{19}$$

This is an instance of Cluster Deletion: we want to find a minimum-size set of edges in (V, E) whose deletion leaves a disjoint union of cliques. In Section 4.3 we give a 4-approximation algorithm for Cluster Deletion. Assuming this, analogous to Proposition 12, we will have:

Proposition 13. *For* Closest-Ultrametric$^+_{\geqslant}$*, using an* $O(1)$*-approximation algorithm for* Cluster Deletion *on each layer, and combining these solutions with the layer combination approach of Lemma 1, we obtain an* $O(1 + (\rho - 1)^{-1})$*-approximately closest ultrametric.*

4.3. A 4-approximation algorithm for Cluster Deletion

We here consider the Cluster Deletion problem: we are given a graph $G = (V, E)$ and we want to partition the graph into cliques by removing the minimum number of edges. In order to obtain a constant-approximation for Cluster Deletion, we will re-analyze an algorithm developed in [18] for MinCC. The main difference in our setting is that all edges which would be "negative" in the MinCC setting are fixed in the present setting — this is constraint (18).

Proposition 14. *There is a 4-approximation algorithm for* Cluster Deletion*.*

Algorithm 1: 4-approximation for Cluster Deletion on $G = (V, E)$.

$M \leftarrow \varnothing$; let x be an optimal solution to (16)–(18)
$U \leftarrow V$
while $U \neq \varnothing$ **do**
  choose any node $v \in U$; $C \leftarrow \{c \in U \mid c \neq v, x_{vc} < 1/2\}$
  if $\sum_{c \in C} x_{vc} \geq |C|/4$ **then** $S = \{v\}$ **else** $S = C \cup \{v\}$
  $M \leftarrow M \cup \delta(S, U \setminus S)$; $U \leftarrow U \setminus S$
output M

Proof. Our approach is based on the ILP (16)–(19) and its linear relaxation (16)–(18), where all variables are constrained to be non-negative. We use the following standard notation: for a subset of edges A, $x(A)$ means $\sum_{e \in A} x_e$; and for $S \subseteq V$, $\delta(S, T)$ denotes the set of edges in E with one endpoint in S and the other in T. The 4-approximation algorithm is shown in Algorithm 1.

In order to check that the solution is feasible, note that the connected components of the graph surviving after the execution of the algorithm are precisely the sets S chosen by the algorithm. For any such set S, if $i, j \in S$ then $x_{vi} < 1/2$ and $x_{vj} < 1/2$, thus implying $x_{ij} < 1$. Since this is only possible if $ij \in E$, we conclude that S is a clique.

In order to verify the approximation guarantee, it suffices to show that in each iteration of the algorithm, we have the following inequality:

$$|\delta(S, U \setminus S)| \leq 4x(\delta(S, U \setminus S)).$$

In other words, we want to show that the deleted edges in each iteration have average x-value at least 1/4. Since each edge is considered in at most one iteration, the result $|M| \leq 4x(E)$ would follow and consequently so would the approximation guarantee.

First consider the case that $\sum_{c \in C} x_{vc} \geq |C|/4$, where the algorithm selects $S = \{v\}$. If we define $T_1 := \{vc : x_{vc} \geq 1/2\}$, then $|T_1| \leq 2x(T_1)$. The edges in $T_2 := \delta(S, U \setminus S) \setminus T_1$ are exactly the edges vc with $c \in C$, so they satisfy $|T_2| \leq 4x(T_2)$ by assumption. Note that $\delta(S, U \setminus S) = T_1 \cup T_2$, and therefore we have $|\delta(S, U \setminus S)| = |T_1| + |T_2| \leq 4x(\delta(S, U \setminus S))$.

Consider now the other case that $\sum_{c \in C} x_{vc} < |C|/4$, where the algorithm selects $S = C \cup \{v\}$. Let T_1 be the set of edges ij with $i \in S$ and $x_{vj} \geq 3/4$. We have $x_{ij} \geq x_{vj} - x_{vi} \geq 1/4$ by the triangle inequality, so this set of edges satisfies $|T_1| \leq 4x(T_1)$.

The remaining edges in $\delta(S, U \setminus S) \setminus T_1$ can be partitioned into disjoint sets according to their endpoint further from v: for each j with $1/2 \leq x_{vj} < 3/4$, define $U_j := \{i \mid ij \in E, x_{vi} < 1/2\}$. If we can show $\sum_{i \in U_j} x_{ij} \geq |U_j|/4$ for each such j, we will be done.

By the triangle inequality, we have $x_{ij} \geq x_{vj} - x_{vi}$ and therefore for each j,

$$\sum_{i \in U_j} x_{ij} \geq |U_j| x_{vj} - \sum_{i \in U_j} x_{vi} \geq \frac{|U_j|}{2} - \sum_{i \in U_j} x_{vi}.$$

We now claim that $\sum_{i \in U_j} x_{vi} \leq |U_j|/4$ which will complete the proof. We know the related fact that $\sum_{c \in C} x_{vc} < |C|/4$ holds and since $x_{vv} = 0$ we also get $\sum_{i \in S} x_{vi} < |S|/4$. For a node i in S but not V_j, it must be that $ij \notin E$. In such a case the triangle inequality gives $x_{vi} \geq x_{ij} - x_{vj} = 1 - x_{vj} > 1/4$. Since the edges from v to S have average x-value at most 1/4, and the edges from

v to $S \setminus U_j$ have average x-value at least $1/4$, it must be that the edges from v to U_j have average x-value at most $1/4$, as needed. □

4.4. Extension to ℓ_p distortion

Recall that additive distortion is the same as ℓ_1 distortion. Our results extend to ℓ_p distortion as follows. As shown in [8, 13], there is a 2-approximate optimal ultrametric with values in the range of d. Thus we can fix the δ_i's to be the distinct values in the range of d. Following [13], the objective function would be

$$\min \sum_e \left(\sum_{t \geq 1 : \delta_t \leq d_e} ((d_e - \delta_{t-1})^p - (d_e - \delta_t)^p)(1 - x_e^t) + \sum_{t : \delta_t > d_e} ((\delta_t - d_e)^p - (\delta_{t-1} - d_e)^p) x_e^t \right)$$

(this function is the pth power of the ℓ^p distortion). The layer-t-formulation is a weighted Multicut problem in the non-contractive model and a weighted MinCC problem in the unrestricted model. In both cases, there is an $O(\log n)$-approximation for the layer-t-formulation. We now show that by using the layer combination approach of Lemma 1 we get an $O\left(((1 + (\rho - 1)^{-1}) \log n)^{1/p}\right)$-approximation.

Proposition 15. *For ℓ_p distortion, using an α-approximation algorithm for* Multicut *(in the non-contractive case) or* MinCC *(in the unrestricted case) on each layer, and combining these solutions with the layer combination approach of Lemma 1, we obtain an $O((\alpha(1 + (\rho - 1)^{-1}))^{1/p})$-approximately closest ultrametric.*

Proof. The proof idea is as in Propositions 10 and 12 (we use the notation introduced there). In this case, when $T > \tau$, after the layer combination we pay at most $(\delta_T - \delta_\tau)^p$, while before we paid at least $(\delta_T - \delta_\tau)^p - (\delta_{T-1} - \delta_\tau)^p$, thus the ratio is

$$\begin{aligned} \frac{(\delta_T - \delta_\tau)^p}{(\delta_T - \delta_\tau)^p - (\delta_{T-1} - \delta_\tau)^p} &= \left(1 - \left(\frac{\delta_{T-1} - \delta_\tau}{\delta_T - \delta_\tau}\right)^p\right)^{-1} \leq \left(1 - \left(\frac{\delta_{T-1}}{\delta_T}\right)^p\right)^{-1} \\ &\leq \left(1 - \frac{1}{\rho^p}\right)^{-1} \leq \left(1 - \frac{1}{\rho}\right)^{-1} = 1 + (\rho - 1)^{-1}. \end{aligned}$$

Therefore we pay an extra factor of $1 + (\rho - 1)^{-1}$. The claimed factor $O((\alpha(1 + (\rho - 1)^{-1}))^{1/p})$ follows as the objective function that we are considering is the pth power of the ℓ^p distortion. □

The claimed result then follows.

Theorem 16. *For ℓ_p distortion, there is an $O\left(((1 + (\rho - 1)^{-1}) \log n)^{1/p}\right)$-approximation algorithm for finding the closest ultrametric both in the non-contractive and unrestricted models.*

5. Open problems

A first open question is whether there is a better approximation for Closest-Ultrametric$^+$ and/or Closest-Ultrametric$^+_{\gtrless}$. For instance, is there any $O(1)$-approximation or at least polylog(n)-approximation for these problems? This is not known even in the case when d is a metric.

Another interesting point is extension to tree metrics. When dealing with ℓ_1 distance, there is a reduction due to Agarwala et al. [27] from finding closest tree metrics to finding closest ultrametrics, where only a constant factor is lost in the optimal (or approximately optimal) value. Is there such a reduction for multiplicative distortion? Is there a reduction for additive distortion that approximately preserves the parameter ρ? These would extend our work to finding tree metrics instead of ultrametrics.

References

[1] J. P. Gogarten, J. P. Townsend, Horizontal gene transfer, genome innovation and evolution, Nature Reviews Microbiology 3 (2005) 679–687.

[2] J. Felsenstein, Inferring phylogenies, Sinauer Associates, Sunderland, MA, 2004.

[3] B. Ma, L. Wang, L. Zhang, Fitting distances by tree metrics with increment error, Journal of Combinatorial Optimization 3 (1999) 213–225, 10.1023/A:1009837726913.
URL `http://dx.doi.org/10.1023/A:1009837726913`

[4] M. Farach, S. Kannan, T. Warnow, A robust model for finding optimal evolutionary trees, Algorithmica 13 (1995) 155–179.

[5] R. Rammal, J. Angles d'Auriac, B. Doucot, On the degree of ultrametricity, J. Physique Lett. 46 (20) (1985) 945–952. doi:10.1051/jphyslet:019850046020094500.
URL `http://dx.doi.org/10.1051/jphyslet:019850046020094500`

[6] M. Křivánek, The complexity of ultrametric partitions on graphs, Information Processing Letters 27 (1988) 265–270.

[7] M. Křivánek, J. Morávek, NP-hard problems in hierarchical-tree clustering, Acta Informatica 23 (1986) 311–323.

[8] B. Harb, S. Kannan, A. McGregor, Approximating the best-fit tree under ℓ_p norms, in: Proceedings of APPROX and RANDOM, Springer, 2005, pp. 123–133.

[9] R. Shamir, R. Sharan, D. Tsur, Cluster graph modification problems, Discrete Applied Mathematics 144 (12) (2004) 173 – 182. doi:10.1016/j.dam.2004.01.007.
URL `http://www.sciencedirect.com/science/article/pii/S0166218X04001957`

[10] A. Dessmark, J. Jansson, A. Lingas, E. M. Lundell, M. Persson, On the approximability of maximum and minimum edge clique partition problems, International Journal of Foundations of Computer Science 18 (2007) 217–226.

[11] I. Abraham, Y. Bartal, O. Neiman, Embedding metrics into ultrametrics and graphs into spanning trees with constant average distortion, in: Proceedings of the 18th Annual ACM Symposium on Discrete Algorithms (SODA), SIAM, 2007, pp. 502–511.

[12] L. L. Cavalli-Sforza, A. W. F. Edwards, Phylogenetic analysis. models and estimation procedures, American Journal of Human Genetics 19 (1967) 233–257.

[13] N. Ailon, M. Charikar, Fitting tree metrics: Hierarchical clustering and phylogeny, SIAM Journal on Computing 40 (2011) 1275–1291, preliminary version in *Proc. 46th FOCS*, pages 73–82, 2005.

[14] A. Van Zuylen, R. Hegde, K. Jain, D. P. Williamson, Deterministic pivoting algorithms for constrained ranking and clustering problems, in: Proceedings of the 18th annual ACM-SIAM symposium on Discrete Algorithms (SODA 2007), 2007, pp. 405–414.

[15] N. Garg, V. V. Vazirani, M. Yannakakis, Approximate max-flow min-(multi)cut theorems and their applications, SIAM Journal on Computing 25 (1996) 235–251.

[16] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, M. Yannakakis, The complexity of multiterminal cuts, SIAM Journal on Computing 23 (1994) 864–894.

[17] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, D. Sivakumar, On the hardness of approximating multicut and sparsest-cut, Computational Complexity 15 (2006) 94–114.

[18] M. Charikar, V. Guruswami, A. Wirth, Clustering with qualitative information, Journal of Computer and System Sciences 71 (2005) 360–383, preliminary version in *Proc. 44th FOCS*, pages 524–533, 2003.

[19] N. Ailon, M. Charikar, A. Newman, Aggregating inconsistent information: Ranking and clustering, Journal of the ACM 55 (2008) 1–27.

[20] D. Emanuel, A. Fiat, Correlation clustering—minimizing disagreements on arbitrary weighted graphs, in: Proceedings of the 11th European Symposium on Algorithms (ESA 2003), Springer, 2003, pp. 208–220.

[21] E. Demaine, N. Immorlica, Correlation clustering with partial information, in: Proceedings of APPROX and RANDOM, 2003, pp. 71–80.

[22] E. D. Demaine, D. Emanuel, A. Fiat, N. Immorlica, Correlation clustering in general weighted graphs, Theoretical Computer Science 361 (2006) 172–187.

[23] N. Bansal, A. Blum, S. Chawla, Correlation clustering, Machine Learning 56 (2004) 89–113, preliminary version in *Proc. 43rd FOCS*, pages 238–247, 2002.

[24] C. Swamy, Correlation clustering: Maximizing agreements via semidefinite programming, in: Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2004), 2004, pp. 526–527.

[25] K. Dhamdere, Approximation algorithms for metric embedding problems, Ph.D. thesis, Carnegie Mellon University, technical Report CMU-CS-05-152 (2005).

[26] V. V. Vazirani, Approximation Algorithms, Springer, 2001.

[27] R. Agarwala, V. Bafna, M. Farach, B. Narayanan, M. Paterson, M. Thorup, On the approximability of numerical taxonomy, SIAM Journal on Computing 28 (1999) 1073–1085.