Metric Clustering via Consistent Labeling

Robert Krauthgamer
Weizmann Institute and IBM Almaden
robert.krauthgamer@weizmann.ac.il

Tim Roughgarden*
Stanford University
tim@theory.stanford.edu

Abstract

We design approximation algorithms for a number of fundamental optimization problems in metric spaces, namely computing separating and padded decompositions, sparse covers, and metric triangulations. Our work is the first to emphasize *relative guarantees* that compare the produced solution to the optimal one for the input at hand. By contrast, the extensive previous work on these topics has sought *absolute* bounds that hold for every possible metric space (or for a family of metrics). While absolute bounds typically translate to relative ones, our algorithms provide significantly better relative guarantees, using a rather different algorithm.

Our technical approach is to cast a number of metric clustering problems that have been well studied—but almost always as disparate problems—into a common modeling and algorithmic framework, which we call the *consistent labeling* problem. Having identified the common features of all of these problems, we provide a family of linear programming relaxations and simple randomized rounding procedures that achieve provably good approximation guarantees.

1 Introduction

Metric spaces[1] arise naturally in a variety of computational settings, and are commonly used to model diverse data sets such as latencies between nodes in the Internet, dissimilarity between objects such as documents and images, and the cost of traveling between physical locations. Additionally, metric spaces are a useful technical tool, for example when analyzing algorithms based on a linear or semidefinite programming relaxation of Sparsest Cut and other NP-hard problems.

Many useful computational tasks in metric spaces revolve around different types of clustering problems. In these problems, the goal is to produce, for a given metric space (X, d), a collection $\mathcal{S}$ of subsets of X such that, vaguely speaking, nearby points in X tend to appear in the same subset.

This paper makes two broad contributions to the study of algorithms for metric clustering problems. First, we study a number of basic metric clustering problems from an optimization perspective, and design polynomial-time algorithms that provably achieve a near-optimal clustering for every metric space. The large literature on these metric clustering problems has focused exclusively on *absolute* (worst-case) bounds, seeking guarantees that hold for *every possible metric space* (or for every metric in a certain family). By contrast, we emphasize *relative guarantees*, where the objective is to compute a clustering that is close to optimal *for the given input*. Most absolute bounds translate easily to relative ones (in particular, they are efficiently computable), but our algorithms provide significantly better relative guarantees than those implied by the known absolute results. At a high level, our work can be viewed as a parallel to computing an optimal embedding of an input metric space into Euclidean space using semidefinite programming [LLR95], or the recent line of research on computing embeddings with (approximately) minimum distortion, initiated by Kenyon, Rabani, and Sinclair [KRS04]; for a recent account, see Badoiu, Indyk, and Sidiropoulos [BIS07].

Why study relative guarantees? The quest for absolute bounds has obviously been very fruitful, but these bounds may not be very strong for a particular instance at hand, which may admit a much better solution than the worst-possible metric. A popular approach for eluding worst-case absolute bounds is to impose additional structure on the input metric, such as planarity or low-dimensionality, and then prove improved absolute bounds for that restricted class of metrics. But given an arbitrary distance matrix representing, say, latencies in the Internet, it may be highly non-trivial to ascertain whether the corresponding metric is close to one of these families. In contrast, an approximation algorithm guarantees a good solution provided only that one exists. Technically, this requires one to design a "unified" algorithm that works regardless of the precise reason the input admits an improved bound.

An approximation algorithm is also useful for inputs

*Supported in part an NSF CAREER Award CCF-0448664, an Alfred P. Sloan Fellowship, and an ONR Young Investigator Award. This research was performed in part while the author was visiting IBM Almaden.

[1] We call (X, d) a metric space if X is a set of points and $d : X \times X \mapsto R$ is a distance function that is nonnegative, symmetric, and satisfies the triangle inequality.

where the known absolute bounds are non-constructive. In this case, the approximation algorithm recovers, from the existential proof, an efficient algorithm that achieves nearly the same absolute guarantees. In a sense, this is true for planar metrics,[2] where, to date, no algorithm is known to efficiently determine whether an input metric is planar (or close to being planar). Consequently, the decomposition algorithm for planar metrics by Klein, Plotkin and Rao [KPR93] can only be applied if the planar metric is accompanied by a planar graph that realizes the metric. One immediate outcome from our approximation algorithms is that several results that rely on this decomposition, such as the low-distortion embedding into normed spaces of [Rao99, KLMN05], do not require a planar realization of the input metric and hold under the weaker assumption that a planar realization exists (or even that the input metric is close, by means of distortion, to a planar metric).[3]

Moreover, our algorithms are based on linear programming (LP) relaxations, and thus automatically generate a "certificate" of near-optimality (namely, the optimal fractional solution). These simple certificates could possibly be used to prove that a good solution does not exist (e.g., by bounding the optimal fractional solution using duality). Our relative guarantees prove that this lower bound approach is universal, in the sense that a near-optimal certificate always exists.

The second contribution of the paper is to cast a number of metric clustering problems that have been well studied—but almost always as disparate problems—into a common modeling and algorithmic framework, which we call the *Consistent Labeling* problem. At a high level, an instance of Consistent Labeling is described by a set A of objects, a list L_a of allowable labels for each object $a \in A$, and a collection $\mathcal{C}$ of subsets of A. The goal is to assign each object few labels so that subsets are *consistent*, in the sense that the objects of a subset are all assigned a common label. The objects possessing a given label can be viewed as a "cluster" of objects (where clusters can overlap because we allow multiple labels per object), and the consistency constraint for a set $S \in \mathcal{C}$ requires that at least one cluster contains all of the objects of S (i.e. there is at least one label common to all objects in S). In this paper, we show that many metric clustering problems are special cases of different variants of Consistent Labeling. We then provide a family of LP relaxations for all of these problems, and design simple randomized rounding procedures that achieve provably good (relative) approximation guarantees.

[2] We call a metric *planar* if it can be derived from the shortest-path distances in a planar graph with nonnegative edge weights.

[3] A similar argument can be made regarding the absolute guarantees of [CCG+98] for metrics that embed with small distortion into a low-dimensional Euclidean space.

1.1 Metric Decompositions We now detail the optimization problems that we study. Let (X, d) be a finite metric space on $n = |X|$ points. A *cluster* is a subset of the points $S \subseteq X$. The *ball* (in X) of radius $r \geq 0$ centered at $x \in X$ is $B(x, r) = \{y \in X : \ d(x, y) \leq r\}$. The *diameter* of a cluster C is $\mathrm{diam}(C) = \max_{x,y \in C} d(x, y)$, and its *radius* is $\mathrm{rad}(C) = \min_{x_0 \in X} \max_{z \in C} d(x_0, z)$; a point x_0 attaining the radius is called a *center* of C.

Perhaps the simplest genre of metric clustering problems asks for a partition of X into clusters of bounded radius while separating "few" points. We address the two fundamental variants of this notion: computing *separating decompositions* and *padded decompositions*. Both of these are central tools in metric embeddings (e.g., for designing probabilistic embeddings into trees [Bar96, FRT04] and embeddings into ℓ_2 [Rao99, KLMN05], respectively) and are very useful in algorithmic applications. We note that earlier incarnations of these concepts appeared e.g. in [AP90, LR88, LS93, GVY96].

Separating Decomposition. Formally, a *decomposition* of X is a probability distribution μ over partitions of X. A partition *separates* a pair of points if it assigns them to distinct clusters. A partition is Δ-*bounded* if each of its clusters has radius at most Δ, and a decomposition is Δ-*bounded* if every partition in its support is Δ-bounded.[4] A Δ-bounded decomposition μ is called α-*separating* for $\alpha \geq 0$ if for all $x, y \in X$,

$$\mathbf{Pr}_{P \in \mu}[P(x) \neq P(y)] \leq \tfrac{\alpha \cdot d(x,y)}{\Delta} \tag{1.1}$$

or, equivalently,

$$\mathbf{Pr}_{P \in \mu}[P(x) = P(y)] \geq 1 - \tfrac{\alpha \cdot d(x,y)}{\Delta}. \tag{1.2}$$

(Throughout, $P(x)$ denotes the cluster to which the partition P assigns the point x.)

We denote the minimum value $\alpha \geq 0$ satisfying (1.1) by $\alpha^*(X, \Delta)$. Bartal [Bar96] designed an algorithm that for every n-point input metric X, achieves $\alpha = O(\log n)$, and showed that this general (worst-case) bound is tight. Better absolute (in fact, constant) bounds are known for planar metrics [KPR93, Rao99] and other restricted classes of metrics [CCG+98, GKL03, KL06].

Our first result is a 2-approximation algorithm for the problem of computing $\alpha^*(X, \Delta)$ (and constructing a corresponding decomposition). Somewhat surprisingly, this problem was not studied before, and it can be seen that the known decomposition algorithms [Bar96, CCG+98, Rao99, CKR04, FRT04, GKL03, Bar04] are unlikely to lead to an approximation ratio better than $O(\log n)$; thus, obtaining a better relative guarantee requires devising a new algorithmic approach.

[4] Previous literature sometimes uses diameter instead of radius. Obviously the two quantities are within a factor of 2 of each other, and for us the radius is more convenient.

To see how this problem relates to the Consistent Labeling problem, take both the object set and the label set to be the points X. The label set L_x for a point $x \in X$ is defined to be the points in the ball $B(x, \Delta)$. We also impose the restriction that each point receives only one label. We can then interpret the set of vertices with a given label $z \in X$ as a cluster of radius at most Δ (centered at z), and these clusters form a partition of X. There is one consistency constraint for each pair of points; the constraint is satisfied if and only if the points are given the same label (i.e., assigned to the same cluster). The goal is to produce a distribution over feasible labelings such that the worst-case probability of a set being labeled inconsistently (i.e., a pair $x, y \in X$ being separated by the partition), with suitable weighting by $1/d(x, y)$, is minimized.

Remark. Recently, [BK07] found another application of the above approximation algorithm for separating decomposition: a constant-factor approximation algorithm for the problem of computing the least distortion embedding of an input metric into a distribution of dominating ultrametrics. This problem falls into the aforementioned category of computing an embedding with approximately minimum distortion [KRS04, BIS07].

Padded Decomposition. Using the definitions above, a Δ-bounded decomposition μ is (β, q)-*padded* for $\beta, q > 0$ if for all $x \in X$,

$$\mathbf{Pr}_{P\in\mu}[B(x, \Delta/\beta) \subseteq P(x)] \geq q. \tag{1.3}$$

For a given q, we denote the smallest $\beta > 0$ satisfying (1.3) by $\beta^*(X, \Delta, q)$. We can model computing a padded decomposition as a Consistent Labeling problem in the same way as for a separating decomposition, except that now the collection $\mathcal{C}$ of consistency sets is not all pairs of points, but rather all balls of radius Δ/β.

Computing near-optimal padded decompositions appears harder than separating decompositions, but we use a more sophisticated rounding algorithm to compute a Δ-bounded decomposition that is $(2\beta^*, q/12)$-padded, where $\beta^* = \beta^*(X, \Delta, q)$. This bicriteria guarantee is often as useful for applications as a true approximation; in fact, in many applications of padded decomposition, the parameter q is fixed to an arbitrary constant such as $1/2$, and relaxing it to $q/12$ is as good as any other constant.

The problem of computing a near-optimal padded decomposition has not been studied previously, and the absolute guarantees yield, at best, an $O(\log n)$-approximation. Also, while there is a relation between padded and separating decompositions of the form $\alpha^*(X, \Delta) \leq 4\beta^*(X, \Delta/2, 1/2)$ [LN03], in general the two quantities can be very different (e.g., in m-dimensional Euclidean space, $\alpha^* = \Theta(\sqrt{m})$ and $\beta^* = \Theta(m)$ [CCG+98]).

1.2 Covering Problems *Covering problems* form a second genre of metric clustering problems, where the goal is to minimize the overlap between clusters subject to some type of covering constraint. In the interest of space, we describe one example in detail here, and treat the other one in Section 4.

Sparse Cover. Consider an undirected graph $G = (V, E)$ with positive edge lengths and a list $C_1, \ldots, C_p$ of subsets of nodes. We will restrict out discussion to the case $p = n$, which includes the typical case where the subsets correspond to balls around the vertices. A *sparse cover* [AP90] is a Δ-bounded set $\mathcal{S}$ of clusters, such that every subset C_i is contained in some cluster of $\mathcal{S}$. The *degree* of a vertex v in $\mathcal{S}$ is the number of clusters of $\mathcal{S}$ that contain v. Awerbuch and Peleg [AP90] use sparse covers as a building block for a number of distributed network algorithms, including a routing scheme with small stretch and small storage at every node. The maximum degree in the sparse cover determines the nodes' storage requirements, and the stretch of the routing is proportional to $\Delta/\max_i \mathrm{rad}(C_i)$. Awerbuch and Peleg [AP90] give absolute bounds for computing a sparse cover (in the typical case of $p = n$ balls); their bound on the maximum degree grows polynomially in n, specifically as $2kn^{1/k}$ where $k = \Delta/\max_i \mathrm{rad}(C_i)$. This immediately implies a similar relative guarantee of $2kn^{1/k}$.

We study the metric variant of this notion, when G is a complete graph representing a metric space, which is also called the *Nagata dimension* (see [Ass82, LS05]).[5]

We model the problem of computing a sparse cover with minimum maximum degree (where $\Delta > 0$ is given as part of the input) as a Consistent Labeling problem and obtain an $O(\log n)$ approximation. We can further prove (details omitted) that the problem cannot be approximated better than $(\ln n)/3$, unless $\mathrm{NP} \subseteq \mathrm{DTIME}(n^{O(\log\log n)})$.

In the Consistent Labeling formulation, both objects and labels correspond to the vertices V, and a label can only be assigned to an object if they correspond to two vertices at distance at most Δ in G. A Δ-bounded clustering induces a feasible labeling, and the degree of a vertex in the clustering is precisely the number of labels the vertex is assigned. Finally, the constraint of containing a set C_i in at least one cluster naturally translates to a consistency constraint for the subset C_i. It is easy to see that the opposite direction is also true, i.e. a feasible labeling induces a sparse cover. Computing a sparse cover with minimum maximum degree thus translates to computing a feasible labeling that labels

[5] This variant measures all distances in the metric space, analogously to the so-called *weak diameter* variant. In the context a graph with shortest-path distances, the construction of [AP90] satisfies the more stringent *strong diameter* bound.

all the sets consistently while minimizing the number of labels allowed at each object.

Metric Triangulation. We also consider finding metric triangulations of small order [GS95, KSW04]. While this problem is quite different from the Sparse Cover application above, we formulate and approximate both in a common way.

1.3 Overview and Techniques We provide approximation algorithms for the two decomposition problems and the two covering problems mentioned above. A summary of our results appears in Table 1. We proceed with an overview of our techniques.

At a high level, our algorithms follow the well-known paradigm of solving a linear programming relaxation of the problem and applying randomized rounding. We thus start, in Section 2, by formulating LP relaxations for several variants of the Consistent Labeling problem.

Section 3 gives approximation algorithms for the problems of computing separating and padded decompositions. We model them as special cases of a maximization version of the Consistent Labeling problem, where the goal is to maximize the fraction of consistent sets while obeying an upper bound on the number of labels assigned to every object. To round our linear programming relaxations (given in Section 2) in a "coordinated" way that encourages consistently labeled sets, we build on a rounding procedure of Kleinberg and Tardos [KT02]. This procedure was designed for the metric labeling problem with the uniform label-metric (which, in turn, is a modification of the multiway cut algorithm of Calinescu, Karloff, and Rabani [CKR00]). The differences between our intended applications and the metric labeling problem necessitate several extensions to their analysis; for example, we require guarantees for maximization rather than minimization problems, and for general set systems rather than for graphs (i.e., for pairs of points). Our extensions to the Kleinberg-Tardos rounding algorithm and analysis lead, for example, to a 2-approximation algorithm for the separating decomposition problem. The padded decomposition problem is significantly more challenging, and requires us to enhance this basic rounding algorithm in two ways: first, we limit the number of rounding phases to control the proliferation of different labels; second, we add two post-processing steps that first weed out some problematic labels and then expand the residual clusters to ensure the padding properties. Overall, we see that the Kleinberg-Tardos rounding procedure is much more widely applicable than was previously realized.

Section 4 gives a family of approximation algorithms that approximate, in particular, the the sparse cover and metric triangulation problems.

2 Linear Programming Relaxations for Consistent Labeling

Motivated by the breadth of applications in the Introduction, we examine several variants of the Consistent Labeling problem. This section formally defines these variants and gives a family of linear programming relaxations for them. We omit the formal proofs that there LPs are indeed relaxations for the corresponding optimization problems.

In all cases, the input includes a set A of objects, a set L_a of allowable labels for each object a (drawn from a ground set L), and a collection $\mathcal{C}$ of subsets of A. In some applications, we also allow each set $S \in \mathcal{C}$ to have a nonnegative weight w_S. A *feasible labeling* is an assignment of each object a to a subset of L_a. Our two main objectives are to minimize the number of labels assigned to each object, and to maximize the number (or total weight) of sets that are consistently labeled, meaning that a common label is assigned to all of the objects in the set.

The following constraints are common to all our relaxations:

$$1 \leq \sum_{i \in L} x_{ai} \leq k \qquad \text{for every object } a \in A \tag{2.4}$$

$$y_{iS} \leq x_{ai} \qquad \text{for every set } S \in \mathcal{C}, \text{ label } i \in L, \text{ and object } a \in S \tag{2.5}$$

$$z_S \leq \sum_{i \in L} y_{iS} \qquad \text{for every set } S \in \mathcal{C} \tag{2.6}$$

$$z_S \leq 1 \qquad \text{for every set } S \in \mathcal{C} \tag{2.7}$$

$$x_{ai} = 0 \qquad \text{for every object } a \in A \text{ and label } i \notin L_a. \tag{2.8}$$

Constraint (2.4) controls the number of (fractional) labels assigned to each object. In some applications, k will be a decision variable; in others, it will be part of the problem input. The variable y_{iS} and constraint (2.5) encode the extent to which set S is consistently labeled with the label i. The variable z_S and constraints (2.6) and (2.7) encode the extent to which set S is (fractionally) consistently labeled. The fifth constraint enforces the restriction that objects are assigned only to allowed labels.

Maximization Version. In the *MAXIMUM CONSISTENT LABELING (MAX CL)* problem, the objective is to compute a feasible labeling that assigns at most k labels to every object (k is part of the input) and maximizes the total weight of the consistently labeled sets. Our LP relaxation for MAX CL is to simply optimize

$$\max \sum_{S \in \mathcal{C}} w_S z_S \tag{2.9}$$

subject to (2.4)–(2.8).

Problem	Approximation factor		Absolute guarantee
Separating Decomposition	2	[Theorem 3.4]	$O(\log n)$ [Bar96]
Padded Decomposition	$O(1)$ bicriteria	[Theorem 3.8]	$O(\log n)$ [Bar96]
Sparse Cover (stretch k)	$O(\log n)$	[Corollary 4.2]	$2kn^{1/k}$ [AP90]
(ε, ρ)-Triangulation	$O(\ln \frac{1}{\varepsilon})$ bicriteria	[Corollary 4.4]	n (trivial)

Table 1: Our approximation factors and those implied by previous work on absolute bounds.

Padded and separating decompositions motivate the *MAXIMUM FAIR CONSISTENT LABELING (MAX FAIR CL)* problem, where given an input as in MAX CL, the goal is to compute a distribution over feasible labelings that assign at most k labels to every object (with probability 1) and maximizes the minimum weighted probability (over $S \in \mathcal{C}$) that a set S is labeled consistently. Computing both separating and padded decompositions are special cases of MAX FAIR CL with $k = 1$, where the sets correspond to pairs of points, and to balls of radius Δ/β around each point in the given metric space, respectively. Our LP relaxation for this problem maximizes a decision variable α subject to (2.4)–(2.8) and

$$w_S z_S \geq \alpha \qquad \text{for every set } S \in \mathcal{C}. \tag{2.10}$$

Minimization Version. In the minimization version of consistent labeling, we constrain (from below) the fraction of consistently labeled sets and seek a labeling that uses as few labels per object as possible. (We could also include set weights, but our applications do not require them.) In the *complete* special case, we demand that all sets are consistently labeled. Formally, the *MINIMUM COMPLETE CONSISTENT LABELING (MIN CCL)* problem is, given the usual data, to compute a feasible labeling that consistently labels all sets and minimizes the maximum number of labels assigned to an object.

As noted in the Introduction, computing a sparse cover of a network is a special case of MIN CCL. In our LP relaxation for MIN CCL, we minimize the decision variable k subject to (2.4)–(2.8) and the additional constraint that (2.7) holds with equality for every set $S \in \mathcal{C}$. Several extensions to the MIN CCL problem are easily accommodated; we use Network Triangulation as a case study in Section 4.

3 Maximum Consistent Labeling

This section gives a generic approximation algorithm for the MAX CL and MAX FAIR CL problems. We then refine the algorithm and its analysis to give an approximation algorithm for computing a separating decomposition (Theorem 3.4). Subsequently, we enhance the algorithm to handle the more difficult task of approximating an optimal padded decomposition (Theorem 3.8).

Approximation Algorithm for MAX CL and MAX FAIR CL. We first give a $\Theta(1/f_{\max})$-approximation algorithm for weighted MAX CL and MAX FAIR CL, where $f_{\max} = \max_{S \in \mathcal{C}} |S|$ denotes the largest cardinality of a set of $\mathcal{C}$. We build on a rounding procedure that was designed by Kleinberg and Tardos [KT02] for the metric labeling problem with uniform metric, even though our context is quite different. First, we wish to maximize the consistency probability, as in (1.2), rather than minimize the probability for inconsistency, as in (1.1). Second, an object may get multiple labels (k) rather than one label ($k = 1$). Third, the notion of consistency is not as simple, as it involves a subset S (whose size may be bigger than 2) and each object in S has k labels (where k may be bigger than 1). Fourth, we may want to produce a distribution (in MAX FAIR CL) rather than only one solution. It is thus a pleasant surprise that the algorithm in [KT02] lends itself to our setting; in fact, our algorithm can be easily seen to generalize theirs from $k = 1$ labels to general k.

Our approximation algorithm is shown in Figure 1. After solving the appropriate LP relaxation, the rounding algorithm is the same for both problems: we repeatedly choose a label $i \in L$ and a threshold $t \in [0, 1]$ independently and uniformly at random, and for all objects a with x^*_{ai} larger than the threshold t, we add i to the set of labels assigned to a. (If i is already assigned to a, then this assignment is redundant.) The algorithm terminates when every object has been assigned a label in at least k iterations (not necessarily distinct labels). To respect the constraint on the number of labels, each object retains only the first k labels that it was assigned. This final step, together with the LP constraint (2.8), ensures that the output of the algorithm is a feasible labeling.

Our analysis hinges on the following lemma (proved in the full version). This lemma lower bounds the probability that a set is consistently labeled by our rounding algorithm. We will also use the lemma in Section 4 for minimization versions of Consistent Labeling.

LEMMA 3.1. *Let* (x^*, y^*, z^*) *be a feasible solution*

Input: an instance of MAX CL or MAX FAIR CL.

1. Solve the appropriate LP relaxation: for MAX CL, maximize (2.9) subject to (2.4)–(2.8); for MAX FAIR CL, maximize α subject to (2.4)–(2.8) and (2.10). Let (x^*, y^*, z^*) denote the optimal LP solution.
2. Repeat until every object has been assigned at least k labels (counting multiplicities):
3. Choose a label $i \in L$ and a threshold $t \in [0, 1]$ uniformly at random.
4. For each object $a \in X$, if $x^*_{ai} > t$, then add i to the set of labels assigned to a.
5. For each object, retain the first k labels received.

Figure 1: The MAX CL and MAX FAIR CL algorithms.

to (2.4)–(2.8)*. For every set* $S \in \mathcal{C}$*,*

$$\mathbf{Pr}\left[S \text{ consistently labeled}\right] \quad \geq \quad 1 - \left(1 - \frac{z^*_S}{k|S|}\right)^k.$$

Using this lemma, linearity of expectation, and the (crude) inequality $(1 - \frac{a}{k})^k \leq e^{-a} \leq 1 - a + \frac{a^2}{2} \leq 1 - \frac{a}{2}$ for $a \in (0, 1)$, we obtain the approximation bounds for the MAX CL and MAX FAIR CL problems.

THEOREM 3.2. *There are randomized polynomial-time* $(1/2f_{\max})$*-approximation algorithms for weighted MAX CL and MAX FAIR CL.*

The bound $1/2f_{\max}$ in Theorem 3.2 can be sharpened; for example, it is $1/f_{\max}$ when $k = 1$.

Theorem 3.2 does not immediately give a useful approximation algorithm for computing separating or padded decompositions; we next give the necessary refinements.

Separating Decomposition. Theorem 3.2 gives an approximation guarantee for the maximum consistency probability (as in (1.2)), rather than for the minimum inconsistency probability (as in (1.1)). These two objectives are equivalent for exact optimization, but not for approximation. We now show how to modify our LP relaxation and analysis for MAX FAIR CL (but using the same rounding algorithm), to obtain an $f_{\max}$-approximation for the latter objective. Choosing the weight w_S of a set $S = \{x, y\}$ to be $\Delta/d(x, y)$, we will immediately get a 2-approximation algorithm for computing an optimal separating decomposition, which matches the integrality gap for our LP relaxation (as we show in the full version).

We address the problem of minimizing the inconsistency probability using LP (3.11) below. This LP differs from the one used for MAX FAIR CL in that we fix $k = 1$, and that now y_{iS} and z_S measure the fractional inconsistency (rather than consistency) of a set S; we then bound these from above (rather than from below) using α.

$$\begin{array}{lll} \text{Min} & \alpha & \\ \text{s.t.} & \sum_{i \in L} x_{ai} = 1 & \forall a \in A \\ & y_{iS} \geq x_{ai} - x_{a'i} & \forall S \in \mathcal{C};\ a, a' \in S \\ & z_S \geq \frac{1}{|S|} \sum_{i \in L} y_{iS} & \forall S \in \mathcal{C} \\ & x_{ai} = 0 & \forall a \in A;\ i \notin L_a \\ & \alpha \geq w_S z_S & \forall S \in \mathcal{C}. \end{array} \tag{3.11}$$

It is straightforward to verify that this LP is indeed a relaxation for the problem of minimizing inconsistencies (in fact, for $|S| = 2$ the LP is essentially equivalent to the previous one); as such, it is also a relaxation for the problem of computing an optimal separating decomposition with objective (1.1). Let $(x^*, y^*, z^*, \alpha^*)$ be the optimal fractional solution; then α^* is a lower bound on the value of an optimal solution. We shall now apply the rounding algorithm of Figure 1. The following lemma is similar to Lemma 3.1 (and also to the analysis in [KT02]).

LEMMA 3.3. *For every set* $S \in \mathcal{C}$*,*

$$\mathbf{Pr}\left[S \text{ is not consistently labeled}\right] \leq |S| z^*_S.$$

This inequality immediately implies an $f_{\max}$-approximation for minimizing the (weighted) inconsistency probability of all sets. In particular, we obtain the following theorem.

THEOREM 3.4. *There is a randomized polynomial-time* 2*-approximation algorithm for computing a separating decomposition.*

Padded Decomposition. Building on these techniques, we now design an algorithm for padded decomposition; the precise statement of the guarantees appears in Theorem 3.8. Recall that the input is a metric space (X, d) and a parameter $q > 0$. We shall use the LP formulation below, motivated by modeling our problem as a MAX FAIR CL as follows: objects correspond to points in X, labels represent cluster centers (generally all of X), and the consistency sets $\mathcal{C}$ corresponds to all balls of radius Δ/β. The LP has nonnegative variables x_{ij}, which represent an assignment of point $j \in X$ to a

cluster centered at $i \in X$ (i.e. labeling an object), and variables y_{ij}, which represent the consistency of the ball around j with respect to the cluster centered at i (i.e. consistency of a set).

$$\begin{array}{ll} \sum_{i\in X} x_{ij} \le 1 & \forall j \in X \\ y_{ij} \le x_{ij'} & \forall i, j \in X;\ j' \in B(j, \Delta/\beta) \\ x_{ij} = 0 & \forall j \in X;\ i \in X \setminus B(j, \Delta) \\ \sum_{i\in X} y_{ij} \ge q & \forall j \in X \end{array} \tag{3.12}$$

We omit the proof that this is a valid relaxation.

LEMMA 3.5. *LP (3.12) is a relaxation of padded decomposition, namely, it is feasible whenever $\beta \le \beta^*(X, \Delta, q)$.*

The algorithm's first step then is to find the smallest $\beta > 0$ such that the LP above is feasible, which can be done e.g. by binary search over the at most $\binom{n}{2}$ distance values appearing in the input metric. (Note that β is not a variable of the LP.)

The rounding procedure for LP (3.12) has three steps. First, use a procedure similar to the Kleinberg-Tardos rounding [KT02], except that exactly n assignment rounds are performed to obtain a collection of disjoint clusters $\{C_i : i \in X\}$ (this is not a partition, since some points might not be assigned at all). Next, check for which points $j \in X$ the ball $B(j, \Delta/\beta)$ meets more than one cluster C_i, and remove all these points (simultaneously) from the clustering. Finally, expand each of the (non-empty) clusters at hand to its $\Delta/2\beta$-neighborhood, and output the partition induced by these clusters (where a point that belongs to no cluster forms a singleton cluster). A formal description of this procedure is given in Figure 2.

LEMMA 3.6. *The algorithm in Figure 2 always outputs a Δ-bounded partition of X.*

We omit the straightforward proof of Lemma 3.6.

LEMMA 3.7. *Denote by μ the decomposition produced by the algorithm in Figure 2. Then for every $j \in X$,*

$$\mathbf{Pr}_{P\in\mu}[B(j, \Delta/2\beta) \subseteq P(j)] \ge q/12.$$

Proof. Fix a point $j \in X$. It suffices to show that $\mathbf{Pr}[j \in \cup_{i\in X} C'_i] \ge q/12$, since once $j \in C'_i$, the entire ball $B(j, \Delta/2\beta)$ will end up inside the cluster C''_i. The event $j \in \cup_{i\in X} C'_i$ is the disjoint union of the events $j \in C'_{i^*}$ over all $i^* \in X$. Thus,

$$\mathbf{Pr}[j \in \cup_{i\in X} C'_i] = \sum_{i^*\in X} \mathbf{Pr}[j \in C_{i^*}] \tag{3.13}$$

We next examine the n iterations over steps 4–5, and refine our earlier analysis of the randomized assignment procedure. Fix $i^* \in F$. For the event $j \in C'_{i^*}$ to occur, we should have $j \in C_{i^*}$ and also $j \notin D^*$, the latter meaning that $B(j, \Delta/\beta)$ is disjoint of $\cup_{i\ne i^*} C_i$. For the purpose of a lower bound on $\mathbf{Pr}[j \in C_{i^*}]$, it suffices to consider the case that $i^* \in X$ is chosen in exactly one of these n iterations, which happens with probability $\binom{n}{1}\frac{1}{n}(1-\frac{1}{n})^{n-1} \ge \frac{1}{e}$. In the iteration i^* is chosen, we need the random threshold T to be smaller than y_{i^*j}, which occurs with probability y_{i^*j}. In each of the remaining $n-1$ iterations, we need the chosen center $i \ne i^*$ to capture no point in $B(j, \Delta/\beta)$, which happens with probability $1-\max\{y_{ij'} : j' \in B(j, \Delta/\beta)\} \ge 1-x_{ij}$ (using the second LP constraint). Recalling that $i \ne i^*$ is chosen uniformly at random, we obtain (for a fixed i^*)

$$\mathbf{Pr}[j \in C_{i^*}] \ge \frac{1}{e} \cdot y_{i^*j} \cdot \Big(\sum_{i\ne i^*} \frac{1-x_{ij}}{n-1}\Big)^{n-1}. \tag{3.14}$$

Finally, using the first LP constraint, we know that $\sum_{i\ne i^*} x_{ij} \le 1$. Combining this inequality with (3.13) and (3.14), and using the last constraint of the LP, we obtain (assuming $n \ge 3$)

$$\begin{aligned} \mathbf{Pr}[j \in \cup_{i\in X} C'_i] &\ge \sum_{i^*\in X} \left(\frac{1}{e} \cdot y_{i^*j} \cdot \Big(1 - \frac{1}{n-1}\Big)^{n-1}\right) \\ &\ge \sum_{i^*\in X} \left(\frac{1}{e} \cdot y_{i^*j} \cdot \frac{1}{4}\right) \ge \frac{q}{12}. \end{aligned}$$

■

The lemmas above immediately yield the following.

THEOREM 3.8. *There is a randomized polynomial-time algorithm that, given (X, d), Δ, and q, produces a Δ-bounded $(\beta', q/12)$-padded decomposition, where $\beta' \le 2\beta^*(X, \Delta, q)$.*

4 Minimum Consistent Labeling

This section gives two approximation algorithms for the minimization version of Consistent Labeling, where the goal is to consistently label a prescribed fraction of the sets while using as few labels as possible. The first algorithm is tailored to the MIN CCL problem, where all of the sets must be consistently labeled. Our algorithm achieves an $O(\log(n + |\mathcal{C}|))$-approximation for the general problem (see Theorem 4.1). Applying this result to the case of Sparse Cover in a distributed network (where $|\mathcal{C}| = n$), we immediately obtain an $O(\log n)$ approximation (see Corollary 4.2). Our second algorithm computes, for given $0 < \varepsilon < 1/4$, a solution that consistently labels a $(1 - 3\varepsilon)$-fraction of the sets using at most $O(\ln \frac{1}{\varepsilon})$ times more labels per object than the minimum necessary to consistently label at least an $(1 - \varepsilon)$-fraction of the sets. (The constant 3 is

Input: an instance of padded decomposition
1. Find the smallest $\beta > 0$ such that LP (3.12) is feasible
2. Initialize a cluster $C_i = \emptyset$ for every $i \in X$.
3. Repeat n times
4. Choose uniformly at random $i \in X$ and threshold $T \in [0, 1]$
5. Add to cluster C_i every yet unclustered point $j \in X$ for which $T < y_{ij}$
6. Let $D^* = \{j \in X : B(j, \Delta/\beta)$ meets more than one cluster $C_i\}$
7. For every $i \in X$, let $C'_i = C_i \setminus D^*$
8. For every $i \in X$, let $C''_i = \{j \in X : d(j, C'_i) \leq \Delta/2\beta\}$
9. Output the partition induced by $\{C''_i : i \in X\}$ (i.e. adding singletons when necessary)

Figure 2: The Padded Decomposition algorithm.

quite arbitrary, and we make no no attempt to optimize it.) This bicriteria guarantee is particularly appropriate for the Network Triangulation application, where one typically permits a small fraction of pairs of points to have inaccurate distance estimates.

Complete Consistent Labeling and Sparse Cover. Our approximation algorithm for MIN CCL is shown in Figure 3. The only difference between this algorithm and that for MAX CL and MAX FAIR CCL is the stopping condition: instead of explicitly controlling the number of labels assigned to each object, we stop once we have obtained a feasible solution (i.e., every set is consistently labeled).

THEOREM 4.1. *The algorithm for MIN CCL in Figure 3 computes, with high probability, an $O(\log(n + |\mathcal{C}|))$-approximation in polynomial time.*

Proof. Let (x^*, y^*, z^*, k^*) denote the optimal LP solution and consider a set $S \in \mathcal{C}$. Let $x_i^{\min}$ denote $\min_{a \in S} x^*_{ai}$. The probability that S is consistently labeled in a given iteration equals

$$\frac{1}{n}\sum_{i \in L} x_i^{\min} \geq \frac{1}{n}\sum_{i \in L} y^*_{iS} \geq \frac{z^*_S}{n} = \frac{1}{n},$$

with the inequalities following from the LP constraints.

On the other hand, constraint (2.4) ensures that the probability that an object $a \in A$ receives a label in a given iteration is

$$\frac{1}{n}\sum_{i \in L} x^*_{ai} \leq \frac{k^*}{n}.$$

Applying Chernoff bounds, it follows that with high probability, the algorithm of Figure 3 terminates in $O(n \log |\mathcal{C}|)$ iterations with each object receiving $O(\log n + k^* \log |\mathcal{C}|)$ labels. ∎

Modeling the Sparse Cover problem as MIN CCL, as explained in the Introduction, the following corollary is immediate.

COROLLARY 4.2. *There is a randomized polynomial-time algorithm that, given an instance of Sparse Cover with n subsets of vertices, outputs a feasible cover whose maximum degree is, with high probability, at most $O(\log n)$ times that of optimal.*

Bicriteria Guarantee. For a consistent labeling instance and $\alpha \in (0, 1)$, let $k_{OPT}(\alpha)$ be the smallest $k > 0$ for which there is a feasible labeling that assigns as most k labels per object and is consistent for an α fraction of the sets. The following theorem achieves a bicriteria guarantee that is often reasonable for α close to 1, but other tradeoffs are also possible.

THEOREM 4.3. *There is a randomized polynomial time algorithm that, given a consistent labeling instance and $\varepsilon > 0$, computes with high probability a labeling that uses at most $O(\ln \frac{1}{\varepsilon}) \cdot k_{OPT}(1 - \varepsilon)$ labels per object and is consistent for a $(1 - 3\varepsilon)$ fraction of the sets.*

Proof. We use our algorithm for MAX CL and MAX FAIR CL (Figure 1). More precisely, modify step 1 and solve the LP relaxation which minimizes k subject to the constraints (2.4)–(2.8) and to $\sum_S z_S \geq (1 - \varepsilon)|\mathcal{C}|$. After solving the LP relaxation, proceed as in Figure 1 but with the threshold k replaced by $\ell = 16k^* \ln \frac{1}{\varepsilon}$, where k^* is the optimal LP value. We can assume without loss of generality that $\varepsilon < 1/4$.

It is easy to verify that this LP is indeed a relaxation. Hence, in the LP solution $k^* \leq k_{OPT}(1 - \varepsilon)$, and by construction, the algorithm always uses at most $\ell = O(\ln \frac{1}{\varepsilon}) \cdot k_{OPT}(1 - \varepsilon)$ labels per object. Now, call a set S *good* if $z^*_S \geq \frac{1}{4}$ in the optimal LP solution. At least $(1 - 2\varepsilon)|\mathcal{C}|$ sets are good, or otherwise $\sum_S z^*_S < (1 - 2\varepsilon)|\mathcal{C}| \cdot 1 + (2\varepsilon)|\mathcal{C}| \cdot \frac{1}{4} < (1 - \varepsilon)|\mathcal{C}|$, contradicting the last LP constraint. For every good set S, by a calculation similar to Lemma 3.1, the probability that S will *not* be consistently labeled is at most

$$\left(1 - \frac{z^*_S}{k^*|S|}\right)^{\ell} \leq \left(1 - \frac{1}{8k^*}\right)^{\ell} \leq e^{-\ell/8k^*} \leq \varepsilon^2.$$

Input: an instance of MIN CCL.

1. Minimize k subject to constraints (2.4)–(2.8) and in addition that (2.7) holds with equality for every set $S \in \mathcal{C}$. Let (x^*, y^*, z^*, k^*) denote the optimal LP solution.
2. Repeat until every set is consistently labeled:
3. Choose a label $i \in L$ and a threshold $t \in [0, 1]$ uniformly at random.
4. For each object $a \in X$, if $x^*_{ai} > t$, then add i to the set of labels assigned to a.

Figure 3: The MIN CCL algorithm.

Thus, among the good sets, the expected fraction of not consistently labeled sets is at most ε^2, and by Markov's inequality, the probability that this fraction exceeds ε is at most ε. In other words, with probability at least $1 - \varepsilon \geq 3/4$, the number of consistently labeled sets is at least $(1 - \varepsilon) \cdot (1 - 2\varepsilon)|\mathcal{C}| \geq (1 - 3\varepsilon)|\mathcal{C}|$. Naturally, we can amplify the success probability by independent repetitions. ■

Metric Triangulation. Network triangulation is a heuristic for estimating distances in a network, initially suggested by Guyton and Schwartz [GS95]. Motivated by the practical success of this heuristic, Kleinberg, Slivkins, and Wexler [KSW04] initiated a theoretical study of triangulation in metric spaces, formally defined as follows. A *triangulation* of a metric (X, d) assigns for every $x \in X$ a collection of *beacons* $S_x \subseteq X$. The triangulation has *order* k if $\max\{|S_x| : x \in X\} \leq k$. We are interested in low-order triangulations in which the distance between every $x, y \in X$ can be estimated from their distances to $S_x \cap S_y$ using the triangle inequality. Formally, define

$$D^+(x, y) = \min_{b \in S_x \cap S_y} [d(x, b) + d(b, y)]$$
$$D^-(x, y) = \max_{b \in S_x \cap S_y} |d(x, b) - d(b, y)|.$$

The triangulation is called an (ε, ρ)-*triangulation* (for $0 \leq \varepsilon \leq 1$ and $\rho \geq 1$) if for all but an ε fraction of the pairs $x, y \in X$ we have $D^+(x, y) \leq \rho \cdot d(x, y)$ and $D^-(x, y) \geq d(x, y)/\rho$. Let $k_{OPT}(X, \varepsilon, \rho)$ denote the smallest $k > 0$ such that (X, d) admits an (ε, ρ)-triangulation of order k.

The problem of computing a near-optimal metric triangulation, i.e. computing $k_{OPT}(X, \varepsilon, \rho)$, has not been studied before, although several absolute guarantees are known. In [KSW04], it is shown that doubling metrics admit an (ε, ρ)-triangulation whose order is upper bounded independently of n (namely, the bound depends only on ε, ρ and the doubling constant), and additional bounds are proved in [Sli05b, Sli05a]. However, in some metrics triangulation requires a very high order (e.g. $\Omega(n)$ in uniform metrics and $n^{\Omega(1)}$ in tree metrics [Kra07], for fixed ε, ρ), and thus absolute bounds cannot yield any nontrivial approximation ratio.

Modeling this Metric Triangulation problem as a consistent labeling problem and using Theorem 4.3, we give for it below a bicriteria approximation algorithm. We will actually require a slight generalization of the consistent labeling problem (and our algorithm), as follows: For every set $S \in \mathcal{C}$, there may be (in the input) a restricted set L_S of labels that can be used to consistently label S. Accommodating this generalization in our algorithms (Theorems 4.1 and 4.3) requires only changing the index set on the right-hand side of (2.6) from "$i \in L$" to $i \in L_S$"; no other changes to our algorithm or their proofs are required.

COROLLARY 4.4. *There is a randomized polynomial-time algorithm that, given a metric triangulation instance (including ρ and ε), outputs a $(1 - 3\varepsilon, \rho)$-triangulation of order $O(\ln \frac{1}{\varepsilon}) \cdot k_{OPT}(X, \varepsilon, \rho)$.*

Proof. We first model the metric triangulation problem as a slight generalization of MIN CCL. Objects correspond to the points X and labels correspond to beacons (generally all of X). For every pair of nodes x, y we want a consistency constraint that reflects our desire that x, y have at least one beacon in $S_x \cap S_y$ attaining $D^+(x, y)$, and similarly at least one common beacon attaining $D^-(x, y)$. We model this by using set-dependent allowable labels L_S, and furthermore replacing the set of constraints (2.6) by two sets of constraints, one with allowable label set $L^+(x, y) = \{b \in X : d(x, b) + d(b, y) \leq \rho \cdot d(x, y)\}$ and one with allowable label set $L^-(x, y) = \{b \in X : |d(x, b) - d(b, y)| \geq d(x, y)/\rho\}$; the extra set of constraints only increases the hidden constants in our analysis. The correspondence between this variant of consistent labeling and network triangulation is immediate, and we can thus use our algorithm from Theorem 4.3 to obtain a bicriteria bound for Metric Triangulation. ■

Acknowledgments We thank Anupam Gupta and James Lee for preliminary discussions about the various concepts used in the paper.

References

[AP90] B. Awerbuch and D. Peleg. Sparse partitions. In *31st Annual IEEE Symposium on Foundations of Computer Science*, pages 503–513, 1990.

[Ass82] P. Assouad. Sur la distance de nagata. *C. R. Acad. Sci. Paris Ser. I Math.*, 1(294):31–34, 1982.

[Bar96] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. In *37th Annual Symposium on Foundations of Computer Science*, pages 184–193. IEEE, 1996.

[Bar04] Y. Bartal. Graph decomposition lemmas and their role in metric embedding methods. In *12th Annual European Symposium*, volume 3221 of *Lecture Notes in Computer Science*, pages 89–97. Springer, 2004.

[BIS07] M. Badoiu, P. Indyk, and A. Sidiropoulos. Approximation algorithms for embedding general metrics into trees. In *18th Symposium on Discrete Algorithms*, 2007.

[BK07] Y. Bartal and R. Krauthgamer. Private communication, 2007.

[CCG+98] M. Charikar, C. Chekuri, A. Goel, S. Guha, and S. Plotkin. Approximating a finite metric by a small number of tree metrics. In *39th Annual Symposium on Foundations of Computer Science*, pages 379–388, 1998.

[CKR00] G. Calinescu, H. J. Karloff, and Y. Rabani. An improved approximation algorithm for multiway cut. *J. Comput. Syst. Sci.*, 60(3):564–574, 2000.

[CKR04] G. Calinescu, H. J. Karloff, and Y. Rabani. Approximation algorithms for the 0-extension problem. *SIAM J. Comput.*, 34(2):358–372, 2004.

[FRT04] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, 2004.

[GKL03] A. Gupta, R. Krauthgamer, and J. R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In *44th Annual IEEE Symposium on Foundations of Computer Science*, pages 534–543, October 2003.

[GS95] J. D. Guyton and M. F. Schwartz. Locating nearby copies of replicated internet servers. In *Proceedings of SIGCOMM '95*, pages 288–298, New York, NY, USA, 1995. ACM Press.

[GVY96] N. Garg, V. V. Vazirani, and M. Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25(2):235–251, 1996.

[KL04] R. Krauthgamer and J. R. Lee. Navigating nets: Simple algorithms for proximity search. In *15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 791–801, January 2004.

[KL06] R. Krauthgamer and J. R. Lee. Algorithms on negatively curved spaces. In *47th Annual IEEE Symposium on Foundations of Computer Science*, pages 119–132. IEEE Computer Society, 2006.

[KLMN05] R. Krauthgamer, J. R. Lee, M. Mendel, and A. Naor. Measured descent: A new embedding method for finite metrics. *Geometric And Functional Analysis*, 15(4):839–858, 2005.

[KPR93] P. Klein, S. A. Plotkin, and S. Rao. Excluded minors, network decomposition, and multicommodity flow. In *25th Annual ACM Symposium on Theory of Computing*, pages 682–690, May 1993.

[Kra07] R. Krauthgamer. On triangulation of simple networks. In *19th Annual ACM Symposium on Parallel Algorithms and Architectures*, pages 8–15. ACM, 2007.

[KRS04] C. Kenyon, Y. Rabani, and A. Sinclair. Low distortion maps between point sets. In *STOC*, pages 272–280, 2004.

[KSW04] J. M. Kleinberg, A. Slivkins, and T. Wexler. Triangulation and embedding using small sets of beacons. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 444–453, 2004.

[KT02] J. Kleinberg and É. Tardos. Approximation algorithms for classification problems with pairwise relationships: metric labeling and markov random fields. *J. ACM*, 49(5):616–639, 2002.

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

[LN03] J. R. Lee and A. Naor. Metric decomposition, smooth measures, and clustering. Preprint, September 2003.

[LR88] F. T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In *29th Annual Symposium on Foundations of Computer Science*, pages 422–431, October 1988.

[LS93] N. Linial and M. Saks. Low diameter graph decompositions. *Combinatorica*, 13(4):441–454, 1993.

[LS05] U. Lang and T. Schlichenmaier. Nagata dimension, quasisymmetric embeddings, and lipschitz extensions. *International Mathematics Research Notices*, 2005(58):3625–3655, 2005. doi:10.1155/IMRN.2005.3625.

[Rao99] S. Rao. Small distortion and volume preserving embeddings for planar and Euclidean metrics. In *Proceedings of the 15th Annual Symposium on Computational Geometry*, pages 300–306. ACM, 1999.

[Sli05a] A. Slivkins. Distance estimation and object location via rings of neighbors. In *Proceedings of the 24th annual ACM symposium on Principles of distributed computing*, pages 41–50. ACM Press, 2005.

[Sli05b] A. Slivkins. Distributed approaches to triangulation and embedding. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 640–649, 2005.

[Tal04] K. Talwar. Bypassing the embedding: Algorithms for low dimensional metrics. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 281–290, 2004.