Approximation Algorithms for Labeling Hierarchical Taxonomies

Yuval Rabani* Leonard J. Schulman† Chaitanya Swamy‡

Abstract

We consider the following *taxonomy labeling problem*. Each node of an n-node tree has to be labeled with the values of k attributes. A partial labeling is given as part of the input. The goal is to complete this labeling, minimizing the maximum variation in labeling along an edge. A special case of this problem (which we call the *label extension problem*), where every node is either completely labeled or not labeled at all, has been considered previously.

We present an $O(\log^2 k)$-approximation algorithm based on a natural linear programming relaxation. Our results reduce the taxonomy labeling problem to another problem we introduce, called the *multicut packing problem* (on trees): given k multicommodity flow instances, find a multicut for each instance so as to minimize the maximum number of multicuts that use any single edge. Our algorithm yields an $O(\log^2 k)$-approximation algorithm for this more general problem. We show that the integrality gap of our relaxation is $\Omega(\log k)$, even when applied to the taxonomy labeling problem with 0-1 labels.

For the label extension problem, we considerably improve the previous $O(\log n)$ approximation guarantee and give the *first constant-factor approximation algorithm* for this problem. Our work relies on relating the label extension problem to questions on Lipschitz extensions of functions into Banach spaces. In particular, our approximation algorithm builds upon Matoušek's tree metrics extension theorem. Our algorithm also works for other metrics on the label-set, such as *edit distance* with unit-cost operations, and more generally any shortest path metric induced by an unweighted graph.

*rabani@cs.technion.ac.il. Computer Science Department, Technion — Israel Institute of Technology, Haifa 32000, Israel. Part of this work was done while visiting UCLA and Caltech. Supported in part by ISF 52/03, BSF 2002282, and the Fund for the Promotion of Research at the Technion.

†schulman@caltech.edu. Caltech, Pasadena, CA 91125. Supported in part by NSF CCF-0515342, NSA H98230-06-1-0074, and NSF ITR CCR-0326554.

‡cswamy@math.uwaterloo.ca. Dept. of Combinatorics and Optimization, Univ. Waterloo, Waterloo, ON N2L 3G1. Research supported partially by NSERC grant 32760-06. Work done while the author was a postdoctoral scholar at Caltech.

1 Introduction

Problem definition and motivation. We introduce the *taxonomy labeling problem*. Informally stated, this is the problem of extending a partial labeling of n items arranged as a hierarchical taxonomy (a tree) into the best complete labeling of those items. For example, we may have a corpus of documents that have been classified into a directory tree, each of whose entries we want to label by a title composed of a list of keywords or attributes that apply to this entry. We may envision that a partial labeling of some entries with some attributes has already been generated, either automatically or by human scrutiny, and our goal is to compute a good complete labeling of the directory that is consistent with this prelabeling. This description motivates the following rigorous problem definition. We are given a tree $T = (V, E)$ with partially labeled nodes. A label consists of the values of k discrete attributes. In other words, a label is an element in the Cartesian product of k discrete sets and can be viewed as a string of length k. A partial labeling of T assigns to each node a partial label, which is simply a string of length k where some (possibly all) of the entries are undetermined. Our goal is to assign values to the undetermined entries of every node, thus creating a valid label for every node. The objective is to minimize the maximum variation in labeling along any edge of T, where the variation in labeling along an edge is simply the number of attributes with different values at the two endpoints of the edge. An important special case of this problem, that we call the *label extension problem*, is the case where nodes are either completely labeled or not labeled at all. This special case was introduced by Ravi and Kececioglu [18], who called it the bottleneck tree alignment problem. The case where the tree is a star has been more extensively studied as the 1-center problem on strings: the problem is *NP*-hard even in this restricted case, even when the labels are 0-1 strings [8], and a PTAS was given by [15].

We also introduce a problem closely related to the taxonomy labeling problem, the *multicut packing problem*, which is an interesting combinatorial optimization problem in its own right. In the multicut packing problem, we are given k multicommodity flow instances.

Each multicommodity flow instance is simply a set of pairs of nodes, called commodities, and a multicut is a set of edges whose removal disconnects all these pairs. Our goal is to compute k multicuts, one for each multicommodity flow instance, such that the maximum number of multicuts that use any single edge is minimized. The connection between taxonomy labeling and multicut packing (on trees) follows from the observation that each attribute i generates a multicommodity flow instance where we have a commodity for every pair of vertices with different values for this attribute; the union of all these instances yields a multicut packing problem.

The taxonomy labeling problem can be motivated by applications involving labeling hierarchical clustering trees. Consider a setting where one is presented with a hierarchical clustering tree of some data. In many cases, such a clustering is obtained by mapping each data item to a point in some high dimensional space, and then clustering these points without considering the semantics of the data (latent semantic indexing methods [7, 17] are an important exception). Given this clustering, one would now like to assign a meaningful label to each cluster (at each level) that captures the underlying "concept" represented by the cluster. However, it may not be possible to project the cluster-centers back into the data-space to assign labels to the clusters (and this may not be the right thing to do), since these centers may not correspond to anything meaningful in the original space. The taxonomy labeling problem suggests an alternative approach. It is plausible to expect that some of the clusters, such as the individual data items represented by the leaves of the clustering tree, and possibly the root which represents the entire data, have already been labeled appropriately. So one might seek to obtain the best complete labeling of the clustering tree that is consistent with this prelabeling — a problem that fits nicely into the framework of taxonomy labeling. We mention that there are two natural objective functions that one could consider to measure the "goodness" of a complete labeling. One is the min-max objective, which we consider, of minimizing the maximum cost (i.e., variation in labeling) of an edge; the other is the min-sum objective of minimizing the sum of edge costs. Both objectives have their pros and cons. The min-sum objective can be biased by over-represented data, and might produce a lopsided labeling where some edges incur an unacceptably high cost so that most edges have a low cost. The min-max objective produces a more equitable labeling but is more susceptible to outliers. (These factors also come into play in other settings, e.g., the k-center vs. the k-median objective functions in clustering.) In our setting, the min-sum objective gives rise to the metric labeling problem [13] (as discussed below), which can be solved in polynomial time on trees and is a relatively well-understood problem. However, to our knowledge, the min-max objective has not been considered previously. We initiate an investigation of this objective by considering the taxonomy labeling problem. Our work exposes connections between this problem and other interesting problems, such as the multicut packing problem, and Lipschitz extension problems considered in the metric embeddings literature (as shown below), which provides further motivation for studying the min-max problem.

Taxonomy labeling is also motivated by the following sequence alignment problem that arises in computational biology: given an evolutionary tree of a set of species with leaves labeled with the DNA sequences of the given species, one has to assign sequences (labels) to the internal nodes (which represent ancestral species) so that an alignment of the sequences along the edges yields a good *multiple sequence alignment* of the leaf-sequences. Ravi and Kececioglu [18] introduced the label extension problem in this context, calling it the bottleneck tree alignment problem. In fact, they propose the min-max objective as a way of preventing bias in the computed alignment that could result due to over-represented sequences (see also [1]).

Taxonomy labeling, especially the special case of the label extension problem, is also related to classical *Lipschitz extension problems* considered in the metric embeddings literature [10, 11, 16, 14]. In these problems, we are given a function f from a subspace X of a given metric space (Y, d) into a Banach space Z, and the objective is to extend f to all of Y while preserving its Lipschitz constant (up to constant factors). Matoušek [16] considered this problem when the space (Y, d) is a tree metric space. Our label extension problem can be viewed as a *discrete analogue* of this problem where the space Z is a discrete metric space.

Our results. We present a polynomial time $O(\log^2 k)$-approximation algorithm for the taxonomy labeling problem, where k is the number of attributes, that is based on a careful rounding of the optimal solution to a natural linear programming relaxation for the problem. This result is described in Section 4. Our algorithm exploits the reduction between the taxonomy labeling and multicut packing problems outlined above, and actually gives an $O(\log^2 k)$-approximation algorithm for the more general multicut packing problem. In Section 6, we complement our upper bounds by proving an integrality gap of $\Omega(\log k)$ for the relaxations of both problems.

In Section 5, we consider the label extension problem and obtain significantly improved results for this

case. We exploit the connection between this problem and Lipschitz extension problems considered in the metric embeddings literature. In particular, we build upon a result of Matoušek [16] to give a *constant-factor approximation algorithm* for this problem. In applications involving labeling evolutionary trees (also called phylogenies), there are other measures of variation in the labeling along edges that might be more suitable, such as edit distance (instead of the Hamming distance in the definition which measures the number of attributes whose values differ). Our results extend with the *same performance guarantees* to the edit-distance version of the problem (with unit-cost operations). In fact, our results extend to a broad class of metrics on the label set which includes *any shortest path metric* induced by an unweighted graph.

We conclude in Section 7 with some generalizations of multicut packing and taxonomy labeling.

Related work. We are not aware of any previous work on the general taxonomy labeling and multicut packing problems, although special cases of the former have been considered. Caprara, Panconesi and Rizzi [3] studied a problem somewhat related to multicut packing; they consider the problem of finding the largest collection of edge-disjoint cuts in an undirected graph and show that this is roughly equivalent to finding the largest independent set.

As mentioned earlier, the label extension problem was introduced by Ravi and Kececioglu [18] in the context of labeling phylogenies. (In their version, all leaves of the tree T are completely labeled and all internal nodes are completely unlabeled, but we show in Section 5 that this is without loss of generality.) Ravi and Kececioglu gave an $O(\log n)$-approximation algorithm (even for the edit-distance version) in the special case where every internal node of T has degree at least 3. In contrast, we obtain a constant-factor approximation guarantee for any tree, thus significantly improving and extending previous work.

A special case of the label extension problem is where we have a leaf-labeled star and the root is unlabeled (and the metric is the Hamming distance). This is the *1-center problem* on strings, also known as the *closest string* problem. Frances and Litman [8] considered this problem in the context of coding theory and showed that it is *NP*-hard, even with only 0-1 strings; a PTAS was obtained recently by Li, Ma and Wang [15]. The label extension problem is related to the *tree alignment problem* that has been widely studied in the sequence-alignment literature. This is the min-sum problem version, where we want to compute sequences (labels) for the internal nodes so as to minimize the *sum* of the (edit) distances along the edges. This problem is also *NP*-hard [20], but various PTAS's are known [22, 19, 21].

The setting of the taxonomy labeling problem is similar to that of the *metric labeling* [13] and *0-extension* [12, 4] problems. In all these problems, we have a set of labels endowed with a metric, and we have to assign a label to each node of an input graph, subject to some constraints that restrict the set of allowed labels for each node.[1] In particular, the input to the label extension problem is exactly the same as that to the 0-extension problem. However, the crucial difference between the taxonomy labeling problem and the metric labeling problem (on trees) lies in the objective function: the former problem is a min-max problem where we seek to minimize the maximum cost of an edge, whereas the latter considers the min-sum objective of minimizing the *sum of costs* of all the edges. Consequently results for the metric labeling and 0-extension problems do not directly carry over to our problem. For example, it is well known that the metric labeling problem can be solved optimally on trees via dynamic programming; in contrast, as mentioned above, the taxonomy labeling problem is *NP*-hard even on a leaf-labeled star with 0-1 attribute values. (Technically, there is also the subtle distinction that in the taxonomy labeling problem the label set and the metric are given *implicitly* and may be exponentially large, whereas in metric labeling and 0-extension the label-sets are explicitly listed. But the min-sum objective of metric labeling allows one to decouple the problem into k separate metric labeling instances, one for each attribute, each of which can be solved optimally in polynomial time, thus, yielding a polynomial time algorithm even for such implicitly given label-sets.)

Finally, as mentioned previously, the taxonomy labeling problem, especially the special case of the label extension problem, is related to questions on the extensions of Lipschitz functions into Banach spaces [10, 11, 16, 14]. These questions are of the following flavor: given a metric space (Y, d), a Banach space Z (a complete vector space with a norm $\|\cdot\|_Z$), and a function $f : X \mapsto Z$ where $X \subseteq Y$, does there always exist an extension $f' : Y \mapsto Z$ of f such that the *Lipschitz constant* of f', defined as $\|f'\|_{lip} = \sup_{x \neq y \in Y} \frac{\|f'(x)-f'(y)\|_Z}{d(x,y)}$, is at most $C \cdot \|f\|_{lip}$ where C is an absolute constant? The most closely related such problem is the problem considered by Matoušek [16], where the space (Y, d) is a tree metric space (T, d_T). Matoušek answered this question positively by actually constructing an extension f'

[1] The statement in [13] also includes label-assignment costs; the equivalent formulation in [6] matches the above framework.

with the desired Lipschitz properties. Our label extension problem can be discussed as a discrete analogue of Matoušek's Lipschitz extension problem, where the target space Z is not a Banach space but a discrete metric space. Whereas such Lipschitz extensions of functions into discrete metric spaces need not always exist, we build upon Matoušek's construction to prove a positive result in a similar vein for our discrete label extension problem. We show that one can always get an extension whose Lipschitz constant, which is precisely the maximum cost of any tree edge, is at most a constant times $\lceil \|f\|_{lip} \rceil$, thus obtaining a constant-factor approximation algorithm for the label extension problem.

2 Problem definition and preliminaries

The Taxonomy Labeling Problem. We are given n partially labeled items that are classified into a hierarchical taxonomy represented by a (rooted) tree $T = (V, E)$. A label consists of the values of k attributes $\{1, 2, \ldots, k\}$ where attribute i takes a value from a finite set Σ_i, that is, a label is an element of $\Sigma_1 \times \cdots \times \Sigma_k$. A partial labeling assigns values to some of the attributes of some of the nodes; more formally, it is a function $\varphi : V \mapsto (\Sigma_1 \cup \{*\}) \times \cdots \times (\Sigma_k \cup \{*\})$, where $* \notin \bigcup_i \Sigma_i$ denotes an undetermined value. We use $\varphi_i(v)$ to denote the value of the i-th attribute of node v, and Σ to denote the tuple $(\Sigma_1, \ldots, \Sigma_k)$. In the taxonomy labeling problem, we are given a partial labeling φ and we have to extend it to a *complete labeling* of V by assigning values to the k attributes for every node. The objective is to minimize the variation in labeling along the edges of T. More precisely, a solution consists of a function $\varphi' : V \mapsto \Sigma_1 \times \cdots \times \Sigma_k$ such that for every $v \in V$ and attribute $i \in \{1, 2, \ldots, k\}$, $\varphi'_i(v) \neq \varphi_i(v)$ iff $\varphi_i(v) = *$. The cost of this solution is $\max_{(u,v)\in E} |\{i \in \{1, \ldots, k\} : \varphi'_i(u) \neq \varphi'_i(v)\}|$, and the objective is compute a solution φ' with minimum cost.

An interesting special case of this problem is when every node v is either completely labeled or unlabeled by the input partial labeling, that is, either $\varphi_i(v) \neq *$ for every i, or $\varphi_i(v) = *$ for every i. We call this special case the ***label extension problem***.

The Multicut Packing Problem. Our algorithm for taxonomy labeling yields an approximation to a closely-related problem, namely the *multicut packing problem* (on trees). In this problem, we have a tree $T = (V, E)$ and k multicommodity flow instances on T. Each multicommodity flow instance consists of a set of (s, t) pairs where $s, t \in V$, called commodities: the ith instance consists of the source-sink pairs $M_i = \{(s^i_1, t^i_1), \ldots, (s^i_{m_i}, t^i_{m_i})\}$, and we have k such sets for $i = 1, \ldots, k$. The goal is to pack multicuts for these multicommodity flow instances so as to minimize the maximum "load" on an edge. That is, we want to compute sets $F_1, \ldots, F_k \subseteq E$ such that the removal of the edges in F_i disconnects all (s, t) pairs in M_i for each i, and the objective is to minimize $\max_{e\in E} |\{i : e \in F_i\}|$. We use M and F to denote respectively $(M_1, \ldots, M_k)$ and $(F_1, \ldots, F_k)$.

Observe that any instance (T, Σ, φ) of the taxonomy labeling problem gives rise to a multicut packing instance (T, M) where each attribute i generates a multicommodity flow instance M_i consisting of the pairs $\{(u, v) \in V \times V : \varphi_i(u) \neq \varphi_i(v),\ \varphi_i(u), \varphi_i(v) \neq *\}$. We show below that this yields an approximation-preserving reduction; that is, any approximation algorithm for the multicut packing problem yields an approximation algorithm for the taxonomy labeling problem with the same approximation ratio.

LEMMA 2.1. *Any solution to the taxonomy labeling problem (T, Σ, φ) yields a solution to the corresponding multicut packing instance (T, M) of no greater cost, and vice versa.*

Proof. Let φ' be any solution to the taxonomy labeling instance. It is easy to see that setting $F_i = \{(u, v) \in E : \varphi'_i(u) \neq \varphi'_i(v)\}$ yields a multicut for M_i, and that the maximum load on an edge under the solution F is exactly the same as the cost of φ'.

Conversely, let F be a solution to the multicut packing problem. We convert it into a solution to the labeling problem as follows. For every $i \in \{1, 2, \ldots, k\}$, the removal of F_i partitions V into disjoint sets with the following property: for any set V_ℓ in the partition, there exists some $j_\ell \in \Sigma_i$ such that $\{\varphi_i(v) : v \in V_\ell\} \subseteq \{j_\ell, *\}$. We set $\varphi'_i(v) = j_\ell$ for every node $v \in V_\ell$. Notice that this adds at most 1 to the variation in labeling along an edge in F_i, and 0 to the variation in labeling along any other edge. Therefore, the cost of the taxonomy labeling solution is at most the cost of the solution F.

We use the following notation throughout. Let P_{uv} denote the unique path between nodes u and v in the tree, and $d_T(u, v)$ denote the number of edges (the length) on this path. If the tree T is rooted at some node r, the depth $d(v)$ of a node v is defined to be $d_T(v, r)$. The depth of the tree is the maximum depth of any node. The distance of an edge (u, v) to a node w is defined as $\min\big(d_T(u, w), d_T(v, w)\big)$, and the depth of an edge is its distance to the root. Let $\mathsf{diam}(T)$ be the diameter of T.

3 LP relaxations

We can formulate the taxonomy labeling problem as a natural integer program, with variables $x^{i,\ell}_u$ indicating

if the i-th attribute of node u is assigned value $\ell \in \Sigma_i$. Relaxing the integrality constraints yields the following linear program (LP). We use e to index the edges, u to index the nodes, and i to index the attributes.

$$\begin{aligned} \min \quad & z & \text{(A-P)} \\ \text{s.t.} \quad & \sum_{\ell \in \Sigma_i} x_u^{i,\ell} = 1 \quad \forall i, u \\ & x_u^{i,\ell} = 1 \quad \text{if } \varphi_i(u) = \ell \in \Sigma_i \\ & \sum_{i=1}^{k} \sum_{\ell \in \Sigma_i} \tfrac{1}{2} |x_u^{i,\ell} - x_v^{i,\ell}| \le z \quad \forall e = (u,v) \in E \\ & x_u^{i,\ell} \ge 0 \quad \forall u, i, \ell \in \Sigma_i. \end{aligned}$$

The first constraint states that every node must be assigned a value for every attribute, and the second one enforces that the assigned values must extend the input partial labeling φ_i. The third constraint bounds the distance along an edge by z. Although this is not written as a linear constraint, we can express it as a linear constraint by introducing variables $z_{uv}^{i,\ell} \ge 0$, and constraints $z_{uv}^{i,\ell} \ge x_u^{i,\ell} - x_v^{i,\ell}$, $z_{uv}^{i,\ell} \ge x_v^{i,\ell} - x_u^{i,\ell}$. We shall refer to this LP as the *assignment LP*.

We also have the following natural LP relaxation of the multicut packing problem. Here e indexes the edges, i indexes the multicommodity flow instances, and j indexes the (s,t) pairs in each M_i.

$$\min \Big\{ z : \sum_{e \in P_{s_j^i t_j^i}} x_e^i \ge 1 \ \ \forall i, (s_j^i, t_j^i) \in M_i; \quad \sum_{i=1}^{k} x_e^i \le z \ \ \forall e; \quad x_e^i \ge 0 \ \ \forall i, e \Big\}. \quad \text{(MC-P)}$$

The LP relaxation seeks to optimize over packings of *fractional multicuts*. A fractional multicut is an assignment of $[0,1]$-weights, which is specified by the variables x_e^i for the instance M_i, to the edges of T so that the total weight of the path P_{st} for any terminal pair (s,t) is at least 1. Let $z^*(T,M)$ denote the optimum value of (MC-P).

Strictly speaking the integrality gap of both of these relaxations can be as bad as $1/m$ where m is the number of edges, e.g., consider a path of length m whose endpoints s and t are labeled 0 and 1 ($k = 1$, $\Sigma_1 = \{0,1\}$), which gives rise to the multicut packing instance $M = (M_1)$ with $M_1 = \{(s,t)\}$. But we know that in fact, $\max(1, z^*)$ is a lower bound on the integer optimum (for the taxonomy problem, we can easily detect if $z^* = 0$ and if so, trivially obtain a complete labeling). So we abuse notation and use the term "integrality gap" in the sequel to refer to the worst ratio of the integer optimum and $\max(1, z^*)$.

4 An $O(\log^2 k)$-approximation algorithm

We now describe a deterministic $O(\log^2 k)$-approximation algorithm for the multicut packing and taxonomy labeling problems based on rounding an optimal solution to (MC-P). We remark that although the assignment LP (A-P) is at least as strong as (MC-P) (because setting $x_{(u,v)}^i = \sum_{\ell \in \Sigma_i} \frac{1}{2} \cdot |x_u^{i,\ell} - x_v^{i,\ell}|$ yields a feasible solution to (MC-P)), Section 6 proves an integrality gap of $\Omega(\log k)$ for *both* (A-P) and (MC-P), which complements our LP-based guarantees for multicut packing and taxonomy labeling.

The Algorithm. Our algorithm proceeds by first solving (MC-P). Let x be an optimal solution to (MC-P) (of value $z^*(T,M)$). Pick an arbitrary node $r \in V$ and root the input tree T at r. Let $M^{\ge 2k}$ and $M^{<2k}$ denote the instances derived from M by taking only commodities $\{s,t\}$ (in all the M_is) with $d_T(s,t) \ge 2k$ and $d_T(s,t) < 2k$ respectively. Clearly, if $F^{\ge 2k}$ and $F^{<2k}$ are multicut packings for $M^{\ge 2k}$ and $M^{<2k}$ respectively, then $F = F^{\ge 2k} \cup F^{<2k} = (F_1^{\ge 2k} \cup F_1^{<2k}, \ldots, F_k^{\ge 2k} \cup F_k^{<2k})$ is a multicut packing for M, and its cost is at most the sum of costs of $F^{\ge 2k}$ and $F^{<2k}$. There is a trivial packing $F^{\ge 2k}$ for $M^{\ge 2k}$ of cost 1: we simply set $F_i^{\ge 2k}$ to be all the edges at depths $(i-1) \pmod k$, so observe that *each* $F_i^{\ge 2k}$ intersects *every* path of length at least k, hence is a feasible solution to $M^{\ge 2k}$. So we concentrate on how to construct a solution $F^{<2k}$ for $M^{<2k}$.

1. We first modify the instance $M^{<2k}$ so that the new instance has the property that for every (s,t) pair, one of s or t is the ancestor of the other. Consider any commodity $(s,t) \in M_i^{<2k}$, for any i. Let u be the least common ancestor of s,t. We replace (s,t) by the commodity (s,u) if $\sum_{e \in P_{su}} x_e^i \ge \frac{1}{2}$, and by (t,u) otherwise. Notice that $2x$ is a feasible solution to the LP relaxation for the modified instance, since we must have either $\sum_{e \in P_{su}} x_e^i \ge \frac{1}{2}$ or $\sum_{e \in P_{tu}} x_e^i \ge \frac{1}{2}$, and its cost is $2z^*(T,M)$. Also, trivially, a multicut packing for the modified instance is also a multicut packing for the original instance. To keep notation simple, we use $M^{<k}$ to also denote the modified instance. We use the convention that in the modified instance, a source-sink pair denoted (s,t) has t as the ancestor of s.

2. We now convert the instance $(T, M^{<2k})$ into a collection of instances (T^q, M^q) such that (a) each tree T^q is a subtree of T of depth at most $4k$; (b) the trees T^q cover T, and any edge $e \in E(T)$ is contained in at most two trees T^q; and (c) for every i, the M_i^q-s form a partition of the instance $M_i^{<2k}$. Given this collection we will focus on computing a solution F^q for (T^q, M^q) for every q. The coordinate-wise union

of all the F^q-s will yield the desired multicut packing $F^{<2k}$ (i.e., $F_i^{<2k} = \bigcup_q F_i^q$) for $(T, M^{<2k})$. By property (b), the cost of $F^{<2k}$ is at most $2\max_q cost(F^q)$.

If the depth of T is at most $4k$, then T is the only tree in the cover. Otherwise, we take the union of two partitions T_0, T_{2k} of the edge-set of T. The partition T_j, where $j \in \{0, 2k\}$, contains a tree T_u for each node u of depth $j \pmod{4k}$, where T_u is the subtree of T rooted at u consisting of all descendants v such that $d_T(v,u) \leq 4k$. This satisfies property (a) by construction, and since there are exactly two partitions involved it is clear that (b) also holds. We now assign each (s,t)-pair to an arbitrary subtree T^q that contains both s and t. Note that such a subtree must exist for any (s,t)-pair. The instance M_i^q consists of all the (s,t)-pairs in $M_i^{<2k}$ assigned to tree T^q, and hence each $M_i^{<2k}$ is partitioned into the sets M_i^q.

3. **LP rounding.** We now consider each instance (T^q, M^q) separately and perform the following rounding. Recall that $h = \min\{8k, 2\,\mathsf{diam}(T)\}$. First, for every $i \in \{1, 2, \ldots, k\}$ and edge $e \in E(T^q)$, if $x_e^i \geq \frac{1}{2\log h}$ then we add e to F_i^q.

   Define the *type* of v to be the largest integer j such that 2^j divides $d(v)$, the depth of v (with respect to the root of T^q). For each node $v \in V(T^q)$ of type $j \geq 1$, we do the following. For every ℓ, $1 \leq \ell \leq j$, consider the path P_v^ℓ of length 2^ℓ leading from v towards the root $r(T^q)$ and let $I_v^\ell = \left\{i : \sum_{e \in P_v^\ell} x_e^i \geq \frac{1}{2\log h}\right\}$; if $d_T(v, r(T^q)) < 2^\ell$, then we set $I_v^\ell = \emptyset$.

   (a) We "load" all the edges connecting v to its children with all the indices in $I_v^1 \cup I_v^2$, that is, for every $i \in I_v^1 \cup I_v^2$, we add all the edges connecting v to its children to F_i^q.

   (b) For $\ell \geq 3$, we consider the $2^{\ell-2}$-depth subtree rooted at v, and we distribute the indices in I_v^ℓ evenly among all edges of the $2^{\ell-3}$-depth bottom-half of this subtree. More precisely, we partition I_v^ℓ arbitrarily into $2^{\ell-3}$ equal-sized (up to an additive 1) subsets indexed by $z = 0, \ldots, 2^{\ell-3}-1$. For each $z = 0, \ldots, 2^{\ell-3}-1$, we also define a set of edges in the subtree of T^q rooted at v: the z-th edge-set contains all the edges at distance $2^{\ell-3}+z$ from v. Note that these edge-sets are disjoint, and some of the edge-sets could be empty. For every i in the z-th index set, we add all the edges in z-th edge set to F_i^q.

THEOREM 4.1. *Given a multicut packing instance (T, M), the above algorithm computes a multicut packing of cost $O(\log h + z^*(T,M)\log^2 h)$, where $h = \min\{8k, 2\,\mathsf{diam}(T)\}$.*

Proof. We now analyze the above algorithm and show that it returns a feasible solution and attains the performance guarantee stated in Theorem 4.1. The final solution is given by $F = F^{\geq 2k} \cup \left(\bigcup_q F^q\right)$, and as mentioned above, $cost(F) \leq cost(F^{\geq 2k}) + cost(F^{<2k}) \leq 1 + 2\max_q cost(F^q)$, where the last inequality follows from property (b) in step 2. So we will consider an arbitrary instance (T^q, M^q) and show that F^q is a feasible solution of cost $O(\log h + z^*(T,M)\log^2 h)$. We use z^* below to denote $z^*(T,M)$.

Feasibility. Let r' be the root of T^q (r' is the node with smallest depth among all nodes in T^q). Let (s,t) be a terminal pair in M_i^q (recall that t is the ancestor of s). Let $d = d_T(s,r')$ be the depth of s in T^q. We construct a decomposition of $P_{sr'}$ into $O(\log d)$ segments such that (a) the length of each segment is a power of 2; (b) the length of the segment adjacent to s is 1; and (c) The lengths of adjacent segments differ by a factor of at most 4.

Consider the binary expansion of d: $d = 2^{\ell_1} + 2^{\ell_2} + \cdots + 2^{\ell_g}$, where $\ell_1 > \ell_2 > \cdots > \ell_g \geq 0$ are integers. Let $\ell_{g+1} = -1$. For every $j \in \{1, 2, \ldots, g\}$, if $\ell_j > \ell_{j+1} + 1$ then we replace 2^{ℓ_j} in this sum by $2^{\ell_j - 1} + 2^{\ell_j - 2} + \cdots + 2^{\ell_{j+1}+1} + 2^{\ell_{j+1}+1}$. The terms in the resulting expansion of d give the lengths of the segments, and these segments are arranged in increasing order of length from s to r'. Property (a) is clearly satisfied, and it is not hard to see that (b) and (c) are also satisfied (the factor of 4 arises because adjacent segments may have lengths $2^{\ell_{j+1}+1}$ and $2^{\ell_{j+1}-1}$). The number of terms is precisely $1 + \ell_1$ (since each ℓ_j contributes $\ell_j - \ell_{j+1}$ terms) which is at most $1 + \log d \leq \log h$. So the number of segments in the decomposition of $P_{sr'}$ is at most $\log h$.

Now consider all the segments in this decomposition that have a non-empty intersection with P_{st}. At least one of these segments, say p, must have weight $\sum_{e \in p} x_e^i \geq \frac{1}{2\log h}$ since $\sum_{e \in P_{st}} x_e^i \geq \frac{1}{2}$. If p is just a single edge e, then $e \in F_i^q$. Otherwise, let 2^j be the length of p, $j \geq 1$. Let u be the endpoint of p that is farther from r'. Clearly u has type at least j. Also, P_{st} must completely contain another segment p' touching u and below u, because the last segment in the decomposition of $P_{sr'}$ (the one touching s) has length 1. If $j < 3$, then the edge $e \in p'$ that is adjacent to u is in F_i^q. Otherwise, the length of p' is at least 2^{j-2}, so one of the edges of p' is in F_i^q. So in every case, we have that s and t are disconnected by F_i^q, and therefore F^q is indeed a multicut packing for the instance (T^q, M^q).

Solution cost. Consider any edge $e = (u,v) \in E(T^q)$, where v is closer to the root r'. Let $I_e = \{i : e \in F_i^q\}$.

We say that an index i "loads" edge e if $i \in I_e$. We upper bound the cardinality of the set I_e. There are three types of indices in I_e. First, we have the indices in $\{i : \ x_e^i \geq \frac{1}{2\log h}\}$, and there are at most $2z^* \log h$ such indices. Next, we have $I_v^1 \cup I_v^2 \subseteq I_e$ by step 3a). There are at most $4z^* \cdot 2\log h = 8z^* \log h$ such indices. The remaining indices in I_e were all added to I_e because there exists some $\ell \geq 3$ and an ancestor w of v with type $j \geq \ell$ such that $I_w^\ell \cap I_e \neq \emptyset$, and e is in one of the edge-sets for I_w^ℓ in step 3b). Observe that $\left|I_w^\ell\right| \leq 2^\ell \cdot z^* \cdot (2\log h)$ because $\sum_{i \in I_w^\ell} \frac{1}{2\log h} \leq \sum_{i \in I_w^\ell} \sum_{e \in P_w^\ell} x_e^i \leq 2^\ell z^*$. Since the indices in I_ℓ^w are distributed evenly among $2^{\ell-3}$ disjoint edge-sets in step 3b), each edge-set contains at most $1+16z^* \log h$ indices. Therefore, $\left|I_w^\ell \cap I_e\right| \leq 1+16z^* \log h$. Notice that it must be the case that $2^{\ell-3} \leq d_T(v,w) \leq 2^{\ell-2}$, and therefore for every ℓ there is at most one such node w. (Also for any w there is at most one such ℓ.) Hence, there are at most $(\log h - 2)$ such sets I_w^ℓ, so $|I_e| \leq 10z^* \log h + (\log h - 2)(1 + 16z^* \log h) \leq \log h + 16z^* \log^2 h$.

COROLLARY 4.1. *There is an $O(\log^2 k)$-approximation algorithm for the multicut packing and taxonomy labeling problems.*

Proof. Clearly $\max\{1, z^*(T,M)\}$ is a lower bound on the optimal cost of a multicut packing for instance (T,M), so the claim for the multicut packing problem follows from Theorem 4.1. The claim for the taxonomy labeling problem then follows from Lemma 2.1.

Extensions and refinements. We can obtain an improved guarantee of $O(\log k)$ in the special cases where after the transformation in steps 1 and 2, (a) the sources s of all the (s,t)-pairs (in an instance (T^q, M^q)) are at a common depth d; or (b) the sources s of all (s,t) pairs lie at depths that are powers of 2. In the former case, the rounding proceeds by considering only nodes v of T^q at depths $d - d/2^j$ for $j = 1, \ldots, \log d$; for each such v at depth $d(v) = d - d/2^j$, we distribute the indices in $\left\{i : \sum_{e \in P_{vw}} x_e^i \geq \frac{1}{\log d}\right\}$, where w is the ancestor of v such that $d_T(v,w) = d - d(v)$, evenly among the edges of the subtree of T^q of depth 2^{j-1} rooted at v. In case (b), for each j, we consider the sub-instance consisting of (s,t)-pairs with $d(s) = 2^j$, apply the rounding procedure for the common-depth case on this sub-instance, and take the union of the resulting multicuts. In Section 6, we prove an integrality gap of $\Omega(\log k)$ on an instance of type (a), which shows that these guarantees are tight.

All the above results generalize to the case where each edge e has *capacity* u_e and we want to minimize the congestion $\max_e\big(1, |\{i : \ e \in F_i\}|/u_e\big)$. We may assume that all capacities are rational, and hence integral (because we can multiply everything by a common denominator). We may further assume that $u_e < k$ for every edge e because we can simply include every edge e with $u_e \geq k$ in all the F_is, delete all such edges and all the (s,t) pairs whose paths contain such edges, and proceed with each component of T separately. We can now replace an edge with capacity u_e by a path of length u_e whose edges have unit capacity, and thus reduce to the unit-capacity case. Note that since each $u_e < k$ this unit-capacity instance has polynomial size.

5 The label extension problem

We now consider the label extension problem, which is an important special case of the taxonomy labeling problem. Recall that here, each node v is either completely labeled or completely unlabeled by the input partial labeling φ. So for every v, either $\varphi_i(v) \neq *$ for every i, or $\varphi_i(v) = *$ for every i. Besides being an interesting special case of the taxonomy labeling problem, this problem finds applications in sequence alignment problems in computational biology, and is related to Lipschitz extension problems considered in the metric embedding literature. In this section, we give a 16-approximation algorithm for this problem.

A key ingredient of our algorithm is a construction of Matoušek [16]. Matoušek proved the following result: given a tree metric (T, d_T), a Banach space Z (a complete vector space with a norm $\|\cdot\|_Z$), and a function $f : X \mapsto Z$ where $X \subseteq T$, there exists an extension $f' : T \mapsto Z$ of f such that the Lipschitz constant of f', $\|f'\|_{lip} = \sup_{x \neq y \in T} \frac{\|f'(x) - f'(y)\|_Z}{d_T(x,y)}$ is at most $C \cdot \|f\|_{lip}$ where C is an absolute constant. Observe that given a tree $T = (V, E)$, if we take the "source" space to be the discrete space (V, d_T) where d_T is the induced shortest path metric on V, then the Lipschitz constant of an extension f' is *precisely* the maximum distance along an edge. Notice that the label extension problem is a discrete version of this *Lipschitz extension problem*, where the "target" label space Z is *not* a Banach space but a discrete space. In our problem, the source space is $(V(T), d_T)$, the target space is $Z = (L, d)$, $L = \Sigma_1 \times \cdots \times \Sigma_k$, $d(x,y) = \sum_{i=1}^k d_i(x_i, y_i)$ where d_i is the uniform metric on Σ_i, $X = \{v \in V(T) : \ \varphi(v) \in L\}$, and $f = \varphi|_X$.

However, note that the *LP relaxation* (A-P) of the problem, where the target space is the space of all fractional assignments of labels to vertices, falls into the setting handled by Matoušek's result. Let $X \subseteq V(T)$ be the set of all completely labeled nodes. Note that $\|\varphi|_X\|_{lip} = \max_{u \neq v \in X} \frac{d(\varphi(u), \varphi(v))}{d_T(u,v)}$ is a trivial lower bound on the optimal value of the LP (A-P).

Surprisingly, Matoušek's result shows (constructively) that there is a fractional solution $\{x_u^{i,\ell}\}$ of value at most a constant times $\|\varphi|_X\|_{lip}$ (thus, yielding a constant-factor approximation algorithm for the linear program).

We prove a discrete version of Matoušek's result. We work with the lower bound $LB = \lceil \|\varphi|_X\|_{lip} \rceil$ and show that one can always find an extension φ' of cost at most $16 \cdot LB$, thus obtaining a 16-approximation algorithm for the label extension problem. We also show that the algorithm generalizes to handle a broad class of label spaces (L, d) that satisfy a certain "Banach-like" property.

Let $\big(T = (V, E), \Sigma, \varphi\big)$ be an instance of the label extension problem. We call a node v prelabeled if $\varphi(v) \in L$ and unlabeled otherwise. Let $X = \{v \in V : \varphi(v) \in L\}$. We argue that we may assume that X consists of exactly all the leaves of T. Suppose that some internal node is prelabeled. Then we can consider each component of $T \setminus X$ with edges joining the component to its neighbors in X, separately; each such component has only its leaves as the prelabeled nodes, and these components partition the edges so the cost for the entire tree is the maximum cost of any component. So we may assume that X is a subset of the leaves of T. Suppose that some leaf is unlabeled. Consider the subtree T' whose leaf-set is X. Suppose we have a labeling of T'. For any component T_i in the edge-set $T \setminus T'$, exactly one of its node v lies in T', and hence is now labeled; we assign the same label to all the nodes of T_i. Doing this for every T_i gives a labeling of T, whose cost is the same as that of the labeling of T'. So assume that X is the leaf-set of T. We now review Matoušek's construction of a suitable tree cover of T, which plays a crucial role in our algorithm as well. We root the tree at an arbitrary unlabeled node $r \in V \setminus X$.

Matoušek's construction. We will build tree-collections $\Gamma_0, \Gamma_1, \ldots$ such that the trees in $\Gamma = \bigcup_j \Gamma_j$ partition V. Let $\mathsf{leaf}(v)$ denote the leaf closest to node v. Γ_0 contains a single tree T_r rooted at r consisting of all nodes v such that $d_T(v, r) \le d_T(r, \mathsf{leaf}(r))/2$. In general $\Gamma_{j+1},\ j \ge 0$ is built from Γ_j as follows: for each child u of each leaf of every tree in Γ_j, we have a tree T_u in Γ_{j+1} rooted at u consisting of all the descendants v of u such that $d_T(v, u) \le d_T(u, \mathsf{leaf}(u))/2$. Note that each leaf forms a singleton tree in Γ.

Now the labeling is constructed as follows. For the single tree $T_r \in \Gamma_0$, we label all nodes of T_r with $\varphi(\mathsf{leaf}(r))$. For singleton trees $T_u \in \Gamma$, we assign u the label $\varphi(\mathsf{leaf}(u))$. For every other tree T_u in $\Gamma_{j+1}, j \ge 0$ rooted at u, where u is the child of a leaf of tree $T_w \in \Gamma_j$ (rooted at w), we do the following: we assign all the leaves of T_u the label $y = \varphi(\mathsf{leaf}(u))$, and assign u the label $x = \varphi(\mathsf{leaf}(w))$. Let $D = d(x, y)$. Note that all leaves in T_u are at the same distance h from u. Now it is easy to assign labels to the intermediate nodes of T_u, incurring cost at most $\lceil \frac{D}{h} \rceil$ along any edge. Essentially, we have a path-labeling problem where the endpoints of a path of length h are labeled x and y. Let $\ell_0 = x, \ell_1, \ldots, \ell_D = y$ be a shortest path between x and y in L, so $d(\ell_j, \ell_{j+1}) = 1$. We assign each node v at distance $d_T(v, u) = j$, the label $\ell_{\lceil \frac{jD}{h} \rceil}$. Doing this for every tree $T_u \in \Gamma$ yields the complete labeling φ'.

THEOREM 5.1. *The above algorithm returns a solution φ' of cost at most $16 \cdot LB$.*

Proof. Clearly for a leaf u, since $\mathsf{leaf}(u) = u$ and u appears as a singleton tree in Γ, we have $\varphi'(u) = \varphi(u)$, so φ' is indeed an extension of φ.

We identify two types of edges: edges that are internal to some tree $T_u \in \Gamma$, and edges (u, v), between trees $T_w \in \Gamma_j$ and $T_u \in \Gamma_{j+1}$ where v is a leaf of T_w and the parent of u. First consider the internal edges. The edges in $T_r \in \Gamma_0$ clearly incur 0 cost. So consider an edge in $T_u \in \Gamma_{j+1}$ where u is the child of a leaf of some tree $T_w \in \Gamma_j$ rooted at w. Let $a = \mathsf{leaf}(w)$, $b = \mathsf{leaf}(u)$, and $x = \varphi(a)$, $y = \varphi(b)$. Let $D = d(x, y)$ and h be the depth of T_u. It is clear from the labeling of T_u that the distance along any edge in T_u is at most $\lceil \frac{D}{h} \rceil$. We show that $d_T(a, b) \le 16h$, which implies that $LB \ge \frac{1}{16} \cdot \lceil \frac{D}{h} \rceil$. We have $d_T(a, b) \le d_T(a, w) + d_T(w, u) + d_T(u, b) \le \frac{3d_T(a,w)}{2} + 1 + d_T(u, b)$. Also $d_T(a, w) \le d_T(b, w) \le d_T(b, u) + d_T(u, w) \le d_T(b, u) + \frac{d_T(a,w)}{2} + 1$, so $\frac{d_T(a,w)}{2} \le 1 + d_T(b, u)$. So we get that $d_T(a, b) \le 4d_T(u, b) + 4$. Finally, note that $h \ge \frac{d_T(u,b)-1}{2}$. So we have $d_T(a, b) \le 8h + 8 \le 16h$. Thus, the cost for the edges of T_u is at most $16 \cdot LB$.

Now consider an edge (u, v) between trees $T_w \in \Gamma_j$ and $T_u \in \Gamma_{j+1}$ as above. Let $a = \mathsf{leaf}(w)$, $b = \mathsf{leaf}(u)$. If T_u is not a singleton, then u and v are both labeled with $x = \varphi(a)$. Otherwise u is assigned label $y = \varphi(b)$, and the distance along edge (u, v) is $d(x, y)$. Note that since T_u is a singleton, we must have $d_T(u, b) \le 1$. So arguing as above, we can show that $d_T(a, b) \le 8$ which implies that $d(x, y) \le 8 \cdot LB$.

Arbitrary label spaces. The connection with the metric labeling and 0-extension problems mentioned in the Introduction suggests a natural generalization where the label space is an arbitrary metric space (L, d). On trees, this generalization (and the taxonomy labeling problem) can be solved *exactly* in time polynomial in the size of the tree and the *size of the label space* via dynamic programming. However in many cases, such as the case above, the label space is described succinctly in the

input, and its size is *exponentially* large in the size of this description. We argue that the above algorithm yields the same approximation guarantee for a large class of such succinctly-describable label spaces. The only property about the space (L, d) that we needed in the algorithm above is that for any $x, y \in L$, if $D = d(x, y)$ then there exists a path $\ell_0 = x, \ell_1, \ldots, \ell_D = y$ in the label space with $d(\ell_j, \ell_{j+1}) = 1$ for all $j = 1, \ldots, D-1$, and $d(x, y)$ and such a path ℓ_0 can be computed in polynomial time. For example, *any shortest-path metric of an (given) unweighted graph* satisfies this property. So instead of the uniform metric on Σ_i, we can take d_i to be any shortest-path metric of an unweighted graph. One very useful example of a shortest-path metric is the edit-distance metric with unit-cost operations, which is widely used in sequence-alignment problems in computational biology. Notice that here the space (L, d), a set of strings with arbitrary lengths with the edit-distance metric, is not a "separable" space; yet for any $x, y \in L$, one can compute a shortest x-y path in time polynomial in the description length and therefore apply our algorithm.

6 An $\Omega(\log k)$ integrality gap

We now show that the integrality gap of (A-P) and (MC-P) is at least $O(\log k)$, even on trees. Here k is the number of attributes in the taxonomy problem, and the number of multicommodity flow instances in the multicut packing problem. We generate an instance of the multicut packing problem, which will also yield an instance of the taxonomy problem.

Let $1 \le c \le k$ be a parameter that we will fix later. Let $A = \{1, \ldots, k\}$. The tree T in the instance is rooted at r and has depth $d = k/c - 1$ (assume that c divides k). Each node in the tree at depth j will correspond to a subset of A of size $k - jc$. For every subset $B \subseteq A$ of size $k-c$, we have a node u_B at depth 1 (i.e., attached to r). In general suppose we have constructed the tree up to depth j. For each node u_B at depth j corresponding to a subset $B \subseteq A$ with $|B| = k - jc$, we add a child $v_{B'}$ for every subset $B' \subseteq B$ of size $|B| - c = k - (j+1)c$. Doing this for all $j = 1, \ldots, d$ gives the tree T of depth d. Note that there could be distinct nodes u_B and v_B that correspond to the same subset B of A. For each leaf node u_B, for every $i \in B$ we add the source-sink pair (u_B, r) to M_i. Note that $|B| = c$. Thus, all sinks are at the root and all sources are at the leaves.

We claim that any integer solution F must use some edge in at least c multicuts, that is, for some e we must have $|\{i : e \in F_i\}| \ge c$. We prove the following by induction: if for some node u_B at depth j, all edges in the subtree rooted at u_B have "load" less than c, then there is a subset $B' \subseteq B$ with $|B'| \ge c$ such that for every $i \in B'$, F_i contains an edge on the $u_B - r$ path. The desired claim now follows, because if we take a node u_B at depth 1, then this implies that either some edge in the subtree rooted at u_B, or the edge (r, u_B) has load at least c. The base case, $j = d$, is trivially true. Suppose the statement holds for $j+1$. Consider a node u_B at depth j and suppose all edges in its subtree have load less than c. Let $B' = \{i : F_i \cap P_{u_B r} \neq \emptyset\}$. If $|B'| < c$, then take some $C \subseteq B \setminus B'$ such that $|C| = |B| - c$ and consider the child v_C of u_B. By the induction hypothesis, there exists some $C' \subseteq C$, $|C'| \ge c$ such that for each $i \in C'$, $F_i \cap P_{v_C r} \neq \emptyset$. Since $|\{i : (u_B, v_C) \in F_i\}| < c$ by assumption, there must be some $i \in C'$ such that $F_i \cap P_{u_B r} \neq \emptyset$. But then $i \in B'$ contradicting the choice of C. This proves the induction step, and hence the statement.

Consider the following fractional solution. For each node u_B at depth $j = 1, \ldots, d$, we set $x^i_e = \frac{c/H_d}{|B|} = \frac{c/H_d}{k-jc}$ for every $i \in B$ on the edge e joining u_B to its parent. Here H_d denotes the d-th harmonic number. All other x^i_e's are 0. This is a feasible solution since for any leaf u_B and any $i \in B$ we have $\sum_{e \in P_{u_B r}} x^i_e = \frac{c}{H_d} \cdot \sum_{j=1}^{d} \frac{1}{k-jc} = 1$. For any edge e, we have $\sum_i x^i_e = c/H_d$. This shows an integrality gap of $\min(c, H_d)$ (recall that we are comparing against the bound $\max(1, z^*)$). Setting $c = \sqrt{k}$ gives an integrality gap of $\Omega(\log k)$.

We can translate the above instance to a taxonomy labeling instance. We have k attributes, with $\Sigma_i = \{0, 1\}$ for all i. Set $\varphi(r) = 1^k$. For each leaf u_B, we set $\varphi_i(u_B) = 0$ for $i \in B$, and $*$ otherwise. Any integer solution must have cost at least c (by Lemma 2.1). Also the above fractional solution yields a solution to (A-P) of the same cost: we set $x^{i,0}_u = 1 - x^{i,1}_u = \sum_{e \in P_u r} x^i_e$ for every node u and every attribute i.

7 Extensions to arbitrary graphs

Multicut packing on graphs. We now consider the multicut packing problem on general graphs and obtain an $O\big(\log |\max_i M_i| \cdot \frac{\log n}{\log \log n}\big)$-approximation algorithm by rounding an optimal solution to (MC-P) (which continues to be a valid relaxation). Let (G, M) be a multicut packing instance. We first solve (MC-P). Suppose we have an LP-based $\alpha(g)$-approximation algorithm for the multicut problem, where g is the number of commodities. Note that $\alpha(g) = O(\log g)$ [9]. The variables $\{x^i_e\}$ yield a fractional multicut for the instance M_i. Using a theorem of Carr and Vempala [5], one can use such an LP-based α-approximation algorithm to decompose this fractional solution x^i in polynomial time into a *convex combination* of integer solutions, $\sum_j \lambda^{i,j} \hat{x}^{i,j}$ (where $\lambda^{i,j} \ge 0$, $\sum_j \lambda^{i,j} = 1$; each $\hat{x}^{i,j}$ is a multicut for M_i), such that, for every edge e

we have $\sum_j \lambda^{i,j}\hat{x}^{i,j}_e \le \alpha \cdot x^i_e$. We obtain such a convex combination for each i, and then choose the solution $\hat{x}^{i,j}$ with probability $\lambda^{i,j}$. We do this independently for each i. Notice that the expected "load" of every edge is at most $\alpha(\max_i |M_i|)z^*$. Thus, by Chernoff bounds one obtains that with high probability, the load of every edge is $O\big(\alpha(\max_i |M_i|) \cdot \frac{\log n}{\log\log n}\big) \cdot \max(1, z^*)$. Also observe that if $z^* = \Omega(\log n)$, then the approximation guarantee improves to $O\big(\alpha(\max_i |M_i|)\big)$.

When each M_i is a multiway-cut instance, that is, (G, M) is a *multiway-cut packing problem*, we have $\alpha(g) \le 1.5$ [2] so one obtains an approximation ratio of $O\big(\frac{\log n}{\log\log n}\big)$ (and $O(1)$ if $z^* = \Omega(\log n)$). Notice that the integrality gap instance in Section 6 can be converted to an *s-t cut packing* problem on general graphs by adding a supersource for every i and connecting it to all the sources in M_i with 0-capacity edges.

THEOREM 7.1. *The multicut packing problem on general graphs admits an $O\big(\log(\max_i |M_i|) \cdot \frac{\log n}{\log\log n}\big)$-approximation algorithm. The ratio improves to $O\big(\frac{\log n}{\log\log n}\big)$ for the multiway-cut packing problem. These guarantees improve to $O\big(\log(\max_i |M_i|)\big)$ and $O(1)$ respectively when $OPT_{\text{(MC-P)}} = \Omega(\log n)$.*

The graph labeling problem. This is the generalization of the taxonomy labeling problem where the underlying graph is an arbitrary graph instead of a tree. Let $\big(G = (V, E), \Sigma, \varphi\big)$ be an instance of the graph labeling problem. We can reduce this to multiway-cut packing as follows. For every attribute i and every $\ell \in \Sigma_i$, we merge all nodes v with $\varphi_i(v) = \ell$ into a terminal t^i_ℓ. Each attribute i gives rise to a multiway-cut instance M_i consisting of the terminals $\{t^i_\ell\}_\ell$. Any feasible multiway-cut-packing solution yields a feasible solution to the graph labeling problem of no greater cost, and vice-versa. Thus, we obtain the same guarantees as those stated in Theorem 7.1 for multiway-cut packing.

References

[1] A. Ben-Dor, G. Lancia, J. Perone, and R. Ravi. Banishing bias from consensus sequences. In *Proceedings of the 8th CPM*, pages 247–261, 1997.

[2] G. Calinescu, H. Karloff, and Y. Rabani. An improved approximation algorithm for MULTIWAY CUT. *Journal of Computer and System Sciences*, 60(3):564–574, 2000.

[3] A. Caprara, A. Panconesi, and R. Rizzi. Packing cuts in undirected graphs. *Networks*, 44(1):1–11, 2004.

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

[5] R. Carr and S. Vempala. Randomized metarounding. *Rand. Struc. and Algorithms*, 20(3):343–352, 2002.

[6] C. Chekuri, S. Khanna, J. Naor, and L. Zosin. A linear programming formulation and approximation Algorithms for the metric labeling problem. *SIAM J. on Discrete Mathematics*, 18(3):608–625, 2004.

[7] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[8] M. Frances and A. Litman. On covering problems of codes. *Theory of Comput. Syst.*, 30:113–119, 1997.

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

[10] W. Johnson and J. Lindenstrauss. Extensions of Lipschitz maps into a Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

[11] W. Johnson, J. Lindenstrauss, and G. Schechtman. Extensions of Lipschitz maps into Banach spaces. *Israel Journal of Mathematics*, 54:129–138, 1986.

[12] A. Karzanov. Minimum 0-extensions of graph metrics. *European J. of Combinatorics*, 19(1):71–101, 1998.

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

[14] J. Lee and A. Naor. Extending Lipschitz functions via random metric partitions. *Inventiones Mathematicae*, 160(1):59–95, 2005.

[15] M. Li, B. Ma, and L. Wang. On the closest string and substring problems. *Journal of the ACM*, 49(2):157–171, 2002.

[16] J. Matoušek. Extension of Lipschitz mappings on metric trees. *Commentationes Mathematicae Universitatis Carolinae*, 31(1):99–104, 1990.

[17] C. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: a probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235, 2000.

[18] R. Ravi and J. Kececioglu. Approximation algorithms for multiple sequence alignment under a fixed evolutionary tree. *Disc. App. Mathematics*, 88:355–366, 1998.

[19] L. Wang and D. Gusfield. Improved approximation algorithms for tree alignment. *Journal of Algorithms*, 25(2):255–273, 1997.

[20] L. Wang and T. Jiang. On the complexity of multiple sequence alignment. *Journal of Computational Biology*, 1(4):337–348, 1994.

[21] L. Wang, T. Jiang, and D. Gusfield. A more efficient approximation scheme for tree alignment. *SIAM Journal on Computing*, 30(1):283–299, 2000.

[22] L. Wang, T. Jiang, and E. Lawler. Approximation algorithms for tree alignment with a given phylogeny. *Algorithmica*, 16(3):302–315, 1996.