The Complexity of Bottlemeck Labeled Graph Problems

Refael Hassin, Jérôme Monnot, Danny Segev

UNIVERSITE PARIS DAUPHINE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision
CNRS UMR 7024

CAHIER DU LAMSADE

257

Juillet 2007

The complexity of Bottleneck Labeled Graph Problems

Refael Hassin, Jérôme Monnot, Danny Segev

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

The Complexity of Bottleneck Labeled Graph Problems

Refael Hassin* Jérôme Monnot† Danny Segev*

Abstract

In the present paper, we study bottleneck labeled optimization problems arising in the context of graph theory. This long-established model partitions the set of edges into classes, each of which is identified by a unique color. The generic objective is to construct a subgraph of prescribed structure (such as that of being an s-t path, a spanning tree, or a perfect matching) while trying to avoid over-picking or under-picking edges from any given color.

Keywords: Bottleneck labeled problems, approximation algorithms, hardness of approximation, s-t path, s-t cut, spanning tree, perfect matching.

1 Introduction

Let $G = (V, E)$ be a directed or undirected graph, with a weight function $w : E \to \mathbb{R}_+$ and a labeling function $\mathcal{L} : E \to \{c_1, \ldots, c_q\}$. We interchangeably refer to the elements of $\mathcal{L}(E)$ as labels or colors. In addition, for $E' \subseteq E$ and $1 \leq i \leq q$, we use $\mathcal{L}_i(E') = \{e \in E' : \mathcal{L}(e) = c_i\}$ to denote the collection of c_i-colored edges in E'. With this notation in mind, the *c_i-color weight* of an edge set $E' \subseteq E$ is defined as $\sum_{e \in \mathcal{L}_i(E')} w(e)$, i.e., the total weight of all c_i-colored edges in E'.

Now let $\mathcal{P}$ be a given graph property defined on subsets of E, such as that of inducing a spanning tree, an s-t path, an s-t cut, or a perfect matching. The *min-max weighted labeled* $\mathcal{P}$ problem (henceforth, WL-min-max $\mathcal{P}$) asks to compute an edge set $E' \subseteq E$ satisfying $\mathcal{P}$ that minimizes $\max_i \sum_{e \in \mathcal{L}_i(E')} w(e)$, the maximum color weight of E'. Similarly, in *max-min weighted labeled* $\mathcal{P}$ (WL-max-min $\mathcal{P}$), the minimum color weight should be maximized. We refer to both versions as *weighted labeled bottleneck* $\mathcal{P}$ problems. Furthermore, for ease of presentation, we denote by UL-min-max $\mathcal{P}$ the unweighted special case of WL-min-max $\mathcal{P}$, that asks to minimize the maximum color frequency. Analogous notation will also be used for the corresponding max-min variant.

The complexity of WL-min-max $\mathcal{P}$ has been investigated for several graph properties by Richey and Punnen [28], Punnen [26, 27], and Averbakh and Berman [5], in the context of "optimization problems under categorization". As indicated in [28, 5], WL-min-max $\mathcal{P}$ contains both min-max weighted $\mathcal{P}$ and min-sum weighted $\mathcal{P}$ as special cases. One simply has to assign a distinct label to each edge in the former variant, and a single label for all edges in the latter variant. Similar arguments lead to an analogous result, stating that max-sum weighted $\mathcal{P}$ can be formulated in terms of WL-max-min $\mathcal{P}$. Consequently, whenever min-sum weighted (respectively, max-sum weighted) $\mathcal{P}$ is NP-hard, so is WL-min-max (respectively, max-min) $\mathcal{P}$.

*School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv 69978, Israel. Email: {hassin,segevd}@post.tau.ac.il.

†CNRS LAMSADE, Université Paris-Dauphine, Place du Maréchal de Lattre de Tassigny, 75775 Paris Cedex 16, France. Email: monnot@lamsade.dauphine.fr.

1.1 Our results

We now provide, for each problem considered in this paper, a brief description of our main findings, accompanied by a concise summary of previous work.

Labeled bottleneck *s*-*t* path

Previous work:

1. Averbakh and Berman [5] showed that WL-min-max s-t path is weakly NP-hard, even in bicolored graphs. Moreover, they proved that UL-min-max s-t path is NP-hard for an arbitrary number of colors. These results apply to both directed and undirected graphs.

2. In [15] (problem [GT54], p. 203), it was mentioned that the *pair-choice vertex* problem is NP-hard. Here, we are given a directed graph $G = (V, E)$, two specified nodes $\{s, t\} \subseteq V$, and a collection of pairwise-disjoint pairs of arcs. The objective is to determine whether there exists an s-t path traversing at most one arc from any given pair. Since UL-min-max directed s-t path can be viewed as a special case of this problem (pairs correspond to colors), the former cannot be approximated within a factor of $2 - \epsilon$ for any fixed $\epsilon > 0$, unless P=NP.

3. It is not difficult to verify that UL-max-min s-t path generalizes the *longest path* problem, even in monochromatic graphs. Therefore, the results of Karger, Motwani and Ramkumar [19, Thm. 11] imply that UL-max-min s-t path cannot be approximated within a factor of $2^{O(\log^{1-\epsilon} n)}$ for any fixed $\epsilon > 0$, unless NP $\subseteq$ DTIME($2^{O(\log^{1/\epsilon} n)}$).

New results:

1. UL-max-min s-t path is not approximable at all, unless P=NP (Theorem 4.1).

2. For a fixed number of colors, there is a fully polynomial-time approximation scheme for WL-min-max s-t path (Corollary 3.2).

3. For an arbitrary number of colors, there is a polynomial-time algorithm that constructs a feasible solution to UL-min-max s-t path in undirected graphs using $O(\sqrt{n\mathrm{OPT}})$ edges from any given color. Here, $n = |V|$ and OPT denotes the objective value of an optimal solution (Subsection 4.2). For directed graphs, the path we construct traverses $O(\sqrt{m\mathrm{OPT}})$ edges from any color, where $m = |E|$ (Subsection 4.3).

Labeled bottleneck spanning tree

Previous work: Richey and Punnen [28] showed that WL-min-max spanning tree is weakly NP-hard, even in bicolored graphs. We are not aware of previous work regarding the max-min version of this problem.

New results:

1. WL-min-max spanning tree is strongly NP-hard (Corollary 5.2); it can be approximated within a factor of $O(\log n)$ (Subsection 5.3).

2. UL-min-max spanning tree can be solved in polynomial time (Theorem 5.4).

3. UL-max-min spanning tree can be solved in polynomial time (Theorem 5.3). WL-max-min spanning tree is strongly NP-hard (Corollary 5.2), and it is also weakly NP-hard in planar bicolored graphs (Theorem 2.1).

4. For a fixed number of colors, there is a fully polynomial-time approximation scheme for both versions of weighted labeled bottleneck spanning tree (Corollary 3.2).

Labeled bottleneck perfect matching

Previous work:

1. Richey and Punnen [28] showed that WL-min-max perfect matching is weakly NP-hard, even in bicolored graphs. A stronger result has recently been obtained by Punnen [27], who proved that even the simpler WL-min-max assignment problem is strongly NP-hard.

2. Itai, Rodeh, and Tanimoto [18] proved that the following problem is NP-complete: Given a bipartite graph and a collection of pairs of edges, decide whether there exists a perfect matching that picks at most one edge from any given pair. This problem remains NP-complete for a collection of disjoint pairs [15] (problem [GT59], p. 203). Since UL-min-max perfect matching can be viewed as a special case of this problem, the former cannot be approximated within a factor of $2 - \epsilon$ for any fixed $\epsilon > 0$, unless P=NP.

3. Karzanov [20], and Yi, Murty and Spera [31] proved that, given a complete bipartite graph $K_{n,n}$ with edges colored either red or blue, the problem of finding a perfect matching consisting of exactly r red edges and $n - r$ blue edges is polynomial-time solvable[1]. Therefore, UL-min-max and UL-max-min perfect matching in complete bipartite bicolored graphs can be solved to optimality in polynomial time.

4. To our knowledge, WL-max-min perfect matching has not been studied in the literature.

New results:

1. WL-max-min perfect matching is weakly NP-hard in bicolored planar graphs (Theorem 2.1). UL-max-min perfect matching is not approximable at all in general graphs, unless P=NP (Theorem 6.1).

2. There is an approximation-preserving reduction from UL-min-max directed s-t path to UL-min-max perfect matching (Theorem 6.2).

3. For a fixed number of colors, there is a fully polynomial-time approximation scheme for both versions of weighted labeled bottleneck perfect matching (Corollary 3.2).

Labeled bottleneck s-t cut

Previous work: To our knowledge, both versions of this problem have not been studied yet.

New results:

1. UL-min-max s-t cut is NP-hard in bicolored graphs (Theorem 2.3). When the underlying graph is planar, UL-min-max s-t cut cannot be approximated within a factor of $2 - \epsilon$ for any fixed $\epsilon > 0$, unless P=NP (Theorem 7.1), and the weighted version of this problem is weakly NP-hard when the graph is bicolored as well (Theorem 2.1).

2. WL-max-min s-t cut is weakly NP-hard in planar bicolored graphs (Theorem 2.1). For an arbitrary number of colors, this problem is not approximable at all in planar multigraphs, unless P=NP (Theorem 7.2).

[1] On the other hand, the complexity of this problem in general bipartite graphs is still open.

1.2 Related work

In this section, we provide a brief survey of several frameworks to which our contributions are related. Since some of the settings under consideration have received a great deal of attention in recent years, it is beyond the scope of this writing to present an exhaustive overview. We refer the reader to the undermentioned papers and to the references therein for a more comprehensive review of the literature.

Multiobjective combinatorial optimization [13, 14, 29, 30]. The basic ingredients of a multiobjective optimization problem are typically: A set of instances $\mathcal{I}$; a set of feasible solutions $\mathcal{F}(x)$ associated with every instance $x \in \mathcal{I}$; and a collection of cost functions $w_1(x, y), \ldots, w_k(x, y)$ associated with every instance $x \in \mathcal{I}$ and feasible solution $y \in \mathcal{F}(x)$. Given an instance $x \in \mathcal{I}$, the goal is to solve $\min_{y \in \mathcal{F}(x)}\{w_1(x, y), \ldots, w_k(x, y)\}$, where the exact meaning of "min" depends on the particular setting in question. For example, it may stand for Pareto optimality (see Section 3), for aiming to minimize the worst cost function, or for lexicographically minimizing the vector of cost functions. It is not difficult to verify that WL-min-max $\mathcal{P}$ is actually a multiobjective optimization problem in disguise: The set of feasible solutions consists of all edge sets that satisfy $\mathcal{P}$; for every color c_i there is a corresponding cost function w_i which is exactly the c_i-color weight; and the goal is to minimize the maximum cost function. Minor adjustments allow us to treat WL-max-min $\mathcal{P}$ in a similar way.

Robust discrete optimization [21, 7, 12]. Very informally, robust optimization deals with decision making in environments of considerable data uncertainty, trying to come up with solutions that hedge against the worst contingency that may arise. Several alternative approaches for coping with uncertainty have been explored and exploited; however, the *scenario-based* framework of Kouvelis and Yu [21] seems most relevant to our paper. In this context, future developments are described by a finite number of scenarios, each of which corresponds to a possible realization of the unknown model parameters. The objective is to optimize against the worst possible scenario by using a min-max objective. Once again, we note that WL-min-max $\mathcal{P}$ can be easily cast as a scenario-based robust optimization problem: For every color c_i there is an analogous scenario s_i, in which the weight $w_{s_i}(e)$ of an edge $e \in E$ is set to $w(e)$ if its color is c_i, and to 0 otherwise. In addition, the cost of an edge set $E' \subseteq E$ in scenario s_i is given by $\sum_{e \in E'} w_{s_i}(e)$, which is exactly the c_i-color weight of this set.

The min-sum-max setting. A complementary line of work [28, 5, 27] on edge-colored graphs attempts to minimize the sum of the maximal edge weight picked from every given color. In particular, when all edges are associated with unit weights, a problem of this nature reduces to that of constructing subgraphs satisfying a required property while minimizing the number of colors used. Some properties that have recently been studied in this context include spanning trees [11, 8, 10, 17], s-t paths [9, 17], and perfect matchings [23].

2 Fixed Number of Colors: Hardness Results

2.1 Weak NP-hardness in bicolored graphs

In what follows, we prove that several weighted labeled bottleneck problems are NP-hard, even in planar bicolored graphs. As indicated in Subsection 1.1, WL-min-max $\mathcal{P}$ is known to be NP-hard in bicolored graphs [28, 5] for $\mathcal{P} \in \{\text{spanning tree}, s\text{-}t \text{ path}, \text{perfect matching}\}$.

Theorem 2.1. *WL-min-max $\mathcal{P}$ and WL-max-min $\mathcal{P}$ are NP-hard, even in planar bicolored graphs, for $\mathcal{P} \in \{s\text{-}t \text{ path}, s\text{-}t \text{ cut}, \text{perfect matching}, \text{spanning tree}\}$.*

Proof. We propose a reduction from the *partition* problem, whose instances consist of a collection of integers $\{a_1, \ldots, a_n\}$, and the goal is to decide whether or not there exists a subset $J \subseteq \{1, \ldots, n\}$ such that $\sum_{j \in J} a_j = \sum_{j \notin J} a_j$. To this end, for every $1 \leq i \leq n$ consider a 4-edge cycle on the set of vertices $\{x_i, l_i, y_i, r_i\}$, listed clockwise, with edge colors $\mathcal{L}(x_i, l_i) = \mathcal{L}(y_i, r_i) = c_1$, $\mathcal{L}(x_i, r_i) = \mathcal{L}(y_i, l_i) = c_2$, and weights $w(x_i, l_i) = w(x_i, r_i) = a_i$, $w(y_i, l_i) = w(y_i, r_i) = 0$. These n cycles describe a graph H'. For the perfect matching case, we define $H = H'$. For the s-t path and spanning tree cases, we construct H from H' by identifying y_i and x_{i+1} to a single vertex (for every $1 \leq i \leq n-1$), and setting $s = x_1$, $t = y_n$. Finally, for the s-t cut case, we construct H from H' by deleting the vertex y_i along with the edges (y_i, r_i) and (y_i, l_i) in each 4-cycle, and identifying $r_1, \ldots, r_n$ to a vertex s, and $l_1, \ldots, l_n$ to t.

In each of these cases, it is easy to verify that the resulting WL-min-max $\mathcal{P}$ instance has an objective value of at most $\frac{1}{2} \sum_{j=1}^{n} a_j$ if and only if there exists $J \subseteq \{1, \ldots, n\}$ with $\sum_{j \in J} a_j = \sum_{j \notin J} a_j$. Similarly, WL-max-min $\mathcal{P}$ has an objective value of at least $\frac{1}{2} \sum_{j=1}^{n} a_j$, for $\mathcal{P} \in \{s\text{-}t \text{ path}, s\text{-}t \text{ cut}, \text{perfect matching}\}$. Regarding WL-max-min spanning tree, some edge weights need to be changed: We set $w(y_i, l_i) = w(y_i, r_i) = \sum_{j=1}^{n} a_j$, for every $1 \leq i \leq n$. Consequently, the latter problem has an objective value of at least $(n + \frac{1}{2}) \sum_{j=1}^{n} a_j$ if and only if there exists $J \subseteq \{1, \ldots, n\}$ with $\sum_{j \in J} a_j = \sum_{j \notin J} a_j$. ∎

2.2 Strong NP-hardness for s-t cuts

Aissi, Bazgan and Vanderpooten [4] proved that min-max robust $\mathcal{P}$ with a fixed number of scenarios admits pseudo-polynomial algorithms for s-t paths and spanning trees in general graphs and for perfect matchings in planar graphs. Since WL-min-max $\mathcal{P}$ can be viewed as a special case of these settings (see Subsection 1.2), it follows that the corresponding min-max labeled problems have pseudo-polynomial algorithms for a fixed number of colors.

In contrast, we proceed by proving that WL-min-max s-t cut is strongly NP-hard in bicolored graphs. A similar result was established for bi-criteria s-t cut [25, Thm. 6], and more recently for min-max robust s-t cut with two scenarios [3, Cor. 1]. Unfortunately, in their reductions the resulting instances do not correspond to WL-min-max s-t cut instances, and it appears as if we cannot conclude the result for WL-min-max s-t cut in an obvious way. However, we can slightly modify the construction of Papadimitriou and Yannakakis [25].

Theorem 2.2. *WL-min-max s-t cut is strongly NP-hard in bicolored graphs.*

Proof. We propose a reduction from the *bisection width* problem. Given a connected graph $G = (V, E)$ on $2n$ vertices, a bisection is a cut (V_1, V_2) of G with $|V_1| = |V_2| = n$. The decision version of bisection width asks to determine, for a given integer k, whether there exists a bisection with at most k edges. This problem is known to be NP-complete [15] (problem [ND17], p. 210).

Given an instance of bisection width, as described above, we construct an instance $I = (G', w, \mathcal{L})$ of WL-min-max s-t cut, with $G' = (V', E')$ and $\mathcal{L}(E') = \{c_1, c_2\}$, as follows:

- G' has two additional vertices, s and t, each of which is connected to every vertex of G.
- $\mathcal{L}(s, v) = c_1$ for every $v \in V$; all other edges have color c_2.

- $w(s, v) = k(n + 1)$ and $w(t, v) = kn$ for every $v \in V$; $w(e) = n$ for every original edge $e \in E$.

We now argue that G has a bisection of size at most k if and only if I has an s-t cut whose min-max value is at most $kn(n + 1)$. If (V_1, V_2) is a bisection with at most k edges, then $(\{s\} \cup V_1, \{t\} \cup V_2)$ is an s-t cut in G' that picks c_1-colored edges of total weight $\sum_{e \in (\{s\},V_2)} w(e) = kn(n + 1)$ and c_2-colored edges of total weight $\sum_{e \in (\{t\},V_1)} w(e) + \sum_{e \in (V_1,V_2)} w(e) = kn^2 + kn = kn(n + 1)$. Conversely, let $(\{s\} \cup V_1, \{t\} \cup V_2)$ be an s-t cut in G' with min-max value of at most $kn(n + 1)$. Since each c_1-colored edge of this cut has a weight of $k(n + 1)$, it follows that $|V_2| \leq n$. In addition, the c_2-colored edges of this cut have a total weight of $n|E''| + kn|V_1|$, where $E'' = (V_1, V_2)$, and we conclude that $|V_1| \leq n + 1$. Now, if $|V_1| = n + 1$ the inequality $n|E''| + kn|V_1| \leq kn(n + 1)$ implies $E'' = \emptyset$, so G is clearly disconnected (contradicting our initial assumption); thus $|V_1| \leq n$. Finally, since $|V_1| \leq n$ and $|V_2| \leq n$, we have $|V_1| = |V_2| = n$, and therefore (V_1, V_2) is a bisection with at most k edges. ■

Theorem 2.3. *UL-min-max s-t cut is NP-hard in bicolored graphs.*

Proof. To prove the theorem, we show that a ρ-approximation for UL-min-max s-t cut can be converted in polynomial time into a ρ-approximation for WL-min-max s-t cut when the edge weights are integers upper bounded by a polynomial in n. The theorem follows from the combination of this result and Theorem 2.2.

Let $I = (G, w, \mathcal{L})$ be an instance of WL-min-max s-t cut, where $G = (V, E)$ has n vertices and $\max_{e \in E} w(e) = O(n^{O(1)})$. We replace each edge $e = (u, v) \in E$ by a collection $H(e)$ of $w(e)$ edge-disjoint paths of length two (connecting u and v), each edge of which is colored by $\mathcal{L}(e)$. The vertices u and v will be called *extreme vertices* of $H(e)$, whereas other vertices of $H(e)$ will be called *inner vertices*. We refer to the resulting UL-min-max s-t cut instance as $I' = (G', \mathcal{L}')$.

Consider an s-t cut (S', T') in G', with $s \in S'$ and $t \in T'$. We iteratively apply the following procedure for each original edge $e \in E$: If the extreme vertices of $H(e)$ appear in the same set of the partition, assign all inner vertices of $H(e)$ to that set. These changes can only decrease the total weight of $\mathcal{L}(e)$-colored edges in the current s-t cut, and therefore also its min-max value. From the resulting s-t cut (S', T'), we can find an s-t cut in G of identical min-max value by considering $(S' \cap V, T' \cap V)$, the restriction of this cut to G. ■

3 Fixed Number of Colors: An FPTAS

In what follows, we present a fully polynomial-time approximation scheme for weighted labeled bottleneck s-t path, spanning tree, and perfect matching, for a fixed number of colors.

3.1 Approximate Pareto curves

Let $\mathcal{P}$ be a property described in Section 1, and consider the multiobjective version of $\mathcal{P}$ (henceforth, $\text{Multi}_k\mathcal{P}$). An instance I of this problem consists of a graph $G = (V, E)$, and a weight vector $w(e) = (w_1(e), \ldots, w_k(e))$ for each edge $e \in E$. An edge set $E' \subseteq E$ forms a feasible solution to $\text{Multi}_k\mathcal{P}$ if it satisfies $\mathcal{P}$, and the objective value of E' is given by the vector $(\sum_{e \in E'} w_1(e), \ldots, \sum_{e \in E'} w_k(e))$. In a minimization problem, we say that a solution E' is *dominated* by E'' if $\sum_{e \in E''} w_i(e) \leq \sum_{e \in E'} w_i(e)$ for every $1 \leq i \leq k$, and the inequality is strict for at least one index; the inequalities are reversed for a maximization problem. The goal is to compute the *Pareto curve* $\mathcal{C}(I)$, which is the set of all undominated solutions to I. Finally, an *ϵ-approximate Pareto curve* for the minimization (respectively, maximization) version of Multi$_k\mathcal{P}$ is a set $\mathcal{C}_\epsilon(I)$ of solutions such that

1. $|\mathcal{C}_\epsilon(I)|$ is polynomially bounded in terms of the input size and $\frac{1}{\epsilon}$.
2. For every $E^* \in \mathcal{C}(I)$, there exists $E' \in \mathcal{C}_\epsilon(I)$ with $\sum_{e\in E'} w_i(e) \le (1+\epsilon)\sum_{e\in E^*} w_i(e)$ for every $1 \le i \le k$ (respectively, $\sum_{e\in E'} w_i(e) \ge (1-\epsilon)\sum_{e\in E^*} w_i(e)$).

When k is fixed, Papadimitriou and Yannakakis [25, Cor. 5] proposed an FPTAS for constructing ϵ-approximate Pareto curves of multiobjective s-t walk, spanning tree, and perfect matching.

3.2 The approximation scheme

We now relate the approximability of several weighted labeled bottleneck problems to that of their multiobjective counterparts. This approach has already been suggested in the context of robust optimization [21, 2], implying that results similar to those described in the next theorem can be immediately derived for the min-max variants.

Theorem 3.1. *For a fixed number of colors, the efficient construction of an ϵ-approximate Pareto curve for the maximization version of* Multi$_k\mathcal{P}$ *implies a $(1-\epsilon)$-approximation to WL-max-min $\mathcal{P}$. A similar result for the minimization version leads to a $(1+\epsilon)$-approximation to WL-min-max $\mathcal{P}$.*

Proof. For purposes of brevity, we deal with the maximization setting, noting that the arguments below can be easily adapted to the corresponding minimization problem. Let $I = (G, \mathcal{L}, w)$ be an instance of WL-max-min $\mathcal{P}$ such that $\mathcal{L}(E) = \{c_1, \dots, c_k\}$. We construct an instance $I' = (G, w')$ of Multi$_k\mathcal{P}$, where the weight vector $w'(e) = (w'_1(e), \dots, w'_k(e))$ associated with each edge $e \in E$ is defined by setting $w'_i(e) = w(e)$ if $\mathcal{L}(e) = c_i$, and $w'_i(e) = 0$ otherwise. As a result, the objective vector of a feasible solution E' to I' is given by $(\sum_{e\in\mathcal{L}_1(E')} w(e), \dots, \sum_{e\in\mathcal{L}_k(E')} w(e))$.

Now let E^* be an optimal solution to the instance I of WL-max-min $\mathcal{P}$, with an objective value of OPT(I); if this instance does not have a unique optimal solution, E^* is picked so that $\sum_{i=1}^k \sum_{e\in\mathcal{L}_i(E^*)} w(e)$ is maximized. Clearly, the Pareto curve $\mathcal{C}(I')$ contains E^*. Now, consider an ϵ-approximate Pareto curve $\mathcal{C}_\epsilon(I')$ for the maximization version of Multi$_k\mathcal{P}$. We pick a solution $E' \in \mathcal{C}_\epsilon(I')$ maximizing $\min_i \sum_{e\in E'} w_i(e)$ as a solution for WL-max-min $\mathcal{P}$; since $\mathcal{C}_\epsilon(I')$ can be constructed and enumerated in polynomial time, so is our algorithm. Moreover, by definition of $\mathcal{C}_\epsilon(I')$, since $E^* \in \mathcal{C}(I')$ there exists $E'' \in \mathcal{C}_\epsilon(I')$ satisfying $\sum_{e\in E''} w_i(e) \ge (1-\epsilon)\sum_{e\in E^*} w_i(e)$ for every $1 \le i \le k$. Hence, we conclude that E' constitutes a $(1-\epsilon)$-approximation to I, since

$$\min_{1\le i\le k} \sum_{e\in\mathcal{L}_i(E')} w(e) \ge \min_{1\le i\le k} \sum_{e\in\mathcal{L}_i(E'')} w(e) \ge (1-\epsilon) \sum_{e\in\mathcal{L}_i(E^*)} w(e) = (1-\epsilon)\mathrm{OPT}(I) \ .$$

■

By combining Theorem 3.1 and the results of Papadimitriou and Yannakakis [25] mentioned earlier, Corollary 3.2 follows. However, an important remark is in place. Even though the algorithm in [25] constructs an ϵ-approximate Pareto curve of multiobjective s-t walk, note that any such walk can be converted (by eliminating cycles) to an s-t path of no greater min-max objective value. An analogous claim regarding the max-min version is incorrect.

Corollary 3.2. *For a fixed number of colors, weighted labeled bottleneck spanning tree and perfect matching admit a fully polynomial-time approximation scheme. A similar result also holds for WL-min-max s-t path.*

4 Arbitrary Number of Colors: Labeled Bottleneck s-t Paths

For a fixed number of colors, UL-min-max s-t path is polynomial time solvable. This claim follows from the observation that we can decide whether there exists a walk connecting s and t whose objective value is exactly $p \in \{1, \dots, n-1\}$ by means of dynamic programming. In contrast, we proceed by showing that both versions of the problem under consideration become NP-hard for an arbitrary number of colors. We complement these results by devising efficient approximation algorithms.

4.1 Hardness results

We now derive new inapproximability bounds for both versions of labeled bottleneck s-t path, in undirected as well as directed graphs. To our knowledge, these results do not follow from existing work.

Theorem 4.1. *UL-min-max s-t path is not* $(2-\epsilon)$*-approximable for any fixed* $\epsilon > 0$*, and UL-max-min s-t path is not approximable at all, unless P=NP. Similar results hold for directed graphs.*

Proof. The oncoming reductions will be based on the *set packing* and *set covering* problems. Given a family of sets $\mathcal{S} = \{S_1, \dots, S_m\}$ over a ground set $X = \{x_1, \dots, x_n\}$ and an integer k, the objective of the set packing problem is to decide whether $\mathcal{S}$ contains at least k pairwise-disjoint sets. The set covering problem asks whether X can be covered by at most k sets in $\mathcal{S}$. Both set packing and set covering are known to be NP-complete [15] (problems [SP3] and [SP5], p. 221–222).

Given $(\mathcal{S}, X)$, we create an instance $I = (G, \mathcal{L})$ of unweighted labeled bottleneck s-t path, where $\mathcal{L}$ has n colors $\{c_1, \dots, c_n\}$. For each set $S_i = \{x_{i_1}, \dots, x_{i_p}\}$ we define a distinct path $H(S_i)$ consisting of p edges, each given a unique color from $\{c_{i_1}, \dots, c_{i_p}\}$. Now, the graph $G = (V, E)$ is constructed in the following way:

- Create a path of length k on the vertices $v_1, \dots, v_{k+1}$, in left-to-right order.
- Replace each edge (v_j, v_{j+1}) by m paths connecting v_j and v_{j+1}; for every $1 \le i \le m$, the i-th path is a copy of $H(S_i)$.
- Finally, set $s = v_1$ and $t = v_{k+1}$.

It is not difficult to verify that the objective value of the UL-min-max s-t path instance I is at most 1 if and only if there exists a set packing of size at least k. In addition, the objective value of the UL-max-min s-t path instance I is at least 1 if and only if there exists a set covering of size at most k. Thus, we conclude that it is NP-complete to distinguish between $\mathrm{OPT}(I) = 1$ and $\mathrm{OPT}(I) \ge 2$ for UL-min-max s-t path, and also between $\mathrm{OPT}(I) = 0$ and $\mathrm{OPT}(I) \ge 1$ for UL-max-min s-t path. ∎

4.2 UL-min-max *s*-*t* path: Approximating the undirected case

In what follows, we show how to efficiently construct an undirected s-t path using $O(\sqrt{n\text{OPT}})$ edges from any given color, where $n = |V|$ and OPT denotes the cost of an optimal solution. An essential building block of our algorithm is a constant-factor approximation for *multi-budget maximum coverage*. An instance of this problem consists of a ground set U and a collection of subsets $\mathcal{S} \subseteq 2^U$, which is partitioned into $\{\mathcal{S}_1, \ldots, \mathcal{S}_r\}$. Given an integral budget b_t for each part $\mathcal{S}_t$, the objective is to find a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ such that $\mathcal{S}'$ picks at most b_t sets from each $\mathcal{S}_t$ and such that the number of elements covered by $\mathcal{S}'$ is maximized. For these particular settings, a performance guarantee of $1 - \frac{1}{e}$ can be achieved by adopting the maximum coverage heuristic of Ageev and Sviridenko [1, Rem. 2].

The algorithm. For simplicity of presentation, it would be convenient to assume that OPT is known in advance. Clearly, this assumption can be enforced by testing $1, \ldots, n-1$ as candidate values, and returning the best solution found. We also make use of $\Delta = \Delta(n, \text{OPT})$ as a parameter whose value will be determined later.

1. $F \leftarrow \emptyset$, $H \leftarrow G$.
2. While $\text{dist}_H(s,t) > \Delta$
   (a) Create a multi-budget maximum coverage instance by: The ground set is $V(H)$; for each edge $e \in E(H)$ there is a corresponding subset V_e, consisting of the endpoints of e; these subsets are partitioned into $\{\mathcal{S}_1, \ldots, \mathcal{S}_q\}$, where $\mathcal{S}_i = \{V_e : \mathcal{L}(e) = c_i\}$; each $\mathcal{S}_i$ has a budget of OPT.
   (b) Approximate the instance defined above, and let F^+ be the collection of edges $e \in E(H)$ for which V_e is picked by the resulting solution.
   (c) $F \leftarrow F \cup F^+$, $H \leftarrow$ the contraction of F^+ in H.
3. Let P be a shortest s-t path in H. Return $F \cup P$.

Theorem 4.2. *By setting $\Delta = \sqrt{n\text{OPT}}$, the subgraph induced by $F \cup P$ picks at most $5\sqrt{n\text{OPT}}$ edges from any given color.*

Proof. We begin by showing that, for any value of Δ, step 2 terminates within no more than $\frac{4n}{\Delta}$ iterations. For this purpose, it is sufficient to prove that the number of vertices in H decreases by at least $\frac{\Delta}{4}$ whenever an edge set is contracted. Let $E^* \subseteq E$ be an optimal solution, with $\max_i |\mathcal{L}_i(E^*)| = \text{OPT}$, and consider a single iteration. Since the edges $E^* \cap E(H)$ form a subgraph of H containing an s-t path, it follows that $\{V_e : e \in E^* \cap E(H)\}$ is a feasible solution to the multi-budget maximum coverage instance defined in step 2a. Moreover, as the s-t distance in H is at least Δ, the latter solution satisfies $|\bigcup_{e \in E^* \cap E(H)} V_e| \geq \Delta$. Consequently, for the current F^+ we must have $|\bigcup_{e \in F^+} V_e| \geq (1 - \frac{1}{e})\Delta$, implying that the contraction of F^+ decreases the number of vertices by at least $\frac{1}{2}(1 - \frac{1}{e})\Delta > \frac{\Delta}{4}$.

Now, starting with an empty set of edges, in each iteration of step 2 we augment F with an edge set F^+ that contains at most OPT edges from each color. Therefore, by setting $\Delta = \sqrt{n\text{OPT}}$, the maximum number of edges we pick from any given color is at most

$$\frac{4n}{\Delta}\text{OPT} + |P| \leq \frac{4n}{\Delta}\text{OPT} + \Delta = 5\sqrt{n\text{OPT}} \ .$$

■

4.3 UL-min-max s-t path: Approximating the directed case

In the following, we demonstrate that ideas similar to those presented in Subsection 4.2 can be employed to construct a directed s-t path using $O(\sqrt{m\text{OPT}})$ arcs from any given color. Here, $m = |E|$ and OPT denotes the cost of an optimal solution.

The algorithm. Once again, we assume that OPT is known in advance, and let $\Delta = \Delta(m, \text{OPT})$ be a parameter whose value will be determined later.

1. $F \leftarrow \emptyset$, $\chi_{E\setminus F} \leftarrow$ characteristic function of $E \setminus F$.
2. While $\text{dist}_{\chi_{E\setminus F}}(s,t) > \Delta$
   (a) Create a multi-budget maximum coverage instance by: The ground set is V; for each arc $e = (u,v) \in E \setminus F$ there is a corresponding singleton $V_e = \{v\}$; these subsets are partitioned into $\{\mathcal{S}_1, \ldots, \mathcal{S}_q\}$, where $\mathcal{S}_i = \{V_e : \mathcal{L}(e) = c_i\}$; each $\mathcal{S}_i$ has a budget of OPT.
   (b) Approximate the instance defined above, and let F^+ be the collection of arcs $e \in E \setminus F$ for which V_e is picked by the resulting solution.
   (c) $F \leftarrow F \cup F^+$.
3. Let P be a shortest s-t path (with respect to $\chi_{E\setminus F}$). Return P.

Theorem 4.3. *By setting $\Delta = \sqrt{m\text{OPT}}$, the directed path P traverses at most $3\sqrt{m\text{OPT}}$ arcs from any given color.*

Proof. We first demonstrate that step 2 consists of at most $\frac{2m}{\Delta}$ iterations, by showing that we always have $|F^+| \geq \frac{\Delta}{2}$. Let P^* be an optimal solution, with $\max_i |\mathcal{L}_i(P^*)| = \text{OPT}$. In each iteration, $\{V_e : e \in P^* \setminus F\}$ is a feasible solution to the multi-budget maximum coverage instance defined in step 2a. Moreover, as $\text{dist}_{\chi_{E\setminus F}}(s,t) > \Delta$, the latter solution satisfies $|\bigcup_{e\in P^*\setminus F} V_e| \geq \Delta$. Consequently, we must have $|F^+| \geq |\bigcup_{e\in F^+} V_e| \geq (1-\frac{1}{e})\Delta > \frac{\Delta}{2}$.

Now, starting with an empty set of arcs, in each iteration of step 2 we augment F with an arc set F^+ that contains at most OPT arcs from each color. Therefore, by setting $\Delta = \sqrt{m\text{OPT}}$, the maximum number of edges P traverses from any given color is at most

$$|F| + \Delta \leq \frac{2m}{\Delta}\text{OPT} + \Delta \leq 3\sqrt{m\text{OPT}} \ .$$

■

5 Arbitrary Number of Colors: Labeled Bottleneck Spanning Trees

In Corollary 3.2 we have shown that, for a fixed number of colors, both versions of weighted labeled spanning tree admit an FPTAS. In this section, we provide hardness results, exact algorithms, and approximation algorithms for the general case of an arbitrary number of colors.

5.1 Hardness results

As indicated in Subsection 1.1, WL-min-max spanning tree is known to be weakly NP-hard [28]. Here, we show that both weighted labeled bottleneck spanning tree problems are in fact strongly NP-hard.

Theorem 5.1. *Both weighted labeled bottleneck spanning tree problems are strongly NP-hard in multigraphs.*

Proof. We propose a reduction from 3-*partition*. The input to this problem consists of a collection of integers $\{a_1, \ldots, a_{3m}\}$, satisfying $\frac{\sigma}{4} < a_i < \frac{\sigma}{2}$, where $\sigma = \frac{1}{m}\sum_{i=1}^{3m} a_i$. The objective is to decide whether there is a partition of $\{1, \ldots, 3m\}$ into sets $S_1, \ldots, S_m$ with $\sum_{i \in S_j} a_i = \sigma$ for every $1 \leq j \leq m$. The 3-partition problem is known to be strongly NP-complete [15] (problem [SP15], p. 224).

Let $\{a_1, \ldots, a_{3m}\}$ be an instance of 3-partition. We construct an instance $I = (G, w, \mathcal{L})$ for both weighted labeled bottleneck spanning tree versions, where $\mathcal{L}(E) = \{c_1, \ldots, c_m\}$ and G is a graph on the set of vertices $u, v_1, \ldots, v_{3m}$. For every $1 \leq i \leq 3m$, there are m parallel edges connecting u to v_i; each of these edges is given a distinct color in $\{c_1, \ldots, c_m\}$ and its weight is set to a_i.

The claim follows from the observation that there is a 3-partition of $\{a_1, \ldots, a_{3m}\}$ if and only if the optimal value of the WL-min-max (respectively, max-min) spanning tree instance is at most σ (respectively, at least σ). ■

Corollary 5.2. *Both weighted labeled bottleneck spanning tree problems are strongly NP-hard in simple graphs.*

Proof. We show how to transform the multigraph defined in the proof of Theorem 5.1 into a simple graph. For this purpose, for every $1 \leq i \leq 3m$ we split the vertex v_i into $v_{i,1}, \ldots, v_{i,m}$ (thereby creating a unique copy for each parallel edge connecting u to v_i), add a new vertex $v_{i,m+1}$, and add the edges $(v_{i,1}, v_{i,2}), \ldots, (v_{i,m}, v_{i,m+1})$. For every $1 \leq j \leq m$, the color of $(v_{i,j}, v_{i,j+1})$ is set to c_j, but its weight depends on the problem in question. For WL-min-max spanning tree, $w(v_{i,j}, v_{i,j+1}) = 0$, whereas for the max-min version $w(v_{i,j}, v_{i,j+1}) = m\sigma$. In the former case, the conclusion is identical to that of the original proof. In the latter case, there is a 3-partition of $\{a_1, \ldots, a_{3m}\}$ if and only if the optimal value of the WL-max-min spanning tree instance is at least $(3m^2 + 1)\sigma$. ■

5.2 Exact algorithms

Broersma and Li [8] devised a polynomial-time algorithm based on matroid intersection for computing a spanning tree using a maximum number of colors. Here, we prove that both unweighted labeled bottleneck spanning tree problems can also be solved in polynomial time by utilizing matroid intersection. It is interesting to observe that this result is in contrast to the weighted case, which was shown to be strongly NP-hard in Corollary 5.2.

Theorem 5.3. *UL-max-min spanning tree can be solved to optimality in polynomial time.*

Proof. Given an instance $(G, \mathcal{L})$ of UL-max-min spanning tree, with $G = (V, E)$, we may assume without loss of generality that OPT is known in advance, since we can test $0, \ldots, n-1$ as candidate values for this parameter, and return the best solution found. Now, since the optimal tree picks at least OPT edges from every color in $\mathcal{L}(E) = \{c_1, \ldots, c_q\}$, it follows that there exists a forest picking exactly OPT edges from any given color. Moreover, such a forest can be efficiently constructed by computing a maximum cardinality intersection[2] of the matroids M_1 and M_2, where

- $M_1 = (E, \mathcal{I}_1)$ is the graphic matroid, that is, $\mathcal{I}_1 = \{F \subseteq E : F \text{ is a forest}\}$.

[2] See, for example, [22, Chap. 8].

- $M_2 = (E, \mathcal{I}_2)$ is a partition matroid, with $\mathcal{I}_2 = \{F \subseteq E : |\mathcal{L}_i(F)| \leq \text{OPT} \text{ for every } 1 \leq i \leq q\}$.

We complete the resulting forest into a spanning tree in an arbitrary way, noting that this augmentation leaves the objective value unchanged. ■

Theorem 5.4. *UL-min-max spanning tree can be solved to optimality in polynomial time.*

Proof. The algorithm for this version is nearly identical to the one given for UL-max-min spanning tree; however, an important remark is in place. After we "guess" OPT and compute a maximum cardinality intersection $F \subseteq E$ of M_1 and M_2, there is no need to complete the subgraph induced by F into a spanning tree, implying that its objective value remains unchanged. This claim follows from observing that $|F| = |V| - 1$, since the edge set of the optimal spanning tree forms a feasible solution to the matroid intersection problem we solve. ■

5.3 WL-min-max spanning tree: A logarithmic approximation

In what follows, we show that a matroid intersection algorithm is not only a useful tool for solving the unweighted version to optimality; rather, it can also be applied to approximate the weighted min-max version.

The algorithm. For ease of exposition, we assume without loss of generality that an estimator of the optimum $W \in [\text{OPT}, 2 \cdot \text{OPT}]$ is known in advance. Otherwise, for every $0 \leq k \leq \lceil \log \frac{n w_{\max}}{w_{\min}} \rceil$, we can test $2^k w_{\min}$ as a candidate value and return the best solution found, where $w_{\min}$ and $w_{\max}$ denote the minimum and maximum non-zero edge weights, respectively.

1. Delete all edges of weight greater than W, and define a partition of the undeleted edges as follows:
   (a) For every $1 \leq i \leq q$ and $0 \leq k \leq \lfloor \log n \rfloor$, let $\mathcal{E}_{i,k}$ be the set of edges e with $\mathcal{L}(e) = c_i$ and $w(e) \in (\frac{W}{2^{k+1}}, \frac{W}{2^k}]$.
   (b) In addition, let $\mathcal{E}_{\text{free}}$ be the set of remaining edges (of weight at most $\frac{W}{n}$).
2. By applying a matroid intersection algorithm, find a spanning tree T that picks at most 2^{k+1} edges from each $\mathcal{E}_{i,k}$ and any number of edges from $\mathcal{E}_{\text{free}}$. Return T.

Note that the suggested algorithm is well-defined. To establish this claim, it is sufficient to show that a spanning tree satisfying the constraints of step 2 indeed exists. It is easy to verify that all edges of the optimal tree T^* survive step 1 and that $|T^* \cap \mathcal{E}_{i,k}| \leq 2^{k+1}$, or otherwise there is a color c_i from which T^* picks edges of total weight strictly greater than $W \geq \text{OPT}$.

Theorem 5.5. *The edges picked by T from any given color have an overall weight of at most $(4\lfloor \log n \rfloor + 6)\text{OPT}$.*

Proof. Consider some color c_i. Then,

$$\begin{aligned}
\sum_{e \in \mathcal{L}_i(T)} w(e) &= \sum_{k=0}^{\lfloor \log n \rfloor} \sum_{e \in T \cap \mathcal{E}_{i,k}} w(e) + \sum_{e \in \mathcal{L}_i(T \cap \mathcal{E}_{\text{free}})} w(e) \\
&\leq \sum_{k=0}^{\lfloor \log n \rfloor} \left(|T \cap \mathcal{E}_{i,k}| \cdot \max_{e \in T \cap \mathcal{E}_{i,k}} w(e) \right) + |T \cap \mathcal{E}_{\text{free}}| \cdot \max_{e \in T \cap \mathcal{E}_{\text{free}}} w(e) \\
&\leq \sum_{k=0}^{\lfloor \log n \rfloor} 2^{k+1} \frac{W}{2^k} + (n-1)\frac{W}{n} \\
&\leq (2\lfloor \log n \rfloor + 3)W \\
&\leq (4\lfloor \log n \rfloor + 6)\text{OPT} .
\end{aligned}$$

The second inequality holds since $|T \cap \mathcal{E}_{i,k}| \leq 2^{k+1}$ for every $0 \leq k \leq \lfloor \log n \rfloor$, and since $|T \cap \mathcal{E}_{\text{free}}| \leq n - 1$. The last inequality follows from the assumption $W \leq 2 \cdot \text{OPT}$. ∎

6 Arbitrary Number of Colors: Labeled Bottleneck Perfect Matchings

In Theorem 2.1 we proved that both versions of weighted labeled bottleneck perfect matching are NP-hard in bicolored graphs. In this section, we provide hardness of approximation results.

Theorem 6.1. *UL-max-min perfect matching is not approximable at all, unless P=NP, even in planar bipartite graphs.*

Proof. We describe a reduction from $(3, B2)$-SAT. An instance $(\mathcal{C}, X)$ of this problem consists of a collection $\mathcal{C} = (C_1, \ldots, C_m)$ of m clauses of size exactly 3 over the set $X = \{x_1, \ldots, x_n\}$ of boolean variables, such that every variable occurs exactly twice positively and twice negatively. The objective is to decide whether there exists a truth assignment that satisfies all clauses. This problem was shown to be NP-complete by Berman, Karpinski and Scott [6, Thm. 1].

Given an instance $I = (\mathcal{C}, X)$ of $(3, B2)$-SAT, as described above, we construct an instance $I' = (G, \mathcal{L})$ of UL-max-min perfect matching, where $G = (V, E)$ is a collection of n vertex-disjoint cycles of length 4 and $\mathcal{L}(E) = \{c_1, \ldots, c_m\}$:

1. Each variable x_i has its own cycle of length 4, to which we refer as $H(x_i)$.

2. Now let $\{C_{j_1}, C_{j_2}\} \subseteq \mathcal{C}$ be the clauses in which x_i appears in positive form, and $\{C_{j_3}, C_{j_4}\} \subseteq \mathcal{C}$ be those in which it appears in negative form. Then, the edges of $H(x_i)$ are colored by $c_{j_1}, c_{j_3}, c_{j_2}, c_{j_4}$ (clockwise).

This construction guarantees that the edges with colors corresponding to clauses containing the literal x_i form a matching in $H(x_i)$, and so do the edges corresponding to clauses containing $\bar{x}_i$. Consequently, the original instance I is satisfiable if and only if there exists a perfect matching M in G using each color at least once. We conclude that it is NP-hard to distinguish between UL-max-min perfect matching instances of objective values 0 and 1. ∎

We now present an approximation-preserving reduction from UL-min-max directed s-t path to UL-min-max perfect matching. By combining this result with Theorem 4.1, we derive new inapproximability results for the latter problem.

Theorem 6.2. *UL-min-max perfect matching is not* $(2-\epsilon)$*-approximable for any fixed* $\epsilon > 0$*, unless P=NP, even in bipartite graphs.*

Proof. Since Theorem 4.1 establishes, for any fixed $\epsilon > 0$, a lower bound of $2 - \epsilon$ on the approximability of UL-min-max directed s-t path, it is sufficient to show that a ρ-approximation for this problem will be implied by the existence of a ρ-approximation for UL-min-max perfect matching in bipartite graphs.

Given an instance $I = (G, s, t, \mathcal{L})$ of UL-min-max directed s-t path, we transform it into an instance $I' = (G', \mathcal{L}')$ of UL-min-max perfect matching, where G' is an undirected bipartite graph. The latter is created by splitting each vertex $v \in V \setminus \{s,t\}$ to v_{in} and v_{out}, followed by adding the edge $(v_{\text{in}}, v_{\text{out}})$. We also cancel the orientation, and replace each arc by a corresponding edge of identical color. Finally, each new edge of the form $(v_{\text{in}}, v_{\text{out}})$ is given a unique color, c'_v.

We first claim that $\text{OPT}(I') \leq \text{OPT}(I)$. To this end, consider a directed path $\langle s = v^1, v^2, \ldots, v^k = t\rangle$ that constitutes an optimal solution to I. Then, by picking the union of $\{(v^i_{\text{out}}, v^{i+1}_{\text{in}}) : 1 \leq i \leq k-1\}$ and $\{(v_{\text{in}}, v_{\text{out}}) : v \in V \setminus \{v_1, \ldots v_k\}\}$ we obtain a perfect matching in G' whose objective value is exactly $\text{OPT}(I)$. Consequently, it remains to show that, given a perfect matching M in G', we can efficiently find a directed s-t path in G whose objective value does not exceed that of M. The reduction described above guarantees that there exists a collection of vertices $V_M \subseteq V$ such that M consists of an edge set $\{(v_{\text{in}}, v_{\text{out}}) : v \in V_M\}$ and an edge set $\{(u_{\text{out}}, v_{\text{in}}) : u, v \in V \setminus V_M\}$. It is not difficult to verify that the latter set induces a directed s-t path in G (and quite possibly one or more directed cycles). The objective value of this path is upper bounded by that of M. ∎

7 Arbitrary Number of Colors: Labeled Bottleneck *s-t* Cuts

In Theorem 2.2, we proved that WL-min-max s-t cut is strongly NP-hard. However, careful examination reveals that the instances resulting from our reduction are not necessarily planar. Since the maximum s-t cut problem is polynomially solvable in planar graphs [24, 16], it is of interest to show that the former problem remains computationally intractable in such graphs. We also provide new inapproximability results for UL-max-min s-t cut.

Theorem 7.1. *UL-min-max s-t cut is not* $(2-\epsilon)$*-approximable in planar graphs for any fixed* $\epsilon > 0$*, unless P=NP.*

Proof. We propose a reduction from the set packing problem, whose definition is given in Theorem 4.1. Let $I = (\mathcal{S}, X, k)$ be a set packing instance, where $\mathcal{S} = \{S_1, \ldots, S_m\}$ is a family of sets over a ground set $X = \{x_1, \ldots, x_n\}$. We define an instance of UL-min-max s-t cut, consisting of a planar graph $G = (V, E)$ whose edges are given colors from the set $\{c_0, c_1, \ldots, c_n\}$. The graph G is initially a collection of k interior-disjoint paths $P_1, \ldots, P_k$ connecting s to t, each of length m. Then, for each path P_i, we perform the following modification: For every $1 \leq j \leq m$, the j-th edge $e_j = (x, y)$ of P_i is replaced by a collection of $|S_j|$ interior-disjoint paths of length 2 connecting x to y; letting $S_j = \{x_{j_1}, \ldots, x_{j_p}\}$, we spread the colors $\{c_{j_1}, \ldots, c_{j_p}\}$ on the first edges of these $|S_j|$ paths, and set the color of their second edges to c_0. The theorem follows from the observation that I has a set packing of size at least k if and only if the objective value of the newly defined UL-min-max s-t cut instance is at most 1. ∎

Theorem 7.2. *UL-max-min s-t cut is not approximable at all in planar multigraphs, unless P=NP.*

Proof. We describe a reduction from the *satisfiability* problem, whose instances consist of a collection $\mathcal{C} = (C_1, \dots, C_m)$ of clauses over a set $X = \{x_1, \dots, x_n\}$ of boolean variables. The goal is to decide whether there is a truth assignment that satisfies all clauses. This problem is known to be NP-complete [15] (problem [LO1], p. 259).

We construct an instance of UL-max-min s-t cut, where $G = (V, E)$ and $\mathcal{L}(E) = \{c_1, \dots, c_m\}$. The set of vertices is $V = \{s, t, v_1, \dots, v_n\}$, and there are n_i (respectively, $\bar{n}_i$) parallel edges joining s (respectively, t) and v_i, where n_i (respectively, $\bar{n}_i$) is the number of x_i (respectively, $\bar{x}_i$) occurrences. Finally, the colors of these edges are given by the clauses containing x_i (respectively, $\bar{x}_i$). It is not difficult to verify that I is satisfiable if and only if there exists an s-t cut in G whose objective value is at least 1. ■

Acknowledgments. We thank Shlomi Rubinstein for suggesting the reduction in Theorem 4.1, which simplified our original proof.

References

[1] A. A. Ageev and M. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.

[2] H. Aissi, C. Bazgan, and D. Vanderpooten. Approximation complexity of min-max (regret) versions of shortest path, spanning tree, and knapsack. In *Proceedings of the 13th Annual European Symposium on Algorithms*, pages 862–873, 2005.

[3] H. Aissi, C. Bazgan, and D. Vanderpooten. Complexity of the min-max (regret) versions of cut problems. In *Proceedings of the 16th International Symposium on Algorithms and Computation*, pages 789–798, 2005.

[4] H. Aissi, C. Bazgan, and D. Vanderpooten. Pseudo-polynomial algorithms for min-max and min-max regret problems. In *Proceedings of the 5th International Symposium on Operations Research and its Applications*, pages 171–178, 2005.

[5] I. Averbakh and O. Berman. Categorized bottleneck-minisum path problems on networks. *Operations Research Letters*, 16(5):291–297, 1994.

[6] P. Berman, M. Karpinski, and A. D. Scott. Approximation hardness of short symmetric instances of MAX-3SAT. Electronic Colloquium on Computational Complexity, Report TR03-049, 2003.

[7] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming Series B*, 98(1):49–71, 2003.

[8] H. Broersma and X. Li. Spanning trees with many or few colors in edge-colored graphs. *Discussiones Mathematicae Graph Theory*, 17(2):259–269, 1997.

[9] H. Broersma, X. Li, G. Woeginger, and S. Zhang. Paths and cycles in colored graphs. *Australasian Journal on Combinatorics*, 31:299–311, 2005.

[10] T. Brüggemann, J. Monnot, and G. Woeginger. Local search for the minimum label spanning tree problem with bounded color classes. *Operations Research Letters*, 31(3):195–201, 2003.

[11] R.-S. Chang and S.-J. Leu. The minimum labeling spanning trees. *Information Processing Letters*, 63(5):277–282, 1997.

[12] K. Dhamdhere, V. Goyal, R. Ravi, and M. Singh. How to pay, come what may: Approximation algorithms for demand-robust covering problems. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 367–378, 2005.

[13] M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum*, 22:425–460, 2000.

[14] M. Ehrgott and X. Gandibleux, editors. *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Survey*, volume 52 of *International Series in Operations Research and Management Science*. Kluwer Academic Publishers, 2002.

[15] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[16] F. O. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM Journal on Computing*, 4(3):221–225, 1975.

[17] R. Hassin, J. Monnot, and D. Segev. Approximation algorithms and hardness results for labeled connectivity problems. In *Proceedings of the 31st International Symposium on Mathematical Foundations of Computer Science*, pages 480–491, 2006.

[18] A. Itai, M. Rodeh, and S. L. Tanimoto. Some matching problems for bipartite graphs. *Journal of the ACM*, 25(4):517–525, 1978.

[19] D. R. Karger, R. Motwani, and G. D. S. Ramkumar. On approximating the longest path in a graph. *Algorithmica*, 18(1):82–98, 1997.

[20] A. V. Karzanov. Maximum matchings of given weight in complete and complete bipartite graphs. *Kibernetika*, 1:7–11, 1987. English translation in *CYBNAW*, 23: 8–13.

[21] P. Kouvelis and G. Yu. *Robust Discrete Optimization and its Applications*. Kluwer Academic Publishers, 1997.

[22] E. L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

[23] J. Monnot. The labeled perfect matching in bipartite graphs. *Information Processing Letters*, 96(3):81–88, 2005.

[24] G. I. Orlova and Y. G. Dorfman. Finding the maximum cut in a graph. *Engineering Cybernetics*, 10(3):502–506, 1972.

[25] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 86–92, 2000.

[26] A. P. Punnen. Traveling salesman problem under categorization. *Operations Research Letters*, 12(2):89–95, 1992.

[27] A. P. Punnen. On bottleneck assignment problems under categorization. *Computers and Operations Research*, 31(1):151–154, 2004.

[28] M. B. Richey and A. P. Punnen. Minimum perfect bipartite matchings and spanning trees under categorization. *Discrete Applied Mathematics*, 39(2):147–153, 1992.

[29] E. L. Ulungu and J. Teghem. Multi-objective combinatorial optimization problems: A survey. *Journal of Multi-Criteria Decision Analysis*, 3:83–104, 1994.

[30] D. J. White. A bibliography on the applications of mathematical programming multiple-objective methods. *Journal of the Operational Research Society*, 41(8):669–691, 1990.

[31] T. Yi, K. G. Murty, and C. Spera. Matchings in colored bipartite networks. *Discrete Applied Mathematics*, 121(1-3):261–277, 2002.