Local search for the minimum label spanning tree problem with bounded color classes

Tobias Brüggemann * Jérôme Monnot † Gerhard J. Woeginger ‡

Abstract

In the Minimum Label Spanning Tree problem, the input consists of an edge-colored undirected graph, and the goal is to find a spanning tree with the minimum number of different colors. We investigate the special case where every color appears at most r times in the input graph. This special case is polynomially solvable for $r = 2$, and NP-complete and APX-complete for any fixed $r \geq 3$.

We analyze local search algorithms that are allowed to switch up to k of the colors used in a feasible solution. We show that for $k = 2$ any local optimum yields an $(r + 1)/2$-approximation of the global optimum, and that this bound is tight. For every $k \geq 3$, there exist instances for which some local optima are a factor of $r/2$ away from the global optimum.

Keywords: Graph algorithms; approximation algorithms; combinatorial optimization; local search; complexity; APX-completeness.

1 Introduction

In the *Minimum Label Spanning Tree problem* (MinLST, for short), we are given a simple, connected, undirected graph $G = (V, E)$ without loops on n vertices. The edges in E are colored (or labeled) with the colors $c_1, c_2, \ldots, c_q$. For $i = 1, \ldots, q$ we denote by $E(c_i) \subseteq E$ the set of edges with color c_i. The goal in MinLST is to find a spanning tree in G that uses the minimum number of colors. An equivalent formulation of MinLST asks to find a smallest cardinality subset $C \subseteq \{c_1, c_2, \ldots, c_q\}$ of the colors, such that the subgraph induced by the edge sets $E(c_i)$ with $c_i \in C$ is connected and touches all vertices in V.

Motivated by certain applications in communication network design, Chang & Leu [4] introduced problem MinLST in 1997 and proved that it is NP-complete. Krumke & Wirth [9] formulated a greedy algorithm for MinLST, and showed that its worst case performance ratio is at most $2 \ln n + 1$. Moreover, [9] proved that no polynomial time approximation algorithm

*`t.brueggemann@math.utwente.nl`. Department of Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

†`monnot@lamsade.dauphine.fr`. CNRS-LAMSADE, UMR 7024, Université Paris-Dauphine, France.

‡`g.j.woeginger@math.utwente.nl`. Department of Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

for MinLST can have a worst case performance ratio $(1-\varepsilon)\ln n$, for any $\varepsilon > 0$. Wan, Chen & Xu [16] provided a better analysis of the greedy algorithm in [9]; they showed that its worst case performance ratio is at most $\ln(n-1)+1$.

Results of this paper. In this paper, we study the special case MinLST_r of MinLST in which every color occurs at most r times ($r \geq 2$) on the edges of G. For $r = 2$, this special case is equivalent to the Graphic Matroid Parity problem, and therefore can be solved in polynomial time (see Observation 5.1 in Section 5). For every $r \geq 3$, this special case MinLST_r is NP-complete and APX-complete; hence, for $r \geq 3$ this special case does not possess a polynomial time approximation scheme unless P=NP (see Theorem 5.2 in Section 5).

In Section 2 we introduce a family of local search algorithms that are based on the so-called *k-switch* neighborhoods, where $k \geq 1$ is an integer. Sloppily speaking, a k-switch replaces up to k of the colors used in a feasible solution by other colors. Local optima for the k-switch neighborhoods can be computed in polynomial time. In Sections 3 and 4 we then discuss how well local optima for k-switch perform in comparison to global optima: For $k = 2$, any local optimum yields an $(r+1)/2$-approximation of the global optimum, and this bound of $(r+1)/2$ is best possible. For every $k \geq 3$, there exist instances for which some local optimum is a factor of roughly $r/2$ away from the global optimum. Hence, from the worst case point of view there is almost no profit in moving from the (small) 2-switch neighborhood to the (much bigger) k-switch neighborhoods with $k \geq 3$.

In studying the worst case quality of local optima of local search algorithms for combinatorial problems, we follow the line of research of Finn & Horowitz [6] (for multiprocessor scheduling), Hurkens & Schrijver [8] (for set packing), Lu & Ravi [11] (for maximum-leaf spanning trees), Ausiello & Protasi [3] (for complexity aspects), Arkin & Hassin [2] (for weighted set packing), and Schuurman & Vredeveld [14] (for multiprocessor scheduling). For more information on this area, we refer the reader to the Ph.D. thesis [15] of Vredeveld.

2 The k-switch neighborhoods

Any spanning tree T for problem MinLST can be represented by the set $C(T) \subseteq \{c_1, \ldots, c_q\}$ of colors used in T. In this section, we prefer to work with color sets. A color set C is *feasible* if and only if the corresponding set of edges is connected and touches all vertices in the graph.

Definition 2.1 *Let $k \geq 1$ be an integer, and let C_1 and C_2 be two feasible color sets for some instance of* MinLST*. Then the set C_2 is in the k-switch neighborhood k-*Switch*(C_1) of the set C_1, if and only if*

$$|C_1 - C_2| \;\leq\; k \quad \textit{and} \quad |C_2 - C_1| \;\leq\; k. \tag{1}$$

In other words, we can get the color set C_2 from the color set C_1 by first removing up to k colors from C_1, and then adding up to k colors to it.

As usual with neighborhood structures, we may build a local search algorithm around the k-switch neighborhood:

Start with an arbitrary feasible color set C. As long as there exists a feasible color set C' in k-SWITCH(C) with $|C'| < |C|$, replace the old set C by the better set C'.

Eventually, the local search algorithm will terminate in a *local optimum* C: For this local optimum C, any set C' in k-SWITCH(C) will satisfy $|C'| \geq |C|$. In a slight abuse of notation, we will say that a spanning tree is a local optimum for the k-switch neighborhood if and only if its associated color set $C(T)$ is a local optimum for the k-switch neighborhood.

The following observation shows that for every fixed value of k, a local optimum for the k-switch neighborhood can be determined in polynomial time.

Observation 2.2 *For any $k \geq 1$, a local optimum with respect to the k-switch neighborhood can be computed in $O(n^{3k+3})$ time.*

Proof. Without loss of generality, we assume that the starting point of the local search algorithm contains at most $n-1$ colors. By equation (1) any neighborhood set k-SWITCH(C) contains at most $O(|C|^k q^k)$ feasible sets. Since $|C| \leq n-1$ and since $q \leq |E| \leq n^2$, we conclude that $|k\text{-SWITCH}(C)| = O(n^{3k})$. Within $O(n^2)$ time, we can determine whether a color set in the neighborhood is feasible and we can determine its objective value. Hence, one replacement step in the local search takes only $O(n^{3k+2})$ time.

Since the possible objective values are integers in the range from 1 up to $n-1$, the local search terminates after at most $n-2$ replacement steps. ■

In the following two sections, we will analyze the quality of local optima with respect to k-switch neighborhoods for $k \geq 2$. The case $k = 1$ is trivial.

Observation 2.3 *Let $r \geq 2$ be an integer. For any instance of* MINLST$_r$*, a local optimum with respect to the 1-switch neighborhood gives an r-approximation of the global optimum. This bound is tight.* ■

3 Local optima for the 2-switch neighborhood

In this section, we provide a complete worst case analysis of local optima with respect to the 2-switch neighborhood: Every local optimum yields an $(r+1)/2$-approximation of the global optimum (Theorem 3.1), and this bound is best possible (Theorem 3.2).

Theorem 3.1 *For any integer $r \geq 2$ and for any instance G of* MINLST$_r$*, the objective value of any local optimum with respect to the 2-switch neighborhood is at most a factor of $(r+1)/2$ above the optimal objective value.*

Proof. Suppose for the sake of contradiction that the statement is false, and consider a counterexample $G = (V, E)$ with the smallest number of edges. Let $T^* = (V, E^*)$ be an optimal spanning tree for G, and let $T^+ = (V, E^+)$ be a locally optimal tree with respect to the 2-switch neighborhood. Let $C^* = C(T^*)$ and $C^+ = C(T^+)$ denote the corresponding color sets with

$$|C^+| \; > \; \frac{r+1}{2}\,|C^*|. \tag{2}$$

We observe that in a smallest counterexample, $C^* \cap C^+ = \emptyset$ must hold: If there is a color $i \in C^* \cap C^+$, then we can contract all edges with this color i in G, and get a smaller instance where the global and local optimum both use one color less. Since this smaller instance still satisfies the inequality (2), we would have found a smaller counterexample. Hence, $C^* \cap C^+ = \emptyset$. Moreover, a smallest counterexample satisfies $E^* \cup E^+ = E$.

Let n denote the number of vertices in G. A color is called *singleton* if it shows up on exactly one edge of G. Let ℓ denote the number of singleton colors in C^+, and let $e_1, \ldots, e_\ell$ be an enumeration of the corresponding edges in T^+. Consider the $\ell + 1$ subtrees $T_1^+, \ldots, T_{\ell+1}^+$ that result from removing the ℓ edges $e_1, \ldots, e_\ell$ from T^+.

Suppose that there exists some color i, such that the edges with color i connect more than two of these subtrees $T_1^+, \ldots, T_{\ell+1}^+$ to each other. Then one could add color i to C^+, remove an appropriate pair of singleton colors from C^+, and get another feasible color set C^- with strictly better objective value. Since the set C^- is in the 2-switch neighborhood of the local optimum C^+, we arrive at a contradiction. Therefore, every color connects at most two of these $\ell + 1$ subtrees to each other. But this implies that also the global optimum must spend at least ℓ colors on connecting the corresponding $\ell + 1$ vertex sets to each other, and we get

$$|C^*| \;\geq\; \ell. \tag{3}$$

Since a spanning tree has $n - 1$ edges, and since every color occurs at most r times, we furthermore have that

$$|C^*| \;\geq\; \frac{n-1}{r}. \tag{4}$$

Now let us estimate the number of colors in the local optimum T^+: There are ℓ edges in T^+ with the ℓ singleton colors. Every non-singleton color i in T^+ occurs at least twice on the edges of G. Since $C^+ \cap C^* = \emptyset$, the color i cannot show up in T^*, and since $E^* \cup E^+ = E$, all edges with color i are contained in T^+. This yields that there are at most $(n - 1 - \ell)/2$ non-singleton colors in C^+. Hence,

$$|C^+| \;\leq\; \ell + \frac{1}{2}(n-1-\ell) \;=\; \frac{1}{2}(n-1) + \frac{1}{2}\ell \;\leq\; \frac{r}{2}\,|C^*| + \frac{1}{2}\,|C^*| \;=\; \frac{r+1}{2}\,|C^*|. \tag{5}$$

Here we used (3) and (4). The inequality (5) blatantly contradicts our initial assumption (2). This contradiction completes the proof of the theorem. ∎

Theorem 3.2 *For any integer $r \geq 2$, there exist an instance G of* MinLST$_r$ *and a spanning tree T for G that is a local optimum with respect to the 2-switch neighborhood, such that the objective value of T is $(r+1)/2$ above the optimal objective value.*

Proof. Consider the graph G with vertices v_0, $x_0, \ldots, x_{r-1}$, and $y_0, \ldots, y_{r-1}$. There is an edge from v_0 to every other vertex. Moreover, the vertices $x_0, \ldots, x_{r-1}$ (in this ordering) induce a cycle and the vertices $y_0, \ldots, y_{r-1}$ (in this ordering) induce a cycle. There are $r + 3$ colors: For $i = 1, \ldots, r - 1$ the two edges $[x_{i-1}, x_i]$ and $[y_{i-1}, y_i]$ have color i. The edge $[v_0, x_0]$ has color r, and the edge $[v_0, y_0]$ has color $r + 1$. The edge $[x_0, x_{r-1}]$ and all edges from v_0 to $x_1, \ldots, x_{r-1}$ have color $r + 2$; the edge $[y_0, y_{r-1}]$ and all edges from v_0 to $y_1, \ldots, y_{r-1}$ have color $r + 3$.

Then the edges with colors $r + 2$ and $r + 3$ form a spanning tree with 2 colors. The edges with colors $1, 2, \ldots, r + 1$ form a spanning tree with $r + 1$ colors that is a local optimum with respect to the 2-switch neighborhood. See Figure 1 for an illustration. ∎

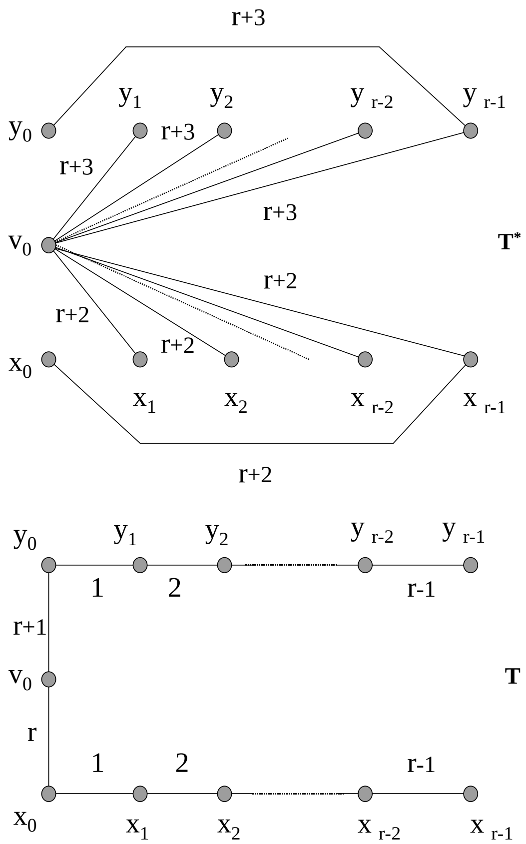

Figure 1: A global optimum and a local optimum for 2-switch in the proof of Theorem 3.2.

4 Local optima for the k-switch neighborhood

In this section we will show that from the worst case point of view, it will not help a lot if we move from the 2-switch neighborhood to the bigger k-switch neighborhoods with $k \geq 3$: There always will be instances for which a local optimum for a k-switch neighborhood is a factor of $r/2$ away from the global optimum.

Lemma 4.1 *For any $k \geq 2$ and for any $r \geq 3$, there exist arbitrarily large undirected, simple graphs $H = (V_H, E_H)$ that satisfy the following three properties.*

- *H is r-regular (i.e., every vertex in H has degree exactly r)*
- *H has girth at least k (i.e., the shortest cycle in H has length at least k)*
- *H contains a perfect matching $\mathcal{M}$.*

Proof. By applying a result of Erdős & Sachs [5], Hurkens & Schrijver [8] construct bipartite r-regular graphs of girth at least k. It is well-known that every regular bipartite has a perfect matching. By taking many disjoint copies of the graph in [8], we get arbitrarily large graphs with the desired three properties. ■

Now consider a graph $H = (V_H, E_H)$ as described in Lemma 4.1. Denote $|V_H| = 2h$, and let $w_1, w_2, \ldots, w_{2h}$ be an enumeration of the vertices in V_H such that for $i = 1, \ldots, h$ the vertices w_i and w_{h+i} form an edge in the perfect matching $\mathcal{M}$. From H we will construct an instance graph $G = (V_G, E_G)$ for $\textsc{MinLST}_r$. The vertex set V_G consists of $2rh + 2$ vertices. There are two special vertices u_1 and u_2, and for $i = 1, \ldots, 2h$ there is a group G_i of r vertices $v_{i,j}$ with $1 \le j \le r$. The edges in G are defined as follows.

- There is an edge between the two special vertices u_1 and u_2.
- The special vertex u_1 is connected to all vertices $v_{i,j}$ with $1 \le i \le 2h$ and $1 \le j \le r$.
- The special vertex u_2 is connected to all vertices $v_{i,1}$ with $1 \le i \le 2h$.
- Every group G_i induces a path through the vertices $v_{i,1}, v_{i,2}, \ldots, v_{i,r}$ in exactly this ordering.

The edge colors are defined as follows.

(C1) The edge $[u_1, u_2]$ has color c^*.

(C2) For $i = 1, \ldots, 2h$ every edge between the special vertex u_1 and the group G_i has color $c(i)$. We say that color $c(i)$ corresponds to the vertex w_i in H.

(C3) For $i = 1, \ldots, h$ the two edges $[u_2, v_{i,1}]$ and $[u_2, v_{h+i,1}]$ have color $\overline{c}(i)$. We say that color $\overline{c}(i)$ corresponds to the edge $[w_i, w_{h+i}]$ in $\mathcal{M}$.

(C4) For every edge $[w_a, w_b] \in E_H - \mathcal{M}$, there is a corresponding color $c(a, b)$. This color $c(a, b)$ shows up exactly once on the path induced by group G_a and exactly once on the path induced by group G_b. Since w_a is incident to $r - 1$ edges in $E_H - \mathcal{M}$, this yields exactly $r - 1$ colors for the $r - 1$ edges in the path induced by G_a. The exact assignment of colors $c(a, b)$ to edges in G_a is irrelevant for our arguments; an arbitrary assignment will work.

We say that color $c(a, b)$ corresponds to the edge $[w_a, w_b]$ in $E_H - \mathcal{M}$.

Note that the color c^* in (C1) occurs once, that every color $c(i)$ in (C2) occurs exactly r times, and that every color $\overline{c}(j)$ in (C3) and every color $c(a, b)$ in (C4) occurs exactly twice. Hence, we have indeed constructed an instance of $\textsc{MinLST}_r$.

Lemma 4.2 *The optimal objective value of instance G is at most $2h + 1$.*

Proof. The edge $[u_1, u_2]$ of color c^* in (C1) together with the edges with colors $c(i)$ with $1 \le i \le 2h$ in (C2) form a spanning tree for G. ■

Lemma 4.3 *There exists a spanning tree T for G*

(a) that has objective value $rh+1$, and

(b) that is a local optimum with respect to the k-switch neighborhood.

Proof. We let T consist of all color classes in (C1), (C3), and (C4). This yields a spanning tree with $rh+1$ colors that satisfies property (a). It remains to prove that T also satisfies the property in (b). Suppose for the sake of contradiction that there is an improving k-switch for T. This k-switch removes $x \le k$ colors from T, and it adds $y \le x-1$ colors to T. We make two observations:

- If the switch removes the color $\overline{c}(i)$ (that corresponds to the edge $[w_i, w_{h+i}]$ in H), then it must simultaneously add the two colors $c(i)$ and $c(h+i)$ (that correspond to the vertices w_i and w_{h+i} in H). Otherwise, one of the groups G_i and G_{h+i} will be separated from the rest of the graph.

- If the switch removes the color $c(a,b)$ (that corresponds to the edge $[w_a, w_b]$ in E_H) then it must simultaneously add the two colors $c(a)$ and $c(b)$ (that correspond to the vertices w_a and w_b in H). Otherwise, some vertices in group G_a or G_b will be isolated from the rest of the graph.

To summarize, whenever the switch removes a color in (C3) or (C4) that corresponds to an edge in H, then it must simultaneously add the two colors in (C2) that correspond to the vertices of this edge in H.

Let $Y \subset V_H$ denote the vertices in H that correspond to the $|Y| \le k-1$ colors from (C2) that the switch adds to T. Then the switch can remove the single color c^*, and it can remove the colors in (C3) and (C4) that correspond to edges induced by vertices in Y. Since H has girth k, the subgraph of H induced by Y is cycle-free. Hence it is a forest, and induces at most $|Y|-1$ edges in H. But this means that the k-switch adds $|Y|$ colors, while it removes at most $|Y|$ colors; hence, it is not an improving k-switch. This contradiction completes the proof. ■

Theorem 4.4 *For any integer $k \ge 2$, for any integer $r \ge 2$, and for any real $\varepsilon > 0$, there exist an instance G of* MinLST$_r$ *and a spanning tree T for G that is a local optimum with respect to the k-switch neighborhood, such that the objective value of T is at least $r/2 + \varepsilon$ above the optimal objective value.*

Proof. Lemmas 4.2 and 4.3 yield a ratio $(rh+1)/(2h+1)$ between the objective values of the local and of the global optimum. As h tends to infinity, this ratio tends to $r/2$. ■

5 Complexity and in-approximability

In this section, we first explain why problem MinLST$_2$ is easy, and then prove that problem MinLST$_3$ is difficult. MinLST$_3$ is APX-complete, which implies that it does not have a polynomial time approximation scheme unless P=NP.

Observation 5.1 *For $r = 2$, the problem* MinLST$_r$ *is polynomially solvable.*

Proof. The problem MinLST$_2$ is essentially equivalent to the *Graphic Matroid Parity problem*; see for instance Lovaśz & Plummer [10] and Gabow & Stallman [7]: In the Graphic Matroid Parity problem, we are given a graph $G' = (V', E')$ and a partition of the edge set E' into disjoint pairs of edges $\{f, f'\}$. The goal is to find a forest F with the maximum number of edges, such that $f \in F$ holds if and only if $f' \in F$ for all pairs $\{f, f'\}$ in the partition.

In problem MinLST$_2$, the edge pairs $\{f, f'\}$ are the pairs of edges with the same color. The goal is to use as many colors twice as possible, and then to connect the resulting forest to a tree by adding color classes of cardinality one. ■

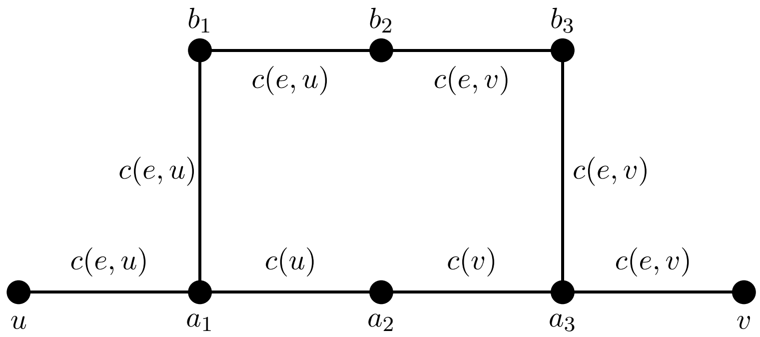

Figure 2: The gadget $Z(u, v)$ as used in the proof of Theorem 5.2.

Theorem 5.2 *For $r \geq 3$ the problem* MinLST$_r$ *is APX-complete even if the input graph G is restricted to be bipartite and of maximum degree* 3.

Proof. The proof will be done via an approximation preserving L-reduction (cf. Papadimitriou & Yannakakis [13]) from the vertex cover problem in 3-regular connected graphs, VC3 for short: An instance of VC3 consists of a connected 3-regular graph $H = (V_H, E_H)$, and the goal is to find a minimum cardinality vertex cover W for H, that is, a subset $W \subseteq V_H$ that intersects every edge in E_H. Alimonti & Kann [1] proved that problem VC3 is APX-hard. This implies that there is some small $\varepsilon > 0$ such that the existence of a polynomial time approximation algorithm with performance guarantee $1 + \varepsilon$ would imply P=NP.

We consider an arbitrary instance $H = (V_H, E_H)$ of problem VC3, with $|V_H| = 2h$ and $|E_H| = 3h$. We construct a corresponding instance $G = (V_G, E_G)$ of problem MinLST$_3$ from it: For every vertex $v \in V_H$, there is a corresponding color $c(v)$. For every edge $e = [u, v] \in E_H$, there are two corresponding colors $c(e, u)$ and $c(e, v)$. G results from H by replacing every edge $e = [u, v] \in E_H$ by a copy of the gadget $Z(u, v)$ depicted in Figure 2. This gadget $Z(u, v)$ has six new vertices a_1, a_2, a_3 and b_1, b_2, b_3. The edges and their colors are defined as follows:

- The edges $[u, a_1]$, $[a_1, b_1]$, $[b_1, b_2]$ are of color $c(e, u)$.

- The edges $[b_2, b_3]$, $[b_3, a_3]$, $[a_3, v]$ are of color $c(e, v)$.
- The edge $[a_1, a_2]$ has color $c(u)$.
- The edge $[a_2, a_3]$ has color $c(v)$.

This completes the description of the graph G. Note that the colors $c(e, u)$ and $c(e, v)$ only show up within the gadget $Z(u, v)$, and there they are used three times. Any color $c(v)$ shows up once in the three gadgets that correspond to the three edges incident to v in H. Hence, we have indeed constructed an instance of MinLST$_3$. Moreover, the graph G clearly is bipartite and of maximum degree 3.

Since every vertex in H is incident to exactly three edges, the optimal vertex cover W^* for H must contain at least $|E_H|/3 = h$ vertices. Since there are altogether $|V_H| + 2|E_H| = 8h$ colors in G, the optimal spanning tree T^* for G uses at most $8h$ colors. Therefore,

$$|C(T^*)| \;\leq\; 8\,|W^*|. \tag{6}$$

Since in every gadget $Z(u, v)$ the vertex b_1 (respectively, the vertex b_3) is only adjacent to edges of color $c(e, u)$ (respectively, to edges of color $c(e, v)$), all these colors $c(e, u)$ and $c(e, v)$ must be used in any spanning tree of G. Moreover, in order to connect the vertex a_2 to the rest of the tree, any spanning tree must use at least one of the two colors $c(u)$ and $c(v)$. Based on these observations, it is easy to translate a spanning tree T for G into a corresponding vertex cover W_T for H: W_T consists of the vertices $v \in V_H$ for which the color $c(v)$ shows up in the tree T. Consequently, $|W_T| = |C(T)| - 6h$. By similar reasoning, we get that the optimal spanning tree T^* of G and the optimal vertex cover W^* of H satisfy $|W^*| = |C(T^*)| - 6h$. This implies that for any spanning tree T, $|W_T| - |W^*| = |C(T)| - |C(T^*)|$. Combining this fact with (6) yields

$$|W_T| - |W^*| \;\leq\; |C(T)| - |C(T^*)| \cdot \frac{8\,|W^*|}{|C(T^*)|}. \tag{7}$$

Now, if $|C(T)| \leq (1+\varepsilon)\,|C(T^*)|$ holds, then the inequality (7) yields $|W_T| \leq (1+8\varepsilon)\,|W^*|$. Hence, the existence of a polynomial time approximation scheme for problem MinLST$_3$ would imply the existence of a polynomial time approximation scheme for problem VC3. This establishes APX-hardness of MinLST$_3$. Since MinLST$_3$ clearly is contained in APX, the proof of the theorem is complete. ■

Mohar [12] has shown that the vertex cover problem is NP-complete for planar 3-regular graphs. With this, the reduction in Theorem 5.2 yields that MinLST$_3$ is NP-complete even in planar, bipartite graphs of maximum degree 3. The approximability of MinLST and MinLST$_r$ in planar graphs remain open.

Acknowledgments. The second author wishes to thank Cristina Bazgan and Refael Hassin for useful discussions on this problem.

References

[1] P. Alimonti and V. Kann [2000]. Some APX-completeness results for cubic graphs. *Theoretical Computer Science 237*, 123–134.

[2] E.M. Arkin and R. Hassin [1998]. On local search for weighted k-set packing. *Mathematics of Operations Research 23*, 640–648.

[3] G. Ausiello and M. Protasi [1995]. Local search, reducibility and approximability of NP-optimization problems. *Information Processing Letters 54*, 73–79.

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

[5] P. Erdős and H. Sachs [1963]. Reguläre Graphen gegebener Taillenweite mit minimaler Knotenzahl. *Wissenschaftliche Zeitschrift der Martin-Luther-Univsersität Halle-Wittenberg 12*, 251–257.

[6] G. Finn and E. Horowitz [1979]. A linear time approximation algorithm for multiprocessor scheduling. *BIT 19*, 312–320.

[7] H.N. Gabow and M. Stallman [1985]. Efficient algorithms for graphic matroid intersection and parity. *Proceedings of the 12th International Colloquium on Automata, Languages, and Programming (ICALP'1985)*, Springer, LNCS 194, 210–220.

[8] C.A.J. Hurkens and A. Schrijver [1989]. On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. *SIAM Journal on Discrete Mathematics 2*, 68–72.

[9] S.O. Krumke and H-C. Wirth [1998]. On the minimum label spanning tree problem. *Information Processing Letters 66*, 81–85.

[10] L. Lovász and M.D. Plummer [1986]. *Matching Theory.* Elsevier Science, Amsterdam.

[11] H. Lu and R. Ravi [1992]. The power of local optimization: Approximation algorithms for maximum-leaf spanning tree. *Proceedings of the 13th Annual Allerton Conference on Communication, Control, and Computing*, 533–542.

[12] B. Mohar [2001]. Face covers and the genus problem for apex graphs. *Journal of Combinatorial Theory B 82*, 102–117.

[13] C.H. Papadimitriou and M. Yannakakis [1991]. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences 43*, 425–440.

[14] P. Schuurman and T. Vredeveld [2001]. Performance guarantees of local search for multiprocessor scheduling. *Proceedings of the 8th Conference on Integer Programming and Combinatorial Optimization (IPCO'2001)*, Springer, LNCS 2081, 370–382.

[15] T. Vredeveld [2002]. *Combinatorial Approximation Algorithms: Guaranteed versus Experimental Performance.* Ph.D. thesis, TU Eindhoven, The Netherlands.

[16] Y. Wan, G. Chen, and Y. Xu [2002]. A note on the minimum label spanning tree. *Information Processing Letters 84*, 99–101.