

Minimum Sum Multicoloring on the Edges of Planar Graphs and Partial k -trees* (Extended Abstract)

Dániel Marx

Department of Computer Science and Information Theory,
Budapest University of Technology and Economics
Budapest H-1521, Hungary
dmarx@cs.bme.hu

Abstract. The edge multicoloring problem is that given a graph G and integer demands $x(e)$ for every edge e , assign a set of $x(e)$ colors to edge e , such that adjacent edges have disjoint sets of colors. In the *minimum sum edge multicoloring problem* the *finish time* of an edge is defined to be the highest color assigned to it. The goal is to minimize the sum of the finish times. The main result of the paper is a polynomial time approximation scheme for minimum sum multicoloring the edges of planar graphs and partial k -trees.

1 Introduction

In this paper we study an edge multicoloring problem that is motivated by applications in scheduling. We are given a graph with an integer demand $x(e)$ for each edge e . A *multicoloring* is an assignment of a set of $x(e)$ colors to each edge e such that the colors assigned to adjacent edges are disjoint. In multicoloring problems the usual aim is to minimize the *makespan* of the coloring, that is, the total number of different colors used. However, in this paper a different optimization goal is studied, which is related to minimizing the average completion time in scheduling problems. Given a multicoloring, the *finish time* of an edge is defined to be the highest color assigned to it. In the *minimum sum edge multicoloring problem* the goal is to minimize the sum of the finish times.

An application of edge coloring is to model dedicated scheduling of biprocessor tasks. The vertices correspond to the processors and each edge $e = uv$ corresponds to a job that requires $x(e)$ time units of simultaneous work on the two preassigned processors u and v . The colors correspond to the available time slots: by assigning $x(e)$ colors to edge e , we select the $x(e)$ time units when the job corresponding to e is executed. A processor cannot work on two jobs at the same time, this corresponds to the requirement that a color can appear at most once on the edges incident to a vertex. The finish time of edge e corresponds to

* Research is supported in part by grants OTKA 44733, 42559 and 42706 of the Hungarian National Science Fund.

the time slot when job e is finished, therefore minimizing the sum of the finish times is the same as minimizing the sum of completion times of the jobs. Using the terminology of scheduling theory, we minimize the mean flow time, which is a well-studied optimization goal in the scheduling literature. Such biprocessor tasks arise when we want to schedule file transfers between processors [4] or the mutual diagnostic testing of processors [7]. Note that it is allowed that a job is interrupted and continued later: the set of colors assigned to an edge does not have to be consecutive, hence our problem models preemptive scheduling.

In [11] it is shown that for trees the problem is NP-hard, but admits a polynomial time approximation scheme (PTAS). In this paper we extend the PTAS to partial k -trees and planar graphs. The problem is NP-hard for partial 2-trees and planar bipartite graphs even in the unit demand case [9], hence the approximation schemes given in this paper cannot be improved to exact polynomial time algorithms (assuming $P \neq NP$).

Recently, the vertex coloring version of minimum sum multicoloring was investigated by several papers [2, 5, 6, 10]. In [5, 6] PTAS is given for the vertex coloring version of the problem in the case when the graph is a tree, partial k -tree, or a planar graph. The line graph of a bounded degree partial k -tree has bounded treewidth, hence the PTAS of [5] for vertex coloring partial k -trees can be used for the edge coloring of bounded degree partial k -trees. However, in Section 3 we present a more efficient, linear time PTAS for edge coloring such graphs. In Section 4 a PTAS is given for general partial k -trees by reducing the problem to the bounded degree case. In Section 5 the PTAS is extended to planar graphs: using standard techniques (the layering method of Baker [1]) we show how the PTAS for partial k -trees can be used for planar graphs.

2 Preliminaries

The problem considered in this paper is the edge coloring version of minimum sum multicoloring, which can be stated formally as follows:

Minimum Sum Edge Multicoloring (SEMC)

Input: A graph $G(V, E)$ and a *demand* function $x: E \rightarrow \mathbb{N}$.

Output: A *multicoloring* $\Psi: E \rightarrow 2^{\mathbb{N}}$ such that $|\Psi(e)| = x(e)$ for every edge e , and $\Psi(e_1) \cap \Psi(e_2) = \emptyset$ if e_1 and e_2 are adjacent in G .

Goal: The *finish time* of edge e in coloring Ψ is the highest color assigned to it, $f_{\Psi}(e) = \max\{c : c \in \Psi(e)\}$. The goal is to minimize $f_{\Psi}(G) = \sum_{e \in E} f_{\Psi}(e)$, the *sum* of the coloring Ψ .

We extend the notion of finish time to a set E' of edges by defining $f_{\Psi}(E') = \sum_{e \in E'} f_{\Psi}(e)$. Given a graph G and a demand function $x(e)$ on the edges of G , the minimum sum that can be achieved is denoted by $\text{OPT}(G, x)$.

As in [5, 6, 11], we divide the infinite color spectrum into geometrically increasing layers. For some $\epsilon > 0$ and integer $\ell \geq 0$, the (ϵ, ℓ) -*decomposition* divides the set of colors into *layers* L_0, L_1, \dots and *zones* $Z_0, Z_1, \dots, Z_{\ell}$. The layers are of geometrically increasing size: layer L_i contains the range of colors from q_i to

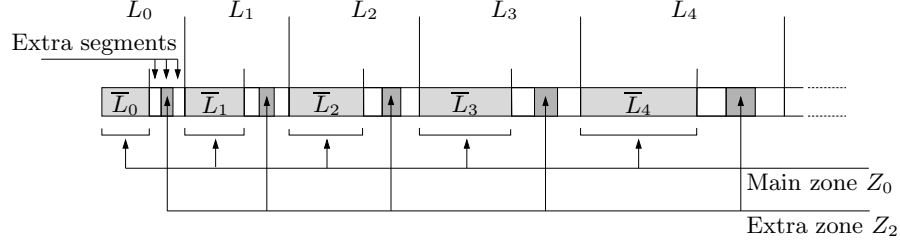


Fig. 1. The decomposition of the colors into layers ($\ell = 3$)

$q_{i+1} - 1$, where $q_i = \lfloor (1 + \epsilon)^i \rfloor$. (If $q_i = q_{i+1}$, then layer L_i is empty). Denote by $Q_i = |L_i| = q_{i+1} - q_i$ the size of the i th layer. The total size of layers L_0, L_1, \dots, L_i is $q_{i+1} - 1$. Later we will use that $(1 + 2\epsilon)q_i \geq q_{i+1} - 1$ for every $i \geq 1$. That is, if we replace a color from layer L_i with another color from L_i , then the new color is at most $(1 + 2\epsilon)$ times larger than the original.

Layer L_i is divided into two parts: the first $\frac{1}{1+\epsilon\ell}Q_i$ colors form the *main block* of layer L_i and the remaining $\frac{\epsilon\ell}{1+\epsilon\ell}Q_i$ colors the *extra block* (see Fig. 1). The main block of layer L_i is denoted by \bar{L}_i . The union of the main block of every layer L_i is the *main zone* Z_0 . Divide the extra block of every layer L_i into ℓ equal parts: these are the ℓ *extra segments* of L_i . The union of the j th extra segment of every layer L_i forms the j th *extra zone* Z_j . Each extra zone contains $\frac{\epsilon}{1+\epsilon\ell}Q_i$ colors from layer L_i . We ignore rounding problems here. It can be shown that by defining the zones carefully, one can achieve the following:

Lemma 1 ([11]). For given ℓ and $\epsilon \leq \frac{1}{2\ell}$, the (ϵ, ℓ) -decomposition of the colors has the following properties:

- (a) For every $c \geq 1$, zone Z_0 contains at least c colors not greater than $\lfloor (1 + \epsilon\ell)c \rfloor$.
- (b) For every $c \geq 1$ and $1 \leq j \leq \ell$, zone Z_j contains at least c colors not greater than $(2/\epsilon) \cdot c$.

Given a multicoloring Ψ , the operation (ϵ, ℓ) -augmentation creates a multicoloring Φ the following way. Consider the (ϵ, ℓ) -decomposition of the colors, and if $\Psi(e)$ contains color c , then let $\Phi(e)$ contain instead the c th color from the main zone Z_0 . By Lemma 1a, $f_\Phi(e) \leq \lfloor (1 + \epsilon\ell)f_\Psi(e) \rfloor$, thus this operation increases the sum by at most a factor of $(1 + \epsilon\ell)$. After the augmentation, the colors of the extra zones are not used, only the colors of the main zone.

A *tree decomposition* of $G(V, E)$ is a tree $T(U, F)$ together with a *bag* $U_x \subseteq V$ for each $x \in U$ such that

- for every $uv \in E$, there is an $x \in U$ with $u, v \in U_x$, and
- for every $v \in V$, the bags containing v induce a connected subtree of T .

The *width* of the decomposition is $\max_{x \in U} |U_x| - 1$, and the *treewidth* of a graph is the smallest width that its tree decompositions can have. A *partial k -tree* is a graph with treewidth at most k . For background on partial k -trees and treewidth, the reader is referred to [3].

In [11], the following scaling property is proved for trees: if the demand of every edge is increased by at most a factor of $1 + \epsilon$, then the sum increases by at most a factor of $1 + \epsilon$. As an application of the layer and zones defined above, we prove a weaker property for partial k -trees: a $1 + \epsilon$ increase of demand causes an at most $1 + 2(k + 1)\epsilon$ increase of the sum (Lemma 3). First we need a special orientation, which will be used later as well.

Lemma 2. *Let G be a partial k -tree. There is a proper $k + 1$ -coloring of the vertices of G , and an orientation of the edges of G such that the outdegree of every vertex is at most k , and the outneighbors of a vertex have distinct colors.*

Proof. Consider a tree decomposition $T(U, F)$ of G . Assume that the tree T is rooted. For each vertex v of G , consider those nodes of T whose bags contain v , and let f_v be the highest such node (i.e., the node that is closest to the root).

The vertices can be colored with $k + 1$ colors such that the vertices in the same bag have distinct colors (that is, we take a coloring of the chordal graph induced by the decomposition).

Let $e = uv$ be an edge of G . The trees T_u and T_v intersect, hence either node $f_u \in U$ of u is descendant of $f_v \in U$ of v , or vice versa. If f_u is the descendant of f_v , then we direct e from u to v , otherwise we direct it from v to u . If $f_u = f_v$, then direct the edge uv arbitrarily. Notice that all the outneighbors of u are contained in the bag of f_u : if a subtree intersects T_u , and it contains an ancestor of f_u , then it has to contain f_u as well. Now it is clear that u has at most k outneighbors, and they have distinct colors.

Lemma 3. *Let (G, x) be an instance of SEMC and let x' be a demand function with $x'(e) \leq (1 + \epsilon) \cdot x(e)$ for every edge e . If G is a partial k -tree, then $\text{OPT}(G, x') \leq (1 + 2(k + 1)\epsilon) \cdot \text{OPT}(G, x)$.*

Proof. Let Ψ be an optimum coloring of (G, x) , and let Ψ' be the $(2\epsilon, k + 1)$ -augmentation of Ψ . By Lemma 1, we have $f_{\Psi'}(e) \leq (1 + 2(k + 1)\epsilon)f_{\Psi}(e)$ for every edge e .

Consider the coloring and orientation given by Lemma 2. Let e_1, \dots, e_ℓ be those edges that enter v , assume that $f_{\Psi'}(e_1) \leq f_{\Psi'}(e_2) \leq \dots \leq f_{\Psi'}(e_\ell)$. Edge e_i requires $x'(e_i) - x(e_i) \leq \epsilon x(e_i)$ extra colors to satisfy demand function x' . If vertex v has color c , then we use extra zone Z_c to give additional colors to these edges. The first $\epsilon x(e_1)$ colors of Z_c are given to edge e_1 , the next $\epsilon x(e_2)$ colors are given to e_2 . It is clear that no conflict arises with the assignment of these new colors. If two edges conflict in zone Z_c , then their end vertices have the same color c . By Lemma 2, this is only possible if they enter the same vertex, but in this case the construction ensures that the edges receive different colors.

We show that these additional colors do not increase the finish time of the edges. The finish time of e_i is clearly at least $\sum_{j=1}^i x(e_j)$ in Ψ' . The last color given to edge e_i is color $\sum_{j=1}^i \epsilon x(e_j)$ from extra zone Z_c . However, by Lemma 1b, zone Z_c contains at least $\sum_{j=1}^i \epsilon x(e_j)$ colors below $\frac{2}{2\epsilon} \cdot \sum_{j=1}^i \epsilon x(e_j) = \sum_{j=1}^i x(e_j) \leq f_{\Psi'}(e_i)$. Therefore it remains true that $f_{\Psi'}(e) \leq (1 + 2(k + 1)\epsilon)f_{\Psi}(e)$ for every edge e .

3 Bounded Degree Partial k -trees

In this section we show that SEMC admits a linear time PTAS for bounded degree partial k -trees. If G is a partial k -tree with maximum degree D , then its line graph has treewidth at most $(k + 1)D - 1$ (see [3, Lemma 32]). Therefore the PTAS given by Halldórsson and Kortsarz [5] for the minimum sum vertex coloring of partial k -trees can be used to edge color bounded degree partial k -trees. However, the time complexity of their algorithm is $n^{O(k^2/\epsilon^5)}$. We show here that for line graphs of bounded degree partial k -trees the running time can be reduced to linear.

The main idea of the PTAS in [5] is that one can assign a polynomial number of color sets to each vertex such that the graph has a good approximate coloring where every vertex uses one of the sets assigned to it. This is proved by using probabilistic arguments and by transforming the solution to a standard form. The best coloring using the selected color sets can be found in polynomial time with the standard dynamic programming algorithm of partial k -trees. In [11] a similar path is taken to give a linear time PTAS for SEMC in bounded degree trees: it is shown that a constant number of color sets can be assigned to the edges such that there is good approximation with these sets. In this case the probabilistic argument of [5] can be replaced by a simple greedy algorithm. Here we construct a constant number of color sets for the edges of bounded degree partial k -trees, yielding a linear time PTAS. The construction depends on some strong results concerning the asymptotics of the chromatic index.

3.1 Approximating the Makespan

As explained in the introduction, in this paper our goal is to minimize the sum of finish times, and we are not interested in minimizing the makespan. However, (as in [5] and [11]) the algorithm for minimizing the sum requires the use of some results on approximating the makespan.

Minimizing the makespan is closely related to the chromatic index problem. Given a graph G and a demand function x on the edges, let (G, x) denote the multigraph obtained by replacing every edge e by $x(e)$ parallel edges, and let $\chi'(G, x)$ be the chromatic index of (G, x) . It is clear that the minimum makespan with demand x is $\chi'(G, x)$.

Shannon [12] has shown that $\chi'(G) \leq \lceil 3\Delta(G)/2 \rceil$ for every multigraph G , where $\Delta(G)$ is the maximum degree of G . Since $\Delta(G)$ is a lower bound on the fractional chromatic index $\chi^*(G)$, this implies that $\chi'(G) \leq 3\chi^*(G)/2$. The following theorem shows that better bounds can be given for $\chi'(G)$ if $\chi^*(G)$ is large:

Theorem 1 (Kahn, [8]).

For every $\gamma > 0$, there exists $D(\gamma)$ such that for any multigraph G with $\chi^(G) > D(\gamma)$, we have $\chi' < (1 + \gamma)\chi^*(G)$.*

We use this result to construct a constant number of color sets that can approximate the makespan:

Lemma 4. *For every $\epsilon > 0$ and integers $\Delta, t > 0$, there is a family of color sets $\mathcal{C}_{t,\Delta,\epsilon}$ such that if the maximum degree of G is Δ , and graph G with demand $x(e)$ has a coloring with makespan t , then it has a coloring with makespan at most $(1 + \epsilon)t$ such that every edge receives a color set from $\mathcal{C}_{t,\Delta,\epsilon}$. Moreover, the size of $\mathcal{C}_{t,\Delta,\epsilon}$ depends only on ϵ and Δ .*

Proof. Set $\gamma = \epsilon/6$, and let $x_1(e) = \lfloor 4D(\gamma)\Delta/(\gamma t) \cdot x(e) \rfloor$, where $D(\gamma)$ is from Theorem 1. It follows that $\chi^*(G, x_1) \leq 4D(\gamma)\Delta/(\gamma t) \cdot \chi^*(G, x) \leq 4D(\gamma)\Delta/\gamma \cdot \chi'(G, x)/t \leq 4D(\gamma)\Delta/\gamma$. Let $x_2(e) = \lceil 4D(\gamma)\Delta/(\gamma t) \cdot x(e) \rceil$. In x_2 every edge requires at most one color more than in x_1 , therefore by Vizing's theorem, x_2 can be satisfied with $\Delta + 1$ new colors: $\chi^*(G, x_2) \leq \chi^*(G, x_1) + \Delta + 1$. If $\chi^*(G, x_2) > 2D(\gamma)\Delta/\gamma$, then we can apply Theorem 1 to show that $\chi'(G, x_2) \leq (1 + \gamma)\chi^*(G, x_2) \leq (1 + \gamma)(\chi^*(G, x_1) + \Delta + 1) \leq (1 + \gamma)^2\chi^*(G, x_1) \leq (1 + \gamma)^2 4D(\gamma)\Delta/\gamma$. If $\chi^*(G, x_2) \leq 2D(\gamma)\Delta/\gamma$, then $\chi'(G, x_2) \leq 3D(\gamma)\Delta/(\gamma)$ follows simply from $\chi' \leq 3\chi^*/2$. Now if we repeat every color $\lceil \gamma t/(4D(\gamma)\Delta) \rceil$ times in the coloring of (G, x_2) , then we obtain a coloring of (G, x) with at most $\chi'(G, x_2) \cdot \lceil \gamma t/(4D(\gamma)\Delta) \rceil \leq \chi'(G, x_2) \cdot (1 + \gamma) \cdot \gamma t/(4D(\gamma)\Delta) \leq (1 + \gamma)^3 t \leq (1 + \epsilon)t$ colors (in the first inequality we assume that t is sufficiently large). Let $\mathcal{C}_{t,\Delta,\epsilon}$ contain those color sets that can be obtained from a subset of $\{1, \dots, \lfloor (1 + \epsilon)4D(\gamma)\Delta/\gamma \rfloor\}$ by repeating every color $\lceil \gamma t/(4D(\gamma)\Delta) \rceil$ times. What we have shown is that there is an $(1 + \epsilon)$ -approximate coloring where every edge uses a set from $\mathcal{C}_{t,\Delta,\epsilon}$. The size of $\mathcal{C}_{t,\Delta,\epsilon}$ is $2^{\lfloor (1 + \epsilon)4D(\gamma)\Delta/\gamma \rfloor} \leq 2^{O((1 + \epsilon)D(\epsilon/4)\Delta/\epsilon)}$, which depends only on ϵ and Δ . \square

3.2 Approximating the Sum

To give a linear time PTAS for SEMC on almost bounded degree partial k -trees, first we construct a constant number of color sets for each edge. Moreover, we give a slightly more general algorithm for SEMC that works on “almost bounded degree” partial k -trees as well. A *pendant edge* is an edge that has a degree one end vertex, an *almost bounded degree* graph is a graph that has bounded degree after deleting the pendant edges. A non-pendant edge will be called a *core edge*, and the number of core edges incident to a vertex will be called the *core degree* of the vertex. Deleting the pendant edges of G gives the *core* G_0 of G . The treewidth of the line graph of an almost bounded degree partial k -tree is not necessarily bounded, hence the vertex coloring algorithm of [5] cannot be used for such graphs.

Lemma 5. *For every $\epsilon_0 > 0$ and integers $k, y, D > 0$ there is a family of color sets $\mathcal{D}_{k,y,D,\epsilon_0}$ such that for every partial k -tree G with maximum core degree D and every demand function there is a $(1 + \epsilon_0)$ -approximate coloring where every core edge e with demand y receives a color set from $\mathcal{D}_{k,y,D,\epsilon_0}$. Moreover, the size of $\mathcal{D}_{k,y,D,\epsilon_0}$ depends only on k, D , and ϵ_0 .*

Proof. Let Ψ be an optimum coloring of the instance (G, x) . With a series of transformations, we modify Ψ to a special form where only a constant number of possible color sets can appear at a given core edge. Moreover, the transformations

will increase the sum only by a factor of $1 + \epsilon_0$. Most of the ideas in this proof are taken from [5].

Set $\epsilon := \epsilon_0/(10D)$ and let Ψ' be obtained by performing an $(\epsilon, D + 2)$ -augmentation on Ψ . By Lemma 1a, we have $f_{\Psi'}(G) \leq (1 + (D + 2)\epsilon)f_{\Psi}(G)$.

The first step is to ensure that every core edge e uses the main zone only between $\epsilon x(e)/2$ and $2x(e)/\epsilon$. This means that e can use the main zone only from layer $\lfloor \log_{1+\epsilon} \epsilon x(e)/2 \rfloor$ to layer $\lceil \log_{1+\epsilon} 2x(e)/\epsilon \rceil$, that is, only from at most $\log_{1+\epsilon}((2x(e)/\epsilon)/(\epsilon x(e)/2)) + 2 = \log_{1+\epsilon} 4/\epsilon^2 + 2 = O(1/\epsilon \cdot \log 1/\epsilon)$ layers.

By Vizing's Theorem, one can assign a type $1, \dots, D + 1$ to every core edge e such that adjacent edges have different types. If $f_{\Psi'}(e) > 2x(e)/\epsilon$ for a core edge e of type j , then modify $\Psi'(e)$ to be the first $x(e)$ colors of zone Z_j . By Lemma 1b, Z_j contains at least $x(e)$ colors not greater than $2x(e)/\epsilon$, therefore the modification does not increase the finish time of edge e .

If $\Psi'(e)$ contains colors from the main zone below $\epsilon x(e)/2$, then delete these colors and let $\Psi'(e)$ contain instead the first $\epsilon x(e)/2$ colors from zone Z_j . There are at least $\epsilon x(e)/2$ colors in Z_j below $2/\epsilon \cdot \epsilon x(e)/2 = x(e)$. Since the finish time of e is at least $x(e)$, hence this modification does not increase the finish time of e .

For every core edge e , define $x_i(e) := |\Psi'(e) \cap \bar{L}_i|$ the number of colors used by e in the main block of layer i . In the core graph G_0 , the multicoloring problem with demand $x_i(e)$ has a solution with makespan $|\bar{L}_i|$: for example, $\Psi'(e) \cap \bar{L}_i$ is such a coloring. Therefore by Lemma 4, there is a coloring Ψ_i with makespan $\lfloor (1 + \frac{\epsilon}{2})|\bar{L}_i| \rfloor$ such that $\Psi_i(e) \in \mathcal{C}_{|\bar{L}_i|, D, \epsilon/2}$. We let Ψ_i determine how the colors are used in the main block of layer L_i . That is, if Ψ_i assigns color c to edge e , then we assign the c th color of \bar{L}_i to e . However, Ψ_i uses $\lfloor (1 + \frac{\epsilon}{2})|\bar{L}_i| \rfloor$ colors, so we run out of the colors of \bar{L}_i . Extra zone Z_{D+2} is used to provide $\lfloor \frac{\epsilon}{2}|\bar{L}_i| \rfloor$ additional colors. We use the first $\lfloor \frac{\epsilon}{2}|\bar{L}_1| \rfloor$ colors of Z_{D+2} when recoloring \bar{L}_1 , the next $\lfloor \frac{\epsilon}{2}|\bar{L}_2| \rfloor$ colors for the recoloring of \bar{L}_2 , and so on. This means that when we recolor \bar{L}_i , only the first $\sum_{j=1}^i \lfloor \frac{\epsilon}{2}|\bar{L}_j| \rfloor$ colors of Z_{D+2} are used. Zone Z_{D+2} contains that many colors below $\frac{2}{\epsilon} \sum_{j=1}^i \lfloor \frac{\epsilon}{2}|\bar{L}_j| \rfloor \leq \sum_{j=1}^i |\bar{L}_j|$ (Lemma 1b). Therefore we do not use colors above layer L_i , and the last color of every edge remains in the same layer. As noted in Section 2, this implies that the finish time is increased by at most a factor of $1 + 2\epsilon$ for every edge.

In the previous paragraph we have recolored all the core edges at the same time, thus no conflict arises between the core edges, but there can be conflicts between a core edge and a pendant edge. However, at each vertex v , the number of colors used by the core edges from the main block of layer i did not increase. In fact, it is possible that it was decreased since in the recoloring extra zone Z_{D+2} was also used. Therefore there remains enough colors in \bar{L}_i to satisfy the requirements of the pendant edges attached to v . This means that the last color of each pendant edge will remain in the same layer, hence the finish time of a pendant edge increases by at most a factor of $(1 + 2\epsilon)$. Thus the finish time of every edge is at most $(1 + 2\epsilon)(1 + (D + 2)\epsilon) \leq (1 + \epsilon_0)$ times higher than in Ψ , and the coloring obtained is $(1 + \epsilon_0)$ -approximate.

It is easy to show that only a constant number of different color sets can arise for a given edge e . This follows from the facts that e uses the main zone only in a constant number of different layers, and in each layer it has a constant number of different possibilities. \square

The PTAS uses standard dynamic programming methods to find the best coloring with the selected color sets. However, there is a slight twist: the pendant edges have to be handled differently. The idea is that if the core edges incident to v are given a coloring, then this determines what is the best way of coloring the pendant edges incident to v . Details omitted.

Theorem 2. *For every $\epsilon_0 > 0$, and integers $k, D > 0$, there is a linear time algorithm that gives a $(1 + \epsilon_0)$ -approximate solution to the SEMC problem for partial k -trees with maximum core degree D .* \square

4 General Partial k -Trees

We prove that SEMC admits a linear time PTAS for partial k -trees. The main idea is to modify the graph to be an almost bounded degree graph. After these modifications the algorithm of Theorem 2 can be used, and we show that the coloring for the almost bounded degree graph can be transformed into a coloring of the original graph with only a small increase of the sum.

Theorem 3. *For every $\epsilon > 0$ and integer $k > 0$, there is a linear time algorithm that gives a $(1 + \epsilon)$ -approximate solution to the SEMC problem for every partial k -tree G and demand function x_0 .*

Proof. We show how to find a $(1 + K\epsilon)$ -approximate solution in linear time, where K is a constant depending only on k . This implies that there is a $(1 + \epsilon)$ -approximation algorithm for every k and $\epsilon > 0$. The algorithm consists of a series of phases, in the following we describe these phases.

Phase 1: Rounding the Demands. Let $x(e)$ be the smallest q_i that is not smaller than $x_0(e)$. Since $q_{i+1} \leq (1 + \epsilon)^{i+1} \leq (1 + \epsilon)(q_i + 1)$, thus $x(e) \leq (1 + \epsilon)x_0(e)$. Therefore by Lemma 3, this modification increases the minimum sum by at most a factor of $1 + 2(k + 1)\epsilon$, hence a $(1 + O(\epsilon))$ -approximation for (G, x) is also a $(1 + O(\epsilon))$ -approximation for (G, x_0) . An edge e with demand q_i will be called a *class i* edge (if $x(e) = q_i$ for more than one i , then take the smallest i).

Assume we have a subset of the edges that are incident to the same vertex, and there are exactly $c(i)$ edges of class i among them. The best way to color these edges is to order them by increasing demand size, and color them in this order. Therefore the sum of these edges in every coloring is at least

$$\sum_{i=1}^{\infty} \sum_{j=1}^{c(i)} \left(\sum_{k=1}^{i-1} c(k)q_k + jq_i \right) = \sum_{i=1}^{\infty} \left(c(i) \sum_{k=1}^{i-1} c(k)q_k + q_i \sum_{j=1}^{c(i)} j \right) = \sum_{i=1}^{\infty} \left(c(i) \sum_{k=1}^{i-1} c(k)q_k + q_i(c(i) + 1)c(i)/2 \right) \quad (1)$$

The parentheses in the first expression contains the finish time of the j th edge of class i . The first term is the contribution of the edges with class less than i , the second term is the contribution of the first j edges of class i .

Phase 2: Classifying the edges. Consider the $(k+1)$ -coloring of the vertices and the orientation of the edges given by Lemma 2. The edges of the graph are divided into *large edges*, *small edges*, and *frequent edges*. It will be done in such a way that at most $D := 6/\epsilon^7$ large edges enter every vertex. Denote by $n(v, i)$ the number of class i edges entering v . Let $N(v)$ be the largest i such that $n(v, i) > 0$ and set $F := 6/\epsilon^5$. Let e be a class i edge entering v . If $n(v, i) > F$, then e is a *frequent edge*. If $n(v, i) \leq F$ and $i \leq N(v) - 1/\epsilon^2$, then e is a *small edge*. Otherwise, if $n(v, i) \leq F$ and $i > N(v) - 1/\epsilon^2$, then e is a *large edge*. Clearly, at most $F \cdot 1/\epsilon^2 = 6/\epsilon^7 = D$ large edges can enter v : for each class $N(v), N(v) - 1, \dots, N(v) - 1/\epsilon^2 + 1$, there are at most F such edges.

Phase 3: Splitting the edges. The graph is split at the tail of every small and frequent edge. That is, if $e = \overrightarrow{uv}$ is a small or frequent edge, then add a new vertex u' , and replace \overrightarrow{uv} by $\overrightarrow{u'v}$. The resulting graph G' is an almost bounded degree graph. Deleting the pendant edges deletes every small and frequent edge, therefore only the large edges remain. We have seen that at most D large edges enter every node, and the outdegree of every vertex is at most k , thus the degree of the remaining graph is at most $D + k$. Therefore graph G' can be colored with the algorithm of Theorem 2 in linear time. This gives a $(1 + \epsilon)$ -approximate coloring Ψ_1 of G' . Coloring Ψ_1 can be used as a coloring for the original graph G , but in this case there might be conflicts: a small or frequent edge might be in conflict with the edges incident to its tail vertex. In the rest of the proof, we transform Ψ_1 into a proper coloring of G in such a way that the sum of the coloring does not increase too much. We will distinguish between the following 4 types of conflicts, they will be handled separately in the phases to follow:

1. Conflicts involving a small edge \overrightarrow{uv} .
2. Conflict between two frequent edges $\overrightarrow{uv_1}$ and $\overrightarrow{uv_2}$.
3. Conflict between a frequent edge $\overrightarrow{uv_1}$ and a large edge $\overrightarrow{uv_2}$.
4. Conflict between a frequent edge \overrightarrow{uv} and a frequent or large edge \overrightarrow{wu} .

Phase 4: Small Edges. In this phase we resolve the conflicts of the first type. Consider the $(\epsilon, k+1)$ -augmentation of the coloring Ψ_1 , this results in a $(1 + O(\epsilon))$ -approximate coloring Ψ_2 . We modify Ψ_2 in such a way that the small edges use

only the extra zones. More precisely, if the head of a small edge e has color $r \in \{1, 2, \dots, k+1\}$, then e is recolored using the colors in Z_r . Since the extra zones contain only a very small fraction of the color spectrum, the recoloring can significantly increase the finish time of the small edges, but not more than by a factor of $2/\epsilon$ (Lemma 1b). However, we show that the total demand of the small edges entering v is so small compared to the demand of the largest edge entering v , that their total finish time will be negligible, even after this large increase.

By definition, the largest edge entering v has demand $q_{N(v)}$. Let S_v be the set of small edges entering v . Let r be the color of vertex v . Color the edges in S_v one after the other, in the order of increasing demand size, using only the colors in Z_r . Call the resulting coloring Ψ_3 . We claim that $f_{\Psi_3}(S_v) \leq \epsilon q_{N(v)}$ for every node v , thus transforming Ψ_2 into Ψ_3 increases the total sum by at most $\sum_{v \in G} f_{\Psi_3}(S_v) \leq \epsilon \sum_{v \in G} q_{N(v)} \leq \epsilon f_{\Psi_2}(G)$ and $f_{\Psi_3}(G) \leq (1+\epsilon)f_{\Psi_2}(G)$ follows. To give an upper bound on $f_{\Psi_3}(S_v)$, we assume the worst case, that is, $n(v, i) = F$ for every $i \leq N(v) - 1/\epsilon^2$. Imagine first that the small edges are colored using the full color spectrum, not only with the colors of zone Z_r . Assume that the small edges are colored in the order of increasing demand size, and consider a class m edge e . In the coloring, only edges of class not greater than m are colored before e . Hence the finish time of e is at most

$$\begin{aligned} \sum_{i=0}^m n(v, i)q_i &\leq F \sum_{i=0}^m (1+\epsilon)^i \leq 6(1+\epsilon)/\epsilon^6 \cdot (1+\epsilon)^m \\ &\leq 14/\epsilon^6 \cdot \frac{1}{2}(1+\epsilon)^m \leq 14/\epsilon^6 \cdot \lfloor (1+\epsilon)^m \rfloor = 14/\epsilon^6 \cdot q_m. \end{aligned}$$

That is, the finish time of an edge is at most $14/\epsilon^6$ times its demand (in the second inequality, we used $\sum_{i=0}^m (1+\epsilon)^i = ((1+\epsilon)^{m+1} - 1)/\epsilon < (1+\epsilon)^{m+1}/\epsilon$). Therefore the total finish time of the small edges is at most $14/\epsilon^6$ times the total demand, which is

$$\begin{aligned} \frac{14}{\epsilon^6} \sum_{i=0}^{N(v)-1/\epsilon^2} n(v, i)q_i &\leq \frac{84}{\epsilon^{11}} \sum_{i=0}^{N(v)-1/\epsilon^2} (1+\epsilon)^i \leq \frac{85}{\epsilon^{12}} (1+\epsilon)^{N(v)-1/\epsilon^2} \\ &\leq \frac{85}{\epsilon^{12}} \cdot 2^{-1/\epsilon} \cdot (1+\epsilon)^{N(v)} \leq \frac{\epsilon^2}{2} \cdot \frac{1}{2} (1+\epsilon)^{N(v)} \leq \frac{\epsilon^2}{2} \cdot \lfloor (1+\epsilon)^{N(v)} \rfloor = \frac{\epsilon^2}{2} \cdot q_{N(v)}. \end{aligned}$$

(In the third inequality we use $(1+\epsilon)^{1/\epsilon} \geq 2$, in the fourth inequality it is assumed that ϵ is sufficiently small that $2^{1/\epsilon} \geq 4 \cdot 85/\epsilon^{14}$ holds.) However, the small edges do not use the full color spectrum, only the colors in zone Z_r . By Lemma 1b, zone Z_r contains at least c colors up to $2/\epsilon \cdot c$, thus every finish time in the calculation above should be multiplied by at most $2/\epsilon$. Therefore the sum of the small edges is at most $2/\epsilon \cdot \epsilon^2/2q_{N(v)} \leq \epsilon q_{N(v)}$ as claimed, and Ψ_3 is a $(1+O(\epsilon))$ -approximate coloring.

Phase 5: Reordering the frequent edges. Now we have a coloring Ψ_3 of G such that the only type of conflict that is possible is between a frequent

edge \overrightarrow{uv} and an edge incident to u . In this section, we resolve the conflicts of the second type. Consider the $F(v, i)$ frequent edges of class i that enter v . We randomly reorder the color sets assigned to these edges. That is, take a uniformly distributed random permutation σ of the class i frequent edges entering v , and set $\Psi_4(e) = \Psi_3(\sigma(e))$ for these edges. Since all the class i edges have the same demand, a class i edge can receive the color set assigned to another edge of class i . Moreover, since the reordered edges have the same head v , the only conflicts that can arise are of types 2–4. We repeat the reordering independently for every v and i , and set $\Psi_4(e) = \Psi_3(e)$ for every small and large edge.

We argue that after the reordering, with high probability there are only relatively few type 2 conflicts, and these conflicts can be resolved with only a small increase of the sum. A frequent edge $\overrightarrow{uv_1}$ that conflicts with another frequent edge $\overrightarrow{uv_2}$ will be called a *marked* edge. Denote by $M(v, i)$ the number of class i marked edges entering v . The following claim bounds the expected number of marked edges:

Claim. For every v and i , $\mathbb{E}[M(v, i)] \leq \epsilon^4 F(v, i)$.

Proof. Consider frequent edges $\overrightarrow{uv_1}$ of class i_1 and $\overrightarrow{uv_2}$ of class i_2 . We bound the probability that they conflict. Let $a_1, \dots, a_{F(v_1, i_1)}$ be the class i_1 frequent edges entering v_1 ordered by increasing finish time. For every $1 \leq j \leq F(v_1, i_1)$, denote by A_j the interval between the first and last color of $\Psi_3(a_j)$, i.e., $A_j = [\min\{c : c \in \Psi_3(a_j)\}, \max\{c : c \in \Psi_3(a_j)\}]$. Since the algorithm of Theorem 2 colors the frequent (pendant) edges entering v_1 one after the other, thus the intervals A_j are disjoint. The intervals B_j are similarly defined for $1 \leq j \leq F(v_2, i_2)$, but now we consider the class i_2 edges entering v_2 . Let A (resp. B) be the interval between the first and last color of $\Psi_4(\overrightarrow{uv_1})$ (resp. $\Psi_4(\overrightarrow{uv_2})$). If $\overrightarrow{uv_1}$ and $\overrightarrow{uv_2}$ conflict in Ψ_4 , then A and B intersect. By bounding the probability that $A \cap B \neq \emptyset$, we give a bound on the probability that the two edges are in conflict.

Because of the way the color sets were reordered, interval A can be any of the intervals A_j with probability $1/F(v_1, i_1)$, and similarly for the interval B . Since the reordering at vertices v_1 and v_2 are independent, $(A, B) = (A_{j_1}, B_{j_2})$ with probability $[F(v_1, i_1)F(v_2, i_2)]^{-1}$ for any pair (A_{j_1}, B_{j_2}) . We show that the number of pairs (A_{j_1}, B_{j_2}) such that $A_{j_1} \cap B_{j_2} \neq \emptyset$ is at most $F(v_1, i_1) + F(v_2, i_2)$. The proof is by induction on $F(v_1, i_1) + F(v_2, i_2)$, the total number of intervals. If either $F(v_1, i_1)$ or $F(v_2, i_2)$ is zero, then the claim trivially holds. Assume without loss of generality that the right end point of A_1 is not greater than the right end point of B_1 . If we remove interval A_1 , then the number of non-disjoint pairs is decreased by at most one, since A_1 can intersect only B_1 . Therefore the number of intervals is decreased by one, the number of non-disjoint pairs is decreased by at most one, completing the induction. Therefore there are at most $F(v_1, i_1) + F(v_2, i_2)$ non-disjoint pairs, and the probability that $A \cap B \neq \emptyset$ is at most

$$\frac{F(v_1, i_1) + F(v_2, i_2)}{F(v_1, i_1)F(v_2, i_2)} = \frac{1}{F(v_1, i_1)} + \frac{1}{F(v_2, i_2)} \leq \epsilon^5/3.$$

Since the outdegree of u is at most k , thus edge $\overrightarrow{uv_1}$ can be in type 2 conflict with at most $k - 1$ possible edges. Therefore $\overrightarrow{uv_1}$ is marked with probability at most

$(k-1)\epsilon^5/3 \leq \epsilon^4$. Thus the expected number of class i marked edges entering v is at most $\epsilon^4 F(v, i)$. \square

Take an $(\epsilon, k+1)$ -augmentation of Ψ_4 . If vertex v has color r , then the marked edges entering v are recolored using extra zone Z_r . The edges are recolored in increasing order of demand size: first the class 1 edges, etc. It can be shown that the expected total sum of the recolored marked edges is at most $2/\epsilon \cdot \epsilon^4 = 2\epsilon^3$ times the total sum of the frequent edges: this follows from Lemma 1b, from the claim, and from the convexity of the expression (1). Thus by Markov's Inequality, with probability at least $1-2\epsilon^2$, the transformation increases the sum by at most a factor of $(1+\epsilon)$ and the resulting coloring Ψ_5 is $(1+O(\epsilon))$ -approximate.

Phase 6: Resolving type 3 conflicts. We show that with high probability there are only few type 3 conflicts in Ψ_5 , therefore they can be resolved with only a small increase of the sum. If a frequent edge $\overrightarrow{uv_1}$ conflicts with a large edge $\overrightarrow{uv_2}$, then we remove the conflicting colors from the large edge. We say that a large edge \overrightarrow{uv} is *marked* if more than $\epsilon x(\overrightarrow{uv})/2$ colors were removed from it. Take an $(\epsilon, k+1)$ -augmentation of Ψ_5 , the removed colors of the marked edges are replaced using the extra zones. This results in a coloring Ψ_6 , where every edge e lost at most $\epsilon x(\overrightarrow{uv})/2$ colors. By an argument similar to the one in Phase 5, it can be shown that with high probability Ψ_6 is an $(1+O(\epsilon))$ -approximate coloring. Furthermore, by Lemma 3 the size of the color sets can be increased by a factor of $(1+\epsilon)$ at the cost of increasing the sum by a factor of $1+2(k+1)\epsilon$. Therefore we get a $(1+O(\epsilon))$ -approximate coloring Ψ_7 that satisfies all the demands.

Phase 7: Low frequent edges. In this phase we ensure that if a frequent edge e is conflicting, then edge e uses only colors above $2x(e)/\epsilon$. This will help us in resolving the type 4 conflicts in Phase 8.

If a frequent edge e has finish time at most $8x(e)/\epsilon^2$ in Ψ_7 , then e will be called a *low* edge. Clearly, there can be at most $8/\epsilon^2$ low edges of a given class entering vertex v . First we ensure that the low frequent edges are not conflicting. Let e_1, \dots, e_m be the low edges entering node v , ordered by increasing demand size. Let us perform an $(\epsilon, k+1)$ -augmentation on Ψ_7 . If v has color r , then we use the colors in zone Z_r to recolor the low edges: the first $x(e_1)$ colors in Z_r are given to e_1 , the next $x(e_2)$ colors are given to e_2 and so on. In the resulting coloring Ψ_8 the finish time of the low edges can increase. However, we show that this increase is negligible compared to the total sum of the frequent edges.

Denote by $F(v, m)$ the number of class m frequent edges entering v , and by $\ell(v, m) \leq 8/\epsilon^2$ the number of class m low frequent edges entering v . Since $F(v, m) \geq 6/\epsilon^5$ whenever $\ell(v, m)$ is non-zero, we have $\ell(v, m) \leq 2\epsilon^3 F(v, m)$. By Lemma 1b and the convexity of the expression (1), this means that the total sum of the low edges, even if they use only an extra zone, is at most $2/\epsilon \cdot 2\epsilon^3 < \epsilon$ times the sum of the frequent edges. Thus Ψ_8 is a $(1+O(\epsilon))$ -approximate coloring.

It is still possible that a conflicting frequent edge e uses colors up to $2x(e)/\epsilon$, but in this case its finish time is at least $8x(e)/\epsilon^2$. We perform once again an $(\epsilon, k+1)$ -augmentation on Ψ_8 . Consider the conflicting frequent edges entering v

in increasing order of demand size, and if we encounter an edge e that uses colors up to $2x(e)/\epsilon$, then these colors are replaced using the next available colors of zone Z_r (r is the color of vertex v). More precisely, we want to ensure that every color of e is at least $2x(e)/\epsilon$, hence we assign to it the next available color above $2x(e)/\epsilon$. At the point when edge e is recolored, only colors up to $2x(e)/\epsilon$ have been migrated to zone Z_r , since the edges before e have demand not greater than $x(e)$. Therefore the new colors are not larger than the smallest $4x(e)/\epsilon$ colors of zone Z_r , and these colors are at most $(2/\epsilon) \cdot 4x(e)/\epsilon = 8x(e)/\epsilon^2$. We have ensured that every frequent edge has finish time greater than $8x(e)/\epsilon^2$, thus the recoloring does not increase the sum. Therefore we obtain a coloring Ψ_9 with $f_{\Psi_9}(G) = f_{\Psi_8}(G)$ where every frequent edge e uses only colors above $2x(e)/\epsilon$.

Phase 8: Resolving type 4 conflicts. First perform an $(\epsilon, k+1)$ -augmentation on Ψ_9 . Let $e = \vec{uv}$ be a frequent edge that conflicts with some of the edges entering u . Let the color of vertex u be r . There are at most $x(e)$ colors that are used by both e and an edge entering u . We resolve these conflicts by recoloring the edges entering u in such a way that they use the first at most $x(e)$ colors in zone Z_r instead of the colors in $\Psi_9(e)$. It is clear that if this operation is applied for every frequent edge e , then the resulting coloring Ψ_{10} does not contain type 4 conflicts.

In Phase 7 we have ensured that a conflicting frequent edge uses only colors above $2x(e)/\epsilon$. Therefore if an edge \vec{wu} is recolored, then it has finish time at least $2x(e)/\epsilon$, otherwise it does not conflict with frequent edge $e = \vec{uv}$. On the other hand, by Lemma 1b, zone Z_r contains at least $x(e)$ colors up to $2x(e)/\epsilon$, thus the recoloring does not add colors above that. Therefore the finish time of \vec{wu} is not increased. Thus in this phase the sum is only increased by the $(\epsilon, k+1)$ -augmentation. Therefore finally we obtained a proper $(1 + O(\epsilon))$ -approximate coloring Ψ_{10} . \square

5 Planar Graphs

In [5] the PTAS for minimum sum multicoloring the vertices of partial k -trees is extended to planar graphs using the layering method of Baker [1]. In this section we follow a similar path to obtain a linear time PTAS for SEMC on planar graphs. To present the algorithm, we will need the definition of t -outerplanar graphs:

Definition 1 (t -outerplanar). *An embedding of graph $G(V, E)$ is 1-outerplanar (or simply outerplanar), if it is planar, and all vertices lie on the exterior face. For $t \geq 2$, an embedding of a graph G is t -outerplanar, if it is planar, and when all vertices on the outer face are deleted, then a $(t-1)$ -outerplanar embedding of the resulting graph is obtained. A graph is t -outerplanar, if it has a t -outerplanar embedding. A t -outerplanar embedding divides the vertices into t layers: layer ℓ_1 contains the vertices on the outer face, while for $i \geq 2$, layer ℓ_i contains those vertices that are on the outer face after deleting layers $\ell_1, \dots, \ell_{i-1}$.*

Theorem 4. For every $\epsilon > 0$, there is a linear time $(1 + \epsilon)$ -approximation algorithm for SEMC on planar graphs.

Proof. A planar embedding of G can be found in linear time. Assume that this embedding is t -outerplanar for some integer t . Set $K = 4/\epsilon^2$. Denote by E_i the edges connecting layer i and $i + 1$. For $1 \leq i \leq K$, let G_i be the subgraph of G spanned by the edges in $E_i, E_{i+K}, E_{i+2K}, \dots$. Each E_i spans a 2-outerplanar graph, hence each G_i is 2-outerplanar. A theorem of Bodlaender [3, Theorem 83] assures that a t -outerplanar graph has treewidth at most $3t - 1$, therefore we can use Theorem 3 to find a $(1 + \epsilon)$ -approximate coloring Ψ_i for each graph G_i in linear time. Notice that the edges in the graphs G_i are pairwise disjoint. Therefore there is a $1 \leq i' \leq K$ such that the minimum sum on $G_{i'}$ is at most $\text{OPT}(G, x)/K$, and consequently, the cost of $\Psi_{i'}$ is at most $(1 + \epsilon)\text{OPT}(G, x)/K$.

Delete the edges of $G_{i'}$ from G , the resulting graph is clearly K -outerplanar. Therefore it has treewidth at most $3K - 1$, and Theorem 3 can be used to find a $(1 + \epsilon)$ -approximate coloring Ψ_1 . Perform an $(\epsilon, 1)$ -augmentation of Ψ_1 , this increases the sum by at most a factor of $(1 + \epsilon)$. We use extra zone Z_1 to color the edges in $G_{i'}$. If $\Psi_{i'}$ assigns color c to edge e , then assign to e the c th color of Z_1 . By Lemma 1b, the sum of the edges in $G_{i'}$ will be at most $2/\epsilon$ times the cost of $\Psi_{i'}$, that is, at most $2/\epsilon \cdot (1 + \epsilon)\text{OPT}(G, x)/K \leq \epsilon\text{OPT}(G, x)$. Therefore the resulting coloring is a $(1 + O(\epsilon))$ -approximate coloring for G . \square

References

1. B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. Assoc. Comput. Mach.*, 41(1):153–180, 1994.
2. A. Bar-Noy, M. M. Halldórsson, G. Kortsarz, R. Salman, and H. Shachnai. Sum multicoloring of graphs. *J. Algorithms*, 37(2):422–450, 2000.
3. H. L. Bodlaender. A partial k -arboretum of graphs with bounded treewidth. *Theoret. Comput. Sci.*, 209(1-2):1–45, 1998.
4. E. G. Coffman, Jr., M. R. Garey, D. S. Johnson, and A. S. LaPaugh. Scheduling file transfers. *SIAM J. Comput.*, 14(3):744–780, 1985.
5. M. M. Halldórsson and G. Kortsarz. Tools for multicoloring with applications to planar graphs and partial k -trees. *J. Algorithms*, 42(2):334–366, 2002.
6. M. M. Halldórsson, G. Kortsarz, A. Proskurowski, R. Salman, H. Shachnai, and J. A. Telle. Multicoloring trees. *Inform. and Comput.*, 180(2):113–129, 2003.
7. J. A. Hoogeveen, S. L. van de Velde, and B. Veltman. Complexity of scheduling multiprocessor tasks with prespecified processor allocations. *Discrete Appl. Math.*, 55(3):259–272, 1994.
8. J. Kahn. Asymptotics of the chromatic index for multigraphs. *J. Combin. Theory Ser. B*, 68(2):233–254, 1996.
9. D. Marx. Complexity results for minimum sum edge multicoloring. Manuscript.
10. D. Marx. The complexity of tree multicolorings. In *Mathematical Foundations of Computer Science 2002 (Warsaw-Otwock)*, pages 532–542. Springer, Berlin, 2002.
11. D. Marx. Minimum sum multicoloring on the edges of trees. In *1st International Workshop on Approximation and Online Algorithms (WAOA) 2003*, volume 2909 of *Lecture Notes in Computer Science*, pages 214–226. Springer, Berlin, 2004.
12. C. E. Shannon. A theorem on coloring the lines of a network. *J. Math. Physics*, 28:148–151, 1949.