Sum Multicoloring of Graphs*

Amotz Bar-Noy† Magnús M. Halldórsson‡ Guy Kortsarz§
Ravit Salman¶ Hadas Shachnai‖

*The extended abstract of this paper appeared in ESA '99 [BHK+99].

†Dept. of EE, Tel-Aviv University, Tel-Aviv 69978, Israel. E-mail: amotz@eng.tau.ac.il.
‡Science Institute, University of Iceland, IS-107 Reykjavik, Iceland. E-mail: mmh@hi.is.
§Dept. of Computer Science, Open University, Ramat Aviv, Israel. E-mail: guyk@tavor.openu.ac.il.
¶Dept. of Mathematics, Technion, Haifa 32000, Israel. E-mail: maravit@tx.technion.ac.il.
‖Dept. of Computer Science, Technion, Haifa 32000, Israel. E-mail: hadas@cs.technion.ac.il.

Proposed Running Head:

BAR-NOY ET AL., SUM MULTICOLORING OF GRAPHS.

Address for Proofs:

Dr. Hadas Shachnai
Department of Computer Science
The Technion IIT, Haifa 32000
Israel.

Abstract

Scheduling dependent jobs on multiple machines is modeled by the graph *multicoloring* problem. In this paper we consider the problem of minimizing the average completion time of all jobs. This is formalized as the *sum multicoloring* problem: Given a graph and the number of colors required by each vertex, find a multicoloring which minimizes the sum of the largest colors assigned to the vertices. It reduces to the known *sum coloring* problem when each vertex requires exactly one color.

This paper reports a comprehensive study of the sum multicoloring problem, treating three models: with and without preemptions, as well as co-scheduling where jobs cannot start while others are running. We establish a linear relation between the approximability of the maximum independent set problem and the approximability of the sum multicoloring problem in all three models, via a link to the sum coloring problem. Thus, for classes of graphs for which the independent set problem is ρ-approximable, we obtain $O(\rho)$-approximations for preemptive and co-scheduling sum multicoloring, and $O(\rho \log n)$-approximation for non-preemptive sum multicoloring. In addition, we give constant ratio approximations for a number of fundamental classes of graphs, including bipartite, line, bounded degree, and planar graphs.

Index Terms: Graph Coloring, Sum Coloring, Chromatic Sums, Multicoloring, Scheduling, Dependant Jobs, Dining Philosophers.

1 Introduction

Any multiprocessor system has certain resources, that can be made available to one job at a time. A fundamental problem in distributed computing is to efficiently schedule jobs that are competing for such resources. The scheduler has to satisfy the following two conditions: (i) *mutual exclusion*: no two conflicting jobs are executed simultaneously. (ii) *no starvation*: the request of any job to run is eventually granted. The problem is well-known in its abstracted form as the *dining/drinking philosophers* problem (e.g., [D68, L81, NS95, SP88]).

Scheduling dependent jobs on multiple machines is modeled as a graph *coloring* problem, when all jobs have the same (unit) execution times, and as graph *multicoloring* for arbitrary execution times. The vertices of the graph represent the jobs and an edge in the graph between two vertices represents a dependency between the two corresponding jobs, that forbids scheduling these jobs at the same time. More formally, for a weighted undirected graph $G = (V, E)$ with n vertices, let the *length* of a vertex v be a positive integer denoted by $x(v)$. We denote by p the maximum length in G, that is $p = \max_{v \in V} \{x(v)\}$. A multicoloring of the vertices of G is a mapping into the power set of the positive integers, $\Psi : V \mapsto 2^N$. Each vertex v is assigned a set of $x(v)$ distinct numbers (colors), and adjacent vertices are assigned disjoint sets of colors.

The traditional optimization goal is to minimize the total number of colors assigned to G. In the setting of a job system, this is equivalent to finding a schedule that minimizes the time for completing *all* the jobs. Another important goal is to minimize the *average* completion time of the jobs, or equivalently, to minimize the sum of the completion times. In the *sum multicoloring* (SMC) problem, we look for a multicoloring Ψ that minimizes $\sum_{v \in V} f_\Psi(v)$, where the completion time $f_\Psi(v)$ is the maximum color assigned to v by Ψ. We study the sum multicoloring problem in three models:

- In the Preemption model (pSMC), each vertex may get any set of colors.
- In the No-Preemption (npSMC) model, the set of colors assigned to each vertex has to be contiguous.
- In the Co-Scheduling model (coSMC), the vertices are colored in rounds: in each round the scheduler fully colors the vertices of an independent set in the graph. The set of colors assigned to each vertex has to be contiguous and whenever two vertices use the same color, their smallest color must also be the same.

The Preemption model corresponds to the scheduling approach commonly used in modern operating systems [SG98]: jobs may be interrupted during their execution and resumed at later time. The No-Preemption model captures the execution model adopted in real-time systems, where scheduled jobs must run to completion. The Co-Scheduling approach is used in some distributed operating systems [T95]. In such systems, the scheduler identifies subsets of non-conflicting or cooperating processes, that can benefit from running at the same time interval (e.g., since these processes do not use the same resources or communicate frequently with each other). Then, each subset is executed simultaneously on several processors, until *all* the processes in the subset complete.

The SMC problem has many other applications, including traffic intersection control [B92, BH94], session scheduling in local-area networks [CCO93], compiler design and VLSI routing [NSS99].

1.1 The sum coloring problem

When all the lengths are equal to 1, the problem, in all three models, reduces to the previously studied *sum coloring* (SC) problem. The first direct study of the problem is in [K89, KS89], where it is shown to be NP-hard, while a polynomial algorithm is given for the case where G is a tree. Lower and upper bounds for the color sum of a graph are given in [TEA$^+$89]. A $\overline{d}/2+1$-approximation is given in [KKK89] for graphs of average degree $\overline{d}$.

We note that coloring the graph with minimum number of colors does not always help in solving the SC problem. For instance, while bipartite graphs can be colored with two colors, there exist bipartite graphs (in fact, trees) for which any optimal solution for the sum coloring problem uses $\Omega(\log n)$ colors [KS89].

An exact algorithm is given in [J97] for partial k-trees (i.e. graphs of treewidth at most k). It holds for a more general problem called the Optimum Chromatic Cost Problem (OCCP). Here a cost function $c : Z^+ \mapsto Z^+$ is associated with the color classes, and we are to assign a (single) color $f(v)$ to each vertex v with the objective of minimizing $\sum_{v \in V} c(f(v))$. In the SC problem we have $c(f(v)) = f(v)$. A polynomial algorithm was given for the OCCP on partial k-trees.

Recent papers have focused on finding approximation algorithms. The SC problem is considered in detail in [BBH$^+$98], including bounds on general graphs, $(\Delta+2)/3$-approximation of graphs of maximum degree Δ, and 2-approximation of line graphs. An improved 10/9-approximation was given in [BK98] for bipartite graphs, and a 2-approximation is given in [NSS99] for interval graphs.

Intuitively, a good sum coloring would tend to use the smaller colors for as many vertices as possible. The following natural heuristic was proposed in [BBH$^+$98].

MAXIS: Choose a maximum independent set in the graph, color all of its vertices with the next available color; iterate until all vertices are colored.

This procedure was shown to yield a 4-approximation. This factor of 4 was shown to be tight in [BHK99], where a family of graphs was constructed for which the approximation ratio of MAXIS is asymptotically 4. In the case that the size of a maximum independent set can only be *approximated* within a factor of ρ, MAXIS yields a 4ρ-approximation.

Known hardness results for the sum coloring problem carry over to the sum multicoloring problem. SC cannot be approximated in general within an $n^{1-\epsilon}$ factor, for any $\epsilon > 0$, unless $NP = ZPP$ [FK96, BBH+98]. Also, it is NP-hard to approximate within some factor $c > 1$ on bipartite graphs [BK98]. Furthermore, the SC problem is NP-hard on interval graphs [S99], and on planar graphs [HK99].

1.2 Our Results for the Sum Multicoloring problem

This paper reports a comprehensive study of the sum multicoloring problem. We detail below our results.

General graphs: We first inspect the MAXIS algorithm which is a natural candidate heuristic also for the SMC problems. We show that MAXIS is only an $\Omega(\sqrt{p})$ approximation algorithm for pSMC (recall that p is the largest length in the graph). For the npSMC problem, its performance cannot even be bounded in terms of p.

In view of that, we resort to different algorithms. In our central results we establish a linear relation between the approximability of the *weighted* maximum independent set problem (WIS) and SMC in all three models, via a link to the SC problem and the MAXIS algorithm. The reductions are universal in that they apply to any hereditary family of graphs. For classes of graphs where WIS is polynomially solvable, we obtain a 16-approximation to both pSMC and coSMC, and a $O(\log \min(n,p))$-approximation to npSMC. The linearity of the reduction implies that for classes of graphs where WIS is ρ-approximable, the ratios are $O(\rho)$ for pSMC and coSMC and $O(\rho \log \min(n,p))$ for npSMC.

Applying known results on the solvability and the approximability of WIS, we get for the pSMC approximation ratio of 16 on perfect graphs [GLS88], $O(\Delta \log\log \Delta / \log \Delta)$ on bounded-degree graphs [H99], and $O(n/\log^2 n)$ on arbitrary graphs [BH92, H99].

Important special classes of graphs: We further study special classes of graphs. For pSMC, we describe an algorithm with a ratio of 1.5 for bipartite graphs. We generalize this to an algorithm with a ratio of $(k+1)/2$ on k-colorable graphs, when the coloring is given. This, for instance, gives a relatively simple and practical algorithm with a ratio of 2.5 for pSMC on planar graphs. We also present an algorithm with ratios of $(\Delta+2)/3$ for graphs of maximum degree Δ, 2 for line graphs, and k for intersection graphs of k-uniform hypergraphs.

For npSMC, we give an algorithm with ratios of 2.796 on bipartite graphs and $1.55k+1$ on k-colorable graphs (with the k-coloring given). These colorings have the advantage of also approximating comparably the number of colors they use.

Relationships among the multicoloring models: We explore the relationship among the three models and give a construction that indicates why the npSMC model is "harder". Namely, while finding independent sets iteratively suffices to approximate both the pSMC and the coSMC problems by a constant factor, any such solution must be $\Omega(\log p)$ off for the npSMC problem.

1.3 Organization of the Paper

In Section 2 we give some definitions and notations. In Section 3 we present the approximation results for general graphs, in all three models. In Section 4 we deal with bipartite graphs both for the PREEMPTION and NO-PREEMPTION models. These results are generalized to k-colorable graphs, when the coloring is given. Finally, in Section 5 we address bounded-degree graphs and line graphs, primarily for the PREEMPTION model.

2 Definitions and notations

When we refer to a graph $G=(V,E)$, we mean a simple, finite, undirected graph, and implicitly assume an associated length mapping $x : V \mapsto N$. Denote by $\mathcal{S}(G) = \sum_v x(v)$ the sum of the lengths of the vertices in G. An *independent set* in G is a subset I of V such that any two vertices in I are nonadjacent. A *multicoloring* of G is an assignment $\Psi : V \mapsto 2^N$, such that each vertex $v \in V$ is assigned $x(v)$ distinct colors and the set of vertices colored by any color i is independent.

Given a *multicoloring* Ψ of G, denote by $c_1^{\Psi}(v) < \cdots < c_{x(v)}^{\Psi}(v)$ the ordered collection of $x(v)$ colors assigned to v, and by $f_{\Psi}(v) = c_{x(v)}^{\Psi}(v)$ the largest color assigned to v. The *multicolor sum* of G with respect to Ψ is

$$\mathrm{SMC}_\Psi(G) = \sum_{v\in V} f_\Psi(v) \ .$$

Let $\mathrm{SMC}_\mathcal{A}(G)$ denote the multicolor sum of the coloring produced by algorithm $\mathcal{A}$ on G.

A multicoloring Ψ is *contiguous* (nonpreemptive), if for any $v \in V$, the colors assigned to v satisfy $c^\Psi_{i+1}(v) = c^\Psi_i(v) + 1$ for $1 \le i < x(v)$. In the context of scheduling, this means that all the jobs are processed without interruption. A contiguous multicoloring Ψ solves the Co-Scheduling problem, if the set of vertices can be partitioned into k disjoint independent sets $V = I_1 \cup \cdots \cup I_k$ with the following two properties: (i) $c^\Psi_1(v) = c^\Psi_1(v')$ for any $v, v' \in I_j$, $1 \le j \le k$, and (ii) $c^\Psi_{x(v)}(v) < c^\Psi_1(v')$ for all $v \in I_j$ and $v' \in I_{j+1}$, $1 \le j < k$. In the context of scheduling, this means scheduling to completion all the jobs corresponding to I_j, and only then starting to process the jobs in I_{j+1}, for $1 \le j < k$.

The *minimum multicolor sum* of a graph G, denoted by $\mathrm{pSMC}(G)$, is the minimum $\mathrm{SMC}_\Psi(G)$ over all multicolorings Ψ. We denote the minimum contiguous multicolor sum of G by $\mathrm{npSMC}(G)$. The minimum multicolor sum of G for the Co-Scheduling problem is denoted by $\mathrm{coSMC}(G)$. When the model under investigation is clear from the context, we denote these numbers by $\mathrm{SMC}(G)$. We further omit G when the graph under investigation is clear from the context.

Since in any multicoloring of G, $x(v) \le f_\Psi(v)$ for all vertices, it follows that $\mathcal{S}(G) \le \mathrm{pSMC}(G)$. Since any co-schedule of G is a nonpreemptive coloring of G and any nonpreemptive coloring of G is also preemptive, it follows that $\mathrm{pSMC}(G) \le \mathrm{npSMC}(G) \le \mathrm{coSMC}(G)$. Hence, we get the following proposition.

Proposition 2.1 *For any graph G,*

$$\mathcal{S}(G) \le \mathrm{pSMC}(G) \le \mathrm{npSMC}(G) \le \mathrm{coSMC}(G) \ . \qquad \square$$

Let P be the SMC problem in any of the three models. We say that an algorithm $\mathcal{A}$ *approximates P within a ratio of ρ*, if for all graphs G (or for all graphs belonging to a specified family of graphs) we have that

$$\frac{\mathrm{SMC}_\mathcal{A}(G)}{\mathrm{SMC}(G)} \le \rho \ ,$$

where $\mathrm{SMC}(G)$ is the optimal multicolor sum for P on G. We also say then that $\mathcal{A}$ gives a *ρ-approximation*. This is the traditional performance ratio; we consider also two stricter measures. An algorithm is said to approximate within an *absolute ratio* of ρ, if

$$\frac{\mathrm{SMC}_\mathcal{A}(G)}{\mathcal{S}(G)} \le \rho \ .$$

An algorithm is said to approximate within a *service-time ratio* of ρ, if

$$\text{For all } v \in V, \ \ \frac{f_\mathcal{A}(v)}{x(v)} \le \rho \ ,$$

that is, the highest color assigned to v is at most $\rho\, x(v)$. It follows that if an algorithm has a service-time ratio ρ, then it also has an absolute ratio ρ, and if an algorithm has an absolute ratio ρ, it also has an approximation ratio ρ.

3 General graphs

In this section we describe a reduction from approximating sum multicoloring in all three models to the problem of approximating weighted maximum independent sets (WIS). This generalizes (using different techniques) a relationship derived for the sum coloring problem [BBH+98]. In this and the following sections, ρ denotes the approximability of WIS on the input graph.

In Section 3.1 we analyze the MAXIS algorithm. We show that MAXIS performs poorly on all three SMC problems. We next describe our main approximation technique in Section 3.2 that involves a different objective function, the *sum of averages* of a multicoloring. We show that the minimum sum of averages of a given graph can be well approximated using the MAXIS algorithm on a derived graph. In Section 3.3 we give an $O(\rho)$-approximation for pSMC; in this and the following sections, ρ denotes the approximation ratio of WIS on the input graph. In Sections 3.4 and 3.5 we present and analyze an algorithm that approximates both npSMC and coSMC — the former within a ratio of $O(\rho \log \min(p, n))$ and the latter within $O(\rho)$. We discuss the time complexity of our algorithms in Section 3.6.

In attempting to explain the logarithmic factor in the approximation of npSMC, we note that the algorithm described in Section 3.5 constructs a co-schedule which in the analysis is compared with an optimal *preemptive* schedule. In Section 3.7, we show that this discrepancy in the performance is inherent by constructing a graph H for which $\mathrm{coSMC}(H) = \Omega(\log p) \cdot \mathrm{npSMC}(H)$.

3.1 The MAXIS heuristic

We consider here the MAXIS heuristic applied to the sum multicoloring in the three models, and show that its performance is not up to par with the sum coloring case. This leads us to modify the heuristic in the following section to better take into account the lengths of the jobs.

In the preemptive case there are several possible definitions for MAXIS. One natural definition for MAXIS is to repeat, as long as needed, the following procedure: color a maximum independent set with one color, then reduce the lengths of all the vertices in the chosen set by one. The following example shows that such an algorithm has an approximation ratio of $\Omega(\sqrt{p})$.

Let G be the following graph with $\sqrt{p} + 2$ vertices: two nonadjacent vertices with length

p, and a clique of $\sqrt{p}$ vertices of length 1. The two special vertices are connected to all the vertices of the clique. MAXIS colors first the two special vertices with p colors and then colors in a sequence each vertex in the clique with one color. This yields a multicolor sum of

$$2p + (p+1) + (p+2) + \cdots + (p + \sqrt{p}) = \Omega(p\sqrt{p}) \ .$$

On the other hand, the optimal solution schedules the clique vertices first. This yields a multicolor sum of

$$1 + 2 + \cdots + \sqrt{p} + 2(p + \sqrt{p}) = O(p) \ .$$

Thus, the approximation ratio is $\Omega(\sqrt{p})$.

In the Co-Scheduling model, MAXIS runs its selected jobs to completion before finding the next set of independent jobs. Consider the tree with vertices v_i, $1 \leq i \leq 2t+1 = n$, and the edges $\{v_i v_{t+i}, v_n v_{t+i} : i = 1, \ldots, t\}$. All vertices have unit length except v_n that requires p colors. MAXIS first colors $v_1, v_2, \ldots, v_t$ and v_n. As a result, the first color for vertices $v_{t+1}, v_{t+2}, \ldots, v_{2t}$ is $p+1$. The cost of this coloring is $t \cdot 1 + p + t \cdot (p+1) \geq np/2$. The optimal solution assigns vertices $v_1, v_2, \ldots, v_t$ color 1, vertices $v_{t+1}, v_{t+2}, \ldots, v_{2t}$ color 2, and vertex v_n colors $3, 4, \ldots, p+2$. The cost of that coloring is $t \cdot 1 + t \cdot 2 + (p+2) \leq 2(n+p)$. Hence, the approximation ratio is $\Omega(\min(n,p))$.

In the No-Preemption model, the algorithm may do still worse. Whenever the set of currently colored vertices becomes a non-maximal independent set, a natural MAXIS in this model would find the largest set of vertices that the solution can be extended with. Consider the graph G composed of independent sets A, B, and C, each with $t = n/3$ vertices. The edges between A and B are given by $\{a_i b_j : |i - j| > 1\}$. Vertices of C are connected to all vertices of $A \setminus \{a_1\}$ and $B \setminus \{b_1\}$. Vertices of A and B have length 2, while vertices of C have length 1. Observe that, for $t > 2$, $C \cup \{a_1, b_1\}$ forms the only maximum independent set in the graph, hence it gets colored in the first iteration of MAXIS. In iteration 2, only a_1 and b_1 are being scheduled and the only vertices nonadjacent to both of them are a_2 and b_2, which are therefore scheduled at that point. In iteration 3, similarly, a_1 and b_1 have been completed and a_2 and b_2 are still being scheduled. Therefore, only a_3 and b_3 can be added. In general, vertices a_i and b_i start to get colored in iteration i. The cost of the schedule is then $t + 2\sum_{i=2}^{t+1} i = \Omega(t^2)$. A better schedule colors first the vertices of C, then the vertices of A, and finally the vertices of B. The total cost is then $t + 3t + 5t = O(t)$. Hence, the approximation ratio is $\Omega(t) = \Omega(n)$, independent of p.

3.2 The main approximation technique

En route to approximating multicoloring sums, we consider another measure of a multicoloring, the sum of the *average* color values assigned to the vertices. We denote by $\mathrm{AV}_\Psi(v)$ the average color assigned to v by a multicoloring Ψ, namely,

$$\mathrm{AV}_\Psi(v) = \frac{\sum_{i=1}^{x(v)} c_i^\Psi(v)}{x(v)} .$$

Let $\mathrm{SA}_\Psi(G) = \sum_v \mathrm{AV}_\Psi(v)$ denote the sum of averages of Ψ, and let $\mathrm{SA}(G)$ be the minimum possible sum of averages. Note that $\mathrm{SA}(G) \leq \mathrm{pSMC}(G)$, and equality holds only in the unit-length case.

We approximate the sum of averages by a reduction to the sum coloring problem on a derived weighted graph. Recall, that ρ denotes the approximability of WIS on the input graph G.

Lemma 3.1 *For any graph G, there exists a polynomially constructible multicoloring Ψ satisfying*

$$\mathrm{SA}_\Psi(G) \leq 4\rho \cdot \mathrm{SA}(G) . \tag{1}$$

Proof. Given a graph G, we construct a *derived weighted graph H* as follows. The graph has a clique of $x(v)$ copies of each vertex v, with each copy of v adjacent to all copies of neighbors of v in G. The weight $w(v_i)$ of each copy v_i of v is $1/x(v)$.

There is a one-to-one correspondence between multicolorings Ψ of G and colorings of H, as the $x(v)$ copies of a vertex v in G all receive different colors. Let Ψ also refer to the corresponding coloring of H. Observe that

$$\mathrm{SC}_\Psi(H) = \sum_{v \in V(G)} \sum_{i=1}^{x(v)} w(v_i)\, c_i^\Psi(v) = \sum_{v \in V(G)} \frac{\sum_{i=1}^{x(v)} c_i^\Psi(v)}{x(v)} = \mathrm{SA}_\Psi(G) .$$

Using the result of [BBH+98] on the performance of MAXIS, which holds also for the weighted case, we obtain the same bound on the approximation of the average completion time.

Observe that each iteration of MAXIS can be applied to a subgraph of H that contains only one copy of each vertex v of G (that has not yet been fully colored), since two copies of the same vertex cannot participate in the same independent set. This is especially significant when ρ, the approximability of WIS, is a function of the number of vertices; hence, a $\rho(n)$-approximation of WIS implies a $4\rho(n)$-approximation of SA. □

3.3 The Preemption model

Our approach for approximating pSMC is to transform multicolorings with small sum of averages into ones with nearly equally small sum of finish times. Note that this is not always the case, as a coloring Ψ with a small sum of averages may assign a few high colors to the vertices, thus resulting in poor finishing times. We resolve this problem by using the vertex sets of the colors of Ψ twice, which ensures that the finish time of a vertex in the new coloring is at most four times its average color in Ψ.

Theorem 3.2 pSMC *can be approximated within a ratio of* 16ρ.

Proof. First, obtain a multicoloring Ψ by applying Lemma 3.1 on the input graph G. Next, form the multicoloring Ψ' that "doubles" certain color sets of Ψ:

$$\Psi'(v) = \left\{2i, 2i+1 : i = c_t^{\Psi}(v), t \leq \lceil x(v)/2 \rceil \right\} .$$

Note that $\Psi'(v)$ may contain $x(v)+2$ colors, in which case we assign to v the subset of $x(v)$ smallest colors.

Let $m_v = c^{\Psi}_{\lceil x(v)/2 \rceil}$ be the median color assigned to v by Ψ. The largest color used by Ψ' is $f_{\Psi'}(v) = c^{\Psi}_{\lceil x(v)/2 \rceil} + c^{\Psi}_{\lfloor x(v)/2 \rfloor} \leq 2m_v$. Also $m_v \leq 2 \cdot AV_{\Psi}(v)$, since fewer than half the elements of a set of natural numbers can be larger than twice its average. Thus $f_{\Psi'}(v) \leq 4 \cdot AV_{\Psi}(v)$ and $\text{SMC}_{\Psi'}(G) \leq 4 \cdot \text{SA}_{\Psi}(G)$. By Lemma 3.1, $\text{SA}_{\Psi}(G) \leq 4 \cdot \text{SA}(G) \leq 4 \cdot \text{pSMC}(G)$, thus Ψ' is a 16ρ-approximation for pSMC(G). □

3.4 The No-Preemption model

The following algorithm LengthGrouping (LG) approximates solutions to both the npSMC and coSMC problems. The idea is to separate vertices of roughly the same lengths into groups, and to ensure that at any given time vertices from only one such group are being colored. Since coloring vertices with large lengths yields relatively limited progress per time unit, we assign to such vertices *small* weights. Specifically, the weight of each vertex is inversely proportional to its length. Then, we run the MAXIS algorithm on the resulting weighted graph, and for each independent set found, color the jobs to completion.

LG(G)

Let $G' = (V, E', x', w)$, where

$x'(v) = 2^{\lceil \log x(v) \rceil}$, for each $v \in V$,

$w(v) = 1/x'(v)$, the weight of each $v \in V$, and

$E' = E \cup \{(u,v) : x'(u) \neq x'(v)\}$.

Apply weighted MAXIS to G', coloring the vertices nonpreemptively in each iteration.

Let Ψ denote the multicoloring produced by LG. Note that Ψ is a co-schedule, since jobs are colored to completion. Since G' is a supergraph of G and the lengths in G' are upper bounds on the lengths in G, Ψ is also a valid multicoloring of G. By a *weight-class* (or *length-class*), we mean a set of vertices in G' of the same weight (and length).

Lemma 3.3 $\mathrm{SMC}_\Psi(G) \leq 4\rho \cdot \mathrm{pSMC}(G')$.

Proof. First, note that for any coloring Ψ, the largest color of a vertex differs from the average color by at least half the length of the vertex, i.e. $f_\Psi(v) \geq AV_\Psi(v) + (x(v)-1)/2$, with equality holding when Ψ is contiguous. Thus,

$$\mathrm{SMC}_\Psi(G') = \mathrm{SA}_\Psi(G') + \frac{\mathcal{S}(G') - n}{2} \ . \tag{2}$$

Also, for a coloring Ψ^* that minimizes $\mathrm{pSMC}(G')$ it holds that

$$\mathrm{SA}(G') \leq \mathrm{SA}_{\Psi^*}(G') \leq \mathrm{pSMC}(G') - \frac{\mathcal{S}(G') - n}{2} \ . \tag{3}$$

Note that the coloring Ψ produced by LG satisfies Lemma 3.1. Thus, combining (2) and (3) with (1) we have that

$$\mathrm{SMC}_\Psi(G') = SA_\Psi(G') + \frac{\mathcal{S}(G') - n}{2} \leq 4\rho \cdot \mathrm{SA}(G') + \frac{\mathcal{S}(G') - n}{2} \leq 4\rho \cdot \mathrm{pSMC}(G') \ .$$

Finally, the cost of Ψ on G is at most its cost on G', since it assigns $2^{\lceil x(v) \rceil} \geq x(v)$ colors to each vertex. □

Lemma 3.4 LG *approximates* npSMC *within a ratio of* $8\rho \log p$.

Proof. We claim that

$$\mathrm{pSMC}(G') \leq 2 \log p \cdot \mathrm{pSMC}(G) \ .$$

The lemma then follows from Lemma 3.3.

Consider a multicoloring Ψ of G. Apply it to the vertices of G', by first repeating each color twice to account for the rounding up of the lengths. Observe that the vertices of individual length classes of G' are properly colored and are pairwise consistent, while vertices from different length classes may conflict. We can combine the colorings of the individual length classes into a coloring of G' by executing one step from each class in turn, in a round-robin fashion. The coloring of each vertex is slowed down by a factor of $\log p$, the number of length classes. Then, we have constructed a multicoloring of G' whose sum is at most $2 \log p \cdot \mathrm{pSMC}(G)$. □

Let r be the number of different lengths in the graph, and let q be the ratio between the largest to the smallest length. Using r and q we now derive a tighter performance bound for LG.

Theorem 3.5 LG *approximates* npSMC *within a ratio of* $O(\rho \min\{r, \log \min(q, n)\})$.

Proof. First, note that the graph can be separated to r length groups; this yields the $O(r)$-approximation. We now show, that the $O(\log p)$ ratio in Lemma 3.4 can be replaced with $min(\log q, \log n)$. The $\log q$ ratio is obtained by scaling; the $O(\log n)$ bound clearly holds when p is polynomial in n. When p is superpolynomial, we add to LG an initial phase, in which we color a subset of the vertices; then, we apply LG on the remaining vertices, which form $O(\log n)$ length groups. More specifically, let $Small = \{v \mid x(v) \leq p/n^3\}$. Consider coloring first the vertices of $Small$ in an arbitrary order nonpreemptively. The color sum of the vertices in $Small$ is at most $(1 + 2 + \ldots + n)p/n^3 < p$. Also, they use at most $n \cdot p/n^3 = p/n^2$ colors; this adds to the cost of coloring each of the remaining vertices p/n^2, i.e., we add to the total cost at most p/n. Hence, we get an additive term of at most $1/n$ in the approximation ratio.

Algorithm LG now yields a performance ratio $O(\log n)$ for npSMC, since at most $\log(p/(p/n^3)) = 3 \log n$ length-classes remain in the graph. □

3.5 The Co-Scheduling model

We find that LG yields a better approximation of the co-scheduling problem. Recall that G' is the derived weighted graph produced by LG.

Theorem 3.6 LG *approximates* coSMC *within a ratio of* 16ρ.

The theorem follows by combining Lemma 3.3, the fact that $\text{pSMC}(G') \leq \text{coSMC}(G')$, and the following lemma, which shows that the optimal co-schedule of G' is within a constant factor of that of the original graph G.

Lemma 3.7 $\text{coSMC}(G') \leq 4 \cdot \text{coSMC}(G)$.

Proof. Call an algorithm *proper* if the algorithm never gives the same color to vertices from different weight-classes. For example, LG is proper. Let OPT denote an optimal co-schedule for G. Let X be the length of the longest job in a given round of OPT. Rounding the length to a power of two results in a length $X' = 2^{\lceil \log X \rceil}$. Note that in contrast to LG, OPT is not necessarily proper. However, we can simulate one round of OPT by at most $\log p$ rounds of a proper algorithm, one for each weight-class.

Consider the schedule that breaks a round of OPT into a sequence of rounds, of length $1, 2, 4, \ldots, X'$. That is, the algorithm takes the independent set of vertices chosen by OPT and schedules them properly, one weight-class after the other. This is a valid proper schedule of G', where each vertex is delayed by at most a factor of 4, in comparison to OPT. This follows since $1 + 2 + 4 + \ldots + X' < 4X$. Since any proper algorithm is also a co-schedule algorithm, $\mathrm{coSMC}(G') \leq 4 \cdot \mathrm{coSMC}(G)$, and the lemma follows. □

3.6 Time complexity

When the maximum job length is super-polynomial in the length of the input, the derived weighted graph G' to which MAXIS is applied also becomes super-polynomial. It is however not necessary to construct the whole graph.

At most one copy of any node is selected in an iteration of MAXIS. Thus, before each iteration we build the weighted graph based on remaining nodes. Without loss of generality, MAXIS repeats the same independent set as long as none of its nodes are fully colored. We can therefore handle r_i color classes in each iteration i, where r_i is the smallest length remaining in that iteration. After each iteration, a new vertex becomes fully colored. Hence, the number of iterations is at most n. The time complexity is therefore $O(nf(n))$, where $f(n)$ is the time complexity of the maximum independent set algorithm.

It also follows that any coloring output by our algorithms can be represented in $O(n^2)$ space.

3.7 A construction separating coSMC and npSMC

We have given $O(1)$-approximations for both the pSMC and the coSMC problems, while only a $\min\{O(\log p), O(\log n)\}$-approximation for the npSMC problem. Notice that the analysis in Lemma 3.4 rates LG, that in fact produces a co-schedule, in terms of a stronger adversary that finds the best possible *preemptive* schedule. This implies the following inequalities:

$$\mathrm{pSMC}(G) \leq \mathrm{npSMC}(G) \leq \mathrm{coSMC}(G) \leq O(\log \min\{p, n\}) \cdot \mathrm{pSMC}(G) \ . \tag{4}$$

It would be interesting to know the precise relationships among these three models. We give a partial answer by constructing a graph H for which $\mathrm{coSMC}(H) = \Omega(\log p) \cdot \mathrm{npSMC}(H)$, matching inequality (4).

Assume for simplicity that p is a power of 2. The graph $H = (V, E)$ is as follows. The vertex set is $V = \{v_{i,j,k}\}$, for $i = 0, 1, \ldots, \log p$, $j = 1, 2, 3, \ldots, 2^{(\log p) - i}$, and $k = 1, 2, \ldots, 2^i$, where the length of $v_{i,j,k}$ is 2^i. The edge set is $E = \{(v_{i,j,k}, v_{i',j',k'}) : i = i' \text{ and } k = k'\}$. In other words, the graph has p vertices of each length $\ell = 2^i$, arranged in p/ℓ completely connected independent sets of size ℓ each; vertices of different lengths are nonadjacent.

Intuitively, sets containing "short" vertices (i.e. with "small" lengths) are decomposed into a large collection of independent subsets of vertices. Each such independent subset contains "few" vertices. On the other hand, each set of vertices with a given "large" length is decomposed into a "small" collection of independent subsets of vertices, where each independent set contains "many" vertices.

Now, consider a co-scheduling algorithm. Suppose that the maximum length chosen in some round is a "large" ℓ. This round delays "many" short vertices which must wait for the ℓ time units to pass. For example, there are $p/2$ size-2 independent sets of length 2. All but one of these (still remaining) sets is delayed ℓ time-units. Instead, in the Preemption or No-Preemption model, we could schedule those $p/2$ size-2 independent sets one after the other.

Indeed, consider the straightforward nonpreemptive coloring where the different lengths are processed independently and concurrently. Then, the number of colors used is p, and since the graph has $p \log p$ vertices, the multicoloring sum is $O(p^2 \log p)$.

On the other hand, any independent set contains at most 2^i vertices from length-class $\ell = 2^i$. Hence, in any independent set whose longest task length is ℓ, there are at most 2ℓ vertices. This follows since vertices with length larger than ℓ were not chosen into this set, plus, the number of vertices of length less than ℓ in the chosen independent set sum to at most $\ell/2+\ell/4+\cdots+1 < \ell$. Thus, at most $2t$ vertices are completed by step t, for each $t = 1, 2, \ldots, p \log p/2$. In particular, at most half of the vertices are completed by step $p \log p/4$. Thus, the color sum of the remaining vertices is $\Omega(p^2 \log^2 p)$. Hence, an $\Omega(\log p)$ separation between these models. As $n = p \log p$, this result also implies an $\Omega(\log n)$-separation.

4 Bipartite graphs and k-colorable graphs

In this section, we turn our attention to graphs of small chromatic numbers. Since it is in general hard to color even 3-colorable graphs efficiently, we assume that we are either given a particular k-coloring or that a k-coloring is easy to compute for the given graph. In addition, our results apply to graphs, in which a maximum independent set can be found in polynomial time. For a given k, let the set of vertices V be partitioned into k disjoint independent sets $V = V_1 \cup \cdots \cup V_k$. In Section 4.1 we study an algorithm for the Preemption model and in Section 4.2 we study another algorithm for the No-Preemption model and the Co-Scheduling model using different techniques.

4.1 The Preemption model

First consider the following Round-Robin (RR) algorithm. At round $t = kh + i$, for $1 \leq i \leq k$ and $h \geq 0$, give color t to all the vertices of V_i that still need a color. Since each vertex v is colored in every k-th time-slot, $f_{RR}(v) \leq k\,x(v)$. Thus, RR has a service-time ratio of k. In particular, it has service-time ratios of 4 on planar graphs and $\Delta + 1$ on graphs of maximum degree Δ. For bipartite graphs, its ratio is 2.

In this subsection we give a nontrivial algorithm with a ratio of $(k+1)/2$, for k-colorable graphs with the coloring given. This gives a ratio of 2.5 for planar graphs, and $k/2+1$ for partial k-trees. In particular, for bipartite graphs the ratio is $1.5 - 1/(2n)$. For simplicity, the result is described for bipartite graphs only, with the result for general k derived similarly.

We need the following definitions and notations. Denote by $\alpha(G)$ the size of a maximum independent set in G. *Processing* an independent set $W \subseteq V$ means assigning the next available color to each v in W, decreasing $x(v)$ by one, and removing fully colored vertices from the graph. The resulting graph is called the *reduced graph* of G. Finally, let $\gamma(n) = 2n^2/(3n-1)$.

Informally, algorithm BipartiteColor (BC) distinguishes between two cases. When the size of the maximum independent set in the current reduced graph is "large", then BC processes a maximum independent set. Otherwise, BC works in a fashion similar to RR. Once a vertex (or a collection of vertices) is assigned its required number of colors, the algorithm re-evaluates the situation.

BC

Let $G = (V_1, V_2, E)$ be a bipartite graph.
Let $n = |V| = |V_1| + |V_2|$.
while $G \neq \emptyset$ do
  if $\alpha(G) \leq \gamma(n)$
  then
    Let $m = \min_{v \in V} x(v)$.
    Assume without loss of generality that V_1 contains at least as many
      vertices v with $x(v) = m$ as V_2.
    Assign the next m colors to the vertices in V_1,
      and the following m colors to the vertices in V_2.
  else
    Find a maximum independent set $I \subseteq V$.
    Let $m = \min_{v \in I} x(v)$.
    Assign the next m colors to the vertices in I.

endif

Reduce the graph G, omitting vertices that were fully colored.

Update the value of n.

end while

Algorithm BC runs in polynomial time, since a maximum independent set can be found in a bipartite graph in polynomial time, using flow techniques (cf., [H69]), and since in each iteration at least one vertex is deleted.

Before we prove the approximation ratio of BC, we need the following observations.

Observation 4.1 *Let H, H' be identical graphs with respective length functions x, x', such that $x'(v) \geq x(v)$, for each $v \in V$. Then,* $\text{pSMC}(H') \geq \text{pSMC}(H) + (\mathcal{S}(H') - \mathcal{S}(H))$.

Proof. First, suppose that H and H' differ only in one vertex v and in one unit. Namely, $x'(v) = x(v) + 1$, and for any $u \neq v$, $x'(u) = x(u)$. Construct a multicoloring Ψ for H as follows. Use an optimal multicoloring of H', except remove v from the last independent set in which v appeared. The coloring Ψ is feasible for H. In this way $f(v)$ is decreased by at least one unit, and we get that $\text{pSMC}(H') \geq \text{pSMC}(H) + 1$. Now, the observation is deduced by repeatedly applying the above. □

For the next observation, let Ψ be some multicoloring and $V' \subseteq V$. Suppose that Ψ assigns the first i colors only to vertices in V' and that $x(v) \geq i$, for all $v \in V'$. Let G' be the resulting reduced graph. Finally, let $\text{SMC}_\Psi(G')$ be the multicolor sum of Ψ restricted to G'. Namely, using independent sets $i+1, i+2, \ldots$ of Ψ, relabeled as $1, 2, \ldots$.

Observation 4.2 $\text{SMC}_\Psi(G) = n \cdot i + \text{SMC}_\Psi(G')$.

Proof. The added cost of each vertex in G over that in G' is exactly i. This holds for vertices in V', since their lengths have reduced by exactly i. It also holds for vertices in $V - V'$, since these were not colored in the first i steps of Ψ, while the color values have decreased by i. □

We are now ready to prove the main theorem of this subsection.

Theorem 4.3 BC *approximates* pSMC *on bipartite graphs within a ratio of* $1.5 - 1/2n$.

Proof. We prove the theorem by induction on $\mathcal{S}(G)$. The basis of the induction is $\mathcal{S}(G) = 1$. Since we assume that the lengths are nonnegative integers, $\mathcal{S}(G) = 1$ implies that the graph contains only a single vertex, whose length is 1. BC is optimal in this case. For $\mathcal{S}(G) \geq 2$, we consider separately the two cases in the algorithm.

Case 1: $\alpha(G) \le \gamma(n)$.

Recall that, without loss of generality, BC first colors vertices of V_1. Let $V_1(m)$ be the subset of vertices in V_1 whose length is m. Let G' be the reduced graph after the first $2m$ colors are assigned. By definition, all lengths decreased by m, and vertices in $V_1(m)$ were omitted. That is, for all v, the residual length is $x'(v) = x(v) - m$. We can apply Observation 4.2 twice, as all vertices are delayed by the first m steps, while the vertices of $V \setminus V_1(m)$ are delayed in the second m steps. Thus, using that $|V_1(m)| \ge 1$,

$$\mathrm{SMC}_{\mathsf{BC}}(G) = (n + (n - |V_1(m)|)m + \mathrm{SMC}_{\mathsf{BC}}(G') \le (2n-1)m + \mathrm{SMC}_{\mathsf{BC}}(G') \ . \qquad (5)$$

Now, let G^* be the reduced graph after an optimal multicoloring has assigned m colors, with reduced lengths $x^*(v)$. Since $x(v) \ge m$ for all v, we have by Observation 4.2 that

$$\mathrm{pSMC}(G) = mn + \mathrm{pSMC}(G^*) \ . \qquad (6)$$

Note that for any $v \in V$, $x(v)$ was decreased by at most m, that is, $x^*(v) \ge x(v) - m$. Since in G' every $x(v)$ was decreased by m, $x^*(v) \ge x'(v)$. Therefore, by Observation 4.1, $\mathcal{S}(G^*) \ge \mathcal{S}(G) - m\alpha(G)$, since $\mathcal{S}(G)$ is reduced by at most $\alpha(G)$ in each of the first m colors. On the other hand, $\mathcal{S}(G') = \mathcal{S}(G) - mn$, since in G' we have reduced the length of *each* vertex in G by m. Thus,

$$\mathcal{S}(G^*) - \mathcal{S}(G') \ge (n - \alpha(G))m \ . \qquad (7)$$

Combining Equations (6) and (7) and Observation 4.1, we get that

$$\mathrm{pSMC}(G) \ge (2n - \alpha)m + \mathrm{pSMC}(G') \ . \qquad (8)$$

We now apply the induction hypothesis and get that

$$\frac{\mathrm{SMC}_{\mathsf{BC}}(G')}{\mathrm{pSMC}(G')} \le 1.5 - 1/(2n) \ . \qquad (9)$$

By the assumption that $\alpha(G) \le \gamma(n)$, it follows that

$$\frac{(2n-1)m}{(2n - \alpha(G))m} \le 1.5 - 1/(2n) \ . \qquad (10)$$

Combining Equations (5), (8), (9) and (10), we get that $\mathrm{SMC}_{\mathsf{BC}}(G)/\mathrm{pSMC}(G) \le 1.5 - 1/(2n)$, as required.

Case 2: $\alpha(G) > \gamma(n)$.

Let G' be the reduced graph after BC assigns the first m colors to the vertices of a maximum independent set I. Using Observation 4.2 with $V' = I$ we have that

$$\mathrm{SMC}_{\mathsf{BC}}(G) = nm + \mathrm{SMC}_{\mathsf{BC}}(G') \ . \qquad (11)$$

By definition, for every $v \in V$, $x(v) \geq x'(v)$. Also, since $\mathcal{S}(G) - \mathcal{S}(G') = \alpha(G)m$, we have by Observation 4.1 that

$$\text{pSMC}(G) \geq \alpha(G)m + \text{pSMC}(G') \ . \tag{12}$$

Now, by the induction hypothesis, inequality (9) holds for G'. Furthermore, since $\alpha(G) > \gamma(n)$,

$$\frac{nm}{\alpha(G)m} \leq 1.5 - 1/(2n) \ . \tag{13}$$

Thus, the required ratio follows from Equations (11), (12) and (13). This completes the proof.

□

4.2 The No-Preemption and Co-Scheduling models

Let $V_i[x]$ denote the set of vertices in V_i of lengths up to $\lfloor x \rfloor$.

We illustrate our approach for k-colorable graphs on bipartite graphs. The following algorithm produces a co-scheduling.

- Color all the vertices in $V_1[1]$ with the first color, then color all the vertices in $V_2[2]$ with the next 2 colors.
- Color the remaining vertices of $V_1[4]$ with the next 4 colors, then color the remaining vertices of $V_2[8]$ with the next 8 colors.
- In general, for $i = 0, 1, \ldots$, color the remaining vertices of $V_1[2^{2i}]$ with the next 2^{2i} colors, then color the remaining vertices of $V_2[2^{2i+1}]$ with the next 2^{2i+1} colors.

For a given vertex $v \in V_1$, the worst case occurs when $x(v) = 2^{2i} + 1$, for some i. Then, v is finished in step $x(v) + \sum_{j=0}^{i}(2^{2j} + 2^{2j+1}) = x(v) + 2^{2i+2} - 1 = 5\,x(v) - 5$. The same can be argued for any $u \in V_2$. Thus, the ***worst case*** completion time of ***any*** vertex is bounded by a factor of 5, giving a service-time ratio of 5.

We can improve this by examining two schedules: in the first we start with $V_1[1]$ and then alternate between V_2 and V_1, and in the second we start with $V_2[1]$ and then alternate between V_1 and V_2. It follows that each vertex v is completed within $3\,x(v)$ steps in one schedule and within $5x(v)$ in the other, or at most $4\,x(v)$ steps on the average. Hence, the sum of the better of the two schedules is at most $4\,\mathcal{S}(G)$. Note that this improves the absolute ratio to 4 but not the service-time ratio.

We now generalize this algorithm, both to general k, as well as to a more refined schedule based on randomly selected starting point. Let α be a constant to be optimized, and let $d = \alpha^k$. Our algorithm is as follows.

Steps(G, α)

Let X be a random number uniformly chosen from $[0, 1)$.

Let $d = \alpha^k$.

Let $Y = d^X$.

for $i = 0$ to $\log_d p$ do

for $j = 1$ to k do

Let $A_{ij} = d^{i-1} \alpha^j\, Y$.

Color the remaining vertices of $V_j[A_{ij}]$,

using the next $\lfloor A_{ij} \rfloor$ colors.

Clearly, the output is a co-schedule. The bipartite graphs case corresponds to Steps(G,2).

Remark: The algorithm can be derandomized, by examining a set of evenly spaced candidates for the random number X. The additive error term will be inversely proportional to the number of schedules evaluated.

Lemma 4.4 *Let X be a uniform random variable on $[0, 1)$. Let $d > 1$ and let $Y = d^X$ be a random variable. Then,*

$$E[Y] = \frac{d-1}{\ln d} \ .$$

Proof. The distribution function of Y is given by

$$F_Y(y) = P[Y \leq y] = P[d^X \leq y] = P[X \leq \log_d y] = \log_d y \ .$$

Thus, the expected value of Y is

$$E[Y] = \int_1^d y F_Y'(y)\, dy = \int_1^d y \frac{1}{y \ln d}\, dy = \frac{d-1}{\ln d} \ .$$

□

Theorem 4.5 *The expected multicolor sum of* Steps *is at most* $(1.544k+1)\, \mathcal{S}(G)$ *on a k-colorable graph G, and at most* $2.796\, \mathcal{S}(G)$ *on a bipartite graph G.*

Proof. Let v be a vertex. We bound the expected delay until v gets colored. Let τ be the color class of v, i.e. such that $v \in V_\tau$. Let $A_v = \min\{A_{i,\tau} : A_{i,\tau} \geq x(v)\}$, representing the length of the interval in which v was colored.

Let $X_v = \log_d A_v - \log_d x(v)$. Thus, $A_v = d^{X_v} x(v)$. Note that X_v is a uniform random variable on $[0, 1)$. This is easy to see when $x(v) = d^{i-1} \alpha^\tau$ for some integer i, since then X_v is identical to the X in the pseudocode. On the other hand, the fact that $\log_d A_{1,\tau}, \log_d A_{2,\tau}, \ldots$ is an arithmetic sequence with unit separation implies that the value of $x(v)$ does not affect the distribution of X_v. Thus by Lemma 4.4,

$$E[A_v] = E[d^{X_v}]\, x(v) = \frac{d-1}{\ln d}\, x(v) \ .$$

Note that the lengths A_{ij} are increased by a factor of α in each iteration of the inner loop of Steps. Therefore, the *delay time* d_v of v, or the time until v starts to get colored, is at most

$$d_v \leq \sum_{i=1}^{\infty} \frac{A_v}{\alpha^i} = A_v \frac{1}{\alpha - 1} \ . \tag{14}$$

Thus,

$$E[d_v] \leq \frac{d-1}{\ln d} \cdot \frac{1}{d^{1/k} - 1}\, x(v) \ . \tag{15}$$

For bipartite graphs, this implies that

$$E[d_v] \leq \frac{\alpha^2 - 1}{2\ln \alpha} \cdot \frac{1}{\alpha - 1}\, x(v) = \frac{\alpha + 1}{2 \ln \alpha}\, x(v) \ .$$

The function $f(x) = \frac{x+1}{\ln(x)}$ is minimized when $x = 3.5911$, which results in the bound $E[d_v] \leq 1.796\, x(v)$. The expected cost of the coloring is thus

$$E[\mathrm{SMC}_\Psi(G)] = \sum_v E[d_v] + \mathcal{S}(G) \leq 2.796\, \mathcal{S}(G) \ .$$

where Ψ is the (probabilistic) coloring output of Steps$(G, 3.5911)$.

For k-colorable graphs, we use the inequality $1 + x \leq e^x$ to bound $d^{1/k} - 1 \geq (\ln d)/k$, obtaining from (15) that

$$E[d_v] \leq \frac{d-1}{\ln d} \cdot \frac{k}{\ln d}\, x(v) \ .$$

The function $g(x) = \frac{x-1}{\ln^2 x}$ takes a minimum at $x = 4.9215$, resulting in $E[d_v] \leq 1.544k\, x(v)$. Hence,

$$E[\mathrm{SMC}_\Psi(G)] \leq (1.544k + 1)\, \mathcal{S}(G) \ .$$

where Ψ is the coloring output of Steps$(G, 4.9215^{1/k})$. □

We can also bound the service-time ratio of k-colorable graphs. For each vertex v, it holds that $A_v \leq d\, x(v)$, since $A_{i+1,j}/A_{ij} = d$. Using that $e^x \geq 1 + x + x^2/2$, it follows from (14) that

$$d_v \leq \frac{d}{d^{1/k} - 1}\, x(v) \leq \frac{dk}{(\ln d)(1 + \ln d/2k)}\, x(v) \ .$$

Let $d = e$, the base of the natural logarithm, we obtain a worst-case bound on the finish time of v of

$$f_\Psi(v) \leq \left(\frac{ek}{1 + 1/2k} + 1\right) x(v) \leq \frac{ek + e/2}{1 + 1/2k}\, x(v) = ek\, x(v) \ .$$

We summarize the results of this section in the following theorem.

Theorem 4.6 Steps *approximates* coSMC *(and thus also* npSMC*) for bipartite graphs with an absolute ratio of* 2.796 *and a service-time ratio of* 5*. For k-colorable graphs, it gives an absolute ratio of* $1.544k + 1$ *and a service-time ratio of* ek*.*

5 Bounded degree graphs and line graphs

In this section we consider two families of graphs for which a variation of the greedy coloring achieves "good" approximation factors. In Subsection 5.1 we consider bounded degree graphs with maximum degree Δ. In Subsection 5.2 we consider line graphs.

The first-fit greedy algorithm gets the vertices of the graph in some order and colors each vertex with the first available colors. It is a reasonable candidate algorithm for the PREEMPTION and the NO-PREEMPTION models but not for the CO-SCHEDULING problem. For bounded-degree graphs, first-fit greedy achieves absolute ratios of $\Delta + 1$ for pSMC and $2\Delta + 1$ for npSMC. For line graphs, it can be shown to achieve an approximation factor of $\Omega(\sqrt{p})$.

In this section we consider a modified first-fit greedy algorithm, Sorted Greedy (SG), which orders the vertices in a nondecreasing order of lengths, before coloring the vertices in a first-fit manner. We show that SG improves the approximation ratios for bounded degree graphs to $(\Delta + 2)/3$ and $\Delta + 1$ for pSMC and npSMC respectively. For line graphs the improvement is more impressive – SG obtains a ratio of 2. In addition, we also bound the service-time ratio of SG, whereas the service-time ratio of first-fit is unbounded.

In the rest of this section, let Ψ denote the multicoloring produced by SG. Let $Q(G)$ denote the quantity $\sum_{uv\in E} \min(x(u), x(v))$. Observe that

$$Q(G) \le \sum_{uv\in E} \frac{x(u) + x(v)}{2} = \frac{1}{2}\sum_{v\in V} d(v)x(v) \le \frac{\Delta}{2}\,\mathcal{S}(G)\ . \tag{16}$$

5.1 Bounded degree graphs

In the first two theorems we state the approximation ratio of SG for pSMC and npSMC. We then present a third theorem that states the service-time ratio in both models.

Theorem 5.1 SG *approximates* pSMC *within a ratio of* $(\Delta + 2)/3$*, and this is tight.*

Proof. First observe that each edge can delay only the incident vertex that is colored later. For SG, this delay amounts to the minimum of the lengths of the endpoints. Thus, the multicolor sum of SG is bounded by

$$\mathrm{SMC}_{\Psi}(G) \le \mathcal{S}(G) + Q(G)\ . \tag{17}$$

For $uv \in E$, denote by $D_u(v)$ the number of colors given to u, that are smaller than $f_\Psi(v)$. Clearly,

$$\mathrm{pSMC}(G) \geq \mathcal{S}(G) + \sum_{v \in V} \max_{u \in N(v)} \{D_u(v)\} \geq \mathcal{S}(G) + \sum_{v \in V} \frac{\sum_{u \in N(v)} D_u(v)}{\Delta} .$$

In the above, each edge uv contributes $(D_u(v) + D_v(u))/\Delta$ to the sum. Since $D_u(v) + D_v(u) \geq \min\{x(u), x(v)\}$, it follows that

$$\mathrm{pSMC}(G) \geq \mathcal{S}(G) + \frac{Q(G)}{\Delta} . \tag{18}$$

Let $d = 2Q(G)/\mathcal{S}(G)$. Then, from (17) and (18),

$$\frac{\mathrm{SMC}_\Psi(G)}{\mathrm{pSMC}(G)} \leq g(d) \equiv \frac{1 + d/2}{1 + d/(2\Delta)} .$$

Since $d \leq \Delta$ by (16), and $g(d)$ is monotonically increasing, we have that

$$g(d) \leq g(\Delta) = \frac{1 + \Delta/2}{3/2} = \frac{2 + \Delta}{3} .$$

A matching lower bound was shown in [BBH+98] for the sum coloring problem, thus it also applies to pSMC. □

Theorem 5.2 SG *approximates* npSMC *within a ratio of* $\Delta + 1$.

Proof. In the No-Preemption model, a vertex v can be delayed not only by the lengths of its neighbors but also by gaps in the set of available colors that are too small for coloring v contiguously. There can be as many gaps as neighbors, and each gap can be of length $x(v) - 1$. Hence,

$$\mathrm{SMC}_\Psi(G) \leq \mathcal{S}(G) + \sum_{uv \in E} (x(u) + x(v)) \leq (\Delta + 1)\, \mathcal{S}(G) + Q(G) . \tag{19}$$

By inequalities (19) and (18), the approximation ratio is then at most $\Delta + 1$. □

With similar arguments we can prove weaker bounds for absolute and service-time ratios of SG.

Theorem 5.3 SG *has absolute ratios of* $\Delta/2 + 1$ *for* pSMC *and* $3\Delta/2 + 1$ *for* npSMC. *It has service-time ratios of* $\Delta + 1$ *for* pSMC *and* $2\Delta + 1$ *for* npSMC.

Proof. The absolute ratios follow directly from Inequalities (16), (17), and (19).

Consider now the service-time. In the preemptive model, each vertex v is delayed only by the lengths of those of its neighbors that appear earlier in the sequence. There are at most Δ such neighbors, each with no greater length, hence v is delayed at most $\Delta\, x(v)$. In the nonpreemptive model, each neighbor can additionally cause a gap of up to $x(v) - 1$ colors that v cannot use, adding at most another Δ factor. □

5.2 Line graphs and intersection graphs of hypergraphs

The *line graph* L_G of a graph $G = (V, E)$ is the intersection graph of E — the vertices in L_G are the edges of G and two vertices in L_G are adjacent whenever the corresponding edges in G are. Note that a multicoloring of L_G corresponds to an edge multicoloring of G.

A property of a line graph L_G is that its edges can be partitioned into cliques, such that each vertex belongs to at most two cliques [H69]. The following theorem capitalizes on this property to achieve a constant approximation factor for this family of graphs.

Theorem 5.4 SG *approximates* pSMC *of line graphs within a ratio of* $2 - 4/(\Delta + 4)$*, and this is tight.*

Proof. Given a line graph L_G of a graph G, form another line graph H as follows. For each vertex v in G, H contains a vertex for each of the $d(v)$ edges incident on v in G. That is, H is a disjoint collection of the maximal cliques in L_G. Each vertex in L_G corresponds to exactly two vertices in H. Further, all the edges in L_G have corresponding edges in H.

The minimum multicolor sum of a clique C is found by ordering the vertices by nondecreasing length, giving a contiguous coloring. Each vertex is delayed by the sum of the lengths of earlier vertices, or in other words, each edge uv causes a delay of $\min(x(u), x(v))$. Thus,

$$\text{pSMC}(C) = \sum_{v \in C} x(v) + \sum_{uv \in E(C)} \min(x(u), x(v)) = \mathcal{S}(C) + Q(C) \ .$$

Since the cliques of H are disjoint, the multicolor sum of H is the sum of the multicolor sums of the cliques, or

$$\text{pSMC}(H) = \mathcal{S}(H) + Q(H) \ .$$

Observe that since each vertex in L_G appears twice in H, $\mathcal{S}(H) = 2\,\mathcal{S}(L_G)$, and any multicoloring of L_G corresponds to a multicoloring of H of double the weight, $\text{pSMC}(H) \leq 2 \cdot \text{pSMC}(L_G)$. Further, there is a one to one correspondence between the edges of L_G and H, thus $Q(H) = Q(L_G)$. Hence, we have

$$\text{pSMC}(L_G) \geq \mathcal{S}(L_G) + \frac{1}{2} Q(L_G) \ . \tag{20}$$

As before, let $d = 2Q(L_G)/\mathcal{S}(L_G)$. Then, from (17) and (20), we bound the approximation ratio by

$$\frac{\text{SMC}_{\mathsf{SG}}(L_G)}{\text{pSMC}(L_G)} \leq g(d) \equiv \frac{1 + d/2}{1 + d/4} \ .$$

Since $g(d)$ is monotone increasing and $d \leq \Delta$, we have

$$g(d) \leq g(\Delta) = 2 - 4/(\Delta + 4) \ .$$

This matches the bound proved for the sum coloring of regular edge graphs [BBH+98]. □

Intersection graphs of set systems: We can generalize the argument for line graphs to intersection graphs of set systems where each set is of size at most k, e.g. k-uniform hypergraphs. In the context of resource allocation, these graphs correspond to the case where each job requires the use of up to k resources to execute.

Theorem 5.5 *Let L_H be an intersection graph of a hypergraph H, where each edge is of cardinality at most k. Then,* SG *approximates* pSMC(L_H) *by a factor of k.*

Proof. Recall, that each edge e_i in H is represented by a vertex e_i in L_H, and all the edges containing some vertex v in H form a clique in L_H. As the maximal cardinality of any edge in H is k, each vertex in L_H belongs to at most k maximal cliques. An argument similar to the one we used for line graphs, shows that

$$\text{pSMC}(L_H) \geq \mathcal{S}(L_H) + \frac{1}{k} Q(L_H) \ . \tag{21}$$

Thus, using (17), we have

$$\text{SMC}_\Psi(L_H) \leq k \left(1 - \frac{2(k-1)}{\Delta + 2k}\right) \text{pSMC}(L_H) \ .$$

□

6 Discussion

Extensions to vertex-weighted graphs: In the weighted SMC problem, each vertex v is associated with a weight $w(v)$ and the goal is to minimize $\sum_{v \in V} w(v)\, f_\Psi(v)$. All of the results in this paper apply also to this optimization goal without any overhead. We have chosen to give the results for the unweighted case in order to simplify the presentation.

We indicate briefly the modifications needed for some of the results. In Lemma 3.1, if G is the weighted input graphs, we define the weights of the derived graph H by $w_H(v_i) = w_G(v)/x(v)$. Other arguments of that section follow directly, if we replace n by $W = \sum_v w(v)$, and redefine $\mathcal{S}(G)$ as $\sum_v x(v)\, w(v)$. In Section 4.1, we redefine $\gamma(n)$ as $2W^2/(3W-1)$ and $\alpha(G)$ as the maximum weight of an independent set in G. Results of Section 4.2 need no modifications. Finally, in Section 5, redefine SG to order the vertices in a nondecreasing order of the $w(v)/x(v)$ values. Let the value of $Q(G)$ be $\sum_{uv \in E} \min(w(v)x(u), w(u)x(v))$ and replace $D_u(v)$ by $w(v)D_u(v)$.

Waiting time: Another measure of a multicoloring of interest is the *total waiting time*, or the sum of the finishing times less the sum of the lengths. Also, the *average waiting time* is the total waiting time divided by the number of jobs. In the unit-weight case, this corresponds to numbering the colors starting from 0 instead of 1. In general, the total waiting time of a multicoloring Ψ is equal to its multicolor sum, $\mathrm{SMC}_\Psi(G)$, less the sum of the lengths, $\mathcal{S}(G)$.

This measure is harder to approximate than SMC. We can observe from the proof of Theorem 5.4 that SG approximates the average waiting time of a preemptive multicoloring of line graphs within a ratio of 2. Also, by subtracting $\mathcal{S}(G)$ from the right hand sides of (17) and (21), it approximates intersection graphs of hypergraphs within a ratio of k. Obtaining nontrivial approximations of other graph classes is an open problem.

New developments: In [HKP+99], efficient polynomial time algorithms were given for the npSMC problem on trees and paths, while an efficient polynomial time approximation scheme (PTAS) was given for pSMC on trees. Shortly after, PTASs were given in [HK99] for both npSMC and pSMC on both partial k-trees and planar graphs.

Open problems: One important issue is to reduce the logarithmic approximation factor in the nonpreemptive case. This appears to be hard in general, but we may at least expect progress on special classes of graphs.

Interval graphs are an important class of graphs, for which our problems have numerous applications. Since WIS is polynomially solvable on this class, we have from our general arguments ratios of 16 for pSMC and $O(\log n)$ for npSMC, as well as 4 for SC from the results of [BBH+98]. All of these ratios can certainly be improved by a MAXIS-like algorithm.

In the preemptive case, we know few cases of polynomially solvable classes of graphs. Resolving the case of trees, or even the case of paths, would be interesting.

Acknowledgments

We would like to thank two anonymous referees, who provided many helpful comments on this paper.

References

[B92] M. Bell. Future directions in traffic signal control. *Transportation Research Part A*, 26:303–313, 1992.

[BBH+98] A. Bar-Noy, M. Bellare, M. M. Halldórsson, H. Shachnai, and T. Tamir. On chromatic sums and distributed resource allocation. *Information and Computation,* 140:183–202, 1998.

[BHK99] A. Bar-Noy, M. M. Halldórsson, and G. Kortsarz. A Matched Approximation Bound for the Sum of a Greedy Coloring. *Information Processing Letters* **71**, 135–140, 1999.

[BHK+99] A. Bar-Noy, M. M. Halldórsson, G. Kortsarz, R. Salman, and H. Shachnai. Sum Multi-Coloring of Graphs. In *Proc. Seventh European Symposium on Algorithms* (ESA '99), Lecture Notes in Computer Science Vol. 1643, Springer-Verlag, July 1999.

[BH92] R. B. Boppana and M. M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. *BIT*, 32:180–196, 1992.

[BH94] D. Bullock and C. Hendrickson. Roadway traffic control software. *IEEE Transactions on Control Systems Technology*, 2:255–264, 1994.

[BK98] A. Bar-Noy and G. Kortsarz. The minimum color-sum of bipartite graphs. *Journal of Algorithms*, 28:339–365, 1998.

[CCO93] J. Chen, I. Cidon, and Y. Ofek. A local fairness algorithm for gigabit LANs/MANs with spatial reuse. *IEEE Journal on Selected Areas in Communications*, 11:1183–1192, 1993.

[D68] E. W. Dijkstra. Cooperating Sequential Processes, In *Programming Languages*, ed. by F. Genuys, Academic Press, 43–112, 1968.

[FK96] U. Feige and J. Kilian. Zero Knowledge and the Chromatic number. *Journal of Computer and System Science*, 57:187–199, 1998.

[GLS88] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, 1988.

[H69] F. Harary, Graph Theory. Addison-Wesley, 1969.

[H99] M. M. Halldórsson. Approximating weighted independent set and hereditary subset problems. In *Proceedings of the Fifth International Computing and Combinatorics Conference* (COCOON '99), Tokyo, Japan, Lecture Notes in Computer Science Vol. 1627, pages 261–270, Springer-Verlag, July 1999.

[HK99] M. M. Halldórsson and G. Kortsarz. Multicoloring Planar Graphs and Partial k-trees. In *Proceedings of the Second International Workshop on Approximation algorithms* (APPROX '99). Lecture Notes in Computer Science Vol. 1671, Springer-Verlag, August 1999.

[HKP+99] M. M. Halldórsson, G. Kortsarz, A. Proskurowski, R. Salman, H. Shachnai, and J. A. Telle. Multi-Coloring Trees. In *Proceedings of the Fifth International Computing and Combinatorics Conference*, Tokyo, Japan, Lecture Notes in Computer Science Vol. 1627, Springer-Verlag, pages 271–280, July 1999.

[J97] K. Jansen. The Optimum Cost Chromatic Partition Problem. In *Proceedings of the Third Italian Conference on Algorithms and Complexity (CIAC '97).* Lecture Notes in Computer Science Vol. 1203, pp. 25–36, Springer-Verlag, July 1997.

[K89] E. Kubicka. The Chromatic Sum of a Graph. PhD thesis, Western Michigan University, 1989.

[K91] H. A. Kierstead. A polynomial time approximation algorithm for Dynamic Storage Allocation. *Discrete Mathematics*, 88:231–237, 1991.

[KKK89] E. Kubicka, G. Kubicki, and D. Kountanis. Approximation Algorithms for the Chromatic Sum. In *Proceedings of the First Great Lakes Computer Science Conference*, Lecture Notes in Computer Science Vol. 1203, pp. 15–21, Springer-Verlag, July 1989.

[KS89] E. Kubicka and A. J Schwenk. An Introduction to Chromatic Sums. In *Proceedings of the Seventeenth Annual ACM Computer Science Conference, "Computing trends in the 1990's"*, pages 39–45, 1989.

[L81] N. Lynch. Upper Bounds for Static Resource Allocation in a Distributed System. *Journal of Computer and System Sciences*, 23:254–278, 1981.

[NS95] M. Naor and L. Stockmeyer. What Can be Computed Locally? *SIAM Journal on Computing*, 24:1259–1277, 1995.

[NSS99] S. Nicoloso, M. Sarrafzadeh, and X. Song. On the Sum Coloring Problem on Interval Graphs. *Algorithmica*, 23:109–126, 1999.

[S99] T. Szkaliczki. Routing with Minimum Wire Length in the Dogleg-Free Manhattan Model is NP-complete. *SIAM Journal on Computing*, 29(1):274-287, 1999.

[SG98] A. Silberschatz and P. Galvin. Operating System Concepts. Addison-Wesley, 5th Edition, 1998.

[SP88] E. Steyer and G. Peterson. Improved Algorithms for Distributed Resource Allocation. In *Proceedings of the Seventh Annual Symposium on Principles of Distributed Computing (PODC)*, pp. 105–116, 1988.

[TEA+89] C. Thomassen, P. Erdös, Y. Alavi, J. Malde, and A. J. Schwenk. Tight Bounds on the Chromatic Sum of a Connected Graph. *Journal of Graph Theory,* 13:353–357, 1989.

[T95] A. S. Tannenbaum. *Distributed Operating Systems.* Prentice-Hall, 1995.