アルゴリズム 53－6
（1996. 9. 13）

彩色和とその応用について

マグナス ハルダーソン
アイスランド大学

グラフの点彩色とは、すべての辺の両端点の値が異なるように各頂点に正整数を割り当てることである。本稿では、各頂点に割り当てられた正整数の和が最小となる点彩色を求める彩色和問題について考察する。

本稿では彩色和がスケジューリング問題など種々の問題に応用できることを示すとともに、近似率の上限および下限について既知のものより改善された結果を示す。特に、ライングラフに対する近似率2の並列アルゴリズム、次数制約のあるグラフに対する近似率$(\Delta+2)/3$のアルゴリズム、二部グラフに対する近似率9/8のアルゴリズムを示す。一方、一般のグラフに対しては、すべての$\epsilon > 0$について$n^{1-\epsilon}$以内の近似が困難であることを示す。

On Chromatic Sums and its Applications

Magnús M. Halldórsson

Science Institute, University of Iceland, IS-107 Reykjavik, Iceland.
e-mail: mmh@rhi.hi.is

Graph coloring is an assignment of positive integers to vertices of a graph so that adjacent vertices receive different values. The *minimum chromatic sum* problem is to find a coloring where the *sum* of the values assigned to the vertices is minimized.

We describe a number of applications of this measure, especially with respect to minimizing average processing times, and survey known results. We present a number of improved results on approximability and non-approximability. In particular, line graphs can be approximated within a factor of 2 in parallel, bounded-degree graphs within $(\Delta+2)/3$, and bipartite graphs within 9/8. On the other hand, we show that for general graphs are hard to approximate within $n^{1-\epsilon}$, for any $\epsilon > 0$.

1 Introduction

Graph coloring is a problem that formalizes a large class of practical problems, particularly those dealing with the scheduling of pairwise incompatible items. We are given a graph, in the form of a base set and a collection of pairs from this base set (representing conflicts), and we are to assign positive integers to the elements so that no conflicting pair gets assigned the same value.

The standard measure of a coloring of a graph is the number of colors used, or the largest positive integer assigned. This corresponds to the length of the schedule; if elements assigned value i are processed at step i, then the number of colors equals the maximum number of steps needed to process any element. The focus of the current paper is to study another measure of a coloring: the *sum* of the values assigned to the vertices, or alternatively, the *average* number of steps until an element is processed.

This measure is known as the *chromatic sum*, and has, as its clean definition suggests, applications in many fields of computing. It has apparently been rediscovered many times, with disparate results. Here are some examples:

Distributed resource allocation [1]: A set of distributed processors compete over resources, which is modeled by a conflict graph, where nodes represent processors and edges represent competition on a resource. A coloring of this graph yields a schedule of allocation of a static set of resources. The sum of the coloring corresponds exactly to the *average response time* of the jobs.

VLSI routing [11]: In a design problem, known as Over-The-Cell Routing, we are given a set of two-terminal nets and a set of parallel, horizontal tracks of distance $d = 1, 2, 3, \ldots$ from the baseline where the terminals lie. The nets are routed with a vertical connection from each terminal to the assigned track along with a horizontal connection within the track. No overlapping nets can be routed within the same track. The objective is to minimize the total wiring length, which, in addition to the fixed and pre-determined horizontal costs, equals twice the sum of the distances from the nets to the assigned tracks.

Register allocation Modern RISC design endow processors with ample registers, not all of which may be immediately accessible. It is logical to assume that the access times of the registers may vary. Furthermore, if we can determine the frequency with which each variable is accessed, the vertices of the variable-conflict graph may be given weights accordingly. We want to assign variables to registers so that variables with overlapping lifespans do not share a register, with the objective of minimizing the total cost of access. This corresponds to finding a "low-cost" coloring of a vertex weighted graph, where the colors have different weights.

Bus rentals We are to schedule a collection of trips at predetermined time periods. The buses we can rent have different costs, depending on the rental company and/or cost-effectiveness of the model. A schedule the trips onto the available buses therefore corresponds to a coloring minimizing the sum of the costs of the trips. Since the trips conflict only in time, the conflict graph here can be seen to be an interval graph.

The case of interval graphs is known as the Fixed Interval Scheduling Problem with machine-dependent processing costs [8]. We shall focus on the pure combinatorial version, the *chromatic sum* problem, where the costs correspond to the sequence of positive integers, and the elements/vertices are unweighted.

Problem definition Given a graph $G = (V, E)$, a *vertex coloring* is a function $\Psi : V \rightarrow \mathbf{N}$ such that adjacent vertices are assigned distinct numbers (*colors*). The Minimum Color problem is to find a vertex coloring which uses the minimum number of colors. In this paper we consider a related problem known as *Minimum Chromatic Sum (MCS)* problem [9, 10].

> Given a graph $G = (V, E)$, find a vertex coloring $\Psi : V \rightarrow \mathbf{N}$ for G such that $\sum_{v \in V} \Psi(v)$ is minimized.

Previous results The chromatic sum problem has been introduced directly or indirectly by various papers in the past. It has been shown to be NP-hard for general graphs [10] and line graphs [1], and it is also easy to show for circular-arc graphs. The case is open for interval graphs, but the cost-dependent version has been shown to be hard.

Polynomial time algorithms are known for trees [10], and these can be extended to k-outerplanar graphs. It is also easily solvable on co-bipartite graphs, and more generally, co-triangle-free graphs by matching, since each color class can contain at most two vertices.

Several results are also known about approximability. We say that a class of graphs can be approximated within a factor of ρ if there is an algorithm which on any instance in the class will output a coloring whose sum is at most ρ times the (optimal) chromatic sum.

Halldórsson and Radhakrishnan [7] gave a general theorem that showed that the approximability of MCS was always at most that of the INDEPENDENT SET problem, within a constant factor. That is, the natural algorithm known as MaxIS that finds (or approximates) a maximum independent set, colors it with the first color, and then iterates, attains this bound. This shows that MCS is approximable within: $O(n/\log^2 n)$ on general graphs [5], $O(\Delta \log\log\Delta/\log\Delta)$ on graphs of maximum degree Δ [12], $O(n^{.2134})$ on 3-colorable graphs [3], and $O(1)$ on all perfect graphs and partial k-trees, among others.

Shachnai et al. [1] improved the constant of the MaxIS algorithm from 12 to 4. They also introduced a still simpler algorithm, known as *compact coloring*, which they showed to achieve a ratio of 6 on line graphs. They also gave an algorithm that approximates bipartite graph within 7/6. Kubicka et al. [9] showed that compact coloring approximates sparse graphs within a factor of $(\bar{d} + 2)/2$, where $\bar{d}$ is the average degree. Nicoloso et al. [11] gave a 2-approximate algorithm for interval graphs.

On the hardness side, it was shown in [7] that there was a constant $\epsilon > 0$, such that it was NP-hard to approximate the chromatic sum of an arbitrary graph within a factor of n^{ϵ}. Previously, [9] had shown weaker hardness of an additive term.

Results presented here We present the following results:

1. The performance ratio of compact coloring is:
   (a) At most 2, on line graphs,
   (b) Precisely $(\Delta + 2)/3$, on bounded-degree graphs, and
   (c) No better than $(\bar{d} + 2)/2$, on sparse graphs.
2. The performance ratio of the MaxIS algorithm is:
   (a) At least 3.39 times the ratio of the independent set algorithm used, improving the bound of [1] of 2.

(b) At least 1.74 on interval graphs, and conjecture it to be exactly 2.

3. Bipartite graphs can be approximated within 9/8.

4. MCS is hard to approximate on general graphs within $n^{1-\epsilon}$, for any $\epsilon > 0$.

Some of these results will appear in [2].

This is a partial step in a project to classify the solvability and approximability of the chromatic sum problem on important classes of graphs, and analyze the performance of these simple, natural, and efficient algorithms.

2 Compact Coloring

A coloring $\Psi : V \to \{1 \dots k\}$ is *compact* if $C_i = \{v \in V \mid \Psi(v) = i\}$ comprises a maximal independent set in $G \setminus \bigcup_{j<i} C_j$, for every $1 \le i \le k$. Alternatively, a coloring Ψ is compact if and only if every vertex v with $\Psi(v) = i$ has a neighbor u with $\Psi(u) = j$ for all $1 \le j \le i-1$.

This suggests a greedy algorithm, often referred to as *first-fit*: Process the vertices in an arbitrary order and assign a vertex to the smallest color with which none of its preceding neighbors have been colored. This method has the advantage of being *on-line*, processing resource requests as they arrive.

The following general upper bound on the chromatic sum has been observed several times in the past. Let m denote the number of edges in the graph.

Lemma 2.1 **([4, 9])** *The sum of any compact coloring is at most $m + n$.*

This bound is tight for disjoint collection of cliques. It can be attained by a parallel algorithm [6].

Bounded-degree graphs

Theorem 2.2 *Any compact coloring of a graph $G = (V, E)$ provides a $\frac{\Delta+2}{3}$-approximation to* MCS(G)*, and that is tight.*

Proof: All edges have at least one endpoint outside the first color class of the optimal solution. Thus, when maximum degree is bounded by Δ, there are at least $\lceil m/\Delta \rceil$ vertices outside the first color class. That is, we have: $MCS(G) \ge n + m/\Delta$.

Thus, by Lemma 2.1, the performance ratio of a compact coloring is at most

$$\frac{m+n}{n+m/\Delta} = \frac{\overline{d}/2+1}{1+\overline{d}/(2\Delta)}.$$

This is maximized at $\overline{d} = \Delta$, for a ratio of $(\Delta + 2)/3$.

This ratio is tight for the graph $B_{p,p}$ formed by a complete bipartite graph from which a single bipartite matching has been removed. Namely, the graph contains vertex set $\{v_1, \dots v_p, u_1, \dots u_p\}$ and the edge set $\{(v_i, u_j) \mid 1 \le i < j \le p, i \ne j\}$. One compact coloring contains p classes with 2 vertices each, for a cost of $2\binom{p}{2} = p(p+1)$ versus an optimal coloring of cost $3p$, for a ratio of $(p+1)/3 = (\Delta+2)/3$. □

Sparse graphs It follows from Lemma 2.1 that the performance ratio of compact coloring on sparse graphs is at most $(\overline{d}+2)/2$, where $\overline{d}$ is the average degree of the graph. We show that this is in fact tight, within a lower order term.

Theorem 2.3 *The performance ratio of compact coloring is no better $(\overline{d}+2)/2 - O(1/\overline{d})$.*

Proof: Consider the following bipartite graph, with vertices $u_{i,j}, i = 1,\ldots,p, j = 1,\ldots,d+1$ and $v_k, k = 1,\ldots d$, and edges $(u_{i,j}, v_k)$ iff $k < j$ or ($i = 1$ and $k \neq j$). The number of edges equals $p\binom{d+1}{2} + \binom{d}{2} = (n-1)d/2$. Hence, the average degree equals $d(1-1/n)$.

One compact coloring has $d+1$ classes, with the i-th class containing $u_{i,j}, j = 1,\ldots,p$, as well as v_i when $i \leq d$. All vertices are adjacent to exactly one vertex in each of the previous classes, hence the cost of the coloring equals $m+n$.

The optimal solution is of cost $n + d = n + \overline{d}(1+1/n)$ and the performance ratio is

$$\frac{m+n}{n+d} = \frac{\overline{d}+2}{2+2\overline{d}/n+2\overline{d}/n^2}$$

which is asymptotically $(\overline{d}+2)/2$. □

Line Graphs We show below that for the subclass of *line graphs*, compact coloring is a 2-approximation to the chromatic sum.

Given a graph $G = (V, E)$, the *line graph* of G, denoted by $L(G)$ is the intersection graph of E: The vertices in $L(G)$ are the edges of G. Two vertices in $L(G)$ are adjacent whenever the corresponding edges in G are. We say that G is a line graph, if there exists some graph G', such that $G = L(G')$.

We use the property of line graphs that its edge set can be partitioned into cliques, such that each vertex belongs to at most two cliques.

Theorem 2.4 *Any compact coloring of a line graph G is a 2-approximation to* MCS(G).

We prove a stronger ratio of $2 - 2/(\overline{d}+4)$, which follows from the combination of Lemma 2.1 and the following lemma.

Lemma 2.5 *For a line graph G, $MCS(G) \geq (m+2n)/2$.*

Proof: Let $Q_1, Q_2, \ldots Q_l$ be the clique partition of G, with q_i denoting the size of each clique. Extend the partitions so that each vertex appears exactly twice, by adding singleton cliques for those vertices that appeared only once. Let Q denote the set of all $2n$ pairs (i, v) where v is contained in clique Q_i.

We define a *clique labeling* to be an assignment of positive integers to the pairs of Q such that, for each Q_i and each distinct u, w in Q_i, (i, v) and (i, w) have different labels. The *cost* of a clique labeling is the sum of the labels. Let $CL(G)$ denote the optimal clique labeling of line graph G. The minimum cost clique labeling has the labels involving a given clique Q_i arranged to be exactly the first q_i positive integers. Hence,

$$CL(G) = \sum_i \binom{q_i+1}{2} = \sum_i \binom{q_i}{2} + q_i = \sum_i |E(Q_i)| + |V(Q_i)| = m + 2n. \qquad (1)$$

Intuitively, we have a labeling of the vertices, where each vertex may receive two labels, one for each of its cliques. An ordinary vertex coloring can easily be extended to a clique labeling by doubling each label. Thus, the optimal chromatic sum is at least half the cost of an optimal clique labeling, i.e. $CL(G) \leq 2 \cdot MCS(G)$. The lemma now follows from (1). □

This has an application to the corresponding *edge coloring* problem.

Corollary 2.6 *Any compact edge coloring of a graph G is a 2-approximation to the minimum edge coloring sum of G.*

3 The MaxIS algorithm

The following theorem [1, 7] illustrates the versatility of the MaxIS algorithm for the MCS problem.

Theorem 3.1 *When using a ρ-approximate independent set algorithm, the* MaxIS *algorithm is a 4ρ-approximation to the* MCS.

We have shown the following corresponding lower bound, improving on the lower bound of 2 of [1]. It is omitted for lack of space.

Theorem 3.2 *The performance ratio of* MaxIS *on general graphs is at least 3.3912.*

Interval graphs The special case of interval graphs is an important one, given the multitude of applications of chromatic sums for that class of graphs. The independent set problem is polynomial solvable for this class, and thus the MaxIS algorithm is a natural candidate. We are only able to give a partial answer to the tantalizing question of the performance guarantee of that algorithm.

Theorem 3.3 *The performance ratio of* MaxIS *on interval graphs is at least* $30/17 \approx 1.764$.

Proof: Consider the following bag of intervals: $(1,4),(5,6),(7,11),(0,2),(3,8),(9,10),(3,8),$ $(9,10)$. An optimal coloring has the first three intervals in one color, the next three in the second, and the remaining two in the last color, for a total cost of 15. MaxIS will produce the coloring $[(0,2),(5,6),(9,10)],[(1,4),(9,10)],[(3,8)],[(3,8)],[(7,12)]$ for a total cost of 21. Hence, a ratio of 7/5.

Now makes t copies of all these intervals. The optimal and heuristic solutions will simply have all color classes repeated t times. E.g. the latter will have $5t$ classes with vertices, $2t$ classes with two vertices, and t classes with three vertices. The cost of the optimal solution is now $\binom{3t+1}{2}+2\binom{2t+1}{2} = 17/2t^2+7/2t$, while the cost of the heuristic solution is $\binom{5t+1}{2}+\binom{2t+1}{2}+\binom{t+1}{2} =$ $15t^2 + 4t$. The ratio is then $30/17 - O(1/t) > 1.764$. □

4 Hybrid approaches

Approximating the Chromatic Sum for Bipartite Graphs In the following we describe a modified algorithm that achieves a ratio of $\frac{9}{8}$ for the MCS of bipartite graphs.

The algorithm colors the graph in two ways, and then chooses the coloring with a smaller sum. One coloring is any two-coloring. The other coloring colors a maximum independent set with the first color, and then two-colors the remaining vertices. Note that a maximum independent set of a bipartite graph can be found in polynomial time by computing a maximum matching.

Theorem 4.1 *The above algorithm achieves a ratio of $\frac{9}{8}$ to the MCS for any bipartite graph.*

Proof: Let α be the size of the maximum independent set of the graph. The cost of our former coloring is at most $3n/2$ and the the latter coloring is at most $\alpha + (n - \alpha) \cdot 5/2 = 5n/2 - 3\alpha/2$. The cost of the optimal coloring is at least $\alpha + 2(n - \alpha) = 2n - \alpha$. Hence, the ratio is at most

$$\min\{\frac{3n/2}{2n-\alpha}, \frac{5n/2 - 3\alpha/2}{2n-\alpha}\} = 1 + \min\{\frac{\alpha - n/2}{2n-\alpha}, \frac{n/2 - \alpha/2}{2n-\alpha}\}$$

which is maximized when $\alpha - n/2 = n/2 - \alpha/2$ or $\alpha = 2n/3$, in which case the ratio is 9/8. □

5 Hardness of Approximation

Theorem 5.1 *If there exists an $f(n)$-approximate algorithm for* MCS, *then there exists an $O(f(n))$-approximate algorithm for* Minimum Color.

Proof: Let G be a k-chromatic graph. Then $CS(G) \leq kn$, and the Sum Coloring algorithm gives a coloring with a sum of at most $knf(n)$. At least half of the vertices are colored with the first $2kf(n)$ colors. Use those $2kf(n)$ color classes, and recursively color the remaining at most $n/2$ vertices.

The total number of colors used is at most

$$2k\sum_{i=0}^{\infty} f(n/2^i).$$

We know from previous results [7] that $f(n) = \Omega(n^{0.2})$. Thus, the convex sum is at most

$$2k\sum_{i=0}^{\infty} \frac{1}{(2^{-0.2})^i} f(n) \leq 15kf(n).$$

Thus, we can color every k-colorable graph with at most $15kf(n)$ colors, for a performance ratio of $O(f(n))$. □

Feige and Kilian have recently shown that Minimum Color (and the chromatic number determination problem) is hard to approximated within $n^{1-\epsilon}$ factor. We obtain strong hardness bounds for MCS as a result.

Corollary 5.2 MCS *cannot be approximated within $n^{1-\epsilon}$, for any $\epsilon > 0$, unless $NP \subseteq ZPP$.*

The only drawback of this reduction is that it applies only to the (more interesting) constructive search problem; not the problem of approximating the minimum sum.

References

[1] Bar-Noy, A., Shachnai, H., and Tamir, T. (1996), On Chromatic Sums and Distributed Resource Allocation, in *Proceedings of the Fourth Israel Symposium on Theory and Computing Systems.*

[2] Bar-Noy, A., Bellare, M., Halldórsson, M., Shachnai, H., and Tamir, T. (1996), On Chromatic Sums and Distributed Resource Allocation. Accepted.

[3] Blum, A., and, Karger, D. An $O(n^{0.2143})$ coloring for 3-colorable graphs, *Inf. Proc. Let.*, to appear.

[4] Boppana, R. B. (1989), Optimal separations between concurrent-write parallel machines, in *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pp. 320–326.

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

[6] Goldberg, M. and Spencer, T. H. (1990), An efficient parallel algorithm that finds independent sets of guaranteed size, in *Proceedings of the First AnnualACM-SIAM Symposium on Discrete Algorithms*, pp. 219–225.

[7] Halldórsson, M.M., and Radhakrishnan, J. (1993), Approximating the Chromatic Sum. *Japan Advanced Institute of Science and Technology (JAIST), Research Report*, IS-RR-93-0002F.

[8] Kroon, L., Sen, A., Deng, H., and Roy, A. The Optimal Cost Chromatic Partition Problem for Trees and Interval Graphs. In *WG '96*, June 1996.

[9] Kubicka, E., Kubicki, G. and Kountanis, D. (1989), Approximation Algorithms for the Chromatic Sum, in *Proceedings of the First Great Lakes Computer Science Conf.*, Springer LNCS 507, pp. 15–21.

[10] Kubicka, E. and Schwenk, A.J. (1989), An Introduction to Chromatic Sums, in *Proceedings of the ACM Computer Science Conf.*, pp. 39-45.

[11] Nicoloso, S., Sarrafzadeh, M. and Song, X. (1994), On the Sum Coloring Problem on Interval Graphs, Istituto di Analisi dei Sistemi ed Informatica (IASI-CNR), R. 390.

[12] Vishwanathan, S. (1995), Private communication.