Coloring Graphs with Minimal Edge Load

Nitin Ahuja  Andreas Baltz  Benjamin Doerr
Anand Srivastav

Mathematisches Seminar, Bereich II, Christian-Albrechts-Universität zu Kiel,
Christian-Albrechts-Platz 4, 24118 Kiel, Germany.
E-mail: \{nia,aba,bed,asr\}@numerik.uni-kiel.de

Abstract

The load of a coloring \( \varphi : V \rightarrow \{ \text{red, blue} \} \) for a given graph \( G = (V, E) \) is a pair \( L_\varphi = (r_\varphi, b_\varphi) \), where \( r_\varphi \) is the number of edges with at least one red end-vertex and \( b_\varphi \) is the number of edges with at least one blue end-vertex. Our aim is to find a coloring \( \varphi \) such that \( l_\varphi := \max \{ r_\varphi, b_\varphi \} \) is minimized. We show that this problem is \( \text{NP-complete} \). For trees, we give a polynomial time algorithm computing an optimal solution. This has load at most \( m/2 + \Delta \log_2 n \), where \( m \) and \( n \) denote the number of edges and vertices respectively. For arbitrary graphs, a coloring with load at most \( \frac{2}{3} m + O(\sqrt{\Delta m}) \) can be found in deterministic polynomial time using a derandomized version of Azuma’s martingale inequality. This bound cannot be improved in general: almost all graphs have to be colored with load at least \( \frac{2}{3} m - \sqrt{3mn} \).

Key words: graph coloring, graph partitioning

1 Introduction

Let \( G = (V, E) \) be a graph. For a coloring \( \varphi : V \rightarrow \{ \text{red, blue} \} \) we define the load of \( \varphi \) by \( L_\varphi := (r_\varphi, b_\varphi) \), where \( r_\varphi \) counts the number of edges incident with at least one red vertex, and \( b_\varphi \) is the number of edges incident with at least one blue vertex. The aim of the Minimum Load Coloring Problem (MLCP) is to find a coloring \( \varphi \) such that \( l_\varphi := \max \{ r_\varphi, b_\varphi \} \) is minimized. To our knowledge there exists no prior literature on this particular problem. A generalization to hypergraphs was regarded by Aveev et al. [1]. They study scheduling aspects of an optical communication network with \( n \) nodes \( V = \{ v_1, \ldots, v_n \} \) and \( n \) hyperedges \( E = \{ E_1, \ldots, E_n \} \). Node \( v_i \) wants to send packets of data to a set \( E_i \subseteq V \) of other nodes. Given a set \( W = \{ w_1, \ldots, w_k \} \) of \( k \) available wavelengths, the aim is to find an assignment \( \varphi : V \rightarrow W \) of wavelengths to

Preprint submitted to Elsevier Science 25 March 2004
nodes such that the maximum load (number of packets) on any wavelength is minimized.

Notation: Throughout the paper let \( n \) denote the number of vertices of the graph under consideration, \( m \) the number of edges and \( \Delta \) the maximum vertex degree. We put \( l(G) \) to be the minimum (hence optimal) load among all vertex colorings of \( G \).

2 \ NP-Completeness

A reduction to \textsc{MinBisection} shows that MLCP is \( NP \)-complete.

Theorem 2.1 \( MLCP \in NPC \).

Proof. [Sketch] Let \( G = (V, E) \) be an instance of \textsc{MinBisection}. We construct an instance \( G' = (V', E') \) of MLCP by adding two cliques \( C_1 = (V_1, E_1) \), \( C_2 = (V_2, E_2) \), \( |V_1| = |V_2| = 2m + 2 \), \( E_1 := \mathcal{P}_2(V_1) \), \( E_2 := \mathcal{P}_2(V_2) \), a set of edges \( E := \{ (v, v') \mid v \in V, v' \in V_1 \cup V_2 \} \) connecting each \( v \in V \) to each clique vertex, and a matching \( M = (V_3, E_3) \) of size \( m \). Moreover, we define another instance \( G'' \) of MLCP by adding one more free matching edge \( e'' = \{ v''_1, v''_2 \} \) to \( G' \).

We claim that optimal solutions to MLCP on \( G' \) and \( G'' \) determine a minimum bisection in \( G \) and vice versa. Since \textsc{MinBisection} is \( NP \)-complete (cf. Karpinski [4]), and, obviously, MLCP \( \in NP \), this implies MLCP \( \in NPC \). Let \( OPT_B \) be the size of a minimum bisection of \( G \). It is enough to show that

\[
\begin{align*}
        l(G') &= \begin{cases}
            \frac{|E'|}{2} + \frac{OPT_B}{2} + n(m + 1) & \text{if } OPT_B \text{ is even} \\
            \frac{|E'|}{2} + \frac{OPT_B}{2} + n(m + 1) + \frac{1}{2} & \text{otherwise},
        \end{cases} \\
        \\
        l(G'') &= \begin{cases}
            \frac{|E''|}{2} + \frac{OPT_B}{2} + n(m + 1) + \frac{1}{2} & \text{if } OPT_B \text{ is odd} \\
            \frac{|E''|}{2} + \frac{OPT_B}{2} + n(m + 1) + 1 & \text{otherwise}.
        \end{cases}
\end{align*}
\]

This implies \( OPT_B = \begin{cases} 2l(G') - |E'| - 2n(m + 1) & \text{if } l(G') < l(G'') \\
2l(G'') - |E''| - 2n(m + 1) - 1 & \text{otherwise} \end{cases} \).

The first step of the proof is to show that each optimal coloring \( \varphi \) has to color \( G' \) such that \( C_1 \) and \( C_2 \) are monochromatic with different colors, and \( V \) contains as many red as blue vertices. This yields \( l_{\varphi} \geq \frac{|E'|}{2} + \frac{OPT_B}{2} + n(m + 1) \).

On the other hand, we may use the edges in \( E_3 \) to balance the number of monochromatic edges in both colors (here the parity of \( |E_3| \) is important). Thus only the number of dichromatic edges is important. This proves the claimed bounds for \( l(G') \) and \( l(G'') \). \( \square \)
3 Bounds and Algorithms for Trees

For trees, we prove the bound $l(G) \leq \frac{n-1}{2} + \Delta \log_2 n$. The key to this is the following more general lemma.

**Lemma 3.1** Given a tree $G = (V,E)$, $|V| = n$, and $p_1, p_2 \in \mathbb{N}$ with $p_1 + p_2 = n-1$, there is a red-blue coloring of $V$ such that at least $p_1 + 1 - \Delta \log_2 n$ edges are monochromatic red and at least $p_2 + 1 - \Delta \log_2 n$ are monochromatic blue.

From the lemma, we easily deduce the following.

**Theorem 3.1** Let $G$ be a tree. Then $l(G) \leq \frac{n}{2} + \Delta \log_2 n$.

The proof of Lemma 3.1 uses an inductive construction. Thus, there is an efficient algorithm for computing colorings with load at most $\frac{n}{2} + \Delta \log_2 n$. However, it is also possible to compute optimal colorings for trees efficiently.

**Theorem 3.2** On trees, MLCP can be solved in time $O(n^3)$.

*Proof. [Sketch] Let $G$ be a tree. We think of $G$ as a directed tree with an arbitrary root $a$ at level 0, the successors $N(a) := \{v \in V \mid (a,v) \in E\}$ of $a$ at level 1, etc. For each $v \in V$ we denote by $T_v$ the induced subtree of $G$ rooted in $v$. We define for each arbitrary subtree $G'$ of $G$ with root $a'$,

$$L_{G'} := \{(r, b) \mid (r, b) = L_\varphi \text{ for some coloring } \varphi \text{ of } G' \text{ with } \varphi(a') = \text{red}\},$$

the set of possible loads for $G'$. Suppose, we can efficiently compute $L_{G'}$. Since $|L_{G'}| \leq (n+1)^2$, we can also efficiently find the maximum load $l(G)$ of an optimal coloring by inspecting all $(r, b) \in L_{G'}$ and selecting the one with smallest maximum component. It is easy to see that $L_{G'}$ can be determined in polynomial time by iteratively computing $L_{T_v}$ for all $v \in V$ in reverse breadth first order. The iteration is based on two operations: consider a subtree $G'$ of $G$ with root $a' \neq a$, $v \in V$ with $(v, a') \in E$, and the tree $v + G' := (V(G') \cup \{v\}, E(G') \cup \{(v, a')\})$ obtained by appending the edge $(v, a')$ to $G'$. We define

$$v + L_{G'} := \{(r+1, b) \mid (r, b) \in L_{G'}\} \cup \{(b+1, r+1) \mid (r, b) \in L_{G'}\} \quad (1)$$

For two subtrees $G'_1, G'_2$ of $G$ that intersect only in their joint root $a'$, let $G'_1 + G'_2 := (V(G'_1) \cup V(G'_2), E(G'_1) \cup E(G'_2))$ be the composite tree. We define

$$L_{G'_1} + L_{G'_2} := \{(r_1 + r_2, b_1 + b_2) \mid (r_1, b_1) \in L_{G'_1}, (r_2, b_2) \in L_{G'_2}\}. \quad (2)$$

It is straightforward to prove that for all subtrees $G' = (V', E')$ of $G$ with root $a'$ and all $v \in V$ with $(v, a') \in E$, $L_{v + G'} = v + L_{G'}$. Moreover, for all subtrees $G'_1 = (V'_1, E'_1), G'_2 = (V'_2, E'_2)$ intersecting only in their joint root $a'$,
\[ \mathcal{L}_{G_1^* + G_2^*} = \mathcal{L}_{G_1^*} + \mathcal{L}_{G_2^*}. \] We conclude that \( \mathcal{L}_{T_v} = \sum_{\sigma \in E(N[v])} \mathcal{L}_{v+T_v} = \sum_{\sigma \in E(N[v])} v + \mathcal{L}_{T_v} \) for all \( v \in V \). Considering the complexity of the operations (1) and (2) we see that \( \mathcal{L}_G \) can be recursively computed in \( \sum_{v \in V} \text{deg}(v) \cdot O(n^3 + n^2) = O(n^3) \) steps.

We can reduce this time to \( O(n^3) \) by considering only “relevant” loads, \( \mathcal{R}_G := \{ (r, b) \mid (r, b) \in \mathcal{L}_G, b = \min \{ b' \mid (r, b') \in \mathcal{L}_G \} \} \). \( \mathcal{R}_G \) can be computed iteratively via operations similar to (1) and (2) that are performed on \( \mathcal{R}_G \) instead of \( \mathcal{L}_G \) and thus require only \( O(n) \) and \( O(n^3) \) steps, respectively. This yields the desired \( O(n^3) \) bound.

The iterative procedure to compute the optimal load can be easily modified to actually compute an optimal coloring. \( \square \)

4 Approximation Algorithms for Arbitrary Graphs

Let us first observe that the load of random colorings is less than \( \frac{3}{4}m + O(\sqrt{\Delta m}) \) with high probability. Since \( \frac{1}{2}m \) is a trivial lower bound for \( l_v \), we obtain a \((1.5 + \varepsilon)\)-approximation algorithm if \( \Delta = o(m) \). We will use the following Martingale inequality that can be found in McDiarmid [3]. It is an application of the well known inequality of Azuma [2].

**Lemma 4.1** Let \( X_1, \ldots, X_n \) be independent random variables taking values in some sets \( A_1, \ldots, A_n \). Let \( f : \prod_{i \in [n]} A_i \to \mathbb{R} \) such that \( |f(x) - f(y)| \leq c_i \) whenever \( x \) and \( y \) differ only in the \( i \)-th coordinate. Let \( X = (X_1, \ldots, X_n) \) and \( \mu = E(f(X)) \). Then for any \( \lambda \geq 0 \), \( \mathbb{P}(f(X) - \mu \geq \lambda) \leq \exp\left(-2\lambda^2 / \sum_{i=1}^n c_i^2\right) \).

**Theorem 4.1** There is a coloring \( \varphi \) such that \( l_\varphi \leq \frac{3}{4}m + \frac{\sqrt{(\ln 2)\Delta m}}{q} \). A random coloring satisfies \( \mathbb{P}\left(l_\varphi \geq \frac{3}{4}m + q\sqrt{(\ln 2)\Delta m}\right) \leq 2^{-q^2+1} \).

**Proof**. Let \( \varphi : V \to \{ \text{red}, \text{blue} \} \) such that \( \mathbb{P}(\varphi(v) = \text{red}) = \frac{1}{2} = \mathbb{P}(\varphi(v) = \text{blue}) \) independently for all \( v \in V \). Clearly, if two colorings \( \varphi_1, \varphi_2 \) differ only in the color of some vertex \( v \in V \), then \( |r_{\varphi_1} - r_{\varphi_2}| \leq \text{deg}(v) \). We compute \( E(r_{\varphi}) = \sum_{v \in E} \mathbb{P}(\exists v \in e : \varphi(v) = \text{red}) = \frac{3}{4}m \). Since \( \sum_{v \in V} \text{deg}(v)^2 \leq \sum_{v \in V} \text{deg}(v) \Delta = 2\Delta m \), for \( \lambda = \sqrt{(\ln 2)\Delta m} \) we have \( \mathbb{P}(r_{\varphi} > \frac{3}{4}m + \lambda) < \frac{1}{2} \). Thus with positive probability, both \( r_{\varphi} \) and \( b_{\varphi} \) are at most \( \frac{3}{4}m + \lambda \). In particular, a coloring with \( l_{\varphi} \leq \frac{3}{4}m + \lambda \) exists. The second statement follows in a similar manner. \( \square \)

**Theorem 4.2** A coloring \( \varphi \) such that \( l_\varphi \leq \frac{3}{4}m + \frac{\sqrt{(\ln 4)\Delta m}}{q} \) can be constructed in \( O(n^3) \) time.
For the proof we need a derandomized version (Theorem 4.3) of Azuma’s martingale inequality.

**Theorem 4.3 ([6])** Let \( f(X) = \sum_{i,j} \alpha_{ij} X_i X_j \) be a quadratic form satisfying the assumptions of Lemma 4.1. Let \( \delta \in (0,1) \) with \( 2 \exp(-2X^2 / \sum_{i=1}^n c_i^2) \leq 1 - \delta \). We can construct a \( X \in \{0,1\}^n \) with \( |f(X) - \mathbb{E}(f(X))| \leq \lambda \) in \( O(n^3 \log(\delta^{-1})) \) time.

**Proof.** [Sketch of Theorem 4.2] Let \((a_{ij})\) be the adjacency matrix of the graph \( G = (V,E) \) under consideration. We identify a two-coloring \( \varphi : V \to \{ \text{blue, red} \} \) with \( X \in \{0,1\}^n \). Let \( r(X) = \sum_{i=1}^n \sum_{j=1}^n \frac{a_{ij} X_i X_j}{2} + \sum_{i=1}^n \sum_{j=1}^n a_{ij} X_i (1 - X_j) \), and \( b(X) = \sum_{i=1}^n \sum_{j=1}^n \frac{a_{ij} (1 - X_i)(1 - X_j)}{2} + \sum_{i=1}^n \sum_{j=1}^n a_{ij} X_i (1 - X_j) \). Thus, \( r_\varphi = r(X), b_\varphi = b(X) \) and \( l_\varphi = f(X) := \max \{ r(X), b(X) \} \). Now, if we consider \( f \), the maximum of two quadratic forms, then the result can be proved by using Lemma 4.1 and Theorem 4.3. \( \square \)

The dependence on \( \Delta \) cannot be avoided. This is shown by star graphs. If \( \Delta = o(m) \), then the resulting bound of \((\frac{3}{4} + o(1))m\) cannot be improved in general, since, for the complete graph \( K_n \), \( l_\varphi \geq \frac{3}{4} n^2 - \frac{1}{2} m = (\frac{3}{4} + o(1))m \) for all colorings \( \varphi \). In a sense, almost all graphs have a load of \((\frac{3}{4} - o(1))m\).

**Theorem 4.4** Let \( m \geq 12n \). For a random multi-graph \( G = (V,E), |V| = n \) obtained by choosing \( m \) edges from \( \binom{n}{2} \) independently with repetition, we have \( l(G) \geq \frac{3}{4} m - \sqrt{3nm} \) with probability \( 1 - 2^{-n} \).

**References**


