Chicago Journal of Theoretical Computer Science

The MIT Press

Volume 2000, Article 1

Heuristics versus Completeness for Graph Coloring

ISSN 1073–0486. MIT Press Journals, Five Cambridge Center, Cambridge, MA 02142-1493 USA; (617)253-2889; *journals-orders@mit.edu, journals-info @mit.edu*. Published one article at a time in LaTeX source form on the Internet. Pagination varies from copy to copy. For more information and other articles see

- *http://mitpress.mit.edu/CJTCS/*
- *http://www.cs.uchicago.edu/publications/cjtcs/*
- *ftp://mitpress.mit.edu/pub/CJTCS*
- *ftp://cs.uchicago.edu/pub/publications/cjtcs*

The *Chicago Journal of Theoretical Computer Science* is abstracted or indexed in *Research Alert®*, *SciSearch®*, *Current Contents®/Engineering Computing & Technology*, and *CompuMath Citation Index®*.

The *Chicago Journal of Theoretical Computer Science* is a peer-reviewed scholarly journal in theoretical computer science. The journal is committed to providing a forum for significant results on theoretical aspects of all topics in computer science.

Heuristics versus Completeness for Graph Coloring

Jörg Rothe
rothe@informatik.uni-jena.de

29 February 2000

Abstract

We study the complexity of the problem `3-Colorability` when restricted to those input graphs on which a given graph coloring heuristic is able to solve the problem. The heuristics we consider include the sequential algorithm traversing the vertices of the graph in various orderings (e.g., by decreasing degree or in the recursive smallest-last order) as well as Wood's algorithm. For each heuristic considered here, we prove that the corresponding restriction of `3-Colorability` remains NP-complete.

1 Introduction

Graph coloring problems are of great importance in both theory and applications and have been intensely studied during the past century. Applications of constructing a graph coloring with as few colors as possible arise, for instance, in scheduling and partitioning problems (see Garey and Johnson [GJ79]). Unfortunately, the (optimization) problem of finding the chromatic number of a given graph is very complex, and even the (decision) problem of determining whether or not a given graph is 3-colorable (i.e., the vertices of the graph can be colored with three colors such that no two adjacent vertices have the same color) is one of the standard NP-complete problems ([Sto73]; see also [GJS76, GJ79]), thus being not efficiently solvable by current methods. However, due to the great amount of practical interest in finding efficient solutions—or at least good efficient approximate solutions—for these problems, it is not surprising that a large body of graph coloring heuristics have been proposed to date.

Such heuristic algorithms have been analyzed in depth both from a practical and a theoretical point of view; see, for example, the paper [MMI72], which compares certain heuristics by empirical tests on random graphs, and the work of Johnson [Joh74], which proves a number of prominent heuristics to have quite poor worst-case behavior in terms of their approximation ratio for the chromatic number. In fact, Feige and Kilian [FK96] recently proved that no deterministic polynomial-time algorithm can approximate the chromatic number within a factor of $\mathcal{O}(n^{1-\epsilon})$ for any fixed constant $\epsilon > 0$, unless NP = ZPP.

Johnson's results [Joh74] are to be taken as a warning that the success or failure of a specific graph coloring heuristic strongly depends on the form of the given input graph. In this paper, we study the complexity of the problem `3-Colorability` when restricted to those input graphs for which a given heuristic is able to solve it.[1] For any fixed heuristic algorithm `A` for graph coloring, define the restriction of `3-Colorability` that is induced by `A`: Given a graph G, can `A` on input G find a proper 3-coloring of G? We denote this problem by `A-3-Colorability`.

This approach is not quite new. Bodlaender, Thilikos, and Yamazaki [BTY97] showed that the problem `Independent Set` remains NP-complete when restricted to those input graphs on which a simple heuristic for finding independent sets (the so-called minimum-degree greedy algorithm, MDG for short) performs well. They also proved that the complexity of recognizing those input graphs for which MDG approximates the independence number (i.e., the size of a maximum independent set) within a certain fixed factor of optimality resides between the complexity classes coNP and P^{NP}. Solving the questions left open by Bodlaender, Thilikos, and Yamazaki [BTY97], Hemaspaandra and Rothe [HR98] determined the exact complexity of this recognition problem by proving it complete for the class $\mathrm{P}_{||}^{\mathrm{NP}}$ of problems solvable in polynomial time via parallel access to NP.

In this note, we investigate the above problem `A-3-Colorability` for a

[1]Usually, NP-complete graph problems are restricted with respect to certain "structural" graph properties such as planarity, bounded maximum degree, bipartiteness, and so on. For instance, it is known that the problem `3-Colorability` is in P when restricted to perfect graphs [GLS84], but remains NP-complete when restricted to planar graphs [Sto73]. In contrast, we restrict the problem with respect to the usefulness of a given heuristic.

number of graph coloring heuristics `A`, all of which are based on the sequential (or greedy) algorithm applied to a certain vertex ordering, such as the order by decreasing degree or the recursive smallest-last order of Matula, Marble, and Isaacson [MMI72]. Other heuristics that we consider, for instance, Wood's algorithm [Woo69], combine the sequential method with certain other strategies. We prove that the problem `A-3-Colorability` remains NP-complete for each heuristic `A` considered in this paper.

2 Preliminaries

All graphs considered in this paper are undirected graphs without reflexive edges. For any graph G, let $V(G)$ denote the set of vertices of G and let $E(G)$ denote the set of edges of G. For any set A, let $||A||$ denote the cardinality of A. For any vertex $v \in V(G)$, the *neighborhood* of v (denoted $N(v)$) is the set of vertices in G that are adjacent to v. For any vertex $v \in V(G)$, the *degree* of v is defined by $deg(v) \stackrel{\mathrm{df}}{=} ||N(v)||$. Given two disjoint graphs G and H, their *union* is defined to be the graph $F = G \cup H$ with vertex set $V(F) = V(G) \cup V(H)$ and edge set $E(F) = E(G) \cup E(H)$.

Given a graph G, a *coloring* of G is a mapping from $V(G)$ to the positive integers, which represent the colors. A coloring ψ of G is called *proper* if, for any two vertices x and y in $V(G)$, if $\{x, y\} \in E(G)$ then $\psi(x) \neq \psi(y)$. The *chromatic number* of graph G (denoted $\chi(G)$) is the minimum number of colors needed to properly color G. Given a fixed constant $k \geq 1$, graph G is said to be *k-colorable* if and only if there exists a proper coloring of G using no more than k colors.

3 Complexity of graph coloring when heuristics do well

Numerous heuristics for graph coloring problems have been proposed. Typically, such a heuristic consists of two parts: In the first part, a suitable ordering of the vertices of the graph is fixed. In the second part, the actual coloring algorithm is applied to the vertices in the fixed order to color the graph. A very basic coloring procedure is the *sequential algorithm* (sometimes called *greedy algorithm*), which proceeds as follows. Assume the vertices of the graph are given in the order $v_1, v_2, \ldots, v_n$. Assign color 1 to v_1. For each of the remaining vertices v_i in order, assign to v_i the minimum color available, that is, the smallest color that, so far, has not been assigned to any vertex adjacent to v_i. The sequential algorithm is denoted by `SEQ`.

Though the local action of the sequential algorithm appears to be quite reasonable, *globally* it may fail miserably, depending on the vertex ordering chosen. Johnson [Joh74] exhibited a sequence $G_3, \ldots, G_m, \ldots$ of graphs such that each G_m is 2-colorable, the size of G_m is linear in m, and yet the number of colors used by the sequential algorithm on input G_m is at least m for some (unfortunate) vertex ordering. Thus, for some ordering, the sequential algorithm achieves the worst approximation ratio (of the chromatic number) possible. Johnson [Joh74] proved similar results for a number of prominent graph coloring heuristics, most of which apply the sequential coloring algorithm to various vertex orderings that are obtained by seemingly reasonable procedures.

One such order-finding procedure is to order the vertices by *decreasing degree*. However, this is a rather static approach, since the place of any vertex in this ordering is independent of previously ordered vertices. A more flexible way of obtaining a vertex ordering is the recursive *smallest-last* ordering proposed by Matula, Marble, and Isaacson [MMI72], which dynamically proceeds as follows. Given a graph G with n vertices, choose any vertex of minimum degree to be the last vertex, v_n. For $i > 1$, let $v_i, \ldots, v_n$ be those vertices that have already been ordered. Choose any vertex of minimum degree in the subgraph of G induced by $V(G) - \{v_i, \ldots, v_n\}$ to be the next vertex, v_{i-1}, and proceed inductively backward until all vertices are ordered. Note that, in both orderings, there are nondeterministic choices to be made whenever there are more vertices than one of minimum degree at any point of the procedure.

We write `DD` to denote (the obvious nondeterministic procedure to obtain) any ordering by decreasing degree, and we write `SL` to denote (the above nondeterministic procedure to obtain) any smallest-last ordering. Combining the ordering and coloring algorithms to one algorithm, `A`, then specifies the metaproblem `A-3-Colorability` defined in the introduction. For instance, combining the smallest-last ordering with the sequential algorithm gives the following:

Decision problem: `SL-SEQ-3-Colorability`.
Instance: A graph G.
Question: Does there exist a sequence of nondeterministic choices

(between vertices of minimum degree) in the smallest-last ordering of $V(G)$ such that the sequential algorithm traversing $V(G)$ in that order properly 3-colors graph G?

First we show that the `3-Colorability` problem, restricted to those input graphs on which the sequential algorithm applied to some `DD` vertex ordering finds a solution, is no easier to solve than the general problem.[2]

Proposition 1 `DD-SEQ-3-Colorability` *is* NP-*complete.*

Proof To reduce `3-Colorability` to its restriction `DD-SEQ-3-Colorability`, fix any graph G and a vertex of largest degree, say w, in G. Without loss of generality, assume $deg(w) \geq 1$. For each vertex $v \in V(G) - \{w\}$, add $deg(w) - deg(v)$ new vertices $x_{v,1}$, $x_{v,2}$, $\ldots$, $x_{v,deg(w)-deg(v)}$ to G, and connect v with each $x_{v,i}$ by an edge. Call the resulting graph G'. Then all vertices in G' that are also vertices of G have the same degree, $deg(w)$, in G'. All new vertices $x_{v,i}$ in G' have degree 1. It follows that $G \in$ `3-Colorability` if and only if $G' \in$ `DD-SEQ-3-Colorability`. □

The construction given in the proof of Proposition 1 fails for the smallest-last ordering, since the new vertices $x_{v,i}$, which are added in order to suitably increase the degree of any given vertex v relative to other vertices in G', themselves have only degree 1. Thus, in general, they occur after v in any smallest-last ordering, and, as soon as they are `SL`-ordered, they are deleted from the graph and no longer increase the degree of v relative to other vertices still to be ordered.

The key construct to avoid this difficulty is given in Lemma 2 and is illustrated for a special case by graph $D_{u,4}$ shown in Figure 1. Consider any graph E' and suppose that some `SL` ordering of $V(E')$ is currently being computed, E is the subgraph of E' induced by the vertices still to be ordered, and that two vertices $u, v \in V(E)$ have a degree in E such that, say, u would be ranked above v in any `SL` ordering. The purpose of Lemma 2 is to show how to flip u and v in the `SL` ordering, assuming that the structure of E' requires such a flip for the sequential algorithm to find a proper 3-coloring of E' (if one exists).

[2] Proposition 1 clearly holds for the more general problem `K-Colorability` ("Given a graph G and a constant $k \leq ||V(G)||$, is it true that $\chi(G) \leq k$?") with $k \geq 3$ as well; we focus on `3-Colorability` for simplicity.

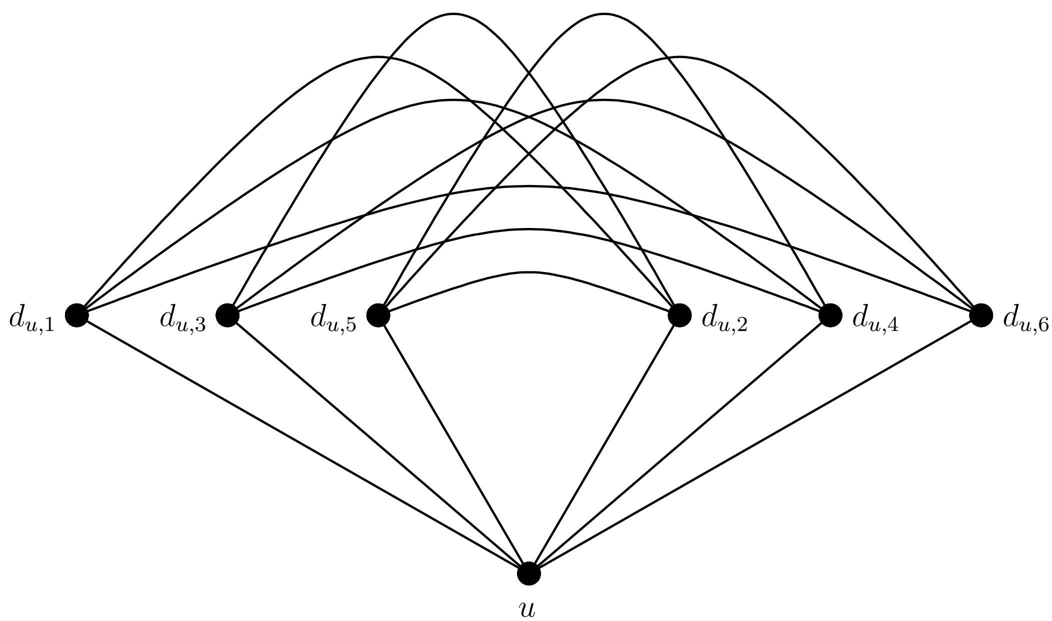

Figure 1: Graph $D_{u,4}$ for Lemma 2

Lemma 2 *Let E be any given graph. Let $u, v \in V(E)$ be vertices such that $deg(v) > deg(u) > 0$ in E, and let $s = deg(v)$. There exists a graph $D_{u,s}$ with $V(E) \cap V(D_{u,s}) = \{u\}$ and such that*

(i) *$deg(u) > deg(v)$ in $E \cup D_{u,s}$,*

(ii) *$V(D_{u,s}) \cup \{v\}$ can be* SL*-ordered such that each element of $V(D_{u,s}) - \{u\}$ is ranked above v and below u, and*

(iii) *algorithm* SEQ *applied to this order properly 3-colors $D_{u,s}$, regardless of which color $i \in \{1, 2, 3\}$ it starts with to color u.*

Proof Define graph $D_{u,s}$ by the vertex set

$$V(D_{u,s}) \stackrel{\text{df}}{=} \{u\} \cup \{d_{u,i} \mid 1 \leq i \leq 2(s-1)\}$$

and the edge set

$$E(D_{u,s}) \stackrel{\text{df}}{=} \{\{u, d_{u,i}\} \mid 1 \leq i \leq 2(s-1)\} \cup \{\{d_{u,i}, d_{u,j}\} \mid i \not\equiv j \bmod 2\}.$$

Note that the degree of u (relative to v) has increased in $E \cup D_{u,s}$ by $2(s-1)$. Since $deg(v) = s > 1$, this proves property (i). Property (ii) follows from property (i) and the fact that for each $d_{u,i}$ in $D_{u,s}$, $deg(d_{u,i}) = s$ in $E \cup D_{u,s}$: the vertex set $V(D_{u,s}) \cup \{v\}$ can be `SL`-ordered as $u, d_{u,1}, d_{u,2}, \ldots, d_{u,2(s-1)}, v$ in $E \cup D_{u,s}$. In particular, the sequential algorithm traversing $V(D_{u,s})$ in this order properly 3-colors $D_{u,s}$ no matter which color it starts with to color u. If color $i \in \{1, 2, 3\}$ is assigned to u, then color $1 + (i \bmod 3)$ is assigned to all vertices $d_{u,j}$ with odd j, and color $2 + (i \bmod 3)$ is assigned to all vertices $d_{u,j}$ with even j. This establishes property (iii) and proves the lemma. □

Theorem 3 `SL-SEQ-3-Colorability` *is* NP-*complete.*

Proof Instead of directly reducing `3-Colorability` to `SL-SEQ-3-Colorability` as in Proposition 1, it is useful to base our reduction on one "generic" instance of `3-Colorability`, namely, on the graph G constructed by Stockmeyer to reduce `3-SAT` to `3-Colorability` ([Sto73]; see also [GJS76]). Simplifying the technical proof details, this approach provides a reduction from `3-SAT` to `SL-SEQ-3-Colorability`.

First we recall the Stockmeyer reduction. Let ϕ be any given instance of `3-SAT` with n variables, $x_1, x_2, \ldots, x_n$, and m clauses, $C_1, C_2, \ldots, C_m$. The reduction maps ϕ to the graph G constructed as follows. The vertex set of G is defined by

$$V(G) \stackrel{\text{df}}{=} \{v_1, v_2, v_3\} \cup \{x_i, \bar{x}_i \mid 1 \leq i \leq n\} \cup \{y_{jk} \mid 1 \leq j \leq m \ \wedge \ 1 \leq k \leq 6\},$$

where the x_i and $\bar{x}_i$ are vertices representing the literals x_i and $\bar{x}_i$. The edge set of G is defined by

$$\begin{aligned}
E(G) \stackrel{\text{df}}{=} & \ \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_3\}\} \\
& \cup \{\{x_i, \bar{x}_i\} \mid 1 \leq i \leq n\} \\
& \cup \{\{v_3, x_i\}, \{v_3, \bar{x}_i\} \mid 1 \leq i \leq n\} \\
& \cup \{\{a_j, y_{j1}\}, \{b_j, y_{j2}\}, \{c_j, y_{j3}\} \mid 1 \leq j \leq m\} \\
& \cup \{\{v_2, y_{j6}\}, \{v_3, y_{j6}\} \mid 1 \leq j \leq m\} \\
& \cup \{\{y_{j1}, y_{j2}\}, \{y_{j1}, y_{j4}\}, \{y_{j2}, y_{j4}\} \mid 1 \leq j \leq m\} \\
& \cup \{\{y_{j3}, y_{j5}\}, \{y_{j3}, y_{j6}\}, \{y_{j5}, y_{j6}\} \mid 1 \leq j \leq m\} \\
& \cup \{\{y_{j4}, y_{j5}\} \mid 1 \leq j \leq m\},
\end{aligned}$$

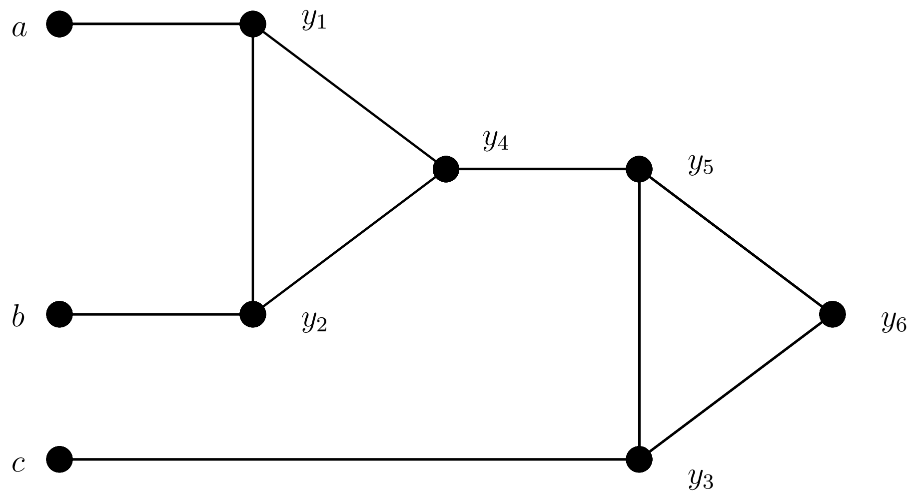

Figure 2: Graph H of the Stockmeyer reduction

where $a_j, b_j, c_j \in \bigcup_{1 \leq i \leq n} \{x_i, \bar{x}_i\}$ are vertices representing the literals occuring in clause $C_j = (a_j \vee b_j \vee c_j)$.

The graph H shown in Figure 2 is the key construct in this reduction, which uses m disjoint copies of H (with corresponding subscripts), one for each clause C_j of ϕ. Crucially, the correctness of the reduction (i.e., ϕ is satisfiable if and only if G is 3-colorable) results from the following two properties of graph H:

> Any coloring of the vertices a, b, and c that assigns color 1 to one of a, b, and c can be extended to a proper 3-coloring of H that assigns color 1 to y_6. (1)

> If ψ is a proper 3-coloring of H with $\psi(a) = \psi(b) = \psi(c) = i$, then $\psi(y_6) = i$. (2)

Now we transform G into a new graph F such that $F \in$ `SL-SEQ-3-Colorability` if and only if $G \in$ `3-Colorability` (if and only if $\phi \in$ `3-SAT`). For each vertex $u \in V(G) - \{y_{j6} \mid 1 \leq j \leq m\}$, we define a graph $D_{u,s}$ associated with u as in Lemma 2, for some suitable s. Lemma 2 merely explains one local part of the overall construction; globally, the size of graph $D_{u,s}$ may affect the size of some other graph $D_{u',s'}$. The respective values of s for the various graphs $D_{u,s}$ are chosen so as to "guide" the `SL` algorithm so that an ordering can be obtained for which the `SEQ` algorithm can properly 3-color F, assuming F *is* 3-colorable.

The vertex set of graph F is given by

$$\begin{aligned} V(F) \stackrel{\text{df}}{=} \; & V(D_{v_1,128}) \cup V(D_{v_2,64}) \cup V(D_{v_3,128}) \\ & \cup \bigcup_{1\le i\le n} V(D_{x_i,32}) \cup \bigcup_{1\le i\le n} V(D_{\bar{x}_i,32}) \\ & \cup \bigcup_{1\le j\le m} V(D_{y_{j1},16}) \cup \bigcup_{1\le j\le m} V(D_{y_{j2},16}) \cup \bigcup_{1\le j\le m} V(D_{y_{j4},8}) \\ & \cup \bigcup_{1\le j\le m} V(D_{y_{j3},4}) \cup \bigcup_{1\le j\le m} V(D_{y_{j5},4}) \cup \{y_{j6} \mid 1 \le j \le m\}. \end{aligned}$$

Note that $V(G) \subseteq V(F)$. The edge set of graph F is given by

$$\begin{aligned} E(F) \stackrel{\text{df}}{=} \; & E(G) \cup E(D_{v_1,128}) \cup E(D_{v_2,64}) \cup E(D_{v_3,128}) \\ & \cup \bigcup_{1\le i\le n} E(D_{x_i,32}) \cup \bigcup_{1\le i\le n} E(D_{\bar{x}_i,32}) \\ & \cup \bigcup_{1\le j\le m} E(D_{y_{j1},16}) \cup \bigcup_{1\le j\le m} E(D_{y_{j2},16}) \cup \bigcup_{1\le j\le m} E(D_{y_{j4},8}) \\ & \cup \bigcup_{1\le j\le m} E(D_{y_{j3},4}) \cup \bigcup_{1\le j\le m} E(D_{y_{j5},4}). \end{aligned}$$

This construction yields only a linear blow-up in the size of graph F (relative to the size of G), and the reduction is polynomial-time computable.

We now argue that the construction is correct. Suppose ϕ is satisfiable (and thus G is 3-colorable). Fix some satisfying assignment $\vec{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, where $\alpha_i = 1$ if variable x_i is set to true under this assignment, and $\alpha_i = 0$ otherwise. For any literal ℓ, let $\vec{\alpha}(\ell)$ denote the value assigned to ℓ by $\vec{\alpha}$; that is, $\vec{\alpha}(\ell) = \alpha_i$ if $\ell = x_i$, and $\vec{\alpha}(\ell) = 1 - \alpha_i$ if $\ell = \bar{x}_i$.

By construction and by properties (i) and (ii) of Lemma 2, the vertex set of F can be `SL`-ordered according to conditions (a)–(d) below. For convenience, we write $\vec{D}_{u,s}$ to denote the vertex set of graph $D_{u,s}$ given in the order $u, d_{u,1}, d_{u,2}, \ldots, d_{u,2(s-1)}$.

(a) $V(F)$ is ordered in three blocks: The first block contains the vertices

$$V(D_{v_1,128}) \cup V(D_{v_2,64}) \cup V(D_{v_3,128})$$

in the order specified by (b); the second block contains the vertices

$$\bigcup_{1\leq i\leq n} (V(D_{x_i,32}) \cup V(D_{\bar{x}_i,32}))$$

in the order specified by (c); and the third block contains all the remaining vertices of F in the order specified by (d).

(b) The first block is ordered as $\vec{D}_{v_3,128}, \vec{D}_{v_1,128}, \vec{D}_{v_2,64}$.

(c) For each i, $1 \leq i \leq n$, the vertex set $V(D_{x_i,32}) \cup V(D_{\bar{x}_i,32})$ can be SL-ordered as $\vec{D}_{x_i,32}, \vec{D}_{\bar{x}_i,32}$ if $\alpha_i = 1$, and it can be SL-ordered as $\vec{D}_{\bar{x}_i,32}, \vec{D}_{x_i,32}$ if $\alpha_i = 0$.

(d) For each j, $1 \leq j \leq m$, let $C_j = (a_j \vee b_j \vee c_j)$ be the jth clause of ϕ, with literals $a_j, b_j, c_j \in \bigcup_{1\leq i\leq n}\{x_i, \bar{x}_i\}$. Let $\vec{\alpha}(C_j)$ be a shorthand for $(\vec{\alpha}(a_j), \vec{\alpha}(b_j), \vec{\alpha}(c_j))$. Note that since $\vec{\alpha}$ satisfies ϕ, $\vec{\alpha}(C_j) \neq (0,0,0)$ for each j. Recall that the literals a_j, b_j, c_j in C_j are identified with the corresponding vertices of G. For each j, let Y_j denote the vertex set associated with C_j, that is, $Y_j \stackrel{\mathrm{df}}{=} V(D_{y_{j1},16}) \cup V(D_{y_{j2},16}) \cup V(D_{y_{j3},4}) \cup V(D_{y_{j4},8}) \cup V(D_{y_{j5},4}) \cup \{y_{j6}\}$. Depending on $\vec{\alpha}(C_j)$, Y_j can be SL-ordered as follows:

  (d1) If $\vec{\alpha}(C_j) \in \{(1,1,1),(1,1,0),(0,1,1),(0,1,0)\}$, then Y_j is ordered as $\vec{D}_{y_{j1},16}$, $\vec{D}_{y_{j2},16}$, $\vec{D}_{y_{j4},8}$, $\vec{D}_{y_{j3},4}$, $\vec{D}_{y_{j5},4}$, y_{j6}.

  (d2) If $\vec{\alpha}(C_j) \in \{(1,0,1),(1,0,0)\}$, then Y_j is ordered as $\vec{D}_{y_{j2},16}$, $\vec{D}_{y_{j1},16}$, $\vec{D}_{y_{j4},8}$, $\vec{D}_{y_{j3},4}$, $\vec{D}_{y_{j5},4}$, y_{j6}.

  (d3) If $\vec{\alpha}(C_j) \in \{(0,0,1)\}$, then Y_j is ordered as $\vec{D}_{y_{j1},16}$, $\vec{D}_{y_{j2},16}$, $\vec{D}_{y_{j4},8}$, $\vec{D}_{y_{j5},4}$, $\vec{D}_{y_{j3},4}$, y_{j6}.

The relative order between vertices not specified by conditions (a)–(d) is irrelevant for the argument and may be fixed arbitrarily (consistent with the rules of the SL-ordering).

The correctness of the reduction follows from the next property, which holds for each j, $1 \leq j \leq m$:

> Assume that the vertices representing the literals of C_j are colored such that only colors 2 and 3 are assigned, and color 2 is assigned to at least one of a_j, b_j, and c_j. Then, the `SEQ` algorithm (traversing Y_j in one of the orders given by (d1), (d2), or (d3), depending on $\vec{\alpha}(C_j)$) properly 3-colors the subgraph of F induced by $\{a_j\} \cup \{b_j\} \cup \{c_j\} \cup Y_j$ such that color 2 is assigned to y_{j6}. (3)

Property (3) is similar to property (1) of the Stockmeyer reduction, suitably tailored to the specifics of the `SEQ` algorithm, and it straightforwardly follows from property (iii) of Lemma 2 and the vertex order of Y_j given in (d) above.

Observe that in the current case (ϕ is satisfiable), there exists a vertex ordering of F consistent with conditions (a)–(d) so that the `SEQ` algorithm can properly 3-color graph F. In particular, it computes a coloring ψ of F such that

- $\psi(v_3) = 1$, $\psi(v_1) = 2$, $\psi(v_2) = 3$;
- for each i with $1 \leq i \leq n$,
  - if $\alpha_i = 1$, then $\psi(x_i) = 2$ and $\psi(\bar{x}_i) = 3$,
  - if $\alpha_i = 0$, then $\psi(x_i) = 3$ and $\psi(\bar{x}_i) = 2$.

Since $\vec{\alpha}$ is a satisfying assignment of ϕ, coloring ψ assigns color 2 to at least one (vertex representing a) literal of C_j, for each j, $1 \leq j \leq m$. By property (3), for each j, ψ is a proper 3-coloring of the subgraph of F induced by $\{a_j\} \cup \{b_j\} \cup \{c_j\} \cup Y_j$ and satisfies $\psi(y_{j6}) = 2$. Property (iii) of Lemma 2 implies that ψ (as specified so far) can be extended to a proper 3-coloring of F.

Conversely, suppose ϕ is not satisfiable, so G is not 3-colorable. By construction, G is a subgraph of F; so F is not 3-colorable. Thus, in particular, $F \notin$ `SL-SEQ-3-Colorability`. □

Matula, Marble, and Isaacson [MMI72] and Johnson [Joh74] proposed generalizations of the sequential algorithm that allow the occasional *interchange* of two colors (in the coloring being computed) subject to certain sets of constraints. Both sequential-with-interchange algorithms may be combined with any vertex ordering; we focus on the `DD` and `SL` orderings. Since the $\texttt{SEQINT}_i$ algorithms include the sequential algorithm as a special case (in which no interchange is performed), we immediately have the following corollaries from, respectively, Proposition 1 and Theorem 3.

Corollary 4 *Both* $\texttt{DD-SEQINT}_1$`-3-Colorability` *and* $\texttt{DD-SEQINT}_2$`-3-Colorability` *are* NP-*complete.*

Corollary 5 *Both* $\texttt{SL-SEQINT}_1$`-3-Colorability` *and* $\texttt{SL-SEQINT}_2$`-3-Colorability` *are* NP-*complete.*

The last heuristic considered in this paper is the algorithm of Wood [Woo69] which, given an input graph G with n vertices, proceeds in two stages as follows. In the first stage, all $n(n-1)/2$ pairs of distinct vertices are ordered by decreasing similarity, where the *similarity* of two distinct vertices x and y is defined to be

$$sim(x, y) \stackrel{\text{df}}{=} \begin{cases} 0 & \text{if } \{x, y\} \in E(G) \\ ||N(x) \cap N(y)|| & \text{otherwise.} \end{cases}$$

Given this order, G is partially colored in the first stage by executing the following steps for each pair $\{x, y\}$ in turn. In what follows, let c be a variable whose value gives the number of colors used so far.

(1) If $sim(x, y) = 0$, then halt.

(2) If both x and y are colored, then go to next pair.

(3) If one vertex, say, x, is colored, and the other one, y, is uncolored, then do the following:

 (3a) if $deg(y) < c$, then go to next pair;

 (3b) if some vertex adjacent to y has the same color as x, then go to next pair;

 (3c) otherwise, assign to y the color assigned to x.

(4) If both x and y are uncolored, then do the following:

 (4a) if both $deg(x) < c$ and $deg(y) < c$, then go to next pair;

 (4b) otherwise, assign to both x and y the minimum color available (i.e., the smallest color $j \geq 1$ such that neither x nor y is adjacent to a vertex colored j).

After the first stage, there may remain some uncolored vertices. If so, the coloring of G is completed in the second stage using the `DD-SEQ` algorithm. Wood's algorithm is denoted by `WOOD`. Note that both stages of Wood's algorithm contain some amount of nondeterminism: In the first stage, we may choose between different vertex pairs of the same similarity (when there are more than one); in the second stage, we may choose among several vertices of minimum degree.

Theorem 6 `WOOD-3-Colorability` *is* NP-*complete.*

Proof The proof is similar to the proof of Proposition 1, the difference being that now we have to equalize the similarity between pairs of vertices instead of the degree of vertices. Let G be any given graph. We transform G into a new graph H such that G is 3-colorable if and only if H can be 3-colored by Wood's algorithm.

Let $s' \stackrel{\text{df}}{=} \max\{sim(x, y) \mid x, y \in V(G) \text{ with } x \neq y\}$ be the maximum similarity of all vertex pairs in G, and let $s \stackrel{\text{df}}{=} \max\{3, s'\}$.

The vertex set of H is given by

$$V(H) \stackrel{\text{df}}{=} V(G) \cup \{v' \mid v \in V(G)\} \cup \{x_{v,i} \mid v \in V(G) \ \wedge \ 1 \leq i \leq s\},$$

where the $||V(G)||$ vertices v' and the $s||V(G)||$ vertices $x_{v,i}$ are new. The edge set of H is given by

$$\begin{aligned} E(H) \stackrel{\text{df}}{=} \ & E(G) \cup \{\{v, x_{v,i}\} \mid v \in V(G) \ \wedge \ 1 \leq i \leq s\} \\ & \cup \{\{x_{v,i}, v'\} \mid v \in V(G) \ \wedge \ 1 \leq i \leq s\}. \end{aligned}$$

This reduction is polynomial-time computable, since the similarity of all vertex pairs in G (and hence s) can be computed in time polynomial in the size of (the encoding of) G. By construction, $sim(v, v') = s$ for all vertices $v \in V(G)$, and the similarity of all other vertex pairs of H is at most s. Thus, all vertex pairs of the form $\{v, v'\}$, for $v \in V(G)$, can be ranked above all other vertex pairs of H in the first stage of Wood's algorithm. Suppose G is 3-colorable. Let ψ be any fixed proper 3-coloring of G, and define the three color classes $V_i \stackrel{\text{df}}{=} \{v \in V(G) \mid \psi(v) = i\}$, for $i \in \{1, 2, 3\}$, that correspond to ψ. Let $v_1, v_2, \ldots, v_n$ be an ordering of $V(G)$ such that all vertices from V_1 come first, followed by all vertices from V_2, which in turn are followed by all vertices from V_3. Consider the corresponding order

$\{v_1, v_1'\}, \{v_2, v_2'\}, \dots, \{v_n, v_n'\}$ of the first n vertex pairs of H. Since $deg(v) \geq s \geq 3$ for all $v \in V(G)$ and since $deg(v') = s \geq 3$ for the corresponding vertices v', line (4b) of the first stage of Wood's algorithm is executed n times and assigns color $\psi(v)$ in G to both v and v' in H. The only vertices of H as yet uncolored are those of the form $x_{v,i}$. It is then not hard to see that ψ can be extended by Wood's algorithm to a proper 3-coloring of H.

Conversely, if G is not 3-colorable, then H is not 3-colorable, and consequently $H \notin$ `WOOD-3-Colorability`. □

Finally, we mention some open questions. What is the complexity of the related problem of recognizing those graphs G for which a fixed heuristic can *find* the chromatic number $\chi(G)$ (instead of merely deciding whether $\chi(G) \leq k$ as with K-`Colorability`)? What about the recognition problem for *approximating* $\chi(G)$ within a fixed factor $r \geq 1$ (r rational) of optimal? As mentioned in the introduction, these questions were successfully resolved [HR98] for the case of approximating the independence number with respect to the minimum-degree greedy heuristic. However, lower bounds for graph coloring problems in general tend to be harder to achieve than those for independent set problems. (But note that $\mathrm{P}_{||}^{\mathrm{NP}}$ also is an upper bound for these chromatic number problems—just like $\mathrm{P}_{||}^{\mathrm{NP}}$ is an upper bound for the independence number problems.) The results of the present paper may be seen as a first step toward resolving the more demanding questions raised above. In fact, the construction given in [HR98] is based on the NP-completeness result of [BTY97] for the restriction of `Independent Set` to those input graphs on which MDG works well. Thus, one may hope that the results of the present paper will lead to progress regarding the above questions.

Acknowledgments

I am very grateful to Lane Hemaspaandra and Dieter Kratsch for helpful comments on an earlier draft of this paper and for pointers to the literature.

Acknowledgment of support

Author was supported in part by grant NSF-INT-9815095/DAAD-315-PPP-gü-ab and a Heisenberg Fellowship of the Deutsche Forschungsgemeinschaft.

References

[BTY97] H. Bodlaender, D. Thilikos, and K. Yamazaki. It is hard to know when greedy is good for finding independent sets. *Information Processing Letters*, 61:101–106, 1997.

[FK96] U. Feige and J. Kilian. Zero knowledge and the chromatic number. In *Proceedings of the 11th Annual IEEE Conference on Computational Complexity*, pages 278–287. IEEE Computer Society Press, 1996.

[GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.

[GJS76] M. Garey, D. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1:237–267, 1976.

[GLS84] M. Grötschel, L. Lovász, and A. Schrijver. Polynomial algorithms for perfect graphs. In *Perfect Graphs*, volume 21 of *Annals of Discrete Mathematics*, pages 325–356. North-Holland, Amsterdam, 1984.

[HR98] E. Hemaspaandra and J. Rothe. Recognizing when greed can approximate maximum independent sets is complete for parallel access to NP. *Information Processing Letters*, 65(3):151–156, February 1998.

[Joh74] D. Johnson. Worst case behavior of graph coloring algorithms. In *Proceedings of the 5th Southeastern Conference on Combinatorics, Graph Theory, and Computing*, pages 513–527. Utilitas Mathematica Publishing, February/March 1974.

[MMI72] D. Matula, G. Marble, and J. Isaacson. Graph coloring algorithms. In R. Read, editor, *Graph Theory and Computing*, pages 109–122. Academic Press, New York, 1972.

[Sto73] L. Stockmeyer. Planar 3-colorability is NP-complete. *SIGACT News*, 5(3):19–25, 1973.

[Woo69] D. Wood. A technique for coloring a graph applicable to large scale time-tabling problems. *The Computer Journal*, 12:317–319, 1969.