The complexity of coloring graphs without long induced paths

Gerhard J. Woeginger [*] Jiří Sgall [†]

Abstract

We discuss the computational complexity of determining the chromatic number of graphs without long induced paths. We prove NP-completeness of deciding whether a P_8-free graph is 5-colorable and of deciding whether a P_{12}-free graph is 4-colorable. Moreover, we give a polynomial time algorithm for deciding whether a P_5-free graph is 3-colorable.

Keywords: graph coloring – chromatic number – computational complexity – induced path.

1 Introduction

A graph $G = (V, E)$ is *k-colorable* if there exists a coloring $f : V \to \{1, \ldots, k\}$ such that $f(u) \neq f(v)$ for every edge $[u, v] \in E$. The *chromatic number* $\chi(G)$ of graph G is the smallest k for which G is k-colorable. A graph $G = (V, E)$ is *P_m-free* if it does not contain the path P_m on m vertices as an induced subgraph. For $v \in V$, we denote by $\Gamma(v) = \{w \in V : [v, w] \in E\}$ the neighborhood of v. For $W \subseteq V$, denote $\Gamma(W) = \bigcup_{w \in W} \Gamma(w)$.

In this note we discuss the computational complexity of deciding whether a given P_m-free graph G is k-colorable. For all $m \geq 2$ and $k \geq 2$, we call the corresponding coloring problem $P(m, k)$.

The problems $P(m, 2)$ are polynomially solvable, since 2-coloring is polynomially solvable even for arbitrary graphs; see e.g. Garey & Johnson [1]. Similarly, the problems of type $P(4, k)$ are polynomially solvable: A graph is P_4-free if and only if it is a cograph, and the chromatic number of a cograph can be determined in polynomial time; see Golumbic [2]. (The special cases $P(2, k)$ and $P(3, k)$ are trivial since P_3-free graphs are disjoint unions of cliques.)

In this note we will prove the following results.

Theorem 1.1 *It can be decided in polynomial time whether a P_5-free graph is 3-colorable.*

Theorem 1.2 *It is NP-complete to decide (a) whether a P_8-free graph is 5-colorable, and (b) whether a P_{12}-free graph is 4-colorable.*

These results and some of their implications are summarized in the table in Figure 1.

The proof of Theorem 1.1 is in Section 2, and the proofs of statements (a) and (b) in Theorem 1.2 are given in Sections 3 and 4, respectively.

[*] gwoegi@opt.math.tu-graz.ac.at. Institut für Mathematik, Technische Universität Graz, Steyrergasse 30, A-8010 Graz, Austria. Supported by the START program Y43-MAT of the Austrian Ministry of Science.

[†] sgall@math.cas.cz. Mathematical Inst., AS CR, Žitná 25, CZ-11567 Praha 1, Czech Republic. Partially supported by grant A1019901 of GA AV ČR, grant 201/01/1195 of GA ČR, and project LN00A056 of MŠMT CR.

m	4	5	6	7	8	9	10	11	12	13	14	15	...
$k=3$	P	P	??	??	??	??	??	??	??	??	??	??	...
$k=4$	P	??	??	??	??	??	??	??	NP	NP	NP	NP	...
$k=5$	P	??	??	??	NP	NP	NP	NP	NP	NP	NP	NP	...
$k=6$	P	??	??	??	NP	NP	NP	NP	NP	NP	NP	NP	...
$k=7$	P	??	??	??	NP	NP	NP	NP	NP	NP	NP	NP	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...

Figure 1: A summary of complexity results for the coloring problems of type (m,k). An entry 'P' means that the problem is polynomially solvable, an entry 'NP' means that the problem is NP-hard, and an entry '??' means that the complexity of the problem is currently unknown.

2 The polynomial time result

In this section we prove Theorem 1.1. Consider a P_5-free graph $G=(V,E)$; without loss of generality we assume that G is connected. Our polynomial time algorithm distinguishes two cases for G; not that it is easy to distinguish the cases in time $O(n^3)$.

- Case 1: G is triangle-free. In this case we prove that the graph is 3-colorable and, in addition, we show how to construct a 3-coloring.
- Case 2: G contains a triangle. In this case we give an algorithm which reduces the problem to 2-satisfiability of propositional formulas.

Case 1. If G is bipartite, 2-coloring is constructed easily. Otherwise G must contain an induced cycle of odd lengh. It cannot contain an induced cycle of length seven or more, since such an induced cycle would also yield an induced P_5. The case of C_3, i.e., a triangle is excluded in Case 1. Thus the induced cycle is C_5; denote its vertices v_0,v_1,v_2,v_3,v_4 in this ordering along the cycle. Denote $V_0=\{v_0,v_1,v_2,v_3,v_4\}$.

Consider any vertex x adjacent to a vertex in V_0. If x is adjacent to only a single vertex from V_0, say to v_0, then x,v_0,v_1,v_2,v_3 would form an induced P_5 in G. If x is adjacent to two adjacent vertices from V_0, then these three vertices would form a triangle in G. As a consequence, x must have exactly two neighbors in V_0, and these two neighbors are not adjacent to each other. Denote by W_i, $0\le i\le 4$, the set of all vertices in $V_0\cup V_1$ that are adjacent to v_{i-1} and v_{i+1} (all indices are taken modulo 5).

We claim that $V=V_0\cup W_0\cup W_1\cup W_2\cup W_3\cup W_4$. If not, since G is connected, there exists a vertex x not adjacent to V_0 but adjacent to some $y\in W_i$. Then $x,y,v_{i+1},v_{i+2},v_{i+3}$ is an induced P_4.

Two distinct vertices x and y in some set $W_i\cup W_{i+2}$ cannot be adjacent to each other, since this would yield a triangle x,y,v_{i+1}.

Consequently G can be 3-colored by coloring all vertices in W_0 and W_2 by 1, all vertices in W_1 and W_3 by 2, and all vertices in W_3 by 3. Moreover, the partition W_0,W_1,W_2,W_3,W_4 can be computed in polynomial time.

Case 2. Consider an arbitrary triangle in G and color its vertices by 1, 2, 3. As long as there is an uncolored vertex v that has neighbors of two different colors, color v with the remaining third color. Note that all these moves are forced. If we find an uncolored vertex that has neighbors of three different colors, then we conclude that G is not 3-colorable. When this process terminates, we denote by V_0 the set of all colored vertices, we denote by V_1 (respectively, V_2 and V_3) the set of all uncolored vertices that are adjacent to some colored vertex of color 1 (respectively, color 2 and 3). Furthermore, we denote by V_4 the set of all vertices in $V - V_0 \cup V_1 \cup V_2 \cup V_3$ that are adjacent to some vertex in $V_1 \cup V_2 \cup V_3$.

We claim that $V = V_0 \cup V_1 \cup V_2 \cup V_3 \cup V_4$. Suppose otherwise. Then since G is connected, there must exist some vertex x outside of $V_0 \cup V_1 \cup V_2 \cup V_3 \cup V_4$ that is adjacent to some vertex y in V_4. Vertex y is adjacent to some vertex in $V_1 \cup V_2 \cup V_3$, say to vertex $z \in V_1$. Vertex z is adjacent to a 1-colored vertex v_1 in V_0, and v_1 is adjacent to a 2-colored vertex v_2 in V_0. But then x, y, z, v_1, v_2 form an induced P_5 in G. This contradiction shows that indeed $V = V_0 \cup V_1 \cup V_2 \cup V_3 \cup V_4$.

Lemma 2.1 *Let $C = \{x\}$ be a connected component in V_4 that consists of a single vertex. If the graph G is 3-colorable, then one of the following two situations holds.*

(i) $\Gamma(x) \subseteq V_i$ for some i with $1 \le i \le 3$. Any 3-coloring of $V_0 \cup V_1 \cup V_2 \cup V_3$ uses at most two colors on $\Gamma(x) \subseteq V_i$. Any such 3-coloring can be extended to vertex x.

(ii) $\Gamma(x) \cap V_i \neq \emptyset$ and $\Gamma(x) \cap V_j \neq \emptyset$ for some i, j with $1 \le i < j \le 3$. In this case $\Gamma(x)$ forms a complete bipartite subgraph of G with bipartition $\Gamma(x) \cap V_i$ and $\Gamma(x) \cap V_j$. In any 3-coloring of G, all vertices in $\Gamma(x) \cap V_i$ must receive the same color and all vertices in $\Gamma(x) \cap V_j$ must receive the same color. Any such 3-coloring can be extended to vertex x.

Proof. First suppose that there are two vertices $w_1 \in V_1 \cap \Gamma(x)$ and $w_2 \in V_2 \cap \Gamma(x)$ such that $[w_1, w_2] \notin E$. Then w_1 is adjacent to a 1-colored vertex v_1 in V_0, and this vertex v_1 is adjacent to a 3-colored vertex v_3 in V_0. But then w_2, x, w_1, v_1, v_3 would form an induced P_5 in G. As a consequence, any two vertices in $\Gamma(x)$ that do not form an edge in G must both belong to the same set V_1, V_2, V_3.

We now consider several cases. First assume that $\Gamma(x)$ intersects all three sets V_1, V_2, and V_3. Then three vertices in $\Gamma(x) \cap V_1$, $\Gamma(x) \cap V_2$ and $\Gamma(x) \cap V_3$ together with x would form a K_4, and G would not be 3-colorable in this case. Next assume that $\Gamma(x) \cap V_i \neq \emptyset$ and $\Gamma(x) \cap V_j \neq \emptyset$ for some i, j with $1 \le i < j \le 3$. By the observation in the preceding paragraph, G must contain all edges between $\Gamma(x) \cap V_i$ and $\Gamma(x) \cap V_j$. Then $\Gamma(x) \cap V_i$ and $\Gamma(x) \cap V_j$ must both be independent sets, since otherwise G would contain a K_4. Hence, in this case we are in situation (ii). Finally, $\Gamma(x) \subseteq V_i$ might hold for some i with $1 \le i \le 3$ as in situation (i). ■

Lemma 2.2 *Let C be a connected component in V_4 that contains at least two vertices. Denote the set $\Gamma(C) \cap (V_1 \cup V_2 \cup V_3)$ by D.*

(i) All vertices in C are adjacent to all vertices in D.

(ii) If the graph G is 3-colorable, then the component C is bipartite and D forms an independent set.

(iii) In any 3-coloring of G, all vertices in D receive the same color.

(iv) Assume that C is bipartite. Then any 3-coloring of $V_0 \cup V_1 \cup V_2 \cup V_3$ in which all vertices in D have the same color can be extended to a 3-coloring of $V_0 \cup V_1 \cup V_2 \cup V_3$ and C.

Proof. (i) Let $x, y \in C$ with $[x, y] \in E$. Suppose that there exists some $z \in D$ that is adjacent to x but not to y; without loss of generality $z \in V_1$. Then z is adjacent to some 1-colored v_1 in V_0, and v_1 is adjacent to a 2-colored vertex v_2 in V_0. Then x, y, z, v_1, v_2 form an induced P_5 in G. This contradiction shows that x and y have exactly the same neighbors in D. This yields statement (i).

Statement (ii) is an immediate consequence of (i): Any 3-coloring of G must color the component C with two colors, and the neighborhood D of C with the third color. This also yields statement (iii). Statement (iv) is straightforward. ■

With the help of Lemmas 2.1 and 2.2, we will now translate the 3-coloring problem into a TWO-SATISFIABILITY problem. Since the colors of vertices in V_0 have already been fixed, we will concentrate on the vertices in V_1, V_2, and V_3.

- For every $v \in V_i$ $(1 \leq i \leq 3)$ we introduce two variables $x(v, j)$ and $x(v, k)$ such that $\{i, j, k\} = \{1, 2, 3\}$. A TRUE variable $x(v, j)$ will mean that vertex v is colored by color j. By introducing two clauses with two literals, we can enforce that exactly one of $x(v, j)$ and $x(v, k)$ must be TRUE and the other one must be false.
- Consider a connected component $C = \{x\}$ in V_4. If neither situation (i) nor (ii) from Lemma 2.1 hold, then we stop the construction since G is not 3-colorable. If situation (i) holds, we do not need to do anything. And if situation (ii) holds, then we introduce several clauses (each with two literals) that enforce that all vertices in $\Gamma(x) \cap V_i$ get the same color and that all vertices in $\Gamma(x) \cap V_j$ get the same color.
- Consider a connected component C in V_4 that contains at least two vertices. If the neighbors of C in $V_1 \cup V_2 \cup V_3$ do not form an independent set, then we stop the construction since G is not 3-colorable by Lemma 2.2. And if they do form an independent set, then we introduce several clauses (each with two literals) that enforce that all these vertices get the same color.

The resulting instance of TWO-SATISFIABILITY can be solved in polynomial time; see e.g. Garey & Johnson [1]. If this TWO-SATISFIABILITY does not have a satisfying truth assignment, then by Lemmas 2.1 and 2.2 the graph G cannot be 3-colorable. On the other hand, if this TWO-SATISFIABILITY does have a satisfying truth assignment, then we can translate it into a 3-coloring for $V_0 \cup V_1 \cup V_2 \cup V_3$ and we can use Lemmas 2.1 and 2.2 to extend this coloring to a 3-coloring for V_4. Since all this can clearly be done in polynomial time, the proof of Theorem 1.1 is complete.

3 The first NP-hardness proof

In this section we prove Theorem 1.2(a). The reduction is from the NP-hard THREE-SATISFIABILITY problem (Garey & Johnson [1]): Given a set $X = \{x_1, \ldots, x_n\}$ of logical variables, and a set $C = \{c_1, \ldots, c_m\}$ of three-literal clauses over X, does there exist a truth assignment for X that simultaneously satisfies all clauses in C?

Now consider an arbitrary instance I of THREE-SATISFIABILITY. We define a P_8-free graph $G_1 = (V_1, E_1)$ that is 5-colorable if and only if this instance I has answer YES:

- For every variable $x \in X$, there is a vertex $a(x)$ that corresponds to the unnegated literal x, and a vertex $a(\overline{x})$ that corresponds to the negated literal $\overline{x}$. These vertices are connected to each other by an edge.

- For every clause $c \in C$ that consists of the literals u_1, u_2, u_3, there are seven corresponding vertices $b_1(c)$, $b_2(c)$, $b_3(c)$, $b_4(c)$ and $b(c, u_1)$, $b(c, u_2)$, $b(c, u_3)$. The three vertices $b_1(c)$, $b_2(c)$, $b_3(c)$ form a triangle. Moreover, $b_1(c)$ is connected to $b(c, u_2)$ and $b(c, u_3)$, $b_2(c)$ is connected to $b(c, u_1)$ and $b(c, u_3)$, and $b_3(c)$ is connected to $b(c, u_1)$ and $b(c, u_2)$. Vertex $b_4(c)$ is connected to $b(c, u_1)$, $b(c, u_2)$, $b(c, u_3)$. For $i = 1, 2, 3$ vertex $b(c, u_i)$ is connected to the vertex $a(u_i)$. See Figure 2 for an illustration.

- All vertices $a(x)$ and $a(\overline{x})$ with $x \in X$ are connected to all vertices $b_1(c)$, $b_2(c)$, $b_3(c)$, $b_4(c)$ with $c \in C$.

- Finally, there is a single dummy vertex d that is connected to all clause vertices $b_1(c)$, $b_2(c)$, $b_3(c)$, $b_4(c)$ and $b(c, u_1)$, $b(c, u_2)$, $b(c, u_3)$ with $c = u_1 \vee u_2 \vee u_3$ in C.

This completes the description of the graph G_1. Note that G_1 may contain an induced P_7 that runs through $a(\overline{x_1})$, $a(x_1)$, $b(c_2, x_1)$, d, $b(c_3, x_4)$, $a(x_4)$, $a(\overline{x_4})$.

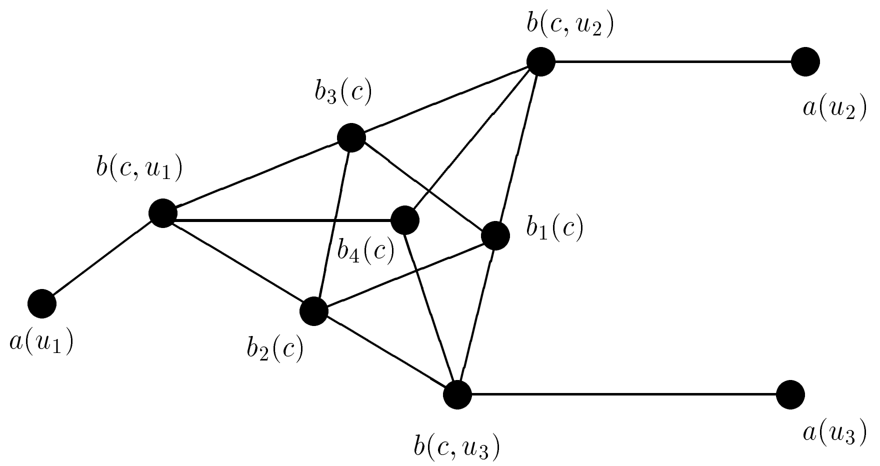

Figure 2: The seven vertex gadget for a clause in graph G_1.

Lemma 3.1 *The graph G_1 is P_8-free.*

Proof. Suppose that to the contrary G_1 would contain an induced path P on eight vertices. Denote by A the set of all vertices $a(u_i)$ on P, denote by B_1 the set of vertices $b_h(c)$ on P, and denote by B_2 the set of vertices $b(c, u_j)$ on P. We start with three observations.

(1) First, suppose that P contains the dummy vertex d. Since d is adjacent to all vertices in B_1 and in B_2, this yields $|B_1 \cup B_2| \leq 2$ and hence $|A| \geq 5$. Since $\{d\} \cup B_1 \cup B_2$ is a connected set, removing it from P decomposes this (induced) path into at most two (induced) subpaths that both are spanned by A. Since $|A| \geq 5$, one of these two subpaths must contain at least three vertices. But the longest induced paths in A have only two vertices. This contradiction yields $d \notin P$.

(2) Next suppose that $A = \emptyset$. Then P only contains vertices from B_1 and B_2. But the longest induced path in $B_1 \cup B_2$ has only four vertices; cf. Figure 2. This contradiction shows $|A| \geq 1$.

(3) Next suppose that $|A| \geq 3$. Since all vertices in B_1 are adjacent to all vertices in A, this would imply $B_1 = \emptyset$. Then P only contains vertices from A and B_2. But the longest induced path in $A \cup B_2$ has only four vertices, and this shows $|A| \leq 2$.

The observations in (2) and (3) yield $1 \leq |A| \leq 2$. Since all vertices in B_1 are adjacent to all vertices in A, $|A| = 1$ implies $|B_1| \leq 2$, and $|A| = 2$ implies $|B_1| \leq 1$. Therefore $|A \cup B_1| \leq 3$ and $|B_2| \geq 5$. But B_2 is an independent set of at least five vertices and cannot be connected to an induced path by adding the at most three vertices from $A \cup B_1$. ■

Lemma 3.2 *If the THREE-SATISFIABILITY instance I has a satisfying truth assignment, then the graph G_1 is 5-colorable.*

Proof. We define a coloring from the truth assignment. If $x \in X$ is TRUE then color $a(x)$ by 4 and $a(\overline{x})$ by 5, and if x is FALSE then color $a(x)$ by 5, and $a(\overline{x})$ by 4. The dummy vertex d receives color 4. Now consider the seven vertices $b_1(c)$, $b_2(c)$, $b_3(c)$, $b_4(c)$ and $b(c, u_1)$, $b(c, u_2)$, $b(c, u_3)$, that correspond to a clause $c = u_1 \vee u_2 \vee u_3$ in C. Without loss of generality we may assume that u_1 is a true literal, and that hence $a(u_1)$ is colored 4. In this case we color $b(c, u_1)$ by 5, $b(c, u_2)$ by 2, $b(c, u_3)$ by 3, and $b_1(c)$ by 1, $b_2(c)$ by 2, $b_3(c)$ by 3, $b_4(c)$ by 1. This coloring is legal: the edges among the seven clause vertices are verified easily, and besides them $b(c, u_1)$ is adjacent only to vertices already colored by 4 and the other six vertices are adjacent only to vertices already colored by 4 and 5. The cases where u_2 or u_3 are true literals are handled analogously. ■

Lemma 3.3 *If the graph G_1 is 5-colorable, then the THREE-SATISFIABILITY instance I has a satisfying truth assignment.*

Proof. Consider an arbitrary triangle $b_1(c)$, $b_2(c)$, $b_3(c)$ in a clause gadget. Without loss of generality, these three vertices are colored by colors 1, 2, and 3. Since the triangle vertices are adjacent to the dummy vertex d and to all literal vertices $a(x)$ and $a(\overline{x})$, all these adjacent vertices must be colored by 4 or by 5. Without loss of generality assume that the dummy vertex has color 4. Furthermore, if $a(x)$ has color 4, then $a(\overline{x})$ has color 5, and if $a(x)$ has color 5, then $a(\overline{x})$ has color 4. Define a truth assignment for X that sets variable x to TRUE if and only if $a(x)$ has color 4.

Suppose that some clause $c = u_1 \vee u_2 \vee u_3$ in C is not satisfied under this truth assignment. Then the three vertices $a(u_1)$, $a(u_2)$, $a(u_3)$ all are colored 5. Then the three clause vertices $b(c, u_1)$, $b(c, u_2)$, $b(c, u_3)$ have a neighbor of color 5, and the dummy vertex as neighbor of color 4, and hence they must be colored by colors 1, 2, 3. The four clause vertices $b_1(c)$, $b_2(c)$, $b_3(c)$, $b_4(c)$ are adjacent to $a(u_1)$ and $a(\overline{u_1})$ of colors 4 and 5, and hence they must be colored by colors 1, 2, 3. This implies that the seven vertices of the clause gadget are legally colored by the three colors 1, 2, 3, which clearly is impossible. This contradicts our assumption that the coloring is legal, and therefore the constructed truth assignment indeed satisfies all clauses. ■

The three Lemmas 3.1, 3.2, and 3.3 together prove Theorem 1.2(a).

4 The second NP-hardness proof

In this section we prove Theorem 1.2(b). Again, the reduction is from the NP-hard THREE-SATISFIABILITY problem; cf. the first paragraph of the preceding section. Consider an arbitrary instance I of THREE-SATISFIABILITY that consists of a set $X = \{x_1, \ldots, x_n\}$ of logical variables, and a set $C = \{c_1, \ldots, c_m\}$ of three-literal clauses over X. We will define a P_{12}-free graph $G_2 = (V_2, E_2)$ that is 4-colorable if and only if this instance I of THREE-SATISFIABILITY has answer YES.

- For every variable $x \in X$, there is a vertex $a(x)$ that corresponds to the unnegated literal x, and a vertex $a(\overline{x})$ that corresponds to the negated literal $\overline{x}$. These vertices are connected to each other by an edge.
- For every clause $c \in C$ that consists of the literals u_1, u_2, u_3, there are nine corresponding vertices $b_1(c, u_1)$, $b_1(c, u_2)$, $b_1(c, u_3)$, and $b_2(c, u_1)$, $b_2(c, u_2)$, $b_2(c, u_3)$, and $b_3(c, u_1)$, $b_3(c, u_2)$, $b_3(c, u_3)$. The three vertices $b_1(c, u_1)$, $b_1(c, u_2)$, $b_1(c, u_3)$ form a triangle. Moreover, for $i = 1, 2, 3$ the vertex $b_1(c, u_i)$ is connected to $b_2(c, u_i)$, the vertex $b_2(c, u_i)$ is connected to $b_3(c, u_i)$, and the vertex $b_3(c, u_i)$ is connected to $a(u_i)$. See Figure 3 for an illustration.
- All vertices $b_2(c, u_i)$ are connected to all vertices $a(x)$ and $a(\overline{x})$.
- Finally, there are two dummy vertices d_1 and d_2 that are connected to each other by an edge. The dummy vertex d_1 is connected to all vertices $b_j(c, u_i)$ that belong to the clause gadgets. The dummy vertex d_2 is connected to all vertices $a(x)$ and $a(\overline{x})$ with $x \in X$, and to all vertices $b_3(c, u_i)$.

This completes the description of the graph G_2. Note that G_2 may contain an induced P_{11} that runs through $b_2(c, u_4)$, $b_1(c, u_4)$, $b_1(c, u_5)$, $b_2(c, u_5)$, $b_3(c, u_5)$, d_2, $b_3(c', u_6)$, $b_2(c', u_6)$, $b_1(c', u_6)$, $b_1(c', u_7)$, $b_2(c', u_7)$.

Lemma 4.1 *The graph G_2 is P_{12}-free.*

Proof. Suppose that to the contrary G_2 would contain an induced path P on twelve vertices. Denote by A the set of all vertices $a(u_i)$ on P, denote by D the set of dummy vertices on P, and for $h = 1, 2, 3$ denote by B_h the set of all vertices $b_h(c, u_i)$ on P. We start with four simple observations.

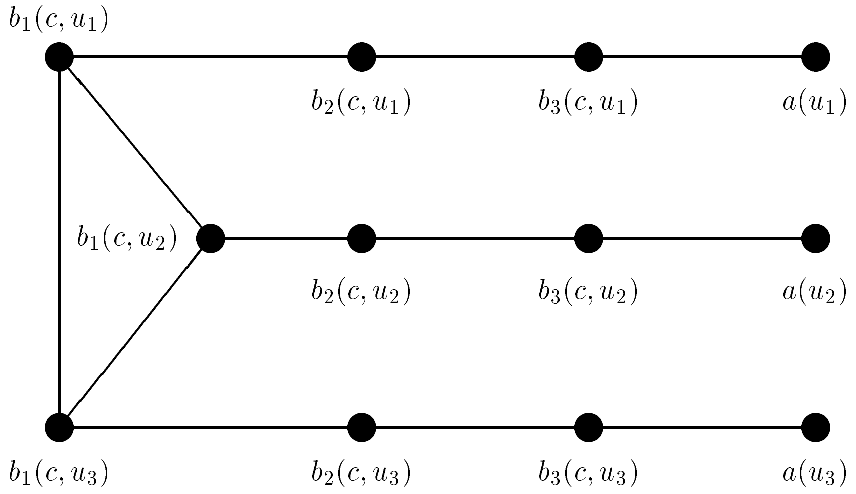

Figure 3: The nine vertex gadget for a clause in graph G_2.

(1) First, suppose that $|A| \geq 3$. Since all vertices in B_2 are adjacent to all vertices in A, this would imply $B_2 = \emptyset$. Moreover, $d_2 \notin D$. Then the only connection from B_1 to the rest of P is via the dummy vertex d_1, and therefore the vertices in $B_3 \cup A$ must induce a path. But there clearly cannot be an induced path in $B_3 \cup A$ that contains three or more vertices from A. This contradiction shows $|A| \leq 2$.

(2) Next suppose that $d_1 \in D$. Since d_1 is adjacent to all vertices in $B_1 \cup B_2 \cup B_3$, this union $B_1 \cup B_2 \cup B_3$ contains at most two elements. But then $|D \cup A| \geq 10$ which contradicts (1). Hence, $d_1 \notin D$.

(3) Suppose that $d_2 \in D$. Since d_2 is adjacent to all vertices in $B_3 \cup A$, we have $|B_3 \cup A| \leq 2$ and $|B_1 \cup B_2| \geq 9$. Every induced path in $B_1 \cup B_2$ has at most four vertices, and thus $B_1 \cup B_2$ must induce at least three subpaths of P. Each of these subpaths needs one adjacent vertex in B_3, which contradicts $|B_3| \leq |B_3 \cup A| \leq 2$. Hence, $d_2 \notin D$.

(4) Next suppose that $A = \emptyset$. Then P only contains vertices from B_1, B_2, and B_3. But the longest induced path in $B_1 \cup B_2 \cup B_3$ has only six vertices; cf. Figure 3. Therefore, $|A| \geq 1$.

The observations in (1) and (4) yield $1 \leq |A| \leq 2$. Since all vertices in B_2 are adjacent to all vertices in A, $|A| = 1$ implies $|B_2| \leq 2$, and $|A| = 2$ implies $|B_2| \leq 1$. Therefore $|A \cup B_2| \leq 3$, and $|B_1 \cup B_3| \geq 9$. But the longest induced paths in $B_1 \cup B_3$ have only two vertices, and thus $B_1 \cup B_3$ induce at least five connected components. There is no way of glueing these at least five components together via the at most three vertices in $A \cup B_2$. ■

Lemma 4.2 *If the THREE-SATISFIABILITY instance I has a satisfying truth assignment, then the graph G_2 is 4-colorable.*

Proof. We define a coloring from the truth assignment. Dummy vertex d_1 is colored 1, and dummy vertex d_2 is colored 2. All vertices $a(u_i)$ will be colored 1 or 4; all vertices $b_3(c,u_i)$ will be colored 3 or 4; all vertices $b_2(c, u_i)$ will be colored 2 or 3; all vertices $b_1(c, u_i)$ will be colored 2, 3, or 4. Clearly, by doing so we will avoid all color conflicts with the two dummy vertices. Moreover, vertices $b_2(c, u_i)$ will receive other colors than vertices $a(x)$ and $a(\overline{x})$.

If $x \in X$ is TRUE then color $a(x)$ by 1 and $a(\overline{x})$ by 4, and if x is FALSE then color $a(x)$ by 4, and $a(\overline{x})$ by 1. Now consider the vertices of some clause gadget for $c = u_1 \vee u_2 \vee u_3$ in C. Without loss of generality we assume that the literal u_1 is true, and that hence $a(u_1)$ is colored 1. Then we color

$b_1(c, u_1)$ by 2	$b_2(c, u_1)$ by 3	$b_3(c, u_1)$ by 4	$a(u_1)$ is 1
$b_1(c, u_2)$ by 3	$b_2(c, u_2)$ by 2	$b_3(c, u_2)$ by 3	$a(u_2)$ is 1 or 4
$b_1(c, u_3)$ by 4	$b_2(c, u_3)$ by 2	$b_3(c, u_3)$ by 3	$a(u_3)$ is 1 or 4

It is easy to verify that we indeed end up with a legal 4-coloring for the graph G_2. ■

Lemma 4.3 *If the graph G_2 is 4-colorable, then the THREE-SATISFIABILITY instance I has a satisfying truth assignment.*

Proof. Without loss of generality we assume that in the 4-coloring dummy vertex d_1 is colored 1 and that dummy vertex d_2 is colored 2. The set of literal vertices $a(x)$ and $a(\overline{x})$ uses at least two different colors. We claim that also the set of clause vertices $b_2(c, u_i)$ uses at least two different colors: Suppose that to the contrary all vertices $b_2(c, u_i)$ are colored by a single color, say, by color 3. Then the triangles $b_1(c, u_1)$, $b_1(c, u_2)$, $b_1(c, u_3)$ cannot use this color 3, nor can they use the color 1 of dummy vertex d_1. But it is impossible to color the triangle legally by colors 2 and 4 only, which proves our claim. Now since the literal vertices use at least two colors, since the vertices $b_2(c, u_i)$ use at least two colors, and since these two vertex classes form a complete bipartite graph, one of these classes must use exactly two colors, and the other class must use the remaining two colors. Without loss of generality we assume that the literal vertices use the colors 1 and 4, and that the vertices $b_2(c, u_i)$ use the colors 2 and 3.

We define a truth assignment from the 4-coloring by setting variable x to TRUE if and only if vertex $a(x)$ has color 1. Suppose that some clause $c = u_1 \vee u_2 \vee u_3$ in C is not satisfied under this truth assignment. Then the three vertices $a(u_1)$, $a(u_2)$, $a(u_3)$ all are colored 4. The three adjacent vertices $b_3(c, u_1)$, $b_3(c, u_2)$, $b_3(c, u_3)$ cannot use color 4, and they also cannot use colors 1 or 2 since they are adjacent to both dummy vertices; therefore, $b_3(c, u_1)$, $b_3(c, u_2)$, $b_3(c, u_3)$ all are colored 3. Then the three adjacent vertices $b_2(c, u_1)$, $b_2(c, u_2)$, $b_2(c, u_3)$ cannot use this color 3; by the above discussion all three vertices $b_2(c, u_1)$, $b_2(c, u_2)$, $b_2(c, u_3)$ must be colored 2. Then the three adjacent vertices $b_1(c, u_1)$, $b_1(c, u_2)$, $b_1(c, u_3)$ cannot use this color 2, and they also cannot use the color 1 of the adjacent dummy vertex d_1. This implies that the triangle $b_1(c, u_1)$, $b_1(c, u_2)$, $b_1(c, u_3)$ is legally colored by the two colors 3 and 4, which is impossible. Consequently, the constructed truth assignment satisfies all clauses in C. ■

The three Lemmas 4.1, 4.2, and 4.3 together prove Theorem 1.2(b).

References

[1] M.R. Garey and D.S. Johnson [1979]. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, San Francisco.

[2] M.C. Golumbic [1980]. *Algorithmic Graph Thery and Perfect Graphs*. Academic Press, New York.

[3] D.S. Johnson [1985]. The NP-completeness column: An ongoing guide. *Journal of Algorithms 6*, 434–451.