SIAM J. DISCRETE MATH.
Vol. 18, No. 1, pp. 30–40

ON THE HARDNESS OF 4-COLORING A 3-COLORABLE GRAPH*

VENKATESAN GURUSWAMI† AND SANJEEV KHANNA‡

Abstract. We give a new proof showing that it is NP-hard to color a 3-colorable graph using just 4 colors. This result is already known [S. Khanna, N. Linial, and S. Safra, *Combinatorica*, 20 (2000), pp. 393–415], but our proof is novel because it does not rely on the PCP theorem, while the known one does. This highlights a qualitative difference between the known hardness result for coloring 3-colorable graphs and the factor n^ϵ hardness for approximating the chromatic number of general graphs, as the latter result is known to imply (some form of) PCP theorem [M. Bellare, O. Goldreich, and M. Sudan, *SIAM J. Comput.*, 27 (1998), pp. 805–915].

Another aspect in which our proof is novel is in its use of the PCP theorem to show that 4-coloring of 3-colorable graphs remains NP-hard even on bounded-degree graphs (this hardness result does not seem to follow from the earlier reduction of Khanna, Linial, and Safra). We point out that such graphs can always be colored using $O(1)$ colors by a simple greedy algorithm, while the best known algorithm for coloring (general) 3-colorable graphs requires $n^{\Omega(1)}$ colors. Our proof technique also shows that there is an $\varepsilon_0 > 0$ such that it is NP-hard to legally 4-color even a $(1-\varepsilon_0)$ fraction of the edges of a 3-colorable graph.

Key words. graph coloring, PCP theorem, NP-hardness, hardness of approximation

AMS subject classifications. 68Q17, 05C15, 68R10

DOI. 10.1137/S0895480100376794

1. Introduction.

The graph coloring problem is to assign colors to vertices of a graph G such that no two adjacent vertices receive the same color; such a coloring is referred to as a *legal* coloring of G. The minimum number of colors required to perform a legal coloring is known as the *chromatic number* of G and is denoted $\chi(G)$. Graph coloring is a fundamental and extensively studied problem. In addition to its theoretical significance as a canonical NP-hard problem [17], it also arises naturally in a variety of applications including register allocation and timetable/examination scheduling.

Since coloring a graph G with the minimum number $\chi(G)$ of colors is NP-hard [17], we shift our focus to efficiently coloring a graph with an approximately optimum number of colors. Garey and Johnson [10] proved that it is NP-hard to approximate the chromatic number within a factor of $(2-\epsilon)$ for any $\epsilon > 0$. The best known algorithm for general graphs appears in [14] and colors a graph using a number of colors that is within a factor of $O(n(\log\log n)^2/\log^3 n)$ of the optimum (here and elsewhere, n refers to the number of vertices in the graph). There is strong evidence that one cannot do substantially better than this for general graphs, as the recent connection between probabilistically checkable proofs (PCPs) and hardness of approximations [7, 2, 1]

*Received by the editors August 8, 2000; accepted for publication (in revised form) October 28, 2003; published electronically July 2, 2004. A preliminary version of this paper appeared in the *Proceedings of the* 15*th Annual IEEE Conference on Computational Complexity*, Florence, Italy, 2000, pp. 188–197.

http://www.siam.org/journals/sidma/18-1/37679.html

†Department of Computer Science, University of Washington, Seattle, WA 98195 (venkat@cs.washington.edu). This work was done while the author was at the Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, and while he was visiting Bell Laboratories, Murray Hill, NJ.

‡Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104 (sanjeev@cis.upenn.edu). This work was done while the author was at Bell Laboratories, Murray Hill, NJ.

has led to strong hardness results for graph coloring also. The first such result was established by Lund and Yannakakis [20], who proved that chromatic number is hard to approximate within n^ϵ for some constant $\epsilon > 0$. Feige and Kilian [8], using the powerful PCP constructions due to Håstad [15], prove that, unless NP $\subseteq$ ZPP, one cannot approximate the chromatic number within a factor of $n^{1-\epsilon}$ for any constant $\epsilon > 0$.

However, *none* of these inapproximability results apply to the case when the input graph is k-colorable for some small constant k. Indeed, better performance guarantees are known in this case. For instance, a polynomial time algorithm that colors 3-colorable graphs using $\tilde{O}(n^{3/14})$ colors is known [23, 5, 16, 6]. It is known that for every constant h, there exists a large enough constant k such that coloring k-colorable graphs using kh colors is NP-hard [20, 19]; it is, however, not known if the order of quantifiers above can be reversed. Khanna, Linial, and Safra [19] proved that it is NP-hard to color a 3-colorable graph using only 4 colors, and to this date no improvement to this hardness result has been obtained.

Our results. Our main result in this paper is a new proof of the above result of [19]. Our result is stated formally below.

THEOREM 1.1 (main theorem). *It is NP-hard to color a* 3*-colorable graph with only* 4 *colors.*

The proof of Khanna, Linial, and Safra [19] uses the result that MAX CLIQUE is NP-hard to approximate within a factor of two, a consequence of the PCP theorem [2, 1]. An important distinguishing aspect of our proof is that it *does not require the PCP theorem* and only relies on the NP-hardness of the MAX CLIQUE problem. The hardness for 3-colorable graphs is the most intricate of the results in [19] and has not been improved upon or simplified ever since. Our work represents the most progress made on this important problem after the result of [19], and we hope our work will spur further improvements. Not relying on PCP machinery implies that this hardness result could have been obtained almost three decades ago, long before the arrival of the PCP theorem. In contrast, the hardness result (for approximating within n^ϵ, for example) for general graph coloring implies some form of PCP [3]; our result therefore also highlights a qualitative difference between the hardness of general graph coloring and the hardness of coloring 3-colorable graphs.[1]

As in essentially all previous reductions showing hardness of graph coloring, our reduction too starts from the hardness of INDEPENDENT SET (MAX CLIQUE): it transforms an instance G of INDEPENDENT SET to an instance H of graph coloring such that a large independent set in G translates into a small collection of (in our case, three) independent sets in H, which together cover all vertices in H. But in addition, our proof is based only on local gadgets and easily leads to the hardness of 4-coloring even bounded-degree instances of 3-colorable graphs, albeit only by resorting to the PCP theorem.

THEOREM 1.2. *There is a constant* Δ *such that given a* 3*-colorable graph with maximum degree at most* Δ*, it is NP-hard to color it using just* 4 *colors.*

Note that, since such graphs can be colored using $O(1)$ colors (in fact, $(\Delta + 1)$ colors) by a simple greedy algorithm, while the best algorithm for general 3-colorable

[1] From a strictly logical point of view, the PCP theorem is true, so every result implies it, including our hardness result for 4-coloring 3-colorable graphs. When we say a hardness of approximation result implies a nontrivial PCP, we mean that one can get such a PCP result, via a *simple* reduction from the inapproximability result, without going through the steps of the current complicated proof of the PCP theorem. We hope this does not cause any confusion for the reader.

graphs uses $n^{\Omega(1)}$ colors, this hardness result is stronger than that of Theorem 1.1. Another strengthening of Theorem 1.1, which the degree-bounded result enables us to deduce, is the following.

THEOREM 1.3. *There is a constant $\varepsilon_0 > 0$ such that it is NP-hard, given a graph G, to distinguish between the case when G is 3-colorable and when any 4-coloring of G miscolors at least an ε_0 fraction of its edges.*

Both of these results do not seem to follow from the proof technique of [19] and therefore appear to be new. Note that the latter claim also generalizes the result of Petrank [22] which shows that there is an $\varepsilon > 0$ such that it is NP-hard to legally color a $(1-\varepsilon)$ fraction of the edges of a 3-colorable graph using only 3 colors.

Inapproximability results and PCPs. In light of our main result, it is natural to ask how far nonPCP techniques can go in proving hardness results for coloring 3-colorable graphs. It turns out that an inapproximability factor of $\Omega(\log n)$ does imply a nontrivial PCP verifier for languages in NP. This follows from a result of Blum [4] (see also [3]), which shows that if coloring a 3-colorable graph using $c \log n$ colors is hard for every constant c, then for every $\epsilon > 0$, it is hard to approximate MAX CLIQUE within a factor of $n^{1-\epsilon}$; using the "reversal" of the FGLSS connection presented in [3], this implies the PCP theorem (in fact a very strong version of it; see [3] for details).[2] However, it seems possible that any $o(\log n)$ hardness bound can be proved for coloring 3-colorable graphs without resorting to PCP techniques.[3]

Expanding the scope of our investigation, it is natural to ask which inapproximability results really require PCPs. It is known, for example, that PCPs are inherent in obtaining strong hardness results for approximating MAX SAT, MAX CLIQUE, chromatic number, and vertex cover. Concretely, inapproximability results for these problems imply, via a simple reduction, a nontrivial PCP verifier for NP languages. Recent work in [12] and [18] proves strong (in fact near-tight) inapproximability results for disjoint paths and longest path problems without requiring PCPs; prior results for these problems always began with the PCP theorem and yet turned out to be weaker. Together with our result, these raise similar questions about the hardness results for certain other fundamental problems like set cover, nearest codeword problem, shortest vector problem, etc. In each of these cases it is interesting to see if a reverse connection to PCPs exists or if PCPs are only an artifact of the current proof techniques.

Notation. We use the standard notation to denote graph-theoretic parameters. For a graph G, we denote by $\chi(G)$, $\alpha(G)$, $\omega(G)$, and $\theta(G)$ the chromatic number of G, the size of a largest independent set in G, the size of a largest clique in G, and the clique cover number of G (the minimum number of cliques to cover all the vertices of G), respectively. Clearly $\alpha(G) = \omega(\bar{G})$ and $\chi(G) = \theta(\bar{G})$, where $\bar{G}$ is the complement of the graph G.

Organization. We present the proof of our main theorem (Theorem 1.1) in section 2. Section 3 describes the hardness result for bounded-degree 3-colorable graphs and sketches the proof of Theorem 1.3.

[2] Since the reduction from 3-coloring to finding large cliques is only a Turing reduction, strictly speaking, we can only conclude that every language in NP Turing reduces to a language in a certain PCP class.

[3] Actually, such a hardness result *does* imply the existence of very good *covering PCPs*, a notion recently introduced in [13] for the purpose of studying minimization problems like coloring. Constructing a "good" covering PCP without resorting to the PCP theorem appears very difficult, so such a PCP-free hardness result for coloring 3-colorable graphs might be hard to come by.

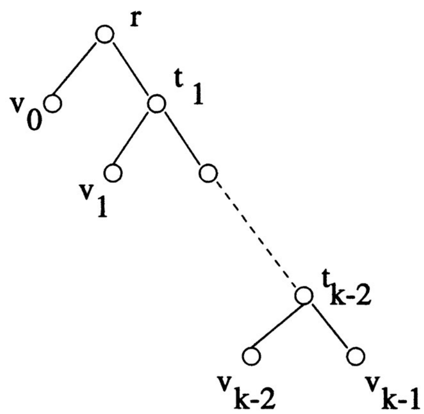


FIG. 1. *High-level structure of each T_i.*

2. Proof of the main theorem. We describe a reduction from the INDEPENDENT SET problem. Specifically, we start with instances of the following form: We are given a graph G along with a partition of the vertices of G into r cliques R_i, $1 \leq i \leq r$, each with exactly k vertices. Clearly, $\alpha(G) \leq r$. It is NP-hard to determine if $\alpha(G) = r$ on instances with this structure even when the partition into the R_i's is given as part of the input. This hardness even holds with $k = 3$—the standard reduction for NP-hardness of INDEPENDENT SET in fact produces such instances [11]. Thus the proof of Theorem 1.1 only requires us to consider the case $k = 3$. However, we will present here a construction for any arbitrary k. This is because the starting point for Theorems 1.2 and 1.3 are INDEPENDENT SET instances that are generated by PCP constructions, and k is a suitably large constant in this case.

Starting with such an instance G, we construct (in polynomial time) a graph H which will have the property that $\chi(H) = 3$ if $\alpha(G) = r$ and $\chi(H) \geq 5$ otherwise. This will clearly prove Theorem 1.1.

2.1. Overview of the reduction. Let G be a graph with vertices partitioned into r cliques R_i, $1 \leq i \leq r$, with exactly k vertices in each clique; i.e., let $R_i = \{v_{i,0}, \ldots, v_{i,k-1}\}$ for $1 \leq i \leq r$. The graph H is comprised of r "tree-like" structures, say $T_1, \ldots, T_r$, one for each clique R_i of G, together with a specific interconnection pattern between the leaves of the different tree structures based on the adjacency of vertices in G. The following are two key properties satisfied by the construction of the T_i's:

- Any 4-coloring of a T_i can be interpreted as "selecting" a unique vertex $v_{i,p}$ in the clique R_i of graph G (section 2.2).
- The edges between the T_i's are such that no 4-coloring is feasible if 2 vertices that are adjacent in G are selected from 2 different trees (section 2.3).

In other words, any 4-coloring of H can be interpreted as selecting a vertex in each of the r cliques R_i of G such that the selected vertices induce an independent set of size r in G, ensuring that if $\alpha(G) < r$, then in fact $\chi(H) > 4$. The other part, namely, that H is 3-colorable if $\alpha(G) = r$, will also be easily seen to hold for our reduction.

2.2. The structure of each T_i. Each T_i will have the structure of a binary tree with k leaves, $\{v_{i,j} : 0 \leq j < k\}$, one for each of the k vertices of G in the clique R_i (see Figure 1). It also has $(k-1)$ additional internal nodes $\{t_{i,j} : 0 \leq j < k-1\}$

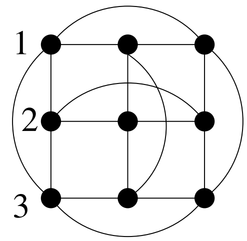


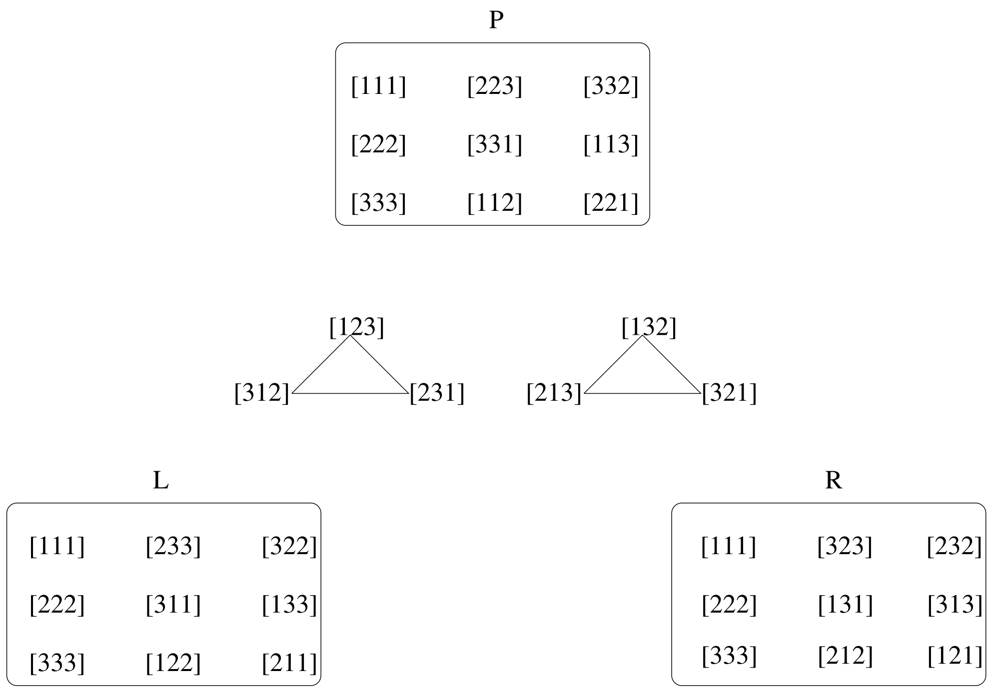
FIG. 2. *The basic template.*

with $t_{i,0}$ being the "root" r_i; by $t_{i,k-1}$ we mean the leaf node $v_{i,k-1}$. (The subscript i is omitted in Figure 1 for the sake of readability. The exact "shape" of the tree T_i is not important; any binary tree with k leaves and with all internal nodes having exactly two children will suffice for our purposes.) Each individual node of T_i is made up of the template shown in Figure 2. This basic template, denoted H_{basic}, may be viewed as a 3×3 grid such that the vertices in each row and in each column of the grid induce a 3-clique. The vertices in the first column of any such template are referred to as *ground vertices* and are in fact *shared* across all such templates in all the tree-structures. Since the ground vertices form a clique, any legal coloring will assign 3 *distinct* colors to them; we refer to these colors as 1, 2, and 3.

The connection pattern between the template at an internal node $t_{i,p}$ and its children, templates $v_{i,p}$ and $t_{i,p+1}$, is best understood by the schematic depicted in Figure 3. (Nodes P, L, and R will play the roles of $t_{i,p}$, $v_{i,p}$, and $t_{i,p+1}$, respectively.) In addition to the templates at these nodes, there are two 3-cliques that are connected to templates at $t_{i,p}, v_{i,p}$, and $t_{i,p+1}$ via appropriate edges. All nodes in the schematic are labeled as 3-tuples of the form $\langle xyz \rangle$, where $x, y, z \in \{1, 2, 3\}$. The edges (not shown) between the various vertices are given by the following simple rule: Two vertices are adjacent if and only if their labels *differ in all three coordinates.*

2.2.1. Node selection. A node of the tree is called *selected* if at least one of the three rows in its template has colors which, reading from left to right, form an even permutation of $\{1, 2, 3\}$ (i.e., the first row has colors 1, 2, 3; the second has 2, 3, 1; or the third has 3, 1, 2). Similarly, we say that a node is *not selected* if at least one of the three rows in its template has colors which, reading from left to right, form an odd permutation. It is easy to see that, in any legal 4-coloring, a node can never be simultaneously selected and not selected. Moreover, in any 4-coloring a node is always either selected or not selected.

2.2.2. Enforcing selection of a leaf node. Our goal now is to enforce that, for any legal 4-coloring of the tree-structure T_i, at least one leaf node is selected. Broadly speaking, our approach here will be to "hardwire" the selection of the root node and then introduce gadgets to ensure that, whenever a node is selected, one of its two children is selected as well. In other words, our construction propagates selection from the root to some leaf node. While one can imagine, at least for the case $k = 3$, that one can construct a "direct" 1-*out-of*-3 gadget, which will ensure that 1 of 3 nodes is always selected, this "top-down" approach works for any value of k and is also more modular and easier to present.



Two nodes are adjacent iff their labels differ in every coordinate.

FIG. 3. *The connection pattern between the templates at a node and its children.*

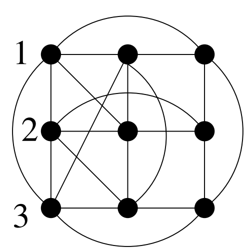


FIG. 4. *Enforcing selection at a root.*

Root selection. In each tree T_i, $1 \leq i \leq r$, we enforce selection of the root using the gadget shown in Figure 4. It is obtained by adding, for each $j \in \{1, 2, 3\}$, edges from the ground vertex colored j to the first vertex in row ($j \bmod 3 + 1$) of the copy of H_{basic} at the root node r_i of T_i. This ensures that, in any 4-coloring of H, there will be 1 row of (each) root which will be selected (and hence the root itself will be selected). Indeed, there must exist 1 row whose vertices are not colored using 4, say, for concreteness, the third row. But since we added an edge between the ground vertex colored 2 and the first vertex in the third row, this vertex cannot be colored 2, and it follows that the third row of the root must be colored $(3, 1, 2)$, as desired.

Propagating the selection. Next, we show how the selection of a node in the tree can be propagated to at least one of the node's children. This ensures that, in each tree, at least one leaf node must be selected. Consider again the schematic in Figure 3 and assign the following interpretation to the node labels:

- Colors in the first coordinate of each node correspond to the situation in which $t_{i,j}$ is selected and these colors correspond to selection at $v_{i,j}$.
- Colors in the second coordinate of each node correspond to the situation in which $t_{i,j}$ is selected and these colors correspond to selection at $t_{i,j+1}$.
- Colors in the third coordinate of each node correspond to the situation in which $t_{i,j}$ is not selected and these colors correspond to both $v_{i,j}$ and $t_{i,j+1}$ not being selected.

It is tedious but straightforward to verify that, for any $l \in \{1, 2, 3\}$, if we assign colors 1, 2, and 3 to the nodes as specified by their lth coordinate, then a feasible coloring is formed. Moreover, for any choice of a leaf node to be selected in T_i, coloring the nodes along the unique root-leaf path as selected (i.e., coloring the 3 rows of the corresponding templates as $\{1, 2, 3\}$, $\{2, 3, 1\}$, and $\{3, 1, 2\}$), and the remaining nodes in T_i as not selected (i.e., coloring the 3 rows of the corresponding templates as $\{1, 3, 2\}$, $\{2, 1, 3\}$, and $\{3, 2, 1\}$), yields a legal 3-coloring of T_i. The following is thus evident for our construction.

LEMMA 2.1. *For each i, $1 \le i \le r$, and for all j, $0 \le j < k$, there is a 3-coloring of the vertices in the tree-structure T_i such that the leaf corresponding to $v_{i,j}$ is the only selected leaf in T_i.*

We can now establish the following key lemma.

LEMMA 2.2. *In any 4-coloring of a tree T_i, whenever an internal node is selected, one of its 2 children must be selected.*

Proof. Consider again the schematic of Figure 3, with P being the parent whose selection we argue implies the selection of one of its children L and R. We consider the following two cases.

Case 1. Both vertices in 1 of the pairs $\{\langle 112\rangle, \langle 113\rangle\}$, $\{\langle 221\rangle, \langle 223\rangle\}$, and $\{\langle 331\rangle, \langle 332\rangle\}$ receive color 4 in the 4-coloring of H.

Suppose it is the pair $\{\langle 331\rangle, \langle 332\rangle\}$ that receives color 4. Since P is selected, the third row of P must be colored $(3, 1, 2)$ in this case. We now claim that 1 of L and R will in fact be selected with their third row being colored $(3, 1, 2)$. Indeed, none of the vertices $\langle 122\rangle$, $\langle 211\rangle$, $\langle 212\rangle$, and $\langle 121\rangle$ (which are the third row nonground vertices of L and R) receive the color 4 as they are all adjacent to $\langle 331\rangle$ or $\langle 332\rangle$. Thus if neither L nor R is selected, $\langle 122\rangle, \langle 212\rangle$ get colored 2 and $\langle 211\rangle, \langle 121\rangle$ get colored 1. Now it is easy to see that each of the vertices $\langle 123\rangle$, $\langle 231\rangle$, and $\langle 312\rangle$ has color 1 as well as color 2 neighbors. Specifically, $\langle 123\rangle$ is adjacent to $\langle 211\rangle$ and $\langle 212\rangle$, $\langle 231\rangle$ is adjacent to $\langle 112\rangle$ and $\langle 122\rangle$, and $\langle 312\rangle$ is adjacent to $\langle 121\rangle$ and $\langle 221\rangle$ (recall that $\langle 112\rangle$ is colored 1 and $\langle 221\rangle$ is colored 2 since the third row of P is colored $(3, 1, 2)$). Thus, all 3 vertices must be colored either 3 or 4. But this is impossible because these 3 vertices form a clique. Therefore, L or R must be selected.

Similar arguments will hold if both vertices $\langle 112\rangle, \langle 113\rangle$ receive color 4 or if both vertices $\langle 221\rangle, \langle 223\rangle$ receive color 4. So it remains to consider the following case.

Case 2. At most 1 of the vertices in each of the pairs $\{\langle 112\rangle, \langle 113\rangle\}$, $\{\langle 221\rangle, \langle 223\rangle\}$, and $\{\langle 331\rangle, \langle 332\rangle\}$ receives color 4 in the 4-coloring of H.

In this case we first claim the following.

CLAIM 1. *At least one of the vertices $\langle 112\rangle, \langle 113\rangle$ gets colored 1, one of $\langle 221\rangle, \langle 223\rangle$ gets colored 2, and one of $\langle 331\rangle, \langle 332\rangle$ gets colored 3.*

To see this, note that P is selected, so we may assume, without loss of generality, that the third row of P is colored $(3, 1, 2)$. Thus, the above claim is trivially verified for colors 1 and 2 (since $\langle 112 \rangle$ is colored 1 and $\langle 221 \rangle$ is colored 2). Now if neither $\langle 331 \rangle$ nor $\langle 332 \rangle$ is colored 3, then in fact both must be colored 4 (since, for instance, $\langle 332 \rangle$ cannot be colored either 1 or 2 because it is adjacent to both $\langle 111 \rangle$ and $\langle 221 \rangle$). But this contradicts the hypothesis of this case, and therefore our claim holds. ∎

We are now ready to finish the proof for Case 2. Suppose P is selected, but neither L nor R is selected. We will call a row of a node *pure* if none of its vertices are colored 4. Clearly, at least one of the rows of both L and R is pure. Since the entire gadget is totally symmetric, assume for definiteness that the third row of L is pure so that it is colored $(3, 2, 1)$ (recall that L is *not* selected, so it cannot be colored $(3, 1, 2)$). Now if the third row of R is pure, then it will also be colored $(3, 2, 1)$, and we will get a contradiction exactly as we obtained in the analysis of Case 1. So one of the first or second rows of R is pure; say, again without loss of generality, that the first row of R is pure so that it is colored $(1, 3, 2)$. The upshot of all this is that the vertices $\langle 122 \rangle, \langle 211 \rangle, \langle 323 \rangle, \langle 232 \rangle$ receive colors $2, 1, 3, 2$, respectively.

Now consider the vertex $\langle 231 \rangle$. It is adjacent (among other vertices) to $\langle 122 \rangle$ (which is colored 2), to $\langle 323 \rangle$ (which is colored 3), and to both $\langle 112 \rangle$ and $\langle 113 \rangle$, one of which is colored 1 by Claim 1. It follows therefore that $\langle 231 \rangle$ is colored 4. A similar argument shows that $\langle 123 \rangle$ must also be colored 4—indeed $\langle 123 \rangle$ is adjacent to $\langle 211 \rangle$ (colored 1), to $\langle 232 \rangle$ (colored 2), and to both $\langle 331 \rangle$ and $\langle 332 \rangle$, one of which is colored 3 by Claim 1. But now both $\langle 231 \rangle$ and $\langle 123 \rangle$ are colored 4 and are adjacent, which is a contradiction. This completes the analysis for Case 2, and the proof is now complete. ∎

2.3. The structure across the trees. We now specify how the nodes across different T_i's are connected. For every pair of leaf nodes $v_{i,p} \in T_i$ and $v_{j,q} \in T_j$ such that $v_{i,p}$ and $v_{j,q}$ are adjacent in G, we insert a gadget (actually a combination of more than one gadget) that prevents both of these leaf nodes from being selected simultaneously in any legal 4-coloring of H. Observe that this would immediately imply that if H is 4-colorable, then there must be an independent set of size at least r in G. This follows from Lemma 2.2 which shows that, in any 4-coloring of H, every tree has at least one selected leaf, and no two vertices of G corresponding to selected leaves can be adjacent in G.

The leaf-level gadget consists of two parts, as shown in Figures 5 and 6. Given two nodes, each a copy of the basic template H_{basic}, we use two kinds of gadgets. The first kind, shown in Figure 5, prevents both nodes from being selected because of the *same* row (for example, because the third row of both nodes is colored $(3, 1, 2)$)—we use three such gadgets, one for each row. It is easy to check that the gadget in Figure 5 is 3-colorable as long as at least one of the two third rows is colored $(3, 2, 1)$, but the gadget is not even 4-colorable if both the third rows are colored $(3, 1, 2)$.

The second kind of leaf-level gadget, shown in Figure 6, ensures that the two nodes are not both selected because of *different* rows, and this gadget is even simpler than the first. Once again it is completely straightforward to check that the gadget works as desired; for instance, for the gadget shown, there exists a valid 3-coloring as long as either the third row of the left-hand-side node is $(3, 2, 1)$ or the first row of the right-hand-side node is $(1, 3, 2)$ (i.e., at least one is not selected), but there is no valid 4-coloring if these rows are colored $(3, 1, 2)$ and $(1, 2, 3)$ (i.e., if both are selected).

The preceding discussion has thus established the following.

LEMMA 2.3. *If the graph H constructed as above is 4-colorable, then $\alpha(G) = r$.*

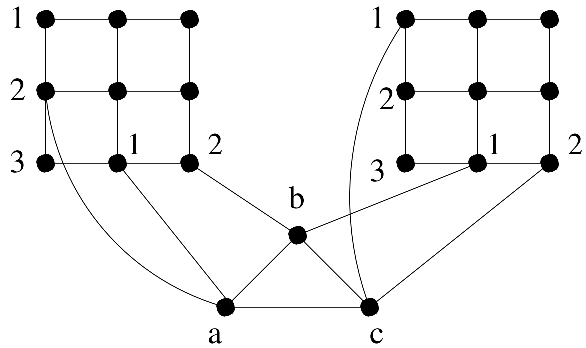


FIG. 5. *The leaf-level gadget: "Same row kind."*

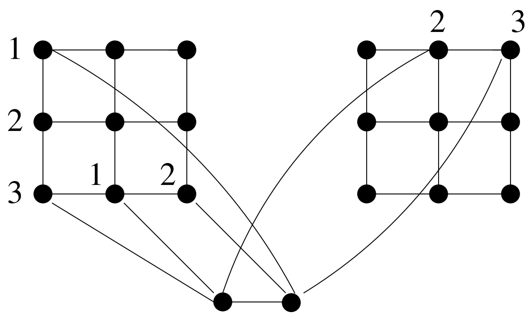


FIG. 6. *The leaf-level gadget: "Different rows kind."*

LEMMA 2.4. *If $\alpha(G) = r$, then H is 3-colorable.*

Proof. Let $K = \{v_{i,p_i} : 1 \leq i \leq r\}$ be an independent set of size r in G, where $0 \leq p_i < k$ for each i. By Lemma 2.1, we can legally color all the vertices of the tree-structures T_i using only three colors such that, for each tree T_i, the leaf corresponding to v_{i,p_i} is the only one that is selected. It remains only to color the vertices used in the leaf-level gadgets. By the argument above we can color the vertices of any leaf-level gadget using just three colors, provided at least one of the two leaf nodes it "connects" is not selected. But this condition is met for every leaf-level gadget in our case, since K is an independent set, and therefore there is no leaf-level gadget between any two of our selected leaf nodes. The entire graph H is thus 3-colorable. ∎

Theorem 1.1 now follows from Lemmas 2.4 and 2.3 since the construction of H can be clearly accomplished in polynomial time.

Tightness of our analysis. We point out here that the graph H constructed in the reduction above is *always* 5-colorable. Our analysis of the reduction is therefore tight in this regard. Indeed, by letting exactly one arbitrarily chosen leaf node of each tree-structure be selected, we can legally color all vertices in the tree-structures using three colors, say 1, 2, and 3. We claim it is now possible to legally color all the vertices in the leaf-level gadgets using only two more colors. Indeed, there are only two new nodes in the leaf-level gadgets of the "different rows kind" (Figure 6), and thus they can be colored 4, 5 arbitrarily. For the leaf-level gadgets of the "same row kind" (Figure 5), we need only worry about the situation where the two leaf nodes to which the gadget connects are *both* selected. This follows from our "completeness" analysis (Lemma 2.4), where we showed that the vertices in the leaf-level gadgets can be properly colored using just colors 1, 2, and 3 when at most one of the two leaf nodes are selected. The case when both leaf nodes are selected is exactly the situation depicted in Figure 5, and it is easily seen that in this case the three new vertices in the leaf-level gadget concerned can be properly colored using the colors 3, 4, and 5.

3. Hardness for degree-bounded 3-colorable graphs. We now show that the result of Theorem 1.1 holds even if the input graph G has degree bounded by some constant Δ, thus establishing Theorem 1.2. Unlike Theorem 1.1, however, we do not see how to prove the result below without using the PCP theorem. Specifically, we use Proposition 3.1 below, which follows from the PCP theorem and MAX SNP-hardness of MAX 3-SAT instances, where each variable appears in at most a constant number of, say five, clauses [21].

PROPOSITION 3.1. *For every constant $t > 1$ there exist constants q, Δ such that, given a graph G whose vertices can be partitioned into r cliques each containing exactly q vertices and in which each vertex has degree at most Δ, it is NP-hard to distinguish between the cases $\alpha(G) = r$ and $\alpha(G) < r/t$.*

Proof of Theorem 1.2. We employ (essentially) the same reduction as in the proof of Theorem 1.1, except that we now start from a hard instance of INDEPENDENT SET, as in Proposition 3.1, with a "gap" (in independent set size) of $t = 24$. The graph H thus constructed will satisfy $\chi(H) = 3$ if $\alpha(G) = r$, while $\chi(H) \geq 5$ if $\alpha(G) < r$. By the nature of the reduction presented in section 2, and the fact that the maximum degree of G is at most Δ, it is easy to see that all vertices in H have very small degree except the three *ground vertices*, which are shared across all the r tree-like structures in H (that correspond to the r cliques in G). We get around this by simply using a distinct set of three ground vertices in each of the r tree-structures to give a new degree-bounded graph H'. By a pigeonhole argument, since there are only 24 different colorings of a (labeled) 3-clique using 4 colors, there are at least $r/24$ of the tree-structures whose ground vertices in rows $1, 2, 3$ are colored using the same three colors c_1, c_2, c_3; we just label these colors as $1, 2, 3$, respectively. Now, applying the argument used in the proof of Lemma 2.3 to the subgraph of G induced by the vertices in the $r/24$ cliques corresponding to these tree-structures, we conclude that, if H' is 4-colorable, then $\alpha(G) \geq r/24$. Of course in the case when $\alpha(G) = r$, the same coloring used to establish Lemma 2.4 with all copies of the ground vertices being colored as $1, 2, 3$ properly implies that H' is 3-colorable. Combining this reduction with Proposition 3.1, therefore, gives us our claimed result. □

It turns out that the above argument also suffices to establish Theorem 1.3.

Proof of Theorem 1.3. Use the same reduction to get a graph H as in the above proof, except now start from a hard instance of INDEPENDENT SET with a "gap" of $t = 48$. If n, m are, respectively, the number of vertices and number of edges in H, then we have $n = O(r)$, and since H is degree-bounded, $m = O(n)$. Thus $m = O(r) \leq c_0 r$ for some absolute constant c_0. Now define $\varepsilon_0 = 1/4c_0$. If a 4-coloring of H miscolors at most $\varepsilon_0 m$ edges, then since $\varepsilon_0 m \leq r/4$, there are at least $r/2$ tree-like structures such that they, and the leaf-level gadgets associated with them, are all legally colored using only 4 colors. Arguing as in the proof of Theorem 1.2, we can now conclude $\alpha(G) \geq r/48$. Thus when $\alpha(G) < r/48$, every 4-coloring of H legally colors at most $(1 - \varepsilon_0)$ fraction of the edges. □

Acknowledgment. We would like to thank Madhu Sudan for several useful discussions.

REFERENCES

[1] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, *Proof verification and hardness of approximation problems*, J. ACM, 45 (1998), pp. 501–555.

[2] S. Arora and S. Safra, *Probabilistic checking of proofs: A new characterization of NP*, J. ACM, 45 (1998), pp. 70–122.

[3] M. Bellare, O. Goldreich, and M. Sudan, *Free bits, PCPs and nonapproximability—towards tight results*, SIAM J. Comput., 27 (1998), pp. 804–915.

[4] A. Blum, *Algorithms for Approximate Graph Coloring*, Ph.D. thesis, Laboratory for Computer Science, MIT, Cambridge, MA, 1991.

[5] A. Blum, *New approximation algorithms for graph coloring*, J. ACM, 41 (1994), pp. 470–516.

[6] A. Blum and D. R. Karger, *An $\tilde{O}(n^{3/14})$-coloring algorithm for 3-colorable graphs*, Inform. Process. Lett., 61 (1997), pp. 49–53.

[7] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy, *Interactive proofs and the hardness of approximating cliques*, J. ACM, 43 (1996), pp. 268–292.

[8] U. Feige and J. Kilian, *Zero-knowledge and the chromatic number*, J. Comput. System Sci., 57 (1998), pp. 187–199.

[9] M. Fürer, *Improved hardness results for approximating the chromatic number*, in Proceedings of the 36th Annual IEEE Symposium on Foundations of Computer Science, IEEE, Piscataway, NJ, 1995, pp. 414–421.

[10] M. R. Garey and D. S. Johnson, *The complexity of near-optimal graph coloring*, J. ACM, 23 (1976), pp. 43–49.

[11] M. R. Garey and D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-completeness*, W. H. Freeman, San Francisco, 1979.

[12] V. Guruswami, S. Khanna, R. Rajaraman, B. Shepherd, and M. Yannakakis, *Near-optimal hardness results and approximation algorithms for edge-disjoint paths and related problems*, J. Comput. System Sci., 67 (2003), pp. 473–496.

[13] V. Guruswami, J. Håstad, and M. Sudan, *Hardness of approximate hypergraph coloring*, SIAM J. Comput., 31 (2002), pp. 1663–1686.

[14] M. M. Halldórsson, *A still better performance guarantee for approximate graph coloring*, Inform. Process. Lett., 45 (1993), pp. 19–23.

[15] J. Håstad, *Clique is hard to approximate within $n^{1-\epsilon}$*, Acta Math., 182 (1999), pp. 105–142.

[16] D. R. Karger, R. Motwani, and M. Sudan, *Approximate graph coloring using semidefinite programming*, J. ACM, 45 (1998), pp. 246–265.

[17] R. M. Karp, *Reducibility among combinatorial problems*, in Complexity of Computer Computations, Plenum Press, New York, 1972, pp. 85–103.

[18] A. Björklund, T. Husfeldt, and S. Khanna, *Approximating Longest Directed Path*, Report TR03-032, Electronic Colloquium on Computational Complexity (ECCC), http://eccc.uni-trier.de/eccc-reports/2003/TR03-032/index.html (2003).

[19] S. Khanna, N. Linial, and S. Safra, *On the hardness of approximating the chromatic number*, Combinatoria, 20 (2000), pp. 393–415.

[20] C. Lund and M. Yannakakis, *On the hardness of approximating minimization problems*, J. ACM, 41 (1994), pp. 960–981.

[21] C. H. Papadimitriou and M. Yannakakis, *Optimization, approximation and complexity classes*, J. Comput. System Sci., 43 (1991), pp. 425–440.

[22] E. Petrank, *The hardness of approximation: Gap location*, Comput. Complex., 4 (1994), pp. 133–157.

[23] A. Wigderson, *Improving the performance guarantee for approximate graph coloring*, J. ACM, 30 (1983), pp. 729–735.