Note

Planar graph coloring is not self-reducible, assuming $P \neq NP$

Samir Khuller*
Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA

Vijay V. Vazirani**
Department of Computer Science and Engineering, Indian Institute of Technology, New Delhi 110016, India

Communicated by D. Coppersmith
Received September 1990
Revised February 1991

Abstract


We show that obtaining the lexicographically first four coloring of a planar graph is NP-hard. This shows that planar graph four-coloring is not self-reducible, assuming $P \neq NP$. One consequence of our result is that the schema of Jerrum et al. (1986) cannot be used for approximately counting the number of four colorings of a planar graph. These results extend to planar graph $k$-coloring, for $k \geq 4$.

1. Introduction

Most known problems in NP are self-reducible. It is because of this property that the search version of an NP problem is Turing-reducible to its decision version. By suitably carrying out this reduction, the lexicographically first solution to the search problem can also be obtained. This property also plays a crucial role in reducing the

*Research done in part while this author was supported by an IBM Graduate Fellowship at Cornell University.

**Supported in part by NSF grant DCR 85-52938 and PYI matching funds from AT&T Bell Labs and Sun Microsystems, Inc. at Cornell University.
problem of approximately counting the number of solutions of an NP problem to
generating a random solution to a given instance of this problem [3].

In this paper we show that planar graph four-colorability is not self-reducible unless
P = NP. To our knowledge, this is the first such result. The usual manner of carrying
out self-reducibility of the general graph k-colorability problem does not work for our
problem since the reduction destroys planarity (see Section 2). Is there some other way
of achieving self-reducibility? We provide a negative answer as follows: the decision
version of planar graph four-colorability is in P [1]. On the other hand, we show that
obtaining the lexicographically first such solution is NP-hard, thereby proving that
the problem is not self-reducible if P \neq NP. The NP-hardness of obtaining the
lexicographically first solution contrasts with the fact that there is a polynomial-time
algorithm for obtaining an arbitrary solution [1].

One consequence of our result is that the schema of [3] cannot be used for
approximately counting the number of four-colorings of a planar graph. (The problem
of exact counting is \#P-complete [6].) We extend these results to planar graph k-
colorability for any fixed \( k \geq 4 \).

Our proofs are quite straightforward; the main interest lies in the peculiar situation
that obtaining the lexicographically first solution is NP-hard even though the decision
version is in P, which enables us to get the result. Clearly, this proof method will not
work for showing the lack of self-reducibility in problems whose decision version is
NP-hard. It will be interesting to find other such natural problems (possibly on
restricted families) that exhibit this situation, as well as discover other situations that
yield proofs of lack of self-reducibility.

2. Self-reducibility

In this section we introduce the formal definition of self-reducibility given in [4].
A different notion of self-reducibility was given in [5]; however, the first notion
appears to be more useful for relating the complexities of search and decision
problems, and random generation and approximate counting. Let \( \Sigma \) be a fixed finite
alphabet in which we are going to encode both our problem instance and the solution.
Let \( R \subseteq \Sigma^* \times \Sigma^* \) be a binary relation over \( \Sigma \). For each string (problem instance) \( x \in \Sigma^* \),
we denote by \( R(x) \) the corresponding solution set:

\[
R(x) = \{ y \in \Sigma^* : (x, y) \in R \}.
\]

By an example we illustrate what \( R \) is: Let \( x \) encode a boolean formula \( B \), and
\( y \) encode a satisfying assignment. Then we define

\[
R = \{ (x, y) : x, y \in \Sigma^* \text{ and } y \text{ is a satisfying assignment to instance } x \}.
\]

We call \( R \) self-reducible if
There exists a polynomial-time computable length function $\ell_R : \Sigma^* \to \mathbb{N}$ such that $\ell_R(x) = O(|x|^{k_R})$ for some constant $k_R > 0$, and

$$y \in R(x) \Rightarrow |y| = \ell_R(x) \quad \forall x, y \in \Sigma.$$

2. For all $x \in \Sigma^*$ with $\ell_R(x) = 0$, the predicate $A \in R(x)$ can be tested in polynomial time. ($A$ denotes the empty string over $\Sigma$.)

3. There exist polynomial-time computable functions $\psi : \Sigma^* \times \Sigma^* \to \Sigma^*$ and $\sigma : \Sigma^* \to \mathbb{N}$ satisfying

$$\sigma(x) = O(\log |x|),$$

$$\forall x \in \Sigma^* \ [\ell_R(x) > 0 \iff \sigma(x) > 0],$$

$$\forall x, w \in \Sigma^* \ [\ |\psi(x, w)| \leq |x|],$$

$$\forall x, w \in \Sigma^* \ [\ell_R(\psi(x, w)) = \max\{\ell_R(x) - |w|, 0\}],$$

and such that each solution set can be expressed in the form

$$R(x) = \bigcup_{w \in \Sigma^*} \{wy : y \in R(\psi(x, w))\}.$$

The first condition simply states that the length of the solution is bounded by some polynomial function of the problem instance. The third condition provides an inductive construction of the solution sets as follows: if the solution length is greater than 0, then $R(x)$ is partitioned into classes according to the initial segment $w$ of length $\sigma(x)$, and each class can then be expressed as the solution set of another instance $\psi(x, w)$ of the same problem, concatenated with $w$. For the set of all possible solutions $R(x)$ to a problem $x$, we can define the usual lexicographic ordering between the strings.

The following proposition is well known.

**Proposition 2.1.** For a problem $X$, that is self-reducible, given a polynomial-time decision oracle for the problem we can construct the lexicographically first solution in polynomial time.

In this formal setting, we now illustrate that general graph $k$-coloring is self-reducible.

**Problem.** Given a graph $G$ and a clique $K$ of size $k$, is $G \cup K$ $k$-colorable?

Clique $K$ has been introduced in the problem instance since it yields an easy self-reducibility. We use $R$ to encode all possible solutions to the problem. Note that $y \in R(x)$ if $y$ denotes a valid color assignment to the vertices (represented by a set of pairs $(v_i, c_i)$, where $v_i$ has color $c_i$). Clearly $G \cup K$ is $k$-colorable if and only if $G$ is $k$-colorable. Assume that $u_i$ gets color $i$ (vertices $u_i$ belong to $K$).
Suppose we wish to color vertex $v_i$ with color $j \in \{1, \ldots, k\}$. Here $w = (v_i, j)$. We now produce the graph $G_i$ as follows: delete $v_i$ from $G$, and add edges from $u_i$ (recall that $u_i$ has color $j$) to $N(v_i)$ (neighbours of $v_i$). It is easy to see that a coloring for $G_i$ can be used to obtain a coloring for $G$ by coloring $v_i$ with the color $j$ (same as $u_i$). Moreover, the size of the graph $G_i$ (measured in the number of vertices) is smaller than the size of $G$.

3. Lexicographic colorings

Every legal $k$-coloring of a graph may be represented as a string $C = c_1 c_2 c_3 \ldots c_n$, where $c_i$ is the color of vertex $v_i$. Assume that $c_i \in \{1, 2, \ldots, k\}$. Note that all strings are of length $n$ (where $n$ is the number of vertices in the graph) and the $LF$-$k$ coloring is the "smallest" (in the usual lexical ordering on strings) legal coloring which uses at most $k$ colors.

We show that computing the $LF$-$k$ coloring for a planar graph is NP-hard for any fixed $k$ ($k \geq 4$) (even though the graph is $k$-colorable) by a reduction from planar graph three-colorability which is known to be NP-hard [2].

Before illustrating the proof for arbitrary $k$, we show a simple proof for the case $k = 4$.

**Theorem 3.1.** Obtaining the $LF$-$4$ coloring for a planar graph is NP-hard.

**Proof.** We prove the problem to be NP-hard by exhibiting a simple reduction from the graph three-colorability problem. Given $G(V, E)$ (a planar graph) construct the following graph $G'(V', E')$. Assuming $G$ has $n$ vertices, the graph $G'$ has $2n$ vertices.

Define $V' = V \cup \{u_i | v_i \in V\}$.

Define $E' = E \cup \{(v_i, u_j) | v_i \in V\}$.

The vertices of $G'$ are numbered as follows: Label each $u_i$ as $i$ and each $v_i$ as $n+i$ (in $G$ each $v_i$ was numbered $i$). Note that $G'$ is planar since each vertex $u_i$ can be embedded in a face adjacent to $v_i$.

Now we obtain a $LF$-$4$ coloring for $G'$. If $G$ was three-colorable then the $LF$-$4$ coloring of $G'$ has the property that all the $u_i$ vertices are colored with color 1. This coloring is valid since no $u_i$ is adjacent to a $u_j$, and since $G$ is three-colorable, the rest of the graph can be assigned a legal coloring (without using the color 1). If the $LF$-$4$ coloring has the property that all the $u_i$ vertices are colored with color 1, then it is easy to see that all the vertices $v_i$ use only the colors from the set $\{2, 3, 4\}$ and, hence, $G$ must be three-colorable. Thus, from the $LF$-$4$ coloring it is easy to check whether the original graph $G$ is three-colorable or not. 

The proof for $k = 5, 6$ is very similar to the proof shown above, where instead of attaching a single vertex to each node of the graph we attach a $K_2$ and $K_3$, respectively, to each node of the graph by adding edges from each node of the
complete graph. The new vertices have to be numbered carefully so that each \( K_3(K_3) \) is colored with the colors 1 and 2 (1, 2 and 3) thus making these two (three) colors “forbidden” colors for the vertices in \( G \). Note that the graph \( G' \) formed in each case will be planar. We cannot attach a \( K_4 \) (for \( k = 7 \)) since that would make \( G' \) nonplanar. We develop a general “gadget” which is planar, and which can be attached to each vertex of the original graph, achieving the effect of introducing “forbidden” colors at each vertex.

**Theorem 3.2.** Obtaining the LF-\( k \) coloring (for any fixed \( k \geq 3 \)) for a planar graph is NP-hard.

**Proof.** Obtaining a LF-3 coloring is obviously NP-hard, so we concentrate our attention on the case \( k > 3 \). The idea is similar to the one used in the previous theorem. We prove the problem NP-hard by exhibiting a reduction from the graph three-colorability problem. Given \( G(V, E) \) (a planar graph) we construct a graph \( G'(V', E') \) as follows: We first show the construction of the subgraph \( H_{k'}(V'_H, E'_H) \) (where \( k' = k - 3 \)) which is used in the construction of the graph \( G' \).

The subgraphs \( H_{k'} \) are defined recursively as follows:

(a) If \( k' \leq 2 \) then \( H_{k'} = K_{k'} \) (complete graph on \( k' \) vertices).

(b) If \( k' > 2 \) then the subgraphs \( H_{k'} \) are defined recursively as follows: Each \( H_{k'} \) consists of a spine, which is a set of \( k' \) vertices \( \{ u_{k'}^0, u_{k'}^1, \ldots, u_{k'}^{k'} \} \) with \( u_{k'}^0 \) adjacent to \( u_{k'}^{k-1} \) (1 \( \leq i \leq k' \)). On each vertex \( u_{k'}^i \) (2 \( \leq i \leq k' \)) of the spine we “attach” the subgraph \( H_{k'-2} \) by adding edges from \( u_{k'}^i \) to each vertex \( u_{k'-2}^j \) on the spine of \( H_{k'-2} \) (1 \( \leq j \leq l-2 \)). More formally, we introduce the edges \( \{(u_{k'}^i, u_{k'-2}^j)\} \) (2 \( \leq i \leq k' \), 1 \( \leq j \leq l-2 \)). We refer to \( u_{k'}^i \) as the \( m \)th vertex on the spine of \( H_{k'} \). The sub-spines of \( H_{k'} \) are the spine and sub-spines of \( H_{l} \) (1 \( \leq i \leq k'-2 \)) if \( H_{l} \) is attached to \( u_{k'-2}^i \). Similarly, we can refer to a vertex as the \( m \)th vertex on a sub-spine of \( H_{k'} \) if it is the \( m \)th vertex on the spine of some \( H_{l} \) that was used in forming \( H_{k'} \). The spine of \( H_{k'} \) is also a sub-spine of \( H_{k'} \).

The graphs \( H_{k'}(i \leq 5) \) are shown in Fig. 1(a). The general structure of \( H_{k'} \) is shown in Fig. 1(b). It is easy to prove by induction on \( k' \) that \( H_{k'} \) is planar.

The graph \( G' \) is constructed by “attaching” copies of \( H_{k'} \) (call them \( H'_{k'}(V'_H, E'_H) \)) to each vertex \( v_i \) of \( G \). We add edges from \( v_i \) to each vertex \( u_{k'}^j \) (1 \( \leq j \leq k' \)) on the spine of \( H_{k'} \). More formally, we have \( V' = V \cup V'_H \) (1 \( \leq i \leq n \)). Also we have \( E' = E \cup E'_H \cup \{(v_i, u_{k'}^j)\} \) (1 \( \leq j \leq k' \}) (1 \( \leq i \leq n \)).

Each vertex \( u_{k'}^j \) is given the number \( i + n \cdot f_{k'} \), where \( f_{k'} \) is the number of vertices in \( H_{k'} \). The graphs \( H_{k'} \) may be embedded with all the sub-spines aligned vertically as shown in Fig. 1(b). Vertices belonging to \( H_{k'} \) are assigned numbers from the set \( \{1, 2, \ldots, f_{k'}\} \), with each vertex being assigned a distinct number. We assign the \( m \)th vertex of a sub-spine belonging to \( H_{k'} \) a smaller number than the \( p \)th vertex of a sub-spine belonging to \( H_{k'} \) if \( m < p \), regardless of them belonging to the same sub-spine or different sub-spines. One scheme to obtain the numbering is to number the rows left to right starting from the topmost row (see Fig. 1(a)). Each vertex in \( V'_H \) is
given the number \((i-1)f_{k'}+j\), where \(j\) is the number of the vertex in the numbering of \(H_{k'}\).

Now we obtain a \(LF-k\) coloring for \(G'\). Assume \(G\) is three-colorable. The \(LF-k\) coloring of \(G'\) has the property that the \(m\)th vertices in sub-spines are colored with color \(m\). The coloring is valid since each \(H_{k'}\) subgraph can be legally colored with \(k'\) colors. This coloring can now be extended to a complete \(k\)-coloring for the graph since
all the original vertices of G can be colored using only three colors. In fact, it is easy to
see that the LF-k coloring will have exactly this property and will color the original
vertices in a lexicographically first manner using only three colors.

If the LF-k coloring has the property that all the vertices in H_j (for 1 \leq i \leq n) in row
j are colored with color j then the graph G is three-colorable since every vertex of
the graph has vertices of the k' colors \{1, 2, 3, ..., k'\} adjacent to it (these colors are
"forbidden" colors for the original vertices of the graph and since the graph
is k-colorable, all the original nodes use only three colors).
The sizes of the graphs we generate in our reductions are easily seen to be
exponential in k. The reduction however is still a polynomial-time reduction since k is
a constant. □

From the proof of the previous theorem and the proposition of Section 2, we have
the following corollary.

**Corollary 3.3.** Planar graph k-coloring is not self-reducible, assuming NP ≠ P.

4. Open problems

As mentioned in the introduction, it will be interesting to identify other problems
that do not possess self-reducibility. Another interesting problem is to determine the
complexity of approximately computing the number of four colorings of a planar
graph. The former appears to be \#P-complete. This problem appears to be intractable – perhaps in the sense that if it were doable in random polynomial time, then
NP = RP. Finally, note that our proof technique breaks down for three-colorability of planar graphs – is this problem self-reducible?

Acknowledgment

We would like to thank Howard Karloff and Mihalis Yannakakis for useful
discussions.

References

(Freeman, San Francisco, 1978).
322–333.