JOURNAL OF ALGORITHMS 11, 102–116 (1990)

Improved Edge-Coloring Algorithms for Planar Graphs

MAREK CHROBAK*

Department of Mathematics and Computer Science, University of California, Riverside, California 92521

AND

TAKAO NISHIZEKI

Department of Electrical Communications, Tohoku University, Sendai 980, Japan

Received October 21, 1988; revised January 25, 1989

We consider the problem of edge-coloring planar graphs. It is known that a planar graph G with maximum degree $\Delta \geq 8$ can be colored with Δ colors. We present two algorithms which find such a coloring when $\Delta \geq 9$. The first one is a sequential $O(n \log n)$ time algorithm. The other one is a parallel EREW PRAM algorithm which works in time $O(\log^3 n)$ and uses $O(n)$ processors. © 1990 Academic Press, Inc.

1. INTRODUCTION

An edge-coloring of a graph G is an assignment of colors to the edges of G such that edges with a common endpoint have different colors. Let $\chi'(G)$ denote the chromatic index of G, that is the minimum number of colors necessary to color the edges of G. Vizing [25] proved that $\chi'(G)$ is either $\Delta(G)$ or $\Delta(G) + 1$ for each graph G, where $\Delta(G)$ denotes the maximum degree of a vertex in G (see also [9]). Therefore each graph belongs to one of two classes: either to *class* 1 which contains all graphs G such that $\chi'(G) = \Delta(G)$, or to *class* 2 which contains all graphs G such that $\chi'(G) = \Delta(G) + 1$. The problem of deciding whether a given graph belongs to class 1 or 2 is known as the *classification problem* and has a rich literature (see the bibliography in [9]).

*On leave from Institute of Informatics, Warsaw University, PKIN VIII p, 00-901 Warsaw, Poland. This research was done in part while the first author was visiting University of Chicago and Columbia University.

0196-6774/90 $3.00

The classification problem is NP-complete even when restricted to cubic graphs [16, 17, 20], and this implies that there are no polynomial-time algorithms for this problem, unless P = NP. Therefore it is natural to look for approximate algorithms which use $\Delta(G) + 1$ colors for each G. A straightforward implementation of the proof of Vizing's theorem yields such an algorithm with complexity $O((m + n)^2)$. With some care the complexity can be improved to $O(mn)$ (see, for example, [12]). Gabow *et al.* [12] describe also algorithms for this problem with complexity $O(\Delta m \log n)$ and $O(m(n \log n)^{1/2})$.

Vizing's theorem does not hold for multigraphs. In this case Shannon [24] proved that, if G is a multigraph, then $\chi'(G) \leq \lfloor 3\Delta(G)/2 \rfloor$. Hochbaum, Nishizeki, and Shmoys [15] give an efficient approximation algorithm for edge-coloring multigraphs.

The problem of coloring graphs with $\Delta(G) + 1$ colors in parallel is one of the major open problems in the theory of parallel algorithms. So far it is only known that, as it was shown by Karloff and Shmoys [18], the problem is in the class NC for graphs G with $\Delta(G) = O(\log^k n)$, where k is a fixed constant. For the general case the problem is open.

In view of the difficulty of the general edge coloring problem, some special classes of graphs have been investigated. Much research has been done on bipartite graphs. A classical theorem of König and Hall (see, for example, [2]) states that all bipartite graphs are in class 1. There exist also efficient algorithms for optimal edge coloring bipartite graphs, both sequential and parallel, [6, 11, 21].

In this paper we concentrate on another class of graphs for which the classification problem has been deeply investigated, that is on the class of planar graphs. In this case the situation is rather peculiar. Vizing [26] proved that all planar graphs G with $\Delta(G) \geq 8$ belong to class 1. He also conjectured that this can be extended to $\Delta(G) = 6, 7$, but this is still open (see [9]). If $2 \leq \Delta(G) \leq 5$ then G can belong to either of the classes. For $\Delta(G) = 2$ the problem is simple: G can be either in class 1 or 2, depending on whether G has all cycles of even length or not. If G is cubic then G is in class 1 providing that it does not have bridges. However, this result was shown by Tait (see [9]) to be equivalent to the Four Color Theorem, which should convince the reader that the problem is already non-trivial for this case. For $\Delta(G) = 4, 5$ the problem is suspected to be NP-complete, but no one has been able to prove it so far, and the constructions from [16, 17, 20] do not work for planar graphs.

The complexity of edge-coloring planar graphs has been considered already in some articles. Gabow *et al.* [12] give an $O(n^2)$ algorithm which colors each planar graph G such that $\Delta(G) \geq 8$ with $\Delta(G)$ colors. This algorithm is based on a modified proof of Vizing's theorem for planar graphs. Boyar and Karloff [3] prove that this problem is in NC if $\Delta(G) \geq 23$.

Their algorithm runs in time $O(\log^3 n)$ and uses $O(n^2)$ processors. These results have been recently improved by Chrobak and Yung [5], who give a linear-time sequential algorithm and a parallel EREW PRAM algorithm for this problem when $\Delta(G) \geq 19$. This parallel algorithm runs in time $O(\log^2 n)$ and uses $O(n)$ processors. Additionally, it was shown in [5] that the methods used there cannot work for graphs of smaller degree.

In this paper we present two algorithms which, given a planar graph G such that $\Delta(G) \geq 9$, color the edges of G with $\Delta(G)$ colors. The first algorithm is sequential and runs in time $O(n \log n)$. The second one is a parallel EREW PRAM algorithm which runs in time $O(\log^3 n)$ and uses $O(n)$ processors.

In the sequential case, this result improves the time complexity of the problem. In the parallel case, we extend the class of planar graphs for which the problem is known to be in NC. Note that the parallel algorithm is close to optimal, since the time-processor product is $O(n \log^3 n)$, missing the optimality only by the factor $O(\log^2 n)$.

The technique we use is based on the proof of Vizing's theorem for planar graphs. However, the original proof was not sufficient for our purpose. Roughly, in the proof of Vizing's theorem it is shown that each planar graph G with $\Delta(G) \geq 8$ contains an edge e with the property that if we remove e and color the remaining graph with $\Delta(G)$ colors, then the obtained coloring can be extended to e without using more colors. Such edges will be called reducible. We strengthen this result for $\Delta(G) \geq 9$ by showing that in this case the number of such edges is $\Theta(n)$, where n is the number of vertices of G. We also present an example that this is not true if $\Delta(G) = 8$; in this case it may happen that G will have only $O(1)$ such edges.

Unfortunately, for this reason, we were unable to extend our method to the case when $\Delta(G) = 8$. It appears that in order to obtain an NC parallel algorithm for this case (and a more efficient sequential one) one has to find a new reduction technique, stronger than the one used in the proof of Vizing's theorem and in this paper.

Our algorithms do not use an embedding of a given graph. In fact, any graph can be given on input. If the algorithm fails, or if it works too long, it will mean that the input graph was not planar. (Actually, the algorithms work correctly also for toroidal graphs and, more generally, for sufficiently large graphs of bounded genus).

Let us finally mention a related problem: vertex coloring of planar graphs. The proof of the Four Color Theorem [1] seems to yield a sequential $O(n^2)$ algorithm for 4-coloring every planar graph, but the parallel complexity of this problem is open. There are, however, many algorithms for coloring planar graphs with five colors. Sequential algorithms in [4, 8, 27, 22] achieve linear-time complexity. In parallel the problem can be solved in time $O(\log n \log^* n)$ [13, 14], even on an EREW PRAM with $O(n/\log n \log^* n)$ processors and without using an embedding of a given graph [14]. (Somewhat less efficient algorithms can also be found in [3, 23].)

2. Some Combinatorial Results

Let $G = (V, E)$ be a planar graph. For $u \in V$ by $\deg(u)$ we denote the degree of u, and $\Delta(G) = \max_{u \in V} \deg(u)$. By $\deg_v^*(u)$ we denote the number of neighbours of u of degree $\Delta(G)$ different than v. By n, and respectively m, we will always denote the number of vertices and edges of G.

An edge $(u, v) \in E$ is called *reducible* if either $\deg(u) + \deg_u^*(v) \le \Delta(G)$ or $\deg_v^*(u) + \deg(v) \le \Delta(G)$. A vertex $u \in V$ is *good* if there is a neighbour v of u such that (u, v) is reducible, otherwise it is *bad*.

The theorem below is proven by a rather complicated manipulation of inequalities, but the intuitions behind the proof are rather simple: A planar graph with n vertices has at most $3n$ edges. On the other hand, if an edge (u, v) is not reducible, then both u and v must have some neighbours of degree Δ. Thus, the more non-reducible edges a graph has, the more dense it becomes. Therefore, a planar graph cannot have too many non-reducible edges, because of its low density.

Theorem 1. *Let $G = (V, E)$ be a planar graph without isolated vertices, and R the set of reducible edges in G. If $\Delta(G) \ge 9$ then $|R| \ge \frac{1}{32}n$.*

Proof. We introduce first some notation. Let $\Delta = \Delta(G)$. We say that $u \in V$ is of type $(i_1, \dots, i_8)$ if u has i_d neighbours of degree d, for each $d = 1, \dots, 8$. By n_d we denote the number of vertices of degree d. We will use the convention, that if x is the number of vertices with some property, then $\bar{x}$ and $\tilde{x}$ denote, respectively, the number of good and bad vertices with this property.

Additionally, we define:

- $\tilde{p}_d$ is the number of bad vertices of degree d which have exactly two neighbours of degree Δ.
- $\bar{n}_{\Delta 1}$ is the number of good vertices of degree Δ which have at least one non-reducible edge incident to it.
- $\tilde{N}_\Delta(i_1, \dots, i_8)$ is the set of bad vertices of degree Δ and of type $(i_1, \dots, i_8)$.
- $\bar{N}_\Delta(i_1, \dots, i_8, c)$ is the set of good vertices u of degree Δ and type $(i_1, \dots, i_8)$ such that at least one of the edges incident to u is not reducible and $c = \min\{\deg(v) | (u, v) \in E \text{ is not reducible}\}$.

We will prove first some auxiliary lemmas.

LEMMA 1. (a) *For each* $u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)$,

$$\sum_{d=c}^{8} \frac{i_d}{d-1} \leq 1.$$

(b) *For each* $u \in \tilde{N}_\Delta(i_1, \ldots, i_8)$,

$$\sum_{d=2}^{8} \frac{i_d}{d-1} \leq 1.$$

Proof. (a) Let $u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)$. Then u has a neighbour v of degree c such that (u, v) is not reducible. Clearly, $c \geq 2$. Since (u, v) is not reducible, $\deg_v^*(u) \geq \Delta + 1 - c$. Therefore u has at most $\Delta - \deg_v^*(u) \leq c - 1$ neighbours of degree smaller than Δ. So we obtain that

$$\sum_{d=c}^{8} \frac{i_d}{d-1} \leq \frac{1}{c-1} \sum_{d=c}^{8} i_d \leq 1.$$

(b) The proof is similar to (a). We only have to choose v to be the neighbour of u of minimum degree c and note that then $i_d = 0$ for $d = 1, \ldots, c-1$. □

LEMMA 2. $\tilde{p}_d \leq n_{\Delta-1}$ *for* $d = 3, 4$.

Proof. Let $\tilde{P}_d$ be the set of bad vertices of degree d, which have exactly two neighbours of degree Δ. We have $|\tilde{P}_d| = \tilde{p}_d$, $d = 3, 4$.

Consider first the case $d = 3$, and let $u \in \tilde{P}_3$. Since u is bad, it has two neighbors of degree Δ and one neighbour of degree $\Delta - 1$. Also, if a vertex v has degree $\Delta - 1$, then v can have at most one neighbour u in $\tilde{P}_3$, because (u, v) is not reducible. Therefore, if we denote by a the number of edges between vertices in $\tilde{P}_3$ and vertices of degree $\Delta - 1$ then we have $\tilde{p}_3 = a \leq n_{\Delta-1}$, which proves the lemma for $d = 3$.

Consider now the case $d = 4$, and let $u \in \tilde{P}_4$. Since u is bad, it has two neighbours of degree Δ and two neighbours of degree $\Delta - 1$. Also, if a vertex v has degree $\Delta - 1$, then v can have at most two neighbours u_1, u_2 in $\tilde{P}_4$, because (u_1, v) is not reducible. Therefore, if we denote by b the number of edges between vertices in $\tilde{P}_4$ and vertices of degree $\Delta - 1$ then we have $2\tilde{p}_4 = b \leq 2n_{\Delta-1}$, which proves the lemma for $d = 4$. □

Now we continue the proof of the theorem. From Euler's theorem that $m \leq 3n$, after substituting $m = \frac{1}{2}\sum_{d \geq 1} dn_d$, $n = \sum_{d \geq 1} n_d$, we have

$$\sum_{d=1}^{5} (6-d)n_d \geq \sum_{d \geq 7} (d-6)n_d \geq 2n_8 + 3\sum_{d \geq 9} n_d. \tag{1}$$

A bad vertex must have at least two neighbours of degree Δ. On the other hand, if $\deg(u) = \Delta$ and u has a bad neighbour of degree $d \leq 8$, then either $u \in \tilde{N}_\Delta(i_1, \ldots, i_8)$, or else $u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)$ for some $c \leq d$. Therefore, counting for each $d = 2, \ldots, 8$ the number of edges between bad vertices of degree d and vertices of degree Δ, we have

$$2\tilde{n}_d \leq \sum_{(i_1, \ldots, i_8)} \left(\sum_{u \in \tilde{N}_\Delta(i_1, \ldots, i_8)} i_d + \sum_{c=2}^{d} \sum_{u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)} i_d \right). \tag{2}$$

For $d = 3, 4$ these inequalities can be improved:

$$2\tilde{p}_d + 3(\tilde{n}_d - \tilde{p}_d) \leq \sum_{(i_1, \ldots, i_8)} \left(\sum_{u \in \tilde{N}_\Delta(i_1, \ldots, i_8)} i_d + \sum_{c=2}^{d} \sum_{u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)} i_d \right). \tag{3}$$

After dividing (2) for $d \neq 3, 4$ and (3) by $d - 1$ and adding them together, we have

$$\begin{aligned}
2\tilde{n}_2 &+ \frac{1}{2}(2\tilde{p}_3 + 3(\tilde{n}_3 - \tilde{p}_3)) + \frac{1}{3}(2\tilde{p}_4 + 3(\tilde{n}_4 - \tilde{p}_4)) + \sum_{d=5}^{8} \frac{2}{d-1}\tilde{n}_d \\
&\leq \sum_{d=2}^{8} \sum_{(i_1, \ldots, i_8)} \left(\sum_{u \in \tilde{N}_\Delta(i_1, \ldots, i_8)} \frac{i_d}{d-1} + \sum_{c=2}^{d} \sum_{u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)} \frac{i_d}{d-1} \right) \\
&\leq \sum_{(i_1, \ldots, i_8)} \sum_{u \in \tilde{N}_\Delta(i_1, \ldots, i_8)} \sum_{d=2}^{8} \frac{i_d}{d-1} \\
&\quad + \sum_{(i_1, \ldots, i_8)} \sum_{c=2}^{8} \sum_{u \in \overline{N}_\Delta(i_1, \ldots, i_8, c)} \sum_{d=c}^{8} \frac{i_d}{d-1} \\
&\leq \tilde{n}_\Delta + \bar{n}_{\Delta 1}.
\end{aligned} \tag{4}$$

In the last inequality we used Lemma 1. Rearranging (4) we have immediately

$$2\tilde{n}_2 + \frac{3}{2}\tilde{n}_3 + \tilde{n}_4 + \sum_{d=5}^{8} \frac{2}{d-1}\tilde{n}_d \le \bar{n}_{\Delta 1} + \tilde{n}_\Delta + \frac{1}{2}\tilde{p}_3 + \frac{1}{3}\tilde{p}_4. \tag{5}$$

Now, from (5)

$$\begin{aligned}
2\sum_{d=1}^{8} \bar{n}_d + \bar{n}_{\Delta 1} &\ge 2n_1 + \left(2\bar{n}_2 + \frac{3}{2}\bar{n}_3 + \bar{n}_4 + \sum_{d=5}^{8}\frac{2}{d-1}\bar{n}_d\right) + \bar{n}_{\Delta 1}\\
&\ge 2n_1 + \left(2n_2 + \frac{3}{2}n_3 + n_4 + \sum_{d=5}^{8}\frac{2}{d-1}n_d\right)\\
&\quad - \left(2\tilde{n}_2 + \frac{3}{2}\tilde{n}_3 + \tilde{n}_4 + \sum_{d=5}^{8}\frac{2}{d-1}\tilde{n}_d\right) + \bar{n}_{\Delta 1}\\
&\ge 2n_1 + 2n_2 + \frac{3}{2}n_3 + n_4 + \sum_{d=5}^{8}\frac{2}{d-1}n_d\\
&\quad - \tilde{n}_\Delta - \frac{1}{2}\tilde{p}_3 - \frac{1}{3}\tilde{p}_4.
\end{aligned} \tag{6}$$

Using (6) and (1) we derive

$$\begin{aligned}
2\sum_{d=1}^{8} \bar{n}_d + \bar{n}_{\Delta 1} &\ge \frac{1}{8}\sum_{d=1}^{8} n_d + \frac{3}{8}\sum_{d=1}^{5}(6-d)n_d + \frac{9}{56}n_8 - \tilde{n}_\Delta - \frac{1}{2}\tilde{p}_3 - \frac{1}{3}\tilde{p}_4\\
&\ge \frac{1}{8}\sum_{d=1}^{8} n_d + \frac{3}{8}\left(2n_8 + 3\sum_{d\ge 9} n_d\right)\\
&\quad + \frac{9}{56}n_8 - \tilde{n}_\Delta - \frac{1}{2}\tilde{p}_3 - \frac{1}{3}\tilde{p}_4\\
&\ge \frac{1}{8}\sum_{d\ge 1} n_d + \frac{51}{56}n_8 + \sum_{d\ge 9} n_d - \tilde{n}_\Delta - \frac{1}{2}\tilde{p}_3 - \frac{1}{3}\tilde{p}_4\\
&\ge \frac{1}{8}n + \frac{51}{56}n_{\Delta-1} - \frac{1}{2}\tilde{p}_3 - \frac{1}{3}\tilde{p}_4.
\end{aligned} \tag{7}$$

Now, using Lemma 2, we derive from (7) that

$$\begin{aligned}
2\sum_{d=1}^{8} \bar{n}_d + \bar{n}_{\Delta 1} &\ge \frac{1}{8}n + \frac{1}{2}(n_{\Delta-1} - \tilde{p}_3) + \frac{1}{3}(n_{\Delta-1} - \tilde{p}_4)\\
&\ge \frac{1}{8}n.
\end{aligned}$$

Finally, observing that $2|R| \geq \Sigma_{d=1}^{8}\bar{n}_d + \bar{n}_{\Delta 1}$, we obtain

$$|R| \geq \frac{1}{4}\left(2\sum_{d=1}^{8}\bar{n}_d + \bar{n}_{\Delta 1}\right) \geq \frac{1}{32}n. \qquad \square$$

Note that in the proof we do not use any property of graphs' embedding in the plane, besides Euler's formula. For graphs of genus g, Euler's formula reads $m \leq 3n + 6(g - 1)$, and it differs from that for planar graphs only by an additive constant $6g$. In the proof we have neglected the constant -6 from Euler's formula, so Theorem 1 is true also for toroidal graphs. For graphs of bounded genus the following formula holds: $|R| \geq \frac{1}{32}n - c_g$, where c_g is a constant dependent on g. Therefore, in the case of graphs of genus g our algorithms will work correctly when n is large enough.

Theorem 1 is optimal, in the sense that it is not true for $\Delta(G) = 8$. We construct first a counterexample on a torus. Imagine a $k \times l$-tiling of a torus with regular hexagons, where both k and l are even. Consider a cubic graph whose regions are these hexagons. Put a vertex on each edge of this graph. For each region, create a length-6 cycle containing new vertices on the boundary of this region. Now old vertices have degree 3 and new vertices have degree 6. Inside this cycle join every second vertex by an edge, obtaining a triangle. It is easy to do it in such a way that new vertices have now degree 8. So finally we obtain a toroidal graph whose vertices have degree 3 or 8, vertices of degree 3 have only neighbours of degree 8, and each vertex of degree 8 has exactly two neighbours of degree 3. If we call the obtained graph by G_1, we have $\Delta(G_1) = 8$, and $\deg(u) + \deg_u^*(v) \geq 9$ for each edge (u, v) in G_1. Therefore G_1 does not have reducible edges at all.

To modify this example, instead of tiling the torus, we start with a $2 \times l$-tiling of the sphere without the poles (that is, we divide the sphere into l horizontal strips, each tiled with two hexagons, and two regions at the poles). We repeat the above procedure. Then the obtained graph G_2 is planar, again $\Delta(G_2) = 8$, but the number of reducible edges in G_2 is $O(1)$.

3. Fan Sequences

Let $G = (V, E)$ be a planar graph such that some of its edges already colored. We use colors from the set $\{1, 2, \ldots, \Delta\}$. By $\mathrm{col}(x, y)$, for $(x, y) \in E$, we denote the color of the edge (x, y). For $x \in V$ we define $\mathrm{Used}(x)$ to be the set of colors used at x and $\mathrm{Free}(x)$ to be the set of colors free at x. Clearly, for each $x \in V$, $\mathrm{Used}(x) \cup \mathrm{Free}(x) = \{1, 2, \ldots, \Delta\}$.

Let $(u, v) \in E$, and suppose that all edges $(u, x) \in E$ except (u, v) are already colored. By a *fan sequence* centered at u and starting at v we mean a sequence of u's neighbours $F = [v = x_0, x_1, \ldots, x_k]$, where all the x_i are different, and

$$(s) \qquad \mathrm{col}(u, x_{i+1}) \in \mathrm{Free}(x_i) \qquad \text{for } i = 0, 1, \ldots, k-1.$$

By $\mathcal{S}_{u,v}$ we denote the family of all fan sequences centered at u and starting at v. If, additionally to (s), a fan sequence $F = [x_0, x_1, \ldots, x_k] \in \mathcal{S}_{u,v}$ satisfies

$$(l) \qquad \mathrm{Free}(x_k) \cap \mathrm{Free}(u) \neq \varnothing,$$

then F is called a *local fan*. $\mathcal{L}_{u,v}$ is the family of all local fans in $\mathcal{S}_{u,v}$. If $F \in \mathcal{L}_{u,v}$ then we can extend the coloring to (u, v) by simply rearranging the colors around u as follows:

begin
  Let $c \in \mathrm{Free}(x_k) \cap \mathrm{Free}(u)$;
  for $i = 0, 1, \ldots, k-1$ **do** $\mathrm{col}(u, x_i) := \mathrm{col}(u, x_{i+1})$;
  $\mathrm{col}(u, x_k) := c$
end.

If c, d are different colors, then by a *(c, d)-path* we will understand a maximal alternating path colored with colors c, d. By *recoloring P* we will mean exchanging the colors c, d on P.

LEMMA 3. *Let $F \in \mathcal{S}_{u,v}$, $F = [x_0, x_1, \ldots, x_k]$, and $c \in \mathrm{Free}(u)$, $d \in \mathrm{Free}(x_k)$. Suppose that the (c, d)-path P starting from u (possibly empty) avoids x_k. Then, after possibly recoloring P, there is a local fan $F_0 \in \mathcal{L}_{u,v}$.*

Proof. F_0 is constructed as follows:

begin
  if $c \in \mathrm{Free}(x_j)$ for some $0 \leq j \leq k$ **then** $F_0 := [x_0, x_1, \ldots, x_j]$
  else begin
    let i be smallest such that $d \in \mathrm{Free}(x_i)$
      and x_i does not belong to P;
    recolor P;
    $F_0 := [x_0, x_1, \ldots, x_i]$
  end
end.

We will show that F_0 is indeed a local fan. The case when $c \in \mathrm{Free}(x_j)$, for some $0 \leq j \leq k$, is trivial. Therefore assume that $c \in \mathrm{Used}(x_j)$ for $j = 0, 1, \ldots, k$. By the assumption that P avoids x_k the number i must exist.

Case 1. Suppose first that $d \in \mathrm{Used}(x_p)$ for every $p = 0, 1, \ldots, i-1$. Then $\mathrm{col}(u, x_p) \neq d$ for $p = 1, \ldots, i$, since otherwise we would have $d \in \mathrm{Free}(x_{p-1})$, a contradiction with the assumption of this case. Thus $\mathrm{col}(u, x_p)$

does not change after recoloring P, for $p = 1, 2, \ldots, i$, and (s) holds. Also, after recoloring P we have $d \in \mathrm{Free}(u) \cap \mathrm{Free}(x_i)$, so (l) is satisfied.

Case 2. The second case is when $d \in \mathrm{Free}(x_p)$ for some $0 \le p \le i - 1$. By the definition of i, there is only one such number p, and x_p belongs to P. Also, $\mathrm{col}(u, x_r) \ne d$ for $r = 1, 2, \ldots, p, p + 2, \ldots, i$, because otherwise $d \in \mathrm{Free}(x_{r-1})$ and $r - 1 \ne p$.

The colors $\mathrm{col}(u, x_r)$, for $r = 1, 2, \ldots, p, p + 2, \ldots, i$ do not change, so after recoloring P we have $\mathrm{col}(u, x_{r+1}) \in \mathrm{Free}(x_r)$ for $r = 0, 1, \ldots, p - 1, p + 1, \ldots, i - 1$. If $\mathrm{col}(u, x_{p+1}) \ne d$ then also $\mathrm{col}(u, x_{p+1})$ does not change and (s) holds for F_0. Else, if $\mathrm{col}(u, x_{p+1}) = d$, then after recoloring P we have $\mathrm{col}(u, x_{p+1}) = c \in \mathrm{Free}(x_p)$, because x_p belonged to P, so (s) is satisfied too. Finally, (l) holds because $d \in \mathrm{Free}(u) \cap \mathrm{Free}(x_i)$. □

Let $F_1, F_2 \in \mathscr{S}_{u,v}$, $F_1 \ne F_2$, where $F_1 = [x_0, x_1, \ldots, x_k]$, and $F_2 = [y_0, y_1, \ldots, y_l]$. Clearly, $x_0 = y_0 = v$. Then the pair $F = (F_1, F_2)$ is called a *double fan* if $x_k \ne y_l$ and

$$(d) \qquad \mathrm{Free}(x_k) \cap \mathrm{Free}(y_l) \ne \varnothing .$$

$\mathscr{D}_{u,v}$ is the family of all double fans (F_1, F_2), for $F_1, F_2 \in \mathscr{S}_{u,v}$. We show how a double fan $F = (F_1, F_2)$ can be transformed into a local fan F_0, thus enabling us to extend the coloring to (u, v):

begin
    let $c \in \mathrm{Free}(u)$;
    let $d \in \mathrm{Free}(x_k) \cap \mathrm{Free}(y_l)$;
    if $c \in \mathrm{Free}(x_j)$ for some $1 \le j \le k$ **then** $F_0 := [x_0, x_1, \ldots, x_j]$
    else
        if $c \in \mathrm{Free}(y_j)$ for some $1 \le j \le l$ **then** $F_0 := [y_0, y_1, \ldots, y_j]$
        else begin
            let P be the (c, d)-path starting from u;
            exchange the colors on P;
            if P contains x_k **then** $F' := F_2$ **else** $F' := F_1 I$;
            construct F_0 from F' as in the proof of Lemma 3.
        end
end.

Using Lemma 3, we immediately obtain that this procedure is correct; that is, F_0 is a local fan.

If F is a double fan, as defined above, and c, d are colors, then F is called a *(c, d)-fan* if $c \in \mathrm{Free}(u)$ and $d \in \mathrm{Free}(x_k) \cap \mathrm{Free}(y_l)$.

Let $G = (V, E)$ be a planar graph with $\Delta = \Delta(G) \ge 8$. Vizing [26] and Gabow *et al.* [12] proved that then G must have a reducible edge. Let (u, v) be such an edge, satisfying $\deg_v^*(u) + \deg(v) \le \Delta$. We will show that after removing (u, v) and coloring the remaining edges with Δ colors we can extend the coloring to (u, v). This will prove that $\chi'(G) = \Delta(G)$, that is G is in class 1.

Let $c \in \text{Free}(v)$. We will attempt to color (u, v) by constructing a fan sequence F_c as follows. The first vertex in F_c is $x_0 = v$. If $c \in \text{Free}(u)$ then we can color (u, v). Else, find an edge (u, x_1) such that $\text{col}(u, x_1) = c$ and extend F_c to $[x_0, x_1]$. Suppose that we have already $F_c = [x_0, x_1, \dots, x_k]$. If $\text{Free}(x_k) \cap \text{Free}(u) \neq \varnothing$ then F_c is a local fan, as before. Also, if $\text{Free}(x_k) \cap \text{Free}(x_j) \neq \varnothing$ for some $0 \leq j \leq k - 1$ then (F_c, F_c'), for $F_c' = [x_0, x_1, \dots, x_j]$, is a double fan and we can color (u, v). Otherwise, if $\text{Free}(x_k) \neq \varnothing$, let x_{k+1} be a vertex such that $\text{col}(u, x_{k+1}) \in \text{Free}(x_k)$ and set $F_c := [x_0, x_1, \dots, x_{k+1}]$. Clearly, x_{k+1} cannot be equal to any x_i, $i = 0, 1, \dots, k$. Note also that $\text{Free}(x_k) = \varnothing$ only when $\deg(x_k) = \Delta$.

Suppose that we have already constructed F_c for each $c \in \text{Free}(v)$ and we failed to color (u, v). In this case all these fan sequences F_c end in vertices of degree Δ different than v. We have $|\text{Free}(v)| = \Delta - \deg(v) + 1$ fan sequences F_c. But u has at most $\deg_v^*(u) \leq \Delta - \deg(v)$ neighbours of degree Δ different than v, so there are two fan sequences, say F_c and F_d, which end in the same vertex x of degree Δ. Remove x from F_c and F_d, and if they still have a common vertex at the end, remove this vertex too, and so on. Finally, we obtain two fan sequences F_c' and F_d'. Then (F_c', F_d') is a double fan and we can color (u, v). This completes the proof that $\chi'(G) = \Delta$.

Our algorithm will be based on the above procedure. Theorem 1 says that $\Delta(G) \geq 9$ implies that the number of reducible edges in G is $\Theta(n)$, and indeed, we will reduce, and later color $\Theta(n)$ edges simultaneously. This cannot be done without care, especially when we have to recolor an alternating path. To avoid difficulties we will consider (c, d)-fans separately for different pairs (c, d), and additionally choose the fans to be, in some sense, independent.

4. Fan-Conflict Graph

Let $e_1 = (u, v)$, $e_2 = (x, y)$ be two edges of G. Then, e_1, e_2 are called *2-independent*, if the distance between the endpoints of e_1 and e_2 is at least 2. In other words, $\{u, v\} \cap \{x, y\} = \varnothing$, and there are no edges (u, x), (x, y), (v, x), and (v, y) in G.

Let $\mathscr{F}$ be a family of (c, d)-fans, such that for each $F, F' \in \mathscr{F}$, *if* $F \in \mathscr{D}_{u,v}$ and $F' \in \mathscr{D}_{x,y}$ then (u, v) and (x, y) are 2-independent.

We define the *fan-conflict graph* for $\mathscr{F}$ as the graph $C(\mathscr{F}) = (\mathscr{F}, \mathscr{E})$, where the set of edges $\mathscr{E}$ is determined as follows. For each $F \in \mathscr{F} \cap \mathscr{D}_{u,v}$ let $P_{c,d}(F)$ be the (c, d)-path starting from u. Then $(F, F') \in \mathscr{E}$ iff either $P_{c,d}(F)$ ends in a vertex of F' or $P_{c,d}(F')$ ends in a vertex of F. (Saying that a path P ends in a fan sequence F we mean that it ends in an element of this sequence or in the center.)

Suppose that $|\mathscr{F}| = f$. Then, obviously, $|\mathscr{E}| \leq f$. Let $\mathscr{H} \subseteq \mathscr{F}$ be the set of vertices of degree at most 2 in $C(\mathscr{F})$. A simple calculation shows that $|\mathscr{H}| \geq \frac{1}{3}f$. Therefore, if $\mathscr{I}$ is a maximal independent subset of $\mathscr{H}$ then $|\mathscr{I}| \geq \frac{1}{9}f$.

These considerations will be applied to find the set of double fans to be recolored in our algorithm. Consider two fans $F, F' \in \mathscr{I}$. By the definition of $\mathscr{I}$, the paths $P_{c,d}(F)$, $P_{c,d}(F')$ are disjoint. Additionally, even though the path $P_{c,d}(F')$ may contain a vertex in F, recoloring $P_{c,d}(F')$ will not affect F. For suppose that $F = [x_0, x_1, \ldots, x_k]$, and that x_i belongs to P. Let $c \in \text{Free}(u)$. Then $\text{col}(u, x_i) \notin \{c, d\}$, and $c, d \in \text{Used}(x_i)$. Recoloring $P_{c,d}(F')$ does not change $\text{Free}(x_i)$, so fan F will also remain unchanged.

Therefore, we can recolor paths $P_{c,d}(F)$ for $F \in \mathscr{I}$ in parallel, without collisions.

5. General Strategy

Both algorithms, sequential and parallel, are based on the same general strategy which will be described in this section. Then, in the next sections we will explain how this strategy can be implemented to achieve the claimed complexity bounds.

If $\Delta(G) \geq 19$ then we can apply the algorithm from [5]. Therefore we can assume from now on that $9 \leq \Delta(G) \leq 18$.

The algorithm is divided into two *stages*: reduction and coloring. Both stages consist of loop statements, a single execution of such loop will be called a *phase*.

During the execution of the algorithm our graph will be reduced or enlarged at each phase. To avoid confusion, by $G_0 = (V_0, E_0)$ we will denote the input graph, $n_0 = |V_0|$, and $G = (V, E)$ is the current graph.

We are concerned mainly with the asymptotic complexity of the algorithm, as well as simple and uniform description. We are aware that some of the steps, especially in the sequential implementation can be realized more efficiently than described below (for example, steps (2)–(5) can be done simultaneously by searching the graph).

```
      begin
         G := G_0;
(1)      construct the representation of G;
         p := 0;
stage 1:
         while V ≠ ∅ do
         begin
            p := p + 1;
(2)         find the set R of reducible edges in G;
```

```
(3)         find a maximal set I_p ⊆ R of 2-independent edges;
(4)         E := E\I_p;
(5)         V := {v ∈ V | deg(v) ≥ 1}
        end
        p_0 := p;
stage 2:
        for p := p_0 downto 1 do
        begin
(6)         E := E ∪ I_p;
(7)         V := V ∪ {x | (x, y) ∈ I_p for some y};
(8)         while there are uncolored edges in I_p do
            begin
                let K be the set of uncolored edges in I_p;
                for each (u, v) ∈ K such that deg(v) + deg*_v(u) ≤ Δ do
(9)                 construct a fan (local or double) F_{u,v} ∈ 𝒮_{u,v};
(10)            for each local fan F_{u,v} do color (u, v);
(11)            partition the set of remaining fans into the
                    sets 𝒜_{a,b} where 𝒜_{a,b} contains (a, b)-fans;
(12)            choose 𝒜_{c,d} with greatest cardinality;
(13)            construct the fan-conflict graph C(𝒜_{c,d});
(14)            find the set ℋ of vertices of degree at most 2 in C(𝒜_{c,d});
(15)            find a maximal independent subset 𝒥 ⊆ ℋ;
(16)            for each F_{u,v} ∈ 𝒥 do
                begin
(17)                recolor P_{c,d}(F_{u,v});
(18)                transform F_{u,v} into a local fan;
(19)                color (u, v)
                end
            end
        end
    end.
```

Step (1) can be realized within the claimed complexity bounds both in sequential and parallel by the standard techniques, so in the next sections we will analyze only stages 1 and 2.

6. Sequential Algorithm

In this section we will prove that the general strategy from the previous section can be realized sequentially in time $O(n_0 \log n_0)$. Let us analyze the consecutive steps of the algorithm.

Steps (2)–(5) can be done easily in time $O(n)$ by searching G. Therefore, a phase of stage 1 costs time $O(n)$. But, by Theorem 1, we have $|R| = \Theta(n)$, and using the fact that $\Delta \leq 18$, we obtain $|I_p| = \Theta(n)$ as well. Therefore the size of G decreases geometrically at each phase, what implies that the total cost of stage 1 is $O(n_0)$.

In stage 2 the steps (6)–(7) are also easy to do in time $O(n)$. Also steps (10)–(19) cost time $O(n)$. By the considerations from Section 5 and by the choice of $\mathscr{A}_{c,d}$ we have either $\Theta(|K|)$ local fans recolored in (10), or else $\mathscr{I} = \Theta(|K|)$. Therefore loop (8) is executed $O(\log n)$ times. So the total time for loop (8) in one phase is $O(n \log n)$. This implies that one phase of stage 2 can be done in time $O(n \log n)$, and by summing these terms for each phase we obtain that the total execution time of stage 2, and of the whole algorithm, is $O(n_0 \log n_0)$.

7. Parallel Algorithm

In this section we will prove that the general strategy from the previous section can be realized on an EREW PRAM in time $O(\log^3 n_0)$ with $O(n_0)$ processors.

Step (2) costs time $O(1)$ because the processor assigned to each edge can check itself whether this edge is reducible or not. To implement (3) compute the graph (R, W), where $(e, e') \in W$ iff e, e' have a common endpoint or some endpoint of e is adjacent to some endpoint of e'. This costs time $O(1)$. And then let I_p be a maximal independent set in this graph. Clearly, I_p satisfies the required condition, and it can be found in time $O(\log^* n_0)$ with $O(n_0)$ processors by the algorithm from [13 or 7]. Steps (4) and (5) both cost time $O(\log n_0)$. Therefore a phase of stage 1 costs time $O(\log n_0)$. But, since the size of G decreases geometrically, the number of iterations is $O(\log n_0)$, and we obtain that the cost of stage 1 is $O(\log^2 n_0)$ with $O(n_0)$ processors.

In stage 2 the steps (6)–(7) are easy to do in time $O(\log n_0)$. Loop (8) is executed $O(\log n_0)$ times, and each execution can be done in time $O(\log n_0)$. So stage 2 costs time $O(\log^3 n_0)$, and the whole algorithm too.

Acknowledgments

We thank M. Yung and the anonymous referee for pointing out several mistakes and many useful comments.

References

1. K. I. Appel and W. Haken, Every planar map is four colorable, *Bull. Amer. Math. Soc.* **82** (1976), 711–712.
2. C. Berge, "Graphs and Hypergraphs," North-Holland, Amsterdam, 1973.
3. J. Boyar and H. Karloff, Coloring planar graphs in parallel, *J. Algorithms* **8** (1987), 470–479.

4. N. CHIBA, T. NISHIZEKI, AND N. SAITO, A linear algorithm for five-coloring a planar graph, *J. Algorithms* **2** (1981), 317–327.
5. M. CHROBAK AND M. YUNG, Fast algorithms for edge-coloring planar graphs, *J. Algorithms* **10** (1989), 35–51.
6. R. COLE AND J. HOPCROFT, On edge coloring bipartite graphs, *SIAM J. Comput.* **11** (1982), 540–546.
7. R. COLE AND U. VISHKIN, Deterministic coin tossing and accelerating cascades: Micro and macro techniques for designing parallel algorithms, *in* "Proceedings 18th ACM STOC, 1986," pp. 206–219.
8. G. N. FREDERICKSON, On linear-time algorithms for five-coloring planar graphs, *Inform. Process. Lett.* **19** (1984), 219–224.
9. S. FIORINI, R. J. WILSON, "Edge-Colourings of Graphs," Pitman, London, 1977.
10. M. R. GAREY, D. S. JOHNSON, AND L. STOCKMAYER, Some simplified NP-complete problems, *Theoret. Comput. Sci.* **1** (1976), 237–267.
11. H. N. GABOW, O. KARIV, Algorithms for edge coloring bipartite graphs and multigraphs, *SIAM J. Comput.* **11** (1982), 117–129.
12. H. N. GABOW, T. NISHIZEKI, O. KARIV, D. LEVEN, AND O. TERADA, "Algorithms for Edge-Coloring Graphs," TR-41/85, Department of Computer Science, Tel Aviv University, 1985.
13. A. V. GOLDBERG, S. A. PLOTKIN, AND G. E. SHANNON, Parallel symmetry breaking in sparse graphs, *in* "Proceedings, 19th Annual ACM Symp. on Theory of Computing, New York, 1987.
14. T. HAGERUP, M. CHROBAK, AND K. DIKS, Optimal parallel 5-colouring of planar graphs, *in* "Proceedings, 14th Int. Conf. on Automata, Languages and Programming, Karslruhe, 1987;" *SIAM J. Comput.*, **18**, No. 2 (1989) 288–300.
15. D. S. HOCHBAUM, T. NISHIZEKI, AND D. B. SHMOYS, A better than "best possible" algorithm to edge color multigraphs, *J. Algorithms* **7** (1986), 79–104.
16. I. J. HOLYER, The NP-completeness of edge coloring, *SIAM J. Comput.* **10** (1981), 718–720.
17. I. J. HOLYER, "The Computational Complexity of Graph Theory Problems," Ph.D. thesis, University of Cambridge, July 1980.
18. H. KARLOFF AND D. SHMOYS, Efficient parallel algorithms for edge coloring problems, *J. Algorithms* **8** (1987), 39–52.
19. M. LUBY, A simple parallel algorithm for the maximal independent set problem, *in* "Proceedings, 17th ACM STOC '85;" *SIAM J. Comput.* **15** (1986), 1036–1053.
20. D. LEVEN AND Z. GALIL, NP-completeness of finding the chromatic index of regular graphs, *J. Algorithms* **4** (1983), 35–44.
21. G. F. LEV, N. PIPPENGER, AND L. G. VALIANT, A fast parallel algorithm for routing in permutation networks, *IEEE Trans. Comput.* **C-30** (1981), 93–100.
22. D. MATULA, Y. SHILOAH, AND R. TARJAN, "Two linear-time algorithms for five-coloring a planar graph," Tech. Rep. No. STAN-CS-80-830, Department of Computer Science, Stanford University.
23. J. NAOR, A fast parallel coloring of planar graphs with five colors, *Inform. Process. Lett.* **25** (1987), 51–53.
24. C. E. SHANNON, A theorem on colouring lines of a network, *J. Math. Phys.* **28** (1949), 148–151.
25. V. G. VIZING, On the estimate of the chromatic class of a p-graph, *Discret. Anal.* **3** (1964), 25–30.
26. V. G. VIZING, Critical graphs with a given chromatic number, *Diskret. Anal.* **5** (1965), 9–17.
27. M. H. WILLIAMS, A linear algorithm for coloring planar graphs with five colors, *Comput. J.* **28**, No. 1 (1985), 78–81.