アルゴリズム 32−12
(1993. 3. 18)

部分k木を辺彩色する線形時間アルゴリズム

周 暁　中野 眞一　西関 隆夫

東北大学工学部情報工学科

部分k木に対しては数多くの組合せ問題が線形時間で解けることが知られている. しかし, 辺彩色問題に対しては効率よいアルゴリズムが得られていなかった. 本論文は与えられた部分k木の辺彩色指数を求め, その指数に等しい色数で辺彩色する線形時間アルゴリズムを与える.

A Linear Algorithm for Edge-Coloring Partial k-Trees

Xiao Zhou, Shin-ichi Nakano and Takao Nishizeki

Department of Information Engineering, Faculty of Engineering
Tohoku University, Sendai 980, Japan

Abstract

Many combinatorial problems can be efficiently solved for partial k-trees. The edge-coloring problem is one of a few combinatorial problems for which no linear time algorithm has been obtained for partial k-trees. This paper gives an algorithm which optimally edge-colors a given partial k-tree in linear time.

1. Introduction

This paper deals with the edge-coloring problem which asks to color, using a minimum number of colors, all edges of a given graph so that no two adjacent edges are colored with the same color. The *chromatic index* $\chi'(G)$ of graph G is the minimum number of colors used by an edge-coloring of G. This problem arises in many applications, including various scheduling and partitioning problems [FW]. Since the edge-coloring problem is NP-complete [Hol], it seems unlikely that there exists a polynomial time algorithm to solve the problem for general graphs. On the other hand, it is known that many combinatorial problems can be solved very efficiently, say in linear time, for series-parallel graphs or partial k-trees [ACPD, AL, BPT, C, TNS]. Such a class of problems has been characterized in terms of "forbidden graphs" or "extended monadic logic of second order" [ACPD, AL, BPT, C, TNS]. However the edge-coloring problem does not belong to such a class, and is indeed one of the "edge-covering problems" which, as mentioned in [BPT], do not appear to be solved efficiently for partial k-trees. Terada and Nishizeki gave an $O(|V|^2)$ algorithm for series-parallel simple graphs, i.e., partial 2-trees [TN]. Zhou, Suzuki and Nishizeki gave a linear algorithm for series-parallel multigraphs [ZSN]. Bodlaender gave an $O(|V|\Delta^{2^{2(k+1)}})$ algorithm for partial k-trees [B] where k is a constant but the maximum degree Δ is not always a constant. In this paper we give a linear algorithm for partial k-trees, which determines the chromatic index $\chi'(G)$ of a given partial k-tree G and actually finds an edge-coloring of G using $\chi'(G)$ colors.

2. Terminology and definitions

In this section we give some definitions. Let $G = (V, E)$ denote a graph with vertex set V and edge set E. We often denote by $V(G)$ and $E(G)$ the vertex set and the edge set of G, respectively. The paper deals with *simple* graphs without multiple edges or self-loops. An edge joining vertices u and v is denoted by (u, v). The class of *k-trees* is defined recursively as follows:

(a) A complete graph with k vertices is a k-tree.

(b) If $G = (V, E)$ is a k-tree and k vertices $v_1, v_2, \cdots, v_k$ induce a complete subgraph of G, then $H = (V \cup \{w\}, E \cup \{(v_i, w) | 1 \le i \le k\})$ is a k-tree where w is a new vertex not contained in G.

(c) All k-trees can be formed with rules (a) and (b).

A graph is a *partial k-tree* if and only if it is a subgraph of a k-tree. Thus partial k-trees are simple graphs. In this paper we assume that k is a constant.

The *degree* of vertex $v \in V(G)$ is denoted by $d(v, G)$ or simply by $d(v)$. The *maximum degree* of G is denoted by $\Delta(G)$ or simply by Δ. For a vertex $v \in V(G)$, denote by $n_\Delta(v)$

the number of vertices which are adjacent to v and have degree $\Delta(G)$. The graph obtained from G by deleting all edges in $E' \subseteq E(G)$ is denoted by $G - E'$. Similarly the graph obtained from G by deleting all vertices in $V' \subseteq V(G)$ is denoted by $G - V'$.

3. Determining the Chromatic Index

By the classical Vizing's theorem, $\chi'(G) = \Delta(G)$ or $\Delta(G) + 1$ for any simple graph G [FW]. In this section we first show that $\chi'(G) = \Delta(G)$ holds for any partial k-tree G with $\Delta(G) \geq 2k$, and then show that the chromatic index $\chi'(G)$ can be determined in linear time for any partial k-tree G.

Hoover [Hoo] has claimed that $\chi'(G) = \Delta(G)$ holds for any partial k-tree G with $\Delta(G) \geq 4k$, but his proof contains a flaw. His "proof" is based on "Theorem 4.5" in [Hoo]: if the chromatic index of a general graph G is $\Delta(G) + 1$ then

$$|E| \geq \frac{|V|\Delta(G)}{4}.$$

However this "Theorem" is incorrect as seen from the following counterexample. Let G be a graph obtained from K_7, a complete graph of seven vertices, by inserting many vertices, say seventy vertices, in an arbitrary edge e of K_7. Then $\Delta(G) = 6$, $|V| = 77$ and $|E| = 91$. Clearly $\chi'(G) = \Delta(G) + 1 = 7$ since $7 \leq \chi'(K_7 - e) \leq \chi'(G)$. However

$$|E| < \frac{|V|\Delta(G)}{4},$$

contrary to the "Theorem." This flaw looks to stem from an incorrect interpretation of a result on "critical graphs," Theorem 13.6 in [FW].

We prove a claim slightly stronger than his: $\chi'(G) = \Delta(G)$ holds for any partial k-tree G with $\Delta(G) \geq 2k$. An edge (u, v) of G is *eliminable* [TN, NC] if the following equations hold:

$d(u) + n_\Delta(v) \leq \Delta$ if $d(u) < \Delta$; and

$n_\Delta(v) = 1$ if $d(u) = \Delta$.

The following lemma is an expression of a classical result on "critical graphs," called "Vizing's adjacency lemma" (see, for example, [FW], [TN] or [NC]).

Lemma 3.1 *If (u, v) is an eliminable edge of a simple graph G and $\chi'(G - (u, v)) \leq \Delta(G)$, then $\chi'(G) = \Delta(G)$.*

For a partition $S_1, S_2, \cdots, S_l$ of $V(G)$, let $U_0 = \phi$ and $U_i = \bigcup_{j=1}^{i} S_j$ for each i,

$1 \le i \le l$. Furthermore for each $v \in S_i$, $1 \le i \le l$, let

$$\begin{aligned} E_b(v,G) &= \{(v,w) \in E \mid w \in U_{i-1}\}, \\ E_f(v,G) &= \{(v,w) \in E \mid w \in V(G) - U_{i-1}\}, \\ d_b(v,G) &= |E_b(v,G)|, \text{ and} \\ d_f(v,G) &= |E_f(v,G)|. \end{aligned}$$

Thus $d(v,G) = d_b(v,G) + d_f(v,G)$. (See Figure 1.) We have the following two lemmas.

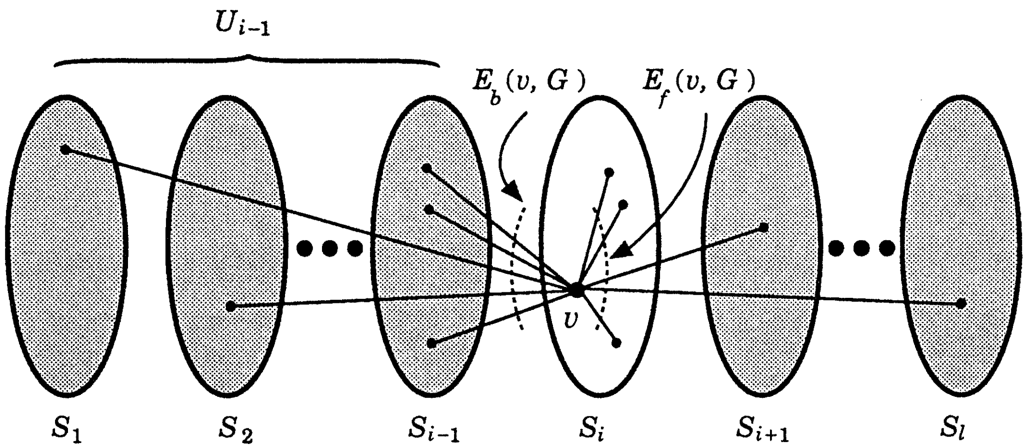

Figure 1. Illustration of notations.

Lemma 3.2 *For any partial k-tree $G = (V,E)$ there exists a partition S_1, S_2, $\cdots$, S_l of V such that $d_f(v,G) \le k$ for every $v \in V$.*

Proof. Since G is a partial k-tree, G has a vertex of degree at most k. Let S_1 be the set of all such vertices, and let $G_1 = G - S_1 = G - U_1$. Since G_1 is also a partial k-tree, G_1 has a vertex of degree at most k. Let S_2 be the set of all such vertices, and let $G_2 = G_1 - S_2 = G - U_2$. By repeating the operation above, one can obtain a required partition S_1, S_2, $\cdots$, S_l of V. $\mathcal{Q.E.D.}$

Lemma 3.3 *If a partial k-tree $G = (V,E)$ has maximum degree $\Delta(G) \ge 2k$, then G has an eliminable edge.*

Proof. Let $S_1 = \{v \in V(G) \mid d(v,G) \le k\}$ and $S_2 = \{v \in V(G - S_1) \mid d(v, G - S_1) \le k\}$. Then $S_1, S_2 \ne \phi$ since $\Delta(G) \ge 2k$. Furthermore there exists an edge joining vertices $u \in S_1$ and $v \in S_2$, because $k+1 \le d(v,G)$ and $d(v, G-S_1) \le k$. Every vertex $w \in S_1$ has degree $d(w,G) \le k < \Delta(G)$, and $d(v, G - S_1) \le k$. Therefore $d(u) \le k < \Delta$, $n_\Delta(v) \le k$, and hence $d(u) + n_\Delta(v) \le 2k \le \Delta$. Thus edge (u,v) is eliminable. $\mathcal{Q.E.D.}$

Using Lemma 3.3, we have the following theorem.

Theorem 3.4 *If a partial k-tree G has maximum degree* $\Delta(G) \geq 2k$, *then* $\chi'(G) = \Delta(G)$.

Proof. By Lemma 3.3 G has an eliminable edge e_1. Since $G-\{e_1\}$ is also a partial k-tree, $G-\{e_1\}$ has an eliminable edge e_2 if $\Delta(G-\{e_1\}) \geq 2k$. Thus there exists a sequence of edges $e_1, e_2, \cdots, e_m$ such that

(a) $\Delta(G') = \Delta(G) - 1$ where $G' = G - \{e_1, e_2, \cdots, e_m\}$; and

(b) e_i, $1 \leq i \leq m$, is eliminable in $G - \{e_1, e_2, \cdots, e_{i-1}\}$.

By the classical Vizing's theorem [FW], $\chi'(G') \leq \Delta(G') + 1 = \Delta(G)$. Therefore, applying Lemma 3.1 repeatedly, we have $\chi'(G) = \Delta(G)$. $\mathcal{Q.E.D.}$

Since $\Delta(G)$ can be computed in linear time, the chromatic index of a partial k-tree G with $\Delta(G) \geq 2k$ can be determined in linear time. On the other hand Bodlaender [B] has given an algorithm which determines $\chi'(G)$ of a partial k-tree G and obtains an edge-coloring of G with $\chi'(G)$ colors total in time $O(|V|\Delta^{2^{2(k+1)}})$. Clearly his algorithm runs in linear time if $\Delta(G) < 2k$. Note that k is a constant. Thus we have the following theorem.

Theorem 3.5 *The chromatic index of a partial k-tree can be determined in linear time if k is a constant.*

4. Obtaining an Edge-Coloring

In Section 3 we have shown that the chromatic index $\chi'(G)$ of a given partial k-tree G can be determined in linear time. In this section we give a linear algorithm which actually obtains an edge-coloring of G with $\chi'(G)$ colors. Using Bodlaender's algorithm [B], one can obtain an edge-coloring of G with $\chi'(G)$ colors in linear time if $\Delta(G)$ is a constant. Therefore it suffices to give a linear algorithm only for the case $\Delta(G) \geq 5k$.

The proofs in the previous section do not yield a linear algorithm for the case $\Delta(G) \geq 5k$, as follows. Lemma 3.3 implies that a partial k-tree G with $\Delta(G) \geq 5k$ necessarily has an eliminable edge. If (u, v) is an eliminable edge in a graph G and an edge-coloring of $G - (u, v)$ with $\Delta(G)$ colors is known, then, using a standard technique of "shifting a fan sequence," one can obtain an edge-coloring of G with $\chi'(G) = \Delta(G)$ ($> 2k$) colors in time $O(|E|)$ [NC, TN]. By Lemma 3.3 there exists an edge-sequence $e_1, e_2, \cdots, e_m$ such that $\Delta(G - \{e_1, e_2, \cdots, e_m\}) = 5k$ and e_i is an eliminable edge in $G - \{e_1, e_2, \cdots, e_{i-1}\}$ for every i, $1 \leq i \leq m$. Using Bodlaender's algorithm, one can obtain an edge-coloring of $G' = G - \{e_1, e_2, \cdots, e_m\}$ with $\chi'(G') = 5k$ ($> 2k$) colors in time $O(|V|)$. Add edges $e_m, e_{m-1}, \cdots, e_2, e_1$ to G' in this order, and modify the edge-coloring of G' to an edge-coloring of G with $\Delta(G)$ colors by repeatedly using the technique of "shifting a fan sequence." Such a repetition of recoloring would require time $O(|V|^2)$.

Our idea is to decompose G into several subgraphs as in the following lemma.

Lemma 4.1 *If a partial k-tree $G = (V, E)$ has maximum degree $\Delta(G) \geq 5k$, then E can be partitioned into subsets $E_1, E_2, \cdots, E_s$ such that the subgraphs G_j, $1 \leq j \leq s$, of G induced by E_j satisfy*

(a) $\Delta(G_j) = 2k$ *for each* j, $1 \leq j \leq s-1$, *and*

(b) $3k \leq \Delta(G_s) = \Delta(G) - 2k(s-1) < 5k$.

Furthermore such a partition of E can be found in time $O(|V|)$.

Since $2k \leq \Delta(G_j) < 5k$ for each j, $1 \leq j \leq s$, by Theorem 3.4 $\chi'(G_j) = \Delta(G_j)$. Using Bodlaender's algorithm, one can obtain an edge-coloring of G_j with $\Delta(G_j)$ colors in time $O(|E_j|)$. Since $\Delta(G) = \sum_{j=1}^{s} \Delta(G_j)$, one can immediately extend these edge-colorings of $G_1, G_2, \cdots, G_s$ to an edge-coloring of G with $\Delta(G)$ colors in linear time.

In order to prove Lemma 4.1, we need the following lemma. Let $V_\Delta(G) = \{v \in V(G) |\ d(v) = \Delta(G)\}$.

Lemma 4.2 *Let $G = (V, E)$ be a partial k-tree, and let $S_1, S_2, \cdots, S_l$ be a partition of V such that $d_f(v, G) \leq k$ for every vertex $v \in V$. Let $I = \{i_1, i_2, \cdots, i_{l'}\}$, $1 \leq i_1 < i_2 < \cdots < i_{l'} \leq l$, and let S'_i, $i \in I$, be a nonempty subset of S_i such that $d_b(v, G) \geq 2k$ for every vertex $v \in S'_i$. Then G has a subgraph G' such that $\Delta(G') = 2k$ and $V_\Delta(G') = \bigcup_{i \in I} S'_i$. Furthermore G' can be found in time $O(|E(G')|)$ if $E_b(v, G)$ for all vertices $v \in V$ are known.*

Proof. A required subgraph G' can be constructed as follows.

1 **Procedure** Subgraph;
2 **begin**
3 let $G' = (\bigcup_{i \in I} S'_i, \phi)$;
4 **for** $j := l'$ **downto** 1 **do**
5 **for** each vertex $v \in S'_{i_j}$ **do**
6 add to G' any $2k - d(v, G')$ edges in $E_b(v, G)$
7 **end**.

Whenever line 6 is going to be executed, $d(v, G') \leq k$ for a current graph G' since $d_f(v, G) \leq k$. Therefore $k \leq 2k - d(v, G') \leq 2k$. Furthermore $d_b(v, G) \geq 2k$, and none of edges in $E_b(v, G)$ has not been added to G' so far. Thus one can always add to G' $2k - d(v, G')$ $(\geq k)$ edges in $E_b(v, G)$ which have not been added to G' so far.

Clearly $d(v, G') = 2k$ holds for the final graph G' if $v \in \bigcup_{i \in I} S'_i$. On the other hand $d(v, G') \leq d_f(v, G) \leq k$ holds if $v \in V(G') - \bigcup_{i \in I} S'_i$. Thus $\Delta(G') = 2k$ and $V_\Delta(G') = \bigcup_{i \in I} S'_i$. Given lists containing $E_b(v, G)$ for all vertices $v \in V$, one can easily execute the procedure above in time $O(|E(G')|)$. *Q.E.D.*

We are now ready to prove Lemma 4.1.

Proof of Lemma 4.1 The following algorithm finds a required decomposition $G_1, G_2, \cdots, G_s$ of G.

1 **Procedure** Subgraphs;

2 **begin**

3 $\Delta := \Delta(G)$;

4 find a partition $S_1, S_2, \cdots, S_l$ of $V(G)$ mentioned in Lemma 3.2 such that $d_f(v, G) \leq k$ for every vertex $v \in V$;

5 **for each** i, $1 \leq i \leq l$, **do** $S_i' := \{v \in S_i | \ d(v, G) \geq 3k\}$; $\{ \ d_b(v, G) \geq 2k$ for every vertex $v \in S_i'$, $1 \leq i \leq l \ \}$

6 $I := \{i | 1 \leq i \leq l$ and $S_i' \neq \phi\}$;

7 $s := \lfloor \frac{\Delta - k}{2k} \rfloor$; $\{ \ 3k \leq \Delta - 2k(s-1) < 5k \ \}$

8 **for** $j := 1$ **to** $s - 1$ **do**

9 **begin** $\{ \ \Delta(G) = \Delta - 2k(j-1) \geq 5k \ \}$

10 find a subgraph G_j of G such that $\Delta(G_j) = 2k$ and $V_\Delta(G_j) = \bigcup_{i \in I} S_i'$; $\{$ Lemma 4.2 $\}$

11 $G := G - E(G_j)$; $\{ \ \Delta(G)$ decreases exactly by $2k \ \}$

12 $S_i' := \{v \in S_i | \ d(v, G) \geq 3k\}$ for all i, $i \in I$; $\{$ update $S_i' \ \}$

13 $I := \{i \in I | \ S_i' \neq \phi\}$ $\{$ update $I \ \}$

14 **end**;

15 $G_s := G$;

16 **return** $G_1, G_2, \cdots, G_s$

17 **end**.

Whenever line 10 is executed for a current graph G, $d_f(v, G) \leq k$ holds for every vertex $v \in V$, and $d_b(v, G) \geq 2k$ holds for every $v \in S_i'$, $i \in I$. Therefore by Lemma 4.2 G has a subgraph G_j such that $\Delta(G_j) = 2k$ and $V_\Delta(G_j) = \bigcup_{i \in I} S_i'$. Since $\Delta(G) \geq 5k$, $\Delta(G)$ decreases exactly by $2k$ whenever line 11 is executed. Thus we have $3k \leq \Delta(G_s) = \Delta - 2k(s-1) < 5k$. Hence the algorithm above correctly finds subgraphs $G_1, G_2, \cdots, G_s$.

We now analyze the time complexity. Lines 4 and 5 can be done in time $O(|E|)$. By Lemma 4.2 line 10 can be done in time $O(|E(G_j)|)$ for every j. Therefore the **for** loop at lines 8–14 can be done total in time $O(\sum_{j=1}^{s-1} |E(G_j)|) \leq O(|E|)$. Since $|E| \leq 2k|V|$ the algorithm above runs in time $O(|V|)$. $\mathcal{Q.E.D.}$

This paper concludes the following theorem.

Theorem 4.3 *The edge-coloring problem can be solved in linear time for partial k-trees if k is a constant.*

Acknowledgment

We would like to thank Dr. Hitoshi Suzuki for various discussions. This research is partly supported by Grant in Aid for Scientific Research of the Ministry of Education, Science, and Culture of Japan under a grant number: General Research (C) 04650300.

References

[ACPD] S. Arnborg, B. Courcelle, A. Proskurowski, and D. Seese, *An algebraic theory of graph reduction,* Tech. Rept. 91-36, Laboratoire Bordelais de Recherche en Informatique, Bordeaux, 1991.

[AL] S. Arnborg and J. Lagergren, *Easy problems for tree-decomposable graphs,* Journal of Algorithms, 12, 2, pp.308-340, 1991.

[B] H. L. Bodlaender, *Polynomial algorithms for graph isomorphism and chromatic index on partial k-trees,* Journal of Algorithms, 11, 4, pp.631-643, 1990.

[BPT] R.B. Borie, R.G. Parker and C.A. Tovey, *Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively constructed graph families,* Algorithmica, 7, pp.555-581, 1992.

[C] B. Courcelle, *The monadic second-order logic of graphs I: Recognizable sets of finite graphs,* Information and Computation, 85, pp.12-75, 1990.

[FW] S. Fiorini and R.J. Wilson, *Edge-Coloring of Graphs,* Pitman, London, 1977.

[Hol] I.J. Holyer, *The NP-completeness of edge-coloring,* SIAM J. on Computing, 10 (1981), pp.718-720.

[Hoo] M. Hoover, *Complexity, structure, and algorithms for edge-partition problems,* Technical Report No. CS90-16, Dept. of Computer Science, The University of New Mexico, 1990.

[NC] T. Nishizeki and N. Chiba, *Planar Graphs: Theory and Algorithms,* North-Holland, Amsterdam, 1988

[TN] O. Terada and T. Nishizeki, *Approximate algorithms for the edge-coloring of graphs,* Trans. Inst. of Electronics and Communication Eng. of Japan, J65-D, 11, pp. 1382-1389, 1982.

[TNS] K. Takamizawa, T. Nishizeki, and N. Saito, *Linear-time computability of combinatorial problems on series-parallel graphs,* J. of ACM, 29, 3, pp. 623-641, 1982.

[ZSN] X. Zhou, H. Suzuki and T. Nishizeki, *Sequential and parallel algorithms for edge-coloring series-parallel multigraphs*, submitted to a journal, 1992.