Unique-Minimum Conflict-Free Coloring for a Chain of Rings

Einollah Pira

The Business Training Center of Tabriz, Iran
pira_ep2006@yahoo.com

Abstract. An optimal algorithm is presented about Conflict-Free Coloring for connected subgraphs of chain of rings. Suppose the length of the chain is $|C|$ and the maximum length of rings is $|R|$. A presented algorithm in [1] for a Chain of rings used $O(\log|C|.\log|R|)$ colors but this algorithm uses $O(\log|C|+\log|R|)$ colors. The coloring earned by this algorithm has the unique-min property, that is, the unique color is also minimum.

Keywords: Conflict-Free Coloring, Chain , Chain of Rings.

1 Introduction

A vertex coloring of graph *G=(V,E)* as an assignment of colors to the vertices such that two adjacent vertices are assigned different colors. A hypergraph *H = (V,E)* is a generalization of a graph for which hyperedges can be arbitrary-sized non-empty subsets of V. A vertex coloring C of hypergraph H is called conflict-free if in every hyperedge there is a vertex whose color is unique among all other colors in the hyperedge.

A *vertex coloring* of a hypergraph such that the minimum (maximum) color of any vertex of a hyperedge is unique (assigned to only one vertex in this hyperedge) is conflict-free and is called *unique-min (resp. unique-max) (confict-free) coloring*. The problems of computing a unique-min coloring is equivalent to computing a unique-max coloring since we can replace every color i by $c_{\max} - i + 1$, where $c_{\max}$ is the maximum color among all vertices [1].

In this paper, first i study unique-min (confict-free) coloring in chain and ring, second, present a new algorithm for a chain of rings.

Conflict-free coloring have various applications. For Example in [2] consider the following scenario: vertices represent base stations of a cellular network interconnected through a backbone. Mobile client connect to the network by radio links and the reception range of each agent is a connected subgraph of the base stations graph. Then it may be desirable that in each agent's range there is a base station transmitting in a unique frequency, in order to avoid interference. The problem of minimizing the number of necessary frequencies is equivalent to Connected Subgraphs Conflict-Free Coloring.

N. Meghanathan et al. (Eds.): CCSIT 2012, Part II, LNICST 85, pp. 319–324, 2012.

Related work. The study of conflict-free coloring was initiated in [2] as a geometric problem with applications to cellular networks. Some of the problems proposed in that paper can be defined as hypergraph conflict-free coloring problems. The algorithm that uses $O(\log^2 n)$ colors (where n is the number of vertices) is given in [1] about CF-coloring for trees and trees of rings. Some of the problems presented in [2] can be defined as hypergraph conflict-free coloring problems. In [3,4] the conflict-free coloring was studied for grids. In [6] the conflict-free coloring of n points with respect to (closed) disks were studied and were proved a lower bound of $\Omega(\log n)$ colors. In [7] the conflict-free coloring of n points with respect to axis-parallel rectangles were studied. Various other conflict-free coloring problems have been considered in very recent papers [8,12,13,14,15,16,17].

The problem becomes more interesting when the vertices are given online by an adversary. For example, at every given time step i, a new vertex v_i is given and the algorithm must assign v_i a color such that the coloring is a conflict-free coloring of the hypergraph that is induced by the vertices $V_i = \{v_1, v_2, ..., v_i\}$. Once v_i is assigned a color, that color cannot be changed in the future. This is an online setting, so the algorithm has no knowledge of how vertices will be given in the future. In [5] there is the online version of conflict-free coloring of a hypergraph. The online version of Connected Subgraphs Conflict-Free Coloring in chains was presented in [8]. Also, in the case of intervals, there are several algorithms [11]. Their randomized algorithm uses $O(\log n \log\log n)$ colors with high probability. Their deterministic algorithm uses $O(\log^2 n)$ colors in the worst case. Recently, randomized algorithms that use $O(\log n)$ colors have been found in [9,10].

2 Preliminaries

The topologies i study during this paper are chain, ring and chain of rings. A graph is a ring when all its vertices *V* are connected in such a way that they form a cycle of length |*V*|. A chain of rings can be defined recursively in the following manner: it is either a single ring or a ring *R* attached to a chain of rings *C* by identifying exactly one vertex of *R* to one vertex of *C*. An Example of a chain of rings is displayed in Fig. 1.

Algorithm for unique-minimum conflict-free coloring in a chain: in [2] there exists an algorithm that uses $\lfloor \log n \rfloor + 1$ colors for chains. The algorithm for a chain *{1,2,…,n}* as follows:

step 1: Color vertex $\left\lceil \frac{n}{2^1} \right\rceil$ with color 1

step 2: Color vertices $\left\lceil \frac{n}{2^2} \right\rceil$, $\left\lceil \frac{n}{2^1} + \frac{n}{2^2} \right\rceil$ with color 2

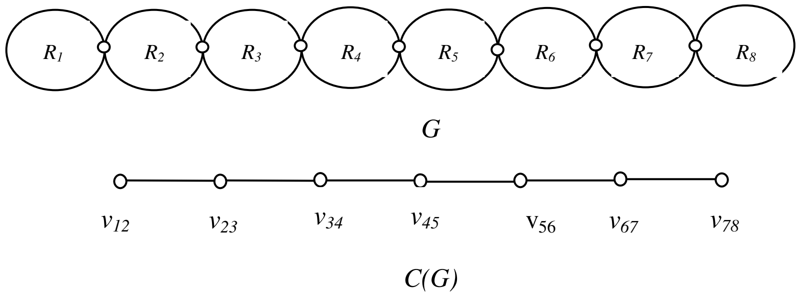

Fig. 1. A chain of rings G and the corresponding chain representation C(G)

step 3: Color vertices $\left\lceil \frac{n}{2^3} \right\rceil, \left\lceil \frac{n}{2^2}+\frac{n}{2^3} \right\rceil, \left\lceil \frac{n}{2^1}+\frac{n}{2^3} \right\rceil, \left\lceil \frac{n}{2^1}+\frac{n}{2^2}+\frac{n}{2^3} \right\rceil$ with color 3

....

step i: Color verices $\left\lceil \frac{n}{2^i} \right\rceil, . \quad . \quad . \quad . , \left\lceil \frac{n}{2^1}+\frac{n}{2^2}+\frac{n}{2^3}+....+\frac{n}{2^i} \right\rceil$ with color i

Color i is used only if $\left\lceil \frac{n}{2^i} \right\rceil = 1$, so in fact $\lfloor \log n \rfloor + 1$ colors are used by the algorithm. For example, if *n=8*, the coloring is 32313234. It is clearly to see that the coloring is unique-minimum conflict-free coloring.

The above algorithm with a small change can be used to solve the unique-minimum conflict-free coloring in a ring. Pick an arbitrary vertex *v* and color it with a color 1 (not to be reused anywhere else in the coloring). The remaining vertices form a chain that color with the algorithm described above. This algorithm colors a ring of n vertices with $\lfloor \log (n-1) \rfloor + 2$ colors. For example, if *n=8*, the coloring is 14342434, where `1' is the first unique color used for *v*. It is not difficult to see that the coloring is conflict-free: All paths that include v are conflict-free colored, and the remaining graph *G–v* is a chain of *n–1* vertices, so paths of *G–v* are also conflict-free colored.

3 An Algorithm for a Chain of Rings

In order to present my algorithm for a chain of rings, i will use the notion of *chain representation* of a chain of rings. Assume a chain of rings *G* by names $R_1, R_2, ..., R_{|C|}$ is $v_{12}, v_{23}, ..., v_{(|C|-1)|C|}$. Let me first describe how to construct such a representation *C(G)* of a chain of rings G: Connect all vertices together that lied in intersection of rings. An Example of a chain of rings and its chain representation is displayed in Fig. 1.

3.1 Analysis of the Algorithm Umccr

Lemma 1. The coloring obtained by Algorithm Umccr is a connected-subgraphs unique-min conflict-free coloring.

Proof. Assume that C is a path in G. There are two cases for C. **Case 1:** C is part of a ring or a ring itself (see Fig. 2). In Fig. 2, if C only contains L_1 (or L_2), C will be colored in a unique-min way because C colored in line 8 from algorithm Umccr. In Fig. 2, if C contains L_1 (or L_2) and vertices u_1, u_2, C will be colored in a unique-min way because the coloring of it start from the max of the colors of the *vertices* u_1 ,u_2 (see lines 5,8 from algorithm Umccr). **Case 2:** C lies on a connected subset of rings, say $R_i,...,R_j$; the corresponding vertices of these rings in *C(G),* say $v_{i(i+1)}...v_{(j-1)j}$. Since these vertices of *C(G)* in line 2 from algorithm Umccr are colored in a unique-min way, and each ring R_k in C lies between vertices $v_{(k-1)k}, v_{k,(k+1)}$ that colored in line 8 from algorithm Umccr, therefore C has been colored in a unique-min way.

Algorithm Umccr. Unique-Min Coloring for a Chain of Rings

Input: a chain of rings G by names $R_1, R_2,..., R_{|C|}$

Output: a coloring of vertices of *G*

1: Construct the chain representation *C(G)* of the chain of rings G.

2: Color chain C(G) with said algorithm in section 2.

3: for i:=1 to *|C|-1* do
    Color the vertex in intersection of the rings
    R_i, R_{i+1} by color vertex $v_{i(i+1)}$ in chain *C(G)*.
  end for
4: for i:=1 to |C| do
5:   set cm:=a max color of the colored vertices of ring R_i .
6:   Delete the colored vertices of ring R_i and connect the neighbors of them.
7:   Let R_i' denote the resulting cycle.
8:   Color cycle R_i' with said algorithm in section 2 by useing colors from $\{cm+1,....,cm+\lfloor \log|R_i'| \rfloor+2\}$.
9: end for

Algorithm Umccr

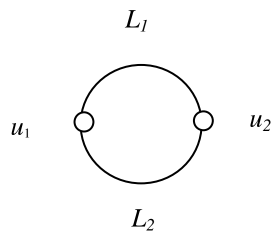

Fig. 2. A ring and the cases for path C: L_1 (or L_2) or L_1 (or L_2) and u_1, u_2

Lemma 2. The Algorithm Umccr uses $O(log|C|+log|R|)$ colors.

Proof. The number of colors for coloring $C(G)$ equal $1+log|C|$. For coloring the rings, in line 5 from algorithm Umccr, the maximum of cm's is $1+log|C|$, therefore the maximum color is used in line 8 are $1+log|C|+2+log|R|$. Thus the Algorithm Umccr uses $O(log|C|+log|R|)$ colors.

4 Conclusions

I have presented an optimal algorithm for coloring a chain of rings such that each connected subgraph has a vertex with a unique minimum color. Also i have proved this algorithm uses $O(log|C|+log|R|)$ colors. An open problem is whether i can achieve a Unique-minimum Conflict-Free Coloring for a tree of rings with $O(log|T|+log|R|)$ colors that $|T|$ is the number of rings.

Acknowledgement. This research has been supported by the business training center of tabriz.

References

1. Georgia, K., Aris, P., Katerina, P.: Conflict-free Coloring for Connected Subgraphs of Trees and Trees of Rings. In: Proc. 11th Panhellenic Conference in Informatics (2007)
2. Even, G., Lotker, Z., Ron, D., Smorodinsky, S.: Conflict-free colorings of simple geometric regions with applications to frequency assignment in cellular networks. SIAM Journal on Computing 33, 94–136; Also in Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS) (2002)
3. Bar-Noy, A., Cheilaris, P., Lampis, M., Zachos, S.: Conflict-free coloring graphs and other related problems (2006) (manuscript)
4. Cheilaris, P., Specker, E., Zachos, S.: Neochromatica (2006) (manuscript)
5. Bar-Noy, A., Cheilaris, P., Smorodinsky, S.: Conflict-free coloring for intervals: from offline to online. In: Proceedings of the 18th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA 2006), Cambridge, Massachusetts, USA, July 30-August 2, pp. 128–137 (2006)

6. Har-Peled, S., Smorodinsky, S.: Conflict-free coloring of points and simple regions in the plane. Discrete and Computational Geometry 34, 47–70 (2005)
7. Pach, J., Toth, G.: Conflict free colorings. In: Discrete and Computational Geometry, The Goodman-Pollack Festschrift, pp. 665–671. Springer, Heidelberg (2003)
8. Alon, N., Smorodinsky, S.: Conflict-free colorings of shallow disks. In: Proceedings of the 22nd Annual Symposium on Computational Geometry (SCG 2006), pp. 41–43. ACM Press, New York (2006)
9. Bar-Noy, A., Cheilaris, P., Smorodinsky, S.: Randomized online conflict-free coloring for hypergraphs (2006) (manuscript)
10. Chen, K.: How to play a coloring game against a color-blind adversary. In: Proceedings of the 22nd Annual ACM Symposium on Computational Geometry (SoCG), pp. 44–51 (2006)
11. Fiat, A., Lev, M., Matousek, J., Mossel, E., Pach, J., Sharir, M., Smorodinsky, S., Wagner, U., Welzl, E.: Online conflict-free coloring for intervals. In: Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 545–554 (2005)
12. Ajwani, D., Elbassioni, K., Govindarajan, S., Ray, S.: Conflict-free coloring for rectangle ranges using $O(n^{.382+\varepsilon})$colors. In: Proc. 19th ACM Symp. on Parallelism in Algorithms and Architectures (SPAA), pp. 181–187 (2007)
13. Bar-Noy, A., Cheilaris, P., Olonetsky, S., Smorodinsky, S.: Online conflict-free coloring for hypergraphs. Combin. Probab. Comput. 19, 493–516 (2010)
14. Chen, K., Kaplan, H., Sharir, M.: Online conflict-free coloring for halfplans, congruent disks, and axis-parallel rectangles. ACM Transactions on Algorithms 5(2), 16:1–16:24 (2009)
15. Lev-Tov, N., Peleg, D.: conflict-free coloring of unit disks. Discrete Appl. Math. 157(7), 1521–1532 (2009)
16. Pach, J., Tardos, G.: Conflict-free colorings of graphs and hypergraphs. Combin. Probab. Comput. 18(5), 819–834 (2009)
17. Smorodinsky, S.: Improved conflict-free colorings of shallow discs (2009) (manuscript)