

Conflict-free coloring with respect to a subset of intervals

Singhal, Vinay 

Meerut Institute of Engineering & Technology, Meerut, India (singhalmca04@gmail.com)

Rastogi, Akanksha

Meerut Institute of Engineering & Technology, Meerut, India (akanksharastogi.mca@gmail.com)



ISSN: 2243-772X
Online ISSN: 2243-7797

Received: 21 August 2013

Revised: 22 October 2013

Accepted: 23 October 2013

Available Online: 29 October 2013

DOI: 10.5861/ijrsc.2013.547

OPEN ACCESS

Abstract

Given a hypergraph $H = (V, E)$, a coloring of its vertices is said to be conflict-free if for every hyperedge $S \in E$ there is at least one vertex in S whose color is distinct from the colors of all other vertices in S . The discrete interval hypergraph H_n is the hypergraph with vertex set $\{1, \dots, n\}$ and hyperedge set the family of all subsets of consecutive integers in $\{1, \dots, n\}$. We provide an algorithm for conflict-free coloring any subhypergraph of H_n . i.e. any subset of possible intervals. If in this hypergraph, the hyperedge have l_n number of vertices then we show that the algorithm uses at most $\lceil l_n/2 \rceil$ colors and we prove that our analysis is tight.

Keywords: frequency-assignment problem; cellular network; conflict-free coloring; intervals; subset of intervals; fully-covered algorithm

Conflict-free coloring with respect to a subset of intervals

1. Introduction

The study of cf-coloring was initiated in the work of Even et al. (2003) and of Smorodinsky (2003) and was extended by numerous other papers (c.f., Alon et al. (2008), Bar-Noy et al. (2010), Chen et al. (2006), Har-Peled et al. (2005), Lev-Tov et al. (2009), Pach et al. (2009)). The study was initially motivated by its **application to frequency assignment for cellular networks**. A cellular network consists of two kinds of nodes: base stations and mobile clients. Base stations have fixed positions, modeled by a finite set of points in the plane, and provide the backbone of the network. Every base station emits at a fixed frequency. If a client wants to establish a link with a base station it has to tune itself to this base station's frequency. Clients, however, can be in the range of many different base stations. To avoid interference, the system must assign frequencies to base stations in the following way: For any closed disk d in the plane (representing the communication range of a client located at the center of this disk), there must be at least one base station which is contained in d and has a frequency that is not used by any other base station contained in d . Since frequencies are limited and costly, a scheme that reuses frequencies, where possible, is desirable.

In section 2, we describe more general definitions of conflict-free colorings for general hypergraphs and intervals. In section 3, we present some previous work on this field by Cheilaris, P., and Smorodinsky S., (2012). In section 4, various applications of conflict-free coloring are presented. Then in section 5, we show the concept of conflict-free coloring with respect to a subset of intervals i.e. subhypergraph of the discrete interval hypergraph and present an algorithm that color the subhypergraph of the discrete interval hypergraph and show that the analysis is tight, i.e., there are subhypergraphs of H_n for which the algorithm computes a conflict-free coloring with twice the optimal (minimum) number of colors. In section 6, we show an example to implement the above algorithm and prove that the algorithm works well & produce valid conflict-free coloring for a given subhypergraph of H_n with at most $n/2$ colors. Finally in section 7, we present some open problems for future work.

2. Conflict free coloring for hypergraphs and intervals:

A hypergraph H is a pair (V, E) , where V is a finite set and E is a family of non-empty subsets of V . We denote by Z^+ the set of positive integers and by N the set of non-negative integers.

Definition 1: Let $H = (V, E)$ be a hypergraph and let C be a coloring $C: V \rightarrow Z^+$: We say that C is a conflict-free coloring (cf-coloring) if for every hyperedge $e \in E$ there exists a color $i \in Z^+$ such that $|e \cap C^{-1}(i)| = 1$. That is, every hyperedge $e \in E$ contains at least one vertex whose color is unique in e .

Here is a more general, formal definition:

Definition 2: Let P be a set of n points in the plane and let R be a family of regions in the plane (e.g., all closed discs). We denote by $H = H_R(P)$ the hypergraph on the set P whose hyperedges are all subsets P' that can be cut off from P by a region in R . That is, all subsets P' such that there exists some region $r \in R$ with $r \cap P = P'$. We refer to such a hypergraph as the hypergraph *induced by P with respect to R* .

Now, consider the hypergraph induced by a set of n collinear points with respect to the family of closed disks in the plane. It is not difficult to see that this hypergraph is isomorphic to the hypergraph induced by a set of n real numbers with respect to the family of closed intervals, which is also isomorphic to the following discrete interval hypergraph.

Definition 3: Let $[n] = \{1, \dots, n\}$. For $s \leq t$, $s, t \in [n]$, we define the (discrete) interval $[s, t] = \{i \in [n] \mid s \leq$

$i \leq t$. The discrete interval hypergraph H_n has vertex set $[n]$ and hyperedge set $I_n = \{[s, t] \mid s \leq t, s, t \in [n]\}$.

It is not difficult to prove that $\lceil \log_2 n \rceil + 1$ colors are necessary and sufficient in order to cf-color H_n [see, e.g., Even et al. (2003)]. An online variation of this cf-coloring problem in which vertices appear one by one and the algorithm has to commit to a color for each point as soon as it appears, maintaining the conflict-free property of the point set at every time, was introduced in Chen et al. (2006) and further studied in Bar-Noy et al. (2008).

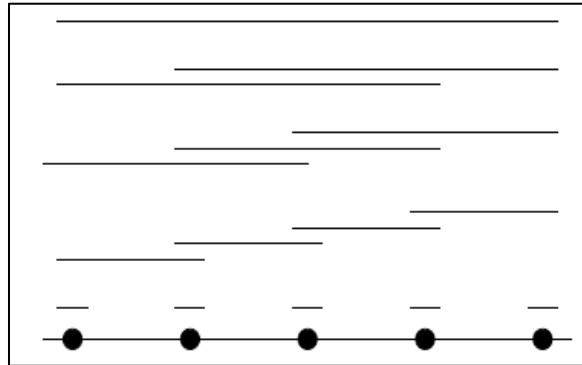


Figure 1. Intervals of five points

To understand the concept of intervals, let us consider an example given below:

For any given set of points $P = \{a_1, a_2, a_3, a_4\}$ we have following intervals-

$$I_1 = \{(a_1, a_2), (a_2, a_3), (a_3, a_4)\}$$

$$I_2 = \{(a_1, a_2), (a_2, a_3), (a_3, a_4)\}$$

$$I_3 = \{(a_1, a_2, a_3), (a_2, a_3, a_4)\}$$

$$I_4 = \{(a_1, a_2, a_3, a_4)\}$$

For all intervals we have to maintain the conflict-free coloring property.

In this paper, we are interested in conflict-free coloring subhypergraphs of H_n of the following form:

$H = ([n], I)$, where $I_{\max} \subseteq I_n$. We define I_{\max} as an interval that contains maximum vertices. Then, H is a hypergraph induced by n points on the real line with respect to a subset of all possible intervals. Conflict-free Colorings of such hypergraphs were studied in the online setting in Bar-Noy et al. (2008). Katz et al. (2007) claim a 4-approximation polynomial time conflict-free coloring for any such hypergraph H (in the offline setting). Studying conflict-free coloring for subhypergraphs of geometric hypergraphs can be justified by applications where only a given subset of the hyperedge set is required to have the conflict-free property. It is also helpful to save energy as well as money.

3. Previous Related Work:

Cheilaris, P., and Smorodinsky S., (2012) provide a polynomial time algorithm for conflict-free coloring any subhypergraph of H_n , they show that the algorithm has approximation ratio 2, and they prove that their analysis is tight. They also show that the problem of deciding whether a given subhypergraph of H_n can be colored with at most k colors has a quasipolynomial time algorithm.

They claim that this variation of conflict-free coloring, with the partial coloring function or the placeholder color '0', is interesting from the point of view of applications. As mentioned in section 1, vertices model base stations in a cellular network. A vertex with no positive color assigned to it can model a situation where a base station is not activated at all, and therefore the base station does not consume energy.

They present the following algorithm to find conflict-free coloring with respect to a subset of Intervals:

Algorithm 1 A hitting set algorithm for conflict-free coloring $H = (V, \mathcal{E})$

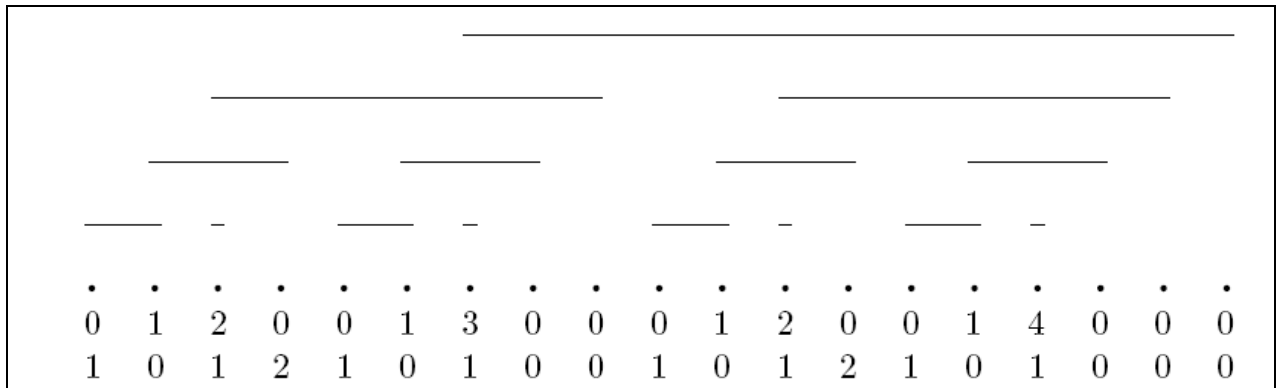
```

 $\ell \leftarrow 0; V^0 \leftarrow V; \mathcal{E}^0 \leftarrow \mathcal{E}$ 
while  $\mathcal{E}^\ell \neq \emptyset$  do
   $S^\ell \leftarrow$  a minimal hitting set for  $(V^\ell, \mathcal{E}^\ell)$ 
  color every  $v \in V^\ell \setminus S^\ell$  with color  $\ell$ 
   $V^{\ell+1} \leftarrow S^\ell$ 
   $\mathcal{E}^{\ell+1} \leftarrow \{e \cap S^\ell \mid e \in \mathcal{E}^\ell \text{ and } |e \cap S^\ell| > 1\}$ 
   $\ell \leftarrow \ell + 1$ 
end while
if  $V^\ell \neq \emptyset$  then color every  $v \in V^\ell$  with color  $\ell$  end if

```

Where minimal hitting set S is computed as follows:

First, we compute a special independent set of intervals $F \subseteq I$ (i.e., in F no two intervals have a common vertex). We compute this independent set F of intervals incrementally. Initially, there is nothing in the independent set. We scan vertices from 1 to n and we include in the independent set the interval $[i, j] \in I$ with minimum j such that $[i, j]$ does not intersect anything already in the independent set. After computing F, for every interval $[i, j] \in F$, we take in S the vertex j (i.e., the maximum or rightmost vertex). Finally, they implement this algorithm to the following figure and calculate the conflict-free coloring.



4. Applications of Conflict-free Coloring:

Application of Conflict-free coloring is very large. Conflict-free coloring arise in the context of frequency assignment to cellular antennae, Distributed Algorithms for Coloring and Domination in Wireless Ad Hoc Networks, in battery consumption aspects of sensor networks, in RFID protocols and several other fields, and has been the focus of many recent research papers. Some applications of Conflict-free Coloring are in the following areas:

4.1 Distributed Algorithms for Coloring and Domination in Wireless Ad Hoc Networks:

Wireless ad hoc networks are composed of a set of mobile nodes which communicate with one another over a shared wireless channel. Unlike wired networks, nodes in an ad hoc network do not rely on a pre-existing communication infrastructure. Instead, they communicate either directly with each other or with the help of intermediate nodes in the network. The distributed, wireless and self-configuring nature of ad hoc networks render them useful for several applications such as mobile battlefields, disaster relief, sensing and monitoring. However, the lack of a fixed communication infrastructure introduces several challenging and interesting research issues in the design of communication protocols for these networks. Any communication protocol for ad hoc networks should also contend with the issue of interference in the wireless medium. When two or more

nodes transmit a message to a common neighbor at the same time, the common node will not receive any of these messages. In such a case, we say that a collision has occurred at the common node.

Coloring and connected domination are two fundamental primitives with several applications in the wireless context. In wireless networks, we seek a conflict-free coloring of the nodes such that two nodes which belong to the same color class may transmit simultaneously without resulting in collisions. Clearly, such a coloring has natural applications to collision-free wireless scheduling. In order to overcome the lack of a fixed routing infrastructure, several researchers have also proposed construction of a virtual backbone in ad hoc networks. A virtual backbone typically consists of a small subset of nodes in the network which gathers and maintains information such as local topology and traffic conditions.

4.2 *Distributed Spectrum Allocation via Local Bargaining:*

Wireless devices are becoming ubiquitous, placing additional stress on the fixed radio spectrum available to all access technologies. To eliminate interference between different wireless technologies, current policies allocate a fixed spectrum slice to each wireless technology. This static assignment prevents devices from efficiently utilizing allocated spectrum, resulting in spectrum holes (no devices in area) and very poor utilization. These results further motivate the *Open Spectrum* approach to spectrum access. Open Spectrum allows unlicensed (secondary) users to coexist with legacy (primary) spectrum holders, thereby “creating” new capacity and commercial value from existing spectrum ranges.

Secondary users opportunistically utilize unused licensed spectrum on a non-interfering or leasing basis based on agreements and constraints imposed by primary users. Open spectrum focuses on controlling the behavior of secondary users while keeping the system transparent to primaries. While maximizing spectrum utilization is the primary goal of dynamic spectrum systems, a good allocation scheme is also needed to provide fairness across users. We hereby use u to represent secondary user. A user seizing spectrum without coordinating with others can cause harmful interference with its surrounding neighbors, and thus reducing available spectrum. Given a fixed topology, existing approaches can efficiently allocate spectrum to users by reducing the problem to a variant of the graph coloring problem. A conflict free spectrum assignment is obtained for the given topology.

In general, a topology-optimized allocation algorithm begins with no prior information, and assigns each user an optimal assignment. In a mobile network, however, users are constantly moving and the network topology changes. Using this global optimization approach, the network needs to completely recompute spectrum assignments for all users after each change, resulting in high computational and communication overhead. This costly operation needs to be repeated frequently to maintain spectrum utilization and fairness.

4.3 *RFID networks:*

Radio Frequency IDentification (RFID) is a technology where a reader device can sense the presence of a close by object by reading a tag device attached to the object. To improve coverage, multiple RFID readers can be deployed in the given region. RFID systems consist of readers and tags. A tag has an ID stored in its memory. The reader is able to read the IDs of the tags in the vicinity by using wireless protocol. In a typical RFID application, tags are attached to objects of interest, and the reader detects presence of an object by using an available mapping of IDs to objects. We focus on passive tags i.e., tags that do not carry a battery. The power needed for passive tags to transmit their IDs to the reader is supplied by the reader itself.

Assume that we are given a set D of readers where each reader is modeled by some disc in the plane. Let P be a set of tags (modeled as points) that lie in the union of the discs in D . Suppose that all readers in D use the same wireless frequency. The goal is to schedule for each reader $d \in D$ a time slot $t(d)$ for which the reader d will be active. That is, at time $t(d)$ reader d would initiate a ‘read’ action. We further assume that a given tag $p \in P$ can be read by reader $d \in D$ at time t if $p \in d$ and d is initiating a ‘read’ action at time t (namely, $t = t(d)$) and

no other reader d' for which $p \in d'$ is active at time t . We say that P is read by our schedule, if for every $p \in P$ there is at least one $d \in D$ and a time t such that p is read by d at time t . Obviously, we would like to minimize the total time slots used in the schedule. Thus our goal is to find a function $t: D \rightarrow \{1, \dots, k\}$ which is conflict-free for the hypergraph $H(D)$. Since we want to minimize the total time slots used, again the question of what is the minimum number of colors that always suffice to CF-color any hypergraph induced by a finite set of n discs is of interest.

4.4 Weighted Coloring based Channel Assignment for WLANs:

With the increased popularity and deployment of WLANs, efficient management of wireless spectrum is becoming increasingly important. Consider an in-building wireless environment in which multiple Access Points (APs) are operational. Each AP operates on an administrator-specified channel. In 802.11 WLANs, the wireless card of a user scans the wireless medium to identify the access point with the strongest signal and associates with it. In order to reduce interference between different APs in the same physical neighborhood administrators conduct detailed Radio Frequency (RF) site surveys, often using spectrum analyzers, prior to setting up APs within the building and assigning specific channels to them. Given the unlicensed nature of WLAN technologies and decreasing costs of APs, the number of APs located in a physical neighborhood has proliferated. In many cases administrators increase the number of APs within a building to improve the wireless coverage. Additionally multiple organizations co-resident in the same building deploy independent wireless LANs, and the channel assignments made for the respective APs are made independent of each other. Hence the assignment of channels to this potentially large set of APs needs to be carefully coordinated, or else the broadcast nature of WLANs can lead to serious performance degradation of the wireless users.

5. Conflict-free coloring for subset of intervals:

In this section, we present an algorithm for conflict-free coloring for subset of intervals. This algorithm colors the vertices in various levels depends on I . In first level, we start coloring of those intervals which contains minimum vertices (most probably two). Then in next level, we color those intervals which contain some more vertices than in previous level. We repeat these steps until all the intervals are considered. It is based on repeatedly computing a fully covered interval set in hypergraphs.

Definition 4: We can define I as a subsets of intervals i.e. $I = \{I_2, I_3, I_4, I_5, \dots, I_n\}$

where in

I_2 contains set of two vertices only

I_3 contains set of three vertices only

.

I_n contains set of n (maximum) vertices.

In the literature, a conflict-free coloring is an assignment of colors (positive integers) to the vertices of the hypergraph. In this work, we introduce and consider a slight variation of conflict-free coloring, in which we allow some vertices to not be assigned colors, as long as in every hyperedge, there exists a vertex with assigned color that is uniquely occurring in the hyperedge. In other words, we allow the coloring function $C: V \rightarrow Z^+$ in definition 1 to be a partial function. Alternatively, we can use a special color '0' given to vertices that are not assigned any positive color and obtain a total function $C: V \rightarrow N$. Then, we arrive at the following variant of definition 1.

Definition 5: Let $H = (V, E)$ be a hypergraph and let $C: V \rightarrow N$. We say that C is a conflict-free coloring if for every hyperedge $S \in E$ there exists a color $i \in Z^+$ such that $|S \cap C^{-1}(i)| = 1$. We denote by $\chi_{cf}(H)$ the minimum integer k for which H admits a cf-coloring with colors in $\{0, \dots, k\}$.

Remark 1: We claim that this variation of conflict-free coloring, with the partial coloring function or the placeholder color '0', is interesting from the point of view of applications. As mentioned in section 1, vertices model base stations in a cellular network. A vertex with no positive color assigned to it can model a situation where a base station is not activated at all, and therefore the base station does not consume energy. One can also think of a bi-criteria optimization problem where a conflict-free assignment of frequencies has to be found with small number of frequencies (in order to conserve the frequency spectrum) and few activated base stations (in order to conserve energy).

A Fully-Covered algorithm for Conflict-free coloring with respect to a subset of intervals

```

Algorithm: CFCforSubInterval
Step 1: Initialize  $I \leftarrow I_i = \{I_1, I_2, \dots, I_n\}$ ;
Step 2: Take first set of elements from  $I_i$ ;
If  $i$  is even
    goto step 3;
Else
    goto step 4;
Step 3: Calculate  $\max = \lfloor i/2 \rfloor$ ;
Check number of uncolored vertices in  $I_i$ , let  $z$ ;
If  $z=0$  {
    Color the middle vertex (i.e.  $(k/2 + 1)^{\text{th}}$  vertex for  $k$  vertices) with  $\max+1$ ;
}
Else {
    Color the uncolored vertices in the following ways
    (i) Firstly, color the middle vertex (i.e.  $(k/2 + 1)^{\text{th}}$  vertex for  $k$  vertices) with  $\max+1$ ;
    (ii) Color other vertices (if any) from left and right of middle vertex in decreasing order of  $\max$  i.e. leftmost
    vertex from middle is colored with  $\max$  then next with  $\max-1$  and so on. Follow the same rule for right side
    vertices from middle vertex;
}
Repeat this step until all sets of  $I_i$  are fully colored;
Remove the visited  $I_i$  from  $I$ ;
Step 4: Calculate  $\max = \lfloor i/2 \rfloor$ ;
Check number of uncolored vertices in  $I_i$ , let  $z$ ;
If  $z=0$  {
    Color the middle vertex with  $\max$ ;
}
Else {
    Color the uncolored vertices in the following ways:
    (i) Firstly, color the middle vertex with  $\max$ ;
    (ii) Color other vertices (if any) from left and right of middle vertex in decreasing order of  $\max$  i.e. leftmost
    vertex from middle is colored with  $\max-1$  then next with  $\max-2$  and so on. Follow the same rule for right side
    vertices from middle vertex;
}
Repeat this step until all sets of  $I_i$  are fully colored;
Remove the visited  $I_i$  from  $I$ ;
Step 5:
If  $I$  is not empty {
    goto step 2;
}
else {
    Color remaining uncolored vertices (if any) with color 0;
END;
}

```

Theorem 1. The conflict-free coloring algorithm for hypergraphs with respect to a subset of intervals is a 2-approximation algorithm.

Proof: It is enough to prove that if some hyperedge (or interval), say ι , reaches iteration with $\ell = k - 1$ of the loop (i.e., the algorithm uses at least k colors), then the input contains as a subset a $J_{\lfloor k/2 \rfloor}$ configuration and moreover this configuration is entirely contained in ι .

We prove it by induction. For $k = 1, 2$, it is true, because there is at least one interval in the input, and

therefore at least one non-zero color is needed in any optimal coloring. For $k > 2$, assume there is a vertex v that gets color k . Then at iteration with $\ell = k - 1$ of the loop there is an interval ι with its rightmost vertex being $v \in S^\ell$ (see figure 2).

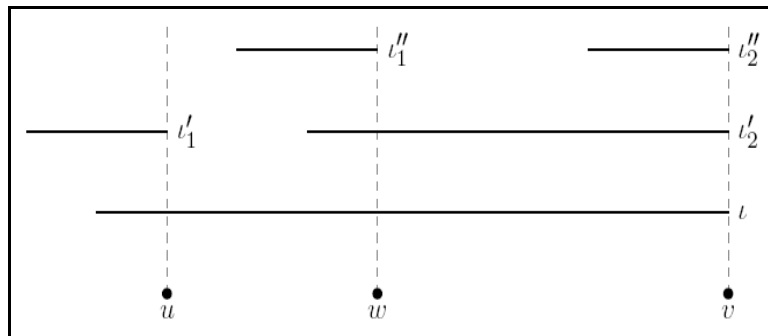


Figure 2. Intervals in an input using k colors

6. An Illustrated Example:

We describe algorithm for conflict-free coloring any hypergraph $H = (V, E)$ that have various level.



Figure 3. An input hypergraph having various subset of interval

Step 1:

- $I_i \leftarrow \{I_2, I_3, I_4, I_5, I_6, I_7\}$
- $I_2 = \{(b, c) (d, e) (h, i)\}$
- $I_3 = \{(a, b, c) (f, g, h)\}$
- $I_4 = \{(a, b, c, d) (g, h, i, j)\}$
- $I_5 = \{(b, c, d, e, f)\}$
- $I_6 = \{\emptyset\}$
- $I_7 = \{a, b, c, d, e, f, g\}$

Step 2:

- Take I_2
- Here 2 is even
- We have to go to step 3

Step 3:

- $\max = 1$
- First set of I_2 is (b, c)
- Here $z = 2$

Assign the color to (b, c)

$b \leftarrow 1$

$c \leftarrow 2$

Second set of I_2 is (d, e)

Here $z = 2$

Assign the color to (d, e)

$d \leftarrow 1$

$e \leftarrow 2$

Third set of I_2 is (h, i)

Here $z = 2$

Assign the color to (h, i)

$h \leftarrow 1$

$i \leftarrow 2$

Now $I_1 \leftarrow \{I_3, I_4, I_5, I_6, I_7\}$

Step 5:

Here I_1 is not empty

We have to go to step 2

Step 2:

Take I_3

Here i is odd

Go to step 4

Step4:

$\max = 2$

First set of I_3 is (a, b, c)

Here $z = 1$

Assign the color to a

$a \leftarrow 2$

Second set of I_3 is (f, g, h)

Here $z = 2$

Assign the color to f, g

$f \leftarrow 1$

$g \leftarrow 2$

Now $I \leftarrow \{I_4, I_5, I_6, I_7\}$

Step 5:

Here I is not empty

We have to go to step 2

Step 2:

Take I_4

Here 4 is even

We have to go to step 3

Step 3:

$\max = 2$

First set of I_4 is (a, b, c, d)

Here $z = 0$

Assign the color to c
 $c \leftarrow 3$
Second set of I_4 is (g, h, i, j)
Here $z = 1$
Assign the color to j
 $j \leftarrow 3$
Now $I \leftarrow \{I_5, I_6, I_7\}$

Step 5:
Here I is not empty
We have to go to step 2

Step 2:
Take I_5
Here i is odd
Go to step 4

Step4:
 $\max = 3$
First set of I_5 is {(b, c, d, e, f)}
Here $z = 0$
Assign the color to d
 $d \leftarrow 3$
Now $I \leftarrow \{I_6, I_7\}$

Step 5:
Here I is not empty
We have to go to step 2

Step 2:
Take I_6
Here 6 is even
We have to go to step 3

Step 3:
 $\max = 3$
First set of I_6 is null
Now $I \leftarrow \{I_7\}$

Step 5:
Here I is not empty
We have to go to step 2

Step 2:
Take I_7
Here i is odd
Go to step 4

Step4:

Max = 4
 First set of I_7 is (a, b, c, d, e, f, g)
 Here $z = 0$
 Assign the color to d
 $d \leftarrow 4$
 Now $I \leftarrow \{\text{null}\}$

Step 5:
 Here I is empty
 END

Final coloring of the vertices is
 $a \leftarrow 2$
 $b \leftarrow 1$
 $c \leftarrow 3$
 $d \leftarrow 4$
 $e \leftarrow 2$
 $f \leftarrow 1$
 $g \leftarrow 2$
 $h \leftarrow 1$
 $i \leftarrow 2$
 $j \leftarrow 3$

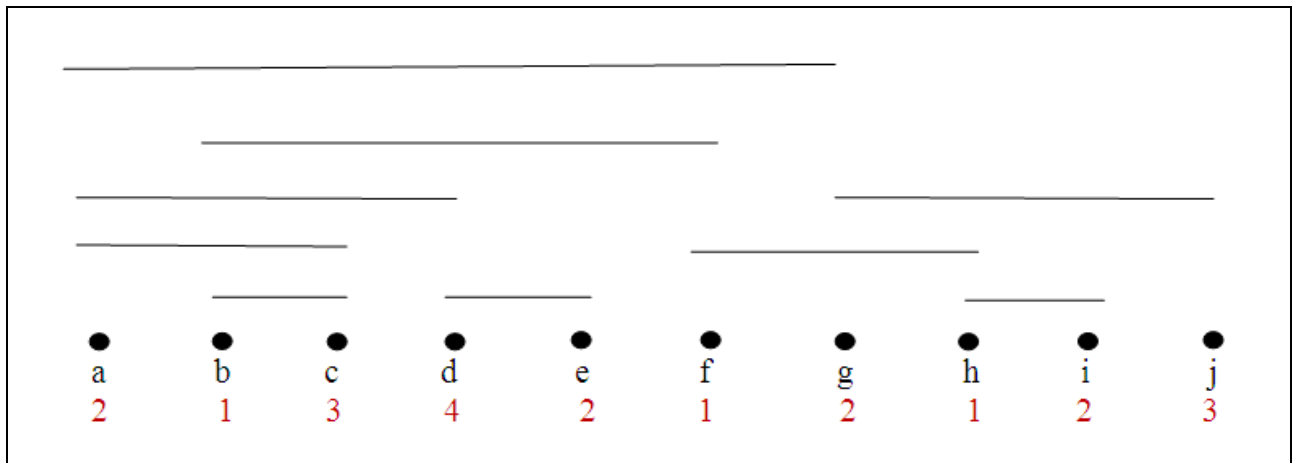


Figure 4. An output hypergraph having various subset of cf-colored interval

Lemma 1: Algorithm 1 terminates.

Proof: At every iteration of the loop, there is some hyperedge $e \in E^i$ for which $|e \cap S^i| = 1$. This follows from the minimality of S^i . Thus, $|E^i| > |E^{i+1}|$. Therefore, the number of hyperedges decreases at every iteration of the loop, and necessarily reaches zero after a finite number of iterations of the loop.

Lemma 2: Algorithm 1 produces a conflict-free coloring.

Proof: We first show that for every hyperedge $e \in E$, there is some i for which $|e \cap S^i| = 1$. Notice that for every iteration $i > 0$, we have $S^{i-1} \supseteq S^i$. If $|e \cap S^0| > 1$, consider the maximum i for which $|e \cap S^i| > 1$. Then, hyperedge $e \cap S^i = e \cap V^{i+1}$ belongs to E^{i+1} and has to be hit by S^{i+1} , i.e., $(e \cap S^i) \cap S^{i+1} = e \cap S^{i+1}$ is non-empty and thus $|e \cap S^{i+1}| = 1$, because of the maximality of i .

Let v be the one element of $e \cap S^i$. Vertex v is colored with some color greater than i by the algorithm

and all other vertices of e are colored with colors which are at most of value t . Thus, e has the conflict-free property.

7. Discussion and open problems:

The exact complexity of computing an optimal CF-coloring for a subhypergraph of the discrete interval hypergraph remains an open problem. One can try to improve the approximation ratio, find a polynomial time approximation scheme, or even find a polynomial time exact algorithm. It would also be interesting to study the complexity of computing optimal conflict-free colorings for subhypergraphs of other geometric hypergraphs, like the hypergraph induced by a set of n points in the plane with respect to a given set of closed disks in the plane. One can also find an optimal algorithm for coloring subintervals in such a way that it must remain conflict-free even some colors fails (frequency spectrum or base station crash).

8. References:

- Alon, N., & Smorodinsky, S. (2008). Conflict-free colorings of shallow discs. *International Journal of Computational Geometry & Applications*, 18(6), 599-604. <http://dx.doi.org/10.1142/S0218195908002775>
- Bar-Noy, A., Cheilaris, P., Olonetsky, S., & Smorodinsky S. (2010). Online conflict-free colouring for hypergraphs. *Combinatorics, Probability and Computing*, 19(4), 493-516. <http://dx.doi.org/10.1017/S0963548309990587>
- Bar-Noy, A., Cheilaris, P., & Smorodinsky S. (2008). Deterministic conflict-free coloring for intervals: From offline to online. *ACM Transactions on Algorithms*, 4(4), 1 - 18. <http://dx.doi.org/10.1145/1383369.1383375>
- Chen, K., Fiat, A., Levy, M., Matoušek, J., Mossel, E., Pach, J., Sharir, M., Smorodinsky, S., Wagner, U., & Welzl, E. (2006). Online conflict-free coloring for intervals. *SIAM Journal Computing*, 36, 545-554.
- Chen, K., Kaplan, H., & Sharir, M. (2009). Online conflict free coloring for halfplanes, congruent disks, and axis-parallel rectangles. *ACM Transactions on Algorithms*, 5(2). <http://dx.doi.org/10.1145/1497290.1497292>
- Even, G., Lotker, Z., Ron, D., & Smorodinsky, S. (2003). Conflict-free colorings of simple geometric regions with applications to frequency assignment in cellular networks. *SIAM Journal of Computing*, 33(1), 94-136. <http://dx.doi.org/10.1137/S0097539702431840>
- Har-Peled, S., & Smorodinsky, S. (2005). Conflict-free coloring of points and simple regions in the plane. *Discrete Computational Geometry*, 34(1), 47-70. <http://dx.doi.org/10.1007/s00454-005-1162-6>
- Katz, M. J., Lev-Tov, N., & Morgenstern, G. (2007). Conflict-free coloring of points on a line with respect to a set of intervals. In *CCCG '07: Proc. 19th Canadian Conference on Computational Geometry*, 45(9), 508-514.
- Lev-Tov, N., & Peleg, D. (2009). Conflict-free coloring of unit disks. *Discrete Applied Mathematics*, 157(7), 1521-1532. <http://dx.doi.org/10.1016/j.dam.2008.09.005>
- Pach, J., & Tardos, G. (2009). Conflict-free colorings of graphs and hypergraphs. *Combinatory & Probability Computing*, 18(5), 819-834. <http://dx.doi.org/10.1017/S0963548309990290>
- Singhal, V. K., & Rastogi, A. (2012). Strong fault-tolerant conflict-free coloring (Strong FTCFCOLORING) for intervals. *International Journal of Research Science & Computing*, 1(2). <http://dx.doi.org/10.5861/ijrsc.2012.137>
- Pach, J., & Toth, G. (2003). Conflict free colorings. In *Discrete and computational geometry*, Springer Verlag, 25, 665-671. http://dx.doi.org/10.1007/978-3-642-55566-4_30
- Smorodinsky, S. (2003). *Combinatorial problems in computational geometry*. Unpublished doctoral dissertation, School of Computer Science, Tel-Aviv University.
- Cheilaris, P., & Smorodinsky S. (2012). Conflict-free coloring with respect to a subset of intervals. Retrieved October 25, 2013, from http://theory.di.uoa.gr/acac12/content/presentations/cheilaris_slcfsubint.pdf