Efficient channel assignment for cellular networks modeled as honeycomb grid

Soumen Nandi[1], Nitish Panigrahy[2], Mohit Agrawal[2], Sasthi C. Ghosh[1], and Sandip Das[1]

[1] Advanced Computing and Microelectronics Unit, Indian Statistical Institute, Kolkata, India, {soumen.nandi_r, sasthi, sandipdas}@isical.ac.in.
[2] Department of Computer Science and Engineering, National Institute of Technology, Rourkela, India, {nitish.pani, mohitag25}@gmail.com.

Abstract. The channel assignment problem with separation is formulated as a vertex coloring problem of a graph $G = (V, E)$ where each vertex represents a base station and two vertices are connected by an edge if their corresponding base stations are interfering to each other. The $L(\delta_1, \delta_2, \cdots, \delta_t)$ coloring of G is a mapping $f : V \rightarrow \{0, 1, \cdots, \lambda\}$ such that $|f(u) - f(v)| \geq \delta_i$ if $d(u, v) = i$, where $d(u, v)$ denotes the distance between vertices u and v in G and $1 \leq i \leq t$. Here λ, the largest color assigned to a vertex of G, is known as the *span*. The same color can be reused in two vertices u and v if $d(u, v) \geq t+1$, where $t+1$ is the *reuse distance*. The objective is to minimize λ over all such coloring function f. Here $(\delta_1, \delta_2, \cdots, \delta_t)$ is called the *separation vector* where $\delta_1, \delta_2, \cdots, \delta_t$ are positive integers with $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_t$. Let λ^* be the minimum span such that there exists an $L(1, 1, \cdots, 1)$ coloring of G. We denote the separation vector $(1, 1, \cdots, 1)$ as (1^t). We deal with the problem of finding the maximum value of δ_1 such that there exists an $L(\delta_1, 1^{t-1})$ coloring with span equal to λ^*. So far bounds on δ_1 have been obtained for $L(\delta_1, 1^{t-1})$ coloring with span λ^* for the square and triangular grids. Shashanka et al. [18] posed the problem as open for the honeycomb grid. We give lower and upper bounds of δ_1 for $L(\delta_1, 1^{t-1})$ coloring with span λ^* of the honeycomb grid. The bounds are asymptotically tight. We also present color assignment algorithms to achieve the lower bound.

1 Introduction

In cellular networks, a large number of base stations is expected to cover a communication region. Such a covering can be achieved by placing the base stations according to a regular plane tessellation. It is well-known that only three different regular tessellations of the plane exist [6]. Specifically, the honeycomb, square and triangular tessellations cover the plane respectively by regular hexagons, squares, and triangles leading to three well-known topologies: honeycomb, square and triangular grids [6]. These three grid structures are shown in Figs. 1 (a), (b) and (c) where each vertex represents a base station and two vertices have an edge between them if their corresponding base stations are interfering to each other. Considering the network cost as a product of degree and diameter the honeycomb grid beats both the triangular and square grids as argued by Bertossi et al. [6]. The brick representation of the honeycomb grid has been shown in Fig. 1 (d) [6]. In this brick representation, the honeycomb grid can be viewed as a 2-dimensional grid. Thus each vertex can be represented by a 2-dimensional cartesian co-ordinate (i, j) where i and j are integers.

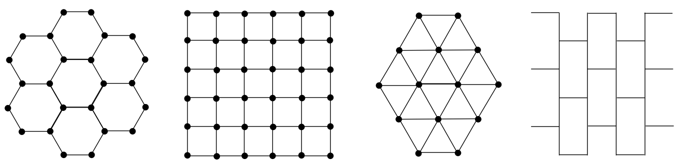

Fig. 1: (a) Honeycomb grid, (b) Square grid, (c) Triangular grid and (d) Brick structure of honeycomb grid.

The assignment of frequency channels to the base stations became a problem for enormous growth of wireless network. Since the number of available frequency channels is very limited, they must be utilized in an efficient manner. The main difficulty in efficient use of these frequency channels is the interference caused by unconstrained simultaneous transmissions of nearby stations. The same frequency channel can be reused by two stations provided that they are sufficiently far away so that the interference arisen between them can be negligible. However, the frequencies assigned to two nearby stations must differ by certain minimum value depending on the distance between them to avoid the channel interference. The channel assignment problem (CAP) deals with the task of assigning frequency channels to the stations such that there is no interference between the frequencies assigned to the nearby stations. The objective is to minimize the required span (bandwidth) where the span is represented by the difference between the least and the highest channel used. The minimum distance at which a channel can be reused with no interference is called the *reuse distance*.

The cellular network is often modeled as a graph $G = (V, E)$ where each vertex represents a base station and there is an edge between two vertices if their corresponding base stations are within the interference range of each other. Thus the channel assignment problem is basically a graph coloring problem on this graph. More formally, the $L(\delta_1, \delta_2, \cdots, \delta_t)$ coloring of a graph $G = (V, E)$ is a way to assign colors in $\{0, 1, \cdots, \lambda\}$ to the vertices of G using as small λ as possible such that the colors assigned to the vertices say u and v which are distance i apart differ by at least δ_i where $1 \leq i \leq t$ and the same color can be reused to two vertices if they are distance $t+1$ or more apart [14]. Here $t+1$ is the reuse distance, λ is the span and $(\delta_1, \delta_2, \cdots, \delta_t)$ is known as the separation vector where $\delta_1, \delta_2, \cdots, \delta_t$ are positive integers with $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_t$. Let λ^* be the minimum span required for any $L(1^t)$ coloring of G. It is evident that minimum span required for any $L(\delta_1, 1^{t-1})$ coloring of G must be greater than or equal to λ^* for any $\delta_1 > 1$. In this paper, we deal with the problem of finding the maximum value of δ_1 such that there exists an $L(\delta_1, 1^{t-1})$ coloring of G using the span equal to λ^*. Keeping the span restricted to the value of λ^* ensures that such an $L(\delta_1, 1^{t-1})$ coloring is always optimal. The rationale behind maximizing δ_1 is as follows. The interference between two adjacent channels can be prevented by using a guard band which is an unused part of the radio spectrum. Interference can also be prevented by using a specific channel separation requirement between two adjacent channels. It has been argued by Bertossi et al. [6] that when no extra colors are used, use of channel separation is always a better option than using guard bands between two adjacent channels. Moreover, higher channel separation between adjacent vertices will give the better quality of communication. So far bounds on δ_1 have been obtained for $L(\delta_1, 1^{t-1})$ coloring with span λ^* for square and triangular grids [5, 18]. Bertossi et al. [6] proposed an algorithm for optimal $L(1^t)$ coloring for the honeycomb grid. Shashanka et al. [18] posed $L(\delta_1, 1^{t-1})$ coloring of honeycomb grid for $\delta_1 > 1$ as an open problem. We found lower and upper bounds of δ_1 for $L(\delta_1, 1^{t-1})$ coloring with span λ^* of the honeycomb grid. We also present color assignment algorithms to achieve the lower bound. The obtained bounds are asymptotically tight.

This paper is arranged in the following way. In section 2, we have described some related works. In section 3, we have presented the basic concepts and notations. In section 4, we have provided the bounds of δ_1 in $L(\delta_1, 1^{t-1})$ coloring of the honeycomb grid. We have also given assignment algorithms to achieve the lower bound in this section. Section 5 concludes the paper.

2 Related work

The $L(1^t)$ coloring has been widely studied by several authors [2, 3, 10, 15, 16] for many special type of graphs. The intractability of optimal $L(1^t)$ coloring, for any positive integer t, has been proved by McCormick [15] for arbitrary graphs. The optimal $L(1^t)$ colorings for rings, square grids, and honeycomb grids have been proposed in [6, 3] and in [1] for trees and interval graphs. The optimal $L(\delta_1, 1^{t-1})$ colorings have been proposed in [5, 18] for rings, square grids, and cellular grids. The optimal $L(\delta_1, \delta_2)$ coloring on square and triangular grids have been proposed [11]. Chang et al. [9] gave bounds for $L(\delta_1, 1)$ coloring of chordal graphs and trees. Griggs and Jin [12, 13] provided optimal $L(\delta_1, 1)$ coloring for buses, rings, wheels, trees, and regular grids where δ_1 is a non-negative real number. The optimal $L(2, 1, 1)$ coloring for square grids [11] and triangular grids, honeycomb grids, and rings [4, 5] have been proposed. The $L(\delta_1, \delta_2, 1)$ coloring for squared and eight-regular grids has been studied in [8]. The $L(2, 1)$ coloring has been investigated in [7, 19] for different graphs.

All these results stated above basically deal with finding minimum span for the concerned coloring of different graphs. Our focus is, however, to find the maximum separation between two colors assigned to two adjacent vertices in an

$L(\delta_1, 1^{t-1})$ coloring of the honeycomb grid. Though the problem has been solved for the triangular and square grids, it was open for the honeycomb grid [18].

3 Basic Concepts and Notations

The required span for $L(\delta_1, 1^{t-1})$ coloring will definitely be greater or equal to the required span for $L(1^t)$ coloring. We now define the *Distance-t clique* of a graph G in order to establish bounds on the required minimum span for $L(1^t)$ coloring of G.

Definition 1. *The Distance-t clique of a graph $G = (V, E)$ is an induced subgraph $G' = (V', E')$ where distance between every pair of vertices in V' is at most t. A maximum Distance-k clique is the Distance-k clique where cardinality of V' is maximum [17]. We denote a maximum Distance-t clique of a graph by D_t.*

The standard graph theoretic term, maximum clique, is a Distance-t clique with $t = 1$. So, Distance-t clique of a graph $G = (V, E)$ is a subgraph of G with *diameter t*. Though finding D_t for general graph is a hard problem, it can be found for the honeycomb grid [6]. As for example, D_1 with 2 vertices, D_3 with 6 vertices, D_5 with 14 vertices, D_0 with 1 vertex, D_2 with 4 vertices and D_4 with 10 vertices of the honeycomb grid are shown in Figs. 2 (a), (b), (c), (d), (e) and (f) respectively.

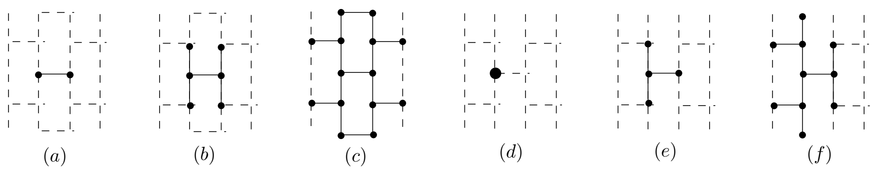

Fig. 2: (a) D_1, (b) D_3, (c) D_5, (d) D_0, (e) D_2 and (f) D_4.

The required span for any $L(1^t)$ coloring of G will be at least the cardinality of D_t. Our objective is to find the maximum value of δ_1 such that there exists an $L(\delta_1, 1^{t-1})$ coloring of G using the set of colors from $\{0, 1, \cdots, \lambda - 1\}$, where λ is the cardinality of D_t. We denote δ_1^{max} as the maximum value of δ_1. Note that δ_1 represents the minimum frequency separation requirement between any two adjacent vertices. In [6], the minimum λ for $L(1^t)$ coloring of D_t in honeycomb grid was computed by considering 8 cases with $t = 8p+q$ where $0 \leq q \leq 7$. Based on the results reported in [6], we can state the following result by considering only 4 cases with $t = 4p + q$ where $0 \leq q \leq 3$.

Result 1 *In honeycomb grid, the minimum λ for any $L(1^t)$ coloring of D_t can be computed as:*

$$\lambda \geq \begin{cases} 6p^2+6p+2, & \textit{if } t=4p+1 \\ 6p^2+12p+6, & \textit{if } t=4p+3 \\ 6p^2+3p+1, & \textit{if } t=4p \\ 6p^2+9p+4, & \textit{if } t=4p+2, \end{cases}$$

where p is a non negative integer.

4 $L(\delta_1^{max}, 1^{t-1})$ coloring of the honeycomb grid

4.1 Upper bound of δ_1^{max}

Let us consider $p = 1$ and $t = 4 \times 1 + 1 = 5$. By putting these values in Result 1, minimum λ is found to be 14. The subgraph D_5 constituted by these 14 vertices is shown in Fig. 2 (c). Observe that in D_5 (Fig. 2 (c)) all the cycles are of even length, i.e., it is a bipartite graph. We see that there are 7 disjoint edges. For a bipartite graph like this each partition contains 7 nodes. We can select seven nodes of one partition and assign colors 0 to 6 to them and for the remaining seven nodes from the other partition, we can assign colors 7 to 13 as shown in Fig. 3. The δ_1^{max} obtained by this assignment is found to be 7. We now state a bound on δ_1^{max} in the following Lemma 1.

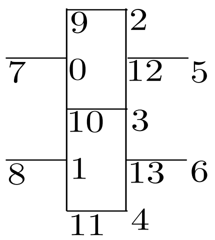

Fig. 3: A coloring of D_5 where $\delta_1^{max} = \frac{\lambda}{2}$.

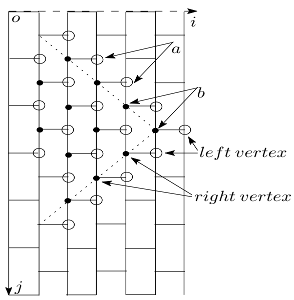

Fig. 4: 18 nodes of label a which are neighbors of 12 nodes of label b.

Lemma 1. *For any $L(\delta_1^{max}, 1^{t-1})$ coloring of D_t of a honeycomb grid, $\delta_1^{max} \leq \frac{\lambda}{2}$, where λ is the cardinality of D_t.*

Proof. For λ number of nodes there can be at most $\frac{\lambda}{2}$ number of disjoint edges in D_t. So we can form a maximum independent set of length at most $\frac{\lambda}{2}$. And our δ_1^{max} therefore, can be at most $\frac{\lambda}{2}$. Thus we can conclude that $\delta_1^{max} \leq \frac{\lambda}{2}$. We can compute λ for D_t by applying Result 1. □

Remark 1. The above result considers the assignment of colors to D_t only. However, as practical networks are very large, we have to repeat the assignment of this subgraph in a regular fashion so as to cover the entire honeycomb grid. If we want to repeat an assignment of D_t in a regular fashion with a view to covering the entire grid, we will see that $\delta_1^{max} = \frac{\lambda}{2}$ may not be achievable. Consider the assignment of D_5 as shown in Fig. 3. It can easily be verified that with this assignment, $\delta_1^{max} = \frac{\lambda}{2} = 7$. Note that here we have considered the assignment of D_5 only but not considered the possibility of repeating this assignment to cover the entire grid. We now consider two different assignments of D_5 and their repetition pattern to cover the entire grid. As for example one of them is shown in Fig. 5 (a). With this repetition pattern, $\delta_1^{max} \neq 7$, rather $\delta_1^{max} = 2$. The other one is shown in Fig. 5 (b), where $\delta_1^{max} = 4$ is achieved. Our objective is to find an assignment of the entire grid for which the value of δ_1^{max} is maximized.

As far as the assignment of D_t is concerned, there are three types of vertices with degrees 1, 2 and 3. Here the degree of a vertex is computed based on the number of adjacent vertices which are already assigned within the subgraph. If we consider the repetition of the assignment of this subgraph to cover the entire grid, the degree of each vertex will eventually become 3. Moreover, the assignment of D_t should be repeated to infinity in such a way that the colors of the adjacent three vertices of a vertex with a particular color remain fixed throughout the entire grid. Such assignment is possible which is given in Fig. 5 (b). As for example, the vertex of color 3 is having the vertices of colors 7, 8 and 10 as its adjacent everywhere. We now have the following result on δ_1^{max} that considers the assignment of the entire honeycomb grid.

Lemma 2. *In $L(\delta_1^{max}, 1^{t-1})$ coloring of honeycomb grid $\delta_1^{max} \leq \frac{\lambda}{2} - 1$, where λ is the cardinality of D_t.*

Proof. From Lemma 1, it follows that the maximum value for δ_1^{max} without repetition is $\frac{\lambda}{2}$. So with repetition, δ_1^{max} cannot exceed this value due to additional constraints on the vertices of degree not equal to 3 which have now become vertices of degree 3 each. Now consider the vertex colored as $\delta_1^{max} - 1$. Because of the color separation δ_1^{max} between two adjacent vertices, the neighbors of the vertex colored with $\delta_1^{max} - 1$, must have a color larger than $\delta_1^{max} - 1$. So its adjacent vertices are of colors with a difference of at least δ_1^{max} which are $2\delta_1^{max} - 1$, $2\delta_1^{max}$ and $2\delta_1^{max} + 1$. But a vertex can have a color at most $\lambda - 1$. So $2\delta_1^{max} + 1 \leq \lambda - 1$, i.e., $\delta_1^{max} \leq \frac{\lambda}{2} - 1$. □

Remark 2. The honeycomb grid is a bipartite graph. We label the vertices of one partition as a and that of other partition as b. The basic idea behind the result stated in Lemma 2 is that there are exactly 3 vertices of label a which are adjacent to a vertex of label b. We observe that there are a minimum of 5 vertices of label a which are adjacent to two vertices of label b. This is because if we choose any two vertices of label b there can be at most one common adjacent (neighbor) vertex of those two b labeled vertices. With a view to generalizing this idea, we required to find the minimum number of vertices of label a which are adjacent to n vertices of label b. Consider $l = 1 + \max \{i \mid \sum_{j=1}^{i} j < n\}$.

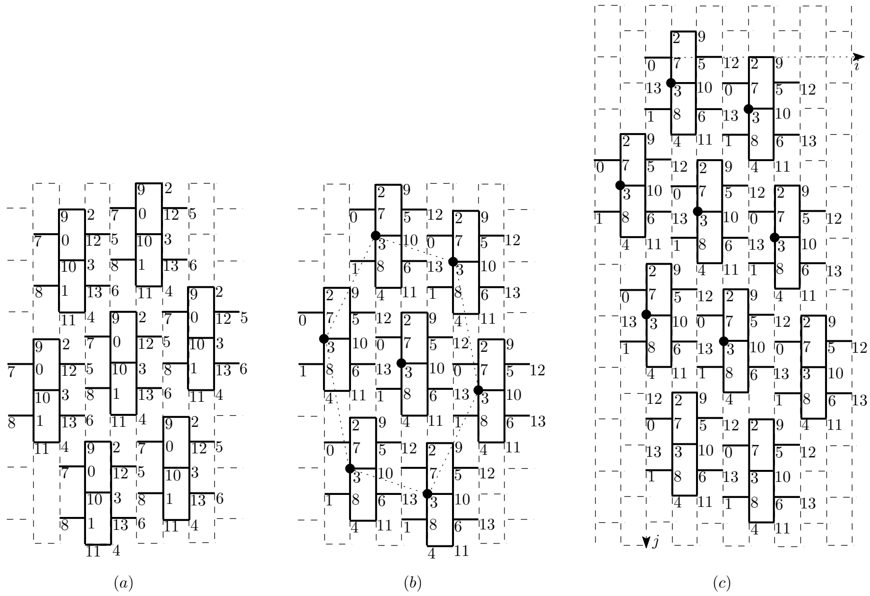

Fig. 5: (a) Coloring of D_5 with repetition where $\delta_1^{max} \leq \frac{\lambda}{2}$ is not achieved, (b) Infinite repetition of D_5 in a honeycomb grid where reuse distance is satisfied and (c) Coloring of D_5 with repetition where $\delta_1^{max} = 4$.

Observe that l depends on n and can be expressed as $l = \lceil \frac{\sqrt{8n+1}-1}{2} \rceil$. We now have the following result.

Lemma 3. *The minimum number of neighbors of n nodes of label b in a honeycomb grid is $n + l + 1$, where no two b labeled vertices are adjacent.*

Proof. Let us consider the honeycomb grid which is shown in Fig. 4. Consider l consecutive columns in that graph. Assume that the number of b labeled vertices in ith column is k_i, $1 \leq i \leq l$, where no two b labeled vertices are adjacent. So, $\sum_{i=1}^{l} k_i = n$. Note that each b labeled vertex has two neighbors within the same column and one neighbor in the $(i+1)$th column. We denote the two neighbors of a b labeled vertex within the same column as the *same column neighbors* and the neighbor in the adjacent column $((i+1))$ as the *adjacent column neighbor*. Observe that the number of same column neighbors of the b labeled vertices of ith column is at least $(k_i + 1)$ and at most $2k_i$. The number of adjacent column neighbors of the b labeled vertices of ith column is exactly k_i. Note, however, that the adjacent column neighbors of the b labeled vertices of ith column may share some same column neighbors of the b labeled vertices of $(i+1)$th column. Observe that the least number of same column neighbors of the b labeled vertices of ith column is $(k_i + 1)$ when all the b labeled vertices are placed one after another at two distance apart. So the total number of same column neighbors of the b labeled vertices across all the l consecutive columns is $\sum_{i=1}^{l}(k_i + 1)$. Note that the k_i number of adjacent column neighbors of the b labeled vertices of ith column may equal or less or greater than the $k_{i+1}+1$ number of same column neighbors of the b labeled vertices of $(i+1)$th column. Let N_i be the number of adjacent column neighbors of the b labeled vertices of ith column. So the total number of adjacent column neighbors of the b labeled vertices across all the l consecutive columns is $\sum_{i=1}^{l} N_i$, where

$$N_i = \begin{cases} k_i, & \text{if } i = l \\ k_i - (k_{i+1}+1), & \text{if } k_i > k_{i+1}+1 \text{ and } 1 \le i \le l-1 \\ 0, & \text{if } k_i \le k_{i+1}+1 \text{ and } 1 \le i \le l-1. \end{cases}$$

So, the total number of neighbors of b labeled vertices across all the l consecutive columns is $\sum_{i=1}^{l}(k_i+1) + \sum_{i=1}^{l} N_i$. If $k_i < k_{i+1}+1$, for all i, then the value of k_l becomes largest and if $k_i > k_{i+1}+1$, for all i, then the total number of adjacent column neighbors will be increased. As our objective is to find the minimum number of neighbors, the best possible situation is

$$k_i = \begin{cases} 1, & \text{if } i = l \\ k_{i+1}+1, & \text{if } 1 \le i \le l-1. \end{cases}$$

So, the minimum value of the above expression is $\sum_{i=1}^{l}(k_i+1) + \sum_{i=1}^{l} N_i = n+l+1$. We conclude that the minimum number of neighbors of n nodes of b labeled vertices in a honeycomb grid is $n+l+1$. □

Example 1. Corresponding to Lemma 3, we are presenting one example for $n = 12$ in Fig. 4, where 12 can be expressed as $2+4+3+2+1$. Here all the 12 nodes of label b are shown by dots and nodes of label a are circled which are the neighbors of b labeled vertices. We see that minimum number of neighbors of 12 nodes of label b is $12+5+1 = 18$.

Theorem 1. *In $L(\delta_1^{max}, 1^{t-1})$ coloring of honeycomb grid, $\delta_1^{max} \le \frac{\lambda}{2} - \lceil \frac{\sqrt{8\lfloor \frac{\lambda}{4} \rfloor + 1} - 1}{2} \rceil - 1$, where λ is the cardinality of D_t.*

Proof. We know that any consecutive δ_1^{max} non negative integers will be forming an independent set. Let us consider such an independent set $S = \{a_1, a_2, \cdots, a_{\delta_1^{max}}\}$ such that all its elements are consecutive and in increasing order. As D_t is a bipartite graph with cardinality λ and λ is even, we can label $\frac{\lambda}{2}$ vertices of it by a label say a and the remaining $\frac{\lambda}{2}$ vertices by an another label say b. Let us denote the vertices colored by $0, 1, \cdots, \frac{\lambda}{2}-1$ as label a vertices and the vertices colored by $\frac{\lambda}{2}, \frac{\lambda}{2}+1, \cdots, \lambda-1$ as label b vertices. We can easily construct an independent set S with $\delta_1^{max}-2$ vertices of label a and 2 vertices of label b. Essentially S contains the a labeled vertices colored by $\frac{\lambda}{2}-(\delta_1^{max}-2), \frac{\lambda}{2}-(\delta_1^{max}-3), \cdots, \frac{\lambda}{2}-1$ and the b labeled vertices colored by $\frac{\lambda}{2}$ and $\frac{\lambda}{2}+1$. Now to form an independent set any one of the δ_1^{max} vertices of set S must not be adjacent to other vertex of S. At minimum there can be 5 vertices which will be adjacent to these 2 vertices of label b. To form an independent set these 5 nodes should not be in the set S. So to choose $\delta_1^{max} - 2$ vertices of label a in the set S we are left with $\frac{\lambda}{2} - 5$ number of choices of a labeled vertices. Hence $\delta_1^{max} - 2 \leq \frac{\lambda}{2} - 5$ i.e. $\delta_1^{max} \leq \frac{\lambda}{2} - 3$.

Similarly, we can construct an independent set S with $\delta_1^{max} - \lfloor \frac{\lambda}{4} \rfloor$ vertices of label a and $\lfloor \frac{\lambda}{4} \rfloor$ vertices of label b. If we take the number of vertices of label b more than $\lfloor \frac{\lambda}{4} \rfloor$, then we can think of switching the labels, i.e., all the a labeled vertices will be changed to label b and all the b labeled vertices will be changed to label a so that same case will again be happened. So $\lfloor \frac{\lambda}{4} \rfloor$ is the threshold value. Hence, by Lemma 3, the minimum number of neighbors of $\lfloor \frac{\lambda}{4} \rfloor$ vertices of label b is $\lfloor \frac{\lambda}{4} \rfloor + \lceil \frac{\sqrt{8\lfloor \frac{\lambda}{4} \rfloor + 1} - 1}{2} \rceil + 1$. Therefore, to choose $\delta_1^{max} - \lfloor \frac{\lambda}{4} \rfloor$ vertices of label a in the set S we are left with $\frac{\lambda}{2} - [\lfloor \frac{\lambda}{4} \rfloor + \lceil \frac{\sqrt{8\lfloor \frac{\lambda}{4} \rfloor + 1} - 1}{2} \rceil + 1]$ vertices of label a. So $\delta_1^{max} - \lfloor \frac{\lambda}{4} \rfloor \leq \frac{\lambda}{2} - [\lfloor \frac{\lambda}{4} \rfloor + \lceil \frac{\sqrt{8\lfloor \frac{\lambda}{4} \rfloor + 1} - 1}{2} \rceil + 1]$. Hence the result. □

4.2 Lower bound of δ_1^{max}

As mentioned earlier, the honeycomb grid can be viewed as a 2-dimensional grid where each vertex can be represented as (i, j) for some integers i and j. We call a vertex (i, j) as a *right vertex* if its adjacent 3 vertices are $(i, j+1)$, $(i, j-1)$ and $(i+1, j)$. Similarly a vertex (i, j) is called a *left vertex* if its adjacent 3 vertices are $(i, j+1)$, $(i, j-1)$ and $(i-1, j)$. Note that a left vertex has a neighbor at left side and a right vertex has a neighbor at right side. In Fig. 4, the vertices marked by circle and dots are left and right vertices respectively. It is easy to verify that all the left and right vertices are forming two separate independent sets. So we can label all the left vertices by a and all the right vertices by b. Let us consider the assignment of the first $\frac{\lambda}{2}$ colors $0, 1, \cdots, \frac{\lambda}{2} - 1$ to the b labeled vertices and the rest $\frac{\lambda}{2}$ colors $\frac{\lambda}{2}, \frac{\lambda}{2} + 1, \cdots, \lambda - 1$ to the a labeled vertices in some fashion. We now describe an assignment scheme for $L(\delta_1^{max}, 1^{t-1})$ coloring of the honeycomb grid, where t is odd and p is any non negative integer.

$L(\delta_1^{max}, 1^{t-1})$ coloring of a honeycomb grid for odd t: We first deal with the $L(\delta_1^{max}, 1^{t-1})$ coloring algorithm for the case of $t = 4p+1$. As our objective is to maximize the value of δ_1^{max}, we assign the colors ranging from 0 to $\frac{\lambda}{2} - 1$ to the right vertices of D_{4p+1} sequentially starting from the left most column, top to bottom, towards the 2nd right most column. Similarly, the colors ranging from $\frac{\lambda}{2}$ to $\lambda - 1$ are assigned to the left vertices of D_{4p+1} sequentially starting from the 2nd left most column, top to bottom, towards the right most column. Though there are $(2p+2)$ columns in D_{4p+1} only the first $(2p+1)$ columns contain vertices of label b. In each column from left most to 2nd right most there are $(p+1)$, $(p+2)$, $\cdots$, $2p$, $(2p+1)$, $2p$, $\cdots$, $(p+2)$, $(p+1)$ number of b labeled vertices respectively. So, the number of colors in column i is denoted by

t_i and can be defined by

$$t_i = \begin{cases} p+i, & \text{if } 1 \leq i \leq p \\ 2p+1, & \text{if } i = (p+1) \\ 3p+2-i, & \text{if } (p+2) \leq i \leq (2p+1). \end{cases}$$

We see that $(t_1 + t_{p+2}) = (t_2 + t_{p+3}) = \cdots = (t_p + t_{2p+1}) = (3p+1)$. An example of execution of this algorithm to the subgraph D_{4p+1} is shown in Fig. 5 (c) for the case where $p = 1$. So far we have considered the color assignment of the vertices of D_{4p+1} only. We now consider the repetition of these colors to cover the entire grid. We use the following repetition pattern to extend the coloring of D_{4p+1} to the entire grid: a color assigned to vertex (i, j) is repeated to exactly six vertices $(i+p+1, j+3p+1)$, $(i+2p+1, j-1)$, $(i+p, j-3p-2)$, $(i-p-1, j-3p-1)$, $(i-2p-1, j+1)$ and $(i-p, j+3p+2)$ forming a hexagon of radius $t+1$ (reuse distance) centered around the vertex (i, j). In Fig. 5 (c), the repetition of color 3 has been explicitly shown by filled circle. The same repetition pattern holds for each color assigned to the vertices of D_{4p+1}. It is evident that this pattern can be repeated infinitely and it satisfies the reuse distance. Using this pattern of repetition, when the assignment of D_{4p+1} is repeated infinitely, we observe that all $\sum_{i=1}^{2p+1} t_i$ colors are placed in each column. The sequence of colors used in each column can be described as follows: The sequence is actually composed of $(2p+1)$ subsequences where kth subsequence starts with color f_k and ends with color $f_k + T_k - 1$, where $1 \leq k \leq 2p+1$. That means, there are T_k many colors in the kth subsequence. Let us define $T_k = t_{\frac{k+1}{2}}$ when k is odd and $T_k = t_{p+\frac{k+2}{2}}$ when k is even. We now define f_k as follows: $f_1 = 0$, $f_2 = \sum_{i=1}^{p+1} t_i = \frac{3p^2+5p+2}{2}$ and in general,

$$f_k = \begin{cases} \sum_{i=0}^{\frac{k-3}{2}} T_{2i+1}, & \text{if } 2p+1 \geq k > 2 \text{ and k is odd} \\ f_k = \frac{3p^2+5p+2}{2} + \sum_{i=1}^{\frac{k-2}{2}} T_{2i}, & \text{if } 2p+1 \geq k > 2 \text{ and k is even.} \end{cases}$$

Thus the sequence of colors used in each column is as follows: $f_1, f_1 + 1, \cdots, f_1 + T_1 - 1$; $f_2, f_2 + 1, \cdots, f_2 + T_2 - 1$; $\cdots$; $f_k, f_k + 1, \cdots, f_k + T_k - 1$; $\cdots$; $f_{2p+1}, f_{2p+1} + 1, \cdots, f_{2p+1} + T_{2p+1} - 1$. If color f_1 is placed on (i, j), then on that same ith column f_1 is again placed at $(i, j + 6p^2 + 6p + 2)$ and on $(i+1)$th column f_1 is placed at $(i+1, j-(6p+3))$. The same pattern holds for all other colors. The coloring of D_5 and its repetition to cover the entire grid has been shown in Fig. 5 (c). In this case, there are 3 subsequences $0, 1$; $5, 6$; and $2, 3, 4$. So, the sequence of colors used in each column are $0, 1, 5, 6, 2, 3, 4$.

So far we have considered the assignment of label b vertices only. We now consider the assignment of a labeled vertices ranging from $\frac{\lambda}{2}$ to $\lambda - 1$. Let color $x \in [0, \frac{\lambda}{2} - 1]$ has been assigned to vertex (i, j). Then color $x + \frac{\lambda}{2} \in [\frac{\lambda}{2}, \lambda - 1]$ will be assigned to vertex $(i+1, j)$. It can now be seen that the infinite honeycomb grid will be filled up by λ colors ranging from 0 to $\lambda - 1$ where $\lambda = 6p^2 + 6p + 2$ and $t = 4p+1$. We observe that this assignment is a *no-hole* assignment meaning every colors ranging from 0 to $\lambda - 1$ has been used. An example of execution of the algorithm to the entire grid for $t = 4p + 1$ is shown in Fig. 5 (c) for the case where $p = 1$. Similarly the coloring scheme for the case of $t = 4p + 3$ can be obtained, which we omitted here due to space restriction.

Theorem 2. *For $L(\delta_1^{max}, 1^{t-1})$ coloring of the honeycomb grid,*

$$\delta_1^{max} \geq \begin{cases} 3p^2 + p, & \text{if } t = 4p + 1 \quad (1) \\ 3p^2 + 3p + 1, & \text{if } t = 4p + 3 \quad (2) \end{cases}$$

where λ is the cardinality of D_t and p is a non-negative integer.

Proof. Consider D_t with odd t. Depending on the value of t there are 2 cases.

Case 1. $t = 4p + 1$**:** We observe that 1st b labeled vertex in D_{4p+1} from the top of pth column is given by $\sum_{i=1}^{p-1} t_i = \frac{3p^2-3p}{2}$. The 1st and 2nd b labeled vertices from the top of $(p+1)$th column are $\sum_{i=1}^{p} t_i = \frac{3p^2+p}{2}$ and $\sum_{i=1}^{p} t_i + 1 = \frac{3p^2+p+2}{2}$ respectively. The 1st a labeled vertex from the top of $(p+1)$th column is $\frac{\lambda}{2} + \frac{3p^2-3p}{2}$, which is adjacent to all three said b labeled vertices. The color gap between the 2nd b labeled vertex of $(p+1)$th column and 1st a labeled vertex of $(p+1)$th column is minimum which is found to be:

$$\delta_1^{max} = \frac{\lambda}{2} + \frac{3p^2 - 3p}{2} - \frac{3p^2 + p + 2}{2} = 3p^2 + p.$$

Case 2. $t = 4p+3$**:** We observe that 1st b labeled vertex in D_{4p+3} of $(p+2)$th column is $\sum_{i=1}^{p+1} t_i = \frac{3p^2+7p+4}{2}$. The 1st a labeled vertex of $(p+1)$th column is $\frac{\lambda}{2} + \sum_{i=1}^{p}(p+i) = \frac{\lambda}{2} + \frac{3p^2+p}{2}$. The color gap between these two adjacent colors is minimum which is found to be:

$$\delta_1^{max} = \frac{\lambda}{2} + \frac{3p^2 + p}{2} - \frac{3p^2 + 7p + 4}{2} = 3p^2 + 3p + 1.$$

□

Observation 1 *For any $L(\delta_1^{max}, 1^{t-1})$ coloring of a honeycomb grid, the lower bound of δ_1^{max} obtained from Theorem 2 and the upper bound of δ_1^{max} obtained from Theorem 1 are asymptotically equal.*

Proof. Observe that the bound of δ_1^{max} obtained from Theorem 1 can be expressed as $3p^2 + cp + d$ where $1.268 \leq c \leq 4.268$ and d is a constant. Hence the result. □

***$L(\delta_1^{max}, 1^{t-1})$* coloring of a honeycomb grid for even *t*:** It follows from Lemmas 2 and 6 of [6] that the $L(\delta_1^{max}, 1^{t-1})$ coloring of D_t for any even t can not be repeated to cover the entire grid using the colors from 0 to $\lambda - 1$, where λ is the cardinality of D_t. It is observed that D_t with even t can be considered as a subgraph of D_{t+1} where $t+1$ is odd. Note that difference between the cardinality of D_{t+1} and D_t is $3p + 1$ when $t = 4p$ and $3p + 2$ when $t = 4p + 2$. Thus by putting these minimum number of extra colors, we can cover the entire grid for the case of even t. So the lower and upper bounds of δ_1^{max} is same as that of the case of odd t. Hence we can conclude the following theorem.

Theorem 3. *For $L(\delta_1^{max}, 1^{t-1})$ coloring of the honeycomb grid,*

$$\delta_1^{max} \geq \begin{cases} 3p^2 + p, & \text{if } t = 4p \quad (3) \\ 3p^2 + 3p + 1, & \text{if } t = 4p + 2 \quad (4) \end{cases}$$

where λ is the cardinality of D_t and p is a non-negative integer.

5 Conclusion

We have derived upper and lower bounds of δ_1^{max} for any $L(\delta_1^{max}, 1^{t-1})$ coloring with span λ^* of honeycomb grid, where λ^* is the minimum span required for any $L(1^t)$ coloring. We have shown that the bounds are asymptotically tight. We have also given assignment algorithms for finding the lower bounds of δ_1^{max}.

References

1. G. Agnarsson, R. Greenlaw, and M. Halldorson. On powers of chordal graphs and their colorings. *Congressus Numerantium*, 144:41–65, 2000.
2. R. Battiti, A. Bertossi, and M. Bonuccelli. Assigning codes in wireless networks: Bounds and scaling properties. *Wireless Networks*, 5:195–209, 1999.
3. A. Bertossi and M. Pinotti. Mappings for conflict-free access of paths in bidimensional arrays, circular lists, and complete trees. *Journal of Parallel and Distributed Computing*, 62:1314–1333, 2002.
4. A. Bertossi, M. Pinotti, and R. Tan. Efficient use of radio spectrum in wireless networks with channel separation between close stations. *DIAL M for Mobility: Intl ACM Workshop on Discrete Algorithms and Methods for Mobile Computing*, Boston, 2000.
5. A. Bertossi, M. Pinotti, and R. Tan. Channel assignment with separation for interference avoidance in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 14:222–235, 2003.
6. A. A. Bertossi, C. M. Pinotti, R. Rizzi, and A. M. Shende. Channel assignment for interference avoidance in honeycomb wireless networks. *Journal of Parallel and Distributed Computing*, 64:1329–1344, 2004.
7. T. Calamoneri. The l(h, k)-labelling problem: An updated survey and annotated bibliography. *Comput. J.*, 54(8):1344–1371, 2011.
8. T. Calamoneri. Optimal $l(\delta_1, \delta_2, 1)$-labeling of eight-regular grids. *Information Processing Letters*, 113:361–364, 2013.
9. G. J. Chang, W. T. Ke, D. Kuo, and R. K. Yeh. $l(d, 1)$-labeling of graphs. *Discrete Math.*, 220:57–66, 2000.
10. I. Chlamtac and S. Pinter. Distributed nodes organizations algorithm for channel access in a multihop dynamic radio network. *IEEE Transactions on Computers*, 36:728–737, 1987.
11. J. V. den Heuvel, R. A. Leese, and M. Shepherd. Graph labelling and radio channel assignment. *Journal of Graph Theory*, 29:263–283, 1998.

12. J. R. Griggs and X. T. Jin. Optimal channel assignments for lattices with conditions at distance two. *5th IEEE International Parallel and Distributed Processing Symposium, extended abstruct*, 2005.
13. J. R. Griggs and X. T. Jin. Real number graph labellings with distance conditions. *SIAM J. Discrete Math.*, 20:302–327, 2006.
14. J. R. Griggs and R. K. Yeh. Labelling graphs with a condition at distance 2. *SIAM J. Discrete Math.*, 5:586–595, 1992.
15. S. McCormick. Optimal approximation of sparse hessians and its equivalence to a graph coloring problem. *Mathematical Programming*, 26:153171, 1983.
16. A. Sen, T. Roxborough, and S. Medidi. Upper and lower bounds of a class of channel assignment problems in cellular networks. *Proc. of the INFOCOM*, 3:1284–1291, 1998.
17. A. Sen, T. Roxborough, and B. P. Sinha. On an optimal algorithm for channel assignment in cellular network. *Proc. of IEEE ICC*, pages 1147–1151, 1999.
18. M. V. S. Shashanka, A. Pati, and A. M. Shende. A characterisation of optimal channel assignments for cellular and square grid wireless networks. *Mobile Networks and Applications*, 10:89–98, 2005.
19. R. Yeh. A survey on labeling graphs with a condition at distance two. *Discrete Mathematics*, 306(12):1217–1231, 2006.