On approximation algorithms for Interference-Aware Broadcast Scheduling in 2D and 3D Wireless Sensor Networks

Ravi Tiwari, Thang N. Dinh, My T. Thai
CISE Department
University of Florida
Gainesville, Fl 32611
Email: {rtiwari,tdinh,mythai}@cise.ufl.edu

***Abstract*—**
Broadcast scheduling is a mechanism for performing interference-aware broadcasting in multi-hop wireless sensor networks (WSNs). Existing studies assume all the WSN nodes lie on a 2D plane. This assumption is not always appropriate, as in practice the sensor nodes may acquire positions in the 3D space. In this paper, we study the broadcast scheduling problem in which we consider two different models of the transmission graph: Disk Graph (DG) in 2D and Ball Graph (BG) in 3D. We consider each node may have different transmission ranges and the interference range is α time of the transmission range (where $\alpha > 1$). We devise efficient coloring methods for coloring a hexagonal tiling in 2D and truncated octahedragonal tiling in 3D, which leads to $O(1)$-approximation ratio for broadcast scheduling problem in 2D and 3D, respectively.

I. Introduction

A Wireless Sensor Network (WSN) is a set of nodes deployed to sense some phenomena, collect information and send it to the base station for further processing on multi-hop paths. The sensor nodes in WSNs communicate via radio transmission. The broadcast nature of the radio transmission is called Wireless Broadcast Advantage (WBA) [27]. This enables a transmitting sensor node to broadcast to all the receiving nodes within its transmission range in a single transmission. However, more than one sensors transmitting simultaneously may result in interference at the receiving sensors. Hence, their transmissions need to be scheduled to avoid interference, such scheduled transmissions are said to be interference-aware transmissions. In WSNs, broadcasting from a source node to all the other nodes in the networks is one of the fundamental operations on which various distributed applications and protocols are based. Broadcast scheduling is a far more effective mechanism in comparison to flooding [19] for mitigating the adverse effect of interference on the performance of broadcasting.

Most of the existing research related to broadcast scheduling in WSNs consider that the sensor nodes are deployed in a 2D plane [7], [8], [10]–[12]. This may be appropriate in some cases, but in general the sensor nodes may acquires locations in a 3D space. For instance, the fire sensors are deployed on trees in the forests at different levels to percept the intensity of forest fire [23]. The underwater sensors reside at different depth of sea in order to sense and collect vital information about aquatic life or to predict the disastrous Tsunami [24]. The WSNs are also deployed on different buildings and bridges at different heights to measure the structural integrity [25], [26]. Furthermore, most of the existing work in 2D [7], [8], [10]–[12], considers all nodes in the network have same transmission range. Only [7] considers different transmission ranges, however, they assume interference range and transmission range are the same, which is not practical.

In this paper, we study the interference-aware broadcast scheduling problem for WSNs in 2D and 3D. We model the WSNs using a Disk Graph (DG) in 2D and a Ball Graph (BG) in 3D. We consider a more realistic network model, in which nodes may have different transmission ranges and their interference range is α times larger than their transmission range (where $\alpha > 1$). This model has not been considered earlier for broadcast scheduling problem in the existing literature.

Since the broadcast scheduling in wireless network is NP-hard [7], we propose an $O(1)$-approximation algorithm for the interference-aware broadcast scheduling problem for WSNs in 2D and 3D. For this, we study two sub problems: 1) Tiling and coloring 2D plane using identical regular hexagons. 2) Tiling and coloring 3D space using truncated octahedrons. The solutions of these two problems lead to the approximation ratio for interference-aware broadcast scheduling problem in 2D and 3D, respectively.

In order to study the tiling and coloring of 2D plane using identical regular hexagons problem, we consider a hexagonal lattice H in the 2D plane. Further, we color all the hexagons in H in such a way that two hexagons h_1 and h_2 having same color are at least a distance $d \in \Re$ apart. The distance d is considered between two closest points p_1 and p_2, such that p_1 is in h_1 and p_2 is in h_2. We optimally solve this problem for any arbitrary distance $d \in \Re$. Notice that a closely related problem to the above problem is studied in [20], where the distance d is considered between the centers of two hexagons. This problem has lot of motivation to channel assignment in cellular network [18], [20], [29], [30], [30]–[34], where the base station is located in the center of the hexagon cell.The solution of this problem cannot be trivially applied to optimally solve the tiling and coloring of 2D plane using hexagonal, as the two problems are not equivalent.

Furthermore, to study the tiling and coloring of 3D space using truncated octahedrons problem, we consider a lattice structure TOC generated by tilling the 3D space using identical truncated octahedrons. All the truncated octahedrons in TOC are colored in such a way, that two truncated octahedrons to_1 and to_2 having same color are least distance $d \in \Re$ apart, the distance d is the distance between two closest points p_1 and p_2 in 3D space, where p_1 is in to_1 and p_2 is in to_2. We provide the lower bound for this problem along with an approximation algorithm in section IV.

The rest of the paper is organized as follows: In section II, we describe the network model and formally define the interference-aware broadcast scheduling problem in WSNs, along with the interference model. We introduce the tilling and coloring of 2D plain using identical regular hexagon in section III. Section IV describes the tiling and coloring of 3D space using identical truncated octahedrons. The O(1)-approximation algorithm for broadcast scheduling in 2D and 3D WSNs, along with theoretical analysis are described in section V. Section VI provides a summary of related work for the broadcast scheduling problem in wireless network and tiling and coloring problem in 2D and 3D. Finally, section VII concludes the paper.

II. Network Model and Problem Definition

A. Network Model

In this paper, we consider each sensor node v_i has a transmission range $r_i^T \in [r_{min}^T, r_{max}^T]$ (where r_{min}^T and r_{max}^T are the minimum and maximum transmission range in the WSN, respectively and $\frac{r_{max}^T}{r_{min}^T} = \beta$) and an interference range $r_i^I = \alpha r_i^T$ ($\alpha > 1$).

1) **The 2D Network Model:** In order to model the WSN in 2D, we consider that each sensor node is represented by a point in 2D plane $\Re^2$. The transmission graph of such a network is represented by a disk graph $G = (V, E)$. Further, each node $v_i \in V$ is associated to two concentric disks D_i^T and D_i^I, centered at v_i, called the transmission disk and the interference disk, with radius r_i^T and r_i^I, respectively. If a sensor node $v_j \in V$ lies within the disk D_i^T associated to node v_i, then there exists an edge (v_i, v_j) in E, which enables v_j to receive any message transmitted by v_i.
2) **The 3D Network Model**: Similar to the 2D network model, each sensor represents a point in 3D space $\Re^3$. The transmission graph of such a 3D network can be modeled as a ball graph $G = (V, E)$. As $r_i^I = \alpha r_i^T$ ($\alpha > 1$), each node $v_i \in V$ is associated to two concentric balls B_i^T and B_i^I, called the transmission ball and the interference ball centered at v_i with radius r_i^T and r_i^I, respectively. If a sensor node $v_j \in V$ lies within the the ball B_i^T of sensor node v_i, then there exists a directed edge (v_i, v_j) in E, which enables v_j to receive any message transmitted by v_i. If v_i and v_j lies within the transmission balls of each other, then there exist a bi-directional edge (v_i, v_j) in E. For a special case when the transmission range of all the sensor nodes in the WSN are same, the ball graph is called a unit ball graph.

Based on our network model, we define our **interference model** as follows:

- If any receiving node v_j is within r_i^T of transmitter v_i, then the energy level of the transmitted signal from v_i is sufficient at v_j to receive and interpret the transmitted data.
- If v_j is outside r_i^T but within r_i^I, then the energy level of the transmitted signal from v_i at v_j is sufficient to interfere its reception from any other node v_k.
- If v_j is outside r_i^I, then the energy level of the transmitted signal from v_i is beyond the perception of v_j.
- The simultaneous transmission of any two nodes v_i and v_j involves interference at some node v_k receiving from v_i, iff $d(v_i, v_k) \leq r_i^T$ and $d(v_j, v_k) \leq r_j^I$.

B. Problem Definition

Interference-Aware Broadcast Scheduling (IABS) problem: Given a WSN modeled as above and a designated sensor node s called the source holding a message m, which is to be broadcasted to all the sensor nodes in the WSN. The problem is to generate an interference-aware broadcast schedule with minimum latency for the transmission of message m from source s to all the other nodes in the network. The interference-aware broadcast schedule must follow the following constraints:

- The source s is scheduled to transmit in the first timeslot t_1.
- A node u is scheduled to transmit in time slot t_j, iff it had already received the message in timeslot t_i, where $i < j$.
- Two nodes u and v are scheduled to transmit in parallel, iff their simultaneous transmission does not interfere the receivers of each other.
- The number of schedules generated i.e the maximum timeslots required for the broadcast to complete, called the broadcast latency should be minimized.

III. Tilling 2D plane using regular hexagons

In this section, we consider the Euclidean plane is divided into regular hexagons of sides $\frac{1}{2}$ to form a hexagonal tilling, as shown in Figure 1. We then, study the problem of coloring these hexagons using minimum number of colors to guarantee any two hexagons h_1 and h_2 with same color are at least some distance $d \in \Re$ apart. The distance d is considered between two closest points p_1 and p_2, such that p_1 is in h_1 and p_2 is in h_2. We define this problem as Distance-d Hexagon coloring problem as follows:

Distance-d Hexagon Coloring problem: *Given a hexagonal lattice H on a 2D plane and a distance $d \in \Re$, find the minimum number of colors needed to color the entire hexagonal lattice, such that two hexagons h_1 and h_2 having same* *color must have the Euclidean distance $distance(h_1, h_2) \geq d$, here the distance $distance(h_1, h_2)$ is measured between two closest points p_1 and p_2, such that, p_1 lies within h_1 and p_2 lies within h_2.*

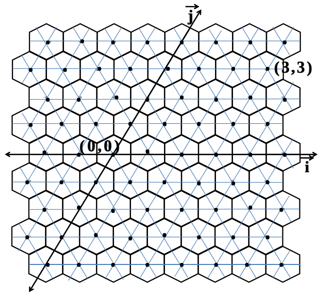

Fig. 1. The new coordinate system in a hexagonal lattice

In order to study the Distance-d Hexagon Coloring problem, we enforce a new coordinate system in the plane, with axes inclined at 60^o. This new coordinates system has two units vectors $\hat{i}(\frac{\sqrt{3}}{2}, 0)$ and $\hat{j}(\frac{\sqrt{3}}{4}, \frac{3}{4})$, as shown in Figure 1. The centers of each hexagon h coincide with the integral coordinates in this coordinate system. Now, each hexagon h can be identified by the coordinates (i, j) of its center as $h(i, j)$. The Euclidean distance between two hexagon centers $h(i_1, j_1)$ and $h(i_2, j_2)$ is given as $\frac{\sqrt{3}}{2}\sqrt{(i_1 - i_2)^2 + (i_1 - i_2)(j_1 - j_2) + (j_1 - j_2)^2}$, hence, the Euclidean distance between a hexagon center $h(i, j)$ from the origin $h(0, 0)$ is given as $\frac{\sqrt{3}}{2}\sqrt{i^2 + ij + j^2}$.

To provide solution for Distance-d Hexagon coloring problem, we first compute the coordinates of hexagon $h(i, j)$ closest to $h(0, 0)$ in the first quadrant, such that two closest points, p_1 in $h(0, 0)$ and p_2 in $h(i, j)$ are at least distance d apart.

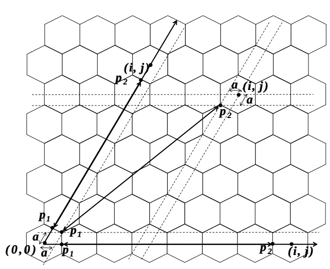

Fig. 2. Transformation between variants of Distance-d hexagon color problems

We observe that the closest points p_1 and p_2 can appears in three ways as shown in Figure 2:

- p_1 is the upper right corner of $h(0, 0)$ and p_2 is the lower left corner of $h(i, j)$, in Figure 2 $(i = 3, j = 5)$.
- p_1 is the mid point of upper right side of $h(0, 0)$ and p_2 is the mid point of lower left side of $h(i, j)$, in Figure 2 $(i = 0, j = 7)$.
- p_1 is the upper right corner of $h(0, 0)$ and p_2 is the middle point of middle right side of (i, j), in Figure 2 $(i = 7, j = 0)$.

Without loss of generality, we considered $i \leq j$, therefore, we get rid of the third case. Now, for the given distance d compute (i, j) as follows:

- First we calculate two pairs (i_1, j_1) and (i_2, j_2) as follows:
  - Compute (i_1, j_1) in first quadrant using the inequality $d^2 \leq \frac{3}{4}((i_1 - 2a)^2 + (j_1 - 2a)^2 + (i_1 - 2a)(j_1 - 2a))$, such that $\sqrt{i_1^2 + j_1^2 + i_1 j_1}$ is minimum among all integral solution of this inequality. Here a is as shown in Figure 2 and is equal to $\frac{1}{3}$.
  - Compute $(i_2 = 0, j_2)$ in first quadrant using the Inequality $d^2 \leq \frac{3}{4}(j_2 - 1)^2$, such that j_2 is minimum among all integral solution of this inequality.
- Finally, if $(i_1^2 + i_1 j_1 + j_1^2) < (i_2^2 + i_2 j_2 + j_2^2)$, we select i, j as i_1, j_1 else we select i, j as i_2, j_2.

This way, we can identify the closest hexagon $h(i, j)$ to the origin $h(0, 0)$ for any given arbitrary distance d, such that two closest points p_1 in $h(0, 0)$ and p_2 in $h(i, j)$ are at least distance d apart. The Euclidean distance between the centers of $h(0, 0)$ and $h((i, j))$ will be $\frac{\sqrt{3}}{2}\sqrt{i^2 + j^2 + ij}$. Subsequently, the distance between the centers of all the **co-color hexagons** (hexagons having the same color) in H will be $\frac{\sqrt{3}}{2}\sqrt{i^2 + j^2 + ij}$. We are now ready to describe the algorithm which optimally finds the co-color hexagons for a given hexagon $h(i', j')$ in H, for any arbitrary distance $d \in \Re$. The algorithm is illustrated as Algorithm 1. The Figure 3 shows an example for distance $d = \frac{\sqrt{31}}{2}$ and $i = 2,\ j = 3$.

Algorithm 1 Co-color hexagon algorithm $(H, d, h(i', j')$,$c)$

```
Input: The hexagonal lattice H, distance d, hexagon
  h(i', j') and a color number c assigned to h(i', j').
Output: Set of co-color hexagons of h(i', j')
Compute i, j
A ← φ;
Queue ← h(i', j');
while (Queue is not empty) do
  H(a, b) ← Queue.Remove()
  A ← H(a, b);
  Color(H(a, b)) ← c
  Insert each of the following hexagons in the Queue if
  they are not inserted in the queue:
      H(a + i, b + j)
      H(a + (i + j), b − i)
      H(a + j, b − (i + j)j)
      H(a − i, b − j)
      H(a − (i + j), b + i)
      H(a − j, j + (i + j))
end while
Return A;
```

Lemma 1: : The Algorithm 1 optimally identifies the co-color hexagons for a given distance $d \in \Re$.

*P*roof: The result of Algorithm 1 is a set of hexagons S, which forms a triangular lattice, as we can see in Figure 3. Now, the problem to find the co-color hexagons is a maximization problem, with a constraint that any two hexagon centers are at least at distance $\frac{\sqrt{3}}{2}\sqrt{(i^2 + j^2 + ij)}$ in the hexagonal lattice.

So the argument is that, if we have a 2D area and we have to fill this area with points in 2D plane with a constraint that any two points should be at least distance $\frac{\sqrt{3}}{2}\sqrt{(i^2+j^2+ij)}$ apart. If we use Algorithm 1, then starting from the center point we can arrange the points in a triangular lattice of equilateral triangles of side $\frac{\sqrt{3}}{2}\sqrt{(i^2+j^2+ij)}$. And the number of points that can be placed is proportional to the number of such non-overlapping equilateral triangles that can fit on this area.

Now, lets consider another method which provides a better solution than Algorithm 1. Then the possible non-overlapping triangles formed by placing the points in 2D plane using this new method , will have have at least one triangle T, having at least one side greater than $\frac{\sqrt{3}}{2}\sqrt{(i^2+j^2+ij)}$ and rest two sides may be greater that or equal to $\frac{\sqrt{3}}{2}\sqrt{(i^2+j^2+ij)}$. Otherwise, the solution of the second method will be same as that of Algorithm 1. It is obvious, that the area of triangle T will be greater than the area of equilateral triangle in the triangular lattice produced by Algorithm 1. Hence, the number of non-overlapping triangles in the arrangement of points produced by the second method will be smaller than produced by Algorithm 1, which in turn shows that using Algorithm 1 we can place more point, which is a contradiction.

□

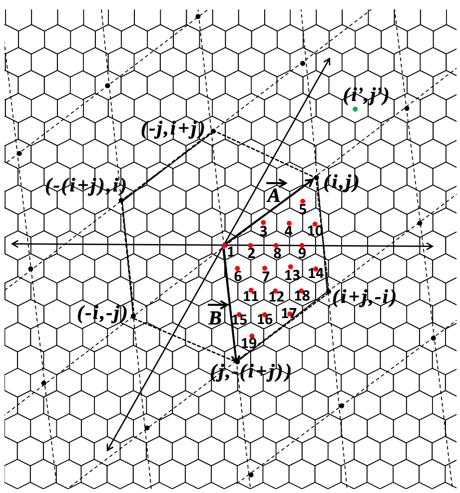

Fig. 3. An instance for Co-Color Hexagon Algorithm for $i=2$ and $j=3$

As shown in Figure 3, notice that the centers of the co-color hexagons for a given hexagon (considering it as origin $h(0,0)$ with out loss of generality) forms a rhombic Sub-Lattice S with basis vectors $\bar{A}=i\hat{i}+j\hat{j}$ and $\bar{B}=j\hat{i}-(i+j)\hat{j}$. Now, the number of colors needed to color H is equal to the number of rhombic sub-lattice similar to S that can be identified in H. This number is equivalent to the index of classes of S in H, which is equal to $|det(\bar{A}.\bar{B})|=i^2+j^2+ij$ [21]. This is also the number of integral points of lattice H within the basic rhombic cell (excluding the lower side and the right side) of S formed by its generator (basis) vectors $\bar{A}$ and $\bar{B}$. These integral points are shown as red color points in Figure 3, lets refer this set of points as X. The set of hexagons centered at these integral points are assigned a unique color number among the points in X, ranging from 1 to $|X|=i^2+j^2+ij$. Further, these hexagons collectively forms a basic pattern which can be repeatedly used to color the entire hexagonal lattice, repeating the color numbers assigned to these hexagons. The coordinates of these integral points can be identified in $O(d^2)$ time, for any given d. Now in order to color the entire 2D hexagonal lattice, the Algorithm 1 can be repeatedly run on each of these integral points in X, assigning its color number to all the co-color hexagons. This will result in the coloring of the entire hexagonal lattice using i^2+j^2+ij colors.

Theorem 1: The coloring generated by the above method is an optimal solution to the Distance-d Hexagon Center Coloring problem.

*P*roof: In our coloring method, we first compute the pair (i,j), such that if the closest distance between two hexagons is at least d, then the distance between their centers is at least $\frac{\sqrt{3}}{2}\sqrt{(}i^2+j^2+ij)$. Further, as proved in Lemma 1, the Algorithm 1 optimally identifies the co-color hexagons, having distance between their centers at least $\frac{\sqrt{3}}{2}\sqrt{(}i^2+j^2+ij)$. Observe in Figure 3, that the co-color hexagons form a rhombic sub-lattice S within the hexagonal lattice H. And the number of possible classes i.e. the index of sub-lattice S in H, which is the number of disjoint sub-lattice similar to S in H is given as $|det(\bar{A},\bar{B})|=i^2+j^2+ij$ [21]. Thus, there can be $\{S_1,S_2\ldots S_{(i^2+j^2+ij)}\}$ disjoint sub-lattices in H. Now if we assign each of them a unique color, then all the hexagons in H will be colored with i^2+j^2+ij colors and the closest distance between two co-color hexagons will be at least d. Hence, coloring method provides the optimal solution to Distance-d Hexagon Coloring problem. □

For a given distance d the hexagonal lattice H is colored in accordance to the Distance-d Hexagon Coloring problem using the algorithm mentioned above. We now discuss a method for identifying the color of any arbitrarily given hexagon $h(i',j')$ in the hexagonal lattice H.

The hexagon $h(i',j')$ can be represented as a vector $\bar{h}=i'\hat{i}+j'\hat{j}$ in the hexagonal lattice H. Further, the basis vectors of sub-lattice S are be represented as vectors $\bar{A}=i\hat{i}+j\hat{j}$ and $\bar{B}=j\hat{i}-(i+j)\hat{j}$. The unit vectors $\hat{i}$ and $\hat{j}$ can be expressed in terms vector $\bar{A}$ and $\bar{B}$ as $\hat{i}=a_1\bar{A}+b_1\bar{B}$ and $\hat{j}=a_2\bar{A}+b_2\bar{B}$, respectively. Now, the vector $\bar{h}$ can be expressed in terms of $\bar{A}$ and $\bar{B}$ as $\bar{h}=i'(a_1\bar{A}+b_1\bar{B})+j'(a_2\bar{A}+b_2\bar{B})=(i'a_1+j'a_2)\bar{A}+(i'b_1+j'b_2)\bar{B}$. Now, if we consider the coordinates $(frac(i'a_1+j'a_2),frac(i'b_1+j'b_2))$ in the sub-lattice S (where $frac(i'a_1+j'a_2)$ and $frac(i'b_1+j'b_2)$ are the fractional part of $(i'a_1+j'a_2)$ and $(i'b_1+j'b_2)$, respectively) then this will be the mapping of $\bar{h}$ in the basic cell of S shown in Figure 3. Now if we convert the coordinates $(frac(i'a_1+j'a_2),frac(i'b_1+j'b_2))$ to the coordinate of the hexagonal lattice H, then we will get the coordinate of one of the red hexagon center (see Figure 3) in basis cell of S. We assign the color number associated to this to hexagon to $h(i',j')$. As an example in Figure 3, an arbitrary hexagon $h(i',j')$ is shown as a green color point, it maps to the red point with color number 12.

Theorem 2: The number of colors for Distance-d Hexagon coloring problem for an arbitrary d is bounded by $\frac{4}{3}d^2+\frac{8}{3}d+\frac{4}{3}$

*P*roof: Consider the hexagon $h(0,0)$, one of its closest co-color neighbors in first quadrant is $h(i,j)$. Without loss of generality, we consider $(j\geq i)$, so now we have two cases:

1) $i = 0, j > 0$ and 2) $i > 0, j > 0$.

Case 1: We use j^2 colors to obtain the distance $d \leq (j-1)\frac{\sqrt{3}}{2}$. Therefore, the number of color is equal to $\frac{4}{3}d^2 + \frac{4}{\sqrt{3}}d + 1 < \frac{4}{3}d^2 + \frac{8}{3}d + \frac{4}{3}$

Case 2: As shown in the proof of Theorem 1, the number of colors used is $i^2 + ij + j^2$ and the guaranteed minimum distance between the closest points of two co-color hexagons is as follows:

$$
\begin{aligned}
d &\leq \sqrt{\frac{3}{4}\left[(i-\frac{2}{3})^2 + (i-\frac{2}{3})(j-\frac{2}{3}) + (j-\frac{2}{3})^2\right]} \\
&= \sqrt{\frac{3}{4}(i^2 + ij + j^2 + \frac{4}{3} - 2(i+j))}
\end{aligned}
$$

It is easy to prove that $i^2 + ij + j^2 \leq \frac{4}{3}d^2 + \frac{8}{3}d + \frac{4}{3}$ for $i, j > 0$. The equality happens iff $i = j$.

□

IV. Tiling and Coloring of the 3D Space

There are basically five 3D space filling primary parallelohedras: cube, hexagonal prism, elongated dodecahedron, rhombic dodecahedron, and truncated octahedron [28]. In this paper, we study the uniform tilling and coloring of 3D space using identical truncated octahedrons. We consider each truncated octahedron has the maximum distance 1 unit within it. Such a truncated octahedron has the side length $\frac{1}{2\sqrt{3}}$ and the distance between its two parallel hexagonal faces is $\frac{1}{2}$, whereas, the distance between its two parallel square faces is $\sqrt{\frac{2}{3}}$. Figure 4 shows the tilling of 3D space using truncated octahedrons. Furthermore, we color the entire 3D space tiling by coloring all the truncated octahedrons in such a way that no two truncated octahedrons can have same color if the distance between the them is less than $d \in \Re^+$. The distance d between two truncated octahedrons to_1 and to_2 is defined as the Cartesian distance between any two closest points p_1 and p_2 in 3D space, such that p_1 is within to_1 and p_2 is within to_2. We formulate the problem of coloring the 3D space tiling as Distance-d Truncated Octahedron Coloring problem. In addition to this, we propose a theoretical lower bound to this problem in Lemma 2 along with an approximation algorithm illustrated as Algorithm 2.

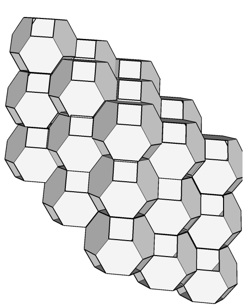

Fig. 4. The tilling of space using truncated octahedrons

Distance-d truncated octahedron coloring problem: *Assume that the entire 3D space is tilled using truncated octahedrons of sides $\frac{1}{2\sqrt{3}}$. Find the minimum number of colors needed to color all the truncated octahedrons in the 3D space, such that any two truncated octahedrons to_1 and to_2 with the same color are at least distance $d \in R^+$ apart. Here the distance considered between to_1 and to_2, represented as $distance(to_1, to_2)$ is the Cartesian distance between two closest points p_1 and p_2 in 3D space, such that p_1 is within to_1 and p_2 is within to_2.*

In order to study the Distance-d truncated octahedron coloring problem, we introduce a new coordinate system in 3D space. In this new coordinate system, the X, Y, and Z axis are inclined as shown in Figure 5. The angle between X and Y axis is $\theta_1 = 2cos^{-1}(\sqrt{\frac{2}{3}})$, whereas, the angle between X and Z axis and Y and Z axis is $\theta_2 = sin^{-1}(\sqrt{\frac{2}{3}})$. Further, the distance along the X, Y and Z axis are the multiple of vectors $\vec{i}$ $(\frac{1}{2}, 0, 0)$, $\vec{j}$ $(\frac{1}{2}cos\theta_1, \frac{1}{2}sin\theta_1, 0)$, and $\vec{k}$ $(\sqrt{\frac{2}{3}}cos\theta_2, 0, \sqrt{\frac{2}{3}}sin\theta_2)$, respectively. The centers of each truncated octahedron in the 3D space coincides with the integral coordinates in the new coordinate system. Hence, every truncated octahedron can be identified by the coordinates (i, j, k) of its center as $to(i, j, k)$.

Observe that the Distance-d truncated octahedron coloring problem can be considered as a special case of graph coloring problem. We define a graph $G_d = (TOC, E_d)$, such that TOC represents the set of vertices representing the truncated octahedrons in 3D space tiling. And the set E_d is a set of edges $E_d = \{(to_1, to_2)|distance(to_1, to_2) < d\}$. Consequently, the chromatic number $\chi(G_d)$ is the solution to the Distance-d truncated octahedron coloring problem i.e. the minimum number of colors needed to color all the truncated octahedrons in the 3D space tiling.

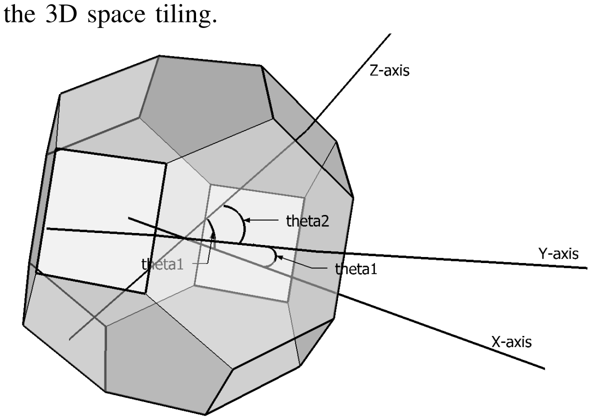

Fig. 5. Modified Co-ordinate System

Lemma 2: For any $d \in R^+$, the chromatic number $\chi(G_d)$ of the graph $G_d = (TOC, E_d)$ is at least $\frac{\pi}{2}\sqrt{\frac{3}{2}}(d + \frac{1}{2})^3$.

*P*roof. For any graph $G = (V, E)$, the relationship between the clique number $\omega(G)$, i.e. number of vertices in the largest clique of G and the chromatic number $\chi(G)$ is given as $\chi(G) \geq \omega(G)$. Hence, we can prove the theorem by showing $\omega(G_d)$ is at least $\frac{\pi}{2}\sqrt{\frac{3}{2}}(d + \frac{1}{2})^3$.

Lets have two spheres $Sphere_1$ and $Sphere_2$ each of radius $d + \frac{1}{2}$, as shown in Figure 6. The $Sphere_1$ intersects with $Sphere_2$, such that the distance between their centers is $d + \frac{1}{2}$. Now, we fit another sphere $Sphere_3$ in the intersecting space of $Sphere_1$ and $Sphere_2$ as shown in the Figure 6. It can be

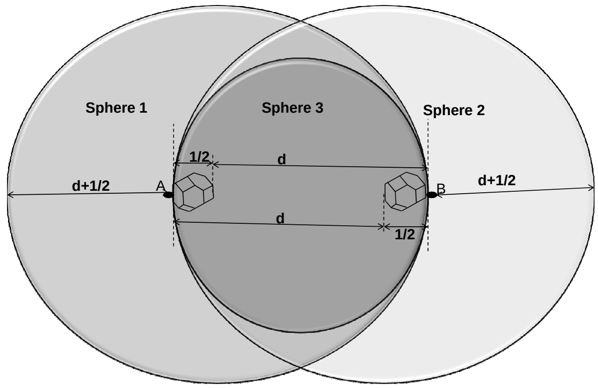

Fig. 6. Lower bound for chromatic number $\chi(G_d)$

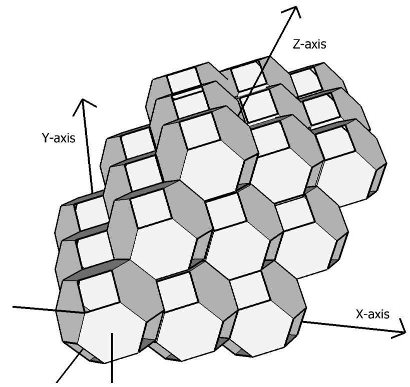

Fig. 7. Coloring pattern for $d = 1$, $\lceil m \rceil = 3$ and $\lceil n \rceil = 3$

observed that any two truncated octahedrons to_1 and to_2 that completely lie within $Sphere_3$ has $distance(to_1, to_2)$ at most d to each other. All such truncated octahedrons in $Sphere_3$ will form the max clique in graph $G_d = (TOC, E_d)$.

Hence, the $\omega(G_d)$ will be the number of truncated octahedrons which can fit into a sphere of diameter $d + \frac{1}{2}$. And is given as:

(Volume of $Sphere_3$)/(Volume of truncated octahedron of side $\frac{1}{2\sqrt{3}}$)= $\frac{\pi}{2}\sqrt{\frac{3}{2}}(d + \frac{1}{2})^3$.

Hence, $\omega(G_d)$ is given as $\frac{\pi}{2}\sqrt{\frac{3}{2}}(d + \frac{1}{2})^3$ □

The m^2n-coloring Algorithm

We are now ready to present the algorithm for coloring the truncated oactahedrons tilling the entire 3D space. The algorithm uses $\lceil m \rceil^2 \lceil n \rceil$ colors , hence, we call it the m^2n-coloring algorithm, here $m, n \in \Re^+$. The m^2n-coloring algorithm is illustrated in Algorithm 2 and it guarantees that for any two truncated octahedrons to_1 and to_2 having $distance(to_1, to_2)$ less than $d = (m-1)\frac{1}{2} = (n-1)\sqrt{\frac{2}{3}}$, (here $d \in \Re^+$), have different colors. Figure 7 shows the basic coloring pattern generated by the m^2n-coloring algorithm for $d = 1$. This basic coloring pattern is repeatedly used to fill the entire 3D space and colors all the truncated octahedrons tiling the 3D space. Further, we see that the coloring pattern for distance $d = 1$, has $\lceil m \rceil = 3$ and $\lceil n \rceil = 3$, hence, it uses 27 colors. Similarly, for $d = 2$, $\lceil m \rceil = 5$ and $\lceil n \rceil = 4$, therefore, the basic coloring pattern uses 100 colors. In general, for distance d, $\lceil m \rceil = \lceil (2d+1) \rceil$ and $\lceil n \rceil = \left\lceil d\sqrt{\frac{3}{2}} + 1 \right\rceil$ the basic coloring pattern uses $\lceil (2d+1) \rceil^2 \left\lceil d\sqrt{\frac{3}{2}} + 1 \right\rceil$ colors. As a result of this, the coloring algorithm uses $\lceil (2d+1) \rceil^2 \left\lceil d\sqrt{\frac{3}{2}} + 1 \right\rceil$ colors. Furthermore, the algorithm assigns the color $(k \mod \lceil n \rceil)\lceil m \rceil^2 + (j \mod \lceil m \rceil)\lceil m \rceil + (i \mod \lceil m \rceil) + 1$ to any truncated octahedron $to(i, j, k)$ in the new coordinate system.

Theorem 3: The approximation ratio of m^2n-coloring Algorithm is $\frac{8\sqrt{2}}{\pi}$

*P*roof. Let $d = (m-1)\frac{1}{2} = (n-1)\sqrt{\frac{2}{3}}$, then $m = 2d+1$ and $n = d\sqrt{\frac{3}{2}} + 1$.

Now, the number of colors required by the m^2n-coloring Algorithm is equal to:

$(2d+1)^2(d\sqrt{\frac{3}{2}} + 1)$

According to Lemma 2, the lower bound on the number of colors is $\frac{\pi}{2}\sqrt{\frac{3}{2}}(d + \frac{1}{2})^3$. Hence, it is easy to see the approximation ratio is $\frac{8\sqrt{2}}{\pi}$ □

Algorithm 2 m^2n-coloring Algorithm

for all hexagon $TOC(i, j)$ **do**
 $Color_{TOC(i,j)} \leftarrow (k \mod \lceil n \rceil)\lceil m \rceil^2 + (j \mod \lceil m \rceil)\lceil m \rceil + (i \mod \lceil m \rceil) + 1$
end for

V. Broadcast Scheduling Algorithm (BSA)

We now present the Broadcast Scheduling Algorithm (BSA) to solve the interference-aware broadcast scheduling problem for the network model defined in section II. BSA works similarly for both 2D and 3D WSNs. In this section, we first describe the BSA and then provide the analysis of $O(1)$-approximation ratio for interference-aware broadcast scheduling problem in 2D and 3D WSNs, respectively.

A. Algorithm Description

The BSA takes the transmission graph $G = (V, E)$ of the WSN and a source node $s \in V$ as input and computes the interference-aware broadcast schedule for broadcasting a message from the source node s to all the other nodes in the network. In order to do this, BSA first partitions the graph $G = (V, E)$ in to a set of layers $\{L_1, L_2, \ldots, L_R\}$ (here R is the radius of G with respect to s). For this, BSA runs the breadth first search on graph G considering s as the root node, any node at level i in the BFS tree belongs to layer L_i. Thenafter, BSA sequentionally transfers the broadcast message between the consecutive layers starting from layer $L_1 = \{s\}$. Consequently, all the nodes in the network receives the broadcast message.

During the first time slot the source node $s \in L_1$ transmits the broadcast message to the nodes in layer L_2. Thenafter, BSA runs $R-2$ iterations of the Interlayer Scheduling Algorithm (ISA). In each iteration, ISA generates the interference-aware schedules for transmission between two consecutive layers. The ISA is illustrated in Algorithm 4.

The ISA takes as input two consecutive layers L_i and L_j and generates an interference-aware transmission schedule from L_i to L_j. In order to do this, ISA first generates a maximal independent set $MIS(L_j)$ of the sub-graph $G(L_j)$ induced by the nodes in L_j in a way that $MIS(L_j)$ is the dominating set of L_j. Thenafter, ISA colors the nodes in $MIS(L_j)$ in such a way that no two nodes $u, v \in MIS(L_j)$ with Euclidean distance $d(u,v) < (\alpha+1)r_{max}$ have same color (see Lemma 3). Subsequently, ISA schedules the transmission in two phases. In each phase it generates number of time slots equal to the number of different colors used to color nodes in $MIS(L_j)$. In the first phase the nodes in $MIS(L_j)$ having same color are scheduled to receive from their corresponding transmitters in L_i in the same timeslot. As a result of this, in the end of the first phase all nodes in $MIS(L_j)$ receives the broadcast message without any interference. In the second phase nodes in $MIS(L_j)$ with same color are scheduled to transmit in the same timeslot. As, $MIS(L_j)$ is a dominating set of L_j, hence, once all the nodes in $MIS(L_j)$ have transmitted all the nodes in $L_j \setminus MIS(L_j)$ must have received the broadcast message. As a result, in the end of phases 2, all the nodes in layer L_j receive the broadcast message without any interference.

Algorithm 3 BSA($G(V,E)$, s)

Partition graph $G = (V,E)$ into distinct layers $\{L_1, L_2, \ldots L_R\}$ by running standard BFS on it.
In time slot 1, $L_1 = \{s\}$ transmits to all the nodes in L_2.
$Time \leftarrow 1$; $i \leftarrow 2$
while $i \neq R-1$ **do**
  $Time \leftarrow Time + ILTS(L_i, L_i + 1)$
  $i \leftarrow i+1$
end while

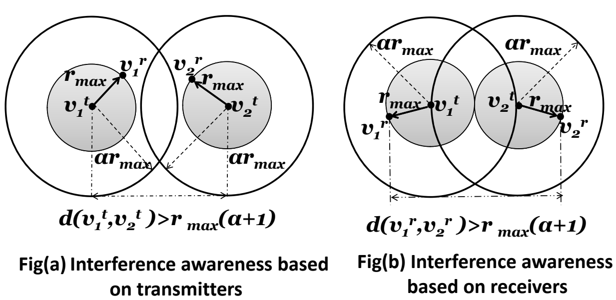

Fig. 8. The sufficient condition for interference-awareness

Lemma 3: For two transmissions $v_1^t \rightarrow v_1^r$ and $v_2^t \rightarrow v_2^r$ scheduled in the same time slot the sufficient condition for ensuring interference awareness is $d(v_1^t, v_2^t) > (\alpha+1)r_{max}$ or $d(v_1^r, v_2^r) > (\alpha+1)r_{max}$.

*P*roof: If the transmitters v_1^t and v_2^t have the transmission range r_{max}, then their interference ranges will be αr_{max}. The receivers v_1^r and v_2^r can be at most at r_{max} distance from their respective transmitters.

Algorithm 4 $ISA(L_i, L_j)$

Generate graph $G(L_j)$ induced by the nodes in L_j.
$MIS \leftarrow GenMIS(G(L_j))$
Perform proper coloring of MIS, such that no two nodes $u, v \in MIS(L_j)$ with $d(u,v) \leq (\alpha+1)r_{max}$ have same color.
$c \leftarrow$ Number of colors used to color MIS
$Time \leftarrow \phi$
Initialize $S_1, S_2, \ldots, S_c$ to ϕ
for all $u \in MIS$ **do**
  Select a node $w \in N^-(u) \cap L_i$
  $S_{color(u)} \leftarrow S_{color(u)} \cup \{w\}$
end for
for $i \leftarrow 1$ to c **do**
  S_i transmits
end for
$Time \leftarrow Time + c$.
Initialize $S_1, S_2, \ldots, S_c$ to ϕ
for all $u \in MIS$ **do**
  $S_{color(u)} \leftarrow S_{color(u)} \cup \{u\}$
end for
for $i \leftarrow 1$ to c **do**
  S_i transmits
end for
$Time \leftarrow Time + c$.
return $Time$

If we have $d(v_1^t, v_2^t) > (\alpha+1)r_{max}$ as shown in Figure 8, then according to triangular inequality $d(v_1^t, v_2^r) \geq \alpha r_{max}$. Similarly $d(v_2^t, v_1^r) \geq \alpha r_{max}$. This will ensure that the transmission from v_1^t to v_1^r will not interfere the transmission from v_2^t to v_2^r.

If we have $d(v_1^r, v_2^r) > (\alpha+1)r_{max}$ as shown in Figure 8, then according to triangular inequality $d(v_1^t, v_2^r) \geq \alpha r_{max}$. Similarly $d(v_2^t, v_1^r) \geq \alpha r_{max}$. This will ensure that the receiving of v_1^r from v_1^t will not be interfere by the receiving of v_2^r from v_2^t.

Hence, $d(v_1^t, v_2^t) > (\alpha+1)r_{max}$ or $d(v_1^r, v_2^r) > (\alpha+1)r_{max}$ is a sufficient conditions that ensure that two parallel transmissions $v_1^t \rightarrow v_1^r$ and $v_2^t \rightarrow v_2^r$ do not interference each other. □

B. $O(1)$-Approximation Ratio for IABS problem in 2D WSN

In case of a 2D WSN, we consider the 2D Euclidean plane is divided into regular hexagons of sides $\frac{r_{min}}{2}$ to form a hexagonal tiling. This hexagonal tiling is colored using the coloring method described in section III, keeping $d = (\alpha+1)r_{max}$. This ensures that when ISA is applied on a 2D WSN only one $MIS(L_j)$ node can be there in a hexagon. Further, a node in $MIS(L_j)$ is assigned the color of the hexagon it belongs to, this ensures no two nodes $u, v \in MIS(L_j)$ have same color if $d(u,v) < (\alpha+1)r_{max}$.

BSA when applied to a 2D WSN provides an approximation ratio of $2\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ for IABS problem. Lemmas 4, 5 and Theorem 4 provides the theoretical analysis.

Lemma 4: The transmission schedules produced by ISA for a 2D WSN are interference free.

*P*roof: When ISA is applied to a 2D WSN then, each hexagon can have only one $MIS(L_j)$ node and two hexagons having same color are at least distance $d = (\alpha+1)r_{max}$ apart. Hence, two $MIS(L_j)$ nodes simultaneously transmitting or receiving follows the sufficient conditions for interference-awareness described in Lemma 3, which results in interference-free scheduling. □

Lemma 5: The $MIS(L_j)$ node in ISA are colored using at most $\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ colors in case of a 2D WSN.

*P*roof: Every $MIS(L_j)$ node acquires the color of the hexagon in which it resides. When the sides of the hexagon are $\frac{1}{2}$ then, according to Theorem 2 for any distance d the number of colors needed to color the entire hexagon tiling is bounded by $\frac{4}{3}d^2 + \frac{8}{3}d + \frac{4}{3}$. From this, it is easy to see that, when the sides of the hexagon are $\frac{r_{min}}{2}$ the number of colors used are bounded by $\frac{4d^2}{3r_{min}^2} + \frac{8d}{3r_{min}} + \frac{4}{3}$. According to Lemma 3, for interference-awareness d should be $(\alpha+1)r_{max}$. Hence, the number of colors needed will be: $\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ □

Theorem 4: The BSA algorithm provides an approximation ratio $2\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ for a 2D WSN.

*P*roof: In the first phase the ISA generates $\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ schedules to make sure all the nodes in $MIS(L_j)$ receives form the nodes in L_i without any interference. Similarly, in second phase it generates another $\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ schedules to make sure all the nodes in $MIS(L_j)$ transmit to the nodes in $L_j \setminus MIS(L_j)$. Hence, it generates a total of $2\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$ schedules for transferring the broadcast message from layer L_i to layer L_j. Now, as BSA need to run $R-1$ iterations of ISA, hence, the total number of schedules generated or the broadcast latency of BSA is given as:

$1 + (R-2)2\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$

Furthermore, the theoretical lower bound of the broadcast scheduling problem is R.

Hence, the approximation ratio is:

$2\left\lceil \frac{4}{3}(\alpha+1)^2\beta^2 + \frac{8\beta(\alpha+1)}{3} + \frac{4}{3} \right\rceil$. □

Corollary 1: The approximation ratio provided by the BSA algorithm for a 2D WSN is $O(1)$ if the ratio $\alpha = \frac{r^I}{r^T}$ and ratio $\beta = \frac{r_{max}}{r_{min}}$ are bounded.

C. $O(1)$-Approximation Ratio for IABS problem in a 3D WSN

In case of a 3D WSN, we consider the 3D space is partitioned into identical truncated octahedrons of sides $\frac{r_{min}}{2\sqrt{3}}$ to form a truncated octaheragonal tiling. This truncated octaheragonal tiling is colored using the coloring method described in section IV, keeping $d = (\alpha+1)r_{max}$. This ensures that when ISA is applied on a 3D WSN only one $MIS(L_j)$ node can be there in a truncated octahedron. Further, a node in $MIS(L_j)$ is assigned the color of the truncated octahedron it belongs to, this ensures no two nodes $u, v \in MIS(L_j)$ have same color if $d(u,v) < (\alpha+1)r_{max}$.

BSA when applied to a 3D WSN provides an approximation ratio of $2\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ for IABS problem. Lemmas 6, 7 and Theorem 5 provides the theoretical analysis.

Lemma 6: The transmission schedules produced by ISA for a 3D WSN are interference free.

*P*roof: This can be proved similar to Lemma 4. When ISA is applied to a 3D WSN each truncated octahedron can have only one $MIS(L_j)$ node and two truncated octahedron having same color are at least distance $d = (\alpha+1)r_{max}$ apart. Hence, two $MIS(L_j)$ nodes simultaneously transmitting or receiving follows the sufficient conditions for interference-awareness described in Lemma 3, which results in interference-free scheduling. □

Lemma 7: When ISA is applied to a 3D WSN, the $MIS(L_j)$ nodes are colored using at most $\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ colors.

*P*roof: Every $MIS(L_j)$ node is assigned the color of truncated octahedron in which it resides. Consequently, the maximum number of colors assigned to the $MIS(L_j)$ nodes are bounded by the number of colors needed to color the entire truncated octahedragonal tiling.

As described in section IV, when the the truncated octahedrons tiling the 3D space have side length $\frac{1}{2\sqrt{3}}$, then for a distance $d = (m-1)\frac{1}{2} = (n-1)\sqrt{\frac{2}{3}}$, the m^2n-coloring algorithm uses $\lceil m \rceil^2 \lceil n \rceil$ colors.

But here, we have the truncated octahedron of side length $\frac{r_{min}}{2\sqrt{3}}$. Hence, $\lceil m \rceil^2 \lceil n \rceil$ colors can be used for $d = (m-1)\frac{r_{min}}{2} = (n-1)r_{min}\sqrt{\frac{2}{3}}\}$. According to Lemma 3, in order to assure interference-awareness d should be $(\alpha+1)r_{max}$. As a consequence of this we have:

$$
\begin{aligned}
d = (\alpha+1)r_{max} &= (m-1)\frac{r_{min}}{2} \\
\Rightarrow 2(\alpha+1)\frac{r_{max}}{r_{min}} &= (m-1) \\
\Rightarrow m &= 2(\alpha+1)\beta + 1 \\
d = (\alpha+1)r_{max} &= (n-1)r_{min}\sqrt{\frac{2}{3}} \\
\Rightarrow (\alpha+1)\beta\sqrt{\frac{3}{2}} &= (n-1) \\
\Rightarrow n &= (\alpha+1)\beta\sqrt{\frac{3}{2}} + 1
\end{aligned}
$$

Hence, the number of colors needed are bounded by $\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ □

Theorem 5: The BSA algorithm provides an approximation ratio $2\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ for a 3D WSN.

*P*roof: In the first phase the ISA generates $\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ schedules to make sure all the nodes in $MIS(L_j)$ receives form the nodes L_i without any interference. Similarly, in second phase it generates another $\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ schedules to make sure all the nodes in $MIS(L_j)$ transmit to the nodes in $L_j \setminus MIS$. Hence, it generates a total of $2\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$ schedules for transferring the broadcast message from layer L_i to layer L_j. Now, as BSA needs to run $R-1$ iterations of ISA, hence, the total number of schedules generated or the broadcast latency of BSA is given as:

$$1+(R-2)2\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$$

Furthermore, the theoretical lower bound of the broadcast scheduling problem is R.

Hence, the approximation ratio is: $2\lceil 2(\alpha+1)\beta+1 \rceil^2 \left\lceil (\alpha+1)\beta\sqrt{\frac{3}{2}}+1 \right\rceil$. □

Corollary 2: The approximation ratio provided by the BSA algorithm for a 3D WSN is $O(1)$ if the ratio $\alpha = \frac{r^I}{r^T}$ and ratio $\beta = \frac{r_{max}}{r_{min}}$ are bounded.

VI. Related Work

A wireless network can be closely modeled using a graph $G = (V, E)$, where the vertices in set V represent network nodes and edges in set E represent the communication links. The broadcast scheduling in wireless network has been studied using different kinds of graph models with an objective to minimize the broadcast latency. The theoretical lower bound to the minimum latency is the radius R of the network with respect to the source node $s \in V$. In [1], Chlamtac et. al. proved that the minimum latency broadcast schedule problem is NP-hard for general graphs. There are several additive [5], [6], and multiplicative [15], [14], [1], [2] approximation algorithms proposed for this problem in general graphs. In [3] Elkin et. al. proved that the minimum latency broadcast scheduling problem cannot have an $\Omega(logn)$ multiplicative approximation unless $NP \subseteq BPTIME(n^{O(loglogn)})$. In [4], they also proved that it is impossible to have an $opt(G) + log^2(n)$ additive approximation algorithm unless $NP \subseteq BPTIME(n^{O(loglogn)})$. However, for wireless networks, restrictive graph models such as Unit Disk Graphs (UDG) and Disk Graphs (DG) in 2D and Unit Ball Graphs (UBG) and Ball Graphs (BG) in 3D are more suitable. The UDG and UBG models a wireless network whose nodes have same transmission range, the DG and BG models the wireless networks where the nodes have different transmission ranges.

In [7], Gandhi et. al. studied the broadcast scheduling problem in Disk Graph (DG). They also proved that for a disk graph the minimum latency broadcast scheduling problem is NP-hard. In [10], Scott et. al. presented a solution for this problem in UDG. They use the geometric property of UDG to prove a lower bound of $16R - 15$. They also extended the pipelined broadcast algorithm in [9] for general graph to the UDG to get a lower bound of $R + O(log(R))$. This algorithm is based on standard ranking algorithm [16] for assigning ranks to the nodes of BFS tree. In [8], Chen et.al. studied this problem in a more realistic model, they considered transmission range and interference range to be different. They proposed an $O(\alpha^2)$ approximation ratio, where α is the ratio of interference range and transmission range, with $\alpha > 1$. In [12] Reza et. al studied this problem, considering $\alpha > 1$ and proposed an $O(\alpha^2)$ approximation with a better analysis. Apart from the interference and transmission range they also considered the carrier sensing range. Using a 2-Disk model, Scott et. al. studied the broadcast scheduling problem and proposed an approximation algorithm with ratio $6\left\lceil \frac{2}{3}(\frac{r_I}{r_T}+2) \right\rceil^2$ [11]. They considered $\alpha > 1$ and each node has two ranges, the transmission range r^T and the interference range r^I.

Unfortunately, almost all the above mentioned works considered the same transmission range for all the nodes. Only [7] considered a disk graph model, but it considers transmission range and interference range to be same, which is not a practical assumption, as interference range is always greater than the transmission range. Furthermore, all the existing works studying broadcast scheduling in wireless networks model the network as a 2D planar graph in which the nodes exists in 2D plane. However, this is not appropriate in all the cases, as most of the time the nodes acquire a locations in 3D. Therefore, in this paper we study the IABS problem for WSNs in 2D and 3D, we model a WSN in 2D as DG and in 3D as BG, considering a more realistic model where the nodes may have different transmission ranges, while their interference range is α times of their transmission range, with $\alpha > 1$. We propose $O(1)$-Approximation algorithm for IABS problem for WSNs modeled in 2D and 3D.

A related problem to tiling and coloring of 2D plane using regular hexagons is studied in [20], where the distance d between two hexagons is considered between their centers. This problem has a lot of motivation in channel assignment in cellular networks [18], [20], [29], [30], [30]–[34], where a base station is located at the center of a hexagon cell. The solution of this problem cannot be directly applied to optimally solve the tiling and coloring of 2D plane using regular hexagons, as the two problems are not equivalent.

The tiling and coloring of the 3D space using truncated octahedrons is not much studied. Tiling of 3D space using truncated octahedrons has earlier been considered in wireless network [22]. However, to the best of our knowledge, the coloring of the truncated octahedragonal tiling in 3D space has not been studied earlier.

VII. Conclusion

In this paper, we studied the interference-aware broadcast scheduling problem in WSNs in 2D and 3D. We considered a more realistic scenario, where the nodes have different transmission range and their interference range is larger than their transmission range. We modeled the transmission graph of a WSN in 2D and 3D using a disk graph (DG) and a ball graph (BG), respectively. Since the interference-aware broadcast scheduling is NP-hard, we propose a $O(1)$-Approximation algorithm for WSNs in 2D and 3D WSN. To the best of our knowledge, we are the first to study the interference-aware broadcast scheduling problem in 3D WSNs.

References

[1] I. Chlamtac and S. Kutten, On Broadcasting in Radio Networks-problems analysis and protocol design, *IEEE Transactions on Communications*, 33:1240-1246, 1985.

[2] I.Chlamtac and O. Weinstein, The wave expansion approach to broadcasting problem in multihop radio network, *IEEE Transaction on Communications*, 39:426-433, 1991.

[3] M. Elkin and G. Kortsarz, Logarithmic inapproximability of the radio broadcast problem, *Journal of Algorithms*, 52:8-25, 2004.

[4] M. Elkin and G. Kortsarz, Polylogarithmic additive inapproximability of the radio broadcast problem, *In APPROX-RANDOM '04: Proc. of 7th International workshop on Approximation Algorithms for Combinatorial Optimization Problems*, 2004.

[5] M. Elkin and G. Kortsarz, An improved Algorithm for Radio Networks, *Symposium On Discrete Algorithms (SODA)*, 2005.

[6] I. Gaber and Y. Mansour, Centralized Broadcast on Multi-Hop Radio Networks, *Journal of Algorithms*, 46(1):1-20, 2003.

[7] R. Gandhi, S. parthasarthy and A. Mishra, Minimizing broadcast latency and redundancy in ad hoc networks, *Mobihoc '03, Proceedings of the 4th international symposium on Mobile Ad hoc Networking and Computing*, pages: 222-232, 2003.

[8] Z. Chen, C. Qiao, J. Xu, and T. Lee, A constant approximation algorithm for interference -aware broadcast in wireless networks, *INFOCOM*, pages: 740-748, 2007.

[9] L. Gasieniec, D. Peleg and Q. Xin. Faster Communication in known topology radio network, *PODC '05: Proceeding of 24th annual symposium on Principles of distributed computing*, pages: 129-137, 2005.

[10] S.C.-H. Huang, P.-J. Wan, X. Jia, H. Du, and W.Shang, Minimum-latency broadcast scheduling in wireless ad hoc networks, *INFOCOM*, pages: 733-739, 2007.

[11] S.C.-H. Huang, P.-J. Wan, J. Deng and Y.S. Han Broadcast Scheduling in interference Environment, *IEEE Transaction on Mobile Computing*, vol. 7, No. 11, 2008.

[12] R. Mahjourian, F.Chen, R. Tiwari, M.T.Thai, H. Zhai, Y. Fang, An approximation algorithm for conflict-aware broadcast scheduling in wireless ad hoc networks, *MobiHoc'08,Proceedings of the 9th ACM international symposium on Mobile Ad hoc Networking and Computing*, pages: 331-340, 2008.

[13] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen and J.-P. Sheu, The broadcast storm problem in wireless ad hoc network, *MobiCom '99*, pages: 151-162, 1999.

[14] D. R. Kowalski and A. Pelc, Centralized deterministic broadcasting in undirected multi-hop radio network, *In APPROX-RANDOM '04: Proc. of 7th International workshop on Approximation Algorithms for Combinatorial Optimization Problems*, pages: 171-182, 2004.

[15] E. Kushilevitz and Y. Mansour, An $\Omega(Dlog(N/D))$ lower bound for broadcast in radio networks, *SIAM journal on Computing*, 27:702-712, 1998.

[16] X. G. Viennot, A strahler bijection between dyck paths and planar trees, *Discrete Math*, 246(1-3):317, 2002.

[17] A. Sen, T. Roxboroough and B. P. Sinha, On an Optimal Algorithm for Channel Assignment in Cellur networks, *ICC'99, International Conference on Communication*, 2:1147-1151, 1999.

[18] Teruya Fujii and Nobuo Nakazima, Radio Channel Assignment System for Cellular Mobile Communications, *Electronics and Communications in Japan*, part 1, Vol. 86, No. 5, 2003.

[19] C. Ho, K. Obraczka, G. Tsudik, and K. Vishvanath, Flooding for reliable multicast in multi-hop ad hoc networks, *In the proceeding of the 3rd international workshop on discrete algorithms and methods for mobile communication*, pages: 64-71, 1999.

[20] Gamst A. Homogeneous distribution of frequencies in a regular hexagonal cell system, *IEEE Transaction on Vehicular Technology* Volume: 31, Issue: 3, pages: 132-144, Aug 1982

[21] J.W.S. Cassels An introduction to the geometry of numbers, *Berlin: Springer-Verlag*, 1971

[22] S. M. Nazrul Alam and Zygmunt J. Haas, Topology Control and Network Lifetime in Three-Dimensional Wireless Sensor Networks, *url:http://www.citebase.org/abstract?id=oai:arXiv.org:cs/0609047*, 2006

[23] R. Kremens, J. Faulring, A. Gallagher, A. Seema and A. Vodacek, Autonomous field-deployable wild land fire sensors, *International Journal of Wild land Fire* Volume: 12, Pages: 237-244, 2003

[24] J. Heidemann, Y. Li, A. Syed, J. Wills and W. Ye, Underwater Sensor Networking: Research Challenges and Potential Applications, *USC/ISI Technical Report ISI-TR-2005-603*, 2005

[25] J. P. Lynch, Y. Wang, K. Loh, J.H. Yi and C.B. Yun, Wireless Structural Monitoring of the Geumdang Bridge using Resolution Enhancing Signal Conditioning, *Proceedings of the 24th International Modal Analysis Conference (IMAC XXIV), St. Louis, MO*, Jan 30-Feb 2, 2006

[26] N. Kurata, B.F.Spencer , M.R. Sandoval, Risk Monitoring of Building with Wireless Sensor Networks, *Structural Control and Health Monitoring*, Volume:12, Issue: 3-4, Pages: 315-327 Jun 2005

[27] Efficient Broadcasting in Multi-Radio Multi-channel, and Multi-hop wireless network based on self pruning. *High Performance Computing and Communications (HPCC 2007)*, Sept 2007

[28] Space-Filling Polyhedron *http://mathworld.wolfram.com/Space-FillingPolyhedron.html*

[29] W.K. Hale, Frequency Assignment: Theory and Applications. *Proceedings of IEEE*, Vol: 68 Dec 1980

[30] A. Sen, T. Roxborough and B.P. Sinha On an Optimal Algorithm for Channel Assignment in Cellular Networks *IEEE International Conference on Communications, ICC '99*, Vol: 2, Pages: 1147-1151, 1999

[31] J.R. Griggs and X.T. Jin, Optimal Channel Assignment for Lattices with Conditions at Distance Two *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, Vol: 13, Pages: 238, 2005

[32] R.A. Leese, A Unified Approach to the Assignment of Radio Channels on a Regular hexagonal Grid *IEEE Transactions on Vehicular Technology*, Vol: 46, Nov 1997

[33] J.F. Arnaud, Frequency Planning for Broadcast Services in Europe *Proceedings of the IEEE*, Vol: 68 Dec 1980

[34] S. Kim, S.L. Kim A Two-Phase Algorithm for frequency assignment in cellular mobile systems *IEEE Transactions on Vehicular Technology*, Vol:43, Aug, 1994