

ACHIEVING CONNECTED k -COVERAGE IN WIRELESS SENSOR NETWORKS USING
COMPUTATIONAL GEOMETRY-BASED APPROACHES

A Thesis

by

VENKATA SWAMY KALYAN NAKKA

Submitted to the College of Graduate Studies
Texas A&M University-Kingsville
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 2023

Major Subject: Computer Science

ACHIEVING CONNECTED k -COVERAGE IN WIRELESS SENSOR NETWORKS
USING COMPUTATIONAL GEOMETRY-BASED APPROACHES

A Thesis

by

VENKATA SWAMY KALYAN NAKKA

Approved as to style and content by:



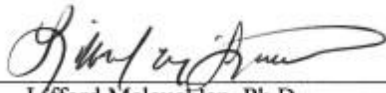
Habib M. Ammari, Ph.D.
Committee Chair



Maleq Khan, Ph.D.
Committee Member



George J. Toscano, Ph.D.
Committee Member



Lifford Mclauchlan, Ph.D.
Interim Department Chair



Jose F. Espiritu, Ph.D.
Interim Associate Vice President for
Research and Graduate Studies

May 2023

ABSTRACT

Achieving Connected k -Coverage in Wireless Sensor Networks using Computational Geometry-based Approaches

May 2023

Venkata Swamy Kalyan Nakka, Bachelor of Technology

IIT (ISM) – Dhanbad

Chair of Advisory Committee: Dr. Habib M. Ammari

The metrics, called coverage and connectivity, are often used to assess the sensor's sensing and communication capabilities in planar wireless sensor networks (PWSNs). A PWSN relies on the detection capabilities of the sensors to provide coverage. However, this is not sufficient for this type of network to function properly. In addition, having all the sensors connected, *i.e.*, they are capable of interacting with each other, is essential for the proper operation of PWSNs. This research aims to solve the connected k -coverage problem in PWSNs by ensuring that all field locations are covered or within the sensing range of at least k sensors ($k > 1$). Here, we provide a solution to the connected k -coverage problem using computational geometry-based approaches. Our goal is to maximize the lifetime of PWSNs by achieving connected k -coverage with a minimum number of sensors. To begin, we propose to tile the whole field of interest with planar tiles, which are convex polygons that do not overlap with each other and do not leave any gap in the underlying field. Following this, we compute the planar sensor density that is required to achieve k -coverage of a planar field of interest using these convex polygonal tiles. In addition, we determine network connectivity by correlating the sensing range of sensors with their communication range. Moreover, we propose energy-efficient

connected k -coverage protocols based on our planar convex polygonal tiles. Finally, we validate our conceptual analysis with facts from simulations.

DEDICATION

I hereby dedicate this thesis entitled “**ACHIEVING CONNECTED k -COVERAGE IN WIRELESS SENSOR NETWORKS USING COMPUTATIONAL GEOMETRY-BASED APPROACHES**” to my parents N. V. V. Narasimha Murthy and N. Uma Maheswari, my sister N. Harika, and all my friends, for their constant support.

Along with all hard working and respected Teachers, who taught and shared their knowledge with me and are responsible for all the success and honor in my career and life.

ACKNOWLEDGEMENTS

I would like to thank my Thesis advisor, Dr. Habib M. Ammari for his guidance and support throughout the course of this research. Also, I am very thankful to the committee members, Dr. Maleq Khan and Dr. George J. Toscano, for their support. In addition, I would like to thank Dr. Robert Diersing for his constant support during this thesis.

My special thanks also go to my friends and colleagues as well as the department faculty and staff for making my time at Texas A&M University-Kingsville a great experience.

Finally, I am very grateful to my mother and father for their encouragement, and to my sister for her patience and love.

NOMENCLATURE

PWSN	Planar Wireless Sensor Network
CERACCK	Centralized Randomized k -coverage protocol
T-CRACCK	Triangle-based Clustered Randomized k -coverage protocol
D-CRACCK	Disk-based Clustered Randomized k -coverage protocol
PR-Het-CCCK	Pseudo-Random Heterogeneous Centralized Connected k -coverage protocol
PR-Het-DCCK	Pseudo-Random Heterogeneous Distributed Connected k -coverage protocol
SCRT-PCAK	Static Centralized Reuleaux Triangle-Based Partition Coverage Algorithm
DCRT-PCAK	Dynamic Centralized Reuleaux Triangle-Based Partition Coverage Algorithm
DIRT-PCAK	Distributed Reuleaux Triangle-Based Partition Coverage Algorithm
DVOC	Distributed Voronoi based Cooperation scheme
ISCPk	Improved Stochastic k -Coverage-Preserving Sensor Scheduling algorithm
IRKCK	Irregular Hexagon-based k -coverage protocol
DIRACCK	Distributed Randomized k -coverage protocol
k -CSqu	k -coverage using Cusp Squares
k -InDi	k -coverage using Inner Diamonds protocol
LOFO	Loss of Overlap

TABLE OF CONTENTS

	Page
ABSTRACT.....	iii
DEDICATION.....	v
ACKNOWLEDGEMENTS.....	vi
NOMENCLATURE.....	vii
TABLE OF CONTENTS.....	viii
LIST OF FIGURES.....	x
LIST OF TABLES.....	xii
CHAPTER 1. INTRODUCTION.....	1
1.1 Motivation of the work.....	1
1.2 Goal and Objectives of the Research.....	1
1.3 Outline of Thesis.....	2
CHAPTER 2. PRELIMINARIES.....	3
2.1 Terminology.....	3
2.2 Sensing Model.....	4
2.3 Energy Model.....	4
2.4 Network Model.....	5
CHAPTER 3. RELATED WORK.....	6
CHAPTER 4. STUDY OF REGULAR CONVEX POLYGONS.....	8
CHAPTER 5. SQUARE TESSELLATION APPROACH.....	10
5.1 Connected k -coverage Theory.....	10

	Page
5.1.1 Square Tessellation generation	10
5.1.2 Sensor placement area construction.....	11
5.2 k -CSqu protocol	18
5.2.1 Cusp Square generation.....	18
5.2.2 Sensor selection and scheduling	18
CHAPTER 6. HEXAGONAL TESSELLATION APPROACH.....	20
6.1 Connected k -coverage Theory	20
6.1.1 Regular Hexagon-based Tessellation	20
6.1.2 Construction of Irregular Hexagonal Tile	21
6.1.3 Generalized Irregular Hexagonal Tile.....	23
6.2 k -InDi protocol.....	35
6.2.1 Irregular Hexagon tessellation generation	35
6.2.2 Sensor selection and scheduling	35
CHAPTER 7. RESULTS AND DISCUSSION	37
7.1 Simulation Environment	37
7.2 Results of k -CSqu	37
7.3 Results of k -InDi.....	40
7.4 Comparison of k -CSqu and k -InDi with DIRACC _{k}	43
CHAPTER 8. CONCLUSION	48
REFERENCES.....	50
VITA.....	54

LIST OF FIGURES

	Page
Figure 1. Biggest regular convex polygons enclosed in a circle.....	8
Figure 2. Square	
tessellation of square tile side length s_t	11
Figure 3. Cusp-square $EFGH$ inside square tile $ABCD$	11
Figure 4. k -Coverage area for a square tile	15
Figure 5. Cusp-square configuration for adjacent square tiles.....	17
Figure 6. Regular hexagonal tessellation with side length r_s	21
Figure 7. Construction of Irregular Hexagon.....	22
Figure 8. Irregular hexagons for different side lengths, proportional to sensing radius r_s	23
Figure 9. Curved area between arc AB and line segment AB.....	27
Figure 10. Curved area formed above the line segment BC.....	28
Figure 11. Inner diamond areas of adjacent generic irregular hexagons $IrHx(r_s/n)$	33
Figure 12. Planar sensor density λ versus (a) Sensing radius r_s and	
(b) Degree of coverage k	38
Figure 13. Number of active sensors n_a versus Number of deployed sensors n_d for different	
(a) Sensing radius r_s and (b) Degree of coverage k	39
Figure 14. Degree of coverage k versus Number of active sensor n_a	39
Figure 15. Planar sensor density λ versus (a) Sensing radius r_s ,	
(b) Degree of coverage k and (c) factor n	40, 41
Figure 16. Number of active sensors n_a versus Number of deployed sensors n_d for different	
(a) Sensing radius r_s , (b) Degree of coverage k and (c) factor n	42

Figure 17. Degree of coverage k versus Number of active sensor n_a for different (a) Sensing radius r_s and (b) factor n	43
Figure 18. Planar sensor density λ versus (a) Sensing radius r_s , (b) Degree of coverage k and (c) factor n , comparing k -CSqu, k -InDi and DIRACC $_k$	44,45
Figure 19. (a) Number of active sensors n_a versus Number of deployed sensors n_d (b) Degree of coverage k versus Number of active sensor n_a , comparing k -CSqu, k -InDi and DIRACC $_k$	45, 46
Figure 20. Remaining Energy versus Time, indicating the network lifetime.....	46
Figure 21. Number of active sensors n_a versus (a) Sensing radius r_s and (b) Communication radius r_c	47

LIST OF TABLES

	Page
Table 1. <i>LOFO</i> values of Regular Convex Polygons.....	9
Table 2(a). <i>IrHx(r/n)</i> structure for n	24
Table 2(b). Number of triangles per ring of <i>IrHx(r/n)</i> for n	24
Table 3. Generic structure of <i>IrHx(r/n)</i>	25
Table 4. $\lambda(k, r_s, n)$ as a function of n	32

CHAPTER 1. INTRODUCTION

Wirelessly linked networks of sensors that are planarly spread in any field, with the sole purpose of collecting data about the field's environment and sending it to a centralized location (the "sink"), make up what are known as planar wireless sensor networks (PWSNs). Humidity, temperature, light, sound, vibrations, pollutants, pressure, and so on, are only some of the factors that may be measured using these sensors. Because of their defining characteristic—their capacity to sense—all sensors are capable of monitoring these external factors. Once they have collected the data, the sensors send it to their neighbors, and so on, until it reaches the sink, where all the information about everything in the environment is stored. The whole field has to be well covered by PWSNs, and all sensors need to be connected to each other so that the collected data can be sent successfully to the sink.

1.1 Motivations of the Work

As we have already established, robust PWSN coverage necessitates dependable and fault-tolerant data collection from all sensors and dependable, loss-free data transfer to the sink. Thus, it is essential that the sensors be placed so as to eliminate dead spots in the network and provide complete coverage. When every point in a field of interest is covered by at least k sensors simultaneously, wherein $k > 1$, we say that we have k -coverage. When all sensors in a field that are involved in a k -coverage process are mutually connected, we say that the underlying PWSN ensures connected k -coverage.

1.2 Research Objectives and Goals

Our primary objective is to achieve connected k -coverage in PWSNs. As such, we need to set up the sensors in a field of interest in a manner that allows us to achieve the following goals:

- Full k -coverage of said field of interest.
- Connectivity among all the sensors participating in the k -coverage process.

1.3 Outline of Thesis

In this thesis, we provide novel approaches based on computational geometry to guarantee k -coverage of a planar field of interest and connectivity among all the actively involved sensors. This thesis is organized as follows: In Chapter 2, we introduce the basic ideas, terms, and models, including network model and energy model, which will be used in our study of the connected k -coverage problem in PWSNs. In Chapter 3, we discuss how others have employed computational geometry techniques to address the problem of connected k -coverage. In Chapter 4, we investigate various two-dimensional convex polygonal shapes for determining the best planar tile that can be used for a planar tessellation. Determining the best tile is achieved using our metric Loss of Overlap (*LOFO*). In Chapter 5, we establish the theory for connected k -coverage using square tessellation and introduce our k -coverage protocol using Cusp Squares (k -CSqu). In Chapter 6, we establish the theory for connected k -coverage using hexagonal tessellation, and present our k -coverage protocol using Inner Diamonds (k -InDi). In Chapter 7, we discuss the simulation results for evaluating our connected k -coverage protocols k -CSqu and k -InDi, and compare these results with those of our theoretical analysis. Moreover, we evaluate the efficiency of our k -CSqu and k -InDi protocols in comparison to that of an established connected k -coverage approach DIRACC_k [14]. In Chapter 8, we conclude and discuss all possible future directions and extensions of the proposed protocols.

CHAPTER 2. PRELIMINARIES

To address the challenge to build connected k -coverage configurations in PWSNs, we employ terminology and models particular to the domain, such as the sensing model and energy model, to provide methods and protocols. Chapter 1 provides a brief overview of the connected k -coverage issue in PWSNs and demonstrates how our solutions take into consideration the unique characteristics of each network. Following that, we offer definitions for numerous essential terminology and some context for the various models that we'll employ all through the rest of this thesis.

2.1 Terminology

In this part of the thesis, we define certain essential terms and offer some background information on some significant ideas.

A sensor's *sensing range* is the region across which it is capable of picking up 100% of events.

A sensor's *communication range* is the region in which it is able to exchange data with any other sensors.

If all the sensors that make up a PWSN have the same power source, sensing and transmission capabilities, storage space, and processing power, we say that the network is *homogenous*. If this isn't the case, we say that the WSN is *heterogeneous*. Our research into sensor heterogeneity centers on the first three characteristics listed: energy capacity, sensing range, and communication range.

We say that an area A is k -covered when every point in it can be viewed by minimum k sensors at once. In this context, " k -coverage" refers to the level of network coverage assurance provided by the PWSN that monitors on that particular region.

A PWSN is said to have connected k -coverage of a region it is paying attention on when all of its sensors are networked together in such a way that there is at least one means for each pair of sensors to interact with one another.

The *planar sensor density* is the number of sensors per unit area.

Tile is a self-replicating, self-adjointing pattern of convex regular polygons that completely covers an area without leaving any gaps or overlaps.

2.2 Sensing Model

In our deterministic sensing model, sometimes called a binary sensing model, every point P in the field may be sensed by a sensor s if and only if the Euclidean distance $\delta(P, s)$ is lower than or same as sensing range. The point P sensed by sensor s is denoted by the expression $Cov(P, s)$ in this context.

$$Cov(P, s) = \begin{cases} 1, & \text{if } \delta(P, s) \leq r_s \\ 0, & \text{otherwise} \end{cases}$$

2.3 Energy Model

It's widely known that sensors drain their power supplies for tasks like sending and receiving data, detecting, moving about, and so on. Active sensors also have an energy footprint even when they are not doing any of the aforementioned tasks. As published in Heinzelman et al. [1], the energy model is specified as, and may be used to calculate the energy required for data transmission and reception:

$$E_t(d) = b \times (\varepsilon d^\alpha + E_e)$$

$$E_r = b \times E_e$$

where, The amount of energy required for a sensor to communicate a message of size b bits over a certain distance d is indicated by $E_t(d)$, while the sensor's energy expenditure in deciphering that message is represented by $E_r(b)$. E_e is the electronic energy, and ε is either free-space (ε_{fs}) or multi-path (ε_{mp}) model and $\alpha \in [2, 4]$ is the path-loss exponent.

2.4 Network Model

For the sake of this investigation into the connected k -coverage problem, we assume that all sensors in a PWSN are similar and remain in the same locations. That is to say, the detecting and communicating capabilities of each sensor are equivalent. Moreover, in two-dimensional PWSNs, they are simulated as disk-shaped with radii of r_s and r_c , respectively, and centers that relate to the locations of the respective sensors; additionally, each sensor will have a distinct id, as well as All sensor's location is known owing to Location services or another location-finding method [2].

CHAPTER 3. RELATED WORK

There has been a lot of research done on the question of how to achieve connected k -coverage in PWSNs [3–13], with several authors contributing solutions that rely on exploiting fundamental computational geometrical characteristics while placing sensors. For the purpose of maintaining connected k -coverage in various configurations of PWSNs, researchers investigated and used numerous geometrical properties of regular and irregular polygons, such as the triangle, pentagon, and hexagon.

To ensure connected k -coverage in homogeneous PWSNs, Ammari *et al.* [14] investigated the Reuleaux triangle and its properties, and as a result, they proposed several protocols CERACCK, T-CRACCK, and D-CRACCK. To maintain connected k -coverage in heterogeneous PWSNs, Ammari [15] later introduced protocols, PR-Het-CCCK and PR-Het-DCCCK. Applying the idea and properties of the Reuleaux triangle discussed by Ammari [14], Yu *et al.* [16] proposed SCRT-PCAK, DCRT-PCAK, and DIRT-PCAK protocols. The Delaunay triangle was utilized by Qiu *et al.* [17] to reduce the number of possible situations that result in gaps in coverage. To maintain k -coverage while allowing the sensors to move to fill the coverage gaps, they used k -order Voronoi diagrams and created DVOC.

In order to guarantee stochastic k -coverage in homogeneous PWSNs, Yu *et al.* [18] presented the ISCPK method, which uses regular pentagons to partition the sensing range of each sensor. The connected k -coverage issue in PWSNs was solved by Ammari [19, 20] using regular hexagons and their attributes. To k -cover a field, he introduced the protocol IRKCK [19], which included creating an irregular hexagon out of regular hexagons. Furthermore, Ammari [20]

suggested, namely Cone-based k -coverage and Perimeter-based k -coverage protocols, that make use of concentric regular hexagons and area stretching.

Motivated by this body of research, we want to make better protocol contributions to the field of computational geometry-based sensor deployment in PWSNs with the goal of achieving connected k -coverage. We will examine many widely-used regular two-dimensional shapes in order to determine the best regular convex polygon for representing the sensor's sensing range and enhancing its area while overlaying a two-dimensional field. Our goal is to find the optimal regular two-dimensional convex polygons for tiling a two-dimensional plane.

To achieve k -coverage, we will tessellate a two-dimensional field using the ideal convex polygon we discover, and then investigate optimal sensor placement inside each tile. The optimal planar sensor density for achieving k -coverage in a two-dimensional field will also be calculated. We will utilize this method to find a better balance between the sensing range and the communication range of the sensors in order to guarantee connectivity among all of them and provide connected k -coverage of such field.

We will next implement all of these methods and create procedures for selecting the minimal number of sensors required to maintain continuous k -coverage in a two-dimensional region. We'll also evaluate the merits of the protocols we suggest against one another in an effort to zero in on the most advantageous option.

CHAPTER 4: STUDY OF REGULAR CONVEX POLYGONS

According to what was stated in Chapter 3, we will look into the regular convex polygons, such as the equilateral triangle, the square, and the regular hexagon, to find the optimal two-dimensional forms for tiling a planar field, Loss of Overlap.

The Loss of Overlap (*LOFO*) of a regular convex polygon of area A_T , compared to the sensing range A_S of a sensor, is given by:

$$LOFO = \frac{A_S - A_T}{A_S}$$

where $A_S = \pi r_s^2$ (based on Network Model).

Our goal is to discover the regular convex polygon which makes the most efficient use of the available sensing range. To rephrase, we need the regular convex polygon to reduce the unused sensing range, and this can be visualized in Figure 1. We set the criteria for best two-dimensional regular convex shape as $LOFO < 0.4$, which means that the sensing range not being utilized by that regular convex polygon is less than 40%. The values of *LOFO* for equilateral triangle, square and regular hexagon, as calculated using equation is presented in Table 1.

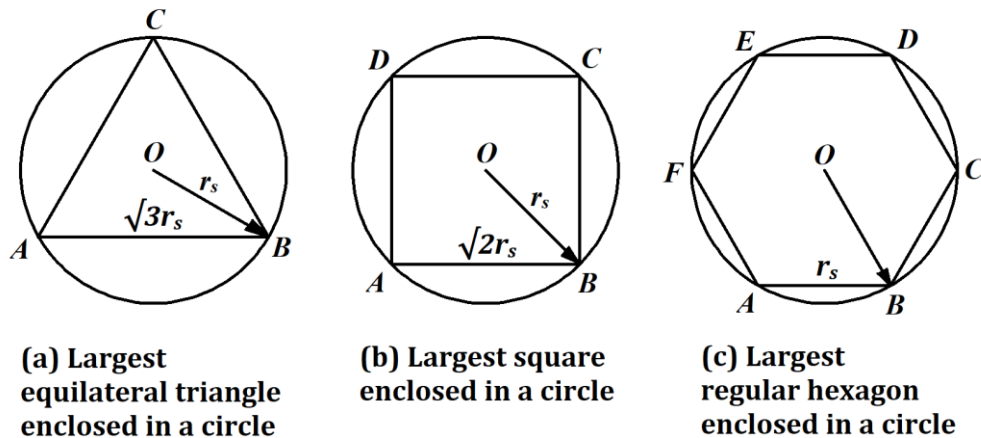


Figure 1. Biggest regular convex polygons inside a circle.

Table 1. *LOFO* values of Regular Convex Polygons.

Regular Convex Polygon	Area (A_T)	Loss of Overlap (LOFO)
Equilateral Triangle	$\frac{3\sqrt{3}}{4} r_s^2$	0.59
Square	$2r_s^2$	0.36
Regular Hexagon	$\frac{3\sqrt{3}}{2} r_s^2$	0.17

In light of the data in Table 1, and the criteria set for best two-dimensional shape, it is clear that square and regular hexagon are the best regular convex polygons for tessellating a planar field. Leveraging the geometrical properties of these two shapes, we will be developing our k -coverage theory for each of the shape and formulate connected k -coverage protocols of them. Of all the researched works discussed in Chapter 3, Ammari's works [10, 14, 15, 19, 20] are the only ones which developed a quantitative estimate for the least sensors needed to achieve k -coverage and has a specific procedure in building the k -coverage theory based on tessellation.

CHAPTER 5. SQUARE TESSELLATION APPROACH

This chapter delves into the challenge of k -covering a plane area of interest using a tessellation constructed from square tiles. In order to proceed with the solution, we state the planar k -coverage problem as:

Instance of the k -coverage problem: If we have a collection of sensors and a square tessellation, how many sensors should we position such that every square tile in the tessellation is covered by some subset of those sensors (i.e., we have k -coverage)?

5.1 Connected k -coverage Theory

The aforementioned k -coverage issue instance may be solved in two stages. In order to achieve k -coverage of the field of interest, we first tessellate it using square tiles of certain size that are proportionate to the sensing range of the sensor, and then we create the sensor placement area.

5.1.1 Square tessellation generation

As an initial step towards solving the problem, we tessellate the planar field of interest into adjacent and non-intersecting square tiles of side length s_t whose value is the value of sensing range r_s of the sensors. Figure 2 shows this generated square tessellation.

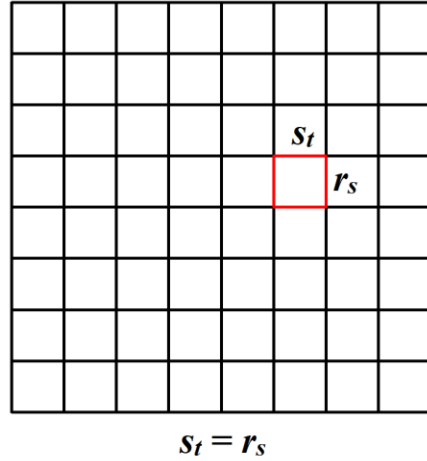


Figure 2. Square tessellation of square tile side length s_t .

5.1.2 Sensor placement area construction

We create a cusp square region within each square tile for sensor placement after tessellating the planar field of interest, as illustrated in Figure 3. To get k -coverage of every square tile, we strategically position k sensors within every cusp square.

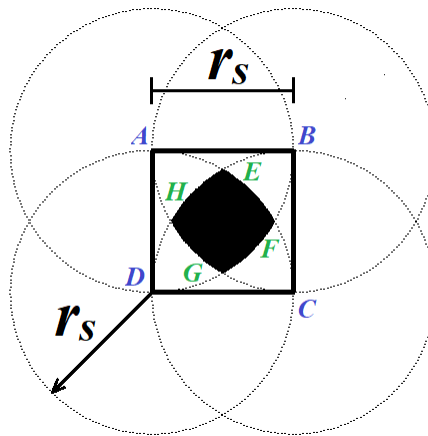


Figure 3. Cusp-square $EFGH$ inside square tile $ABCD$

Let us consider a square tile T of the tessellation. Now, we draw circles of radius r_s centered at each vertex of T , where an enclosed area is formed due to the intersection of these four circles. We call this enclosed area, *cusp square*. The distance between any neighboring pair

of vertices in this cusp square is the same (as in a square), and the sides of this cusp square are the arcs of the four circles.

Lemma 1 establishes a maximum allowable distance between any two vertices of a cusp square and a square tile.

Lemma 1 (Largest Distance in Square Tile): The maximum Euclidean distance between two points X and Y is r_s , where the domain of X is the set of vertices of a square tile and the domain of Y is the set of vertices of the corresponding cusp square. That is, $\delta(X, Y) \leq r_s$, $X \in \{\text{vertices of square tile}\}$ and $Y \in \{\text{vertices of cusp square}\}$, where δ is the Euclidean distance function.

Proof: Let us consider vertex A of square tile (Figure 3) and the farthest vertices of cusp square from vertex A are the vertices F and G . However, the vertices F and G are two points located on the circle whose center is vertex A and radius is r_s . Also, the other vertices H and E are inside the circle, making their distance from vertex A lesser than r_s . Similarly, the other vertices B , C and D exhibit the same geometric properties towards the vertices of the cusp square. Therefore, we can say that the largest possible distance between any vertex of a square tile and any vertex of its corresponding cusp square is r_s .

Using the conclusions of Lemma 1, the following Lemma 2 illustrates the requirement for k -coverage of a tile in a square tessellation.

Lemma 2 (Square Tile k -Coverage): If there are k active sensors set in the cusp square area of the tile, then the tile is k -covered.

Proof: In order to go on, we need just think about the minimum and maximum possible sensor placements in the tile's cusp square area. The minimum value scenario is represented by a sensor

located in the exact middle of the cusp square, while the upper bound case corresponds to the placement of a sensor at any vertex of the cusp square.

Case 1 (Lower bound): In this case measuring from one sensor to the furthest point of a square tile (vertex of the square tile) is given by:

$$d_1 = \frac{\sqrt{2}}{2} r_s = 0.707 r_s$$

Case 2 (Upper bound): From Lemma 1, measuring from one sensor to the furthest point of a square tile (vertex of the square tile) is computed as follows:

$$d_2 = r_s$$

Therefore, the distance between a sensor and the farthest point on a square tile falls in the range of $[0.707r_s, r_s]$. From this, it is clear that any sensor placed between these lower and upper position bounds will be able to cover the entire square tile. Consequently, if there are k sensors placed in between these lower and upper position bounds, all these k sensors will be able to k -cover the entire square tile. Therefore, for k -coverage of a square tile, k sensors must be placed in the tile's cusp square region.

Using Lemma 2, Theorem 1 provides the necessary and sufficient conditions for complete k -coverage of the target area.

Theorem 1 (k -Covered Field): Coverage of a region of interest is said to be " k -covered" if every cusp of the tessellation's square tiles has the minimum k active sensors.

Proof: From the Lemma 2, a square tile is k -covered if it has k active sensors in its corresponding cusp square area. Therefore, if all the square tiles of the tessellation are k -covered then the entire field of interest is said to be k -covered.

Relying on the cusp square area of the appropriate square tile, Lemma 3 below calculates the amount of the k -covered region.

Lemma 3 (k-Covered Area of Square tile): The k -covered area A_k formed by the bifurcation of the sensing disks of k sensors, which are placed in a cusp square area of its corresponding square tile, can be computed as follows:

$$A_k = \left(\frac{2\pi + 3 - 3\sqrt{3}}{3} \right) r_s^2$$

where r_s is the sensing range of the sensors.

Proof: The k -covered area corresponds to the largest area of the intersection, which is formed by the sensing areas of the sensors that are placed at cusp square area's vertices of corresponding square tile. This k -covered area A_k is calculated as:

$$A_k = A_{\text{Square } ABCD} + A_{\text{arc } AB \text{ \& line } AB} + A_{\text{arc } BC \text{ \& line } BC} + A_{\text{arc } CD \text{ \& line } CD} + A_{\text{arc } DA \text{ \& line } DA}$$

where $A_{\text{Square } ABCD}$ is the area of square ABCD and $A_{\text{arc } AB \text{ \& line } AB}$ is the curved area formed between the arc AB and line segment AB, as shown in Figure 4.

Notice that we have:

$$A_{\text{arc } AB \text{ \& line } AB} = A_{\text{arc } BC \text{ \& line } BC} = A_{\text{arc } CD \text{ \& line } CD} = A_{\text{arc } DA \text{ \& line } DA}$$

This implies that:

$$A_k = A_{\text{Square } ABCD} + 4 \times A_{\text{arc } AB \text{ \& line } AB}$$

where both $A_{\text{Square } ABCD}$ and $A_{\text{arc } AB \text{ \& line } AB}$ are computed as follows:

$$A_{\text{Square } ABCD} = r_s^2$$

$$A_{arc\ AB\ \&\ line\ AB} = \left(\frac{\pi}{6} - \frac{\sqrt{3}}{4} \right) r_s^2$$

Therefore, we obtain:

$$A_k = \left(\frac{2\pi + 3 - 3\sqrt{3}}{3} \right) r_s^2$$

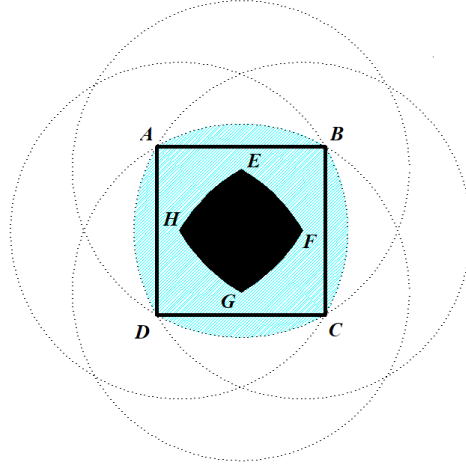


Figure 4. k -Coverage area for a square tile

Lemma 3 is used in Theorem 2 to calculate the planar density of sensors required to provide k -coverage of an area of interest. It depends on where the sensors are located within a square.

Theorem 2 (Planar Sensor Density of Square tile): To k -cover an area of interest, we calculate the planar density of sensors needed, denoted by $\lambda(k, r_s)$, as follows:

$$\lambda(k, r_s) = \frac{0.734k}{r_s^2}$$

where r_s is the sensing range of the sensors, and $k \geq 1$.

Proof: The planar sensor density of a square tile is the number of sensors deployed per area of the tile. As k -coverage is achieved when k sensors are distributed inside a cusp square A_k , this planar sensor density, denoted by $\lambda(k, r_s)$, is given by:

$$\lambda(k, r_s) = \frac{k}{A_k}$$

By substituting the value of A_k from Lemma 3, we get:

$$\lambda(k, r_s) = \frac{k}{\left(\frac{2\pi + 3 - 3\sqrt{3}}{3}\right)r_s^2} = \frac{0.734k}{r_s^2}$$

Notice that the planar sensor density $\lambda(k, r_s)$ varies on k and r_s .

Remark 1: Theorem 2 demonstrates that the planar sensor density we have is lower than that calculated in Ammari's publications [10, 14, 15]. That is, our approach requires fewer sensors for k -coverage process of a planar field of interest, compared to the one proposed in Ammari's works [10, 14, 15].

Lemma 4 below specifies the correlation between the radii of the sensing range and the communication range of the sensors that should exist to provide network connectivity of PWSNs. This type of relationship is essential for producing connected k -coverage configurations during the whole operational lifetime of PWSNs.

Lemma 4 (Network Connectivity for Square Tile): A Square tessellation-based k -coverage configuration is said to be interconnected if the radii of sensing and communication disks of the sensors, r_s and r_c , respectively, comply with the following inequality:

$$r_c \geq 2r_s$$

Proof: Two of the most distant sensors must be able to exchange data with each other in order to guarantee network connection. Let us consider the cusp square configuration of adjacent square tiles, as shown in Figure 5. Also, sensors are far from each other when they are placed at the vertices of cusp square areas that correspond to adjacent square tiles. For cusp square $E_1F_1G_1H_1$, if a sensor s_i is placed at vertex E_1 , the farthest sensor s_j from s_i should be placed at vertex G_3 of cusp square $E_3F_3G_3H_3$, which corresponds to a square tile that is adjacent to the one associated with cusp square $E_1F_1G_1H_1$. The distance between vertex E_1 and vertex G_3 is twice the radius of the circle passing through the vertices E_1, E_2, F_2, F_3, G_3 and H_1 . The line segment E_1G_3 forms the diameter of that circle, *i.e.*, $E_1G_3 = 2r_s$. In other words, to ensure that the two sensors s_i and s_j at vertex E_1 and vertex G_3 , respectively, can communicate with each other, the communication range of the sensors should be at least the length of line segment E_1G_3 . Therefore, $r_c \geq 2r_s$.

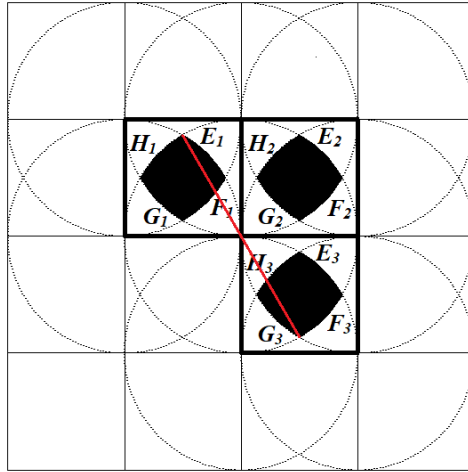


Figure 5. Cusp-square configuration for adjacent square tiles

Leveraging all the above-discussed and proved properties, we introduce our k -coverage protocol, called k -CSqu, which utilizes the cusp square areas of the square tiles of the square tessellation of a field of interest. Next, we discuss k -CSqu in detail.

5.2 *k*-CSqu Protocol

In this section, we discuss our *k*-coverage using *Cusp Squares (k-CSqu)* protocol is a centralized protocol for deterministic sensing process, which is performed by the sink. Our protocol has two major phases:

- Phase 1: Cusp Square generation for Square tessellation
- Phase 2: Sensor selection and scheduling

5.2.1 Cusp square generation

In this first step, before starting the rounds of *k*-coverage, the sink tessellates a planar field of interest using square tiles. Then, to schedule the sensors for each *k*-coverage round, it builds the cusp square regions for all square tiles of the tessellation, as was previously stated. The generated cusp square areas remain static (unchanged) throughout all the *k*-coverage rounds for all square tiles, and act as restriction areas for the sensor selection process for *k*-covering the entire field of interest.

5.2.2 Sensor selection and scheduling

In this second step, the sink selects the sensors and schedules their participation in the *k*-coverage process. Selecting and scheduling sensors such that their energy consumption drops down at about the same pace is the primary focus at this stage. This aids in ensuring that the battery power (or energy) of all the sensors runs out at about the same time, i.e., that they all have comparable lifespan. The second objective is to reduce the amount of energy used throughout each round of play so that the sensors' batteries last as long as possible. This will in turn help elongate the network operation lifetime. The sink identifies every sensor uniquely using its *id* and considers the sensors' locations and remaining battery power in the selection process for sensor scheduling (or duty-cycling) in every round of *k*-coverage process.

All sensors start out in sleep mode, but they all wake up to receive scheduling instructions from the sink at the beginning of each k -coverage cycle. Sometimes, the scheduled sensors might be around the cusp square areas but not inside. In such cases, the sensors will move inside cusp square area of their square tiles, thus, consuming battery power due to their mobility. The sink uses the sensor selection results from Lemma 2 and Theorem 1 to k -cover the region. This phase concludes with sink sending out a schedule broadcast to all sensors, which includes the sensor ids that were chosen for this iteration of k -coverage. When a sensor gets the schedule, it looks for its id to see whether it is included. In this case, the sensor will remove its id from the program, broadcast the updated schedule to its immediate neighbors, and continue to function as a live participant in the coverage procedure. If not, it will only share the schedule with its immediate neighbors and enter sleep mode.

CHAPTER 6. HEXAGONAL TESSELLATION APPROACH

Using hexagonal tiles, we explore the issue of k -covering the field in this chapter. Next, we state our instance of the planar k -coverage problem as follows:

Instance of the k -coverage problem: What is the optimal sensor deployment method for attaining k -coverage of a field with a given set of sensors and a hexagonal tessellation of that field, whereby each hexagonal tile of the tessellation may very well be k -covered by at least k sensors, where $k > 1$?

6.1 Connected k -coverage Theory

First, using a hexagonal tessellation, we want to resolve the 1-coverage problem of PWSNs, such that each point with in field is covered by at least 1 sensor(s).

6.1.1 Regular hexagon-based tessellation

We consider regular hexagon as the tile, of side length r_s , where r_s is the sensing range of sensor, for tessellating the field, and specify the sensor deployment strategy for attaining 1-coverage in PWSNs.

Sensor deployment for 1-coverage: As mentioned earlier, we tessellate the field using regular hexagons of side r_s , and area $A_{RH} = \frac{3\sqrt{3}}{2} r_s^2$, as shown in Figure 6, and place 1 sensor at the center of each regular hexagonal tile to achieve 1-coverage of PWSN.

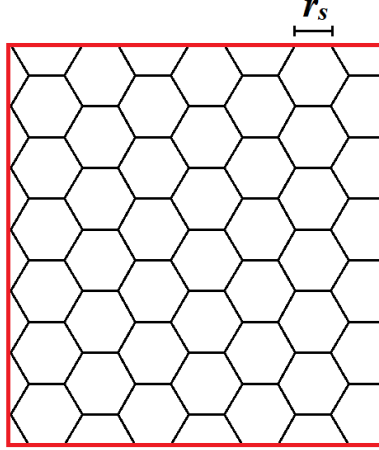


Figure 6. Regular hexagonal tessellation with side length r_s

Planar sensor density for 1-coverage: In light of this sensor deployment approach and the regular hexagon tiling, we can calculate the resulting planar sensor density as:

$$\lambda(r_s) = \frac{1}{A_{RH}} = \frac{1}{\frac{3\sqrt{3}}{2}r_s^2} = \frac{2}{3\sqrt{3}r_s^2} = \frac{0.38}{r_s^2}$$

We can easily determine that the farthest distance between any two neighboring sensors is $2r_s$. Therefore, network connectivity is ensured if $r_c \geq 2r_s$.

To apply this method of sensor deployment to the k -coverage problem, k sensors must be positioned in the middle of the area of every regular hexagonal tile. But placing k sensors at the center is unrealistic and not possible in real time. So, in order to place k sensors we need a dedicated area in every tile, which is more realistic.

6.1.2 Construction of irregular hexagonal tile

We change the approach by, first modifying the regular hexagon tessellation side length to $r_s / 2$, and second consider a diamond area formed by two equilateral triangles of same base in

the tessellation for placing k sensors, and then construct an irregular hexagon, which can be used as tile for tessellating the field.

Let us consider a diamond area D of the tessellation. We draw circles of radius r_s centered at each vertex of D , where an enclosed area is formed by the intersection of these four circles. In the intersection area, we obtain an irregular hexagon $IrHx(r_s/2)$ that is formed by 10 equilateral triangles with $r_s/2$ side length, as depicted in Figure 7, that can be used as a tile to tessellate a field.

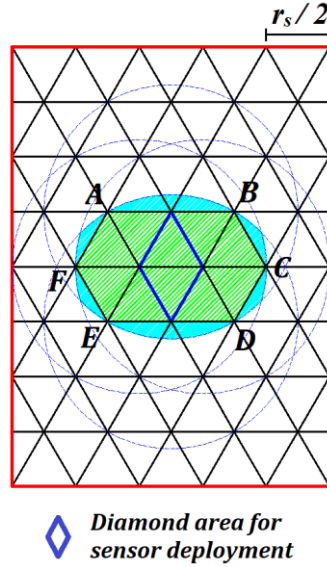


Figure 7. Construction of Irregular Hexagon

The coverage area for the configuration shown in Figure 7, is the sum of all 10 equilateral triangle areas and curvature areas formed at each side of the irregular hexagon, which is computed as:

$$A_k = \left[\pi + \frac{\sqrt{3}}{8} - \frac{\sqrt{15}}{4} - 4 \sin^{-1} \left(\frac{1}{4} \right) \right] r_s^2 = 1.3791 r_s^2$$

Thus, the corresponding planar sensor density for the above irregular hexagon $IrHx(r_s/2)$ based sensor deployment is computed as follows:

$$\lambda(k, r_s) = \frac{k}{A_k} = \frac{k}{1.3791r_s^2} = \frac{0.7251 k}{r_s^2}$$

Similarly, we can generate $IrHx(r_s/3)$, $IrHx(r_s/4)$ and $IrHx(r_s/5)$, by modifying the side length of the regular hexagonal tessellation as $r_s/3$, $r_s/4$ and $r_s/5$ respectively, as shown in Figure 8. We observed that planar sensor density $\lambda(k, r_s)$ pertaining to our irregular hexagon $IrHx(r_s/3)$ is lesser than that of $IrHx(r_s/2)$. Likewise, we found that the results achieved by $IrHx(r_s/4)$ are better than that of $IrHx(r_s/3)$, and results achieved by $IrHx(r_s/5)$ are better than that of $IrHx(r_s/4)$.

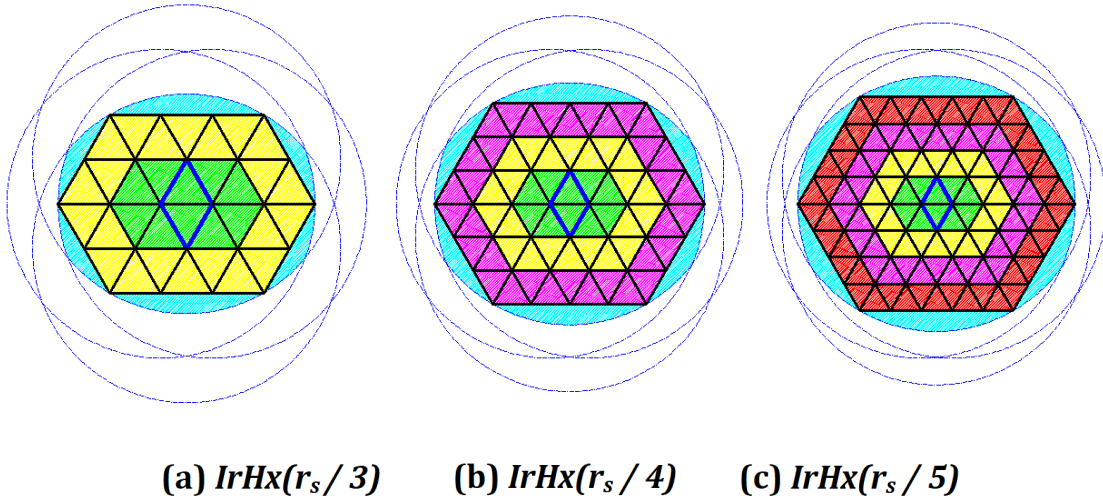


Figure 8. Irregular hexagons for different side lengths, proportional to sensing radius r_s

6.1.3 Generalized irregular hexagonal tile

We identified that the planar sensor density $\lambda(k, r_s)$ of our irregular hexagon-based configuration, is not only dependent on number of sensors k and sensing radius r_s , but also on side length of the regular hexagonal tessellation, which is proportional to sensing radius r_s .

Let us consider a more generic side length of r_s/n for our regular hexagonal tessellation, for understanding the properties of our irregular hexagon $IrHx(r_s/n)$. As noted from our previous cases ($n = 2, 3, 4$ and 5), our irregular hexagon $IrHx(r_s/n)$ has a distinct shape determined by the proportions of its six sides' lengths, *i.e.*, \overline{AB} , \overline{BC} , \overline{CD} , \overline{DE} , \overline{EF} and \overline{FA} , number of rings of equilateral triangles comprising it, and number of triangles per ring (shown with different colors in Figure 8). Table 2(a) demonstrates the configuration of our irregular hexagon $IrHx(r_s/n)$, and Table 2(b) shows the total of equilateral triangles for each ring, both for $n = 2, 3, 4, 5$.

Table 2(a). $IrHx(r_s/n)$ structure for n

n	\overline{AB}	\overline{BC}	\overline{CD}	\overline{DE}	\overline{EF}	\overline{FA}	# Rings
2	r_s	$r_s/2$	$r_s/2$	r_s	$r_s/2$	$r_s/2$	1
3	r_s	$2r_s/3$	$2r_s/3$	r_s	$2r_s/3$	$2r_s/3$	2
4	r_s	$3r_s/4$	$3r_s/4$	r_s	$3r_s/4$	$3r_s/4$	3
5	r_s	$4r_s/5$	$4r_s/5$	r_s	$4r_s/5$	$4r_s/5$	4

Table 2(b). Number of triangles per ring of $IrHx(r_s/n)$ for n

n	Ring #1	Ring #2	Ring #3	Ring #4
2	10			
3	10	22		
4	10	22	34	
5	10	22	34	46

We generalize the results from Table 2(a) above, in which the side length of the equilateral triangle is r_s/n , to our irregular hexagon $IrHx(r_s/n)$. Table 3 illustrates those generic results for $IrHx(r_s/n)$.

Table 3. Generic structure of $IrHx(r_s/n)$

\overline{AB}	\overline{BC}	\overline{CD}	\overline{DE}	\overline{EF}	\overline{FA}	# Rings
r_s	$\frac{(n-1)r_s}{n}$	$\frac{(n-1)r_s}{n}$	r_s	$\frac{(n-1)r_s}{n}$	$\frac{(n-1)r_s}{n}$	$n-1$

Lemma 5 below, which exploits the results illustrated by Table 3, computes the number of equilateral triangles for any ring l of our irregular hexagon $IrHx(r_s/n)$.

Lemma 5 (Number of triangles per ring): In an irregular hexagon $IrHx(r_s/n)$, number of equilateral triangles in ring l can be computed as:

$$N_l = 12l - 2$$

Proof: The results of Table 2(b) illustrate that for any value of n , the number of triangles for a certain ring of $IrHx(r_s/n)$ is constant, *i.e.*, for a ring p , the number of equilateral triangles $N_{l=p}$ is same for all values of n , where n is natural number and $n > 1$.

Let us consider the results in Table 2(b) for $n = 5$, we have $N_{l=1} = 10$, $N_{l=2} = 22$, $N_{l=3} = 34$, and $N_{l=4} = 46$. It is clear that the difference between the consecutive pairs of N_l values is constant, *i.e.*, $N_{l=2} - N_{l=1} = N_{l=3} - N_{l=2} = N_{l=4} - N_{l=3} = 12$. Thus, we can say that the N_l values for all rings of our irregular hexagon are in arithmetic progression (or sequence), where the initial term $N_0 = 10$ and common difference $d = 12$. This means that l^{th} term of this arithmetic progression gives us the number of equilateral triangles present in the ring l .

Therefore, number of equilateral triangles in ring l is computed as:

$$N_l = 10 + (l - 1) \times 12 = 12l - 2$$

Lemma 6 below computes the total number of equilateral triangles in our generic irregular hexagon $IrHx(r_s/n)$, by leveraging the results of Lemma 5.

Lemma 6 (Total triangles in IrHx(r/n)): Total number of equilateral triangles N , that comprise our generic irregular hexagon $IrHx(r/n)$ is given by:

$$N = 2(n - 1)(3n - 1)$$

Proof: From Lemma 3, we know that number of equilateral triangles for any ring l of $IrHx(r/n)$ is $N_l = 12l - 2$, and total number of rings of $IrHx(r/n)$ is $n - 1$.

So, total number of equilateral triangles of $IrHx(r/n)$ can be calculated as:

$$\begin{aligned} N &= \sum_{l=1}^{n-1} N_l = \sum_{l=1}^{n-1} 12l - 2 \\ &= \sum_{l=1}^{n-1} 12l - \sum_{l=1}^{n-1} 2 \\ &= 12 \times \sum_{l=1}^{n-1} l - 2 \times \sum_{l=1}^{n-1} 1 \\ &= 12 \times \frac{(n-1)n}{2} - 2 \times (n-1) \\ &= 6n(n-1) - 2(n-1) \\ &= (n-1)(6n-2) \\ &= 2(n-1)(3n-1) \end{aligned}$$

Therefore, total number of equilateral triangles of $IrHx(r/n)$ is given by $N = 2(n-1)(3n-1)$

Lemma 7 below computes the curved area between the arc AB and line segment \overline{AB} , which is also the same curved area between the arc DE and line segment \overline{DE} .

Lemma 7 (Curved Area above Longer Side): The curved area suspended between the arc AB and line segment \overline{AB} is computed as:

$$A_{LS} = \left(\frac{\pi}{6} - \frac{\sqrt{3}}{4} \right) r_s^2$$

where r_s is the sensing range of sensor.

Proof: From Figure 9, it is evident that the circle c_1 passing through the vertices A and B is centered at vertex O. Therefore, curved area formed between the arc AB and line segment \overline{AB} is:

$$\begin{aligned} A_{LS} &= A_{Sector\ AOB} - A_{Triangle\ AOB} \\ &= \frac{\pi r_s^2}{6} - \frac{\sqrt{3} r_s^2}{4} \\ &= \left(\frac{\pi}{6} - \frac{\sqrt{3}}{4} \right) r_s^2 \end{aligned}$$

Lemma 8 below computes the curved area formed above the line segment \overline{BC} , which is formed by intersection of two circles. It is worth noting that the curved formed above the sides \overline{BC} , \overline{CD} , \overline{EF} , and \overline{FA} are all equal.

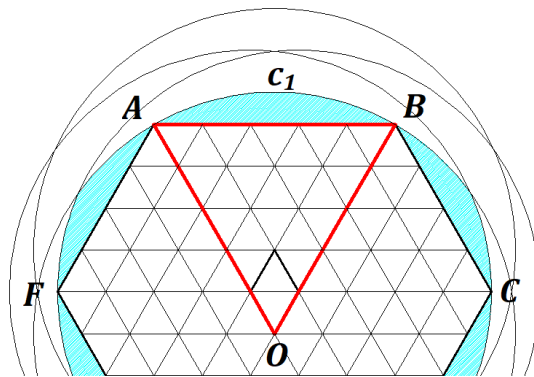


Figure 9. Curved area between arc AB and line segment AB

Lemma 8 (Curved Area above Shorter Side): The curved area formed above the side \overline{BC} is computed as:

$$A_{SS} = \left[\frac{\pi}{6} + \sin^{-1}\left(\frac{-1}{2n}\right) - \frac{\sqrt{4n^2 - 1}}{4n^2} - \frac{\sqrt{3}(n-2)}{4n} \right] r_s^2$$

where r_s is the sensing range of sensor.

Proof: Let us consider the coordinate system, demonstrated in Figure 10, where side BC is X-axis and B is the origin. Also, it is evident that the area suspended by arc BI and line segment BH is equal to the area suspended by arc CI and line segment CH, due to symmetry, where vertex I is the intersection point of two circles and H is the mid-point of side \overline{BC} .

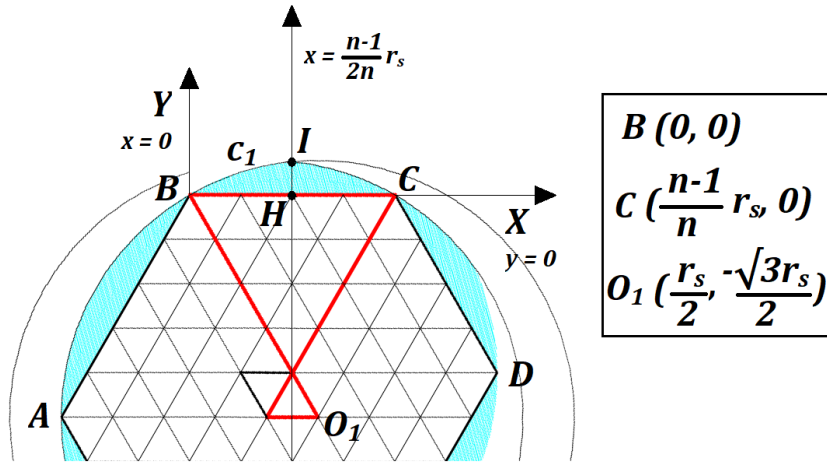


Figure 10. Curved area formed above the line segment BC

As circle c_1 is centered at vertex $O_1\left(\frac{r_s}{2}, -\frac{\sqrt{3}r_s}{2}\right)$, we get the equation of circle c_1 as:

$$c_1: \left(x - \frac{r_s}{2}\right)^2 + \left(y + \frac{\sqrt{3}r_s}{2}\right)^2 = r_s^2$$

$$\Rightarrow y_{c_1} = \sqrt{r_s^2 - \left(x - \frac{r_s}{2}\right)^2} - \frac{\sqrt{3}r_s}{2}$$

Now, we can compute the area between the arc BI and line segment BH as:

$$\begin{aligned}
A_{BIH} &= \int_0^{\frac{n-1}{2n}r_s} \int_0^{y_{c_1}} dy dx \\
&= \int_0^{\frac{n-1}{2n}r_s} y_{c_1} dx \\
&= \int_0^{\frac{n-1}{2n}r_s} \sqrt{r_s^2 - \left(x - \frac{r_s}{2}\right)^2} - \frac{\sqrt{3}r_s}{2} dx \\
&= \left[\frac{\pi}{6} + \sin^{-1}\left(\frac{-1}{2n}\right) - \frac{\sqrt{4n^2 - 1}}{4n^2} - \frac{\sqrt{3}(n-2)}{4n} \right] \frac{r_s^2}{2}
\end{aligned}$$

Therefore, the curved area formed above the side \overline{BC} is:

$$\begin{aligned}
A_{SS} &= 2 \times A_{BIH} \\
&= \left[\frac{\pi}{6} + \sin^{-1}\left(\frac{-1}{2n}\right) - \frac{\sqrt{4n^2 - 1}}{4n^2} - \frac{\sqrt{3}(n-2)}{4n} \right] r_s^2
\end{aligned}$$

Lemma 9 below computes the size of k -covered area based on our generic irregular hexagon $IrHx(r_s/n)$, exploiting the results of Lemma 6, Lemma 7, and Lemma 8.

Lemma 9 (k-Covered Area): The k -covered area A_k formed by the convergence of the sensing disks of k sensors, which are placed in inner diamond area of our generic irregular hexagon

$IrHx(r_s/n)$, can be computed as follows:

$$A_k = \left[\pi + \frac{(3n^2 - 6n + 2)\sqrt{3}}{4n^2} - \frac{\sqrt{4n^2 - 1}}{n^2} - 4 \sin^{-1}\left(\frac{1}{2n}\right) \right] r_s^2$$

where r_s is the sensing range of sensor, and n is a natural number such that $n \geq 1$.

Proof: The k -covered area is the largest convergence area, which is formed by the sensing disks of the sensors that are placed at the vertices of inner diamond area of our generic irregular hexagon $IrHx(r_s/n)$. This k -covered area A_k is computed as:

$$A_k = A_{Hexagon\ ABCDEF} + A_{Curved\ area\ above\ AB} + A_{Curved\ area\ above\ BC} + A_{Curved\ area\ above\ CD} \\ + A_{Curved\ area\ above\ DE} + A_{Curved\ area\ above\ EF} + A_{Curved\ area\ above\ FA}$$

Notice that we have:

$$A_{Hexagon\ ABCDEF} = N \times \frac{\sqrt{3}}{4} \left(\frac{r_s}{n}\right)^2 = (n-1)(3n-1) \frac{\sqrt{3}}{2} \left(\frac{r_s}{n}\right)^2 \text{ (Lemma 4),}$$

$$A_{Curved\ area\ above\ AB} = A_{Curved\ area\ above\ DE} = A_{LS} \text{ (Lemma 5) and,}$$

$$A_{Curved\ area\ above\ BC} = A_{Curved\ area\ above\ CD} = A_{Curved\ area\ above\ EF} =$$

$$A_{Curved\ area\ above\ FA} = A_{SS} \text{ (Lemma 6)}$$

Thus, we have the k -covered area as:

$$A_k = A_{Hexagon\ ABCDEF} + 2A_{LS} + 4A_{SS} \\ = (n-1)(3n-1) \frac{\sqrt{3}}{2} \left(\frac{r_s}{n}\right)^2 + 2 \times \left(\frac{\pi}{6} - \frac{\sqrt{3}}{4}\right) r_s^2 + 4 \times \left[\frac{\pi}{6} + \sin^{-1}\left(\frac{-1}{2n}\right) - \frac{\sqrt{4n^2-1}}{4n^2} - \frac{\sqrt{3}(n-2)}{4n}\right] r_s^2 \\ = \left[\pi + \frac{(3n^2 - 6n + 2)\sqrt{3}}{4n^2} - \frac{\sqrt{4n^2-1}}{n^2} - 4 \sin^{-1}\left(\frac{1}{2n}\right)\right] r_s^2$$

Lemma 9 is used in Theorem 3 below to calculate the required planar sensor density to guarantee k -coverage of the field. It is based on the sensor deployment within the inner diamond area of $IrHx(r_s/n)$.

Theorem 3 (Planar Sensor Density): The planar sensor density $\lambda(k, r_s, n)$, that is required in order to k -cover a field, is calculated as:

$$\lambda(k, r_s, n) = \frac{k}{\left[\pi + \frac{(3n^2 - 6n + 2)\sqrt{3}}{4n^2} - \frac{\sqrt{4n^2 - 1}}{n^2} - 4 \sin^{-1}\left(\frac{1}{2n}\right) \right] r_s^2}$$

where r_s is the sensing range of sensor, k and n are natural numbers, such that $k \geq 1$ and $n > 1$.

Proof: The planar sensor density of an irregular hexagonal tile is the number of sensors deployed per unit tile area. Hence, given that k sensors are placed within the inner diamond area to k -cover A_k , this planar sensor density, denoted by $\lambda(k, r_s, n)$, is given by:

$$\lambda(k, r_s, n) = \frac{k}{A_k}$$

By substituting the value of A_k from Lemma 9, we get:

$$\lambda(k, r_s, n) = \frac{k}{\left[\pi + \frac{(3n^2 - 6n + 2)\sqrt{3}}{4n^2} - \frac{\sqrt{4n^2 - 1}}{n^2} - 4 \sin^{-1}\left(\frac{1}{2n}\right) \right] r_s^2}$$

Notice that the planar sensor density $\lambda(k, r_s, n)$ depends only on k , r_s and n .

Remark 2: It is evident from Table 4 that, our result from Theorem 3 is a lesser planar sensor density compared to that of one deduced in Ammari's works, $\lambda(k, r_s) = \frac{0.8141 k}{r_s^2}$ [10, 14, 15], regardless of the value of n . This means, our approach achieves k -coverage of field with smaller number of sensors compared thereto of Ammari's works [10, 14, 15].

Table 4. $\lambda(k, r_s, n)$ as a function of n

n	2	3	4	5	10	20	100	∞
$\lambda(k, r_s, n)$	$\frac{0.7251 k}{r_s^2}$	$\frac{0.4267 k}{r_s^2}$	$\frac{0.3511 k}{r_s^2}$	$\frac{0.3168 k}{r_s^2}$	$\frac{0.2639 k}{r_s^2}$	$\frac{0.2431 k}{r_s^2}$	$\frac{0.2252 k}{r_s^2}$	$\frac{0.2252 k}{r_s^2}$

Lemma 10 indicates requisite relationship between sensing and communication ranges of sensors for continuous network operation of PWSNs. This relationship is the key attribute for attaining connected k -coverage during the entire operational lifetime of PWSNs.

Lemma 10 (Network Connectivity): An Irregular Hexagonal tessellation-based k -coverage configuration of sensors are mutually connected to each other, directly or indirectly, if the sensing and communication radii of sensors, r_s and r_c respectively, comply with the following inequality:

$$r_c \geq \frac{\sqrt{3n^2 + 1}}{n} r_s$$

Proof: The essential condition for maintaining network connectivity is that A two-way communication between the farthest sensors is necessary. Let us consider the adjacent generic irregular hexagonal configuration comprising the tiles T_1 , T_2 , T_3 and T_4 , as shown in Figure 11. Also, sensors are far from each other when they are placed at the vertices of inner diamond areas that correspond to adjacent irregular hexagonal tiles. For inner diamond $A_1B_1C_1D_1$ of tile T_1 , if a sensor s_i is placed at vertex B_1 , the farthest sensor s_j from s_i should be placed at either vertex D_3 of inner diamond $A_3B_3C_3D_3$ of tile T_3 , or vertex D_4 of inner diamond $A_4B_4C_4D_4$ of tile T_4 . We also have two other sensor placements, vertex A_1 to vertex C_3 and vertex C_1 to vertex A_4 , that are equivalent to the ones discussed above, but as all the lengths are equal, we will compute the length of $\overline{B_1D_4}$.

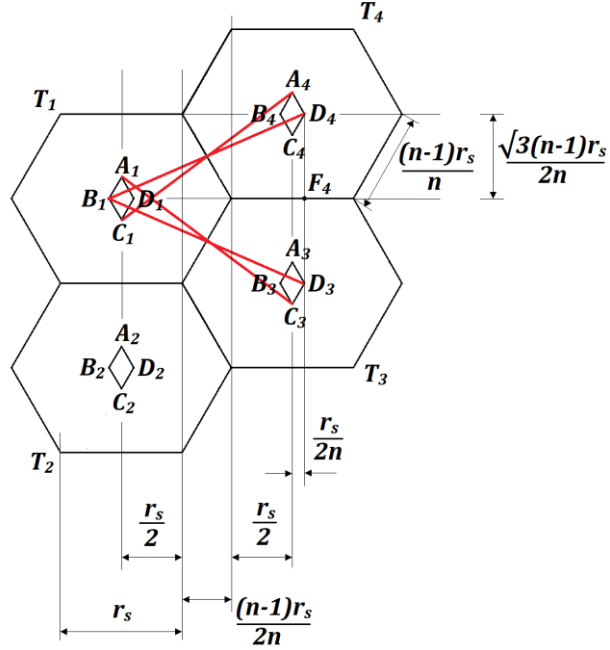


Figure 11. Inner diamond areas of adjacent generic irregular hexagons $IrHx(r_s/n)$

Length of $\overline{B_1D_4}$ is computed as:

$$\begin{aligned}
 \overline{B_1D_4}^2 &= \overline{B_1F_4}^2 + \overline{F_4D_4}^2 \\
 &= \left(\frac{r_s}{2n} + \frac{r_s}{2} + \frac{(n-1)r_s}{2n} + \frac{r_s}{2} + \frac{r_s}{2n} \right)^2 + \left(\frac{\sqrt{3}(n-1)r_s}{2n} \right)^2 \\
 &= \left[\frac{(3n^2 + 1)r_s^2}{n^2} \right]
 \end{aligned}$$

$$\Rightarrow \overline{B_1D_4} = \frac{\sqrt{3n^2 + 1}}{n} r_s$$

$$\text{Therefore, } r_c \geq \frac{\sqrt{3n^2 + 1}}{n} r_s.$$

Lemma 11 below, deduces a generic inequality involving the value of n .

Lemma 11 (n-Inequality): For any value of n , the following inequality holds true,

$$\frac{\sqrt{3n^2 + 1}}{n} < 2$$

where n is a natural number and $n > 1$.

Proof: Let us consider the domain of n , i.e., $n > 1$.

Raise to the power 2 on both sides of inequality, we get,

$$n^2 > 1$$

Add $3n^2$ on both sides of inequality, we get,

$$\Rightarrow n^2 + 3n^2 > 1 + 3n^2$$

$$\Rightarrow 4n^2 > 1 + 3n^2$$

Apply square root on both sides of inequality, we get,

$$\Rightarrow 2n > \sqrt{1 + 3n^2}$$

Divide by n on both sides of inequality, we get,

$$\Rightarrow 2 > \frac{\sqrt{3n^2 + 1}}{n}$$

Therefore, $\frac{\sqrt{3n^2 + 1}}{n} < 2$

Remark 3: Based on the results of Lemma 10 and Lemma 11, it is evident that our approach necessitates $r_c \geq \frac{\sqrt{3n^2 + 1}}{n} r_s < 2r_s$, which is better compared to that Wang *et al.* [21, 22] approach that requires $r_c \geq 2r_s$. This means, our approach requires lesser powered sensors that of Wang *et al.* [21, 22].

Based on all the proven mathematical properties, we can now generate a guided tessellation, utilizing the existing regular hexagonal tessellation, with our irregular hexagon $IrHx(r_s/n)$ as the new tile. Leveraging all the above-examined and established mathematical properties, we introduce our connected k -coverage protocol, called k -InDi, which utilizes the inner diamond areas of the irregular hexagonal tiles of the tessellation of field. Next, we discuss k -InDi in detail.

6.2 k -InDi Protocol

In this section, we discuss our k -coverage using *Inner Diamonds* (k -InDi) protocol, which is a centralized protocol of deterministic sensing model. Our protocol has two major phases:

- Phase 1: Irregular Hexagon tessellation generation
- Phase 2: Sensor selection and scheduling

6.2.1 Irregular hexagon tessellation generation

In this first phase, before initiating the k -coverage rounds, the sink tessellates the planar field using regular hexagonal tiles of side length r_s/n , based on the n value. Then, as discussed earlier, it initially generates one base irregular hexagon $IrHx(r_s/n)$ using a diamond area and utilizing this base hexagon, it guided tessellates over the existing regular hexagonal tessellation. This newly generated tessellation of $IrHx(r_s/n)$ remains static (unchanged) throughout the k -coverage process, and the inner diamond areas of each irregular hexagonal tile act as the bounded area for the sensor selection process for k -covering the entire planar field.

6.2.2 Sensor selection and scheduling

The sink chooses the sensors and arranges for participation in the k -coverage rounds during this second phase. This phase's main objective is to choose and duty-cycle (schedule) the sensors in a way that will maintain a nearly constant energy depletion rate over the course of the

whole k -coverage rounds. This implies that all sensors will have comparable lifetimes. This phase's secondary objective is to reduce battery power consumption every k -coverage round, which implicitly ensures prolonged lifetime and extends the operational lifetime of the network. Every sensor is assigned a unique identifier (id) by the sink, and the selection process for sensor scheduling (or duty-cycling) for each k -coverage round takes into account the sensors' locations and remaining battery power.

All sensors will initially be in the sleep mode, and for each k -coverage round, the sleep mode sensors will awaken in order to receive the scheduling instructions from the sink. The scheduled sensors may occasionally be outside the interior diamond sections but not within. In these situations, the sensors would move outside the confines of the inner diamond area of their asymmetric hexagonal tile, using battery power for mobility. The schedule, which contains a list of the scheduled sensors' ids for that particular k -coverage round, is broadcast by the sink at the conclusion of this step to all sensors. A sensor checks to see if its id is listed in the scheduling list after receiving the schedule. If this is the case, the sensor removes its id from the scheduling list, sends the modified schedule to its one-hop neighbors, and continues to be active for the k -coverage procedure. If not, it merely enters sleep mode after forwarding the schedule to its one-hop neighbors without making any changes to the scheduling list.

CHAPTER 7. RESULTS AND DISCUSSION

The simulation results of our suggested k -coverage methods are presented in this chapter, k -CSqu and k -InDi, using an open source high-level simulator [23] which is built using *C* and *Python* languages. We have modified the simulator to simulate k -coverage scenarios where degree of coverage k and tiling shape are provided as inputs. Also, we have added functionalities for plotting multiple results of the simulations performed. Next, our simulation environment is described in detail. Finally, we talk about our protocols' simulation performance for solving the connected k -coverage problem in PWSNs.

7.1 Simulation Environment

Our k -coverage protocols k -CSqu and k -InDi can be employed on any planar area regardless of its geometrical shape, be an irregular or regular polygon, like triangle, rectangle, hexagon, or decagon, to name a few. This study takes into account a 250-m-square area of interest. To ensure the proper functioning of our connected k -coverage protocols, k -CSqu and k -InDi, we assume that the initial battery capacity of sensor is 70 J, and we account for all forms of battery energy usage, including data sensing, data transfer, data reception, sensor agility, and control messages, in our energy model (Section 2.3). We presume a monitoring field of 1000 sensors, distributed equally and at random across the region of focus (Section 2.4). In addition, we presume that the sensor disk radii, both for sensing and communication, are 25 m and 50 m. Every simulation is repeated a hundred times, and the mean of the results is selected.

7.2 Results of k -CSqu

The results of the simulation for our k -CSqu protocol are presented here.

Figure 12 shows the variation of theoretical planar sensor density λ_{theo} (based on Theorem 2) and simulation planar sensor density λ_{sim} (for our k -CSqu protocol) with changing sensing radius r_s and degree of coverage k , for specific conditions. Figure 12(a) plots both λ_{theo} and λ_{sim} for varying r_s where $k = 3$. As expected, both λ_{theo} and λ_{sim} decreases with increasing r_s for a constant k . Figure 12(b) plots λ_{theo} and λ_{sim} for varying k where $r_s = 25$ m. We observe that, both λ_{theo} and λ_{sim} increases with increasing k , as expected for a constant r_s . Though the behavior is as expected, there is slight difference between the λ_{theo} and λ_{sim} that is clearly visible, and our protocol k -CSqu requires slightly higher sensor density compared to that of computed in Theorem 2. This is because, as calculated in Theorem 2, λ_{theo} considers all the common k -cover regions around the square tile, which includes the square area as well as the four curved areas on each side of each square tile. These curved areas form over k -cover regions in the field of interest which are not considered in λ_{sim} .

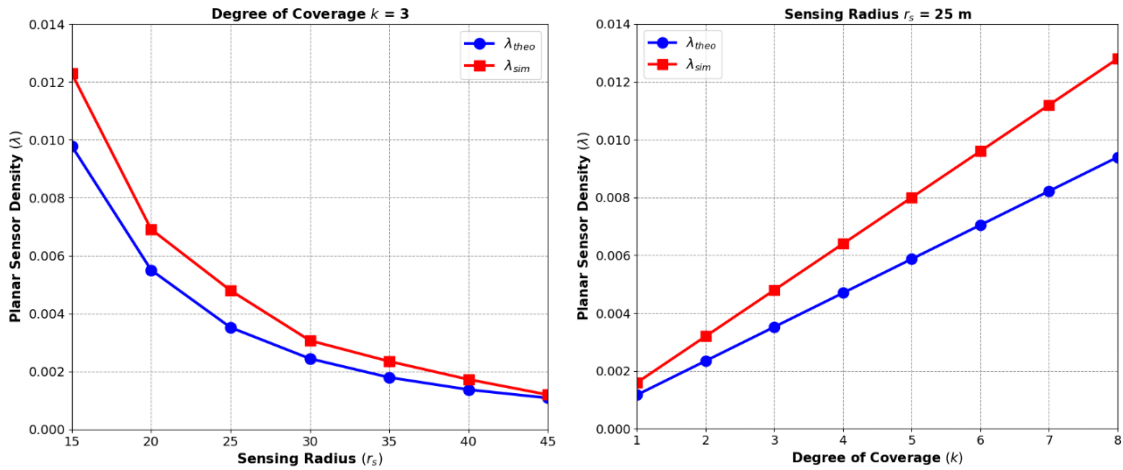


Figure 12. Planar sensor density λ versus (a) Sensing radius r_s and (b) Degree of coverage k

Figure 13 shows the number of active sensors n_a required compared to that of the number of deployed sensors n_d for our k -CSqu protocol. In Figure 13(a), we performed experiments by varying r_s , whereas in Figure 13(b), we performed experiments by varying k . For larger r_s values, less sensors are needed to achieve the necessary k level, while a larger k value necessitates a

larger number of sensors to achieve the same r_s level. In addition, in both evaluations, n_a is seen to be independent of n_d , and only dependent on r_s and k .

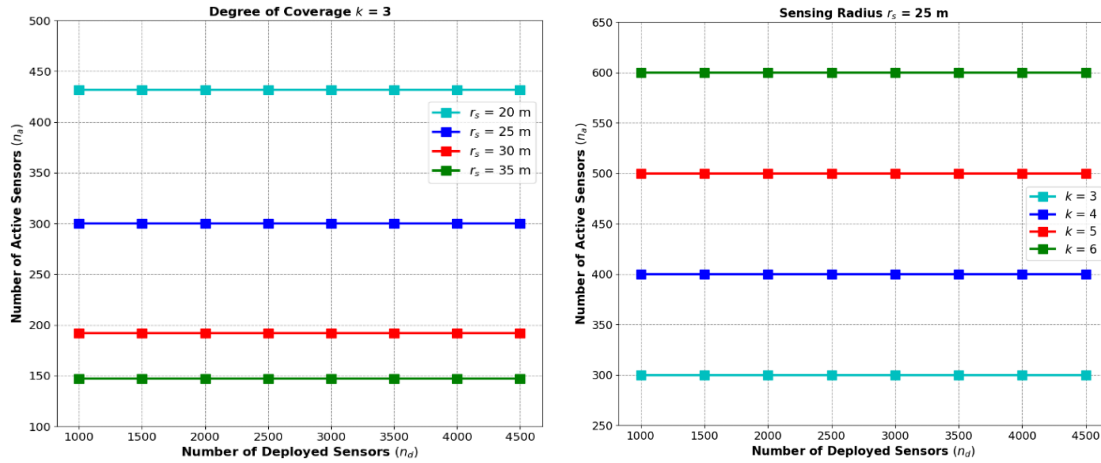


Figure 13. Number of active sensors n_a versus Number of deployed sensors n_d for different (a) Sensing radius r_s and (b) Degree of coverage k

In Figure 14, we can see the dependence of our k -CSqu protocol's degree of coverage k on the number of active sensors n_a . It is clear that k increases proportionally with n_a , also it should be noted that for a constant number of active sensors, k increases with increase in r_s indicating that larger field of interest can be k -covered.

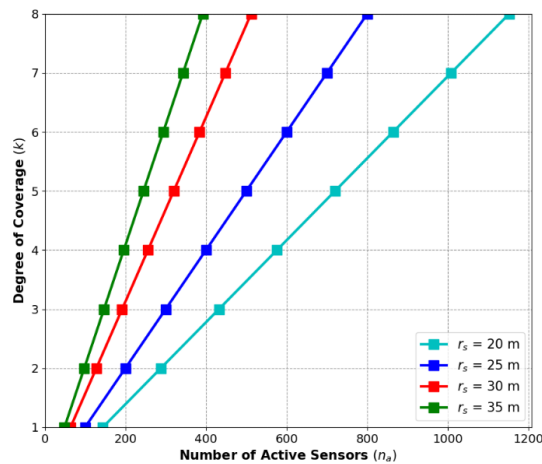
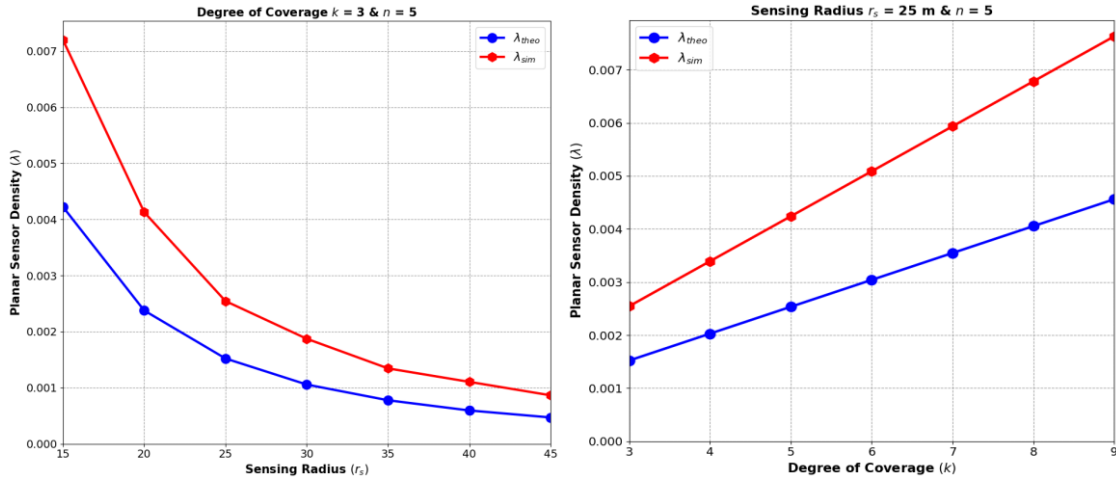


Figure 14. Degree of coverage k versus Number of active sensor n_a

7.3 Results of k -InDi

We demonstrate the simulation outcomes for our k -InDi procedure in this section. In Section 7.4, we contrast DIRACC $_k$ [14] with k -CSqu and k -InDi.

Figure 15 shows the variation of theoretical planar sensor density λ_{theo} (based on Theorem 3) and simulation planar sensor density λ_{sim} (for our k -InDi protocol) with changing sensing radius r_s , degree of coverage k and factor n , for specific conditions. Figure 15(a) plots λ_{theo} and λ_{sim} for varying r_s where $k = 3$ and $n = 5$. As expected, all three λ_{theo} and λ_{sim} decreases with increasing r_s for a constant k and n . Figure 15(b) plots λ_{theo} and λ_{sim} for varying k where $r_s = 25$ m and $n = 5$. We observe that, λ_{theo} and λ_{sim} increases with increasing k , as expected for a constant r_s and n . Figure 15(c) plots λ_{theo} and λ_{sim} for varying n where $k = 3$ and $r_s = 25$ m. As expected, λ_{theo} and λ_{sim} decreases with increasing n for a constant k and r_s .



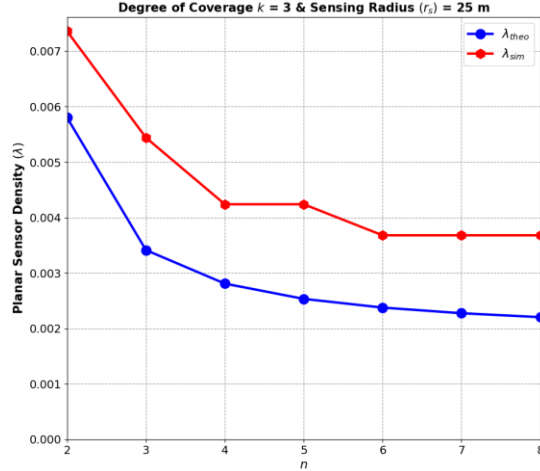


Figure 15. Planar sensor density λ versus (a) Sensing radius r_s , (b) Degree of coverage k and (c) factor n

Though the behavior is as expected, there is slight difference between the λ_{theo} and λ_{sim} that is clearly visible, and our protocol k -InDi requires slightly higher sensor density compared to that of computed in Theorem 3. This is because, as calculated in Theorem 3, λ_{theo} considers all the common k -cover regions around the sides of the irregular hexagonal tile, which includes the hexagonal area as well as the six curved areas on each side of each irregular hexagon tile. These curved areas form over k -cover regions in the field of interest which are not considered in λ_{sim} .

Figure 16 shows the number of active sensors n_a required compared to number of deployed sensors n_d for our k -InDi protocol. In Figure 16(a), we performed experiments by varying r_s , in Figure 16(b), we performed experiments by varying k and in Figure 16(c), we performed experiments by varying n . Clearly, as r_s and n increase, the number of sensors needed to achieve a given k decreases, but as k increases, the number of sensors needed to achieve that k increases as well. Furthermore, it is evident from all experimental studies that the values of r_s , n , and k influence n_a , but not n_d .

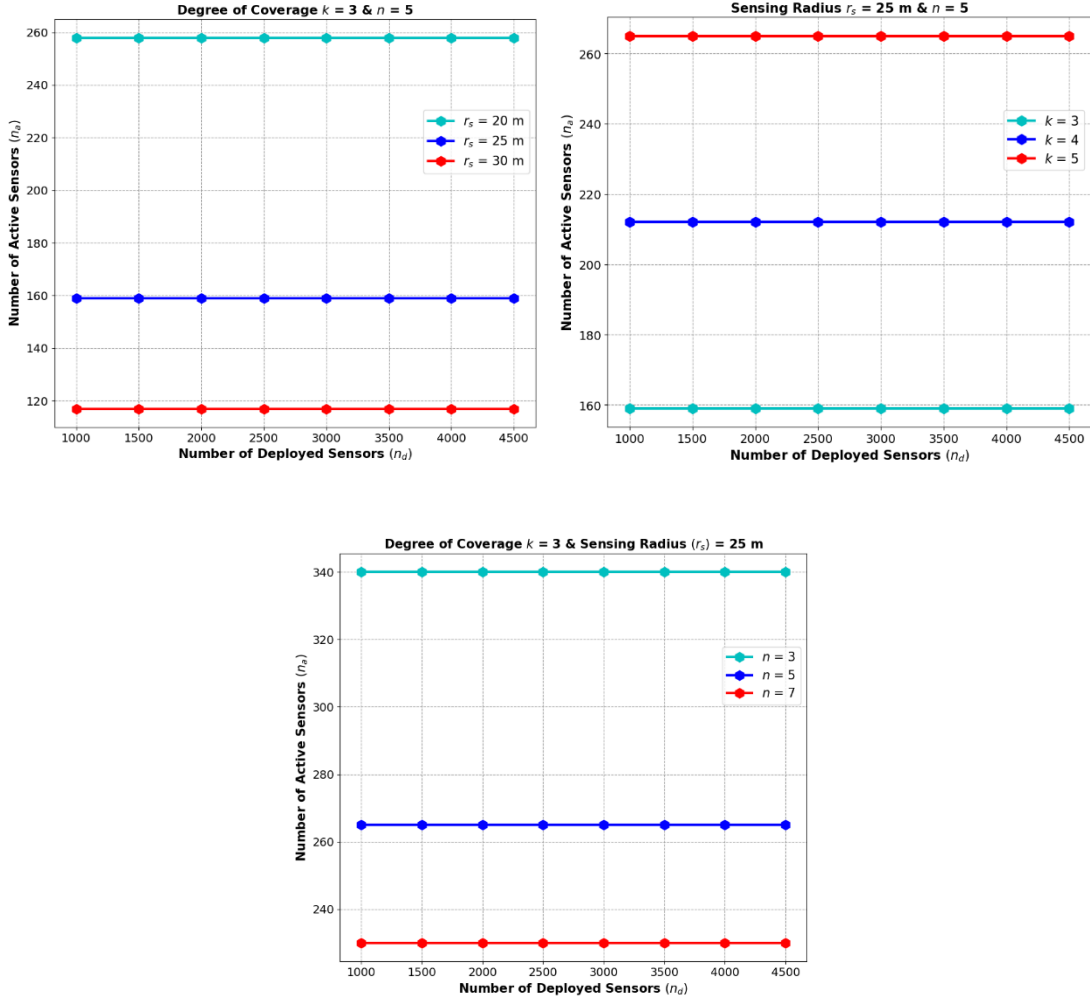


Figure 16. Number of active sensors n_a versus Number of deployed sensors n_d for different (a) Sensing radius r_s , (b) Degree of coverage k and (c) factor n

Figure 17 shows the correlation between the number of active sensors (n_a) and the coverage level (k) achieved by our k -InDi protocol. The experiments in Figure 17(a) are conducted by varying r_s for constant $n = 5$, whereas in Figure 17(b), experiments are conducted by varying n for constant $r_s = 25$ m. It is evident that for a fixed number of active sensors, k grows with r_s , indicating that a wider area of interest can be covered by k . It is also evident that, k increases with increase in n indicating that a smaller number of larger irregular hexagonal tiles can be used for k -coverage process.

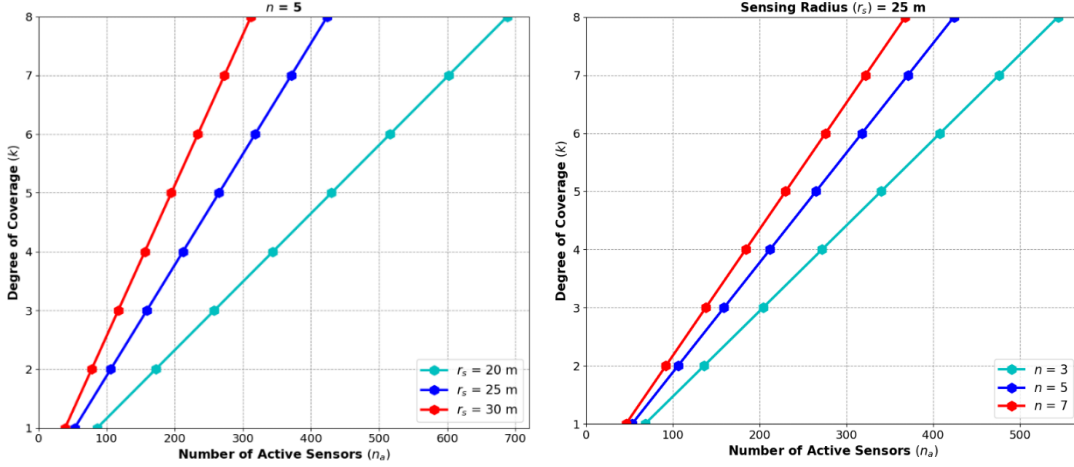


Figure 17. Degree of coverage k versus Number of active sensor n_a for different (a) Sensing radius r_s and (b) factor n

7.4 Comparison of k -CSqu and k -InDi with DIRACC $_k$

In this section we compare our protocols k -CSqu and k -InDi with DIRACC $_k$ [14]. In Chapter 3, we could see Ammari's research [10, 14, 15] looked into the issue of k -coverage in PWSNs and presented Reuleaux triangle-based protocols taking into account a degree of coverage $k = 3$. Hence, we consider $k = 3$ for comparing our protocols k -CSqu and k -InDi with Ammari's protocol DIRACC $_k$ [14], which is proved to be better than CCP [21, 22].

Figure 18 plots the comparison of simulation obtained planar sensor density λ of k -CSqu, k -InDi and DIRACC $_k$ protocols with varying sensing radius r_s in Figure 18(a), varying degree of coverage k in Figure 18(b) and varying factor n in Figure 18(c). From the plots in Figure 18 (a) & (b), it is clear that our protocol k -InDi and k -CSqu have lower λ compared to that of DIRACC $_k$, for a specific r_s as well as k . Therefore, it is evident that for a specific level of k , our protocols k -InDi and k -CSqu allows for k -coverage of the field to be attained with a smaller sensor deployment compared to that of DIRACC $_k$. In other words, we can say that for specific fixed λ , our protocols k -InDi and k -CSqu offers higher level of k compared to DIRACC $_k$. Also

for a constant λ , k -InDi and k -CSqu requires low sensing range sensors for achieving desired level of k . Thus, it is clear that k -InDi and k -CSqu requires less powered sensors for k -coverage process (for desired conditions) compared to $DIRACC_k$. Furthermore from Figure 18(c), we see that with increase in the factor n , the value of λ decreases proportionally, for a constant sensing radius r_s and degree of coverage k . This indicates that our protocol k -InDi achieves same level of coverage for different λ , and it is clear that for higher value of n , our protocol achieves desired level of coverage with lesser number of sensors of same characteristics as initial power, sensing range, communication range and etc.

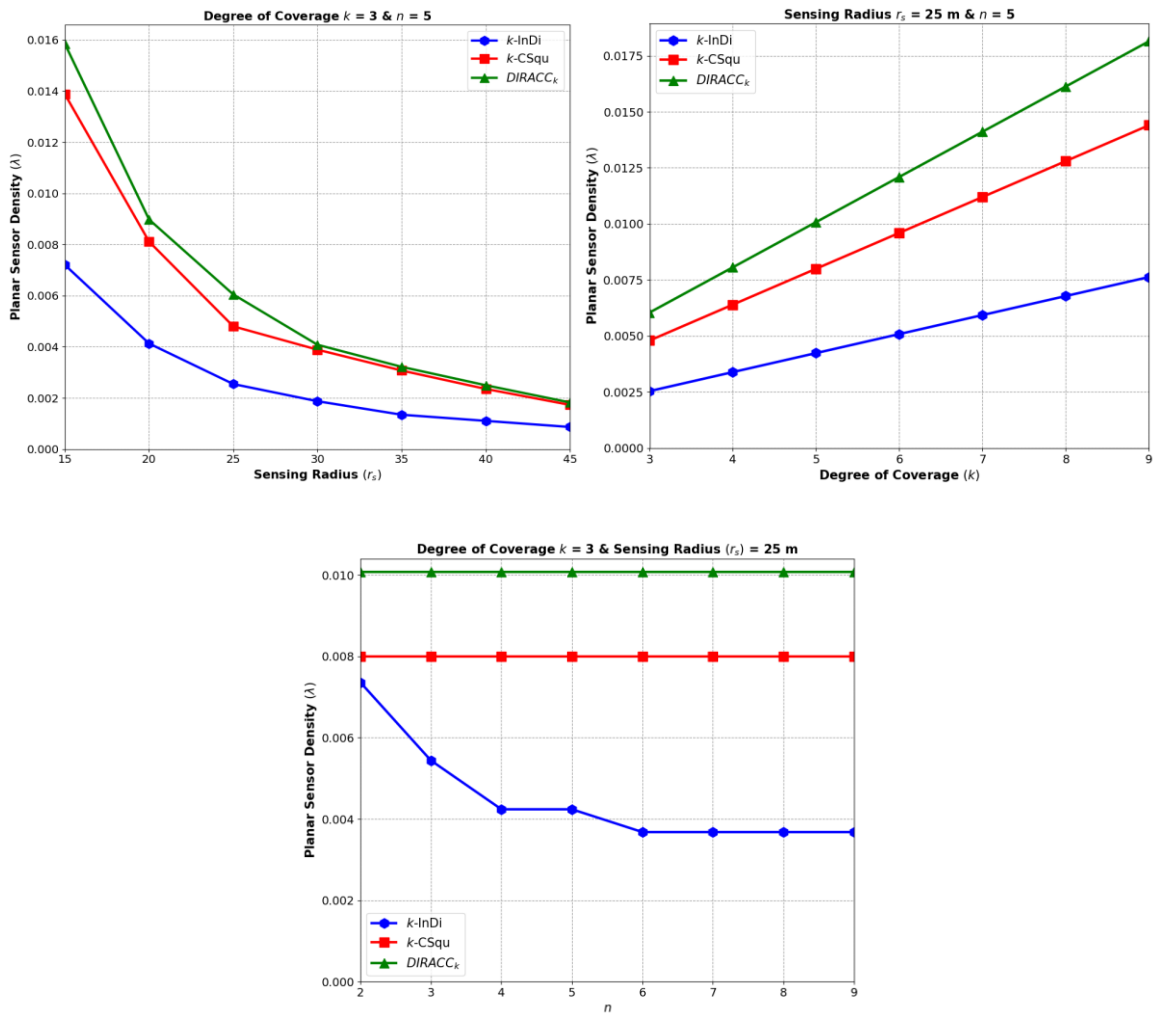


Figure 18. Planar sensor density λ versus (a) Sensing radius r_s , (b) Degree of coverage k and (c) factor n , comparing k -InDi and DIRACC_k

Figure 19(a) shows the differences between k -InDi, k -CSqu and DIRACC_k in respect to required total number of active sensors n_a compared to the total number deployed sensors n_d . As inferred earlier from Figure 18 results, Figure 19(a) proves that our k -InDi and k -CSqu techniques achieves the same level of degree of coverage k , with fewer active sensors. Figure 19(b) shows the relationship between the number of active sensors (n_a) and the coverage (k) for all k -InDi, k -CSqu, and DIRACC_k protocols. It is clear and supports our prior conclusion that, for a given number of active sensors, k -InDi and k -CSqu provides a higher level of coverage than DIRACC_k. As a result, given a particular targeted level of k , our protocols k -InDi and k -CSqu produces significant energy savings, hence increasing the operational time of the underlying sensor network. Reason being, DIRACC_k requires more active sensors, which uses more energy for sensing, and messaging between active sensors using DIRACC_k creates a communication overhead that prevents active sensors from cooperating with one another and providing the anticipated level of degree of coverage. As a result, DIRACC_k network operations will use more energy to k -cover the area of interest.

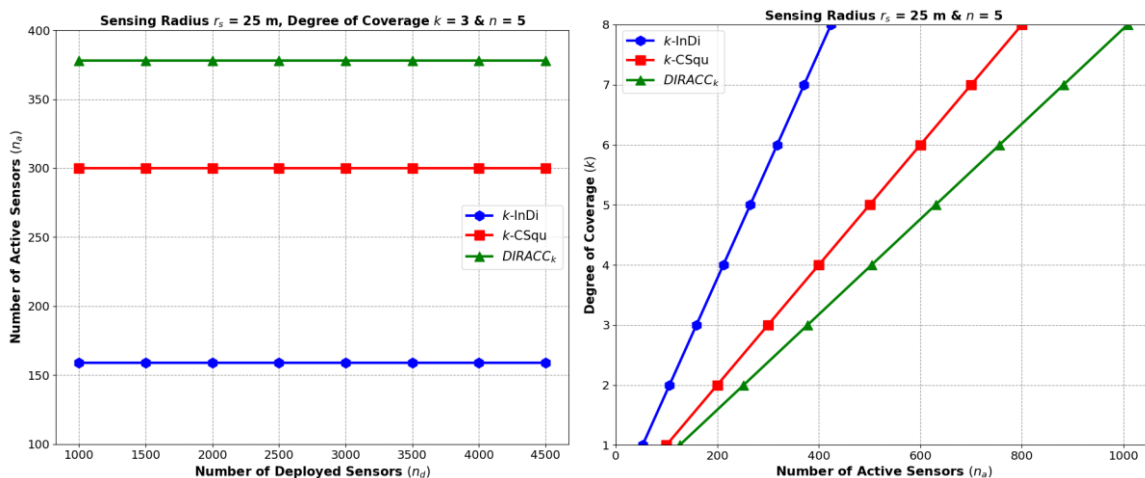


Figure 19. (a) Number of active sensors n_a versus Number of deployed sensors n_d (b) Degree of coverage k versus Number of active sensor n_a , comparing k -InDi and $DIRACC_k$

Figure 20 plots remaining energy versus time, shows the k -InDi, k -CSqu and $DIRACC_k$ protocols' energy consumption rate and operational network lifetime. Our previous hypotheses are confirmed by these results, that are deduced from the results of Figure19 (a) and (b) and is clear that the operational network lifetime of k -InDi and k -CSqu are greater than $DIRACC_k$.

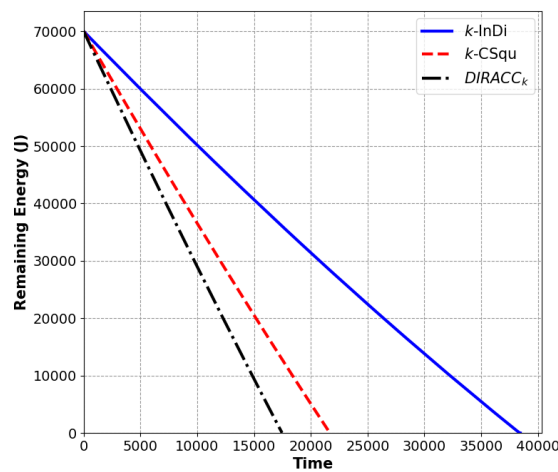


Figure 20. Remaining Energy versus Time, indicating the network lifetime

Figure 21(a) plots n_a versus r_s with $n = 5$, whereas Figure 21(b) plots n_a versus r_c with $r_s = 25$ m and $n = 5$, for different values of k . In all cases we have considered the value of k as 3 and 4. These results solidify our prior results, that k -InDi and k -CSqu offers higher level of k with fewer active sensors compared to $DIRACC_k$. Also, for our experimental conditions, k -InDi offers 4-coverage with n_a lesser than that of required for 2-coverage by $DIRACC_k$. It is worth noting that n_a depends on only r_s , n and k , even for $DIRACC_k$ and for both k -CSqu and $DIRACC_k$, n_a does not depend on the value of r_c .

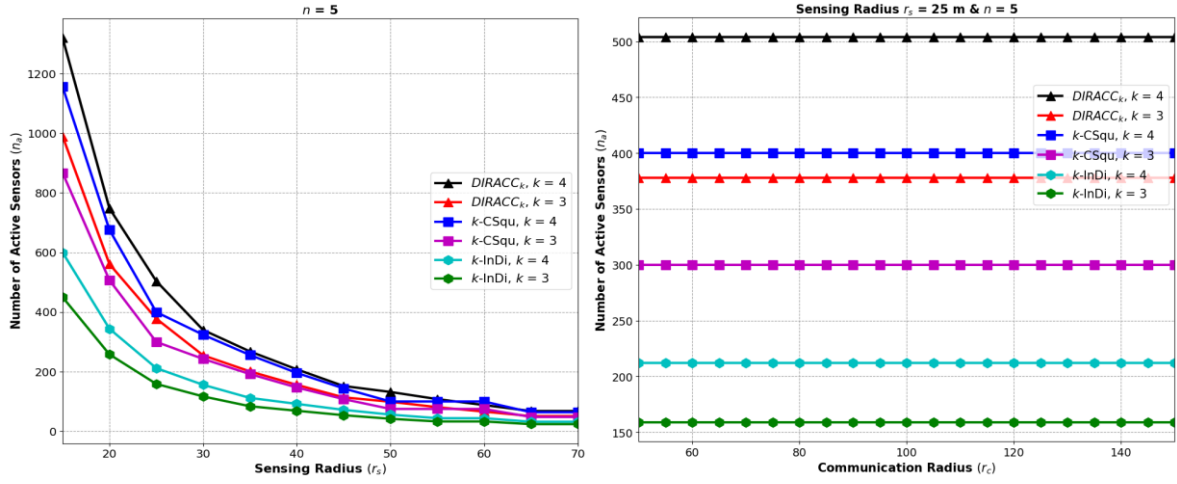


Figure 21. Number of active sensors n_a versus (a) Sensing radius r_s and (b) Communication radius r_c

CHAPTER 8. CONCLUSION

In this research, we investigate connected k -coverage problem of PWSNs using square tessellation and irregular hexagonal tessellation-based approaches. For square tessellation, we solve the issue concerning where to position sensors by constructing a cusp square area inside each square tile, and for irregular hexagonal tessellation, we address the sensor deployment problem by utilizing diamond areas in the tessellation and calculate the number of planar sensors required to maintain k -coverage of the field for each tessellation.. Then, we establish a correlation for both the sensing range and communication range of sensors necessary to keep a network up, thus, ensuring connected k -coverage pattern during the network operation. Furthermore, based on all the proved theoretical results and properties, we propose centralized k -coverage protocols, k -CSqu and k -InDi. Based on the simulation results, it is proved that k -CSqu and k -InDi have better performance and more energy efficient compared to that of DIRACC_k [14], in respect to number of active sensors required for k -cover process and network operational time, where DIRACC_k was proved to be better than CCP [21, 22] in Ammari's work [14].

Our future work has multiple directions. Firstly, we are interested in finding an optimum value of n , the proportional factor used to standardize the regular hexagonal tessellation, for generating irregular hexagonal tessellation. Second, we plan to extend our k -CSqu and k -InDi approaches to heterogeneous sensors, which have varied characteristics, as far as available initial power, sensing, and transmission ranges are concerned [24]. Third, with a more comprehensive sensing model, that is stochastic [10, 18, 20] rather than deterministic, we concentrate on generalizing our k -CSqu and k -InDi approaches to take into account the sensors irregularity of the sensing and communication ranges. Fourth, We plan to study the issue of connected k -coverage in three-dimensional WSNs, like underwater WSNs, by expanding our previously

presented theory of k -CSqu and k -InDi methods [25]. Finally, with the help of a sensor-testbed, we'll put our protocol through its paces in real-world conditions.

REFERENCES

- [1]. Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, 1(4), 660-670.
- [2]. Bulusu, N., Heidemann, J., & Estrin, D. (2000). GPS-less low-cost outdoor localization for very small devices. *IEEE personal communications*, 7(5), 28-34.
- [3]. Chenait, M., Zebbane, B., & Badache, N. (2018, October). A new k-coverage model to determine redundant sensors in wireless sensor networks. In *2018 International Conference on Smart Communications in Network Technologies (SaCoNeT)* (pp. 149-154). IEEE.
- [4]. Adulyasas, A., Sun, Z., & Wang, N. (2015). Connected coverage optimization for sensor scheduling in wireless sensor networks. *IEEE Sensors Journal*, 15(7), 3877-3892.
- [5]. Ahmadi, A., Shojafar, M., Hajeforosh, S. F., Dehghan, M., & Singhal, M. (2014). An efficient routing algorithm to preserve k-coverage in wireless sensor networks. *The Journal of Supercomputing*, 68(2), 599-623.
- [6]. Li, F., Luo, J., Wang, W., & He, Y. (2014). Autonomous Deployment for Load Balancing k-Surface Coverage in Sensor Networks. *IEEE Transactions on Wireless Communications*, 14(1), 279-293.
- [7]. Nan, G., Lian, H., & Li, M. (2013). Distributed k-Coverage Decision Scheme for System Deployment in Mobile Sensor Networks. *International Journal of Distributed Sensor Networks*, 9(10), 485250.
- [8]. Sahoo, P. K., Wueng, M. C., & Hwang, I. S. Approximate K-Coverage Configuration in WirelessSensor Networks.

- [9]. Wueng, M. C., & Hwang, I. S. (2010, September). Quality of Surveillance Measures in K-covered Heterogeneous Wireless Sensor Networks. In *2010 39th International Conference on Parallel Processing Workshops* (pp. 586-593). IEEE.
- [10]. Ammari, H. M. (2009, August). Stochastic k-coverage in wireless sensor networks. In *International Conference on Wireless Algorithms, Systems, and Applications* (pp. 125-134). Springer, Berlin, Heidelberg.
- [11]. Liu, Y., Pu, J., Zhang, S., Liu, Y., & Xiong, Z. (2009). A localized coverage preserving protocol for wireless sensor networks. *Sensors*, 9(1), 281-302.
- [12]. Wueng, M. C., Hwang, S. I., & Ho, C. H. (2008, September). Akce: An efficient and accurate k-coverage eligibility algorithm in wireless sensor networks. In *2008 IEEE International Symposium on Modeling, Analysis and Simulation of Computers and Telecommunication Systems* (pp. 1-8). IEEE.
- [13]. Xing, G., Wang, X., Zhang, Y., Lu, C., Pless, R., & Gill, C. (2005). Integrated coverage and connectivity configuration for energy conservation in sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 1(1), 36-72.
- [14]. Ammari, H. M., & Das, S. K. (2011). Centralized and clustered k-coverage protocols for wireless sensor networks. *IEEE Transactions on computers*, 61(1), 118-133.
- [15]. Ammari, H. M. (2016). A unified framework for k-coverage and data collection in heterogeneous wireless sensor networks. *Journal of Parallel and Distributed Computing*, 89, 37-49.
- [16]. Yu, J., Wan, S., Cheng, X., & Yu, D. (2017). Coverage contribution area based k-coverage for wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 66(9), 8510-8523.

- [17]. Qiu, C., Shen, H., & Chen, K. (2017). An Energy-Efficient and Distributed Cooperation Mechanism for k-Coverage Hole Detection and Healing in WSNs. *IEEE Transactions on Mobile Computing*, 17(6), 1247-1259.
- [18]. Yu, J., Ren, S., Wan, S., Yu, D., & Wang, G. (2012). A stochastic k-coverage scheduling algorithm in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 8(11), 746501.
- [19]. Ammari, H. M. (2019, November). Achieving Sensing k-Coverage Using Hexagonal Tiling: Are We Done Yet?. In *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)* (pp. 73-81). IEEE.
- [20]. Ammari, H. M. (2021). Connected k-coverage in two-dimensional wireless sensor networks using hexagonal slicing and area stretching. *Journal of Parallel and Distributed Computing*, 153, 89-109.
- [21]. Wang, Y. C., & Tseng, Y. C. (2008). Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage. *IEEE Transactions on Parallel and Distributed Systems*, 19(9), 1280-1294.
- [22]. Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., & Gill, C. (2003, November). Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems* (pp. 28-39).
- [23]. A Wireless Sensor Network simulator in Python and C++ (via SWIG). <https://github.com/darolt/wsn/>, 2022 (accessed 9 September 2022)
- [24]. Yarvis, M., Kushalnagar, N., Singh, H., Rangarajan, A., Liu, Y., & Singh, S. (2005, March). Exploiting heterogeneity in sensor networks. In *Proceedings IEEE 24th Annual Joint*

Conference of the IEEE Computer and Communications Societies. (Vol. 2, pp. 878-890).
IEEE.

- [25]. Pompili, D., Melodia, T., & Akyildiz, I. F. (2006, September). Deployment analysis in underwater acoustic wireless sensor networks. In *Proceedings of the 1st ACM international workshop on Underwater networks* (pp. 48-55).

VITA

Venkata Swamy *Kalyan* Nakka

Kalyan is an International student from India. He attained his undergraduate degree, Bachelor of Technology in Mechanical Engineering, with first class (CGPA: 3.67/4.0) from IIT (ISM)-Dhanbad, India in 2016. After his undergraduate studies, he worked in multiple companies from 2016 to 2021 in Software and Information Technology domain. Later in 2021, he moved to Texas, for pursuing his graduate studies in Computer Science at Texas A&M University-Kingsville. He is currently working as Graduate Research Assistant in the Department of Electrical Engineering and Computer Science at Texas A&M University-Kingsville and conducting research at Wireless Sensor and Mobile Ad-hoc Networks, Internet of Things, and Applied Cryptography Engineering, (WiSeMAN-IoT-ACE) Research Lab of Texas A&M University-Kingsville. He also worked as Graduate Teaching Assistant in the Department of Electrical Engineering and Computer Science for Foundations of Computer Science and Massive Parallel Computing courses. In particular, he is interested in the research areas of Computational Geometry, Machine Learning and Artificial Intelligence.

ProQuest Number: 30420280

INFORMATION TO ALL USERS

The quality and completeness of this reproduction is dependent on the quality and completeness of the copy made available to ProQuest.



Distributed by ProQuest LLC (2023).

Copyright of the Dissertation is held by the Author unless otherwise noted.

This work may be used in accordance with the terms of the Creative Commons license or other rights statement, as indicated in the copyright statement or in the metadata associated with this work. Unless otherwise specified in the copyright statement or the metadata, all rights are reserved by the copyright holder.

This work is protected against unauthorized copying under Title 17, United States Code and other applicable copyright laws.

Microform Edition where available © ProQuest LLC. No reproduction or digitization of the Microform Edition is authorized without permission of ProQuest LLC.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346 USA