



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJIRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 13, April 2023

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

9940 572 462

6381 907 438

[ijirset@gmail.com](mailto:ijirset@gmail.com)

[www.ijirset.com](http://www.ijirset.com)

# Novel Honeycomb-Based Wireless NoC Architecture in Many-Core Era

Mr. T. Prasanth<sup>1</sup>, Dr.O.Cyril Mathew<sup>2</sup>, Mr. K.K.Kavin<sup>3</sup>, Ms.S.Shruthi<sup>4</sup>,  
Mrs.F.Jessy Nirmala<sup>5</sup>

**ABSTRACT:** In mission critical applications of wireless sensor networks (WSNs), multiple sinks can be associated to first responders such as firefighters, but also to unmanned aerial vehicles (UAVs). In such scenarios, data dissemination of events towards mobile sinks should be performed reliably. In this paper we present Honeycomb Architecture which enables data dissemination considering dynamic conditions of multiple sinks and sources. Honeycomb Architecture exploits a virtual infrastructure called ‘highways’, which is an area where all event data are cached. The ‘highways’ act as rendezvous regions of the events and queries. Once a query is issued, it is sent to one of the ‘highways’ and searches relevant data stored in the ‘highway’. When the data is found, it is sent to the sink which has issued the query. Our data dissemination protocol is fault-tolerant, i.e., it can bypass holes in the network. We evaluate and compare the data delivery ratio and latency of our data dissemination protocol with previous approaches. We also analyze the hotspot regions in the network with different protocols. Simulation results show that our work significantly reduces overall energy consumption while maintaining comparably high data delivery ratio.

## I. INTRODUCTION

The primary goal of a wireless sensor network (WSN) is to collect useful information by monitoring phenomena in the network. In WSNs, each sensor individually senses the environment, but collaboratively achieves complex information gathering and dissemination tasks. Typically a WSN follows the communication pattern of convergecast, where sensors collect data about a phenomenon and relay streams of data to a common sink. Data dissemination is a preplanned way of distributing queries and data among the nodes.

Our work is motivated by disaster management scenarios where the deployment of the sensors is performed in a random fashion, e.g., dropping sensors from UAVs flying above the field [1]. Wireless sensors might be attached to mobile objects. In such scenarios, UAVs, emergency responders, e.g. firefighters, or vehicles, e.g. firetrucks, carry sink nodes on-board. These mobile sinks are used to collect more reliable data about the event in the dangerous/inaccessible regions. In this scenario, which was under the consideration of the European research project AWARE [2], we focus on data dissemination of emergency messages towards mobile sinks. In this context it is needed to achieve two main goals: (i) To accommodate the dynamics of the WSN due to stimulus and sink mobility, in such a way that avoids excessive updates caused by frequently changing environment, (ii) To detect routing holes and establish alternative forwarding paths.

For these purposes, we present the Honeycomb Architecture and our geographical routing protocol Hexagonal Cell-Based Data Dissemination (HexDD) for Mobile Multi-Sink WSNs. Our proposed scheme is based on a virtual infrastructure proposing rendezvous regions for events (data caching) and queries (look-up). It supports both mobility of sinks and sources. We evaluate and compare the data delivery ratio and latency of HexDD protocol with previous approaches. We also analyze the hotspot regions in the network with different protocols. Simulation results show that our work significantly reduces overall energy consumption in the network while maintaining comparably high data delivery ratio.

The rest of this paper is organized as follows: The related works are introduced with their strengths and weaknesses in Section 2. In Section 3 we present the honeycomb architecture and basic operations of HexDD. Section 4 gives the simulation results to evaluate the performance of the proposed protocol. Finally, Section 5 concludes the paper.

## II. RELATED WORK

Several data dissemination protocols have been proposed for mobile WSNs. The proposed protocols fall in two major categories: (i) Flooding-based and (ii) Virtual infrastructure-based. As the flat architectures and flooding-based protocols do not scale, overlaying a virtual infrastructure over physical network often has been

investigated as an efficient strategy for data dissemination in mobile WSNs [3]. In general, virtual infrastructure-based protocols can be divided into (i) backbone-based approaches (e.g. [4]), and (ii) rendezvous-based approaches (e.g. [5]) depending on how the virtual infrastructure is formed by the set of potential storing nodes. Since we also propose a rendezvous-based protocol, we mainly discuss rendezvous-based approaches [6], [7] in this section.

In TTDD [6], each source node builds a uniform virtual grid structure throughout the sensor field. A sink floods a query within its local grid cell. The query then propagates along the grid to reach the source node. While the query is disseminated over the grid, a reverse path is established towards sink and data is sent to the sink via this reverse path. If the stimulus is mobile, number of sources and grids increase. This situation can lead to excessive energy drain, and therefore limit the network lifetime. LBDD [7] defines a vertical line that divides the sensor field into two equal sized parts. This wide line acts as a rendezvous area for data storage and look-up. When a sensor detects a new event, it transmits a data report towards the nodes in the virtual line. This data is stored on the first node encountered in the virtual line. To collect the generated data reports, the sink sends its query toward the rendezvous area. This query is flooded along the virtual line until it arrives to the inline node that owns the requested data. Thus, data reports are sent directly to the sink. However, using a line as rendezvous area at the middle of the network can result in high latency for the nodes near the boundary of the network. Although there are many proposals for data dissemination in WSNs with mobile sinks, most of them suffer from high communication cost resulting in high energy consumption as the number of queries and data increases. Also, most of the previous works do not discuss how to maintain the virtual infrastructure if there are holes, a large space without sensors, which is a common behavior in any real WSN deployment.

### III. HONEYCOMB ARCHITECTURE

In this section, we describe how the physical network is partitioned into virtual hexagonal cells in the Honeycomb imposes a two dimensional hexagonal grid structure. Using hexagonal cells in order to divide the network into sub-regions is a simple and useful method for geographical routing since a node in a given hexagonal-cell knows all its neighbors with their associated-cell addresses that are the indications of which direction a neighboring node lies on. The architecture also defines three principle diagonal virtual lines – ‘highways’ (or ‘border lines’) – which divide the sensor field into six parts. The lines, which intersect at the center of the network, are used as rendezvous regions for the queries and the data.

Hexagonal cells were used in literature for various applications [4], [9], [10]. It is important to point out that in this paper, we use hexagonal cells to propose a novel rendezvous based fault-tolerant routing protocol for supporting mobility of sinks in a WSN. In this protocol, we don't assume a regular topology in which there is one node on every corner of a hexagonal cell [9]; instead, we assume a random deployment.

To begin with, we indicate some assumptions for virtual honeycomb architecture. To enable geographic routing, every node has its location information as also assumed in [4]- [7]. Through periodic beacon packets, a sensor can learn the location and cell of its neighbors. Second, all the nodes know the coordinates of the center of the network. It is required to form rendezvous region at the setup stage. A simple method to get the coordinates of the center of the network is given in [5]. Third, there are multiple sinks moving randomly in the sensor field. Sinks are equal from the information point of view; it does not matter to which sink data is sent. This section introduces operations of our protocol. The first phase is Hexagonal Cell-Based Network Partitioning. Honeycomb cells and rendezvous areas are formed for the honeycomb architecture in the first phase. This phase is performed in the network setup. After it is finished, the network becomes ready to execute Hexagonal Cell-Based Data Dissemination (HexDD) protocol in case an emergency message, e.g. fire alarm, is generated somewhere in the network.

#### A. Hexagonal Cell-Based Network Partitioning

1) Honeycomb Cell Structure: Firstly, a honeycomb cell structure is presented in comparison with the traditional rectangular grid. It is assumed that the sensor field be overlaid with a honeycomb virtual mesh based tessellation. A honeycomb mesh is shown in Fig. 1. Each cell has six neighbors covering the surroundings from all directions. For two adjacent cells, every node in one cell can communicate with all the nodes in the other cell. The honeycomb mesh has a nice property that for a given cell, all of its next hop cells are also adjacent cells. They have the same maximum distance to this cell. Honeycomb architecture provides grid homogeneity in sense that



there exists no neighboring cell that shares a corner.

The longest distance between two adjacent cells is  $l = 13r$ , where  $r$  is the edge length of the hexagon. In order for all the nodes in adjacent cells to be able to communicate with each other, the longest length must satisfy  $l = \sqrt{13}r \leq R$ , where  $R$  is the transmission range. Therefore, we choose the edge length of the hexagon,  $r = R / \sqrt{13}$ , such that sensors in adjacent cells are within communicable distance of each other. At the first step, with the given edge length of a hexagon,  $r$ , each sensor node in a cell uses its location information to associate itself with a honeycomb virtual cell having a name of  $[i, j]$ . Fig. 1(a) shows the cell naming in honeycomb architecture. Next, we transform the cell names of the form  $[i, j]$  into special cell addresses of the form  $[H, I]$ . The addresses of the honeycomb cells are used in the data dissemination. The details of honeycomb cell-node association [10] and cell name-address transformation can be found in [11]. The algorithms for cell-node association and cell name-address transformation are lightweight in computing. There is no communication overhead. Each node executes the algorithms locally.

- 1) Construction of Border Lines and Hextants: Our honeycomb architecture defines three principle diagonal lines labeled as  $l$ ,  $b$ , and  $r$  which are drawn through the origin of center cell, as illustrated in Fig. 2. These lines divide the sensor field into six regions which are named as Hextants. Each of six hextants is marked with roman numerals in the figure.

All hexagonal cells on diagonal lines  $l$ ,  $b$ , and  $r$  are borders of hextants so called as border cells. These three diagonal lines act as rendezvous regions for data storage and look-up. Each half line, which starts from the origin, is the rendezvous area for the hextant which starts at this border line, assuming a counter-clockwise direction.

- 2) Cell Addressing: We assign addresses of the form  $[H, I]$  to each sensor in the same cell, where  $H$  is the shortest cell-count of the node from the origin cell and  $I$  denotes

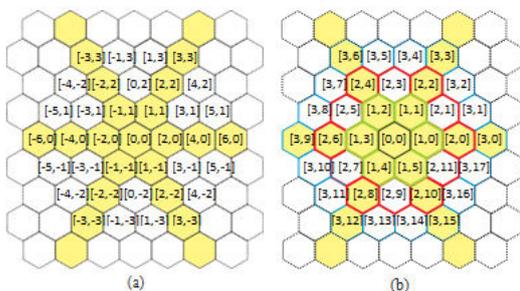


Fig. 1. (a) Cell Naming, (b) Cell Addressing in Honeycomb Architecture

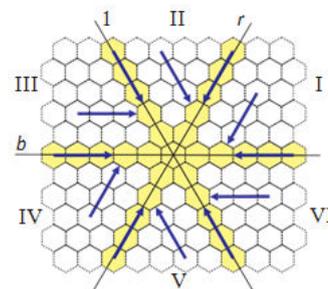


Fig. 2. Virtual Infrastructure in Honeycomb Architecture.

the index of the hop- $H$  hexagonal cell. The index starts at line  $b$  of hextant I and increases in the counter-clockwise direction. Hence, the nodes in the first-hop cells are addressed as  $[1, 0], [1, 1], \dots, [1, 5]$ . Observe that nodes of the form  $[H, \cdot]$  are all located on the same hexagonal ring at distance  $H$  from the center cell. Since the number of cells on  $H^{\text{th}}$  hop hexagonal ring is  $6H$ , the cell addresses range from  $[H, 0]$  to  $[H, 6H - 1]$ . This special addressing has some useful properties which enable to make use of the cells on the diagonal lines as rendezvous regions for data dissemination. If we define the border cells more formally, all the cells addressed as  $[H, I]$  are border cells if  $I = kH$ , where integer  $k = 0, 1, \dots, 5$ . The nodes associated with border cells are called border nodes.

### B. Hexagonal Cell-Based Data Dissemination

In our data dissemination protocol, we use the concept of central re-dissemination in which the packets flow towards the center cells following previously selected directions. Instead of sending packets directly to the center cell by using a simple geographic routing, we send data through border lines towards center cell because the aim is to store the generated data reports in the border lines such that the mobile sinks can easily collect them using a query-based data reporting method. However, our approach is purely geographical which means that we do not use flooding for route setup. The only required information is the node position which is associated with a hexagonal cell in the honeycomb architecture. With the given virtual infrastructure, we propose Hexagonal Cell-based Data Dissemination (HexDD) which has the following functions:

- 1) Data (Event) Forwarding: Event data forwarding in HexDD is done through border nodes towards center

region according to Algorithm 1-I. As shown in Fig. 2 with arrows, sensors route the packets to border cells in the first line segment of the hexant, e.g. line  $r$  for hexant II, following a direction parallel to the second line segment of the hexant,

e.g. line  $l$  for hexant II. When the data reaches one of the diagonal lines, it is forwarded along the diagonal line towards center cell. After the center cell receives data, it is directed to one of the diagonal lines according to sink's location, which is specified in the sink's query. Sensors in the border lines act as rendezvous points for data storage and look-up which means border nodes have a replica of data in their cache.

---

#### Algorithm 1 Hexagonal Cell-based Data Dissemination

---

```

1: Input:  $[H, l]$ : address of the current cell
2: Input:  $[H_s, l_s]$ : address of the sink's current cell
3: Output:  $[H, l]$  be the address of next hop cell
4: I. Find next hop cell towards center
5:    $k = \lfloor l/H \rfloor$ 
6:    $[H, l] \leftarrow [H - 1, l - k]$ 
7: II. Find next hop cell towards sink
8:    $k = \lfloor l_s/H_s \rfloor$ 
9:    $H \leftarrow H + 1$ 
10: if  $H \leq kH_s - l_s$  then
11:    $l \leftarrow l + k - 1$  // In the border line
12: else
13:    $l \leftarrow l + k$  // within the hexant
14: end if

```

---

The HexDD keeps the traffic flow in all regions of the network nearly balanced because honeycomb architecture divides the network space into six partitions and each partition uses a different border line segment for data dissemination; therefore, the traffic is spread among the different border lines.

2) Querying: In order to retrieve a specific data, a sink sends a query towards center by using the same Algorithm 1-

*I.* The first border node which receives the query forwards it towards the center cell. Each node in the border cells checks its cache when it receives a query. If the data requested is in the cache of a border node, it sends data back to sink through the reverse path. Replicating data on the border cells can decrease the cost of data look-up and the data delivery latency.

*3)* Data (Event) Delivery to Sink: To send data towards the sink, the reverse path of the sink's query forwarding path can be calculated by using the cell address of the sink as given in the Algorithm 1-II, or can be stored in the query

## IV. PERFORMANCE EVALUATION

For the purpose of performance evaluation, we have compared our protocol HexDD with two other rendezvous-based approaches LBDD and TTDD. We choose TTDD and LBDD for the comparison since we would like to investigate the effect of using hexagonal cells instead of rectangular grids and using three diagonal lines acting as rendezvous area instead of only one line-based region. The simulations have been carried out to evaluate the fault-tolerance performance of the protocols. For this purpose, we vary the total number of holes and the size of holes in the network. We also have analyzed the protocols' energy distribution maps which are important to see hotspot regions created by each protocol in the network.

### A. Simulation Environment

We have implemented and tested our protocol in NS2. To guarantee a fair comparison between TTDD and HexDD, we set simulation parameters comparable to those used in [6]. This includes simulation of IEEE 802.11 DCF as the underlying MAC and an energy model in which a sensor's transmitting, receiving and idling power consumptions are set to 0.66W, 0.395W and 0.035W, respectively. The cell size in TTDD is set to 600 m. In



LBDD, width of the virtual line is set to 250 m. Each node has a transmission range of 250 m and 210 nodes are randomly distributed on a 1500 1500 m<sup>2</sup> field. Each simulation run lasts for 200 seconds, and each result is averaged over six random network topologies. A source generates one data packet per second, so there are in total 200 data packets/source sent. Sinks’ mobility follows the standard Linear Mobility model. Mobile sinks could attain a maximum speed up to 14 m/s with 5 seconds pause time. The stimuli remain static during the simulation time.

We used the following metrics to evaluate the performance of the protocols: (i) Average Data Delivery Ratio: defined as the ratio between the total number of data packets received by the sinks and the total number of data generated by the sources; (ii) Average Delay: defined as the total time elapsed between the data generation by a source and its reception by a sink, also averaged over all source-sink pairs.

**B. Simulation Results**

1) **Impact of the number of hole regions in the network:** Each different shaped and middle sized holes covering 5 cells in the network is randomly generated and positioned on the hextants or on the border lines. The number of sources is equal to the number of holes in the network. Each source is placed on a location where it is affected by at least one hole. The 3 destination sinks are chosen randomly in the network.

Fig. 3(a) shows the impact of increasing number of holes on the average data delivery ratio (DDR). Fig. 3(b) presents the average data delivery delay. We observe that for all protocols when we increase the number of holes in the network, the DDR decreases. In TTDD, routing crosses over the holes for most of the data packets. TTDD does not have a specific hole recovery mechanism; however, it builds grids and its routing strategy is along the grid so if some parts of the grid is missing, there are still alternative paths on the grid to forward packets towards sinks. However, TTDD has the highest delay since the alternative paths along the grids to bypass holes are in most of the cases longer than the possible shortest path between sources and sinks. In LBDD, on the other hand, more packets get stuck in the holes when we increase the number of holes. The packets in LBDD are always directed to one single strip vertically positioned at the center of the network and for the packets coming across holes, greedy forwarding sometimes fails to route across the holes. Since a data packet is forwarded along the boundary of a hole in LBDD until greedy forwarding becomes possible again, it has the lowest delay. We also observe a slight decrease in the delay of LBDD since when we increase the number of holes, the number of packets routed across the holes decrease and the packets routed across the holes experience shorter paths. The special hole recovery mechanism in HexDD can also achieve a high DDR. It also tries to forward packets along the boundary of a hole via the shortest possible path so it also has a low delay.

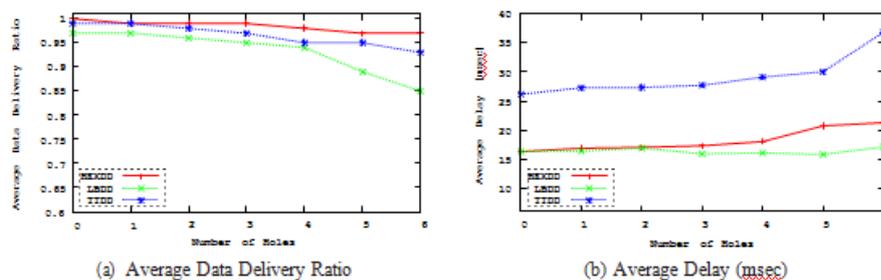


Fig. 3. Impact of increasing number of holes in the network

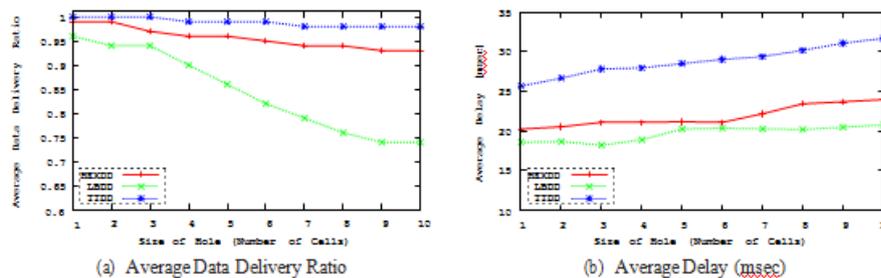


Fig. 4. Impact of increasing size of the hole in the network

- 2) **Impact of the size of holes in the network:** In this set of simulations, there are one source-sink pair and one hole in the network. The size and the shape of the hole is changed for different runs. The size of the hole is represented as the number of cells that the hole covers.

Fig. 4(a) shows the average DDR and Fig. 4(b) presents the average delay for increasing hole size. The grid structure of TTDD allows the data packets to find other paths to bypass hole even when we have a large hole in the network so DDR of TTDD is the highest. However, the delay of TTDD is also the highest, because the alternative paths along the grid may be much more longer than the path along the boundary of a hole. When we have larger holes, the DDR of LBDD rapidly decreases since it is getting harder for greedy forwarding to find a path along the boundary of a large hole. More packets get stuck in the hole, e.g., 26% of the packets get lost in the hole covering 10 cells as shown in Fig. 4(a). That is why also the delay of LBDD is the lowest. Simulations show that the hole recovery mechanism in HexDD works efficiently since the DDR of HexDD is high. On the other hand, the data delivery delay of HexDD is much more shorter than that of TTDD because HexDD tries to go along the boundary of the hole and finds the shortest possible path.

## V. CONCLUSIONS

In this paper, our goal was designing a routing protocol which supports mobility of sink and source by keeping the data delivery ratio as high as possible and energy consumption and delivery delay as low as possible. We proposed a virtual infrastructure called Honeycomb Architecture which allows an efficient geographical routing of event messages, namely HexDD. The HexDD uses the concept of rendezvous region for events and queries. The border lines used as rendezvous areas, which lie on every direction of the network, make it faster for sinks to access data. In addition, HexDD makes the system resistant to node failures in the virtual infrastructure and help in quick routing hole recovery in the network. The simulation results show that our architecture helps to minimize overall energy consumption and keeps the data delivery ratio high even when routing holes exist in the network. To avoid the hot region problem which may be observed in the border lines and the central cells, one solution is to adjust the size of the border lines and shape of the central region according to the size of the network and the network traffic. As recent studies have been exploiting heterogeneity in the WSNs, deployment of higher energy and communication capacity nodes can be used at the center of the network to leverage the overall system capability of HexDD. Deploying more nodes to these regions.

## REFERENCES

- [1] A. Erman-Tuysuz, L. van Hoesel, P. Havinga, and J. Wu, "Enabling mobility in heterogeneous wireless sensor networks cooperating with UAVs for mission-critical management," *IEEE Wireless Communications*, vol. 15, no. 6, pp. 38–46, 2008.
- [2] A. Ollero, M. Bernard, M. L. Civita, L. van Hoesel, P. Marron, J. Lepley, and E. de Andres, "Aware: Platform for autonomous self-deploying and operation of wireless sensor-actuator networks cooperating with unmanned aerial vehicles," in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR 2007)*, Rome, 2007, pp. 1–6.
- [3] E. Hamida and G. Chelius, "Strategies for data dissemination to mobile sinks in wireless sensor networks," *IEEE Wireless Communications Magazine*, vol. 15, no. 6, pp. 31–37, December 2008.
- [4] X. Chen and M. Xu, "A geographical cellular-like architecture for wireless sensor networks," in *Proceedings of the 1st International Conference Mobile Ad-hoc and Sensor Networks (MSN 2005)*, Wuhan, China, December 2005, pp. 249–258.
- [5] J.-H. Shin, J. Kim, K. Park, and D. Park, "Railroad: virtual infrastructure for data dissemination in wireless sensor networks," in *PE-WASUN '05: Proceedings of the 2nd ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*. Montreal, Quebec, Canada: ACM, 2005, pp. 168–174.
- [6] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A two-tier data dissemination model for large-scale wireless sensor networks," in *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*. Atlanta, Georgia, USA: ACM, 2002, pp. 148–159.
- [7] E. B. Hamida and G. Chelius, "A line-based data dissemination protocol for wireless sensor networks with mobile sink," in the *IEEE International Conference on Communications (ICC 2008)*. Beijing, China: IEEE, May 2008.
- [8] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in



MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking. Rome, Italy: ACM, 2001, pp. 70–84.

- [9] A. Shariéh, Q. Mohammad, W. Almobaideen, and A. Sliet, “Hex- cell: Modeling, topological properties and routing algorithm,” Europeanjournal of Scientific Research, vol. 22, no. 2, pp. 457–468, 2008.
- [10] R. P. Liu, G. Rogers, and S. Zhou, “Honeycomb architecture for energy conservation in wireless sensor networks,” in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM 2006), San Francisco, CA, USA, November 2006, pp. 1–5.



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  [ijirset@gmail.com](mailto:ijirset@gmail.com)



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details