

Position and hop-count assisted full coverage control in dense sensor networks

Abhiram Singh · T. P. Sharma

Published online: 13 September 2014
© Springer Science+Business Media New York 2014

Abstract In recent years, wireless sensor networks (WSN's) have gained much attention due to its various applications in military, environmental monitoring, industries and in many others. All these applications require some target field to be monitored by a group of sensor nodes. Hence, coverage becomes an important issue in WSN's. This paper focuses on full coverage issue of WSN's. Based on the idea of some existing and derived theorems, Position and Hop-count Assisted (PHA) algorithm is proposed. This algorithm provides full coverage of the target field, maintains network connectivity and tries to minimize the number of working sensor nodes. Algorithm works for communication range less than root three times of sensing range and it can be extended for arbitrary relation between communication range and sensing range. By using hop-count value, three-connectivity in the network is maintained. Also, neighbors information is used to create logical tree structure which can be utilized in routing, redundant data removal and in other areas. Simulation results show that PHA algorithm outperforms layered diffusion-based coverage control algorithm by providing better area coverage and activating fewer nodes.

Keywords Wireless sensor networks · Coverage · Connectivity · Full coverage

1 Introduction

Wireless sensor networks (WSN's) consist of a number of sensor nodes communicating through wireless channel. Each sensor node is a converter which measures a physical quantity and gives output in a format which can be read by an observer or by an instrument. Due to small size, self-configuration capability and low cost of sensor nodes, WSNs have many applications such as in the development of smart homes [1], in health care [2, 3], military [4], environment monitoring [5], forest fire detection [6] and in many others [7].

In all the applications, proper coverage of target field is very important. For a given sensor network, coverage measures how closely target area is observed by the sensor nodes. Formally, coverage denoted by C is defined as the mapping from a set of sensors S , to the set of area A , in the given field of interest F or Coverage C is defined as a mapping $C : S \rightarrow A$ such that $\forall s_i \in S, \exists! a_i \in A$ and overlapping between each a_i is removed, where: S is the set of sensor, denoted by s_i , and identified by their coordinates (x_i, y_i) , A is the set of area denoted by a_i which are identified by calculating $\pi * Rs_i^2$, Rs_i is the sensing area of sensor s_i , F is the given area which has to be monitored, C is termed as coverage which maps S to A and the cardinality of S and A are same.

Full coverage of the target area means each point within the area is monitored by at-least one sensor node. Full coverage is also known as area coverage and blanket coverage. Full coverage may also be defined as: Given field of interest F , covered in such a way that every point p_i in F is covered in the area $a_i \in A$ of some sensor $s_i \in S$, which is selected by mapping C .

Connectivity can be defined as the ability of sensor nodes to communicate with the sink node or any other

A. Singh (✉) · T. P. Sharma
Department of Computer Science and Engineering, National
Institute of Technology Hamirpur, Hamirpur 177005, India
e-mail: abhiram25.1990@gmail.com

T. P. Sharma
e-mail: teek@nith.ac.in

active sensor nodes in the network directly or by multi-hop transmission. k -connectivity in a network is defined as the ability of any active sensor node to communicate with other active nodes in the network, even if it's $(k - 1)$ active neighbors are failed.

Due to design constraints, sensor nodes are equipped with small battery and hence they have limited energy [8, 9]. Therefore, any algorithm developed for providing full coverage, tries to minimize number of active sensor nodes.

In this paper, three issues of sensor networks are considered: full coverage, network connectivity and active node minimization. Results of some existing theorems are used in the proposed work. These theorems are given with proof in the work of Zhang et al. [10], Wang et al. [11] and Gallais et al. [12]. Apart from existing theorems, some new theorems are given. It is shown that the condition *communication range is at-least twice of sensing range* is only a sufficient and not a necessary condition for ensuring that *full coverage of the target field implies connectivity in the network*. Another theorem helps to understand that how one can achieve connected sensor network with full coverage without considering the connectivity issue separately. Based on the idea of the theorems, decentralized and localized Position and Hop-count Assisted (PHA) algorithm is proposed.

The rest of the paper is organized as follows. Section 2 gives overview of existing work. Section 3 explains some theorems for providing full coverage. In Sect. 4, Position and Hop-count Assisted (PHA) algorithm is proposed. Section 5 gives the performance analysis of PHA algorithm and finally Sect. 6 gives conclusion and future work.

2 Related work

Some work in the field of network coverage and connectivity is discussed below.

Wang et al. [13] propose layered diffusion-based coverage control (LDCC) protocol. This protocol utilizes the hop count value to classify sensor nodes into different levels. In initial stage, sensors wait until some ACTIVE message is received. After receiving the message, sensors start their timer and wait to receive some more ACTIVE message. Sensor node becomes active after timeout period and sends ACTIVE message with its hop-count value. Neighbor node after receiving the message, compare its hop-count value with hop-count value received in the message and then based on the comparison result either resets its timer or enters in sleep state. In this method if sensor receives two messages from the sensors of the same level, then it changes its state to sleep. LDCC protocol ensures the full coverage and connectivity. However, full

coverage detection scheme and active node minimization scheme is not utilized properly.

Zhang et al. [10] give optimal geographical density control (OGDC) algorithm for dense sensor network. The authors have given different theorems to suggest best possible geometrical structure in which number of active sensor nodes are minimum. The algorithm defines three states for sensor nodes undecided, on and off. Based on the threshold, one node activates and sends the power on message. Then, all neighbors set their timer in such a way that node at a distance of $\sqrt{3}$ times of sensing range (R_s) gets activated. If the node is not present at $\sqrt{3}R_s$ distance, then node close to this distance gets high priority. In this way, sensor nodes are selected for activation in different rounds. OGDC algorithm is able to select very few sensor nodes in high density of sensor nodes, but in the area of high noise, positioning error is high and number of working sensor node increases [13].

Wang et al. [11] propose coverage configuration protocol (CCP). This protocol solves the k -coverage problem which demands that each point of the field is monitored by at-least k sensor nodes. To find the relation between communication and sensing range, Voronoi diagram is used. In this protocol to collect the information of the neighbors HELLO messages are used. Three states of nodes are defined listen, active and sleep. At any duration of time node detects that its sensing area is covered by its neighbors, then it enters in the sleep state for a fixed time. On wake up it again checks its sensing area and then takes the decision to go in sleep state or in active state. The problem with this protocol is that it does not ensure connectivity when communication range is less than twice of sensing range.

Approach of Tian et al. [14] use sponsored area technique to provide full coverage. The maximum area covered by the neighbors of a sensor node is termed as sponsored area. When a sensor node receives some message from its neighbors, then it calculates sponsored area. After calculating sponsored area if node detects that sponsored area of adjacent sensor nodes fully cover its sensing area, then the sensor node decides to go into a sleep state. By calculating sponsored area, only full coverage is addressed without considering connectivity issue.

Fan et al. [15] present probing environment and adaptive sleeping (PEAS) algorithm. This algorithm is position independent and range free. So, it has less effect on coverage maintenance due to positioning error. This algorithm assumes that sensor nodes can change their communication range after the deployment. From deployed sensor nodes, the algorithm selects some of them for full coverage of the field. The remaining nodes are kept in a sleep state for a limited time. On wake up, sensor node detects whether

adjacent sensor nodes are working in its area or not. If it cannot discover any node, then it changes its state to active, otherwise it again sets its state to sleep. This algorithm provides connected network, but does not focus on the full coverage issue.

Sibbineni et al. [16] suggest a Scalable selfConfigurable and Adaptive REconfiguration scheme (SCARE) for dense sensor network. This method classifies the sensor nodes into coordinators and non-coordinators. Coordinators are sensor nodes in active state which are used for field monitoring and non-coordinator sensor nodes are kept in the sleep state. Coordinators are selected from some threshold. When node becomes coordinator, it waits for random time and then sends a HELLO message. Nodes which are not coordinator, on receiving this message, calculate distance from the coordinator and based on some comparison, decide whether to become a coordinator or non-coordinator. Authors only solve network connectivity issue without considering full coverage issue.

More related work is given in [17–19]. Positioning is a very important and closely related to the coverage issue. Description of various localization algorithms is given in the work of Nazir et al. [20], Guangjie et al. [21]. Some data collection and aggregation techniques can be found in [22–25]. Survey on ietf protocol and duty cycled WSN's can be found in [26, 27].

3 Theorems used in the algorithm

Some existing theorems used in the development of PHA algorithm are given below.

Theorem 1 For ensuring full coverage in a region, one can minimize the number of active sensor nodes by minimizing the overlapping sensing area of all active sensor nodes under the condition that all sensor nodes have same sensing area [10].

According to this theorem, overlapping of sensing area is just re-monitoring of the same set of field points say P_i 's. So one can shift the sensor node away to cover a new set of field points P_j 's. Otherwise, some extra sensor will be required to monitor the P_j set of points.

Theorem 2 To provide full coverage with minimum overlap, active sensor nodes should be placed according to the equilateral triangle pattern having side length $\sqrt{3}R_s$ (Fig. 1), where R_s is the sensing radius of the sensor node [10].

According to this theorem, triangular pattern having side length of $\sqrt{3}R_s$ is the best possible geometrical structure for sensor placement. In this condition, there is minimum

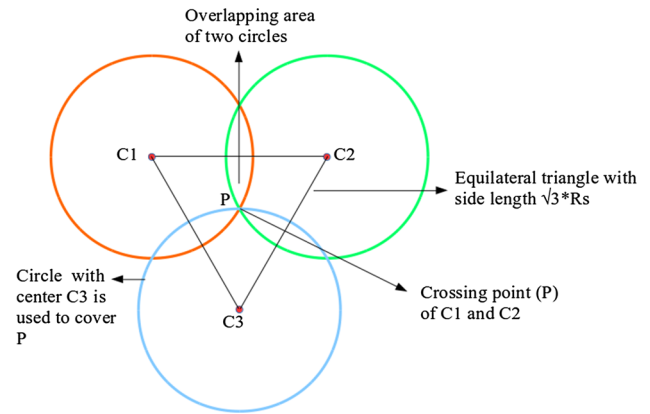


Fig. 1 Illustration of Theorem 2

overlap in sensing area of sensors and still full coverage of target field is ensured.

Theorem 3 If there are at least two covering circles and any intersection point of two covering circles inside the sensing area is covered by a third covering circle, then the sensing area is fully covered (Fig. 2) [12].

This theorem can be used by the sensor node to check full coverage of its sensing area by the active neighbor nodes.

Two new theorems with their proof are stated below.

Theorem 4 $R_c \geq 2R_s$ is not a necessary condition (however sufficient) for ensuring that *full coverage of the target field implies network connectivity*.

Proof $R_c \geq 2R_s$ provides the sufficient condition for ensuring *full coverage of the target field implies network connectivity* is already proved in the work of Wang et al. [11]. So theorem is given only for necessary condition. Initially, basics of necessary condition are discussed and then how to show some condition as not a necessary condition is given.

Suppose, condition A is necessary for B. It means, in all those cases where A is false, B must be false. However, if A is true, nothing can be said about B. In the form of logical statement, above statement can be written in the form $B \rightarrow A$, where \rightarrow is the usual implication sign. Here, A is a necessary condition for B. Its contra-positive is written in the form $\neg A \rightarrow \neg B$. However, if in some case, A is false and B is true, then A is no more necessary for B and in this case one can say that A is not a necessary condition for B. This technique is used to prove that statement $R_c \geq 2R_s$ does not provide a necessary condition for saying full coverage implies connectivity in the network.

Fig. 2 Illustration of Theorem 3: *green circle* represents the sensing area of sensor node checking for full coverage of its region, *blue circles* are sensing area of neighbors, *red dots* represent intersection points not within the sensing area of any neighbor and *yellow dots* represent intersection points within the sensing area of some neighbor. When all intersection points are within sensing range of neighbors (*yellow*) then sensing area is fully covered (Color figure online)

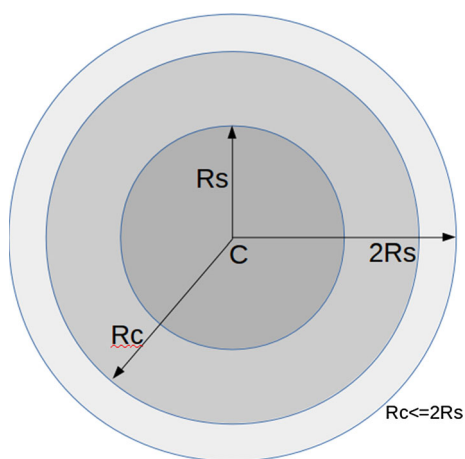
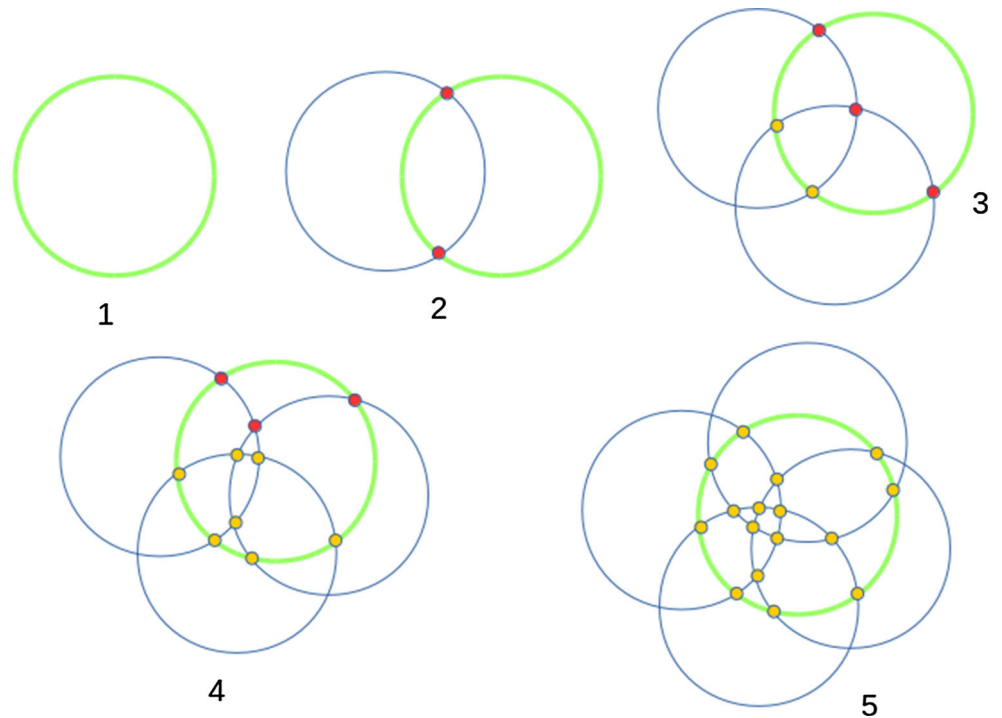


Fig. 3 Diagram representing the case where communication range is less than twice of sensing range

Assume, $R_c \geq 2R_s$ is necessary for the ensuring that full coverage of the target field ensures connectivity in the network. If $R_c < 2R_s$ (Fig. 3) then full coverage in the network should not ensure connectivity.

Case 1 Suppose sensor nodes are activated in such a way that they are just outside the communication range of node placed at C Fig. 4. In this condition, all nodes provide full coverage of the region, but the resultant network is not connected. This fulfills the requirement for the necessary condition. Figure 4 shows an example in which $R_c < 2R_s$ and full coverage is not ensuring connectivity.

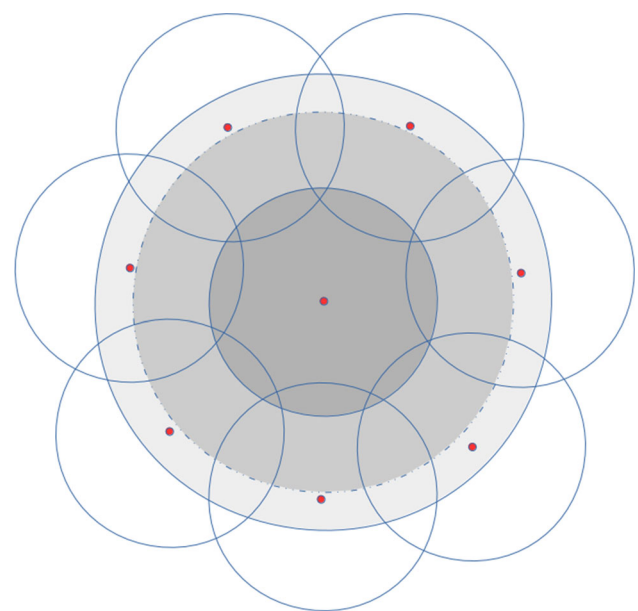


Fig. 4 Diagram representing the case in which full coverage of the area is not ensuring network connectivity when communication range is less than twice of sensing range

Case 2 Suppose sensor nodes are activated in such a way that they are inside the communication range of central node C. In this condition, all nodes are providing full coverage of the region and the resultant network is connected. This does not fulfill the requirement for the necessary condition which says if $R_c < 2R_s$, then full

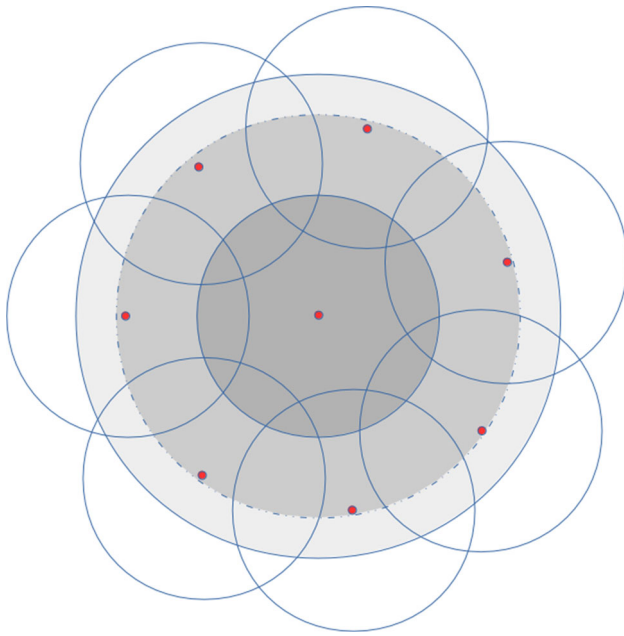


Fig. 5 Diagram representing the case in which full coverage of the area is ensuring network connectivity even if communication range is less than twice of sensing range

coverage in the network should not ensure connectivity. Figure 5 shows an example in which $R_c < 2R_s$ and still full coverage is ensuring connectivity.

Hence, $R_c \geq 2R_s$ does not provide a necessary condition for ensuring that *full coverage of the target field implies network connectivity*.

Theorem 5 Suppose, A is the wireless network whose elements are sensor nodes s_i such that all these s_i in working state form a connected network. If there is a sensor s_j (not belonging to A) is present in communication range of any sensor $s_i \in A$ and this sensor s_j is activated for network monitoring then the network A after including node s_j remains connected.

Proof Network A is connected it means each node $s_i \in A$ in the network can communicate any other node $s_k \in A$ directly or by multi-hop communication. Suppose, s_j (not belonging to A) is in communication range of some node $s_a \in A$. By the definition of connectivity, all nodes in the network can communicate with s_a and s_a in turn can communicate with s_j . So, by transitivity all $s_i \in A$ can communicate with s_a and similarly s_a can communicate with all $s_i \in A$ in the network A . Hence the theorem.

4 Algorithm

This section explains the PHA algorithm for maintaining full coverage in the target field.

4.1 Assumptions

PHA algorithm is developed under following assumptions.

1. *Nodes are homogeneous* Each node has the same processing speed, same communication and sensing range.
2. *Each node knows its position* Each sensor has inbuilt global positioning system (GPS) or some localization algorithm, which calculates its position.
3. *Sensor nodes and base-station are static* Sensor nodes and sink nodes are unable to change their position once they are deployed in the target region.
4. $R_c \leq \sqrt{3}R_s$ Communication range (R_c) is less than or equal to root three times of sensing range (R_s).
5. If timer of all sensor nodes is activated simultaneously and each sensor node selects a random value for timer, then the timer of nodes expires in the non-decreasing order of their values.
6. *If all sensor nodes in the target field are activated, then resultant network is connected* Without this full coverage of the target region cannot be obtained.

4.2 Network structure

Network structure defines nodes organization, communication medium and type of communication.

1. *Flat network* There is no clustering in the network.
2. *Dense deployment* A large number of sensor nodes are deployed in the target field.
3. *Random deployment* There is no predetermined position for node placement.
4. *Two-dimensional structure* Sensor nodes are identified by their x and y coordinates.
5. *Multi-hop transmission* Sensor nodes have message forwarding capability.

4.3 Overview

In the initial stage, all sensor nodes are in waiting state to receive some message from their neighbors. Suppose node A starts the process. It creates the activation message and sends it over wireless channel. Nodes in its communication range receive this message and based on the value contained in the message and their position, calculate their waiting time. This time is calculated in such a way that node close to the distance $\sqrt{3}R_s$, has short timer value than the nodes at higher distances. Suppose node B is at a distance of $\sqrt{3}R_s$, and at hop-count value one from node A . In this case the timeout value of B is least among all sensor nodes. When the timer expires, B changes its state to active

and sends the activation message. Now nodes at the hop count of one and two receive this message. But, priority is given to nodes with one hop-count value so that nodes can be activated according to their hop-count value.

Among nodes with one hop-count value, node at a distance close to $\sqrt{3}R_s$ from node A and B is selected for activation. In this way nodes at one hop-count value from node A change their state to active. Nodes at two and higher hop-count value are then selected in the similar way. Nodes keep on scheduling their timers based on the activation message received from neighbors. Before timeout period, if node detects that its sensing area is already covered by its active neighbors, then it changes its state to sleep.

This algorithm divides the sensor nodes into different levels according to their hop-count value from sink node (if node A is sink). The algorithm uses hop-count value to select nodes from different levels and position to activate the sensor node at proper place. For full coverage detection, sensor node uses the idea of the Theorem 3. Selection of distance $\sqrt{3}R_s$ for node activation is based on the idea of Theorem 2. The new node is always activated in communication range of an already active set of nodes so that network connectivity can be obtained as mentioned in Theorem 5. Proof of theorem 4 helps to develop an algorithm for the case where $R_c \leq \sqrt{3}R_s$. The algorithm can be extended for an arbitrary relation between R_c and R_s and Sect. 4.5.2 gives an overview of how to extend the algorithm for $R_c > R_s$.

4.4 Details of PHA algorithm

This algorithm tries to divide the sensor nodes into different levels. For classifying the nodes into different levels, their hop-count value from the sink is used. One can look sink node as the root of the tree and the remaining nodes as the children of the root node. Section 4.5.3 defines how logical tree is created for data communication when all sensor nodes have successfully executed the algorithm. After the algorithm completion, each node has its parent, children and sibling in active state. Each node after changing its state to active, sends the activation message which contains its position and hop-count value from the sink node. This algorithm is given for the case where $R_s < R_c \leq \sqrt{3}R_s$. Section 4.5.2 defines how to extend the algorithm for communication range greater than root three times of sensing range.

Starting node This algorithm assumes the sink node as the starting node. Any sensor node in the communication range of sink node can also start this process. The reason for choosing sink node as starting node is that all sensed data of sensor nodes is finally forwarded to the sink node so

that data processing can be done effectively. Starting the algorithm from the sink node will create a logical path between sink node and sensor nodes which will be useful in the data communication process.

Another option is to use any sensor node in the communication range of sink node. But then, all sensed data is received by this sensor node and then forwarded to the sink. Due to frequent message communication, energy of this sensor node is fully consumed much earlier than other sensor nodes and resultant network is disconnected. In that case, another sensor node is required to start the process again. This causes frequent execution of the algorithm and thus high energy usage. Therefore, sensor node as the starting node is not preferred and algorithm starts from sink node. If the sink is selected as starting node, then it creates an activation message containing its position and hop-count value as zero and transmits this message. Nodes in the communication range of sink (in waiting state) receive this activation message and change their state to sleep or active according to the conditions given below.

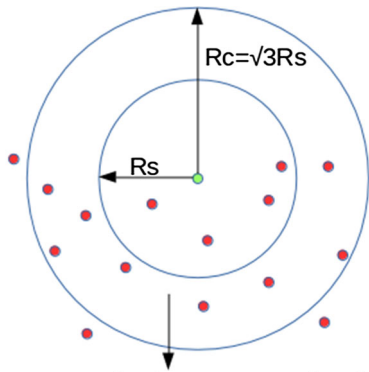
4.4.1 Message reception

When a node in waiting state receives an activation message from its neighbor, then it extracts the message parameters and calculates the distance (Euclidean) and angle from the message transmitter. The calculated distance and angle are used to provide a unique waiting time to the sensor nodes. After that, sensor node makes entry of the transmitter node (in the form of coordinates) in its neighbor list of active sensor nodes. The following conditions are used to set different waiting time of sensors and their hop-count value from the sink.

Condition 1: If received activation message is the first message of its kind (Fig. 6). In this case, set the hop-count value as one greater than the hop-count value contained in the message. Set message transmitter as the parent node by saving the coordinates of the message transmitter in a separate variable. Then calculate a unique timer value which is used to define the upper limit that a sensor must wait before changing its state to active. The timer value is calculated as follows:

$$timer1 = ((crange - distance) \times c1) + \left(\frac{angle}{c2}\right) \quad (1)$$

Where, *crange* is the communication range of sensor nodes, *distance* is the Euclidean distance of sensor from its parent, *angle* is the measured (in degree) from the positive direction of the x-axis and its range is 0–360°. For angle calculation, the parent node is shifted to origin and then sensor node according to the parent. Then, a line segment is drawn joining new sensor position to the origin. The angle made by this line segment with the positive direction of the



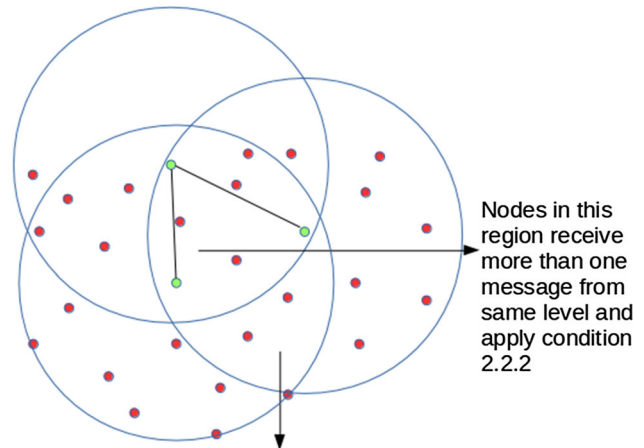
Nodes in the communication range receive their first message and activate their timer according to condition 1

Fig. 6 Diagram representing the case in which condition 1 is true

X-axis is the desired angle, $c1$ and $c2$ are positive real valued constants.

If the distance of sensor nodes is different from message transmitter, then first term $(crange - distance) * c1$ dominates and provide different timer value to different sensors. If the distance of sensor nodes is same then the term $(crange - distance) * c1$ gives the same value to different sensors. If nodes are at the same distance, then the second term $(\frac{angle}{c2})$ is used to distinguish between sensor nodes. Therefore, the combination of both terms provides the unique value of waiting time to different sensor nodes.

The reason behind calculating difference of $crange$ and $distance$ in the first term is as follows. This algorithm is given in the case where communication range is less than or equal to root three times of sensing range and according to Theorem 2, active sensor nodes must form an equilateral triangle pattern with side length root three times of sensing range. Suppose, communication range is equal to root three times of sensing range. Then, node just at the boundary of the circular area with radius equal to communication range must be activated before any other node to fulfill the condition given in Theorem 2. Term $(crange - distance)$ gives the minimum value to the node present near the boundary of the circular area with radius equal to communication range. Therefore, timer of node present at the boundary expires earlier than other nodes and this node gets chance for activation before other nodes. According to Theorem 1, less overlapping in sensing area activates less number of nodes in the target area. If the communication range is less than root three times of sensing range, then same method can be used to activate the farthest node so that overlapping between the sensing area can be minimized. However, this is not the only way to calculate timer value. One can define its own way to calculate the timeout period, but it should maintain the triangular geometry with specified parameters as given in Theorem 2.



Nodes in this region receive more than one message from parent level and apply condition 2.1 to select their actual parent.

Fig. 7 Diagram representing one case in which condition 2.1 and 2.2.2 is true

Condition 2: If received activation message is not the first message of its kind. Calculate the distance of the message receiving node with its parent and name it $ndistance$. This distance is used in timer calculation in following steps.

Condition 2.1: If the hop-count value of a node is greater than the hop-count value contained in the message and $ndistance$ is greater than distance calculated in Condition 1 (Fig. 7). In this case, the message is received from the level of the parent and message transmitter is eligible to become the parent of the message receiver. But message transmitter is actually new parent is determined by checking the distance of message receiver with its existing parent and message transmitter. If the distance to message transmitter is less than the existing parent, then coordinates of the parent are changed with the coordinates of message transmitter. This leads to the cancellation of existing timer and reschedule as done in Condition 1 but now with new parent.

Condition 2.2: If the hop-count value of the node is equal to hop-count value contained in the message. In this case, sensor node has received message from its sibling or sensor node with same hop-count value but having different parent.

Condition 2.2.1: If received activation message is first message from node with same hop-count value (Fig. 8). In this case, node cancels its previous timer and computes new waiting period $timer2$. This timer is calculated as follows.

$$timer2 = \frac{1}{distance^2 + ndistance^2 + c} \tag{2}$$

where, $distance$ is calculated at the starting of the message reception algorithm, $ndistance$ is the distance calculated at

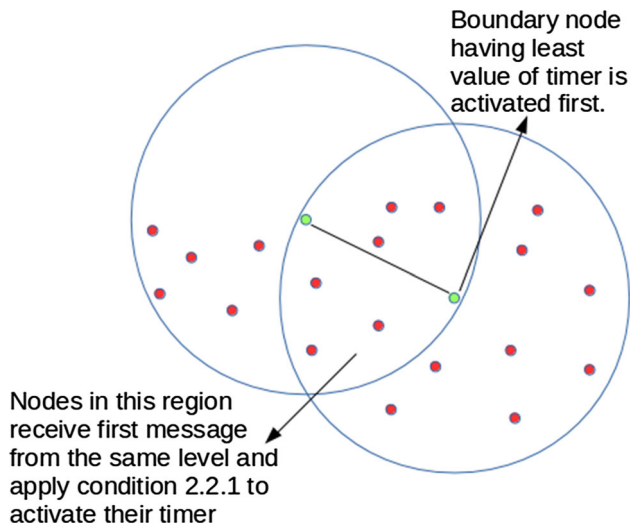


Fig. 8 Diagram representing one case in which condition 2.2.1 is true

the starting of condition 2, c is random value between 0 and 1. Reason to calculate timer value in this way is defined as follows. According to theorem 2, suppose nodes X and Y are in active state and distance between them is $\sqrt{3}R_s$. If $R_c \leq \sqrt{3}R_s$, then third node Z must be activated near the intersection point of two circles, having a radius equal to $\sqrt{3}R_s$ and circles centered at X and Y . If Z is placed at the intersection point of these two circles, then it has maximum combined distance from both X and Y . So, inversion of this combined distance gives the minimum value to the node Z and this value is treated as the timeout value. The constant c helps to avoid activation of the two nodes at the same time.

If the node at desired position is not found, then the node near to this position gets a chance for activation before any other node. One can use any other way to activate node Z but the position of Z which best approximates triangular pattern with $\sqrt{3}R_s$ side length must be selected for activation.

Condition 2.2.2: More than one activation message is received from the node with same hop-count value. In this case, the sensing area of the sensor node may be fully covered. For full coverage detection, idea of Theorem 3 is used. If the sensing area of the sensor is fully covered, then the state of sensor node is changed to sleep state for energy saving. If part of sensing area is not covered, then the sensor node waits for activation of more sensor nodes because at this stage all of its neighbors may not be activated (such as nodes with high hop-count value). If nodes with high hop-count value are unable to fully cover the sensing area of the sensor node then sensor node is activated. Waiting time of sensor node is set as mentioned in condition 1 so that all possible neighbors of sensor node are

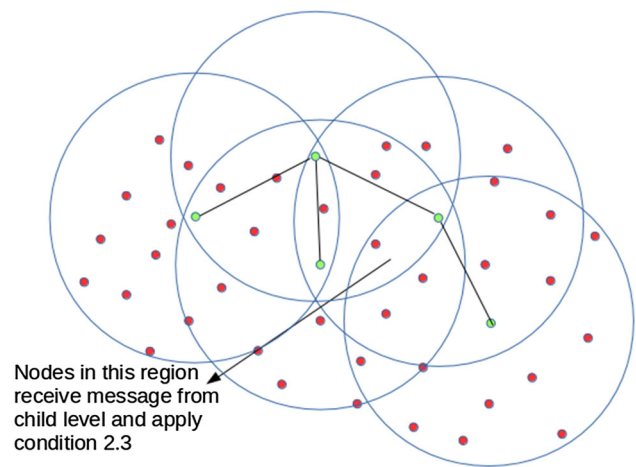


Fig. 9 Diagram representing one case in which condition 2.3 is true

activated and sensor node has enough nodes to check full coverage of its region. Figure 7 represents this case.

Condition 2.3: Activation message is received from the node with high hop-count value (Fig. 9). In this case, the node checks the condition of full coverage detection. If sensing area is fully covered, then the state of sensor node is changed to sleep otherwise sensor node extends its timer value to receive some more activation messages. Timer value is set as in condition 2.2.2.

4.4.2 Message transmission

After timeout period, sensor node changes its state to active and creates an activation message. The activation message contains the position of the sensor node in the form of its x and y coordinates and its hop-count value from the sink node. This activation message is then transmitted to the nodes in its communication range. Nodes which receive this message if not in active or sleep state, executes message reception algorithm.

4.5 Discussion

First part of this section describes how the PHA algorithm provides 3-connectivity in the sensor network. Second and third part of the section are extension of the algorithm and provides only a brief description about how to set the timer value if $R_c > \sqrt{3}R_s$ and how to form the logical tree among the sensor nodes.

4.5.1 3-connectivity

PHA algorithm achieves 3-connectivity in the sensor network by activating at-least one parent, sibling and child of the sensor node. The validity of this statement is as follows. When sensor node activates at that time it has received

at-least one message from its parent. So, in this case node is connected to only one node. When it sends the activation message, then two nodes with the same hop-count value activates in its communication range. These two nodes are on left and right side of the sensor. So, now it is 3-connected. The sensor node receives more activation messages from nodes with high hop-count value. So, connectivity of active sensor node further increases. Boundary nodes are not considered because they may not have any nodes on its left, right side or even child nodes may not be present. In this case, condition of 3-connectivity is not fulfilled. The meaning of left and right side is that when a line segment is drawn from the sensor node (A) to its parent (B) and if position of new sensor nodes is checked from the sensor node A then one node is on the left side of the line segment AB and another on the right side of the line segment AB.

4.5.2 $R_c > \sqrt{3}R_s$

Only short description is given in this case because full description requires a separate algorithm for its explanation. If $R_c > \sqrt{3}R_s$ and node at farthest distance is activated (as done in the PHA algorithm), then full coverage in the network is not ensured. To handle such cases, a separate timer is required. Condition 1 in the Sect. 4.4.1 defines one timer. That timer is used for the nodes whose distance from transmitter is less than $\sqrt{3}R_s$. For nodes with distance greater than $\sqrt{3}R_s$, a random constant value a is added to the timer so that nodes with distance less than $\sqrt{3}R_s$ gets priority for activation over those nodes having distance greater than $\sqrt{3}R_s$. Timer defined in condition 2.2.1 needs to be recalculated otherwise node activation will not ensure full coverage. For timer calculation, the intersection points of circles centered at X and Y (defined in condition 2.2.1) with radius $\sqrt{3}R_s$ are calculated. Now, nodes which are near to this intersection point gets priority for activation. Also in that nodes, node whose distance to both X and Y is less than $\sqrt{3}R_s$ gets priority so that triangular pattern can be approximated and activated sensor nodes provide full coverage of the target region.

4.5.3 Logical tree formation

In wired communication, tree topology is formed by connecting nodes in such a way that each node has exactly one parent node (for rooted tree). If there is no information about the source and destination at the intermediate node and it receives message on one port, it broadcasts that message to all other ports and in this way communication is made. Here, message broadcasting does not flood the network with a lot of messages, because in tree topology for n nodes only $n - 1$

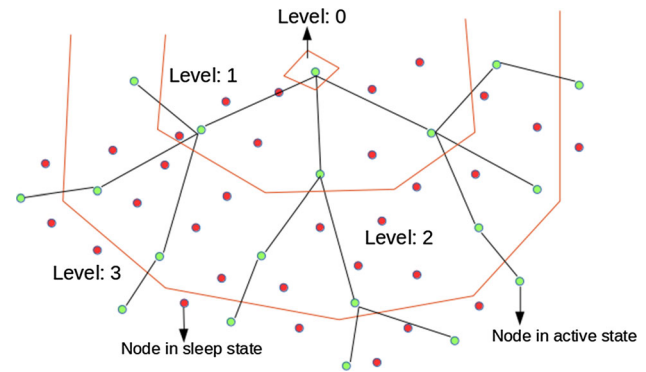


Fig. 10 Diagram representing logical tree structure formed after the execution of PHA algorithm. Green nodes represent active sensor nodes and red nodes represent nodes in sleep state (Color figure online)

messages are created. However, in wireless communication, message transmitted from one node is received by all the nodes in its communication range. If each node starts retransmission of the received message, then message is delivered to the destination, but it requires a lot of communication overhead. Due to excessive message transmission and reception, energy of sensor nodes is consumed much earlier than required. So, there should be some logical tree structure in the wireless networks which can prevent excessive message overhead.

In the case of wireless communication, one way to form a tree like structure is to create a logical tree by creating routing tables in the node. Each entry of the table represents the node position and its entry as parent, sibling or child. When condition 1 or 2.1 (message reception algorithm) is true, then node creates or replaces the entry of parent node. If condition 2.2 is true, then entry of sibling is done and for condition 2.3 entry of child is done. Selection of child node is done by comparing the distance to child node with sibling. If the child is near to the one of the sibling, then its entry is removed from the table. In this way, each sensor node is designated as a child node in only one routing table. Now, data communication proceeds as actual tree is formed by joining the links between sensor nodes.

Suppose, sensor node M has node N as parent and O, P as child nodes. If node M receives some message. Then it checks whether it is the query message transmitted from parent node N. If both of these conditions are satisfied, then only message is forwarded by the node M otherwise it is dropped. Another case of message forwarding occurs when received message is a reply message and is received from child node (O or P). In all other cases received message is dropped. This way of message forwarding proceeds in the way as actual link are established between the sensor nodes to form a tree structure. Figure 10 shows the tree structure formed by sensor nodes with different levels according to their hop-count value from starting node. It is clear from

the figure that each sensor node has exactly one parent and it may have one, two, three or more children. There may be some case in which a sensor node does not have any child node because child selection process is totally dependent on its distance from neighbors.

5 Performance analysis and comparison

This section gives simulation details, performance metrics, performance analysis of PHA algorithm and its comparison with existing schemes.

5.1 Network setup

Simulation of PHA algorithm is performed in OMNET++ [28]. For simulation, target field of $120 \times 120 \text{ m}^2$ is selected and sensor nodes are randomly deployed in the target area. For node deployment, the uniform distribution is used so that it can properly model the actual sensor deployment from an airplane or helicopter. Sensor nodes are placed in a randomly generated value for x and y coordinate whose value lie between 0 and 120. For simplicity, sink node is placed at the origin, so that node activation starts from one direction. However, sink node is not necessary to be placed at the origin, position of sink node can be changed according to the requirement. Sensing range (R_s) of each sensor node is fixed at 10 m and communication range (C_r) varies from 12 to 17.32 m. 17.32 communication range (it is the radius of the circle centered at sensor node position) is selected as the upper limit because it is root three times of sensing radius (10 m). Each value used in the graph is averaged value for 50 different network deployment.

To change the network deployment, seed value of random number generator (RNG) is changed in each network setup. By, changing the seed value, the RNG is initialized to different values and random position assigned to sensor node is changed from its previous value. High density of sensor nodes is assumed in the algorithm so simulation is performed in the area of $120 \times 120 \text{ m}^2$ by varying number of nodes in the range of 500–1,500. According to assumption six of the algorithm, in each deployment, sensor nodes form a connected network. Hence, special care is taken so that network formed just after the deployment is a connected network.

5.2 Performance metrics

5.2.1 Coverage ratio

It measures how much percentage of the target area is actually covered by the sensor nodes. It is calculated as the ratio

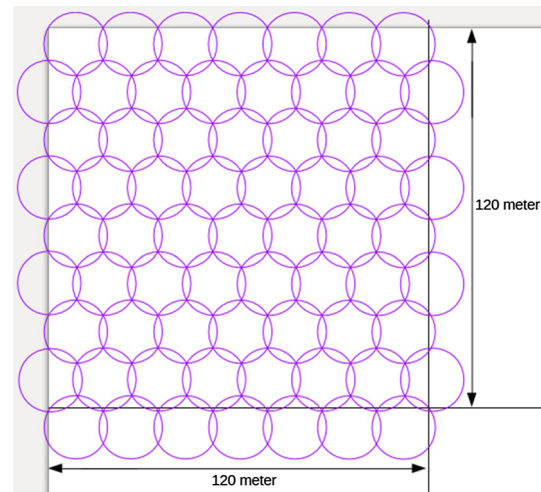


Fig. 11 Diagram representing node placement according to the Theorem 2 and number of nodes required to fully cover the target area

of the sensing area to the actual target area. In measuring the coverage ratio, the target area is divided into several grids where the length and breath of one grid is set to 0.25 m. The end vertex of grid is used in coverage measurement. The matrix A of size equal to the number of grid points is created such that (x, y) coordinate of the end vertex of any grid represents (x, y) entry of the matrix A . This matrix A is initialized to zero. If the end vertex of some grid is inside the sensing range of any sensor node, then corresponding (x, y) entry in the matrix is set to one. Finally, coverage ratio is calculated by calculating the ratio of number of one's in the matrix A to the total number of entries in the matrix A .

5.2.2 Number of active sensor nodes

This metric counts the number of active sensor nodes providing full coverage of the target region.

5.2.3 Number of activation messages used

This metric counts the total number of activation messages sent by the sensor nodes after the execution of full coverage control algorithm. During simulation only those activation messages are counted which are given by the application layer to the network layer (TCP/IP stack). If some collision occurs during the message transmission, then it is left over the MAC layer to deliver the message correctly.

5.3 Performance analysis and comparison

5.3.1 Number of nodes in deterministic deployment

This setup is done to find out the lower bound on the number of nodes required to fully cover the region.

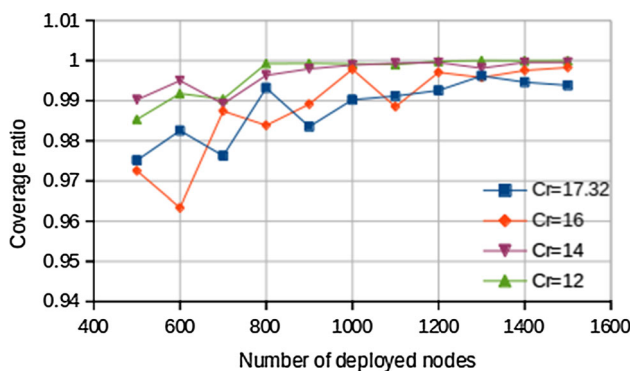


Fig. 12 Coverage ratio versus total deployed nodes

Network parameters are same as in Sect. 5.1 and positioning is done according to theorem 2 (Fig. 11). In this arrangement, number of nodes required to fully cover (coverage ratio = 1) the region comes out to be 67.

5.3.2 Performance analysis

Based on the performance metrics, different graphs are plotted to check the performance of PHA algorithm.

Figure 12 shows the achieved coverage ratio by varying number of deployed nodes. It can be seen that, as the number of sensor nodes increases, coverage ratio also increases. The reason is on increasing the number of sensor nodes, possibility of uncovered area decreases and hence coverage ratio increases. The achieved coverage ratio in most of cases is above 98 %. This confirms that PHA algorithm provides full coverage.

Figure 13 shows the variation in number of active sensor nodes for different number of deployed nodes. For specific value of Cr, curve increases with increase in number of deployed nodes. The reason is that with increase in number of deployed nodes, probability of sensor placement in uncovered region also increases. Figure 13 also shows that for high value of Cr, number of active sensor nodes are less. The reason is that for high value of Cr, large number of sensor nodes receive activation message and change their state to sleep. The another reason is that for high value of Cr, active sensor nodes are at higher distance, so overlapping between their sensing area is less and hence according to theorem 1 number of active sensor nodes are less.

The minimum number of active sensor nodes by PHA algorithm are 90 which are greater than 67. The reason is random deployment due to which sensor nodes cannot be activated according to triangular geometry.

Figure 14 shows the graph for total number of activation message required in the process of providing full coverage. This graph is similar to the graph shown in Fig. 13. The

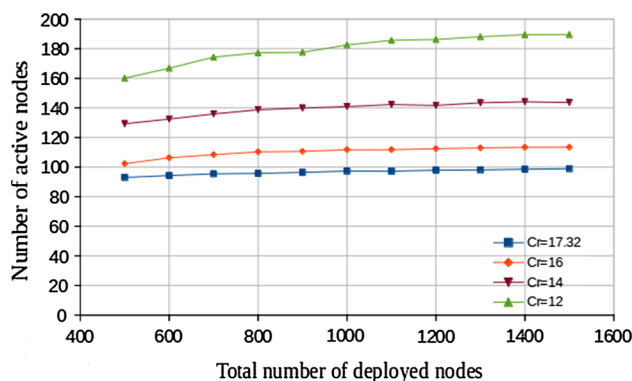


Fig. 13 Working sensor nodes versus deployed sensor nodes

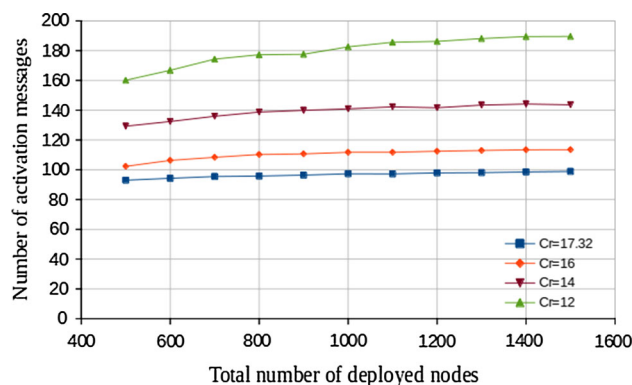


Fig. 14 Number of activation messages used versus deployed sensor nodes

reason is that each active sensor node broadcasts exactly one activation message to its neighbors.

5.3.3 Comparison

Comparison metrics

- a) Coverage ratio.
- b) Number of active sensor nodes.

Figures 15 and 16 compares the PHA algorithm with the LDCC algorithm. Target area is of $120 \times 120 m^2$.

Figure 15 is the graph between the number of active nodes and communication range for different number of deployed nodes (N). This graph shows that PHA algorithm activates less number of nodes than nodes activated with LDCC algorithm. The reason behind is that PHA algorithm tries to approximate the triangular geometry with more accuracy.

Figure 16 shows the graph between coverage ratio and communication range. The coverage ratio of PHA algorithm is better than that of the LDCC algorithm because of

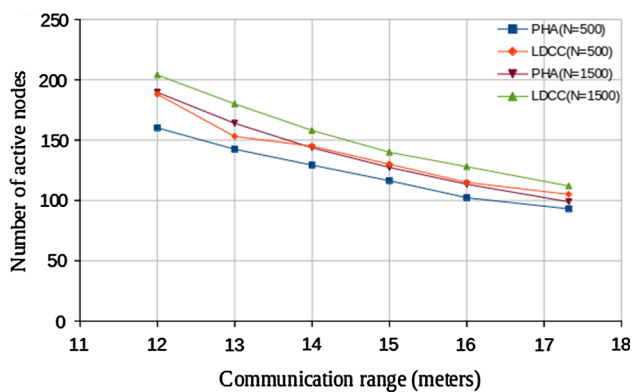


Fig. 15 Active nodes versus communication range

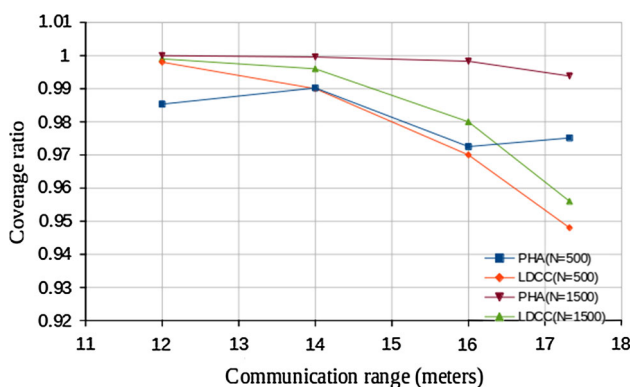


Fig. 16 Coverage ratio versus communication range

better implementation of the full coverage detection scheme at each sensor node.

Figures 17 and 18 compares the PHA algorithm with the OGDC, PEAS and CCP algorithm. Target area is of $50 \times 50 \text{ m}^2$.

Figure 17 shows that PEAS algorithm with probing range of 8 m provides maximum coverage. But high coverage ratio is achieved by activating large number of sensor nodes (Fig. 18). On the other side, PHA and OGDC algorithm provides full coverage by using very less number of sensor nodes than PEAS with probing range of 8 m.

Figure 18 shows number of active sensor nodes for different algorithms. OGDC activates minimum number of sensors by approximating triangular pattern very efficiently. PHA algorithm uses few more sensors than OGDC algorithm because it also provides 3-connectivity in the network. However, performance of PHA algorithm is equivalent to CCP and PEAS with probing range 10.

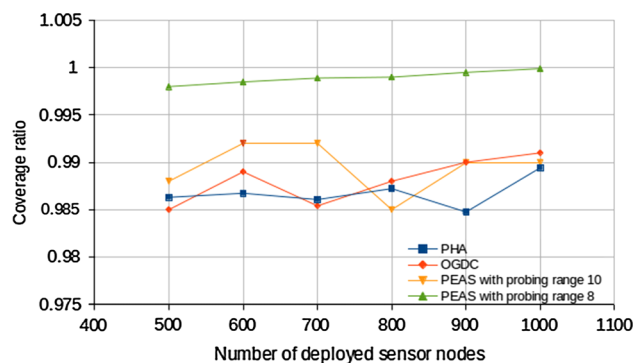


Fig. 17 Coverage ratio versus number of deployed sensor nodes

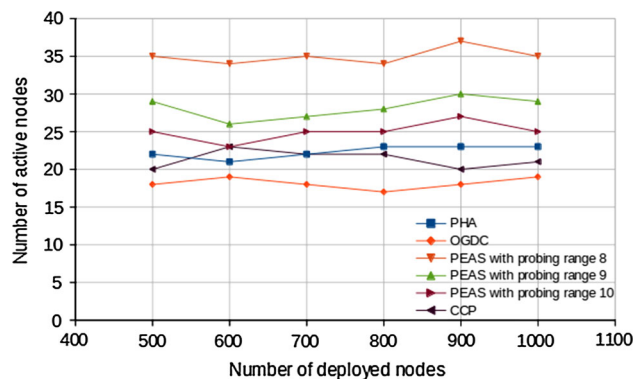


Fig. 18 Number of active nodes versus number of deployed nodes

6 Conclusion and future work

This paper explores a new algorithm for providing full coverage of the target field. Proposed algorithm provides full coverage of the target field, maintains network connectivity and tries to minimize the sensor nodes in the active state. The algorithm is designed in such a way that it is able to provide 3-connectivity in the network. This algorithm also provides logical tree structure under the condition that each sensor node has enough space to store the position information of all its active neighbors. Algorithm is given for the case where communication range is less than root three times of sensing range. However, only timer adjustment enables algorithm to work for the case where communication range is greater than root three times of sensing range. Simulation shows that PHA algorithm is able to provide full coverage of the target field by activating very less number of sensor nodes and uses very less number of activation message. Hence, it saves the energy of sensor nodes and helps to increase the sensor network's lifetime.

One can extend the algorithm to include the heterogeneous nodes in the network or add mobile sensor nodes. The algorithm can be extended to provide coverage and connectivity issue with routing in the network. Load balancing factor can also be included the algorithm to save the energy of node present near the sink node. It is not always desired to provide full coverage of the target area. So, some extension can be given to convert the algorithm from full coverage to target coverage and vice versa. The fault tolerance issue can be included in the algorithm so that full coverage is maintained even if some nodes are destroyed or their battery power is zero.

The proposed algorithm can also be extended to execute in several rounds, selecting different nodes in each round. Criteria for selection may include position, hop-count value and remaining energy of the sensor node. This properly distributes the energy consumption among the sensor nodes. Logical tree structure suggested in the work can be used to route the data effectively and some redundancy removal techniques can also be applied by the sensor nodes to the data obtained from the child nodes. It helps in energy saving because data for transmission is reduced. The algorithm can be extended to provide full coverage in three dimensions.

References

- Bangali, J., & Shaligram, A. (2013). Energy efficient smart home based on wireless sensor network using labview. *American Journal of Engineering Research (AJER)*, 2(12), 409–413.
- Ko, J. G., Lu, C., Srivastava, M. B., Stankovic, J. A., Terzis, A., & Welsh, M. (2010). Wireless sensor networks for healthcare. *Proceedings of the IEEE*, 98(11), 1947–1960.
- He, D., Chen, C., Chan, S., Bu, J., & Vasilakos, A. V. (2012). ReTrust: Attack-resistant and lightweight trust management for medical sensor networks. *Information Technology in Biomedicine, IEEE Transactions on*, 16(4), 623–632.
- Duriscic, M. P., Tafa, Z., Dimic, G., & Milutinovic, V. (2012). A survey of military applications of wireless sensor networks. In *Embedded computing (MECO)*, 2012 Mediterranean Conference on, pp. 196–199, June 2012.
- Othman, M. F., & Shazali, K. (2012). Wireless sensor network applications: A study in environment monitoring system. *Procedia Engineering*, 41(0), 1204–1210. International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012).
- Aslan, Y. E., Korpeoglu, I., & Ulusoy, O. (2012). A framework for use of wireless sensor networks in forest fire detection and monitoring. *Computers, Environment and Urban Systems*, 36(6), 614–625. Special Issue: Advances in Geocomputation.
- Wireless sensor network. Web resource from Wikipedia. http://en.wikipedia.org/wiki/Wireless_sensor_network. Accessed 27 May 2014.
- IRIS sensor specification from MEMSIC. Web resource. http://www.memsic.com/userfiles/files/Datasheets/WSN/IRIS_Datasheet.pdf. Accessed 27 May 2014.
- Wireless sensor nodes. Web resource from Wikipedia. http://en.wikipedia.org/wiki/List_of_wireless_sensor_nodes. Accessed 27 May 2014.
- Zhang, H., & Hou, J. C. (2005). Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc and Sensor Wireless Networks*, 1(6), 89–124.
- Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., & Gill, C. (2003). Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of the 1st international conference on embedded networked sensor systems, SenSys '03*, pp. 28–39. New York, NY: ACM.
- Gallais, A., Carle, J., Simplot-Ryl, D., & Stojmenovic, I. (2006). Localized sensor area coverage with low communication overhead. In *Proceedings of the fourth annual IEEE international conference on pervasive computing and communications, PERCOM '06*, pp. 328–337, Washington, DC, USA, 2006. IEEE Computer Society.
- Wang, B., Fu, C., & Lim, H. B. (2009). Layered diffusion-based coverage control in wireless sensor networks. *Computer Networks*, 53(7), 1114–1124.
- Tian, D., & Georganas, N. D. (2002). A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the 1st ACM international workshop on wireless sensor networks and applications, WSNA '02*, pp. 32–41. New York, NY: ACM.
- Ye, F., Zhong, G., Cheng, J., Lu, S., & Zhang, L. (2003). PEAS: A robust energy conserving protocol for long-lived sensor networks. In *Distributed computing systems, 2003. Proceedings. 23rd international conference on*, pp. 28–37, May 2003.
- Sabbineni, H., Chakrabarty, K., Ji, X., Zha, H., Lee, D., Varaiya, P., et al. (2005). *Sensor deployment, self-organization, and localization* (pp. 11–90). London: Wiley.
- Li, M., Li, Z., & Vasilakos, A. V. (2013). A survey on topology control in wireless sensor networks: Taxonomy, comparative study, and open issues. *Proceedings of the IEEE*, 101(12), 2538–2557.
- Zhu, C., Zheng, C., Shu, L., & Han, G. (2012). A survey on coverage and connectivity issues in wireless sensor networks. *Journal of Network and Computer Applications*, 35(2), 619–632. Simulation and Testbeds.
- Sengupta, S., Das, S., Nasir, M., Vasilakos, A. V., & Pedrycz, W. (2012). An evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in wireless sensor networks. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(6), 1093–1102.
- Nazir, U., Arshad, M. A., Shahid, N., & Raza, S. H. (2012). Classification of localization algorithms for wireless sensor network: A survey. In *Open source systems and technologies (ICOSST)*, 2012 international conference on, pp. 1–5, Dec 2012.
- Han, G., Xu, H., Duong, T. Q., Jiang, J., & Hara, T. (2013). Localization algorithms of wireless sensor networks: A survey. *Telecommunication Systems*, 52(4), 2419–2436.
- Wei, G., Ling, Y., Guo, B., Xiao, B., & Vasilakos, A. V. (2011). Prediction-based data aggregation in wireless sensor networks: Combining grey model and Kalman filter. *Computer Communications*, 34(6), 793–802.
- Xiang, L., Luo, J., & Vasilakos, A. (2011). Compressed data aggregation for energy efficient wireless sensor networks. In *Sensor, mesh and ad hoc communications and networks (SECON)*, 2011 8th annual IEEE communications society conference on, pp. 46–54, June 2011.
- Yao, Y., Cao, Q., & Vasilakos, A. V. (2013). EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for wireless sensor networks. In *Mobile ad-hoc and sensor systems (MASS)*, 2013 IEEE 10th international conference on, pp. 182–190, Oct 2013.

25. Yao, Y., Cao, Q., & Vasilakos, A. V. (2014). EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks. *Networking, IEEE/ACM Transactions on, PP(99)*, 1–1.
26. Sheng, Z., Yang, S., Yu, Y., Vasilakos, A., McCann, J., & Leung, K. (2013). A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities. *Wireless Communications, IEEE, 20(6)*, 91–98.
27. Han, K., Luo, J., Liu, Y., & Vasilakos, A. V. (2013). Algorithm design for data communications in duty-cycled wireless sensor networks: A survey. *Communications Magazine, IEEE, 51(7)*, 107–113.
28. OMNET++ network simulation framework. <http://www.omnetpp.org/>. Accessed 27 May 2014.



Abhiram Singh obtained his B.E. degree in Computer Science and Engineering Department of Oriental Institute of Science and Technology, Bhopal, India in 2012 and his M.Tech degree in Computer Science and Engineering Department of National Institute of Technology, Hamirpur, India in 2014. His research interest includes wireless sensor networks.



T. P. Sharma is an Associate Professor and Head in Computer Science and Engineering Department of National Institute of Technology, Hamirpur, India. He has done his Ph.D from Indian Institute of Technology, Roorkee (Electronics and Computer Engineering Department), India in the area of Wireless Sensor Networks. He has published numerous high quality research papers in International/National journals and conferences, and has also contributed in various books of standard international publishers. His research interest includes distributed systems, wireless sensor networks, mobile ad hoc networks and wireless networks.