

Coverage hole detection and restoration algorithm for wireless sensor networks

Tarachand Amgoth¹ · Prasanta K. Jana¹

Received: 16 October 2014 / Accepted: 13 August 2015 / Published online: 23 August 2015
© Springer Science+Business Media New York 2015

Abstract In wireless sensor networks (WSNs), preserving coverage of the target area by the sensor nodes is crucial to deliver certain services. However, a few sensor nodes die during the network operation due to energy constraints which may break the coverage of the target area. We refer this as a coverage hole problem. In this paper, we propose a new algorithm called CHD-CR which attempts to restore the coverage after detecting a coverage hole in the target area. The proposed algorithm mainly consists of coverage hole detection (CHD) and coverage restoration (CR) phases. We design an efficient local mechanism for coverage hole detection and energy efficient technique for its restoration. We test the performance of the proposed algorithm through simulation and the results are compared with the existing algorithms. Simulation results show that the proposed CHD-CR algorithm outperforms the existing algorithms in terms of coverage preservation for longer time.

Keywords Wireless sensor networks · Coverage preservation · Coverage hole · Coverage restoration · Energy efficient

1 Introduction

Wireless sensor networks (WSNs) are known for potential applications such as in environment monitoring and surveillance

etc. [1]. Sensor nodes are deployed in a target area to sense physical conditions of the area. The sensor nodes collect certain data from the target area, process and send to a special command node called sink. In some applications, such as vehicular tracking or intrusion detection, sensor nodes need to maintain the complete coverage of the target area to fulfill its objective. During the network operation, few sensor nodes die due to energy exhaustion and it may result in breakdown of the network coverage partially. We refer this as coverage hole problem. In this context, we define coverage lifetime or coverage-time is the time until the formation of coverage hole or sensing hole due to death of few sensor nodes. Therefore, there is a need to adopt an efficient coverage technique to prolong coverage lifetime of the network despite death of few sensor nodes.

Many coverage-aware techniques have been proposed [2–15] for WSNs to maximize the coverage lifetime. They are broadly categorized into coverage-aware scheduling and coverage-aware clustering algorithms. In the former [2–6], a subset of sensor nodes is selected in such way that they cover the whole target area completely and other nodes move to sleep state. Quite a few scheduling algorithms have also been reported in the recent years [7–9]. In the latter [10–15], a node is given higher priority to be selected as cluster head (CH) whose sensing area is completely covered by the sensing area of its neighbor nodes. Early death of such CHs does not breach the coverage of the target area since the CHs are overburdened as compared to their member sensor nodes. However, after some rounds coverage hole is inevitable due to death of few more sensor nodes. Coverage hole can be restored by increasing the sensing range of the sensor nodes near the hole subject to maximum sensing range. This can delay the breakdown of the network coverage. Note that, all the above mentioned techniques lack in detecting the coverage hole and therefore they do not have any strategy to restore it.

In this paper, we propose a new algorithm called CHD-CR which attempts to maximize the coverage lifetime by restoring

✉ Tarachand Amgoth
tarachand.ism@gmail.com

Prasanta K. Jana
prasantajana@yahoo.com

¹ Department of Computer Science and Engineering, Indian School of Mines, Dhanbad, India 826004

the coverage after detecting a sensing hole in the network. The proposed algorithm mainly consists of two phases, namely coverage hole detection (CHD) and coverage restoration (CR). In CHD, each sensor node independently detects any hole by updating certain information with its neighbor nodes. To restore the coverage, a sensor node with relatively higher residual energy is given priority to cover the hole closer to it by increasing its sensing range up to a maximum limit. We test the performance of the proposed algorithm and compare the simulation results with the existing coverage-aware algorithms such as CPCP-cc (coverage-preserving clustering protocol with coverage cost) [12] and flow-balanced routing (FBR) [13] in terms of coverage lifetime.

The remaining part of the paper is organized as follows. Existing works are reviewed in Section 2. Network model and terminologies of the proposed algorithm are described in Section 3. We present our proposed algorithm in Section 4. Simulation results are discussed in Section 5 and the paper is concluded in Section 6.

2 Related works

Many algorithms [2–11] have been proposed to solve the coverage problem in WSNs. They are mainly divided into two types; i) coverage-aware clustering algorithms and ii) coverage-aware scheduling algorithms which are discussed as follows.

Several coverage-aware clustering algorithms have been reported in [10–15]. The algorithm called CPCHSA (coverage-preserving cluster head selection algorithm) in [10] is modified from the LEACH (low energy adaptive clustering hierarchy) [16] protocol. In this algorithm, different nodes are assigned different probabilities for being a cluster head. The probability depends on the normalized effective sensing area of a node, in order to maximize the network sensing coverage. The normalized effective sensing area is defined as the ratio of the effective sensing area to the maximum sensing area of the node. In [8], a novel centralized algorithm has been proposed for near optimal assignment of sensors' state. This algorithm computes a near optimal network topology in which each sensor nodes can be activated, put in sleep mode or promoted as CH. Ultimately, the objective of the algorithm is to maximize the network lifetime while ensuring the full coverage of the monitored area and the connectivity of the obtained network topology. In [12], various coverage-aware cost metrics have been proposed for the selection of the CHs, active nodes and routers in order to maintain the full coverage of the target area. In this techniques, both the remaining residual energy of the sensor nodes as well as redundancy in their coverage are considered together to determine the better candidates for CH, active nodes and routers. A new flow-balanced routing (FBR) protocol has been proposed in [13]. In FBR, selection of CHs is determined on the basis of the overlapping degrees of sensor nodes. Some sensor nodes whose

sensing areas are covered by others are put to sleep state in order to save the energy. Then a multi-level architecture which consists of CHs has been constructed to route the data to the sink. In [14], a new coverage-aware clustering protocol has been proposed called CACP. The CACP uses a coverage-aware cost metric for CH selection and applies a layered self-activation scheme to emulate the most tessellation for active nodes selection. Other coverage related algorithms are reported in [17–25]. One common objective of these algorithms is to maximize the coverage lifetime despite death of few sensor nodes. However, once common draw is that they cannot detect coverage hole after its formation in the network and therefore no strategies to overcome it.

Numerous coverage-aware scheduling algorithms have also been proposed [2–9]. The algorithm in [2] is an extension of the LEACH. In this algorithm, each sensor node decides to turn off when it discovers that its sensing area is completely monitored by its neighbor nodes. In the algorithm [3], a subset of sensor nodes are activated to monitor the target area with by adjusting their sensing ranges so that the whole area is completely covered and other nodes are moved to sleep mode. Many sensor devices with adjustable sensing ranges are available in the market [26, 27]. In [4] proposed an algorithm called adjustable range set covers (AR-SC). Its objective is to find a maximum number of set of sensor nodes and the sensing ranges associated with sensor nodes, such that each set covers all the targets in the area to be monitored. However, AR-SC is not energy-aware because adjusting sensing range of the nodes depends on the number of targets monitored by them only. A similar problem has also been tackled in [5, 6] using linear programming and Voronoi diagram-based heuristic algorithm. In all the above mentioned techniques, once major common drawback is that they cannot detect the formation of coverage holes in the network and therefore no strategy to overcome it. Coverage restoration is not new in WSNs. In [28], focus on how to schedule mobile sensor nodes and their locations in order to overcome the coverage problem as well as minimize their energy consumption due to mobility. A similar coverage problem has been addressed with mobile nodes in [29]. Recently, yet another algorithm has been proposed based on mobile sensor nodes [30]. However, these techniques are not suitable for static sensor networks.

The proposed CHD-CR algorithm has additional benefits which are summarized as follows

- 1). Local mechanism for coverage hole detection – each sensor node detects the formation of holes in the network independently and tries to cover the coverage hole upon its detection. Therefore, the proposed algorithm is distributed and scalable.
- 2). Energy efficient technique for coverage restoration – a sensor node with relatively higher residual energy which has detected the hole is given higher priority to increase

its sensing range since the energy consumption of the sensor nodes is also depends on their sensing ranges. This balances energy consumptions of the sensor nodes.

3 Network model and terminologies

In this section, we present the network model of the proposed algorithm followed by definition of some of the terminologies used in the proposed algorithm.

3.1 Network model

We present here some assumptions for the network model used in the proposed algorithm as follows. A set of sensor nodes is deployed in a two-dimensional plane. All the sensor nodes become static once they are deployed. We assume that all the sensor nodes are provisioned with equal amount of energy. We also assume that all sensor nodes are aware about their locations. Initially, sensing range of all the sensor nodes is assumed to be equal, i.e., r_s . We consider that each sensor node has the ability to increase its sensing range up to R^* maximally where $R^* > r_s$. We assume that, with the sensing range R^* , the target area is completely and redundantly covered whereas with the sensing r_s , the target area is partially covered. It means that, initially coverage holes exist in the network. These holes will cover once the CHD-CR starts running. In other words, sensor nodes starts perform sensing tasks with variable sensing radii.

The energy model of the WSN is adopted from [16] in which both the free space and multi-path fading channels are used depending on the distance between the transmitter and receiver node. If the distance is less than a threshold value d_0 , then the free space (*fs*) model is used, otherwise, the multipath (*mp*) model is considered. Let ε_{fs} and ε_{mp} be the energy required by amplifier in free space and multipath respectively, α_{tx} and α_{rx} be the energy dissipated in transmitting and receiving one bit respectively. Then the energy consumed by node i to transmit β -bit data packet to node j is given as follows

$$E_{tx}(i, j) = \begin{cases} (\alpha_{tx} + \varepsilon_{fs} D_{(i,j)}^2) \beta, & D_{(i,j)} < d_0 \\ (\alpha_{tx} + \varepsilon_{mp} D_{(i,j)}^4) \beta, & D_{(i,j)} \geq d_0 \end{cases} \quad (1)$$

where $D_{(i,j)}$ is the Euclidean distance between the node i and j and the energy consumed in receiving the β -bit data by node j is given by

$$E_{rx}(j) = \alpha_{rx} \beta \quad (2)$$

We adopt sensing energy model from [3]. The energy consumed by node i due to sensing is given as

$$E_s(i) \propto r_s^4(i), \quad (3)$$

where, $r_s(i)$ is the sensing range of node i .

3.2 Terminologies

We use the following terminologies in the design of the proposed algorithm as follows.

Definition 1 (Target area). Let A be the target area to be sensed by the sensor nodes. We assume that the entire target area is organized as $n \times n$ virtual square grids called cells. The cell in the x^{th} row and y^{th} column of this area is denoted by $c(x, y)$. For the simplicity and for the sake of designing the algorithm, we assume that the coordinates of the center of the cell $c(x, y)$ is (x, y) and accordingly each cell will be identified by its center coordinate as shown in Fig. 1a. We also assume that each cell c is of size $l_c \times l_c$.

Definition 2 (Complete coverage). Let $s_1, s_2, s_3, \dots, s_n$ be the set of sensor nodes deployed in the target area. The complete coverage of the target area by the sensor nodes will be known by using the formula.

$$\rho = \frac{\left(A \cap \left(\bigcup_{i=1}^N A(s_i) \right) \right)}{A} \quad (4)$$

If $\rho=1$ means that every cell in the target area is covered by at least one sensor node where $A(s_i)$ is the sensing area of the node s_i . An example is shown in Fig. 1b.

Definition 3 (Covered cell). A cell c is covered by the sensor node s_i if it satisfies the following condition

$$c(x, y) \in A(s_i) \text{ if } \sqrt{(x_i - x)^2 + (y_i - y)^2} \leq r_s(i) \quad (5)$$

where $r_s(i)$ is the sensing range of s_i . This is illustrated in Fig. 1c.

Definition 4 (Uncovered cell). A cell c is not covered by the sensor nodes s_i if it satisfies the following condition

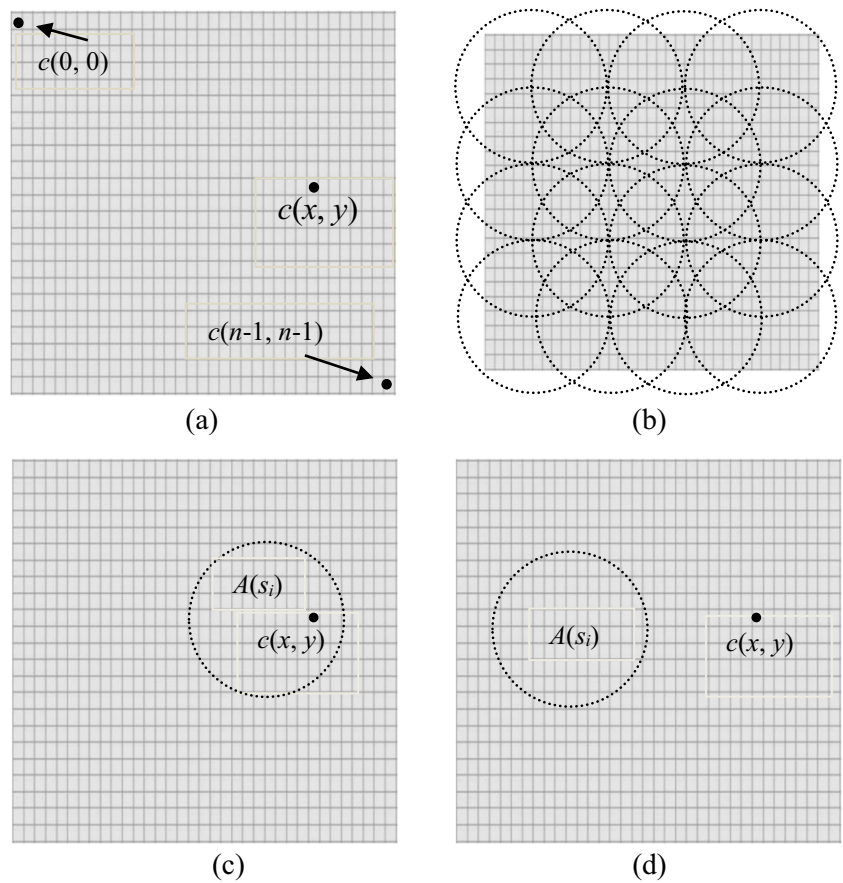
$$c \notin A \cap A(s_i) \quad (6)$$

This is illustrated in Fig. 1d in which c is not covered by sensor node s_i .

4 Proposed algorithm

The proposed CHD-CR is divided into three phases namely update, coverage hole detection and coverage restoration. The proposed algorithm is not a data gathering algorithm. So, the proposed algorithm depends on certain coverage-aware clustering and routing algorithms, and

Fig. 1 **a** Target area is divided into cells; **b** complete coverage of the target area; **c** cell c lies within the sensing range of node s_i ; and **d** cell c lies outside the sensing range of node s_i



tries to take advantage of certain control messages exchanged during clustering. So, we adopt CPCP-cc [12]

as our data gathering algorithm. The proposed CHD-CR replaces the information update of CPCP-cc as shown in

Fig. 2 Information update phase of CPCP-cc is replaced with CHD-CR

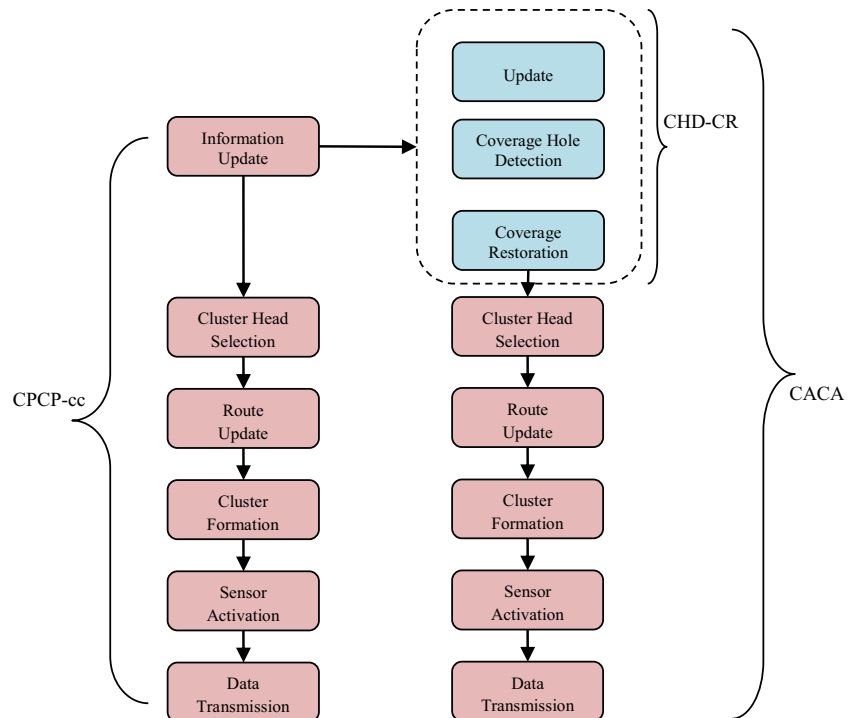


Fig. 2 and we refer it as coverage-aware clustering algorithm (CACA). In other words, last five phases of CACA is same as CPCP-cc. If the performance of CACA is better than CPCP-cc, it means that, this is due to CHD-CR algorithm.

Here, we present the three phases of the proposed CHD-CR algorithm in the following sections subsequently.

4.1 Update

Initially, a sensor node, say s_i , finds the cells that fall within its sensing range $r_s(i)$. Let all such cells be denoted by $Q(i)$. Let node s_i be located at (x_i, y_i) in the target area. Then a diameter of size $2r_s(i)$ is drawn horizontally and vertically centered at location (x_i, y_i) . Divide $r_s(i)$ into H parts such that

$$H = \left\lceil \frac{r_s(i)}{l_c} \right\rceil \tag{7}$$

The cells with the following co-ordinate points are fall within the sensing range $r_s(i)$ of node s_i

$$\begin{aligned} Q_1 &= (x_i + k_1 \times l_c, y_i - k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, H-1 \\ Q_2 &= (x_i - k_1 \times l_c, y_i - k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, H-1 \\ Q_3 &= (x_i - k_1 \times l_c, y_i + k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, H-1 \\ Q_4 &= (x_i + k_1 \times l_c, y_i + k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, H-1 \end{aligned}$$

For each value of k_1 , the value of k_2 ranges from 0, 1, 2, 3, ..., $H-1$. Then,

$$Q(i) = \left(\bigcup_{k=1}^4 Q_k(i) - \varphi_k(i) \right) \tag{8}$$

Here, $\varphi_1 \subset Q_1(i)$, i.e., set of cells lying outside the range $r_s(i)$ as shown in Fig. 3. Note that, we add some refinement steps in the simulation program to eliminate such

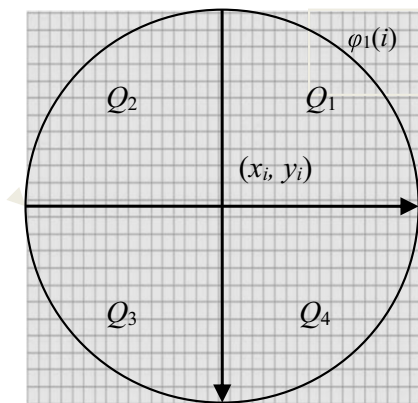


Fig. 3 Illustrating example how to find set Q

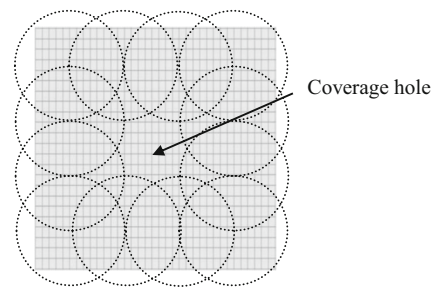


Fig. 4 Formation of coverage hole in the network

cells i.e., φ . In the refinement, each node, say node s_i calculates its distance from the cell belong to $Q_k(i)$, ($1 \leq k \leq 4$) using Eq. 5. This leads to elimination of all the cells belong to φ_k , ($1 \leq k \leq 4$).

Similarly, sensor node s_i finds the set of cells, it can cover in the maximum sensing range R^* , and let all such cells be denoted by $Q^*(i)$ and accordingly

$$H^* = \left\lceil \frac{R^*}{l_c} \right\rceil \tag{9}$$

The cells with following co-ordinate points are fall within the sensing range R^* of node s_i

$$\begin{aligned} Q_1^* &= (x_i + k_1 \times l_c, y_i - k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, (H-1)^* \\ Q_2^* &= (x_i - k_1 \times l_c, y_i - k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, (H-1)^* \\ Q_3^* &= (x_i - k_1 \times l_c, y_i + k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, (H-1)^* \\ Q_4^* &= (x_i + k_1 \times l_c, y_i + k_2 \times l_c), \quad k_1 = 0, 1, 2, 3, \dots, (H-1)^* \end{aligned}$$

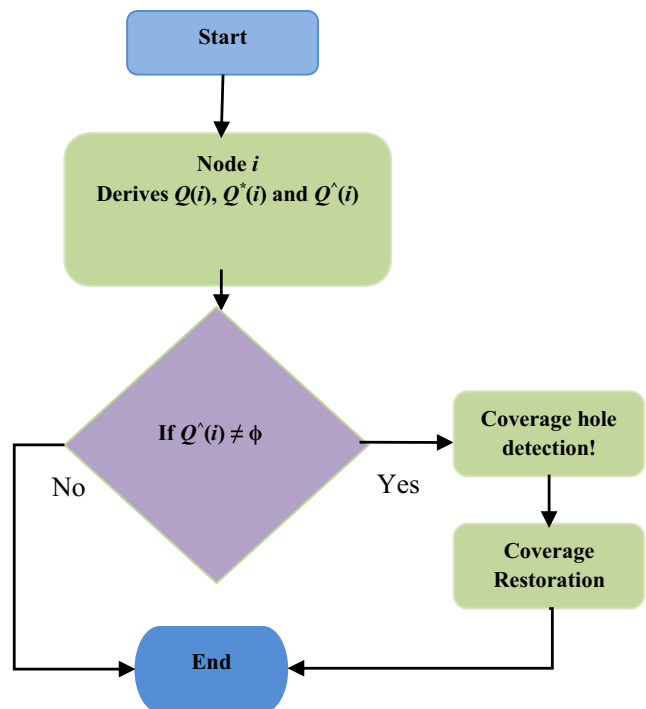
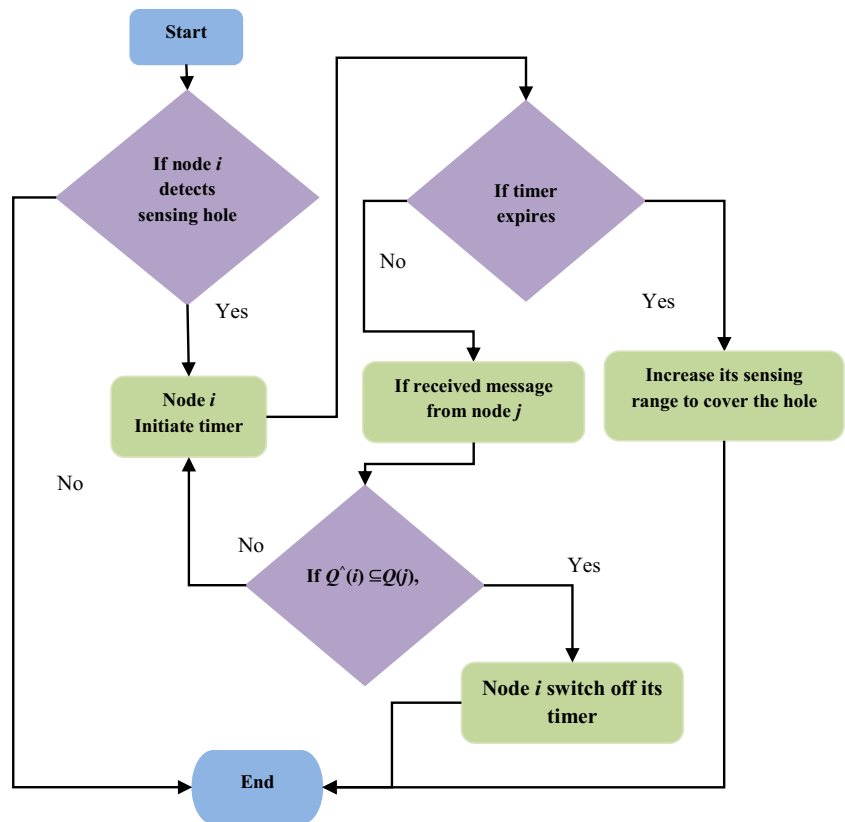


Fig. 5 Flow chart of the coverage hole detection

Fig. 6 Flow chart of coverage restoration



For each value of k_1 , the k_2 value range from 0, 1, 2, 3, ..., $(H-1)^*$. Then,

$$Q^*(i) = \left(\bigcup_{k=1}^4 (Q_k^*(i) - \varphi_k^*(i)) \right) \tag{10}$$

Here, $\varphi_1^* \subset Q_1^*(i)$, i.e., set of cells lying outside its range R^* . Note that, the node s_i derives the set $Q^*(i)$ only once in the lifetime of the network since R^* is fixed. Then the sensor node updates the set $Q'(i)$ as follows

$$Q'(i) = Q^*(i) - Q(i) \tag{11}$$

Here, $Q'(i)$ represents the set of cells lying outside of the sensing range $r_s(i)$ and inside of the sensing range R^* . Here, R^* is same as R_{sense} of CPCP-cc.

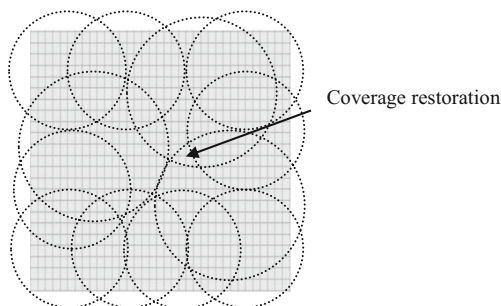


Fig. 7 An illustrating example for coverage restoration

Remark 1 Finding sets Q , Q' and Q^* needs computation rather than communication. In other words, energy consumed due to computation is very less as compared to communication. Hence the proposed update is energy efficient.

Remark 2 In CPCP-cc, it is not shown how to find overlapping areas of the sensor nodes. So, this update phase of CHD-CR can be considered to find overlapping areas in CPCP-cc. Therefore, update phase of CHD-CR is not the additional cost on CACA as compared to CPCP-cc in terms of energy consumption and computation.

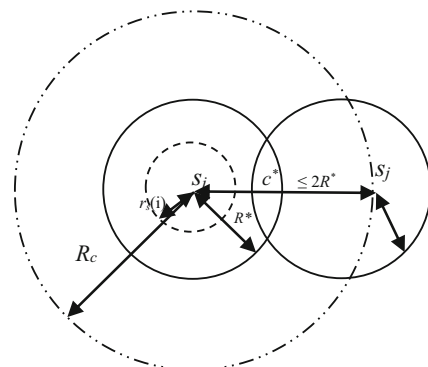


Fig. 8 Illustrating $R_c = 2R^*$

Table 1 Parameters used in simulation

Parameter name	Notation	Value
Communication range	R_c	20 m
Initial energy	E_m	0.5 J
Initial sensing range	r_s	5 m
Max. sensing range	R^*	10 m
T_x or R_x energy	$\alpha_{tx}=\alpha_{rx}$	50 nJ/bit
Amplifier energy(ε_{fs})	ε_{fs}	10 pJ/bit/m ²
Amplifier energy(ε_{mp})	ε_{mp}	0.0013 pJ/bit/m ⁴

4.2 Coverage hole detection

Initially, a sensor node s_i broadcasts a control message to all of its neighbor sensor nodes in the range R_c where $R_c=2R^*$. The message includes its identification (ID) number, location information and $Q(i)$. Let $N^*(i)$ denote the set of all neighbor

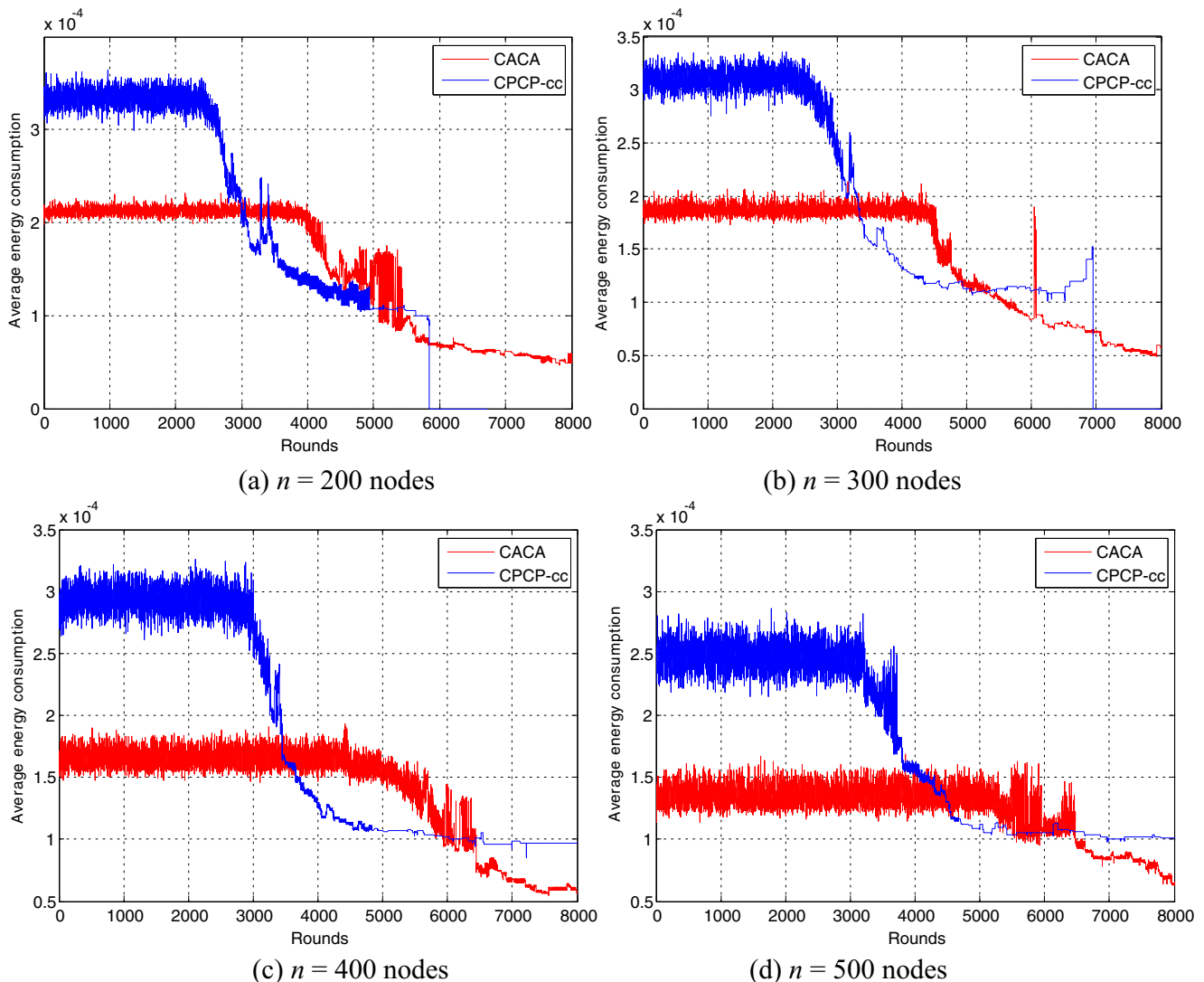
sensor nodes of s_i which fall within the range R_c . If the node s_i receives control message from all of its neighbors belonging to $N^*(i)$, then the node derives a new set $Q^\wedge(i)$ as follows

$$Q^\wedge(i) = Q'(i) - \left(Q'(i) \cap \left(\bigcup_{s_k \in N^*(i)} Q(s_k) \right) \right) \quad (12)$$

If $Q^\wedge(i)=\emptyset$ (empty), it means that the cells belonging to the set $Q'(i)$ are covered by the neighbor nodes of s_i with their sensing ranges. If $Q^\wedge(i) \neq \emptyset$, it means a coverage hole is formed near the sensing area of the node s_i due to death of a few neighbor sensor nodes. An example illustrated in Fig. 4. The flowchart of the coverage hole detection is shown in Fig. 5.

4.3 Coverage restoration

In every round, each sensor node, say s_i updates its set Q^\wedge (refer Eq. 12) to detect the coverage hole near to it. Upon

**Fig. 9** Average energy consumption of sensor nodes

detecting the coverage hole, the sensor node s_i starts estimating its new sensing range as follows. Let $c(x, y)$ be the farthest uncovered cell belonged to $\hat{Q}(i)$. To cover the cell, the node s_i calculates the distance with the cell as follows

$$D_{(s_i,c)} = \sqrt{(x_i-x)^2 + (y_i-y)^2} \quad (13)$$

Before going to increase its sensing range, the node s_i initiates its timers by using the formula

$$t(i) = \left(\frac{E_m(i) - E_r(i)}{E_m(i)} \right) T_{\max} \quad (14)$$

where, T_{\max} is the maximum allotted time to broadcast a control message, $E_m(i)$ and $E_r(i)$ are the initial maximum energy and residual energy of the node s_i , respectively. Once the timer of node s_i expires, then the node updates its new sensing range which is equivalent to $D_{(s_i,c)}$, and then updates $\hat{Q}(i)$. It then broadcasts a control message in the communication range R_c .

The message includes its ID and $\hat{Q}(i)$. A node, say s_j , whose timer is already set and receives the control message from node s_i , then the node s_j makes certain decision based on the following conditions

- 1). If $\hat{Q}(j) \subseteq \hat{Q}(i)$, i.e., the node s_i has covered all the cells that it intends to cover, then the node s_j cancels its timer and maintains its sensing range unchanged. Cancelling timer also indicates that the residual energy of node s_i is higher than node s_j .
- 2). If $\hat{Q}(j) \not\subseteq \hat{Q}(i)$, i.e., coverage hole is not covered completely by the node s_i alone, then the node cancels its timer and updates its set $\hat{Q}(j)$ as follows

$$\hat{Q}(j) = \hat{Q}(j) - (\hat{Q}(j) \cap \hat{Q}(i)) \quad (15)$$

If $\hat{Q}(j) \neq \emptyset$ then the node s_j starts finding the farthest uncovered cell (using Eq. 13) belonged to $\hat{Q}(j)$ and initializes its timer again. Once its timer expires, then the node updates

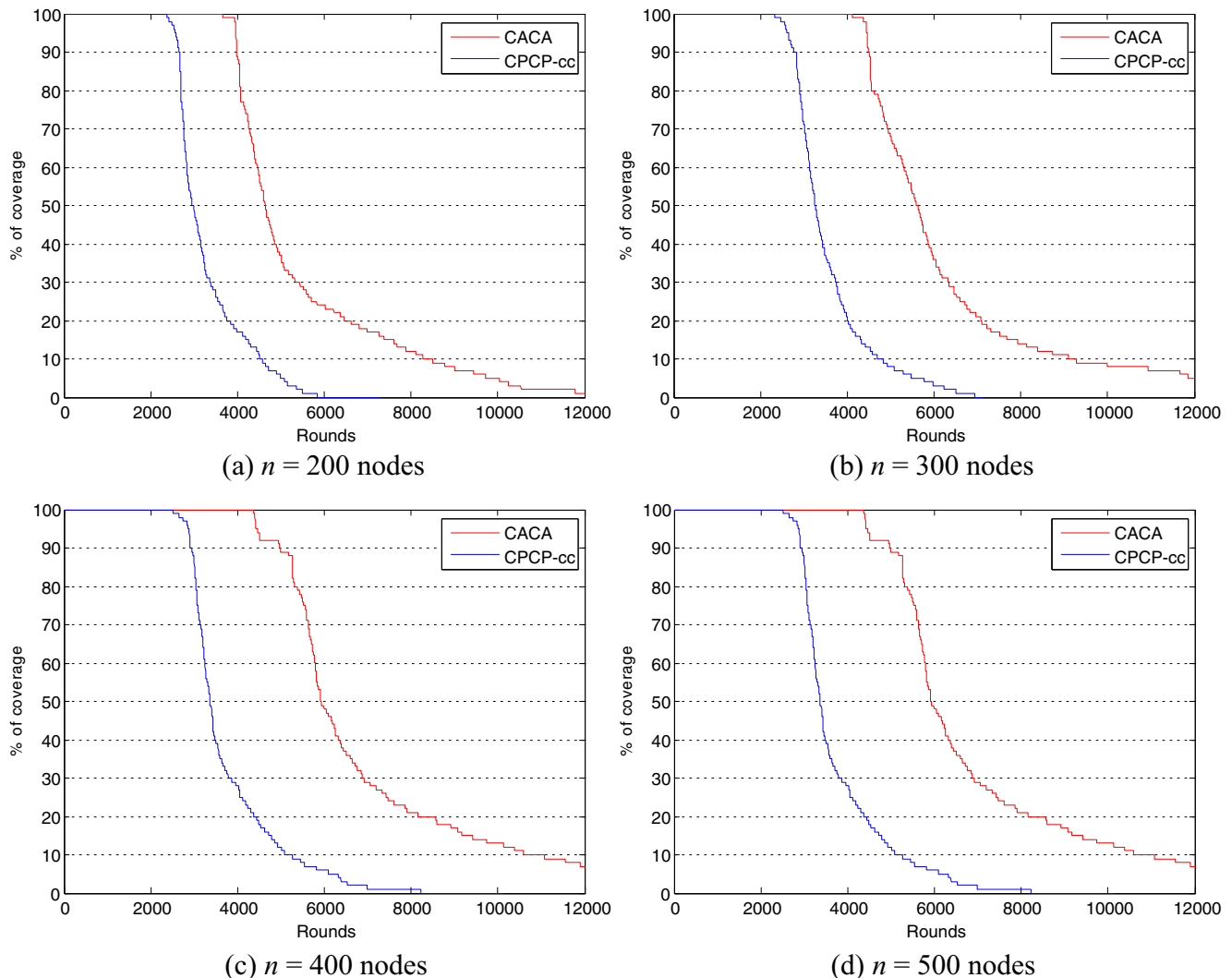
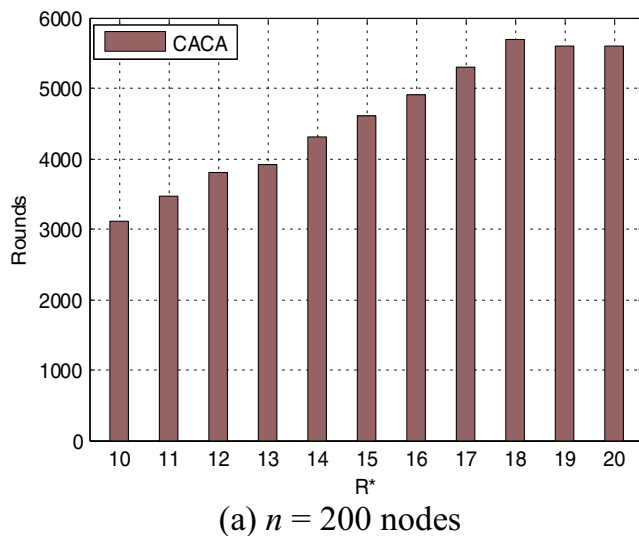


Fig. 10 Coverage lifetime

its new sensing range, $Q(j)$ and $\hat{Q}(j)$. It then broadcasts a control message in the communication range R_c . The flow-chart of the coverage restoration is shown in Fig. 6. An example for coverage restoration is shown in Fig. 7.

Remark 3 Note that a node with relatively higher residual energy is likely to increase its sensing range once it detects a coverage hole. This balances the energy consumption of the sensor nodes since the energy consumption is proportional to the sensing range (refer Eq. 3). Therefore, the proposed algorithm is energy-aware.

Remark 4 The coverage hole detection and restoration process starts after the update phase and before the sensor activation phase of CACA. At this point, no sensor nodes are in sleeping state. Therefore, increasing the sensing ranges of sensor nodes is a reasonable option.



Theorem 4.1 To detect a coverage hole in the network, sensor nodes need to broadcast the control message in the range R_c where $R_c \geq 2R^*$.

Proof Let $r_s(i)$ and R^* be the sensing range and maximum possible sensing range of the node s_i respectively. Node s_i builds $Q(i)$ and $\hat{Q}(i)$ based on the sensing ranges $r_s(i)$ and R^* respectively. Consider a situation where the node s_j has sensing range $r_s(j) \leq R^*$ as shown in Fig. 8.

Let c^* be the farthest cell belonged to $Q^*(i)$, i.e., the distance between the node s_i and c^* is R^* maximally. If c^* belongs to $Q(j)$ of node s_j , then the distance between the node s_i and s_j is not larger than $2R^*$. If s_j is the only node which covers the cell c^* , i.e., $c^* \in Q(j)$ and if s_j cannot broadcast the control message in the range R_c , then the node s_i cannot confirm that the cell is covered by the node s_j . Therefore, each node need to broadcasts the control message in the range R_c where $R_c = 2R^*$.

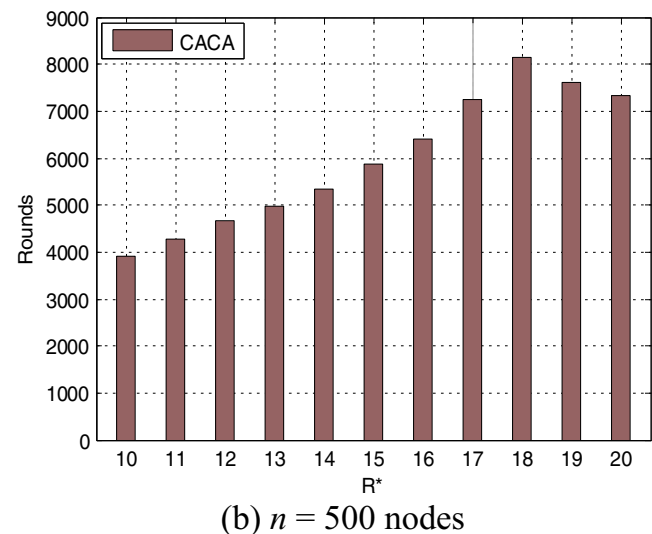
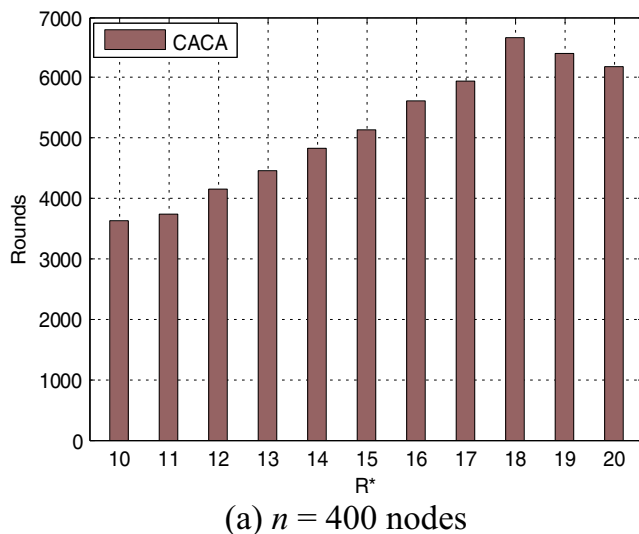
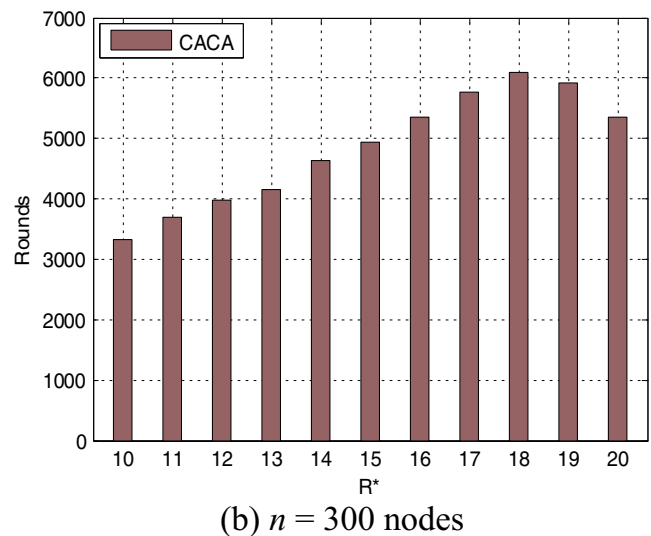


Fig. 11 Impact of R^* on the coverage lifetime

Remark 5 Note that, communicating in R_c is not extra energy consuming as compared to CPCP-cc since R_c is equal to $2R_{sense}$ of CPCP-cc. During the information update phase of CPCP-cc [9], nodes exchange their location information in the communication range which is equal to $2R_{sense}$.

Theorem 4.2 The time complexity of the proposed CHD-CR is $O(n)$ for n sensor nodes in the network.

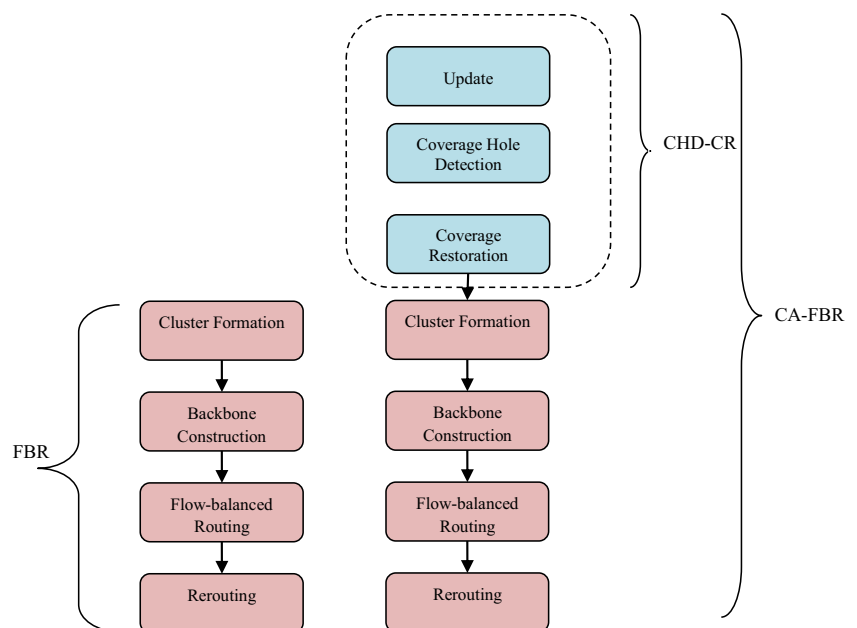
Proof To find set Q^{\wedge} , each sensor node needs to process $n-1$ neighbor nodes in worst case. This takes $O(n)$ time complexity. Once a node detects coverage hole after processing its set Q^{\wedge} , it sets its timer independently; but it broadcasts a control message in the range R_c once its timer expires. This can be done in constant time. Therefore, the time complexity of the proposed algorithm is $O(n)$.

5 Simulation results

In this section, we present the experimental results of the CACA and their comparisons with the two existing algorithms, CPCP-cc [12] and FBR [13]. We incorporate the energy consumed due to sensing in both the existing techniques (refer Eq. 3). We first compare the average energy consumption of CACA and CPCP-cc. Next, we compare CACA and CPCP-cc in terms of coverage lifetime. We also evaluate the impact R^* on CACA. Finally, we show the results with similar experiments conducted using FBR as an underlying data gathering algorithm.

The whole target area of size $100\text{ m} \times 100\text{ m}$ is divided into 1000×1000 cells. Sensor nodes are deployed in the target area randomly. Simulation program was written in Dev C++

Fig. 12 CHD-CR added as an extra phase to FBR



and Matlab. The parameters and their values used in the simulation are given in Table 1. Here, coverage lifetime is defined as the number of rounds until 100 % coverage of the target is maintained by the sensor nodes with their sensing abilities.

5.1 Comparison with CPCP-cc

Here, we first compare the average energy consumption of the sensor nodes of CACA as well as CPCP-cc for various densities of sensor nodes in the same target area. Next, we show comparison of coverage lifetime of the algorithms for the same node densities respectively.

Figure 9a, b, c and d show the average energy consumption of the sensor nodes for 200, 300, 400 and 500 sensor nodes respectively. From these figure we observe that the CACA consume less energy as compared to CPCP-cc. Figure 10a, b, c and d show the coverage lifetime of the sensor nodes for 200, 300, 400 and 500 sensor nodes respectively. From these figures we observe that the CACA performs better than CPCP-cc.

The reasons for the better performance of the CACA over CPCP-cc are justified as follows

1. According to *Remark 2* and *Remark 5*, CHD-CR is not extra cost on CACA as compared CPCP-cc in terms of energy consumption and computation.
2. Energy consumed due to sensing is high in CPCP, i.e., all the nodes perform sensing in the range $R_{sense} (=R^*)$ whereas in CHD-CR, sensing ranges are r_s where $r_s \leq R^*$.
3. Sensor nodes with relative higher residual energy are allowed to increase their sensing ranges to cover the hole. This is not only preserves the coverage but also balances the energy consumption of the sensor nodes.

4. The performance of CACA is better than CPCP-cc. This is due to CHD-CR algorithm since last seven phases of CACA is same as CPCP-cc (refer Fig. 2).

5.2 Impact of R^*

Here, we study the impact of R^* on CACA. In these experiments, we increase the value of R^* from 10 m to 20 m while keeping other parameters remain unchanged. Figure 11 shows the coverage lifetime for 200, 300, 400 and 500 sensor nodes of CACA for different values of R^* .

From these, we observe that the coverage lifetime of the proposed algorithm start increasing for values of R^* ranging from 15 to 18 m and starts decreasing thereupon. This is may be due to maintain of high sensing ranges by the sensor nodes

to cover the holes, increase in data redundancy because of higher sensing overlap among the nodes, and other overheads also associated with them. R_c also increases since $R_c = 2R^*$ and it leads to extra energy consumption and thereby reducing the lifetime of the sensor nodes.

5.3 Comparison with FBR

Here, we adopt FBR [13] as data gathering algorithm. The proposed CHD-CR added to FBR as shown in Fig. 12 and we refer it as coverage-aware FBR (CA-FBR). In other words, last four phases of CA-FBR is same as FBR. If the performance of CA-FBR is better than FBR, it means that, this is due to CHD-CR algorithm.

Figure 13a, b, c and d show the average energy consumption of the sensor nodes for 200, 300, 400 and 500

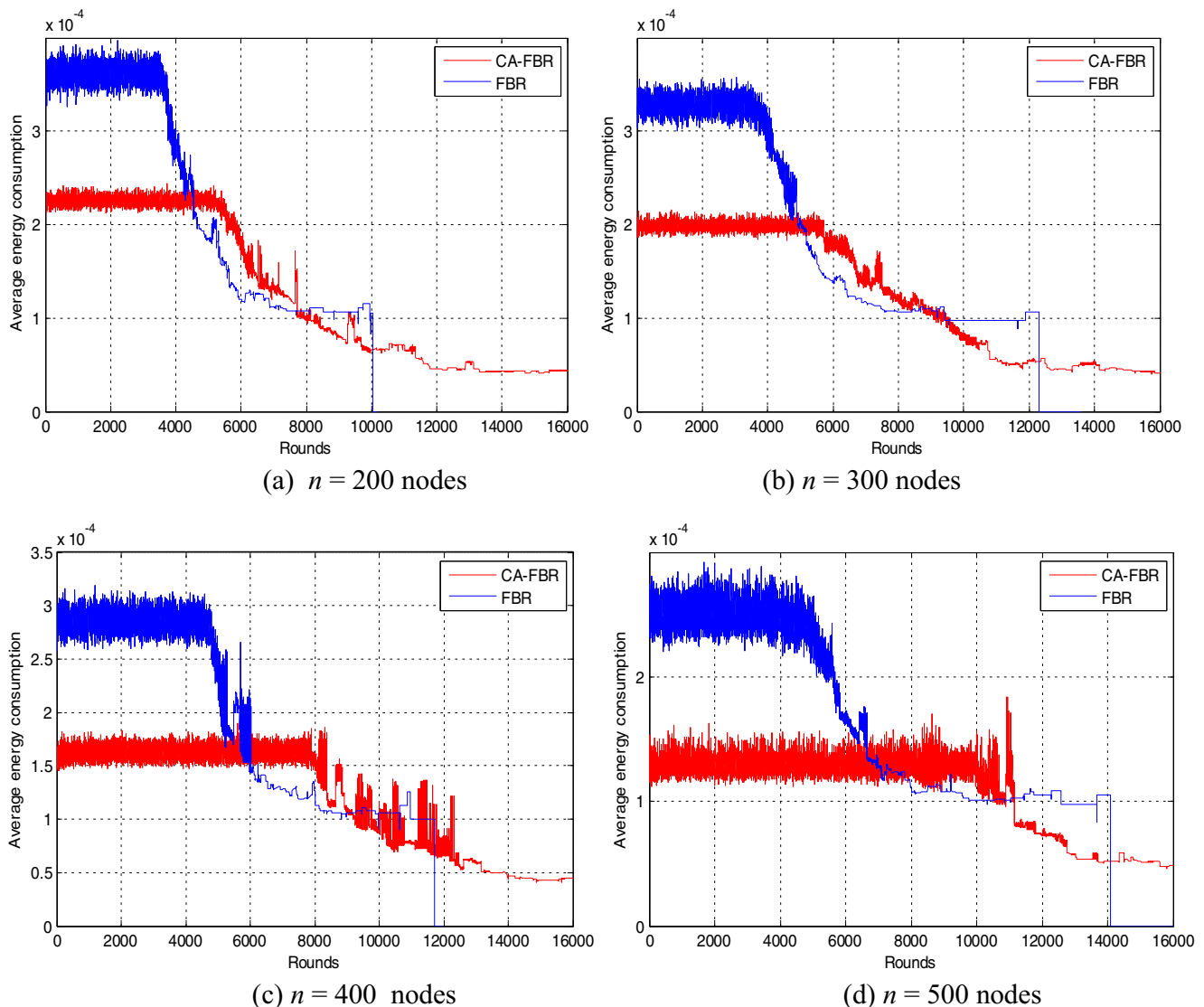


Fig. 13 Average energy consumption of sensor nodes

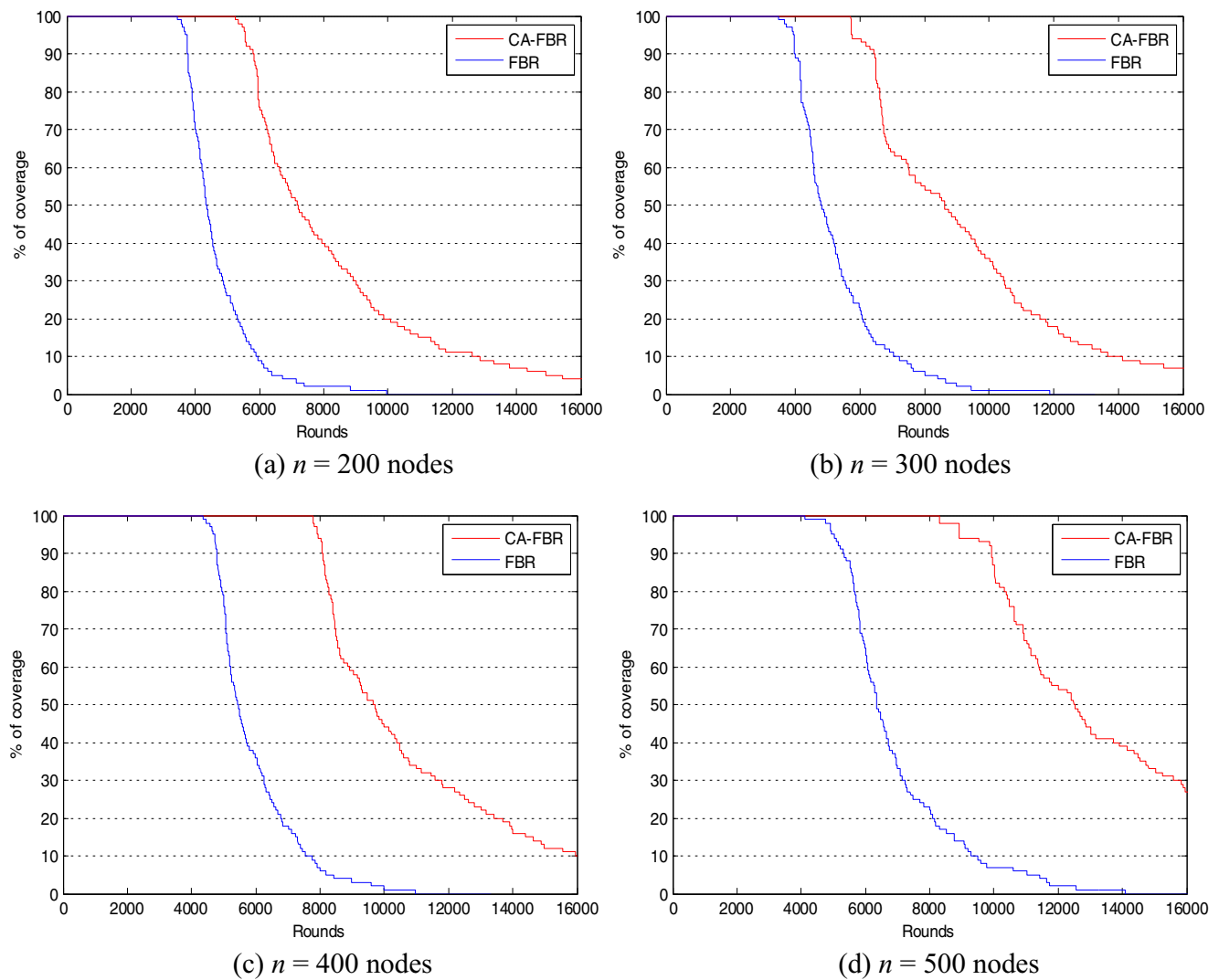


Fig. 14 Coverage lifetime

sensor nodes respectively. From these figure we see that CA-FBR consume less energy as compared to FBR. Figure 14a, b, c and d show the coverage lifetime of the sensor nodes for 200, 300, 400 and 500 sensor nodes respectively. From these figure we find that CA-FBR performs better than FBR.

6 Conclusion

In this paper, we have presented a new distributed algorithm for coverage hole detection and coverage restoration in wireless sensor networks. We have shown that the proposed algorithm detects the coverage hole locally and makes decision independently to restore it. Therefore the proposed algorithm is distributed and scalable. We have also shown that the sensor node which

has relatively higher residual energy than the others was preferred to cover the hole by increasing its sensing range. Therefore, the proposed algorithm is also energy-aware. Through analysis and simulation results, we have shown that, adding CHD-CR to the existing algorithms improves their performance in terms of coverage lifetime.

References

1. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E (2002) Wireless Sensor Networks: survey. *Comput Netw* 38:393–422
2. Tian D, Georganas ND (2002) A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the ACM international workshop on wireless sensor networks and applications*, pp. 32–41

3. Wu, Yang S (2004) Coverage issue in sensor networks with adjustable ranges. In proceedings of the IEEE ICPPW
4. Cardei M, Wu J, Liu M, Pervaiz MO (2005) Maximum network lifetime in wireless sensor networks with adjustable sensing ranges. In Proceedings of WiMob 2005
5. Dhawan A, Vu CT, Zelikovsky A, Li Y, Prasad SK (2006) Maximum lifetime of sensor networks with adjustable sensing range. In Proceedings of the IEEE SNPD
6. Zhou Z, Das S, Gupta H (2004) Variable radii connected sensor cover in sensor networks. In Proceedings of the IEEE SECON
7. Alam KM, Kamruzzaman J, Karmakar G, Murshed M (2014) Dynamic adjustment of sensing range for event coverage in wireless sensor networks. *J Netw Comput Appl* 46:139–153
8. Shi K, Chen H, Lin Y (2015) Probabilistic coverage based sensor scheduling for target tracking sensor networks. *Inf Sci* 292:95–110
9. Tian J, Zhang W, Wang G, Gao X (2014) 2D k-barrier duty-cycle scheduling for intruder detection in Wireless Sensor Networks. *Comput Commun* 43:31–42
10. Tsai YR (2007) Coverage-preserving routing protocols for randomly distributed wireless sensor networks. *IEEE Trans Wirel Commun* 6:1240–1245
11. Chamam A, Pierre S (2009) On the planning of wireless sensor networks: energy-efficient clustering under the joint routing and coverage constraint. *IEEE Trans Mob Comput* 8:1077–1086
12. Heinzelman WB, Soro S (2009) Cluster head election techniques for coverage preservation in wireless sensor networks. *Ad Hoc Netw* 7:955–972
13. Tao Y, Zhang Y, Ji Y (2013) Flow-balanced routing for multi-hop clustered wireless sensor networks. *Ad Hoc Netw* 11:541–554
14. Wang B, Lim HB, Ma D (2012) A coverage-aware clustering protocol for wireless sensor networks. *Comput Netw* 56:1599–1611
15. Gu X, Yu J, Yu D, Wang G, Lv Y (2014) ECDC: an energy and coverage-aware distributed clustering protocol for wireless sensor networks. *Comput Electr Eng* 40:384–398
16. Heinzelman WB, Chandrakasan A, Balakrishnan H (2000) Energy-efficient communication protocols for wireless microsensor networks. In Proceedings of Hawaii international conference on system sciences
17. Jin Y, Jo J-Y, Wang L, Kim Y, Yang X (2008) ECCRA: an energy-efficient coverage and connectivity preserving routing algorithm under border effects in wireless sensor networks. *Comput Commun* 31:2398–2407
18. Torkestani JA (2013) An adaptive energy-efficient coverage algorithm for wireless sensor networks. *Ad Hoc Netw* 11:1655–1666
19. Huang C-F, Tseng Y-C (2003) The coverage problem in a wireless sensor networks. In Proceedings WSNA
20. Jia J, Chang JCG, Tan Z (2009) Energy efficient coverage control on wireless networks based on multi-objective genetic algorithm. *Comput Math Appl* 57:1756–1766
21. Misra S, Kumar MP, Obaidat MS (2011) Connectivity preserving localized coverage algorithm for area monitoring using wireless sensor networks. *Comput Commun* 34:1484–1496
22. Di X (2012) A novel coverage-preserving clustering algorithm for wireless sensor networks. *Phys Procedia* 33:1064–1069
23. Deng X, Yu D, Wang G, Gu X (2013) CWSC: Connected k-coverage working sets construction algorithm in wireless sensor networks. *Int J Electr Commun* 67:937–946
24. Nghiem TP, Kim JH, Lee SHo, Cho TH (2009) A coverage and energy aware cluster-head selection algorithm in wireless sensor networks. In Proceedings of ICIC 2009, LNCS, vol. 5754, pp. 696–705
25. (2009) A distributed coverage-aware sleep scheduling algorithm for wireless sensor networks. In Proceedings of ITNG 2009, pp. 524–527
26. Photoelectric Sensors, <http://www.schneider-electric.ca>
27. Wu J, Yang S (2004) Coverage and Connectivity in Sensor Networks with Adjustable Ranges. International workshop on mobile and wireless networking
28. Nguyen DT, Nguyen NP, Thai MT, Helal A (2011) An optimal algorithm for coverage hole healing in hybrid sensor networks. In Proceedings of IWCMC, pp. 494–499
29. Li F, Xiong S, Wang L (2011) Recovering coverage holes by using mobile sensors in wireless sensor networks, In Proceedings of the ICCIS, pp. 746–749
30. Abo-Zahhad M, Ahmed SM, Sabor N, Sasaki S (2015) Rearrangement of mobile wireless sensor nodes for coverage maximization based on immune node deployment algorithm. *Comput Electr Eng* <http://dx.doi.org/10.1016/j.compeleceng.2015.04.003>



Tarachand Amgoth received B.Tech in Computer Science and Engineering from JNTU, Hyderabad and M.Tech in Computer Science Engineering from NIT, Rourkela in 2002 and 2006 respectively and Ph.D. from Indian School of Mines, Dhanbad in 2015. Presently, he is working as an Assistant professor in the Department of Computer Science and Engineering, Indian School of Mines, Dhanbad. His research interest includes wireless sensor networks and design and analysis of algorithms.



Prasanta K. Jana received M. Tech. degree in Computer Science from University of Calcutta, in 1988 and Ph. D. from Jadavpur University in 2000. Currently he is a Professor in the department of Computer Science and Engineering, Indian School of Mines, Dhanbad, India. He has contributed 118 research publications in his credit and coauthored five books. He has also produced six Ph. Ds. As a recognition of his outstanding research contributions, he has been awarded **Senior Member of IEEE**

(The Institute of Electrical and Electronics Engineers, USA) in 2010. He is in the editorial board of two international journals and acted as referees in many reputed international journals. Dr. Jana has served as the General Chair of the International Conference RAIT-2012, Co-chair of national conference RAIT-2009 and the workshop WPDC-2008. He has also acted as the PC members of several conferences of national and international repute. His current research interest includes wireless sensor networks, parallel and distributed computing, cloud computing and data clustering. He visited University of Aizu, Japan, Las Vegas, USA, Imperial college of London, UK and University of Macau, Macau for academic purpose.