

A Genetic Algorithm based Sleep-Wake up Protocol for Wireless Sensor Networks

Seyed Mahdi Jameii, Arash Nikdel

Abstract— Energy efficiency is an important issue in the field of Wireless Sensor Networks (WSNs). So, minimizing the energy consumption in this kind of networks should be an essential consideration. Sleep/wake scheduling mechanism is an efficient approach to handling this issue. In this paper, we propose a Genetic Algorithm-based Sleep-Wake up Area Coverage protocol called GA-SWAC. The proposed protocol puts the minimum of nodes in active mode and adjusts the sensing radius of each active node to decrease the energy consumption while maintaining the network's coverage. The proposed protocol is simulated. The results demonstrate the efficiency of the proposed protocol in terms of coverage ratio, number of active nodes and energy consumption.

Keywords- Wireless Sensor Networks, Genetic Algorithm, Coverage, Connectivity.

I. Introduction

The advanced technology in the fields of sensor systems, networking and communications lead to development of modern Wireless Sensor Networks (WSNs). WSNs are consisted of large numbers of nodes with radio communications, sensing, and low-power processing capabilities. The lifetime of a wireless sensor network is an important issue due to limited battery power in sensors and infeasibility of replacing and recharging sensor batteries [1-2]. So, it is important to dynamically put sensor nodes into sleep/wakeup statuses [3-4]. In this paper, we design a scheduling mechanism in which only a small number of sensors are active, while all other sensors are in sleep mode. Using the property that sensors have adjustable sensing ranges, the goal is to set up minimum sensing ranges for the active sensors, while satisfying the coverage requirements. Thus, we propose a Genetic Algorithm based Sleep Wake up Area Coverage protocol called GA-SWAC to deal with the problem of Area Coverage. In this protocol, proper active nodes set and their sensing radii can be determined using Genetic Algorithm. The remaining of the paper is organized as follows: In section 2 we present the related work in the context of area coverage. The proposed solution is presented in section 3. In section 4, the simulation results are provided and section 5 concludes the paper.

II. Related Works

In [5], a distributed sleep-wake up scheduling mechanism is presented with the aim of prolonging the network lifetime while guaranteeing network coverage. The proposed mechanism mitigates scheduling process to be more stable by avoiding useless transitions between states without affecting the coverage level required by the application. In [6], the authors evaluated the coverage-aware sleep scheduling algorithm in areas with uneven event distributions and then, its performance on event detection rate and event detection delay have been studied. Authors of [7] proposed a multi-objective optimization algorithm for scheduling the nodes in WSNs to achieve maximum lifetime. They focused on the differentiated or probabilistic coverage where different regions require different levels of sensing. The authors of [8] developed the sleep-wake up algorithms for WSNs aiming to maximize the lifetime of network, while ensuring the monitoring quality for specific applications. In [9] a wakeup scheduling method and power management mechanism is proposed to reduce the energy consumed for WSNs and each sensor node selects its own wake up time intervals based on its pending packets. The authors in [10] focused on topology control and power management and proposed a node scheduling solution that solves the coverage and connectivity problem in sensor networks in an integrated manner. In [11] an energy efficient scheduling algorithm called (ALARM) has been proposed for WSNs. ALARM uses a probabilistic model that permit to cover fault tolerance and a dynamic and distributed sleep timer computation. In [12], by turning on some redundant nodes in the WSNs, a centralized solution is proposed to reduce the energy consumption of the network. A straightforward solution is to use a transition range R_T that is at least twice the sensing range R_S (i.e. $R_T \geq 2R_S$), such that area coverage guaranties network connectivity of active nodes [13]. The authors in [14] addressed the problems of k-connected coverage set in WSNs with the objective of minimizing the total energy consumption while achieving k-coverage. The research presented in [15] used the binary particle swarm optimization (BPSO) approach to solve the disjoint set covers (DSC) problem in WSNs. In [16], the authors proposed a scheduling protocol to identify the redundant sensors and put them to sleep. The proposed protocol is fully distributed and does not use any geographical information. In [17] an Energy Preserving Sleep Scheduling strategy called EPSS has been proposed. In the proposed strategy, the sleeping probability of a node varies according to the density of the deployed region and the energy consumption within a cluster is balanced.

Seyed Mahdi Jameii, Department of Computer Engineering, College of Engineering, Islamic Azad University, Shahr-e-Qods Branch, Tehran, Iran, Jamei@Qodsiau.ac.ir.

Arash Nikdel, Sama technical and vocational training college, Islamic Azad University, Shoushtar Branch, Shoushtar, Iran, A.nikdel@khuzestan.srbiau.ac.ir.

III. The Proposed Protocol

In the proposed protocol, we assume that each node has adjustable sensing radius R_S ($R_{min} \leq R_S \leq R_{max}$) and the solution is represented by a bit-string $a = (a_1, a_2, \dots, a_B)$, as shown in Fig.1.

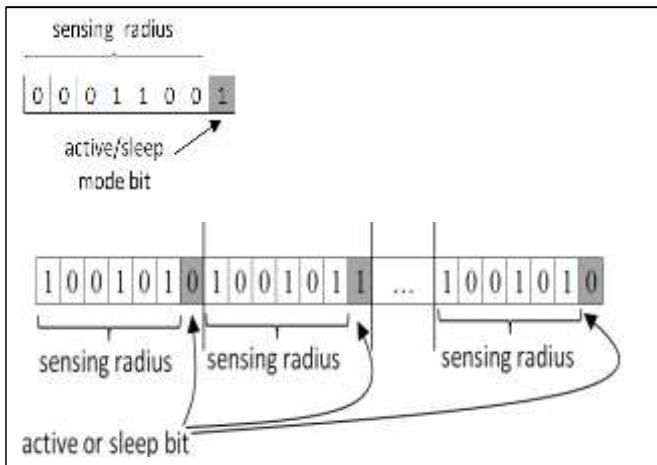


Fig. 1 Example of a solution in the proposed algorithm

The sensing radius coding is binary and required B bit. B is calculated based on (1):

$$B = (\lceil \log_2 (R_{max} - R_{min} + 1) \rceil \cdot (1/\mu)) + 1 \quad (1)$$

where, $\lceil x \rceil$ is the ceiling function that makes a real value x to be rounded up to the closest large integer value.

The sensing radius of each node is calculated based on (2):

$$R_S = R_{min} + (R_{max} - R_{min} + 1) (\sum_{b=2}^B 2^{b-2} a_{b-1} / \sum_{b=2}^B 2^{b-2}). \mu \quad (2)$$

Where μ is the interval variations rate of nodes sensing radius.

Moreover, the least bit is used to show active/sleep node. "1" means active mode and "0" means sleep mode.

Regarding that it is impossible to evaluate the coverage of all network points, we use the cellular network for doing area coverage. In this method, the coverage is done according to nodes location and sensing radius such a way that the network area with dimensions $x \times y$ is divided in square cells with dimensions $c \times c$. the sensing area of each node is showed by R_S . The cell is covered if it is completely within the sensing area of a sensor node. We consider a calculated sensing radius (R_{CS}) for each sensor node defined on the basis of real sensing radius and cell size as (3):

$$\begin{cases} R_{CS} = R_S - (2C)^{1/2} \\ R_{C\ min} = R_{min} - (2C)^{1/2} \\ R_{C\ max} = R_{max} - (2C)^{1/2} \end{cases} \quad (3)$$

Where $(2C)^{1/2}$ is the length of cell diameter. The reason for such definition is that if one of the cell vertices has the

some overlap with a node calculated sensing radius, it will be covered completely based on the real sensing radius of node. As a result, one cell is covered if it is within the calculated sensing radius of an active sensor node as shown in Fig. 2. In this figure, the dark cells are covered by node N based on R_{CS} . The number of covered cells is divided by all area cells to obtain the network area coverage.

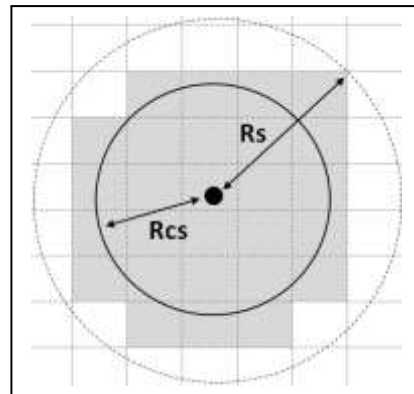


Fig. 2 The Covered cells by active node N based on R_{CS}

When one cell has some overlap with $R_{C\ max}$ node $_i$, we assume it is one of cells in sensing set of node $_i$. Because we sure that this node covered by R_{max} .

Cells of sensing set are in four different groups. Sets of A_{min} , A_S and A_{max} are calculated according to the following equations (c_i is cell number):

$$\begin{cases} c_i \in \text{Sensing Set} & \text{if } c_i \text{ covered by } R_{max} \\ c_i \in A_{min} & \text{if } c_i \text{ covered by } R_{min} \\ c_i \in A_S & \text{if } c_i \text{ covered by } R_S \text{ \& not covered by } R_{min} \\ c_i \in A_{max} & \text{if } c_i \text{ covered by } R_{max} \text{ \& not covered by } R_S \end{cases} \quad (4)$$

So we have:

$$\begin{cases} \text{Sensing Set} = \text{all cells have overlap with } R_{C\ max} \\ A_{min} = \text{all cells have overlap with } R_{C\ min} \\ A_S = \text{all cells have overlap with } R_{CS} - A_{min} \\ A_{max} = \text{all cells have overlap with } R_{C\ max} - (A_S \cup A_{min}) \end{cases} \quad (5)$$

Therefore:

$$A_{max} \cup A_S \cup A_{min} = \text{Sensing Set} \quad (6)$$

$$A_{max} \cap A_S \cap A_{min} = \{ \} \quad (7)$$

A_C consists of cells that are sensed by selective sensing area or are covered by selective sensing area of other nodes. A_{min} is proper subset of A_C , because each node has minimum sensing area A_{min} . In this paper, we will deal with the nodes deployed randomly. We assume that the nodes are static once deployed, and each one knows its own location which can be achieved by using some location system [18]. The coverage model of the node n_i is supposed as a circle centered at its coordinates (x_i, y_i) with radius r_i .

$$\begin{aligned} &\text{If } (x-x_i)^2 + (y-y_i)^2 \leq r_i^2 \\ &\quad P(c_i) = P_{\text{cov}}(x,y,n_i) = 1 \\ &\text{Else} \\ &\quad P(c_i) = P_{\text{cov}}(x,y,n_i) = 0 \end{aligned} \quad (8)$$

We assume that any random event c_i is independent of the others, so c_i and c_j are unrelated, $i, j \in [1, N]$ and $i \neq j$. Then the following two relationships can be concluded:

$$P(c'_i) = 1 - P(c_i) = 1 - P_{\text{cov}}(x,y,n_i) \quad (9)$$

$$P(c_i \cup c_j) = 1 - P(c'_i \cap c'_j) = 1 - P(c'_i) \cdot P(c'_j) \quad (10)$$

Where c'_i is the complement of c_i , denoting that sensor n_i fails to cover (x,y) .

It can be considered that the pixel (x,y) is covered by the node set if any node in the set covers it. As a result, the probability of the event that the pixel (x,y) is covered by the node set can be denoted as the union of c_i :

$$P_{\text{cov}}(x,y,N) = P(\cup_{i=1 \text{ to } N} c_i) = 1 - P(\cap_{i=1 \text{ to } N} c'_i) = 1 - \prod_{i=1 \text{ to } N} (1 - P_{\text{cov}}(x,y,n_i)) \quad (11)$$

Also, we define the coverage rate of the sensor set $R_{\text{area}}(C)$ as the proportion of the monitoring area $A_{\text{area}}(C)$ to the total area A_S :

$$R_{\text{cov}}(R) = A_{\text{area}}(C) / A_S = \sum_{x=1 \text{ to } m} \sum_{y=1 \text{ to } n} P_{\text{cov}}(x,y,C) / (m \times n) \quad (12)$$

The cluster head node tries to select the minimum number of nodes using an algorithm based on Genetic Algorithm such a way that the full coverage of the target area is provided. Also, the proposed protocol tries to select the smallest possible sensing radius for each node. The proposed algorithm can be described briefly in some steps as follows:

Pahse1. Problem and algorithm parameter initialization:

Step1: Initializing A_{min} , A_S and A_{max} sets for each node.

Step2: producing a sensing radius mask and a sensing radius mask operator for each node.

Step3: Initializing sensing radius node, R_S , for each node randomly.

Pahse2. Repeating main loop of algorithm until meeting termination criteria:

Step4: Calculating the fitness rate of nodes.

Step5: Selecting the nodes with more fitness rate as parent nodes by using the roulette wheel and generating new population.

Step6: Mutating the sensing radius of nodes with predetermined mutation rate.

Step7: Checking the loop termination criteria and jumping to step 4.

In this section, we describe the proposed algorithm in detail:

Step1: Initializing A_{min} , A_S and A_{max} sets for each node.

At first, according to (13), the sensing radius of each node is set between R_{min} and R_{max} .

$$\begin{aligned} R_S &= (R_{S1}, R_{S2}, R_{S3}, \dots, R_{Sn}) \\ \forall R_{Si} \in R_S: R_{Si} &= R_{\text{min}} + (R_{\text{min}} - R_{\text{max}}) / \delta \end{aligned} \quad (13)$$

Such a way that δ is a constant factor (e.g. $\delta=2$) which its rate can be determined regarding nodes density. According to (14), if δ is considered much more than R_{max} , R_{Si} will almost equal R_{min} .

$$\text{If } \delta = \infty \Rightarrow R_{Si} \cong R_{\text{min}} \quad (14)$$

Then according to (3), (4) and (5) as mentioned before, A_{min} , A_S and A_{max} sets for each node is created.

Step2: producing a Sensing Radius Mask and a Sensing Radius Mask Operator for each node.

Then A_{min} , A_S and A_{max} sets are updated for each node N . Whenever one cell of A_S and A_{max} sets becomes a member of the A_{min} set of another node, that will be removed from these sets. For this purpose, at first, we initialize A_C set by A_{min} set content and then the sets are updated according to (15):

$$A_C(N) \leftarrow A_{\text{min}}(N)$$

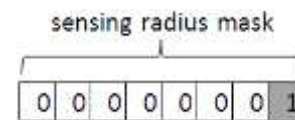
Then:
(15)

$$\begin{aligned} \forall c_i \in A_S(N) \text{ Or } A_{\text{max}}(N) \text{ that} \\ \exists n_j : c_i \in A_{\text{min}}(n_j) \Rightarrow \\ \begin{cases} A_{\text{min}}(N) = A_{\text{min}}(N) + c_i \\ A_S(N) = A_S(N) - c_i \text{ Or } A_{\text{max}}(N) = A_{\text{max}}(N) - c_i \end{cases} \\ \forall c_i \in A_S(N) \text{ Or } A_{\text{max}}(N) \text{ that} \\ \exists n_j : c_i \in A_S(n_j) \Rightarrow \\ \begin{cases} A_{\text{min}}(N) = A_{\text{min}}(N) + c_i \\ A_S(N) = A_S(N) - c_i \text{ Or } A_{\text{max}}(N) = A_{\text{max}}(N) - c_i \end{cases} \end{aligned}$$

Regarding to A_S and A_{max} condition, the node performs a sensing radius mask ($\text{Mask}_{\text{sensing}}$) and determines a sensing radius mask operator ($\text{Operator}_{\text{mask_sensing}}$) with OR/AND. The method of determining sensing radius mask and sensing radius mask operator is calculated according to the four conditions:

- Both A_S and A_{max} sets are empty

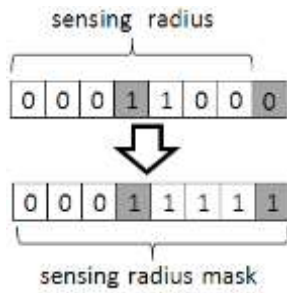
The sensing radius of node is equal to A_{min} . So, the Sensing Radius Mask is as below (Mask Operator is AND):



$$\begin{aligned} \text{If } A_S = \emptyset \text{ and } A_{\text{max}} = \emptyset \Rightarrow \\ \begin{cases} \text{Mask}_{\text{sensing}} = 1 \\ \text{Operator}_{\text{mask_sensing}} = \text{'AND'} \end{cases} \end{aligned} \quad (16)$$

- A_S set is not empty and A_{max} set is empty

The node can select its sensing radius between both R_{min} and R_S . The Sensing Radius Mask of this sensing radius is as below (Mask Operator is AND):

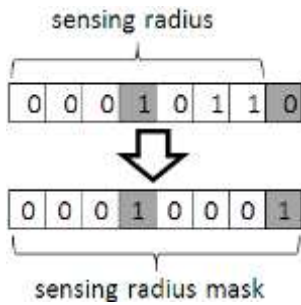


If $A_S \neq \emptyset$ and $A_{max} = \emptyset \Rightarrow$

$$\begin{cases} X = \lceil \log_2 R_S \rceil + 1 \\ \text{Mask}_{\text{sensing}} = 2^{X+1} - 1 \\ \text{Operator}_{\text{mask_sensing}} = \text{'AND'}$$

- A_S set is empty and A_{max} set is not empty

The node can select its sensing radius between R_S and R_{max} . The sensing radius mask of this sensing radius is as below (Mask Operator is OR):

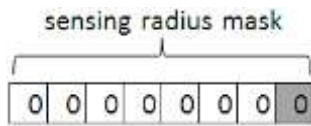


If $A_S = \emptyset$ and $A_{max} \neq \emptyset \Rightarrow$

$$\begin{cases} X = \lceil \log_2 R_S \rceil \\ \text{Mask}_{\text{sensing}} = 2^{X+1} + 1 \\ \text{Operation}_{\text{mask_sensing}} = \text{'OR'}$$

- Both A_S and A_{max} sets are not empty

The node can select its sensing radius between A_{max} and A_{min} . So, the Sensing Radius Mask is as below (Mask Operator is OR):



If $A_S \neq \emptyset$ and $A_{max} \neq \emptyset \Rightarrow$

$$\begin{cases} \text{Mask}_{\text{sensing}} = 0 \\ \text{Operator}_{\text{mask_sensing}} = \text{'OR'}$$

Step3: Initializing sensing radius node, R_S , for each node randomly.

In this step, according to (20), the sensing radius node for each node is initialized randomly. Also, the active/sleep bit of each node is initialized randomly.

$$R_S = (R_{S_1}, R_{S_2}, R_{S_3}, \dots, R_{S_n}) \quad (20)$$

$$\forall R_{S_i} \in R_S : R_{S_i} \leftarrow R_{min} + rand() * (R_{max} - R_{min} + 1)$$

Where $rand()$ is a random number between 0 and 1. Also R_{min} and R_{max} are the lower and upper bounds for sensing radius of each node.

Then, according to (21), for each node, the Sensing Radius Mask is applied to sensing radius of node by the Sensing Radius Mask Operator (AND/OR).

$$R_S = (R_{S_1}, R_{S_2}, R_{S_3}, \dots, R_{S_n}) \quad (21)$$

$$\forall R_{S_i} \in R_S : R_{S_i} \leftarrow R_{S_i} \text{Operator}_{\text{mask_sensing}}(\text{AND/OR}) \text{Mask}_{\text{sensing}}$$

Step4: Calculating the fitness rate of nodes.

The process of calculating the fitness rate for each node N is as follows:

At first, whenever one cell of A_C becomes a member of A_S or A_{max} , that cell is removed from A_S or A_{max} and adds to the A_C set. See more details in (22):

$$\forall c_i \in A_C(N)$$

$$\exists n_j : (c_i \in A_S(n_j) \text{ Or } c_i \in A_{max}(n_j)) \Rightarrow \quad (22)$$

$$\begin{cases} A_C(N) = A_C(N) + c_i \\ A_S(N) = A_S(N) - c_i \text{ Or } A_{max}(N) = A_{max}(N) - c_i \end{cases}$$

Where $A_x(y)$ shown A_x set of y node.

After updating the sets, for each node, we determine the fitness rate of its selected sensing radius (regarding to its neighbor's selected sensing radius). For this purpose, for each node, we consider a temporary TA_C set. As can be seen in (23), this set, at first, is initialized to A_C .

$$TA_C(N) \leftarrow A_C(N) \quad (23)$$

Then, according to (24), we add the A_S set of its neighbors to the same neighbor A_C set:

$$\forall n_i : A_C(n_i) \leftarrow A_C(n_i) + A_S(n_i) \quad (24)$$

After that, regarding the (25), we update TA_C set of node:

$$\forall n_i \exists c_j \in TA_C(n_i) \text{ And } (c_j \in A_S(N) \text{ Or } n_j \in A_{max}(N)) \Rightarrow \quad (25)$$

$$TA_C(N) = TA_C(N) + c_j$$

After updating TA_C set, the process of determining sensing radius fitness of node is as below:

- If $A_S \subset TA_C$ and $A_{max} \subset TA_C$

At this situation, more closely the sensing radius rate to R_{min} , more fit the sensing radius. So:

$$\text{fitness} = \lambda_1 + \psi_1 * (\text{my-cell-A}/\text{max-cells-A})(1/(R_S - R_{min} + \epsilon)) \quad (26)$$

Where ϵ shows very small positive number, λ_1 shows the minimum acceptable rate for fitness of node and ψ_1 is selected as the fitness rate doesn't exceed a given limit.

- If $A_S \not\subset TA_C$ or $A_{max} \not\subset TA_C$

The node adds A_S set to TA_C set and updates TA_C set again by (25). Then, If $A_{max} \subset TA_C$ (evidently

$A_S \subset TA_C$), so the sensing radius will be fit and can be smaller. The details can be seen in (27):

$$\text{fitness} = \lambda_2 + \psi_2 * (\text{my-cell-A} / \text{max-cells-A}) / (R_S - R_{\min} + \epsilon) \quad (27)$$

If $A_{\max} \subset TA_C$, more closer the node sensing radius to R_{\max} , more fitness of it. So, fitness can be defined as (28):

$$\text{fitness} = \lambda_2 + \psi_2 * (\text{my-cell-A} / \text{max-cells-A}) / (R_{\max} - R_S + \epsilon) \quad (28)$$

Where ϵ shows very small positive number, λ_2 shows the minimum acceptable rate for fitness node, and ψ_2 is selected as the fitness rate doesn't exceed a given limit.

In (26), (27) and (28), *my-cells-A* shows all member of two A_S , A_{\max} sets of cell. The rate of *max-cells-A* is calculated by (29). In this relation $A_x(y)$ shows the A_x set of node y .

For each node N: (29)

$$\text{max-cells-A} = \max(a, b)$$

$$\begin{cases} a = \max(|A_S(n_j)| + |A_{\max}(n_j)|) \quad \forall n_j \text{ that is neighbor of N} \\ b = \text{my-node-A} = |A_S(N)| + |A_{\max}(N)| \end{cases}$$

Step5: Selecting the nodes with more fitness rate as parent nodes by using the roulette wheel and generating new population.

After calculating the fitness rate of the nodes, α percent of them (e.g. 50%) are selected as elite (parent) nodes for generating new population of sensing radius set. The parent nodes are selected randomly using roulette wheel. As a result, the probability of selecting the nodes with more fitness as parent ones is more. After determining the parent nodes, the new population of the nodes sensing radius is generated by applying cross over operation to the sensing radius of the parent nodes.

Step6: Mutating the sensing radius of nodes with predetermined mutation rate.

After generating the new population of the nodes sensing radius, the nodes sensing radius is mutated with the predetermined rate z .

The mutation operator is applied to each bit of an individual with a probability of mutation rate. When applied, a bit whose value is 0 is mutated into 1 and vice versa.

Step7: Checking the loop termination criteria and jumping to step 4.

The main loop of algorithm (steps 4, 5 and 6) continues until meeting one of the conditions stated bellow:

- The fitness rate of all active nodes, become better than TF (Threshold Fitness) threshold rate. And also these sensing radii can provide the full coverage of network.
- The number of performing the main loop of the algorithm exceed TC (Threshold Cycles) threshold rate. Then, the final active nodes and their sensing radius regarding the condition of their sets are determined. In the way that for each active node

if the A_{\max} set of node is not empty, R_{\max} sensing radius is selected. If A_S set is not empty, sensing radius R_S is selected, otherwise sensing radius R_{\min} will be selected.

IV. Simulation Results

In this section, we evaluate the proposed protocol (GA-SWAC) in terms of coverage ratio, number of active nodes and energy consumption and compare it with the OGDC [8] protocol with different sensing radius 8, 10, 12 meters. We assumed an area with a size of 150×150 which is divided in cells with size of 1×1 and deploy the sensor nodes randomly in this area. The number of nodes, N, in different configurations are considered as 100, 150, 200, 250, 300, 350, 400, 450 and 500 respectively. The sensing radius of the nodes in the proposed protocol is considered as $R_S \in [8, 23]$ and interval variations rate is equal to $\mu = 0.25$. As a result, the number of bits being required to preserve the selected sensing radius of each node equals $B = 6 + 1$. As shown in Fig. 3, due to its ability to adjust sensing radius, the proposed protocol is able to provide higher coverage rate with less energy consumption in comparison with OGDC. Due to the proposed protocol accuracy in adjusting the nodes sensing radius, it is able to provide the full coverage in less densities. Moreover, as the result of increasing nodes density, the proposed protocol decreases both nodes sensing radius and energy consumption, but OGDC protocol is not able to decrease energy consumption and to provide the full coverage in low density because of using fixed sensing radius.

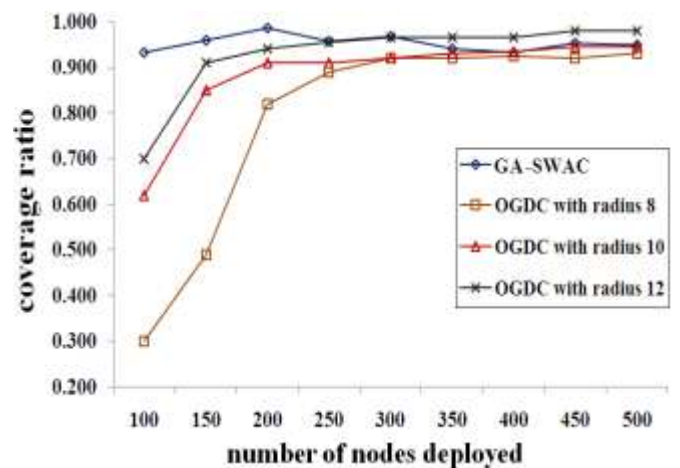


Fig. 3 The coverage rate of the networks with different deployed nodes

On the other hand, as shown in Fig. 4, the energy consumption per area in different configurations of our protocol is less than the OGDC protocol.

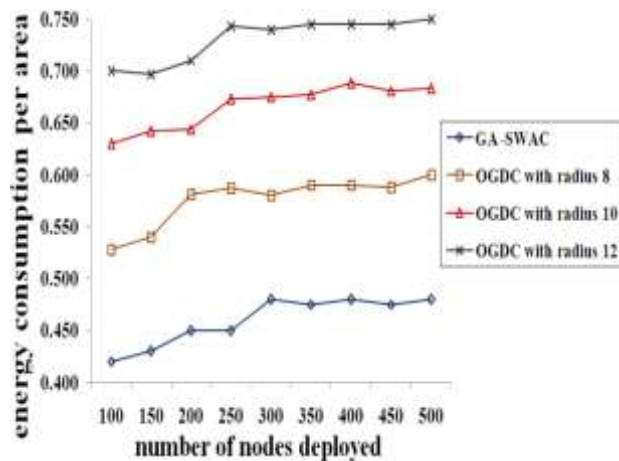


Fig. 4 The energy consumption per area in networks with different deployed nodes

Another noticeable point in the proposed protocol is the number of active nodes. As shown in Fig. 5, the number of active nodes is greater than other protocols while energy consumption in the proposed protocol is less in different configurations. As a result, the proposed protocol maintains more balance in using nodes energy so that it prolongs the network lifetime.

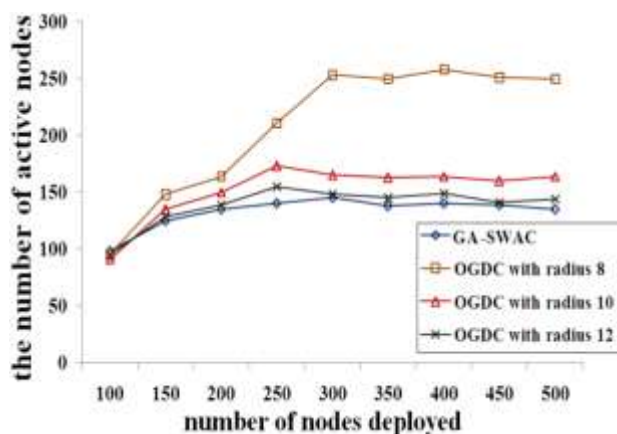


Fig. 5 The number of active nodes in the networks with different deployed nodes

V. Conclusion

In this paper, we proposed a genetic algorithm-based sleep-wake up area coverage protocol. The proposed protocol puts the minimum of nodes in active mode and adjusts the sensing radius of each active node to decrease the energy consumption while maintaining the network's coverage. The proposed protocol is simulated. The results demonstrated the efficiency of the proposed protocol in terms of coverage ratio, number of active nodes and energy consumption.

Acknowledgements

The authors wish to thank Islamic Azad University, Shahr-e-Qods branch for supporting this work through grants.

References

- [1] K. Li, J. Jie, "Optimal energy allocation in heterogeneous wireless sensor networks for lifetime maximization", *Journal of Parallel and Distributed Computing*, Vol. 72, pp. 902–916, 2012.
- [2] I. S. Alshawi, L. Yan, W. Pan, and B. Luo, "Fuzzy chessboard clustering and artificial bee colony routing method for energy efficient heterogeneous wireless sensor networks," *International Journal of Communication Systems*, vol. 26, no. 6, pp. 114–129, 2013.
- [3] F.-C. Jiang, D.-C. Huang, C.-T. Yang, C.-H. Lin, and K.-H. Wang, "Design strategy for optimizing power consumption of sensor node with Min(N,T) policy M/G/1 queuing models," *International Journal of Communication Systems*, vol. 25, no. 5, pp. 652–671, 2012.
- [4] J. Jia, J. Chena, G. Changa, Y. Wena, J. Songa, "Multi-objective optimization for coverage control in wireless sensor network with adjustable sensing radius", Elsevier, *Computers and Mathematics with Applications*, Vol. 57, pp. 1767_1775, 2009.
- [5] M. Chenait, B. Zebbane, H. Belbezza, H. Balli, N. Badache, "Distributed and stable energy-efficient scheduling algorithm for coverage in wireless sensor networks", 9th International Conference on Wireless Communications and Mobile Computing (IWCMC), Italy, pp. 418 – 423, 2013.
- [6] S. Fangyang, L. Chunlei Liu, Q. Bing, J. Yanqing, D. Caban, "Building Effective Scheduling Algorithms for Sensor Networks", 11th International Conference on Information Technology: New Generations (ITNG), Las Vegas, USA, pp. 403 – 406, 2014.
- [7] S. Sengupta, S. Das, M. Nasir, A. V. Vasilakos, W. Pedrycz, "An Evolutionary Multi-objective Sleep-Scheduling Scheme for Differentiated Coverage in Wireless Sensor Networks", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 42, No. 6, pp. 1093 – 1102, 2012.
- [8] D. Junzhao, L. Chang, L. Hui, S. Kewei, "Sleep-Wakeup Algorithms for Virtual Barriers of Wireless Sensor Networks in 3D Space", Sixth International Conference on Mobile Ad-hoc and Sensor Networks (MSN), USA, pp. 47-52, 2010.
- [9] L. Sherly Puspha Annabel, K. Murugan, "An energy efficient wakeup schedule and power management algorithm for wireless sensor networks", International Conference on Recent Trends In Information Technology (ICRTIT), India, pp. 314 – 319, 2012.
- [10] L. Aslanyan, H. Aslanyan, H. Khosravi, "Optimal node scheduling for integrated connected-coverage in wireless sensor networks", 5th International Conference on Computer Science and Information Technologies (CSIT), Jordan, pp. 1-13, 2013.
- [11] D. Diongue, O. Thiare, "ALARM: An energy aware sleep scheduling algorithm for lifetime maximization in wireless sensor networks", *IEEE Symposium on Wireless Technology and Applications (ISWTA)*, Malaysia, pp. 74-79, 2013.
- [12] D. Tian, N.D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks", in: *Proc. of the First ACM Intel workshop on Wireless Sensor Networks and Applications*, New York, USA, pp. 32_41, 2002.
- [13] H. Zhang, J.C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks", *Ad-hoc and Sensor Wireless Networks*, pp. 89_124, 2005.
- [14] S. Yang, F. Dai, M. Cardei, J. Wu, F. Patterson, "On connected multiple point coverage in wireless sensor networks", *Wireless Information Networks*, Vol. 13, pp. 289_301, 2006.
- [15] Z. Zhi-Hui, Z. Jun, D. Ke-jing, X. Jing, "Extended Binary Particle Swarm Optimization Approach for Disjoint Set Covers Problem in Wireless Sensor Networks", 17th annual Conference on Technologies and Applications of Artificial Intelligence (TAAI), Taiwan, pp. 327-331, 2012.
- [16] H. P. Gupta, S. V. Rao, T. Venkatesh, "Sleep scheduling for partial coverage in heterogeneous wireless sensor networks", Fifth International Conference on Communication Systems and Networks (COMSNETS), India, pp. 1-10, 2013.
- [17] B. Singh, D. K. Lobiyal, "Energy preserving sleep scheduling for cluster-based wireless sensor networks", Sixth International Conference on Contemporary Computing (IC3), India, pp. 97-101, 2013.
- [18] N. Bulusu, J. Heidemann, D. Estrin, "GPS-less low-cost outdoor localization for very small devices", *IEEE Personal Communications*, Vol. 7, pp. 28_34, 2000.