



# Optimal Scheduling of Directional Sensors with QoS Constraints to Enhance the Lifetime

M. Anitha<sup>1</sup> · P. Somasundaram<sup>2</sup> · S. Balaji<sup>3</sup> · S. Swetha<sup>4</sup>

Accepted: 7 November 2021 / Published online: 15 November 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

## Abstract

Wireless sensor networks have become one of the prominent and persuasive methods for surveillance of inaccessible physical areas and are employed in innumerable applications in various fields. Some applications may require directional sensor nodes in contrast to the conventional omnidirectional ones. Many challenges emerge during deployment of such sensor networks especially in terms of energy constraints. These sensor nodes are furnished with non-rechargeable energy sources which may lead to inefficiency in the network within a very short span of time. Conserving energy of the sensor nodes by organizing the sensor nodes into cover sets and actuating them one after the other, while ensuring maintenance of the complete coverage of all targets, is one of the common approaches to extend the network lifetime. This paper addresses Q-coverage problem of directional sensor networks in which each target is required to be covered by different number of sensor nodes for effective surveillance and Q-coverage constraints are considered while segregating the sensor nodes operating in different sensing directions to a different cover sets. An approach based on genetic algorithm is proposed to find an ideal solution to the directional Q-coverage problem and the simulation results confirm that the network lifetime is prolonged.

**Keywords** Directional sensor network · Q-coverage problem · Target coverage problem

---

✉ S. Balaji  
skbalaji28@gmail.com

M. Anitha  
aamizh@yahoo.com

P. Somasundaram  
mpsomasundaram@annauniv.edu

S. Swetha  
swethasundar94@gmail.com

<sup>1</sup> Department of Electrical and Electronics Engineering, St. Joseph's College of Engineering, Chennai, India

<sup>2</sup> Department of Electrical and Electronics Engineering, Anna University, Chennai, India

<sup>3</sup> School of Advanced Sciences, Vellore Institute of Technology Chennai Campus, Chennai, India

<sup>4</sup> Department of Mathematics, Anna University, Chennai, India

## 1 Introduction

Currently Wireless Sensor Networks (WSNs) find use in innumerable fields such as geographical area monitoring which includes military surveillance, geo-fencing; Medicine and healthcare applications like patient monitoring, clinical data collection; Industrial applications like monitoring machine health, water management, structural health management; environment-related applications that support disaster management, landslides and forest fire detection, water quality management, pollution control and monitoring [1]. A WSN is a group of spatially dispersed collaborative devices that work together to provide surveillance services and recording of information on various physical conditions within a real world environment. WSNs have become a pivotal research area due to many challenges aroused over time in terms of energy management, security and privacy, data aggregation, clock synchronization, unattended operation, maintenance, performance and latencies, decentralized architecture, etc., [2].

A sensor node is one of the most significant elements in a WSN and consists of a sensing device for observation, controllers for information processing, communication devices for data transfer and power sources for energy supply. A large number of sensor nodes are dispersed over the area of surveillance based on different topologies depending upon the quality requirements and the type of application. Data collected by the sensors or actuators from a physical environment are transferred to an application in a main server for processing with the help of gateways that constitute a data path for exchange of information [3].

Specific applications such as multimedia sensor networks may require the deployment of sensor nodes capable of directional sensing i.e. enabling surveillance in a particular direction within a specific sensing range. This is made possible through employment of directional sensor nodes in the network instead of the conventional ones that facilitate omnidirectional sensing. Such directional sensors are becoming increasingly popular because of their manufacturing size and cost. They are specially designed to serve a number of civil as well as military applications. A study of characteristics such as coverage, scheduling, maintenance, connectivity and deployment of directional sensor nodes which vary when compared to networks with omnidirectional sensor nodes is important.

Several constraints to be considered while designing a WSN which are deal with power supply, memory, hardware circuits, data size, deployment etc. One such constraint is exiguous energy supply to the sensor nodes as the sensor devices are generally established in an unattended area where recharging or replacing the energy source is either complicated or impossible and hence conservation of energy becomes important. Enhancing energy efficiency of a WSN is one of the main subjects on which most of the research works directed in recent years [4, 5]. One of the common strategies for energy conservation is designing protocols based on which sensors are scheduled to operate in a network. In directional sensor networks, these protocols facilitate operation of selected sensor nodes in specific sensing directions to ensure complete coverage (Target or Point coverage [6], Area Coverage [7], Barrier Coverage [8]), while other inactive sensor nodes are reserved for further utilization, thus extending the lifetime of the network. Each sensor node can be actuated in only one specific direction at any point of time. Target Coverage, in particular, deals with a set of points of interest called targets that are established in an environment and sensors are deployed and actuated to cover all these target points. Coverage can be a quality element in service requirement in some applications i.e. targets need to be covered by more than one sensor node for effective data collection. Target coverage in such applications is generalized as  $K$ -coverage which aims to ensure that each target is monitored by at least  $K$

different sensor nodes. In a directional sensor network,  $K$ -coverage aims to ensure coverage of every target by at least  $K$  different sensor nodes, each operated in a different specific direction.

Some applications involve targets associated with different importance and require to be covered by different number of sensors based on the importance. For such applications, a  $Q$ -coverage problem has been proposed in which the minimum number of sensors (sensing directions of various sensor nodes in case of directional sensor networks) required to cover each target is different from others. The sensing directions of the sensor nodes are designed for actuation in cover sets one after another with the requirement of each sensing direction in a cover to cover at least one target. This paper addresses the  $Q$ -coverage problem of directional sensor networks with respect to target coverage and a solution based on genetic algorithm is discussed.

## 2 Related Works

Enhancing energy efficiency of a WSN is one of the main goals for many researchers in the recent years. The SET  $K$ -COVER problem was proposed in [7] to find the maximum number of disjoint set covers consisting of a set of sensor nodes that can ensure complete coverage of the physical area and these set covers can be activated one after another to extend the lifetime of the network. Cardei et al. [9] have proposed Maximum Set Cover problem in which sensors are scheduled for actuation in different set covers that ensure coverage of specific points in the physical area referred to as targets. In Maximum Disjoint Set Covers problem [10], no sensor can be grouped into any multiple set cover this was to be ensured. Different protocols have been proposed to solve the target coverage problem in a WSN, which dealt with omnidirectional sensors.

Directional sensor networks facilitate operation of sensor nodes in specific sensing directions. Such directional sensors have many applications including video sensor networks that provide useful visual information [11], image sensors [12], ultrasonic sensors etc. The target coverage problem of a directional sensor network in which the sensors can be operated in different directions or orientations was modeled as Maximum Coverage with Minimum Sensors (MCMS) problem in [13] and has also been solved using an Integer Linear Programming approach that provides the most appropriate solution and a centralized greedy algorithm that provides an approximate solution but can be used in a large scale. The aim of the MCMS problem was to find the minimum number of directions that can cover the maximum number of targets. Cai et al. have proposed several algorithms for the multiple directional cover sets (MDCS) problem which aims at organizing the directions of sensors into disjoint groups [14]. A priority augmented graph and direction partition algorithm have been discussed in [15] to ensure coverage of targets based on different priorities by directional sensors. A column generation algorithm has been proposed for lifetime maximization in any directional sensor network [16]. It consists of sensors with predefined directions and also for the network in which sensing directions can be varied. A search heuristic to find non-disjoint cover sets for Multiple Directional Cover Sets problem has been proposed and aimed to compute a group of cover sets that are non-disjoint and in each of these cover sets the directions ensure coverage of all the targets [17]. A randomized technique is used to approximate the network lifetime by scheduling the directional sensors for efficient target coverage [18].

Genetic algorithms based solutions are proposed for target coverage problem in directional sensor networks [19, 20]. These works deal with single coverage of targets i.e. all targets are entitled for coverage by at least one sensor meant only for efficient data collection. A  $Q$ -coverage problem [21] in a WSN is formulated for addressing the different QoS constraints faced by targets and various solutions are proposed [22, 23]. These solutions for  $Q$ -coverage problems deal with wireless omnidirectional sensor networks. A protocol for identifying the cover sets for  $Q$ -coverage in directional sensor networks with bounded service delay has been proposed by formulating the problem into a mixed integer programming problem [24]. A scheduling scheme is provided for reducing the energy consumed for rotating the directional sensor nodes while switching from one group to the next [25]. Quality based target coverage through use of directional sensor nodes was modeled as integer programming problem and two heuristics were discussed [26]. In this paper, we propose the Genetic Algorithm for Directional  $Q$  coverage (GA-DQC) for finding the maximum number of cover sets.

### 3 Problem Description

Consider a directional sensor network with a set of  $n$  sensor nodes  $S = \{S_1, S_2, S_3, \dots, S_n\}$  that are distributed in a particular locale for observation of a set of  $m$  targets  $T = \{T_1, T_2, T_3, \dots, T_m\}$ . A particular target  $T_j$  in the plane  $(x_j, y_j)$  is said to be covered by a sensor node  $S_i$  in the plane  $(x_i, y_i)$ , if it satisfies the condition  $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \leq R$ , where  $R$  is the sensing range of the sensor,  $1 \leq i \leq n$  and  $1 \leq j \leq m$ .

The sensor coverage of the targets in a network is represented in form of a matrix,

$$\phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \dots & \phi_{1m} \\ \phi_{21} & \phi_{22} & \dots & \phi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{n1} & \phi_{n2} & \dots & \phi_{nm} \end{bmatrix}, \text{ where } \phi_{ij} = \begin{cases} 1, & \text{if sensor } S_i \text{ covers target } T_j \\ 0, & \text{otherwise} \end{cases}$$

In contrast to the conventional sensors which are omnidirectional in nature, directional sensors are capable of being actuated in varied sensing directions at different instances of time. For each sensor node  $S_i$  in set  $S$ , there can be  $p$  possible sensing directions and the set of all sensing directions for sensors in the network is represented by  $D = \{D_{11}, D_{12}, \dots, D_{1p}, D_{21}, D_{22}, \dots, D_{2p}, \dots, D_{n1}, \dots, D_{np}\}$  and the number of directions  $p$  is determined by the desired sensing angle  $\theta$  of the sensors as  $p = 2\pi/\theta$ . The omnidirectional sensing area within the sensing radius  $R$  of the sensors is partitioned into  $p$  different directional sensing areas. The  $k$ th sensing direction of  $i$ th sensor is represented as  $D_{ik}$ . A particular target  $T_j$  is said to be covered by  $D_{ik}$ ,  $k$ th sensing direction of sensor  $S_i$ , if value of  $\phi_{ij}$  is 1 in the sensor coverage matrix  $\phi$  and the location of target  $T_j$  lies in the area within the  $k$ th direction of sensor  $S_i$ . A target  $T_j$  can be covered by only one operable sensing direction of a sensor  $S_i$ .

The target coverage in terms of directions of each of the sensors in a network is represented as a directional sensor coverage matrix,

$$\omega = \begin{bmatrix} \omega_{11} & \omega_{12} & \dots & \omega_{1m} \\ \omega_{21} & \omega_{22} & \dots & \omega_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{n1} & \omega_{n2} & \dots & \omega_{nm} \end{bmatrix}, \text{ where } \omega_{ij} = \phi_{ij} \times (\alpha_{ij}/\theta).$$

The angle at which a particular target  $T_j$  lies with respect to the sensor  $S_i$  is calculated as  $\alpha_{ij} = \arctan 2(y_j - y_i, x_j - x_i)$  in degrees. If  $\arctan 2$  function returns a negative value, it is converted to positive by adding  $2\pi$ . For example, assume the directional sensing sensor node  $S$  as shown in Fig. 1 at position  $(0, 0)$  could operate in four different sensing directions with 90 degrees sensing angle and sensing range  $R = 60$ . Assume a target  $T$  is located at location  $(-40, -40)$  on the plane. Since the target  $T$  is located within the sensing range of the sensor  $S$ , the sensor coverage  $\phi_{ST}$  is 1. The result of  $\arctan 2(y_T - y_S, x_T - x_S)$  function is obtained as  $-135$ (when converted to degrees) and  $2\pi$  is added as the result is negative ( $\alpha_{ST} = 225$ ). The direction coverage for the target  $T$  and sensor  $S$  in figure is given by  $\omega_{ST} = 1 \times (225/90) = 2.5$  which is rounded off to 3. Thus the target  $T$  is covered by third sensing direction of sensor  $S$ .

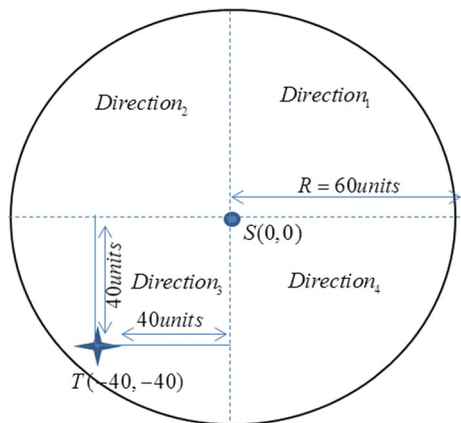
In general, the target coverage problem deals with the observation of the entire set of targets in a network while managing the energy consumption of the sensor nodes and thus extending the lifetime of the network. One of the approaches is to partition the sensor nodes into sensor covers and to actuate these covers in succession. But it is ensured that each cover set monitors the complete set of  $m$  targets in that network.

Most of the communication and scheduling protocols of a WSN are aimed at minimizing the energy consumption of the network. Every sensor  $S_i$  in the network is equipped initially with a definite amount of energy,  $e$ . Once the sensor nodes are installed in the environment, recharge or replacement of the source of energy in the sensors is difficult. When a particular sensor with initial energy  $e$  is activated in any one of the directions,  $e/p$  amount of energy gets dissipated. The residual energy in the sensor, given by  $(p - 1)e/p$ , can be used for further consumption during the lifetime of the network.

In some applications, there may be a requirement for a specific target  $T_j$  to be covered by at least  $K$  number of sensors at a time to ensure desired quality of service. A specific coverage parameter  $K$  is defined as the minimum number of sensor nodes to cover each target in set  $T$ , such that every target  $T_j$  lies within the sensing range of at least  $K$  different sensor nodes. The coverage parameter, in specific applications, can also be different for each target based on the difference in importance or QoS requisites of the targets. In such cases, a set  $Q = \{q_1, q_2, q_3, \dots, q_m\}$  denotes the minimal number of sensor nodes  $q_j$  that are necessary to cover each target  $T_j$  at any specific moment.

The set  $u = \{u_1, u_2, u_3, \dots, u_m\}$ , where  $u_k$  denotes the total number of sensors in the network that cover the target  $T_k$ . If a target  $T_k$  is covered by  $u_k$  number sensors, then it

**Fig. 1** An illustration to find direction coverage



is possible to form a maximum of  $(u_k \times e)/q_j$  covers in case of  $Q$ -coverage problem and  $(u_k \times e)/K$  covers in case of  $K$ -coverage problem. The target  $T_k$  will be left uncovered or would not meet the required  $Q$ -coverage or  $K$ -coverage constraints if  $(u_k + 1)^{th}$  cover is determined. The minimum value among the set  $u = \{u_1, u_2, \dots, u_m\}$  is set as the upper bound,  $ub$ . Since a target is covered by only one direction of a particular sensor, the upper bound for the directional sensor network is same as the upper bound for omnidirectional sensor network and is defined as.

$$ub = \text{Min}_j \left( \left( \sum_i \phi_{ij} \right) \times v \right), \text{ where } v = e/q_j \text{ for } Q\text{-coverage problem and } v = e/K \text{ for } K\text{-coverage problem.}$$

### 3.1 Illustration

The sample network depicted in Fig. 2 consists of a set of 3 targets observed by a set of 6 sensor nodes each with a fixed sensing range  $R$ , initial energy  $e = 3$  and fixed sensing angle of  $\theta = 120$ . Let the randomly generated  $Q$ -coverage constraints for the targets be defined by  $Q = [2 \ 1 \ 2]$  which implies that targets  $T_1, T_2$  and  $T_3$  are to be covered by at least 2, 1 and 2 sensor nodes respectively in order to meet the QoS requisites.

The sensor coverage ( $\phi$ ) and direction coverage ( $\omega$ ) for the sample network depicted in Fig. 2 are represented in a matrix form as follows:

$$\phi = \begin{matrix} & T_1 & T_2 & T_3 \\ S_1 & \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ S_2 & \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \\ S_3 & \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \\ S_4 & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \\ S_5 & \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ S_6 & \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \end{matrix} \quad \omega = \begin{matrix} & T_1 & T_2 & T_3 \\ S_1 & \begin{bmatrix} 3 & 0 & 0 \end{bmatrix} \\ S_2 & \begin{bmatrix} 2 & 3 & 0 \end{bmatrix} \\ S_3 & \begin{bmatrix} 0 & 2 & 3 \end{bmatrix} \\ S_4 & \begin{bmatrix} 0 & 0 & 2 \end{bmatrix} \\ S_5 & \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ S_6 & \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \end{matrix}$$

sum = ( 3 3 3 )

The upper bound for the coverage problem in the given sample network is calculated as minimum value in the set,  $u = \{ \lfloor (3 \times e)/2 \rfloor, \lfloor (3 \times e)/1 \rfloor, \lfloor (3 \times e)/2 \rfloor \}$ , where  $e$

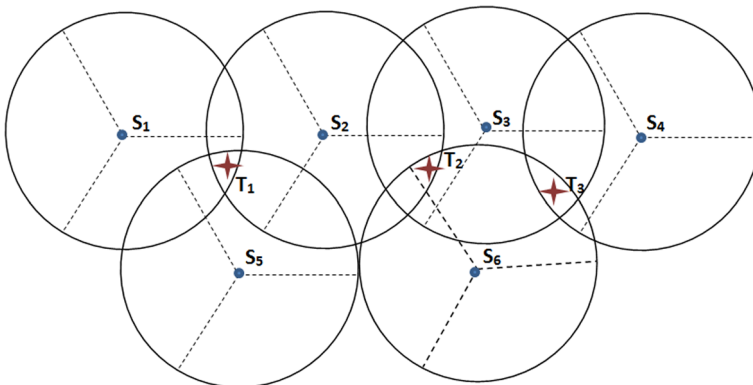


Fig. 2 A sample directional sensor network with 6 sensors and 3 targets

is the initial energy of the sensors. The minimum value among the set  $u, 4$  is the upper bound ( $ub$ ) of the network in Fig. 2. The possible set of covers that could be obtained for the network are  $Cover_1 = \{D_{13}, D_{23}, D_{33}, D_{42}, D_{51}\}$ ,  $Cover_2 = \{D_{13}, D_{42}, D_{51}, D_{61}\}$ ,  $Cover_3 = \{D_{13}, D_{22}, D_{33}, D_{61}\}$  and  $Cover_4 = \{D_{22}, D_{42}, D_{51}, D_{61}\}$ .

## 4 Enhancing Lifetime of Directional Sensor Network

Genetic algorithm is a heuristic search algorithm used for solving optimization problems which imitate certain characteristics of natural evolution process. The individual solutions are upgraded repeatedly to achieve an ideal solution. In order to steer towards a potential solution for a specific problem, the individual solutions are encoded as chromosomes and refined using three main operators namely selection, crossover and mutation. A selection operator randomly chooses individual candidates from a population based on the fitness criterion for introduction into the successive stages of the algorithm. The crossover operator diversifies the population from one generation to the next by producing child solutions from two parent solutions. Mutation operator facilitates alteration of one or more values in individual solutions based on a user defined probability to vary the population in successive generations. The fittest population survives at each successive generation and finally an optimal solution is obtained after maximum number of generations. The pseudo code of the genetic algorithm is depicted as:

---

**Input:** *Initial\_pop, generation\_size*

---

**Output:** *Coversets*

---

```

1      for 1 to generation_size
2          Parent_fitness = Fitness(Initial_pop);
3          Offspring = Crossover(Initial_pop);
4          Offspring = Mutation(Offspring);
5          Child_fitness = Fitness(Offspring);
6          Initial_pop = Selection(Initial_pop, Offspring, Parent_fitness, Child_fitness);
7      End for;
9      Coversets = Initial_pop;

```

---

### 4.1 Representation

Representation involves encoding the solutions in a form that is suitable for evolutionary computations. The solutions are encoded in the form of a matrix with  $n$  rows and  $ub$  columns, where  $n$  is the number of sensor nodes and  $ub$  is the upper bound of the network. Each entry in the matrix is randomly assigned sensing directions of the sensors that cover at least one target. These sensing directions are distributed over the covers based on the residual energy of the sensors. Since the expected number of covers is predicted as  $ub$  and the solution sets are confined to  $ub$  number of columns, the ideal solution is obtained with minimal amount of computation. A sample representation of the matrix is given by:

$$\text{Population} = \begin{matrix} & C_1 & C_2 & \dots & C_{ub} \\ \begin{matrix} S_1 \\ S_2 \\ \cdot \\ \cdot \\ S_n \end{matrix} & \begin{bmatrix} \beta_{1,1} & \beta_{1,2} & \dots & \beta_{1,ub} \\ \beta_{2,1} & \beta_{2,2} & \dots & \beta_{2,ub} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \beta_{n,1} & \cdot & \cdot & \beta_{n,ub} \end{bmatrix} \end{matrix}$$

where  $\beta_{i,j}$  is a randomly selected sensing direction of  $S_i$  that covers at least one target in set  $T$  and  $1 \leq \beta_{i,j} \leq p$ .

### 4.2 Generation of Initial Population

Genetic algorithm involves random generation of a set of 100 initial populations that participate in the successive stages of the algorithm. The initial population is formed by identifying the sensing directions of each sensor that cover at least one target from the direction coverage matrix. These identified sensing directions are randomly distributed over the  $e$  number of columns out of the  $ub$  number of columns for the respective sensors in the population matrix where  $e$  denotes the energy of the sensors. The remaining column values of each sensor are set to zero. In a similar manner, other initial populations are generated randomly.

The matrices  $pop_1$  and  $pop_2$  form the first two initial population matrices for the network mentioned in Fig. 2. The sensing directions of each sensor that covers at least one target are identified to be  $S_1 = \{3\}$ ,  $S_2 = \{2,3\}$ ,  $S_3 = \{2,3\}$ ,  $S_4 = \{2\}$ ,  $S_5 = \{1\}$ ,  $S_6 = \{1\}$  and the energy of the sensors  $e = 3$ . In population matrix  $pop_1$ , for sensor node,  $S_1$  the direction 3 is randomly assigned to 3 columns as the total energy is 3 and for sensor node  $S_2$  the directions 2 and 3 are randomly assigned to 3 columns. In a similar manner, the direction values are generated for all the sensors in the  $pop_1$  matrix. The remaining values are set to zero.

$$\begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{matrix} \begin{bmatrix} 3 & 3 & 3 & 0 \\ 3 & 0 & 2 & 2 \\ 3 & 3 & 0 & 0 \\ 2 & 0 & 2 & 2 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix} \quad \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{matrix} \begin{bmatrix} 3 & 0 & 3 & 3 \\ 2 & 3 & 0 & 2 \\ 2 & 2 & 0 & 3 \\ 2 & 2 & 0 & 2 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

### 4.3 Fitness Function

A fitness function is designed for analysis of the extent of a given solution to the ideal solution to be obtained. Selection of population in each generation is based on the outcome of the fitness function. The fitness value of a population is determined by the sum of fitness values of all individual covers in that population. The fitness value of a cover is one if it covers all the targets with respect to the  $Q$ -coverage or  $K$ -coverage constraints of each target. If not, the fitness value is zero.

The set of sensors with respective sensing directions that belong to  $C_1$  in the matrix  $pop_1$  is  $C_1 = \{D_{13}, D_{23}, D_{33}, D_{42}, D_{51}, D_{61}\}$ . The sensor nodes  $S_1, S_2$  and  $S_3$  are activated



in the third direction while the sensor nodes  $S_5, S_6$  are to be activated in the first direction and  $S_4$  in the second directions. The targets covered by the sensors in the set described by  $C_1$  are  $S_1 = \{T_1\}, S_2 = \{T_2\}, S_3 = \{T_3\}, S_4 = \{T_3\}, S_5 = \{T_1\}, S_6 = \{T_2, T_3\}$ . Thus, the targets  $T_1$  and  $T_2$  are covered twice and target  $T_3$  is covered thrice. This signifies the cover  $C_1$  covering all the targets with respect to their coverage constraints as mentioned in set  $Q = [2 \ 1 \ 2]$  and its fitness value is one. Similarly, the fitness values of  $C_1, C_2, C_3, C_4$  in  $pop_1$  are 1, 0, 1 and 0 respectively. Thus the fitness value of  $pop_1$  is 2, which is the sum of the individual fitness values of  $C_1, C_2, C_3$  and  $C_4$ .

### 4.4 Selection

The selection operator is responsible for the selection of candidates on the basis of the fitness values for the formation of the mating pool from the initial population and the random selection of parent populations from the mating pool to produce offspring. Based on the fitness function results, the best individuals are selected for the next generation.

### 4.5 Crossover

The crossover operator remodels the population in each generation by producing offsprings through exchange of genes between two parent solutions at specific locations determined by the randomly chosen crossover point which could vary between 1 and  $n$ . The result of performing crossover operation for  $pop_1$  and  $pop_2$  with crossover point chosen to be 3 is given by  $Offspring_1$  and  $Offspring_2$ .

$$\begin{array}{c}
 \begin{array}{cccc}
 C_1 & C_2 & C_3 & C_4 \\
 S_1 & \left[ \begin{array}{cccc} 3 & 3 & 3 & 0 \end{array} \right] \\
 S_2 & \left[ \begin{array}{cccc} 3 & 0 & 2 & 2 \end{array} \right] \\
 S_3 & \left[ \begin{array}{cccc} 3 & 3 & 0 & 0 \end{array} \right] \\
 S_4 & \left[ \begin{array}{cccc} 2 & 2 & 0 & 2 \end{array} \right] \\
 S_5 & \left[ \begin{array}{cccc} 1 & 1 & 0 & 1 \end{array} \right] \\
 S_6 & \left[ \begin{array}{cccc} 0 & 1 & 1 & 1 \end{array} \right] \\
 \end{array} \\
 \end{array}
 \quad
 \begin{array}{c}
 \begin{array}{cccc}
 C_1 & C_2 & C_3 & C_4 \\
 S_1 & \left[ \begin{array}{cccc} 3 & 0 & 3 & 3 \end{array} \right] \\
 S_2 & \left[ \begin{array}{cccc} 2 & 3 & 0 & 2 \end{array} \right] \\
 S_3 & \left[ \begin{array}{cccc} 2 & 2 & 0 & 3 \end{array} \right] \\
 S_4 & \left[ \begin{array}{cccc} 2 & 0 & 2 & 2 \end{array} \right] \\
 S_5 & \left[ \begin{array}{cccc} 1 & 1 & 1 & 0 \end{array} \right] \\
 S_6 & \left[ \begin{array}{cccc} 1 & 0 & 1 & 1 \end{array} \right] \\
 \end{array} \\
 \end{array}
 \end{array}$$

### 4.6 Mutation

The mutation operator further refines the population matrices of successive generations by altering the sequence of a particular sensor chosen at random on the basis of probability. The sensing directions of the chosen sensor are altered in between the covers in such a way to ensure satisfactory fitness for the obtained offspring. The outcome of mutation operation on  $Offspring_1$  and  $Offspring_2$  when the mutation point is set as 3

$$\begin{array}{c}
 \begin{matrix} C_1 & C_2 & C_3 & C_4 \end{matrix} \\
 \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{matrix} \begin{bmatrix} 3 & 3 & 3 & 0 \\ 3 & 0 & 2 & 2 \\ 3 & 0 & 3 & 0 \\ 2 & 2 & 0 & 2 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\
 \text{Offspring}_1 =
 \end{array}
 \qquad
 \begin{array}{c}
 \begin{matrix} C_1 & C_2 & C_3 & C_4 \end{matrix} \\
 \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{matrix} \begin{bmatrix} 3 & 0 & 3 & 3 \\ 2 & 3 & 0 & 2 \\ 2 & 0 & 2 & 3 \\ 2 & 0 & 2 & 2 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix} \\
 \text{Offspring}_2 =
 \end{array}$$

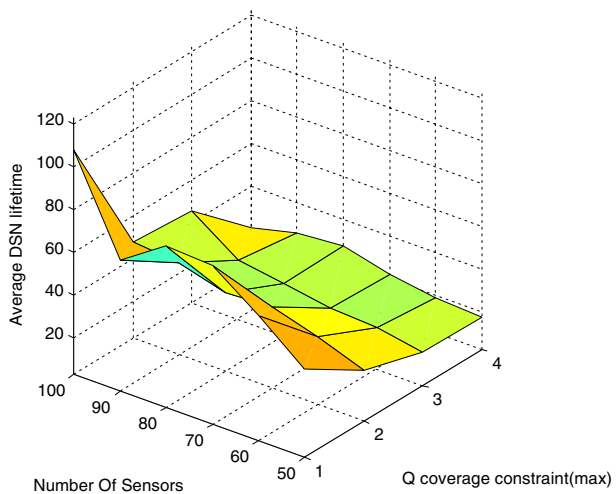
The best solution is identified on the basis of fitness values of  $pop_1, pop_2, Offspring_1$  and  $Offspring_2$  which are found to be 2, 1, 4 and 3 respectively. The fitness value of the population matrix  $Offspring_1$  is the highest and hence identified as the ideal solution given in Fig. 2 and the cover sets are  $Cover_1 = \{D_{13}, D_{23}, D_{33}, D_{42}, D_{51}\}, Cover_2 = \{D_{13}, D_{42}, D_{51}, D_{61}\}, Cover_3 = \{D_{13}, D_{22}, D_{33}, D_{61}\}$  and  $Cover_4 = \{D_{22}, D_{42}, D_{51}, D_{61}\}$ .

### 5 Simulation

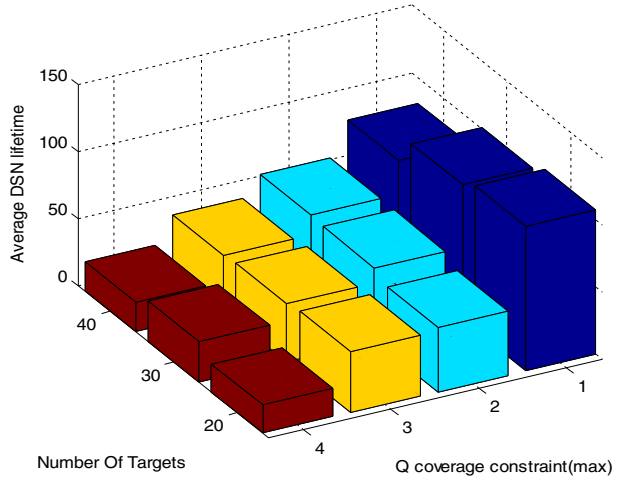
The proposed algorithm has been simulated using MATLAB. Simulations are implemented in a Windows7 operating system on a Intel Core processor with 4 GB RAM and 2.3 GHz speed. The sensors and targets are located randomly on a plane of  $100 \times 100$  square units. Since the sensors are randomly deployed, there may be slight variation in results based on the random locations at which they are placed in different instances. For each experiment, five instances are simulated and the average is calculated. The number of generations and the population size of the genetic algorithm are set to a fixed value of 100 in all the experiments. The mutation rate is fixed at a constant rate of 0.7.

The results for GA-DQC problem with the number of sensors varying between 50 and 100 and the number of targets from 20 to 40 with a fixed sensing range  $R = 60$  and a fixed sensing angle  $\theta = 120$  are simulated and the average lifetime is calculated. Figures 3 and 4 show the variation in lifetime of the network with respect to variations in number of sensor nodes, number of targets and different maximum  $Q$ -Coverage constraints ( $Q_{max}$ ). The different  $Q$ -coverage values of targets are randomly assumed between 1 and  $Q_{max}$ . The  $Q_{max}$

**Fig. 3** Network lifetime for varying number of sensors and Q coverage constraints

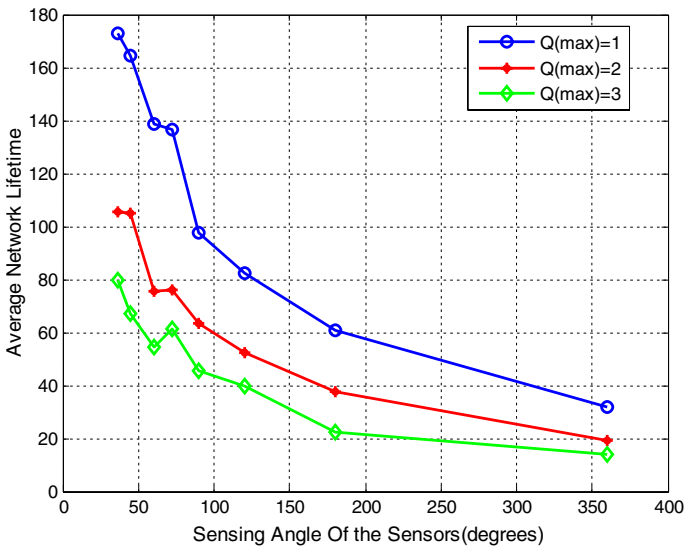


**Fig. 4** Network lifetime for varying number of targets and Q coverage constraints



values are varied as 1, 2, 3 and 4. When  $Q_{max}$  value is set to 1, the problem is generalized as a  $K$ -coverage problem or a single coverage problem where all the targets are required to be covered by at least one sensor node. Figure 3 represents Lifetime for varying number of sensors, 20 targets and varying  $Q$ -coverage constraints and Fig. 4 represents Lifetime of the network for 100 sensors with varying number of targets and  $Q$ -coverage constraints. It is found, that for all varying  $Q$ -coverage values, the lifetime tends to increase with an increase in the number of sensor nodes in general.

The effect of variations in sensing angle of the sensors on the lifetime of a network with 100 sensors and 40 targets is shown in Fig. 5 for  $Q_{max}$  values 1, 2 and 3. The sensing angles are varied as 36, 45, 60, 72, 90, 120, 180 and 360 degrees, while the sensing range is set at a constant of 60 units. The initial energy of the sensors is assumed to be the number of



**Fig. 5** Effect of varying sensing angles in the network lifetime

total sensing directions in each sensor. There is only a decrease in the lifetime when the sensing angle of sensors is increased for each of the  $Q_{\max}$  constraints. This is so because, the number of different directions in which a sensor can be activated is decreased when the sensing angle increases which limits the number of possible cover sets that can be formed.

Figure 6 shows the results of the algorithm simulated for varying number of sensors and targets with  $K$ -coverage constraints as 1 and 2. Similar to simulations of  $Q$ -coverage, the sensing angle is set to 120 degrees and sensing range is 60 units, with variations in the number of sensor nodes between 50 and 100, variations in the number of targets between 20 and 40. The average lifetime of  $Q$ -coverage problem is found to be slightly higher in most of the cases as compared to lifetime of the  $K$ -coverage problem (i.e. when comparing results of networks with maximum  $Q$ -coverage and  $K$ -coverage constraint set as 2).

The simulations in Tables 1, 2, 3, 4 and 5 are aimed at comparison of the results of the proposed GA-DQC with the results of existing algorithms [8, 11, 12] for complete target coverage problem ( $Q_{\max} = 1$ ) in which omnidirectional sensors are used in the network ( $\theta = 360$ ). The simulations are tabulated for varied number of targets and varying number of sensor nodes with differing sensing ranges deployed on a  $500 \times 500$  plane. The results of improved Memetic Algorithm (iMA) [27], Memetic Algorithm (MA) [28], integer based Genetic Algorithms (iGA) [29], MCMCC [7], and the results of the proposed GA-DQC are compared.

The average upper bound, average and standard deviation of network lifetime of five instances of proposed GA-DQC are tabulated in columns 2, 3 and 4 respectively. In columns 5 and 6, the average upper bound and average lifetime of 100 instances of iMA are tabulated. Column 7 represents the average upper bound of MA, iGA and MCMCC for varying number of sensor nodes with different sensing ranges and varying number of targets. The averages and standard deviations of network lifetime obtained by MA, iGA and MCMCC are specified in columns 8 to 13 in the corresponding tables.

Table 1 shows the results of small scale networks with 10 targets and 90 sensor nodes with varying sensing ranges between 100 and 500. Table 2 shows the results for large scale networks with 300 sensors and 500 targets with varying sensing ranges between 100 and 500. The lifetime obtained in the simulation instances of GA-DQC have reached the upper bound in almost all cases, which implies the optimal nature of the solution. The GA-DQC algorithm tends to assume the upper bound taking into consideration the coverage constraints and energy of the sensors, which minimize the search space and facilitate obtaining of an optimal solution. The lifetime increases with increase in the sensing ranges of the sensor nodes.

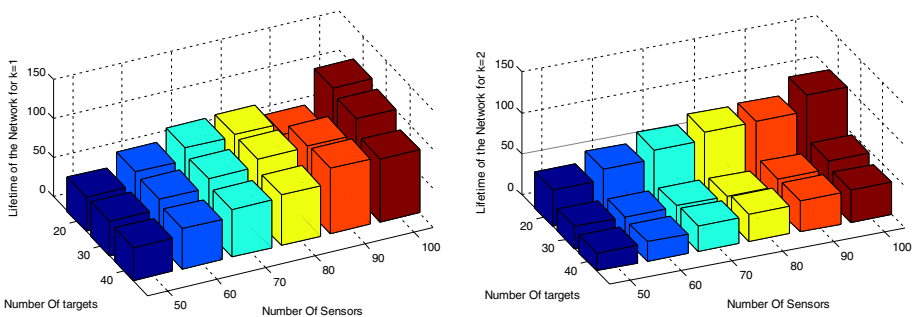


Fig. 6 Network lifetime for varying number of sensors and targets with K-coverage constraints 1 and 2

**Table 1** Network lifetime vs. sensing ranges (for small scale networks)

| Sensing range | GA-DQC   |          | iMA      |                | ub (MCMCC, iGA, MA) |       | Lifetime |      | MCMCC |      |
|---------------|----------|----------|----------|----------------|---------------------|-------|----------|------|-------|------|
|               | ub (avg) | Lifetime | ub (Avg) | Lifetime (Avg) | MA                  | iGA   | Avg      | SD   | Avg   | SD   |
|               |          |          |          |                |                     |       |          |      |       |      |
| 100           | 5.40     | 5.40     | 4.19     | 4.19           | 4.13                | 4.13  | 4.13     | 1.61 | 4.13  | 1.61 |
| 150           | 13.00    | 13.00    | 9.82     | 9.82           | 9.84                | 9.84  | 9.84     | 2.64 | 9.78  | 2.64 |
| 200           | 16.80    | 16.80    | 16.91    | 16.91          | 16.77               | 16.77 | 16.77    | 4.47 | 16.41 | 4.47 |
| 250           | 25.60    | 25.60    | 24.91    | 24.91          | 25.18               | 25.18 | 25.18    | 4.82 | 24.55 | 4.82 |
| 300           | 40.40    | 40.40    | 34.89    | 34.89          | 34.78               | 34.78 | 34.76    | 6.10 | 33.97 | 6.09 |
| 350           | 49.00    | 49.00    | 47.46    | 47.31          | 45.54               | 45.54 | 45.28    | 6.49 | 44.67 | 6.33 |
| 400           | 55.00    | 55.00    | 56.75    | 56.21          | 58.31               | 58.31 | 57.39    | 5.89 | 56.70 | 5.74 |
| 450           | 68.00    | 68.00    | 69.86    | 69.27          | 70.37               | 70.37 | 69.39    | 5.81 | 69.12 | 6.12 |
| 500           | 79.20    | 79.20    | 80.25    | 79.81          | 80.53               | 80.53 | 80.06    | 4.27 | 80.03 | 4.55 |

**Table 2** Network lifetime vs. sensing ranges (for large scale networks)

| Sensing range | GA-DQC   |        | iMA            |        | ub (MCMCC, iGA, MA) |      | LifETIME |      | MCMCC  |      |
|---------------|----------|--------|----------------|--------|---------------------|------|----------|------|--------|------|
|               | LifETIME |        | LifETIME (Avg) |        | MA                  |      | iGA      |      | MCMCC  |      |
|               | ub (avg) | Avg    | ub (Avg)       | Avg    | Avg                 | SD   | Avg      | SD   | Avg    | SD   |
| 100           | 7.20     | 7.20   | 8.57           | 8.57   | 8.69                | 2.17 | 8.65     | 2.11 | 8.28   | 1.93 |
| 150           | 21.00    | 21.00  | 20.06          | 19.94  | 20.20               | 3.35 | 20.07    | 3.19 | 19.64  | 3.01 |
| 200           | 34.60    | 34.60  | 37.73          | 37.44  | 36.01               | 4.36 | 35.87    | 4.21 | 35.59  | 4.15 |
| 250           | 55.80    | 55.80  | 57.53          | 57.28  | 58.16               | 5.03 | 57.46    | 4.32 | 56.46  | 4.42 |
| 300           | 85.60    | 85.60  | 84.19          | 83.79  | 84.71               | 5.66 | 78.85    | 4.04 | 80.55  | 4.01 |
| 350           | 118.00   | 118.00 | 115.68         | 115.63 | 115.57              | 6.02 | 100.07   | 5.57 | 110.34 | 4.40 |
| 400           | 153.20   | 153.20 | 152.68         | 151.79 | 151.43              | 4.99 | 125.18   | 7.02 | 142.51 | 4.09 |
| 450           | 200.80   | 200.80 | 193.24         | 178.94 | 193.62              | 4.00 | 149.43   | 8.04 | 174.71 | 4.41 |
| 500           | 239.20   | 239.20 | 238.62         | 209.82 | 238.08              | 5.28 | 179.30   | 8.73 | 207.86 | 5.29 |

**Table 3** Network lifetime vs. number of targets (for large scale networks)

| Targets | GA+DQC   |          | iMA      |                | ub (MCMCC, iGA, MA) |        | LifETIME |        |       |        |       |
|---------|----------|----------|----------|----------------|---------------------|--------|----------|--------|-------|--------|-------|
|         | ub (avg) | LifETIME | ub (Avg) | LifETIME (Avg) | iGA                 | MA     | iGA      |        | MCMCC |        |       |
|         |          |          |          |                |                     |        | Avg      | SD     | Avg   | SD     |       |
| 10      | 191.00   | 186.20   | 194.34   | 192.11         | 193.67              | 190.93 | 19.34    | 188.16 | 18.81 | 189.23 | 19.39 |
| 20      | 178.80   | 178.80   | 180.05   | 177.73         | 181.94              | 178.54 | 13.99    | 173.40 | 13.02 | 176.72 | 13.93 |
| 30      | 169.60   | 169.40   | 176.18   | 173.25         | 178.43              | 174.49 | 13.26    | 167.25 | 11.46 | 171.56 | 13.27 |
| 40      | 170.40   | 170.40   | 173.61   | 171.12         | 173.02              | 169.47 | 12.04    | 160.38 | 10.12 | 165.69 | 11.81 |
| 50      | 174.00   | 174.00   | 171.07   | 168.5          | 167.79              | 164.55 | 10.72    | 154.94 | 8.01  | 160.78 | 10.19 |
| 75      | 169.80   | 169.80   | 166.75   | 164.47         | 167.39              | 163.35 | 8.20     | 148.73 | 9.35  | 158.77 | 7.91  |
| 100     | 167.60   | 167.60   | 162.28   | 160.84         | 162.87              | 159.78 | 7.42     | 144.30 | 8.17  | 154.59 | 7.36  |
| 150     | 156.80   | 156.60   | 159.67   | 158.59         | 159.84              | 156.46 | 6.38     | 137.20 | 8.40  | 150.54 | 5.78  |
| 200     | 153.00   | 153.00   | 156.66   | 156.16         | 157.96              | 154.39 | 6.76     | 133.93 | 6.67  | 148.98 | 6.04  |
| 250     | 151.00   | 150.40   | 156.1    | 155.59         | 155.91              | 152.80 | 5.15     | 131.10 | 7.45  | 147.01 | 4.93  |
| 500     | 154.20   | 154.20   | 152.11   | 151.79         | 151.43              | 148.74 | 4.99     | 125.18 | 7.02  | 142.51 | 4.09  |

**Table 4** Network lifetime vs. number of sensor nodes (for small scale networks)

| Sensors | GA-DQC   |          | iMA      |                | ub (MCMCC, iGA, MA) |       | Lifetime |       | MCMCC |       |
|---------|----------|----------|----------|----------------|---------------------|-------|----------|-------|-------|-------|
|         | ub (avg) | Lifetime | ub (Avg) | Lifetime (Avg) | MA                  | iGA   | MA       |       | MCMCC |       |
|         |          |          |          |                |                     |       | Avg      | SD    | Avg   | SD    |
| 90      | 27.40    | 27.20    | 24.91    | 24.91          | 25.18               | 4.82  | 25.18    | 4.82  | 24.55 | 4.43  |
| 100     | 27.20    | 27.20    | 28.54    | 28.54          | 28.76               | 6.75  | 28.73    | 6.69  | 27.94 | 6.03  |
| 200     | 59.20    | 59.00    | 57.28    | 57.21          | 57.98               | 11.88 | 57.97    | 11.73 | 55.86 | 9.98  |
| 300     | 79.40    | 79.40    | 87.69    | 87.69          | 88.85               | 14.74 | 88.85    | 14.19 | 85.78 | 13.10 |



**Table 5** Network lifetime vs. number of sensor nodes (for large scale networks)

| Sensors | GA-DQC   |          | iMA      |                | ub (MCMCC, iGA, MA) |      | Lifetime |      | MCMCC  |      |
|---------|----------|----------|----------|----------------|---------------------|------|----------|------|--------|------|
|         | ub (avg) | Lifetime | ub (Avg) | Lifetime (Avg) | MA                  | iGA  | MA       |      | MCMCC  |      |
|         |          |          |          |                |                     |      | Avg      | SD   | Avg    | SD   |
| 90      | 41.00    | 41.00    | 43.88    | 43.73          | 43.57               | 3.17 | 42.94    | 2.68 | 42.11  | 2.47 |
| 100     | 49.00    | 49.00    | 49.04    | 48.89          | 48.28               | 3.15 | 47.63    | 2.80 | 46.87  | 2.61 |
| 200     | 101.00   | 101.00   | 100.22   | 100.02         | 99.57               | 4.35 | 89.32    | 4.81 | 95.18  | 3.98 |
| 300     | 156.40   | 156.40   | 152.11   | 151.79         | 148.74              | 4.99 | 125.18   | 7.02 | 142.51 | 4.09 |

Table 3 show tabulation of results for large networks (300 sensors with sensing range 400) with varied number of targets (varied between 10 and 500). In most of the cases, the average lifetime of GA-DQC is said to reach the average upper bound. The upper bound of GA-DQC algorithm is found appropriate for the coverage of targets while the fitness function evaluates each solution for efficient coverage of each target with respect to  $Q$ -coverage constraints.

Tables 4 and 5 show the results of networks with 10 targets and 500 targets respectively. The sensing range in Table 4 is maintained at 250, while in Table 5, the sensing range is set to 400. The number of sensors is varied between 90 and 300 with constant sensing ranges of 250 (Table 4) and 400 (Table 5). The GA-DQC algorithm finds different directions of sensors that can be activated in different covers for ensuring complete coverage. The initial population, though it is randomly generated, consists only of the sensing directions of sensors that cover at least one of the targets and this gives a population that has better fitness as the unused directions (directions of sensor nodes that do not cover any of the targets) are eliminated even at the initial stage. This results in better performance of the proposed algorithm when compared to existing algorithms.

## 6 Conclusion

This paper discussed the  $Q$ -coverage problem, a generalized case of  $K$ -coverage and single coverage problem (Target coverage) of directional sensor networks with respect to power supply constraints and proposed a genetic algorithm based solution. The sensing ranges of sensor nodes are varied based on the requirements and each sensor node can operate in multiple directions (the number of sensing direction is decided by the sensing angle). Most common method for extending the lifetime of the network is to divide the sensor nodes in a wireless sensor network (sensing directions of different sensor nodes in a directional sensor network) into groups called cover sets and schedule the network such that the cover sets are activated one at a time. It is assured that the coverage constraints of all targets are met by the set of sensing directions in each cover set. The results of GA-DQC for varying  $Q$ -coverage and  $K$ -coverage constraints, various sensing ranges and varying count of sensor nodes and target points are simulated. The results are also compared with those of memetic and genetic algorithms proposed earlier for a complete coverage problem (the  $K$ -coverage value is set to 1) of omnidirectional sensor networks (the sensing angle is set to 360 degrees). The proposed GA-DQC reached the upper bound in most of the simulations and an optimal solution was obtained for both small scale and large scale networks. GA-DQC performs better than other algorithms as the target coverage is ensured from the initial population formation and as a result of the ability of each sensor nodes to be activated in different sensing directions in various covers.

**Funding** Authors received no Grants from any funding agency in the public, commercial, or not-for-profit sectors.

## Declarations

**Conflict of interest** The authors declare that they have no conflicts of interest.

**Data Availability** Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

**Code Availability** The code that is used in this study are available from the corresponding author, upon reasonable request.

## References

1. Arampatzis T., Lygeros J., & Manesis S. (2005). A survey of applications of wireless sensors and wireless sensor networks. *IEEE International Symposium on Mediterrean Conference on Control and Automation Intelligent Control*, 719–724.
2. Waltenegus, D. W., & Poellabauer, C. (2010). *Fundamentals of wireless sensor networks: theory and practice*. London: Wiley.
3. Arivudainambi D., Balaji S., Deepika S. & Swetha S. (2015). Connected coverage in wireless sensor networks using genetic algorithm. *IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions*, 1–6.
4. Arivudainambi, D., & Balaji, S. (2016). Improved memetic algorithm for energy efficient sensor scheduling with adjustable sensing range. *Wireless Personal Communication*, 95(2), 1737–1758.
5. Arivudainambi D., Sreekanth G. & Balaji S. (2016). Energy efficient sensor scheduling for target coverage in wireless sensor network, *Wireless Communications, Networking and Applications, LNEE*, vol. 348.
6. Vijayaraju, P., Sripathy, B., Arivudainambi, D., & Balaji, S. (2017). Hybrid Memetic algorithm with two dimensional discrete haar wavelet transform for optimal sensor placement. *IEEE Sensors Journal*, 17(7), 2267–2278.
7. Sasa Slijepcevic, & Miodrag Potkonjak (2001). Power efficient organization of wireless sensor networks. *IEEE International Conference on Communications*, 472–476.
8. Kumar, S., Lai, T. H., & Arora, A. (2007). Barrier coverage with wireless sensors. *Wireless Networks*, 13, 817–834.
9. Mihaela Cardei, My T. Thai, Ying shu Li. & Weili Wu (2005). Energy-efficient target coverage in wireless sensor networks. *IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 1976–1984.
10. Mihaela Cardei & Ding-Zhu Du. (2005). Improving wireless sensor network lifetime through power aware organization. *Wireless Networks*, 11(3), 333–340.
11. Wu-chi Feng, Ed Kaiser, Wu Chang Feng, & Mikael Le Bailif (2005). Scalable low-power video sensor networking technologies. *ACM Transactions on Multimedia Computing, Communications, and Applications* 1(2), 151–167.
12. Mohammad Rahimi, Rick Bae, Obimdinachi I Iroez, Juan C Garcia, Jay Warrior, Deborah Estrin, & Mani Srivastava (2005). Cyclops: in situ image sensing and interpretation in wireless sensor networks, *Third International Conference on Embedded Networked Sensor Systems*, pp. 192–204.
13. Ai, J., Alhussein, A., & Abouzeid,. (2006). Coverage by directional sensors in randomly deployed wireless sensor networks. *Journal of Combinatorial Optimization*, 11(1), 21–41.
14. YanliCai, W. L., Li, M., & Li, X.-Y. (2009). Energy efficient target-oriented scheduling in directional sensor networks. *IEEE Transactions on Computer*, 58(9), 1259–1274.
15. Wang, J., Niu, C., & Shen, R. (2009). Priority-based target coverage in directional sensor networks using a genetic algorithm. *Computer Mathematics Applications*, 57(11), 1915–1922.
16. Rossi, A., Singh, A., & Sevaux, M. (2013). Lifetime maximization in wireless directional sensor network. *European Journal of Operational Research*, 231(1), 229–241.
17. YanliCai, Wei Lou, Minglu Li, & Xiang-Yang Li (2007). Target-oriented scheduling in directional sensor networks, *ICCC*, pp. 1550–1558.
18. Jian Wang, Changyong Niu & Ruimin Shen (2007). Randomized approach for target coverage scheduling in directional sensor network. *International Conference on Embedded Software and Systems*, pp. 379–390.
19. Singh, A., & Rossi, A. (2013). A genetic algorithm based exact approach for lifetime maximization of directional sensor networks. *Ad Hoc Network*, 11(3), 1006–1021.
20. Gil, J.-M., & Han, Y.-H. (2010). A target coverage scheduling scheme based on genetic algorithms in directional sensor networks. *Sensors*, 11(2), 1888–1906.
21. Yu Gu, Hengchang Liu, & Baohua Zhao (2007). Target coverage with QoS requirements in wireless sensor networks. *IEEE ICIPC*, pp. 35–38.
22. Singh, A., Rossi, A., & Sevaux, M. (2013). Matheuristic approaches for Q-coverage problem versions in wireless sensor networks. *Engineering Optimization*, 45(5), 609–626.

23. Manju Chaudhary, & Arun K. Pujari (2009). Q-coverage problem in wireless sensor networks. *DCN*, pp. 325–330.
24. Li, D., Liu, H., Xianling, Lu., Chen, W., & Hongwei, Du. (2012). Target Q-Coverage problem with bounded service delay in directional sensor networks. *International Journal of Distributed Sensor*. <https://doi.org/10.1155/2012/386093>
25. Singh, A., & Rossi, A. (2015). Group scheduling problems in directional sensor networks. *Engineering Optimization*, 47(12), 1651–1669.
26. Huiqiang Yang, Deying Li, & Hong Chen (2010). Coverage quality based target-oriented scheduling in directional sensor networks. *IEEE ICC*, pp. 1–5.
27. Arivudainambi D., Balaji S., & Rekha D (2014). Improved memetic algorithm for energy efficient target coverage in wireless sensor networks. *IEEE ICNSC*, pp. 261–266.
28. Chih-Chung Lai, Chuan-Kang Ting, & Ren-Song Ko (2007). An effective genetic algorithm to improve wireless sensor network lifetime for large-scale surveillance applications. *IEEE CEC*, pp. 3531–3538.
29. Ting, C.-K., & Liao, C.-C. (2010). A memetic algorithm for extending wireless sensor network lifetime. *Information Sciences*, 180(24), 4818–4833.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**M. Anitha** received her B.E degree in Electrical and Electronics Engineering from Madras University in 2002, M.E in Power Electronics and Drives from CEG Campus, Anna University in 2010 and Ph.D. degree from CEG Campus, Anna University. She is currently working as an assistant professor in Department of Electrical and Electronics Engineering, St. Joseph's College of Engineering, Chennai. Her area of research includes multilevel inverter and grid connected renewable energy systems.



**Dr. P. Somasundaram** was born 1977, Tamil Nadu, India. He got his B.E degree in 1998 from Madras University, M.E degree in Power System Engineering in 2000 from Annamalai University and Ph.D. in Power System Engineering in 2005 from Anna University. His area of interest is Application of Artificial Intelligence Techniques to Power System Optimization Problems, HVDC, FACTS and Power Quality. He is at present Associate Professor in the Department of Electrical and Electronics Engineering, College of Engineering, Anna University, Chennai, Tamil Nadu..



**S. Balaji** is currently working as an Assistant Professor in the School of Advanced Sciences, VIT University, Chennai Campus. He completed his Ph.D. degree from CEG Campus, Anna University in 2019. His research areas cover wireless sensor networks, evolutionary algorithms and wavelets. He published several research papers in the reputed journals particularly in the area of wireless sensor network..



**S. Swetha** received her M.Sc [Integrated] degree in Information Technology from College of Engineering, Guindy, Anna University, Chennai in 2016. She has worked in research papers on wireless sensor networks for achieving better coverage.