



An effective hybrid genetic algorithm and tabu search for maximizing network lifetime using coverage sets scheduling in wireless sensor networks

Nemat allah Mottaki¹ · Hodayun Motameni¹ · Hosein Mohamadi²

Accepted: 6 July 2022 / Published online: 1 September 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Two important issues in Wireless Sensor Networks (WSNs) are coverage and network lifetime. Network operation in an environment can be divided into two phases, first, making coverage sets and then scheduling them. The preliminary reviews have provided a number of solutions for problem-solving in the first phase, but there is no enough solution provided for the second one. Once the number of sensors in the environment increases, the number of coverage sets as well as the number of order sequences (in which the coverage sets can be scheduled) will also increase. The present study aims to detect a near-optimal scheduling for coverage sets to prolong the network lifetime. This problem was previously introduced as MCSS (Maximum Coverage Sets Scheduling) and it was proved to be an NP-hard problem. This paper presents two algorithms, a Genetic Algorithm (GA) and a Hybrid Algorithm (HA) integrating GA and Tabu Search (TS) to solve the MCSS problem. To reveal its efficiency, the algorithms were compared with a recently proposed algorithm in terms of scheduling the coverage sets. The simulations performed in this study showed that the proposed algorithms were more successful in finding near-optimal scheduling coverage sets.

Keywords Directional sensor networks · Genetic algorithm · Tabu search · Coverage set scheduling

✉ Hodayun Motameni
motameni@iausari.ac.ir

¹ Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

² Department of Computer Engineering, Azadshar Branch, Islamic Azad University, Azadshar, Iran

1 Introduction

Wireless sensor network (WSN) is one of the most important technologies employed extensively during the last two decades. WSNs have been applied recently to other applications such as internet of things and home automation [1]. A WSN is composed of a large number of multi functional sensors that are small in size, each of which equipped with a small radio controller, a transmitter, and a receiver. WSNs are applied to battlefield security monitoring, wildlife habitat monitoring, etc. [2]. Sensors are generally divided into two categories according to their application, omnidirectional and directional sensors. In omnidirectional, sensors are able to sense a full 360-degree view of their surroundings and can cover targets simultaneously in all directions. A directional sensor can perform its monitoring task in several directions; however, it can only be active in one direction and at a given angle within a defined time unit. Directional sensors basically involve the ultrasonic, infrared, and video sensors [3].

Coverage is considered a major problem in directional sensor networks (DSNs) and is a criterion for quality of service, which means that how any point or area in a network is monitored by a sensor [4]. Based on sensor networks, coverage is categorized into three groups: target coverage, area coverage, and barrier coverage [5]. In cases where the sensors monitor some targets or points defined in an area, it is called target coverage [6]. While, area coverage refers to the situations where sensors cover an entire area regardless of a definite point [7]. Barrier coverage, on the other hand, creates a barrier using a chain of sensors and minimizes the possibility of intrusion into the environment in which the sensor network is active [8].

Another major problem in sensor networks is to prolong the network lifetime [9]. Energy consumption management in sensors is of great importance because it is almost impossible to recharge or replace the sensors batteries [10]. The lifetime of a DSN is the time period during which the network can operate properly, from the time the network is set up until the time it fails to meet certain coverage requirements [11]. Sensor scheduling is one of the techniques proposed in the literature for energy consumption management [12]. This technique divides the sensors into a number of coverage sets each of which can meet the network coverage requirement. Two types of coverage sets can be formed: disjoint and non-disjoint. In the former, each sensor is a member of only one coverage set that do not share the sensors. On the other hand, the latter contains sensors that can participate in several coverage sets sharing a number of sensors with each other. When a coverage set is implemented within a given time period, the rest switch to inactive mode. When a coverage set is inactive, its associated sensors also switch to inactive mode. The activity time of each sensor can be scheduled so efficiently and practically that the network lifetime could be prolonged as much as possible [6].

The current paper presents a GA-based algorithm and a hybrid GA-TS algorithm to achieve a near optimal scheduling for coverage sets. GA is a search algorithm based on natural selection and natural genetics. In GA, instead of a

point, a population of points is searched. The solution to a problem in the GA is encoded with a chromosome. Several operations such as reproduction, crossover and mutation, the chromosome which is fitter than the others get better chance to reproduce. Tabu Search (TS) is a neighborhood-based local exploration approach, which examines solution space by continuously replacing the recent solution with the best unvisited neighbor solution. As mentioned earlier, the algorithm uses the global search ability of GA as well as the local search ability of TS to achieve the optimal or near-optimal solution. The algorithms proposed in this paper can be applied to both WSNs and DSNs, but here the DSNs is considered.

1.1 Necessity of coverage set scheduling in sensor networks

To provide better understanding of the effectiveness of sensor scheduling technique in prolonging the DSNs lifetime, an example is illustrated in Fig. 1.

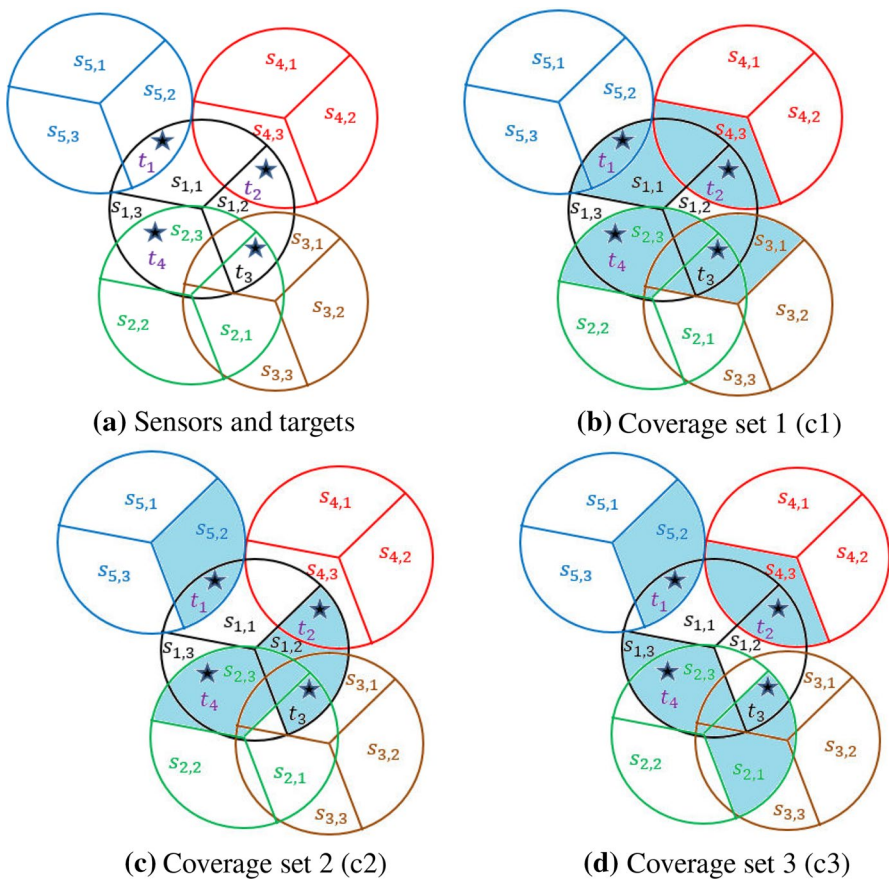


Fig. 1 Directional sensors and targets distributed in environment, and 3 coverage sets that can be created through them

Suppose an environment in which 5 sensors (s_1, s_2, s_3, s_4, s_5) with 4 targets (t_1, t_2, t_3, t_4) are randomly distributed and each sensor's lifetime is equal to h . Each sensor covers a number of targets in this environment (for example, s_{ij} indicates that sensor i covers a target using sector j). When all sensors simultaneously switch to active mode, the maximum lifetime of the network will be equal to the lifetime of one sensor (in this example, it is equal to h).

To achieve the advantages of the sensor scheduling method, it is assumed that the sensors are categorized into three non-disjoint coverage sets (See Fig. 1). The coverage sets $c_1 = (s_{1,1}, s_{4,3}, s_{2,3}, s_{3,1})$, $c_2 = (s_{1,2}, s_{2,3}, s_{5,2})$ and $c_3 = (s_{2,1}, s_{1,3}, s_{4,3}, s_{5,2})$, can be created, each of which is allowed to be active for $0.5 h$. By activating each coverage set, the energy of its sensors will decrease. Coverage sets also have some sensors in common. As illustrated by this example, the network lifetime is increased to $1.5 h$ by activating the coverage sets constantly, assuming that the sensor's energy consumption is insignificant in inactive mode.

As mentioned before, inside an environment, the activity of network is combined of two phases: creation of covering sets and their scheduling [11]. The initial examination have provisioned numerous solutions for the phase 1, however, not conveniently enough results for the phase 2. It must be noted that as the sensors' number increases, so do the quantities of the covering sets and order sequences, wherein the volume of the scheduled covering sets will be enlarged, too. The number of strategies based on which the coverage sets can be scheduled successively is characterized as a permutation. The permutation of a set is to arrange its members into a sequence or linear order, or just to rearrange them if the members are already in a sequence. Consequently, the number of permutations of a set with n elements is $n!$ [13]. For example, if there is a network with three coverage sets (a, b, c); then, six different sequences, namely (a, b, c) , (a, c, b) , (b, a, c) , (b, c, a) , (c, a, b) , and (c, b, a) can be regarded for scheduling them. The problem of optimizing the coverage sets scheduling, called Maximum Coverage Sets Scheduling (MCSS), was first raised in [11], which aimed to prolong the network lifetime.

1.2 Our contribution and organization of the paper

In a sensor network with a large number of sensors, more coverage sets could be built. How to schedule coverage sets and the order of execution can have a great impact on the network lifetime. In this paper, the main objective is to find near-optimal scheduling coverage sets. The present study proposes two algorithms to find the scheduling strategy of coverage sets. In the first, a GA-based algorithm is proposed to find coverage set scheduling. In the second algorithm, a hybrid GA-TS algorithm is developed. In this algorithm, the high global search ability of GA and the local search ability of TS are used to provide a high search capacity. Several simulations are performed on the algorithms, and the results are compared with those of an algorithm presented previously in the literature.

The remainder of this paper is organized as follows: Sect. 2 briefly reviews the studies conducted on making coverage sets. Network model and problem definition are given in the Sect. 3. Section presents the algorithms. Section 5 provides the

simulation results of the algorithm. Finally, Sect. 6 presents the research conclusion and further work.

2 Related work

A major technique for energy consumption management is to schedule the sensor activity into two states: sleep and active. This technique is intended to maximize coverage and the network lifetime. The problem is defined as the maximum coverage with minimum sensors and it is an NP-Complete problem [14].

The literature consists of some methods proposed to solve the scheduling sensors problem in directional networks. In [15], the directional cover and transmission in DSNs was introduced, which was intended to prolong the network lifetime, transmit the information, and solve the targets overlapping problem.

In [16, 17], many attempts were made to solve the coverage problem in directional sensors. Two techniques, i.e., sensors scheduling and adjusting sensing range, were used to prolong the network lifetime. In [16], for the first time, the adjustable sensing ranges in DSNs were investigated. In this respect, two greedy algorithms were proposed to solve the problem. In [17], the problem of Maximum Network Lifetime with Adjustable Ranges (MNLAR) was addressed and a target-oriented GA-based algorithm was presented to solve it. In [14], the coverage in the visual sensor networks was examined, in which the targets have different predetermined coverage requirement. In [3], an energy-aware method was proposed to cover all the targets in the network and also to extend the network lifetime. The authors used the multi-objective evolutionary algorithm method, recognizing that the scheduling of the sensors to solve the coverage problem is inherently a multi-objective problem. To solve a multi-objective problem, two parameters, i.e., the minimum number of sensors and the maximum target coverage, are defined. In [5], a method was proposed to solve the coverage problem in DSNs. The authors used GA and repair operator to select a set of sensors and solve the priority-based target coverage problem. In this method, each target initially has a different predetermined coverage requirement.

In [18] the cuckoo search algorithm has been developed to find the maximum number of coverage sensors. In [19] a cuckoo search algorithm is developed to identify the sensor coverage set by optimizing the sensing range to reduce the redundancy sensors in the network. The algorithm creates non-disjoint cover sets with variable sensing range so that each cover set provide k-coverage. In [20], the problem of prolonging network lifetime was investigated as a bi-objective optimization problem. The authors consider the first objective to find the maximum coverage sets and the second objective to solve the problem of energy loss. Authors in [21] propose a novel and energy efficient target coverage algorithm to identify disjoint and non-disjoint coverage sets. the developed algorithm identifies coverage sets to cover the targets and gives the optimal minimum energy paths from the cover set to the sink and the sink to the sensor node.

In [11] the authors proved that coverage set scheduling is an NP-hard problem, and also they formulated the MCSS problem by ILP. In addition, the Greedy-MCSS was proposed to solve the coverage sets scheduling problem and an approximation

algorithm, called MCSSA, was proposed based on this method. After simulation, the superiority of MCSSA method over Greedy-MCSS was proved, and the MCSSA algorithm was ultimately implemented under different conditions. Although the methods proposed in [11] will result in achieving some solutions, such solutions are inherently problematic. A lot of computational complexities can be found in the Integer Linear Programming (ILP) method used for solving large-scale problems, and it is not an efficient solution to such problems. Due to the fact that the greedy-based solution methods are used in solving many optimization problems, they may be caught into the local minimum depending on the problem conditions, and consequently, they will not achieve the global optima [22, 23].

3 Network model and problem definition

This paper follows the scenario presented below. The directional sensors and targets are distributed randomly and uniformly at given spots in an Euclidean environment. Targets and sensors are aware of their locations. The energy level of each sensor is equal to b_i , and a negligible amount of energy is consumed to change the sector. Each sensor is in one of the two modes: active or sleep. It is supposed that a group of coverage sets (C) is made for the scheduling sensors, $C=(c_1, c_2, \dots, c_k)$. In this group, each c_k is a non-disjoint coverage set and it can meet the network requirements during a given period by itself. If the period in which the c_k coverage set can be activated in a sequence regarding the energy of its sensors is equal to t_k , and $T=(t_1, t_2, \dots, t_k)$, a coverage set scheduling strategy should be found in a way to satisfy Eq. (1):

$$\text{Maximize}(T_{\text{networklifetime}} = \sum_{i=1}^k t_i) \tag{1}$$

Table 1 gives the parameters used in this paper.

Problem : How to find the optimal or near-optimal schedule of coverage sets in a way to maximize the network lifetime by activating it.

Table 1 Notations

Notations	Definition
n	Number of sensors
m	Number of targets
C	Set of coverage sets
s_i	The i th sensor $1 \leq i \leq n$
S	Set of sensors $\{s_1, s_2, \dots, s_n\}$
T	Set of targets $\{t_1, t_2, \dots, t_m\}$
$d_{i,j}$	j th direction of i th sensor
c_k	k th coverage set
b_i	Energy level of i th sensor

In the following, an example is presented to illustrate the problem: There are seven sensors $S = (s_1, s_2, \dots, s_7)$ with energy levels of $b = (4, 3, 5, 5, 5, 2, 3)$ in an environment. It is assumed that three non-disjoint coverage sets are created as $c_1 = (s_1, s_2, s_5), c_2 = (s_3, s_4, s_5), c_3 = (s_6, s_7, s_3, s_1)$. According to the definition of permutation, it is possible to obtain six different implementation sequences through these three coverage sets. Moreover, the assumption is that the activity time of each sensor is equal to the energy level of the sensor. Accordingly, Table 2 presents the total network lifetime while scheduling the six possible strategies.

In Table 2, each of the two coverage schedules of c_3, c_2, c_1 and c_3, c_1, c_2 can be accepted as the final answer. For a better understanding, consider schedule c_3, c_2, c_1 . The first coverage set consists of sensors s_6, s_7, s_3 , and s_1 . When this coverage set is used to cover the targets in the network, it can cover the targets up to 2 (energy level of s_6). After this, because the energy of sensor s_6 is exhausted, this coverage set is no longer able to cover all the targets, the energy of all the sensors in this coverage set is reduced and the remaining energy of the sensors in this coverage set is updated. the new energy levels of the sensors in the coverage set becomes (0, 1, 3, 2). The next coverage set to run in this schedule is c_2 . In this coverage set, there are sensors s_3, s_3 , and s_5 . This coverage set runs as long as the energy level of s_3 , and when the sensor runs out of power, the coverage set no longer covers all the targets in the network. After destroying sensor s_3 , the new energy levels of the sensors is updated, and the new energy levels of the sensors in coverage set becomes (0, 2, 2). Finally, coverage set c_1 is running with sensors s_1, s_2 , and s_5 . The new energy level of the sensors becomes (0, 1, 0). The sum of the lifetime that all 3 coverage sets in the network could be executed is 7. The way of calculating the scheduling lifetime of $(c_3c_2c_1)$ is illustrated in Fig. 2.

4 Proposed algorithms

The section starts with a general review of the GA; after that, it introduces two scheduling solutions proposed in this study to solve the coverage scheduling problem in DSNs.

Table 2 Cover sets scheduling strategy

Coverage sets scheduling	Total network lifetime
$c_1 c_2 c_3$	6
$c_1 c_3 c_2$	6
$c_2 c_1 c_3$	5
$c_2 c_3 c_1$	5
$c_3 c_1 c_2$	7
$c_3 c_2 c_1$	7

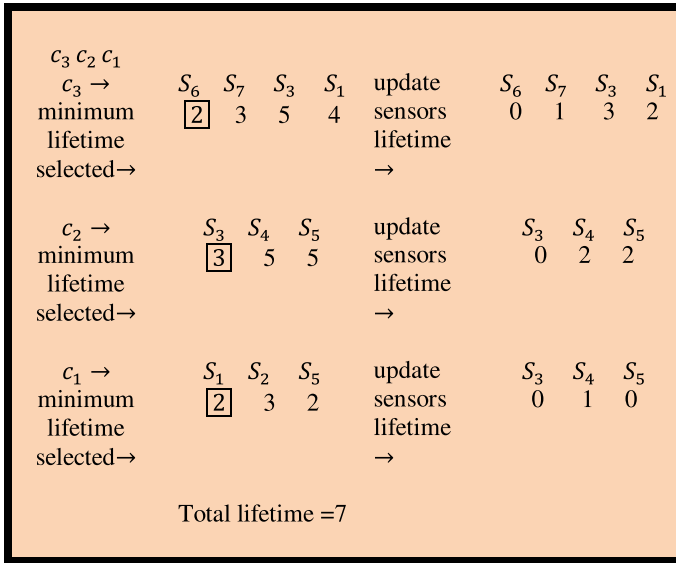


Fig. 2 Method of calculating the lifetime of the coverage sets

4.1 Overview of the genetic algorithm

GA is able to solve optimization problems through modelling the natural evolution principles. At the initial step, the algorithm generates a population of possible solutions each of which is denoted by a set of genes (also known as chromosome or individuals). A fitness function is defined for each certain problem, which aims to examine each chromosome’s performance. A chromosome fitness is dependent on how well the chromosome performs the task defined in the process of solving the problem of interest. Following the generation of the initial population, the algorithm starts the execution of three key operators, i.e., selection, crossover, and mutation. Through the selection operator, a set of possible solutions are selected by the algorithm from among the initial population. Afterwards, two selected chromosomes, which are known as parents, are applied to the creation of two child chromosomes through the crossover operator. This operator exchanges the genetic information between the parent chromosomes. Then, it is the time for the child chromosomes to execute the mutation operator, which is expected to result in the generation of a better solution. After the completion of his operation, the fitness function is implemented again in order to assess the child chromosomes. To do this, the chromosome’s values are compared to values of all the chromosomes belonging to previous generation. Finally, from among the initial population and the population produced by the crossover and mutation operators, the individuals that hold greater fitness values are selected to be transferred to the next generation [24].

4.2 GA-base algorithm

In this section, the details of GA-based algorithm is explained.

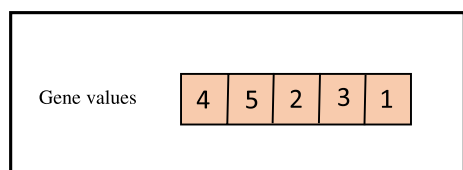
Representation and the initial population The way of modeling the chromosome to solve the problem is considered as a major part of the problem solving process in GA. The present study uses an order representation to model a chromosome as a solution to the problem space. Each gene on the chromosome represents a coverage set, and each chromosome in this model is an indicative of a scheduling of coverage sets. Each chromosome is a solution to a problem. The assumption is that there are n coverage sets, according to the permutation definition, there will be $n!$ different orders. The initial population in this problem is a subset of all possible coverage sets sequence orders ($n!$). For example, if 5 coverage sets with labels 1–5 are considered, one chromosome of this model is shown in Fig. 3.

Fitness function: The fitness value obtained by the fitness function is applied to evaluating the quality of a chromosome. Each sensor s_i in a coverage set has an energy level denoted by b_i . Due to the fact that if even one sensor in coverage set runs out of power, it cannot meet the network requirements, the lifetime of a coverage set is measured based on the activity time of the sensor with the least energy in the coverage set. In this study, the value of the objective function is equal to the sum of the lifetime of the coverage sets that can be implemented spatially on a chromosome. It is worth mentioning that after activating each coverage set, a number of sensors are destroyed due to the discharge of batteries in the network, and the energy levels of a number of sensors change as well. Once a coverage set is activated in a chromosome, the lifetime of the remaining coverage sets of the chromosome changes with a new level of sensor energy; thus, it must be recalculated.

Selection: This operator is used to detect two parents and generate the offsprings. This study uses the binary tournament selection method, based on which two chromosomes are randomly and uniformly selected from among the current population. The most appropriate chromosome is selected as a parent with regard to the fitness value. The same process is repeated to select the second parent.

Crossover: Crossover is the main operator of GA and, similar to its counterpart in nature, it generates new members whose genes are formed from their parents. Depending on the problem nature, there are various methods proposed for crossover in GA. This study uses the order-based chromosomes; it is not suitable to use classical crossover operators such as one-point, two-point, and uniform crossovers. For order-based chromosomes, there are some crossover

Fig. 3 Method of calculating the lifetime of the coverage sets



operators as follow: Partially-Mapped Crossover Operator(PMX), Order Crossover Operator(OX), Cycle Crossover Operator(CX) [25]. This research uses the CX2 method recently presented in [25]; its superiority over the methods PMX, OX and CX has been proven.

Mutation: It is a random process through which a gene content is replaced by another to produce a new chromosome. In this study, the contents of two genes in a chromosome are randomly selected; then, their contents are replaced to perform the mutation operation.

Stopping criterion: The stopping criterion of the proposed algorithm is met when it reaches a predetermined number of maximum iterations. In this case, an appropriate solution (a sequential order of coverage sets) is returned as the output of the algorithm. The proposed GA algorithm is described in algorithm 1.

Algorithm 1 : A GA-based algorithm for coverage set scheduling

```

01. Input: Non-disjoint coverage sets in sensor network
02. Output: Near optimal coverage sets scheduling
03.  $SOL \leftarrow \emptyset$ 
04. Initialize population ( $pop$ )
05. Fitness ( $pop$ )
06.  $k \leftarrow 0$ 
07. While  $k \leq MAX$  do
08.    $p_s = Select(pop)$ 
09.    $p_c = Crossover(p_s)$ 
10.    $p_m = Mutation(p_c)$ 
11.    $pop = Survival(p_c, p_m, pop)$ 
12.   Fitness ( $pop$ )
13.    $k++$ 
14. End While
15.  $SOL \leftarrow BEST(pop)$ 
16. return  $SOL$ 

```

4.3 Hybrid GA-TS algorithm

The second algorithm presented in this paper is hybrid algorithm (HA), combining the GA and TS. Pure evolutionary algorithms are not suitable for precise search in complex hybrid spaces, and hybridization along with other techniques improve the efficiency of the search process. In general, GA detects the optimal region and then a local optimizer is performed to find the optimal solution. The combination of an evolutionary algorithm with a local search approach is referred to as HA [26]. To achieve the advantages of this approach, we develop a HA-based algorithm to solve the problem. The algorithm provides higher capacity in search by combining GA and TS, and it can effectively balance the intensification and diversification [27].

Tabu Search (TS) is an optimization algorithm first introduced by Glover in 1986 [28]. TS is based on a fundamental idea: the algorithm makes a list of movements or Tabu points, called tabu list, to prevent from doing those movements in the subsequent searches. This way, the algorithm desires to go beyond the local optimization and moves toward the global optimization. TS consists of two major

elements: tabu list and aspiration criterion. The tabu list keeps a few recent movements and avoids selecting those movements as many times as possible. The time period during which the movements are on the tabu list is determined by a parameter called tabu tenure. If a movement in the tabu list results in a better solution, it is selected based on a criterion called aspiration criterion, even though it is on the tabu list. After making a new selection and including in the tabu list, a number of movements that were previously on the tabu list are removed from the list [28].

In proposed algorithm to improve the quality of chromosomes, a new function is included in GA, which is applicable to all chromosomes in the current population. The TS optimization for all chromosomes is computationally expensive; therefore, it is not possible to apply it extensively. The method effectively regulates the GA convergence since TS usually improves the chromosomes in an effective way. Once the operation is done, if the stop criterion is not met, from among the initial population, the chromosomes that are more suitable are selected as the next offspring, and the population is generated through the operations of crossover, mutation and improvement. As evaluation progresses, GA seeks to drive the search toward the desired global value. The flowchart of proposed algorithm is described in Fig. 4. The steps of performing the TS algorithm are as follow:

1. Generating an initial solution, evaluating it, and considering it as the best solution.
2. Making a list of authorized operations (based on the adjacent generating methods).
3. Performing authorized and non-tabu operations (all or part of the operations) and obtaining solutions based on the objective function.

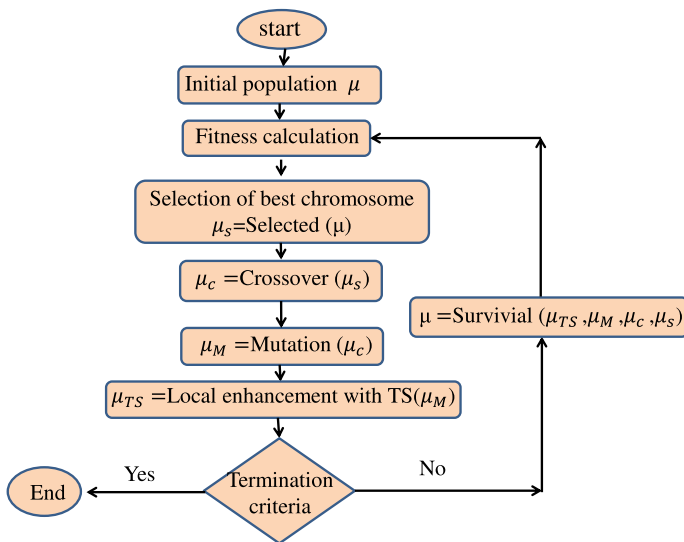


Fig. 4 Flowchart of the proposed hybrid GA-TS algorithm

4. Selecting the best solution from among those obtained in the previous step (if there is an operation in the tabu list that has obtained the best solution so far, the action will be selected by ignoring the tabu).
5. Updating the tabu list (adding the selected operations to the tabu list and removing one or more operations from the tabu list), Updating the best solution found.
6. If the stopping criterion is not met, returning to step 4.

5 Simulation results

To have an highly accurate evaluation, the effects of both the network parameters and the algorithm parameters over the network lifetime were examined. Each experiment was developed to examine the effect of a certain parameter on the network lifetime. The obtained results were compared with those of the MCSSA algorithm presented in [11]. In experiments, N sensors and T targets were randomly distributed within a $1000m \times 1000m$ square area. The sensing radius of all sensors was equal to R . Each sensor had an energy level equal to b_i , which disappeared when the power is exhausted. A number of C coverage sets were randomly constructed to perform the experiments. The activity time of each coverage set depends on the energy of the sensors available in the coverage set. Given that a coverage set should meet the network requirement by itself, the network targets are not completely covered if even one sensor is lost in the coverage set. In such conditions, the coverage set can no longer be activated and its lifetime will expire. The proposed algorithm measures the network lifetime by scheduling coverage sets and calculates their activating time. In this paper, the result of each experiment is an average of 20 runs. The simulation parameters are listed in Table 3.

The experiments were divided into two general categories. In the first category (Experiments 1–4), the impacts of network parameters on the proposed algorithms and MCSSA were investigated to evaluate their efficiency in prolonging the network lifetime. On the other hand, the second category (Experiments 5–8) examined the impacts of network parameters on the proposed algorithm, and coverage set scheduling was evaluated under different conditions. The network parameters used in all

Table 3 Simulation table

Simulation parameters	
Monitored region	$1000_m \times 1000_m$
Number of directional sensors (S)	200–1200
Number of targets (T)	5–30
Deployment	Random
Sensing radius (R)	100–300
Comm. radius	$2 \times$ Sensing radius
Working directions	4
Active time slots of sensors	[0, 80]
Number of coverage sets (M)	10–80, 100–800

experiments were the sensor activity time, number of coverage sets, number of targets, number of sensors, and sensors' sensing range.

Experiment 1. This experiment was aimed at proving the hypothesis that the network lifetime will extend by increasing the active time slots of the sensors. As the coverage sets are made up of sensors, it is expected that increasing the activity time of the sensors will lead to increasing the coverage sets lifetime and the targets are covered longer by a coverage set. This experiment determined the impact of increasing the activity time of the sensors on the three algorithms. To perform the experiment, the sensors activity time during different performances was determined from [0, 10] to [50, 60] (note that [0, 10] is a random value within the range of 1–10), and the network parameters were set as $T = 20$, $N = 100$, $M = 100$, and $R = 200$. As indicated in Fig. 5, the proposed algorithms has a longer lifetime than the MCSSA algorithm for the same amount of activity time. The hybrid of GA and TS algorithms has better results than GA.

Experiment 2. The second experiment was designed to prove the hypothesis that the network lifetime will prolong by increasing the number of coverage sets. In this regard, the number of coverage sets was uniformly increased from 10 to 60 with incremental step of 10, the values of $T = 10$, $N = 300$, and $R = 200$ were set, and the activity time of the sensors was randomly selected within the range of 10–30. The results indicated that the network lifetime was not prolonged significantly by increasing the number of coverage sets. It can be explained by the fact that coverage sets have many sensors in common and consuming the sensors' energy will decrease the coverage sets lifetime, which also results in limiting the network lifetime. As it can be seen in Fig. 6, the proposed algorithms performed

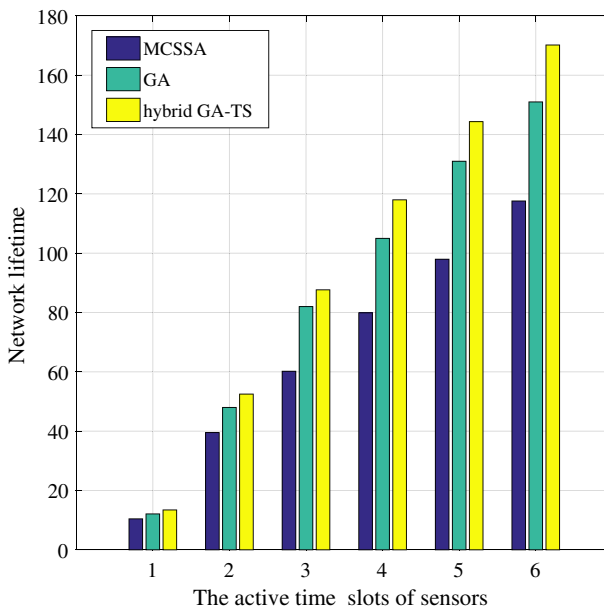


Fig. 5 Effect of the active time slots on the network lifetime

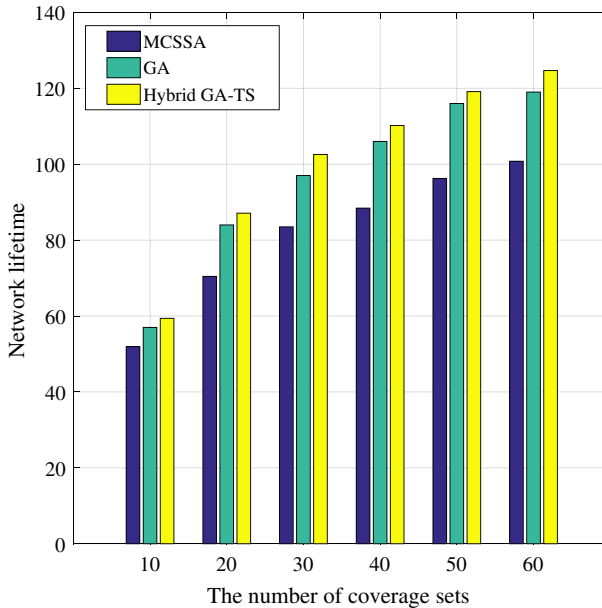


Fig. 6 Effect of the number of coverage sets on the network lifetime

better than the MCSSA algorithm in this regard. The results of the two proposed algorithms showed that the hybrid of the TS and GA algorithms contributed to finding better chromosomes and prolonging the network lifetime.

Experiment 3. This experiment was carried out to compare the impacts of increasing the number of targets on three algorithms. To do this, the number of targets was increased from 5 to 30 units with incremental step of 5. To perform the experiment, the values of $N = 100$, $M = 100$, and $R = 200$ were determined and the energy value of each sensor was set to 10. More sensors should be included in a coverage set in order to cover new targets. Once each coverage set was implemented, more sensors lost their energy and the network lifetime slowly decreased as the number of targets increased. As reported by the diagrams in Fig. 7, the proposed algorithms performed better than the MCSSA algorithm in terms of prolonging the network lifetime. The reason can be stated as follows: TS is able to overcome local minima by guiding the optimisation to new areas, and also it can reach the global optimum.

Experiment 4. This experiment was aimed at determining the impact of increasing the number of sensors on the network lifetime. In this regard, the parameters of $T = 20$, $M = 100$, and $R = 200$ were determined, and the active time of each sensor was set to 10. Therefore, the number of sensors increased from 200 to 1200 with incremental step of 200. The network lifetime prolonged by increasing the number of sensors. Accordingly, the result is acceptable because each target was covered by more sensors and the number of shared sensors among different coverage sets decreased. The results of the implementation

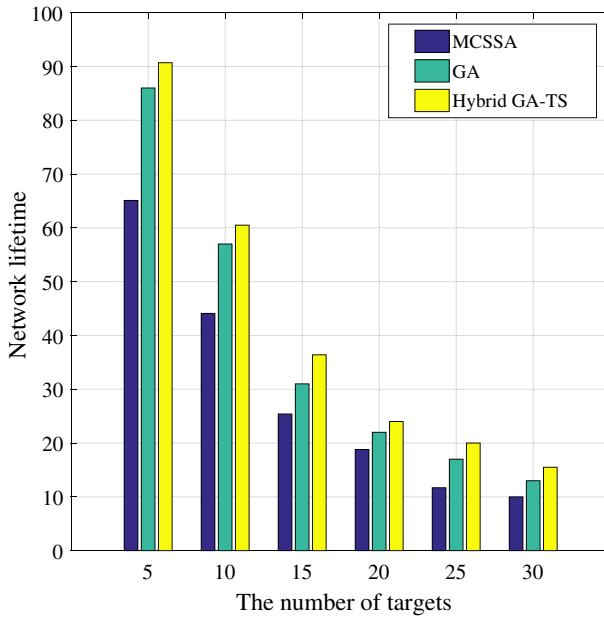


Fig. 7 Effect of the number of targets on the network lifetime

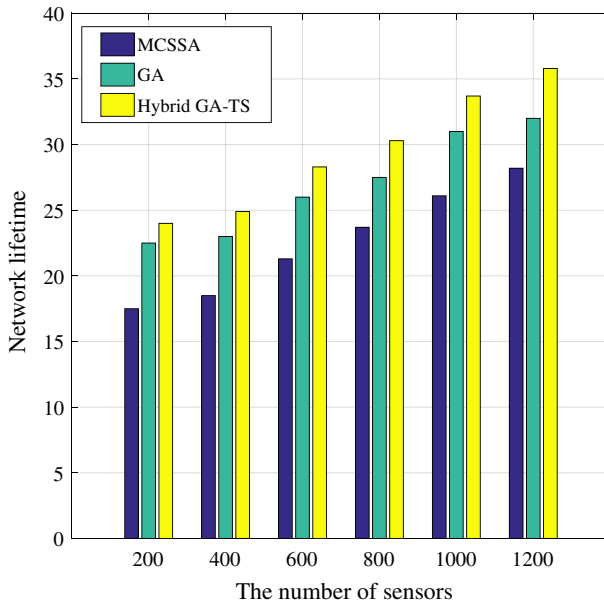


Fig. 8 Effect of the number of sensors on the network lifetime

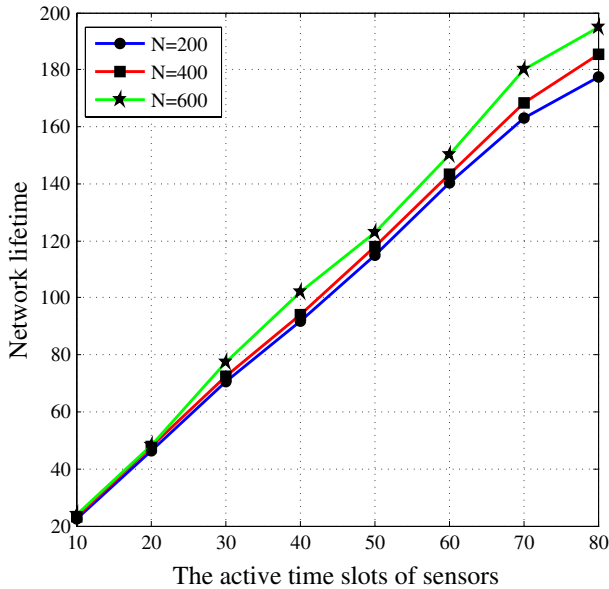


Fig. 9 Effect of different values of active time slots and different values of sensors on the network lifetime

of the three algorithms are shown in Fig. 8. The two proposed algorithms showed better results than those of the MCSSA algorithm.

Experiment 5. This is the first experiment of the second group. As mentioned earlier, experiments 5–8 examine the impacts of the network parameters on the proposed scheduling algorithm. The fifth experiment was aimed at investigating how the network lifetime in the proposed algorithm will change with different values of sensor activity time. Network parameters were determined as $T = 20$, $M = 100$, and $R = 200$, and the numbers of sensors were set into three values of $N = 200$, 400, and 600 in three different implementations, respectively, as shown in Fig. 9. The active time slots of the sensors in all three runs vary from 10 to 80 with incremental step of 10. As it was expected, the results indicated that as the sensors' activity time increased, the coverage sets' lifetime and consequently the network lifetime prolonged. Moreover, the network lifetime prolonged by increasing the number of sensors for fixed points of active time slots. When the active time slots of the sensors remain unchanged and the number of sensors increases, the number of common sensors among different coverage sets decreases, and they can perform their tasks longer, hence increasing the network lifetime.

Experiment 6. This experiment was also designed to investigate the impact of changing the active time slots of the sensors on the network lifetime. However, the difference is that the number of coverage sets was set to $M = 100$, 200, and 300 in three separate performances, and the values of $T = 20$ and $N = 1000$ and the active time slots of the sensors were changed from $[0, 10]$ to $[70, 80]$. For each distinct implementation, it was observed that the network lifetime extended by increasing

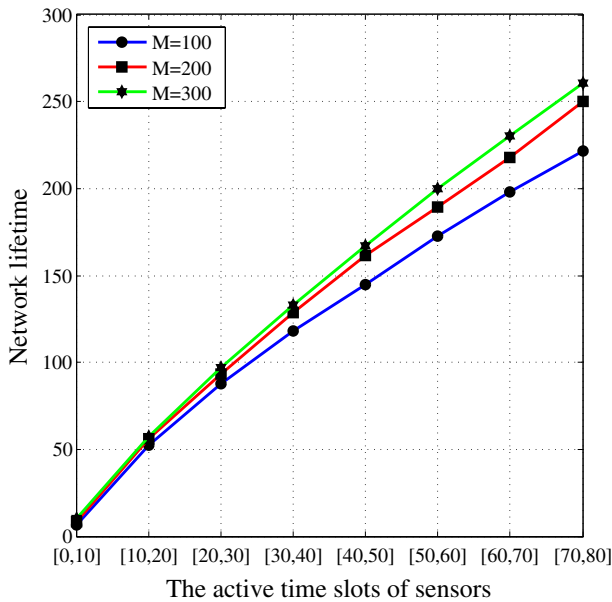


Fig. 10 Effect of different values of active time slots and different values of targets on the network lifetime

the active time slots of the sensors. After comparing all three runs, the diagrams indicated that the network lifetime extended slightly as the number of coverage sets increased at the fixed points during the sensors' activity time. Accordingly, it can be said that as the number of coverage sets increases, the number of common among the coverage sets increases, and the lifetime of the coverage sets will run out by discharging the sensors energy, as shown in Fig. 10.

Experiment 7. This experiment was performed to evaluate the impact of the number of coverage sets under different conditions of the proposed algorithm. In this experiment, we set the active time slots of all sensors equal to 10 (a fixed value) and the values of $N = 100$ and $R = 200$; then, the number of coverage sets was increased from 10 to 80 with incremental step of 10. This experiment was performed in three runs with values of $T = 10, 20$, and 30 . As indicated by the diagrams in Fig. 11, as the number of coverage sets increased, the lifetime of the network increased. Based on the diagram, it can be concluded that the number of sensors shared among the coverage sets will increase and therefore the network lifetime will be limited by fixing the number of sensors and at the same time increasing the number of coverage sets.

Experiment 8. This experiment was aimed at evaluating the effect of the number of coverage sets under different conditions (as in Experiment 7) of the proposed algorithm. The difference is that the values of sensing radius was set to $R = 100, 200$, and 300 in three different runs. The other parameters were set as follow: the active time slots of the sensors were randomly selected within the range of $[0, 10]$ and the values of $N = 1000$ and $T = 20$. According to these three runs, it is obvious

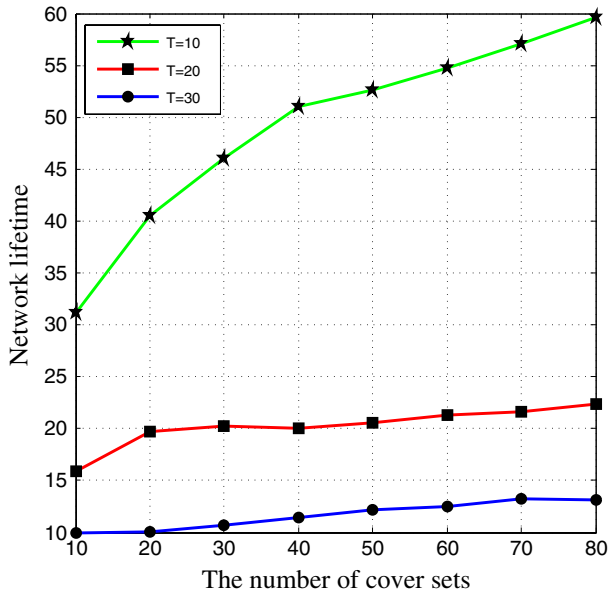


Fig. 11 Effect of the number of coverage sets and different values of T on the network lifetime

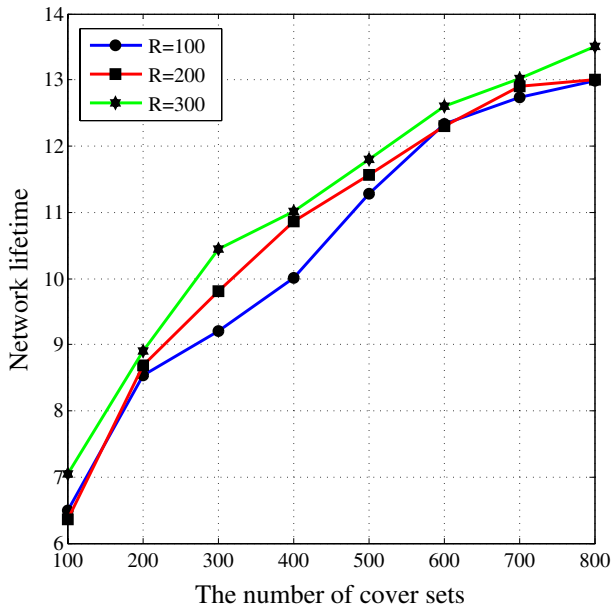


Fig. 12 Effect of the number of coverage sets and different values of R on the network lifetime

that for a fixed number of coverage sets, the longer the sensing radius, the longer the network lifetime will be. The reason is that, in such condition, more targets are covered by each sensor and the number of shared sensors decreases as the sensing radius increases. Consequently, the sensor energy is not discharged quickly; thus, the coverage sets can run longer, hence increasing the network lifetime (see Fig. 12).

5.1 Discussion on simulation results

Greedy-based algorithms can not schedule coverage sets in a way to maximize the network lifetime in real time. Remember that the initial candidates closeness to the optimal solution greatly affects the greedy-based algorithms performance, and the scheme may lead to a local minimum because of its heuristic search. Many scholars have applied GA to the solution of optimization problems, especially in association with scheduling algorithms.

When using GA, different outcomes may be achieved in independent operations, and there is not any guarantee to obtain an optimal result. Acceptable results could be achieved in cases where the “global” search space is tested globally. The use of evolutionary algorithms and high iterations in the global search space would result in higher computational cost in comparison with the greedy algorithm. When dealing with optimization problems with large search space, to achieve a global (or a near global) optimization, a high computational cost is predictable, which is worth it. GA is often capable of identifying the promising areas for global optima within a certain search space; however, in some cases, it may face difficulty in locating the exact minimum of these optima [10]. Particularly, for scheduling coverage set, the search space is extremely large; for that reason, designs found by GA could still have room for further improvement. The current study uses a novel process based on TS to refine the designs. GA is an algorithm inspired by the evolution process and works on a group of solutions at each unit of time, whereas TS works based on concepts from artificial intelligence and operates on only a single solution at each unit of time [16]. TS applies basic, problem-specific operators to the exploration of a search space and memory, called the tabu list, with the aim of keeping track of the parts already visited. TS guides the optimization to new areas; this way, it could overcome local minima, and also it could achieve the global optimum.

6 Conclusion and further work

The target coverage and prolonged network lifetime are regarded as two major problems in DSNs. To solve the coverage problem by using the sensor scheduling technique, the network sensors can be divided into a number of coverage sets. Making use of the GA and TS algorithms, this article presented a hybrid GA-TS algorithm to achieve the optimal or near-optimal coverage set scheduling. With the aim of proving the superiority of the proposed algorithms, its results were compared with those of one of the recently presented algorithm (MCSSA). The results indicated that the proposed algorithms had a better performance than the MCSSA algorithm and it

successfully increased the network lifetime by finding the optimal scheduling coverage sets in the network. Coverage management in sensor networks is an important issue and more research is needed in this area. Existing studies have shown that multi objective evolutionary algorithms (MOEAs) leads to better coverage management in sensor networks, conservation of sensor energy, and thus extended the network lifetime. A part of our future work is to use MOEAs for coverage set scheduling in sensor networks and will study its efficiency in regard to this problem.

Data Availability The following information was supplied regarding data availability: Data will be made available on reasonable request.

Declarations

Conflict of interest All authors declare that they have no conflict of interest that are relevant to the content of this article.

References

1. Gulati K et al (2022) A review paper on wireless sensor network techniques in internet of things (IoT). *Mater Today Proc* 51:161–165. <https://doi.org/10.1016/j.matpr.2021.05.067>
2. Mohammed SB et. al (2020) Wireless sensor network design methodologies: a Survey. In: *Journal of Sensors*. 2020, <https://doi.org/10.1155/2020/9592836>
3. Mottaki NA et al (2021) Multi-objective optimization for coverage aware sensor node scheduling in directional sensor networks. In *J Appl Dyn Syst Control* 4(1):43–52
4. Sangwan A, Singh RP (2015) Survey on coverage problems in wireless sensor networks. *Wirel Pers Commun* 80(4):1475–1500
5. Ajam L, Nodehi A, Mohamadi H (2021) A Genetic-based algorithm to solve priority-based target coverage problem in directional sensor networks. In: *Journal of Applied Dynamic Systems and Control*, 4(1):89-96
6. Mottaki NA, Motameni H, Mohamadi H (2022) A genetic algorithm-based approach for solving the target Q-coverage problem in over and under provisioned directional sensor networks. *Phys Commun* 54:101719. <https://doi.org/10.1016/j.phycom.2022.101719>
7. Hanh NT et al (2019) An efficient genetic algorithm for maximizing area coverage in wireless sensor networks. *Inf Sci* 488:58–75
8. Tao D, Wu T (2015) A survey on barrier coverage problem in directional sensor networks. In: *IEEE Sensors Journal*. 15(2):876-885
9. Singh j, Kaur R, Singh D (2020) A survey and taxonomy on energy management schemes in wireless sensor networks. In: *Journal of Systems Architecture* 111:101782
10. Boukerche A, Sun P (2018) Connectivity and coverage based protocols for wireless sensor networks. *Ad Hoc Netw* 80:54–69. <https://doi.org/10.1016/j.adhoc.2018.07.003>
11. Luo C et al (2020) Maximizing network lifetime using coverage sets scheduling in wireless sensor networks. In: *Ad Hoc Networks*. 98:102037
12. Chowdhury SM, Hossain A (2020) Different energy saving schemes in wireless sensor networks: a survey. *Wirel Pers Commun* 114:2043–2062. <https://doi.org/10.1007/s11277-020-07461-5>
13. SPermutations (2022) ByDarrell whitley book evolutionary computation 1 Edition1st Edition first published2000 ImprintCRC Press Pages12 eBook ISBN9781315274638, <https://doi.org/10.1016/j.jksuci.2019.05.006>
14. Zishan AA et al (2018) Maximizing heterogeneous coverage in over and under provisioned visual sensor networks. *J Netw Comput Appl* 124:44–62
15. Kim Y et al (2013) Lifetime maximization considering target coverage and connectivity in directional image/video sensor networks. In: *The Journal of Supercomputing*. 65(1): 365-382

16. Mohamadi H, Salleh S, Razali MN (2014) Heuristic methods to maximize network lifetime in directional sensor networks with adjustable sensing ranges. In: *Journal of Network and Computer Applications*. 26-35
17. Alibeiki A, Motameni H, Mohamadi H (2019) A new genetic-based approach for maximizing network lifetime in directional sensor networks with adjustable sensing ranges. In: *Pervasive and Mobile Computing*. 52:1-12
18. Balaji S et al (2020) Energy efficient target coverage for a wireless sensor network. *Measurement* 165:108167. <https://doi.org/10.1016/j.measurement.2020.108167>
19. Arivudainambi D et al (2021) Cuckoo search algorithm for target coverage and sensor scheduling with adjustable sensing range in wireless sensor network. *J Discret Math Sci Cryptogr*. <https://doi.org/10.1080/09720529.2020.1753301>
20. El-Sherif M et al (2018) Lifetime maximisation of disjoint wireless sensor networks using multiobjective genetic algorithm. *IET Wirel Sens Syst* 8(5):200–207. <https://doi.org/10.1049/iet-wss.2017.0069>
21. Mohamadi H, Salleh S, Ismail AS (2014) A learning automata-based solution to the priority-based target coverage problem in directional sensor networks. *Wirel Pers Commun* 79(3):2323–2338
22. Gil JM, Han YH (2011) A target coverage scheduling scheme based on genetic algorithms in directional sensor networks. In: *Sensors*. 1888-1906
23. Katti A (2019) Target coverage in random wireless sensor networks using cover sets. *J King Saud Univ Comput Inf Sci* 34:734–746
24. Katoch S et al (2021) A review on genetic algorithm: past, present, and future. *Multimed Tools Appl* 80:8091–8126
25. Hussain A et al (2017) Genetic algorithm for traveling salesman problem with modified cycle crossover operator. *Comput Intell Neurosci* 2017:1–7
26. Garai G, Chaudhuri BB (2013) A novel hybrid genetic algorithm with Tabu search for optimizing multi-dimensional functions and point pattern recognition. *Inf Sci* 221:28–48
27. Li X, Gao L (2016) An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem. *Int J Prod Econ* 174:93–110
28. Prajapati V, p et al Tabu Search Algorithm (TSA): A Comprehensive Survey. In: 2020 3rd International Conference on emerging technologies in computer engineering: Machine Learning and Internet of Things (ICETCE), <https://doi.org/10.1109/ICETCE48199.2020.9091743>.

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

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.