Sensing node selection and mobile sink displacement in the environments with multiple targets

Leila Saadatdoost1 | Homayun Motameni2 | Seyed Mehdi Hosseini Andargoli3 | Reza Ghazalian4

1Department of Computer, Islamic Azad University, Babol Branch, Babol, Iran
2Department of Computer, Islamic Azad University, Sari Branch, Sari, Iran
3Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran
4Faculty of Electrical and Computer Engineering, Buein Zahra Technical University, Buein Zahra, Iran

Correspondence
Homayun Motameni, Department of Computer, Islamic Azad University, Sari Branch, Sari, Iran.
Email: motameni@iausari.ac.ir

Abstract
In target tracking scenarios, a suitable sensor selection method in combination with designing mobile sinks (MSes) movement path is a promising solution to maximise the lifetime of the wireless sensor network (WSN). The lifetime optimization problem is non-deterministic polynomial-time hardness (NP-hard) and its optimal solution requires an exhaustive search with the exponential complexity. To tackle this issue, the original problem is separated into two more tractable sub-problems: first, the MSes' assignment and their positions' calculations, second, the SN assignment problem. The SNs assignment problem is an integer programming, which is relaxed to the more tractable form. Moreover, its solution results in a priority function which is combined with a numerical search algorithm to select the SNs. In the proposed framework, based on the estimated path of each target, a MS is assigned to the target. Besides, the MSes optimum position in the next rounds are calculated. Afterwards, for each target at each round, sensing nodes (SNs) are selected based on the targets' position and the MSes' position. Simulation results show that proposed solution increases the network's lifetime considerably in comparison with the benchmark algorithms while its complexity is much lower than that of the optimal exhaustive search algorithm.

1 | INTRODUCTION

WSN is an interesting technology for the remote monitoring and the automatic detection. The network lifetime is one of the main challenges in the WSNs. Since in these networks, SNs usually operate under limited energy budgets. In the other words, they are typically powered by batteries, which are depleted after some operations. Battery replacement often comes with the time and the finance costs, and is impractical in special situations. Thus, after a while, the SNs lost their energy and die one by one.

As a result, it is clear that, the more the dead SNs increase, the more the network coverage loses. Clearly, the WSN cannot be alive for a long time. In some applications, it is not essential that all sensors participate in the sensing phase. In fact, applying some nodes may be reduced the efficiency of the sensing performance as well as quality of service (QoS) satisfaction. In contrast, these measures, that is the sensing performance and the QoS, can be noticeably improved using appropriate SNs. Thus, in this condition, it is better that some nodes can go to sleep mode to preserve energy. Hence, applying task allocation, which is proposed in the literature, the network lifetime can be considerably improved.

The majority of the literature's reports reveal that, the data transmission energy is the main part of energy consumption at each SN. Data integration and data gathering, which is done by some nodes, can considerably reduce this energy consumption. Based on this view, the network clustering and the cluster head selection have been proposed in the literature. It is intuitively obvious that, a large volume of computations must be performed by the cluster head. Hence, applying some specific nodes with more computational abilities and more energy capacity as a cluster head, can significantly improve the network lifetime. It should be noted that, herein, the cluster head nodes are named as a sink. As the cost of sink hardware is a challenge, using a large number of sinks cannot be cost-efficient. Thus,
In Refs. [1] and [2], a MS periodically moves in a circular path on the border of the network to gather information from the cluster heads. After each period of the circulation, a new path is calculated through the ant colony algorithm. Herein, it is assumed that the information can be saved at the cluster heads. The saved information is transferred to the BS when the MS is located close to the BS. This strategy cannot be effective in the delay sensitive applications. In Ref. [3], it is supposed that there are some specific nodes with the sufficient memory storage. These nodes can receive information from the normal nodes, and deliver them to the MS in the next appropriate time. The position of these specific nodes along with the MS displacement is designed. It should be stated there is no analysis in this research. Clearly, this method cannot decrease the cost of the network. In Ref. [4], the data gathering problem has been formulated as a network utility maximisation problem. A MS travels along a pre-defined path to collect data from the SNs deployed in the monitoring environment. By solving the problem via the convex optimization framework, the optimal data gathering scheme has been proposed. In this research, the network lifetime has not been directly considered as a performance criterion. In Ref. [5], a tree-based clustering routing scheme has been proposed to optimise the SNs energy consumption. Indeed, the key aim of using this routing was to reduce the data transmission distance among the SNs. It should be noted that, although, the MS has been applied in this work, the MS path movement optimization has not been taken into account. In Refs. [6] and [7], the MS mobility path has been proposed to optimise the network lifetime. In this study, the MS moves on the pre-defined pattern path to gather the data from the cluster heads in each region of the network. In fact, the network region is divided into the same geographical size. Seemingly, this pattern cannot be well-organised in the delay sensitive applications. In addition, the MS mobility pattern has been not optimised. To prolong the network lifetime, an efficient MS mobility path has been presented [8, 9]. In this method, first, an appropriate set of data collection points is selected. Then, based on the selected data collection points, the MS constructs its path to visit these points to collect data form them. It seems that, applying this method makes a severe challenge in a network with a few number of the SNs as well as in a larger network size. Since in a network which uses a few number of the SNs, selecting the collected points will be challengeable. In that scenario, each SN should be equipped with high energy capacity or should consume more energy for data transmission. Therefore, this method cannot be effective in terms of energy consumption in the mentioned scenario. It should be stated that, in Ref. [9], if there is no collected point in a communication range of the SN, this SN directly sends its data to the MS. Based on the particle swarm optimization method, the algorithm for selecting cluster head nodes has been proposed [10]. This selection is done at a MS based on the nodes’ parameters, that is the residual energy and the positions. After selecting some proper cluster heads, the MS moves towards to the selected cluster heads node position to collect data. In this work, there is not any optimization for the MS path movement. A routing algorithm based on virtual grid infrastructure and mobility of the sink is proposed [11]. The MS randomly moves across the network to gather the data from the SNs. A new mobile path planning has been proposed based on the priority ordered dependent nonparametric trees for WSNs [12]. Hierarchical and dependent infinite tree structure provides a robust link connection between nodes, making it easier to reselect ancestor nodes. Ancestor node is a node with the highest priority and energy among nodes. MS visits ancestor node to collect data from all nodes. In this work, MS path have not been determined based on the event position. In Ref. [13], MS moves towards the SN that has sent the request signal to transfer data through MS to the BS. MS collects data sent by SNs in single-hop communication. To improve the system performance, the requests were also combined. MS serves SN based on the first come first service protocol. In this research, the path of MS has not been optimised. In Ref. [14], using modified travelling salesman problem, the optimal path for a MS that collects data from SNs has been presents. MS travels along the chords of the circles that represent the communication range of the SNs.

Herein, multiple MSes have been applied to monitor possible multiple events occurring simultaneously in the different geographical positions. In fact, to integrate the information received from SNs, the MSes are directed to the event positions. The integrated information transfers to the BS by the MSes. For the network lifetime maximisation, just part of the nodes is assigned to sense events and other nodes are put into sleep mode. It is assumed that the position of each event is known at the BS. All sensing assignments are carried out at the BS according to the solution of the lifetime maximisation problem.

Herein, in Section 2, the network's structure, the energy consumption model and the detection performance metric are described. The lifetime optimization problem along with its challenges is addressed in Section 3. Section 4 presents the simulation results and the performance evaluation of the proposed algorithm. Finally, conclusions are given in Section 5.

## 2 | SYSTEM MODEL

The considered WSN model is defined in this section. Herein, it is supposed that the length of the considered WSN is $L \times L \text{ m}^2$ in which $N$ SNs are distributed randomly with uniform distribution. It is assumed that the BS is located at the centre of the network. $M$ MSes transfer the gathered information from the SNs to the BS. Indeed, the targets' information are obtained by the SNs. Then, this information aggregates by the MSes. It should be noted that, the MSes are initially distributed uniformly in the area of the network. Besides, it is presumed that the energy consumption of the MSes is not a challenge. The maximum speed for each MS is limited to $V_{\text{max}}$. Moreover, at most $N_c$ targets are entered in the network based on the Poisson stochastic process with the known average
occurrence time. Also, the targets are passing through the network at a constant speed along a straight-line path. Although, in practice, the path and the velocity of the targets are estimated based on consecutive observations, it is assumed that these estimations are ideal and accurate. In addition, the maximum number of the targets passing through the network is assumed to be lower than the number of the MSes. It is worth mentioning that, due to the high-computational ability of the BS, all calculations and programing are carried out at the BS. Regarding to the results of the instantaneous information processing, the BS sends a command control data to the SNs and the MSes. This data contains the mode of the SNs, that is active or sleep, the direction of each MS and so on. The considered scenario has been shown in Figure 1.

Since the instantaneous positions of the targets are available, it is not rational to active all SNs to gather the target’s information. In fact, the far SNs cannot definitely provide accurate information about the targets. Thus, given the energy consumption of these SNs, the network lifetime dramatically decreases. Therefore, some SNs go to the sleep mode to preserve energy until they have been called by the BS via command control data. Furthermore, due to the mobility of the sinks, the BS calculates the movement path and the velocity of each MS based on the targets’ information, that is the position and the velocity. Regarding to these calculations, which are done in each \( \Delta t \) seconds, the SNs and the MSes receive the appropriate data about their future action via the command control sent by the BS. \( P_i^s \) denotes the position of \( i \)th SN which is static and \( P_{m}^n(t) \) and \( P_{j}^f(t) \) show the position of \( m \)th MS and \( j \)th target at time \( t \), respectively.

2.1 Energy consumption model

The energy consumption of each active SN contains three main parts: first, the energy consumed for the target detection, which is denoted by \( C_d \). Second, energy consumption for processing and preparing information for transmission, it is denoted by \( C_{elec} \). It should be noted that \( C_d \) and \( C_{elec} \) are constant and the same for all SNs. Third, the required energy for data transmission which is shown by \( C_t \). To satisfy minimum sensitivity at the \( m \)th sink’s receiver, \( C_t \) can be obtained [15, 16]:

\[
C_t = e_{amp} \left( d_{i,m}^{m-n} \right)^2
\]  

(1)

where \( d_{i,m}^{m-n} \) is the distance between \( i \)th SN and \( m \)th MS, \( e_{amp} \) is the constant related to the sensitivity of the receiver at the MSes. The total consumed energy at the \( i \)th SN can be modelled:

\[
C = C_t + C_{elec} + e_{amp} \left( d_{i,m}^{m-n} \right)^2
\]

(2)

It is obvious that the main parts of energy consumption at each SN are related to the data transmission. This fact has been also reported in the literatures and data sheets of SNs. Therefore, sinks can absolutely assist to reduce the amount of data transmission energy consumption by gathering information from near SNs and sending integrated information to the far BS. In addition, providing a large number of sinks with high computational and energy resources is impossible in the network. Thus, the mobility of the sinks helps to reduce communication range between each sink and the SNs across the entire of the network which considerably reduces energy consumption.

2.2 The target detection performance model

It is assumed that the key goal of the active SNs is to identify the presence/absence of a target near them. To this end, each SNs transmits a signal and processes the received echo signalling order to detect the presence of the target. Furthermore, in this processing, the distance of the target to the SN’s location can be measured. Due to the presence of random thermal noise with the received echo signal, the performance of target detection should be defined by the probability of detection \( P^D \). Based on envelope detector’s theory, detection probability of a target in \( i \)th sensor \( P^D_i \) can be modelled as [17]:

\[
P^D_i = Q \left( \sqrt{2T_hr}, T_hr \right) = \int_{T_hr}^{\infty} r e^{-r^2/2} I_0 (r \sqrt{2T_h}) dr,
\]  

(3)

where \( I_0(x) \) denotes zero-order modified Bessel function of the first kind, \( T_hr \) and \( \gamma_r \) denote threshold at the detector of the SNs and the signal-to-noise ratio (SNR) of the received echo signal, respectively. \( Q(.) \) shows the first-order Marcum Q-function. The detector threshold is adjusted according to the desired false alarm probability [17].

\[
T_hr = \sqrt{-2 \ln(P_{FA})}
\]  

(4)

\( \ln(.) \) shows the natural logarithm function. \( P_{FA} \) is the desired false alarm probability. The SNR of received echo signal can be...
obtained based on the wave propagation model versus distance between SN and target [17]:

$$\gamma_i = \frac{P^i \sigma}{(4\pi)^\frac{3}{2} (d_i^m)^4 KTBN_i}.$$  (5)

where $P$ and $l$ denote the power of transmitted signal, and the wavelength of signal, respectively. Besides, $\sigma$ is the radar cross section (RCS) of the target. The distance between the $i$th SN and the target is denoted by $d_i$. $K = 1.38 \times 10^{-23}$ J/K is the Boltzmann constant. $T$ and $B$ are, respectively, the standard noise temperature in kelvin and the bandwidth of the detector. $N_f$ is the noise figure of the receiver's circuit. $KTBN_i$ denotes the power of the received thermal noise. The omni-directional antenna with unit gain is assumed for SNs. Note that, we assumed that the ground effect and clutters can be cancelled in the SN's receiver by the Doppler processing, which is approximately compatible in terms of practical view.

With regards to (5), the distance between the target and SN plays the key role in the detection probability. Due to the random location of SN and the random direction of the target's movement path, it is likely that in some situations, there is no SN near the target. Thus, the detection probability considerably decreases. To tackle this issue, the cooperation between the SNs has been proposed in the literature. It is crystal clear that the combination of the received information from the multiple SNs greatly improves the reliability and the accuracy of the target detection. Two different combination strategies can be used in the network: soft combination, and hard combination. In the soft combination, the samples of the received signal at each SN are transmitted to the sinks or BS. Then, the sink combines these signals samples to obtain a signal with a better SNR, which results in more accurate detection. Transmission of all signal samples and synchronization of the SNs are accounted for the bottleneck of this method. Indeed, the implementation of the soft combination method is much complicated. In contrast, in the hard combination method, the signal processing is performed at the SN. The detection result at each SN is transmitted to the sinks (or BS). Then, the sink combines the received results of the multiple SNs based on a pre-determined combination rule. The prevalent combination rule includes AND, OR and M out of N [18]. In the considered network, each SN transmits the result of the presence/absence of the target, which can be one bit. Therefore, the complexity of the hard combination method is much lower than the soft combination one. With regard to the afore-mentioned materials, the OR combination rule is thus applied. In this combination rule, if one SN detects the target's presence, the overall decision is the presence of the target. When the OR combination is used, the global detection probability of the target $P^D$ can be obtained based on the local detection probability of the SNs $P^D_i$:

$$P^D = 1 - \prod_{i=1}^{N} (1 - P^D_i).$$  (6)

3 | THE PROBLEM STATEMENT AND PROPOSED METHOD

In WSNs, several SNs use battery as the energy source. Battery replacement often comes with time and the financial costs, and is impractical in special situations. Therefore, in the WSNs, presentation of an appropriate energy optimization technique is essential. Different strategies have been proposed in the literature to reduce the energy consumption or to manage energy consumption. One of the interesting approaches is task allocation. In this strategy, some nodes are assigned to sensing task and others go to sleep mode. Since the targets pass across the network, the new situation will be happened. Thus, it is essential that the task assignment changes. Indeed, by changing in the situations, some SNs’ mode must be change to meet the targets detection probability constraint. Applying the task allocation approach, the energy consumption of the slept SNs noticeably reduces while the performance of the network is maintained. Clustering is the other proposed method to reduce energy consumption. In this method, multiple SNs make a cluster and send their target detection results to the cluster head. Herein, MSes play the role of the cluster head. The cluster head aggregates the information sent by the cluster's member, integrates or summarises this information and then, transmits the summarised results to the BS. Applying this method leads to reduce the transmission energy of the SNs considerably. Cluster head can be definitely a special node with the high abilities in terms of energy and computational resources. Due to the high cost of the cluster head, it is thus hard to supply sufficient them to cover the entire network.

The MS has been recently considered to decrease the cost of multiple expensive sinks. In some papers, MS moves circularly on the middle circle of the network to cover inner and outer nodes. This movement is periodically repeated. At each time instance, MS gathers information from the near SNs. It is worthwhile to mention that; this method is not suitable for delay-sensitive applications. Furthermore, for monitoring the multiple events, this method cannot be effective. Since an immediate decision cannot be made about all the events by performing this method. To tackle this problem, it is assumed that there are multiple MSes with mobility in all directions in the network. These MSes move towards the targets and gather the information from the SNs. Indeed, in this strategy, it is attempted to respond to the multiple challenges of the network. First, there is no longer to require to the many fixed sinks. Therefore, the whole of the network can be covered by a few number of the MSes. Second, the MSes can approach to the targets, which cause to decrease the transmission energy of the SNs considerably. Certainly, by applying this method, the network lifetime significantly increases. Third, due to the presence of several MSes, this method provides an opportunity to simultaneously cover the several targets in the different geographical positions.

Besides, it is assumed that each MS only gathers and integrates the information of the SNs assigned to the specific target. In addition, each SN just can detect one target's scattered signal at any time.
The main goal is to optimise the MSs allocation to the targets, their movement path, and allocation of the SNs to the targets while lifetime of the network is maximised and desired detection probability of each target is met. There are different definitions for the network lifetime of a WSN in the literature. One of the interesting definitions of the network lifetime is a time duration in which specific part of SNs, for example 90%, are alive such that they can detect targets. An alive SN is a node which is able to send the sensed data to the sink successfully. In other words, an alive SN is a node which has the remained energy higher than the minimum required energy for sensing and transmission data to the sink, for example 10% of the initial energy. For the network lifetime extension, the energy consumption of the network should be not only minimised, but also uniformly distributed among the nodes. Based on the above-mentioned assumption, the BS manages the network and determines the assignments. This management is carried out based on the provided information obtained via the network’s component, that is MS and the SNs. Note that, energy consumption of active SN, for example ith SN, calculated by (2). Assuming that the remained energy of the ith SN is $E^{\text{remained}}_i$ and both of the $m$th MS and ith SN have been assigned to the $j$th target for the next round by BS, then the remained energy of ith SN will be $E^{\text{remained}}_i - \left[C_i + C_{\text{elec}} + e_{\text{amp}}(d_{m,j}^{m\rightarrow})^2\right]$ at the next round. We presumed that $\{\rho_{ij}\}$ and $\{\chi_{mj}\}$ are the allocation index of the SNs to the targets and allocation index of the MSs to the targets, respectively. In fact, $\rho_{ij} = 1$ reveals that the ith SN has been assigned to the jth target. In addition, $\chi_{mj} = 1$ demonstrated that the $m$th MS has been assigned to the jth target. Indeed, $\chi_{mj} \rho_{ij} = 1$ shows that the $m$th MS and the ith SN have been assigned to the jth target. Therefore, the ith SN sends the target detection results to the $m$th sink. Thus, in the general case, the remained energy of the ith SN at the end of the next round will be $E^{\text{remained}}_i - \rho_{ij}[C_i + C_{\text{elec}} + \sum_{m=1}^{M} \chi_{mj} e_{\text{amp}}(d_{m,j}^{m\rightarrow})^2]$.

Due to the random nature of events in the future, sink is forced to using per round decisions to manage the network. Consequently, based on the above-mentioned assumptions, the lifetime maximisation problem for each round can be formulated as:

$$\begin{align*}
\max & \{\rho_{ij}\}, \{\chi_{mj}\}, \{\bar{V}_m\} \\
\times \min_i \left(E^{\text{remained}}_i - \rho_{ij}[C_i + C_{\text{elec}} + \sum_{m=1}^{M} \chi_{mj} e_{\text{amp}}(d_{m,j}^{m\rightarrow})^2]\right) \\
\text{Subject to:} & \quad P_j^D = 1 - \prod_{i=1}^{N}(1 - \rho_{ij}P_j^D) \geq P^D_{TH} \forall j \\
& \in \{1, \ldots, N_t\} \\
& \|V_m\| \leq V_{\text{max}} \forall m \in \{1, \ldots, M\} \\
& \rho_{ij} \in \{0, 1\} \forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, N_t\}
\end{align*}$$

(7)

where $N_t$ is the number of targets have been simultaneously presented to the network. $M$ is the number of MSes. $N$ is the number of SNs are distributed in the network. The velocity vector of the $m$th MS at the next round is shown by $V_{\text{max}}$. Besides, (7) guarantees that the speed of the $m$th MS does not exceed the maximum speed of the MS. Constraint (7.5) ensures that each SN can be only assigned to one target. The fact that, each MS can be just dedicated to one target, has been shown in (7.6). Equation (7.7) guarantees that one MS has been assigned to each target. In Equation (7.1), $P^D_{TH}$ is the desired detection probability. The global target detection probability of the jth target is denoted by $P_j^D$. Plus, $P^D_{ij}$ shows the local detection probability of jth target at the ith SN. Considering (7), it can be easily found that the objective function of the mentioned optimization problem tries to balance the remained energy of all SNs per round.

The problem (7) is challenging in two aspects: First, this problem is NP-hard mixed integer programming. Finding the optimum solution requires an exhaustive search over the all possible allocation schemes with $O\left(M^{N_t}N^{2N_t}\right)$ complexity. Second, the optimum MS movement path must be directly calculated from the target path and the position of the allocated SNs at the next rounds. The SN allocation is unknown at the next rounds. Jointly solving of the problem for the multiple consecutive rounds can be a solution. However, finding the solution for the SN allocation problem is extremely complicated just for one round. Therefore, the complexity of the computation exponentially grows by increasing the number of rounds. The inherent complexity of this problem makes it impossible to solve it over successive rounds. Thus, to reduce complexity, the original problem is divided into two sub-problems, that is the MS allocation along with its movement path optimization problem, and the SN allocation problem. It must be noted that optimization per round cannot guarantee an optimal solution of the network lifetime maximisation problem while it is the only applicable strategy.

### 3.1 The MS allocation and the velocity vector calculation

At each round, the assigned MS’s calculated velocity vector certainly affects its position at the next rounds. Therefore, in the time duration in which a target enters and exits the network, the
velocity vectors are calculated. The position of the target can be exactly estimated at all rounds. At each round, it seems that, the best position of the MS is the nearest position to the assigned SNs. Since in this position, the energy consumption of each SN for data transmission can be noticeably decreased. Therefore, it can be concluded that the position near the target is desired since the SN near the target have the best detection performance and is likely to be candidate for the sensing task. Therefore, the sum of distances between the target and the MS in the time duration in which the target passing through the network, has been selected as the objective function of the optimization problem. For a MS which has been assigned to a target, the velocity vector calculation problem can be defined as follows:

\[
\begin{align*}
\text{min} & \sum_{t=2}^{t_{\text{end}}} (x^{1\text{st}}(t) - x(t))^{2} + (y^{1\text{st}}(t) - y(t))^{2} \\
= & \sum_{t=2}^{t_{\text{end}}} \left[ (x^{1\text{st}}(t) - x(1) - \sum_{z=1}^{t_{\text{end}}-1} V_{x}(z) \Delta t)^{2} \\
+ (y^{1\text{st}}(t) - y(1) - \sum_{z=1}^{t_{\text{end}}-1} V_{y}(z) \Delta t)^{2} \right] \\
\text{subject to} : & \left\| V_{t} \right\|^{2} = V_{x}^{2}(t) + V_{y}^{2}(t) \\
\leq & (V_{\text{max}}^{2})^{2} \forall t \in \{1, \ldots, t_{\text{end}} - 1\}
\end{align*}
\]

where \( x^{1\text{st}}(t) \) and \( y^{1\text{st}}(t) \) denote the target position at the \( t \)th round. In addition, \( x(1) \) and \( y(1) \) denote the initial position of the MS at the first round. Also, \( V_{x}(t) \) and \( V_{y}(t) \) show the elements of the target's velocity vector at the \( t \)th round.

It can be easily found that, this problem is a standard convex optimization problem (See the Appendix). Thus, it can be solved based on standard Lagrangian function and KKT (Karush Kuhn Tucker) optimality conditions. With regards to the objective function and constraints of the problem (8), the Lagrangian function can be obtained as [19]:

\[
L(V_{x}(t), V_{y}(t), \lambda_{z}) = \sum_{t=2}^{t_{\text{end}}} \left[ (x^{1\text{st}}(t) - x(1) - \sum_{z=1}^{t_{\text{end}}-1} V_{x}(z) \Delta t)^{2} \\
+ (y^{1\text{st}}(t) - y(1) - \sum_{z=1}^{t_{\text{end}}-1} V_{y}(z) \Delta t)^{2} \right] \\
- \sum_{z=1}^{t_{\text{end}}-1} \lambda_{z} \left( V_{x}^{2}(z) + V_{y}^{2}(z) - (V_{\text{max}}^{2})^{2} \right)
\]

where \( \lambda_{z} \) are the Lagrangian coefficients for the velocity constraints. Regarding the primitive optimality conditions, derivation of the Lagrangian function with respect to the optimization variables must be equal to zero [19].

\[
\frac{\partial L}{\partial V_{x}(z)} = \sum_{t=z+1}^{t_{\text{end}}} 2\Delta t \left[ x(1) + \sum_{z=1}^{t_{\text{end}}-1} V_{x}(z) \Delta t - x^{1\text{st}}(t) \right] \\
+ 2\lambda_{z} V_{x}(z) = 0, \forall z
\]

The optimality conditions result in \( 2(t_{\text{end}} - 1) \) linear equations with \( 3(t_{\text{end}} - 1) \) unknown variables. Furthermore, based on KKT complementary slackness conditions, we have:

\[
\lambda_{z} \left( V_{x}^{2}(z) + V_{y}^{2}(z) - (V_{\text{max}}^{2})^{2} \right) = 0 \quad \forall z
\]

\[
\in \{1, \ldots, t_{\text{end}} - 1\}
\]

As can be seen, (12) provides \( (t_{\text{end}} - 1) \) nonlinear equations. Considering (10), (11), and (12), the optimization variables can be obtained. The solution of the nonlinear equation system, that is (10) and (11), are calculated by numerical methods. If \( \{\lambda_{z}\} \) is known, the optimization variables will be obtained according to the solution of the equation system with \( 2(t_{\text{end}} - 1) \) linear equations as follows:

\[
[C_{ij}]_{t_{\text{end}} - 1 \times t_{\text{end}} - 1} \begin{bmatrix} V_{x}(1) \\ V_{x}(t_{\text{end}} - 1) \end{bmatrix} = [x_{i}]_{t_{\text{end}} - 1 \times 1}, \quad (13)
\]

\[
[C_{ij}]_{t_{\text{end}} - 1 \times t_{\text{end}} - 1} \begin{bmatrix} V_{y}(1) \\ V_{y}(t_{\text{end}} - 1) \end{bmatrix} = [y_{j}]_{t_{\text{end}} - 1 \times 1}, \quad (14)
\]

where elements of the matrix \( C = [C_{ij}]_{t_{\text{end}} - 1 \times t_{\text{end}} - 1} \) and elements of the known vectors \( [x_{i}]_{t_{\text{end}} - 1 \times 1} \) and \( [y_{j}]_{t_{\text{end}} - 1 \times 1} \) are:

\[
C_{ij} = \begin{cases} \Delta t(t_{\text{end}} - i) & j < i \\
\Delta t(t_{\text{end}} - i) + \lambda_{j} & i = j \\
\Delta t(t_{\text{end}} - i) & j > i
\end{cases}
\]

\[
x_{i} = x^{1\text{st}}(t_{\text{end}} - i) - \sum_{k=1}^{t_{\text{end}}-1} x^{t_{\text{end}}-1}(k)
\]

\[
y_{j} = y^{1\text{st}}(t_{\text{end}} - i) - \sum_{k=1}^{t_{\text{end}}-1} y^{t_{\text{end}}-1}(k)
\]

Taking into account (13)-(17), the unknown optimization variables are obtained:

\[
\begin{bmatrix} V_{x}(1) \\ V_{x}(t_{\text{end}} - 1) \end{bmatrix} = C^{-1} \times [x_{i}]_{t_{\text{end}} - 1 \times 1}, \quad (18)
\]

\[
\begin{bmatrix} V_{y}(1) \\ V_{y}(t_{\text{end}} - 1) \end{bmatrix} = C^{-1} \times [y_{j}]_{t_{\text{end}} - 1 \times 1}, \quad (19)
\]

To obtain \( \{\lambda_{z}\} \), a sub-gradient search algorithm such as ellipsoid is applied [19, 20]. A common choice of the candidate region is the minimal sized ellipsoid containing all candidate,
that is \( \{k_i\} \) 's. An ellipsoid with a centre \( \mu = [k_1, \ldots, k_{\text{pre}}] \) and a shape defined by the positive semi-definite matrix \( A^{-1} \) is defined as \[ (20) \]

\[
E(A^{-1}, \mu) = \{ x \mid (x - \mu)^T A^{-1} (x - \mu) \leq 1 \}
\]

Let \( d \) be the gradient of the Lagrangian function at the centre of the ellipsoid, that is \( \mu = [k_1, \ldots, k_{\text{pre}}] \):

\[
d = \left[ \left( V_s^2(1) + V_y^2(1) - (V_y^\text{max})^2 \right), \ldots, \left( V_s^2(t_{\text{end}} - 1) + V_y^2(t_{\text{end}} - 1) - (V_y^\text{max})^2 \right) \right]^T
\]

In each iteration, half of the ellipsoid is eliminated based on \( d \). A new ellipsoid, which is the minimal-volume ellipsoid containing the other half is formed. Mathematically, the update algorithm is as follows \[ (20, 21) \]:

1. \[ \hat{d}_i = \frac{d_i}{d_i^T A_i d_i} \]
2. \[ \mu_{i+1} = \mu_i - \frac{1}{t_{\text{end}}} A_i \hat{d}_i \]
3. \[ A_{i+1} = \frac{(t_{\text{end}} - 1)^2}{(t_{\text{end}} - 1)^2 - 1} \times \left( A_i - \frac{2}{t_{\text{end}}} A_i \hat{d}_i \hat{d}_i^T A_i \right) \]

The volumes of these ellipsoids can be demonstrated to decrease exponentially, that is \( \operatorname{Vol}(E_i) < e^{-\frac{1}{2} (t_{\text{end}} - 1)} \) \[ (19) \]. The iteration stops when \( \sqrt{\hat{d}_i^T A_i \hat{d}_i} < \epsilon \). Therefore, if the volume of the initial ellipsoid is \( \operatorname{Vol}(E_0) = \nu \), then the number of the iteration \( K \) is:

\[
K < 2(t_{\text{end}} - 1) \log \left( \frac{\nu}{\epsilon} \right)
\]

Based on the aforementioned materials, the velocity vector calculation algorithm from start round, which the target enters the network, to final round, which the target exits the network, can be summarised in Algorithm 1. It is worth stating that the MS displacement algorithm is performed after the MS assignment to the target. It is clear that, if the MSes velocity vectors are known, their positions will be also known.

### 3.2 MS to target allocation

Based on the key goal mentioned earlier, the MS to target allocation process is not performed in all rounds. Indeed, this process will be changed under specific conditions. Two main conditions cause to do the MS to target allocation process: when a new target enters the network, and when a MS is far away from its pre-designed position. When a MS goes far away from its pre-designed zone and enters to the another MS's zone, it is possible that a target will be entered this MS's pre-designed area. In this case, there is no MS near this target to desirably cover it. Thus, due to the aforementioned conditions, the MS to target allocation process should be executed. It should be noted that this process is carried out based on the distance between the MSes and the targets, and the distance between the MSes and their pre-designed location. The MS to target allocation algorithm has been summarised in Algorithm 2.

#### Algorithm 1 MS velocity calculation algorithm or MS displacement algorithm

Initial settings: appropriate settings for the matrix \( A_1 \) and the vector \( \mu_1 \). While \( \sqrt{d_i^T A_i d_i} > \epsilon \):

- Update elements of matrix \( C = [C_{ij}]_{i=1,t_{\text{end}}-1,j=1,t_{\text{end}}-1} \) based on vector \( \mu_t \) (15) and calculate velocity vectors of sink based on (18) and (19).
- Based on updated velocity vector, calculate the vector \( d_i \) based on (21)
- Update \( \mu_{i+1} \) and \( A_{i+1} \) based on (22)

End while.

#### Algorithm 2 MS to Target Allocation Algorithm

- Has a new target been entered the network or has a MS been gone away from its pre-designed position? If yes, do the following steps:
  - Calculate the distance of each MS from each target. \( d^\text{pre}-\text{tar}_m \) denotes the distance between the \( m \)th MS and the \( j \)th target
  - Calculate the distance of each MS from its pre-designed position. \( d^\text{pre}_m(t) \) denotes the distance between the \( m \)th MS and its pre-designed position.
  - For each target, sort \( d^\text{pre}-\text{tar}_m \) and select the \( \hat{j} \)th target where \( \hat{j} = \arg\max_j \left( \min_m d^\text{pre}-\text{tar}_m \right) \). Assign the \( \hat{j} \)th target to the \( \hat{m} \)th MS based on \( \hat{m} = \arg\min_m (d^\text{pre}-\text{tar}_m + d^\text{pre}_m(t)) \). Remove the \( \hat{j} \)th target and the \( \hat{m} \)th sink from the remained list and repeat this step for the remained targets and the MSes until all targets have been assigned.
  - If a MS has been not yet assigned to any targets, this MS comes back to its pre-designed location with the maximum speed on the straight path.
  - Based on the new MS to target allocation, runs MS velocity calculation algorithm, that is Algorithm 1.

### 3.3 Sensing node selection

After running the MS to target allocation algorithm and the MS velocity vector calculation algorithm, the position of each target and its assigned MS have been obtained at all rounds.
Therefore, with regards to the target position and their assigned MS, the SNs can be selected in each round. To this end, the solution of the following problem should be found in each round. Remained energy of all SNs at the end of previous round, local detection probability of all SNs in the current round and the distances between SNs and MSes are inputs to the problem.

\[
\max \min_{\{\rho_{ij}\}} \left( E_{i}^{\text{remained}} - \rho_{ij} \left[ C_i + C_{\text{elec}} + \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 \right] \right) \\
(24-1)
\]

subject to:

\[
P_j^{D} = 1 - \prod_{i=1}^{N} \left( 1 - \rho_{ij} P_{ij}^{D} \right) \geq P_{TH}^{D} \\
\forall j \in \{1, \ldots, N_i\} \\
(24-2)
\]

\[
\rho_{ij} \in \{0, 1\} \ \forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, N_i\} \\
(24-3)
\]

\[
\sum_{i=1}^{N} \rho_{ij} \leq 1 \ \forall i \in \{1, \ldots, N\} \\
(24-4)
\]

where \(X_{m,j}\) and \(d_{i,m}^{en}\) are known. The problem (24) is an integer programming. Thus, finding optimum solution of the problem (24) requires an exhaustive search over all feasible schemes with \(O(N^N)\) complexity. It is crystal clear that, the exponential complexity of exhaustive search algorithm makes it impractical. To tackle this issue, the problem (24) should be relaxed into more tractable form as:

\[
\max_{\{\rho_{ij}\}, TH} TH \\
\text{Subject to:} \rho_{ij} \left( E_{i}^{\text{remained}} - \left[ C_i + C_{\text{elec}} + \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 \right] \right) \geq \rho_{ij} TH \\
\forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, N_i\} \\
(25-1)
\]

\[
P_j^{D} = 1 - \prod_{i=1}^{N} \left( 1 - \rho_{ij} P_{ij}^{D} \right) \geq P_{TH}^{D} \\
\forall j \in \{1, \ldots, N_i\} \\
(25-2)
\]

\[
\rho_{ij} \in [0, 1] \ \forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, N_i\} \\
(25-3)
\]

\[
\sum_{i=1}^{N} \rho_{ij} \leq 1 \ \forall i \in \{1, \ldots, N\} \\
(25-4)
\]

To tackle the discrete nature of assignment index, that is \(\rho_{ij}\), it can be regarded as a priority index. It has been used an innovative technique to change this integer problem into a convex problem. Thus, this discrete variable, that is assignment index, has been changed into continuous variable on the interval of \([0,1]\). Furthermore, by defining \(TH = \min \left( E_{i}^{\text{remained}} - \rho_{ij} \left[ C_i + C_{\text{elec}} + \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 \right] \right)\) as the objective function and considering the constraint (25-2), the problem (24) is converted to the equivalent problem (25-1). In this problem, objective function (25), that is \(TH\), is a convex function since it is a linear function with respect to optimising variable, that is \(TH\) [19]. It is clear that (25-3) is linear to \(\rho_{ij}\).

Thus, (25-3) is a convex set. Further, It is intuitively clear that (25-4) is affine with respect to \(\rho_{ij}\) [19]. Therefore, constraint (25-4) is a convex set, too. However, the constraint (25-2), may not be a convex region [20, 22]. To prove this fact, we have defined \(g(\{\rho_{ij}\}) = 1 - \prod_{i=1}^{N} \left( 1 - \rho_{ij} P_{ij}^{D} \right) - P_{TH}^{D}\). It can be easily showed that in general Hessian matrix of function \(g\) is indefinite. In fact, the Hessian matrix of function \(g\) could be positive definite for some values of \(\{P_{ij}^{D}\}\), thus, this problem is not always a standard convex optimization problem. Although, If this problem becomes a non-convex optimization problem, the convex standard framework can be used to obtain an efficient solution, which may not be the global optimal solution. Based on the reported results in the literature, applying the convex optimization framework to find the solution of a non-convex optimization leads to reach local optimum solutions. These solutions may be often global optimum [20].

Based on the convex optimization framework, the Lagrangian function is formed as follows:

\[
L(\{\rho_{ij}\}, \{\xi_i\}, \{\nu_{ij}\}, \{\beta_i\}, \{\alpha_i\}) = -TH \\
- \sum_{i=1}^{N} \sum_{j=1}^{N} \eta_{ij} \rho_{ij} \left( E_{i}^{\text{remained}} - \left[ C_i + C_{\text{elec}} + \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 \right] \right) \\
+ \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 - TH \right) \\
- \sum_{j=1}^{N} \sum_{i=1}^{N} \rho_{ij} \left[ 1 - \prod_{i=1}^{N} \left( 1 - \rho_{ij} P_{ij}^{D} \right) - P_{TH}^{D} \right] \\
+ N \sum_{i=1}^{N} \sum_{j=1}^{N} \rho_{ij} (\rho_{ij} - 1) + N \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i (\rho_{ij} - 1) \\
= (26)
\]

Primitive optimality conditions reveal that:

\[
\frac{\partial L}{\partial \rho_{ij}} = -\eta_{ij} \left( E_{i}^{\text{remained}} - \left[ C_i + C_{\text{elec}} + \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 \right] \right) \\
+ \sum_{m=1}^{M} X_{m,j} e_{\text{amp}} \left( d_{i,m}^{en} \right)^2 - TH \right) \\
- \xi_j P_{ij}^{D} \sum_{k=1}^{N} \left( 1 - \rho_{k,j} P_{k,j}^{D} \right) - \nu_{ij} + \beta_i \alpha_i = 0 \\
(27)
\]

\[
\forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, N_i\} \\
\]

Clearly, (27) includes \(N \times N\) non-linear equations. Based on the KKT complementary slackness conditions, the following equations can be obtained:
\[ \eta_{ij} \rho_{ij} \left( E_{i}^{\text{remained}} - \left[ C_s + C_{elec} + \sum_{m=1}^{M} \chi_{m} e_{\text{amp}} \left( d_{i,m}^{e} \right)^2 \right] - \text{TH} \right) = 0 \]

\[ \forall i \in \{1, \ldots, N\} \quad \forall j \in \{1, \ldots, N_i\} \]

(28)

\[ \xi_j \left[ 1 - \frac{N}{\sum_{i=1}^{N} \left( 1 - \rho_{ij} \rho_{ji} \right)^2} \right] - \text{P}_{TH} \right] = 0 \]

\[ \forall j \in \{1, \ldots, N_i\} \]

(29)

\[ \nu_{ij} \rho_{ij} = 0 \quad \forall i \in \{1, \ldots, N\} \quad \forall j \in \{1, \ldots, N_i\} \]

(30)

\[ \beta_{ij} \left( \rho_{ij} - 1 \right) = 0 \quad \forall i \in \{1, \ldots, N\} \quad \forall j \in \{1, \ldots, N_i\} \]

(31)

\[ \alpha_i \left( \sum_{j=1}^{N_i} \rho_{ij} - 1 \right) = 0 \quad \forall i \in \{1, \ldots, N\} \]

(32)

which include \(3N \times N_i + N_i + N\) non-linear equations. A system of \(4N \times N_i + N_i + N\) non-linear equations can be solved by a complicated numerical method. In addition, the convergence of the complex numerical method, is much time consuming. Thus, it is tried to simplify the solution. Based on (27), we have:

\[ \frac{\partial L}{\partial \rho_{ij}} = 0 \Rightarrow \beta_{ij} \]

\[ = \eta_{ij} \left( E_{i}^{\text{remained}} - \left[ C_s + C_{elec} + \sum_{m=1}^{M} \chi_{m} e_{\text{amp}} \left( d_{i,m}^{e} \right)^2 \right] - \text{TH} \right) \]

\[ - 2 \rho_{ij} \text{TH} + \xi_j \rho_{ij} \sum_{k=1}^{N_i} \left( 1 - \rho_{kj} \rho_{kj} \right) + \nu_{ij} - \alpha_i \]

(33)

In addition, (31) can be written:

\[ \beta_{ij} \left( \rho_{ij} - 1 \right) = 0 \Rightarrow \beta_{ij} = \beta_{ij} \rho_{ij} \]

(34)

Considering (33) and (34), the priority index can be obtained:

\[ \rho_{ij} = \eta_{ij} \left( E_{i}^{\text{remained}} - \left[ C_s + C_{elec} + \sum_{m=1}^{M} \chi_{m} e_{\text{amp}} \left( d_{i,m}^{e} \right)^2 \right] - \text{TH} \right) \]

\[ + \xi_j \rho_{ij} \sum_{k=1}^{N_i} \left( 1 - \rho_{kj} \rho_{kj} \right) + \nu_{ij} - \alpha_i \]

\[ \beta_{ij} \]

(35)

(35) such that the constraints are manually satisfied. If it is assumed that, no SN have been activated yet, that is \(\rho_{kj} = 0 \quad k \neq i\), the priority index can be simplified to:

\[ \rho_{ij} = \eta_{ij} \left( E_{i}^{\text{remained}} - \left[ C_s + C_{elec} + \sum_{m=1}^{M} \chi_{m} e_{\text{amp}} \left( d_{i,m}^{e} \right)^2 \right] - \text{TH} \right) \]

\[ + \xi_j \rho_{ij} \]

(36)

For finding the optimal coefficients, that is \(\{\eta_{ij}\}\) and \(\{\xi_j\}\), the ellipsoid search algorithm can be applied. In the ellipsoid method, search space is updated based on the sub-gradient vector. Based on the Lagrangian function, elements of sub-gradient vector are:

\[ \frac{\partial L}{\partial \eta_{ij}} = \text{TH} - E_{i}^{\text{remained}} + \left[ C_s + C_{elec} + \sum_{m=1}^{M} \chi_{m} e_{\text{amp}} \left( d_{i,m}^{e} \right)^2 \right] \]

\[ + \xi_j \rho_{ij} \]

(37)

\[ \frac{\partial L}{\partial \xi_j} = \text{P}_{TH} - 1 + \sum_{i=1}^{N} \left( 1 - \rho_{ij} \rho_{ij} \right) \]

(38)

It should be noted that the updating process is the same as (22). In each iteration of the ellipsoid method, \(\{\eta_{ij}\}\) and \(\{\xi_j\}\) are updated based on the sub-gradient vector. Then, the priorities of the SNs are updated based on (36). Moreover, the SNs are selected based on the priority index values. These trends are repeated until the algorithm converges. The SN selection algorithm has been summarised as:

Algorithm 3 SN selection algorithm

Initial setting: set \(\mu_i\) and \(A_i\)

WHILE \(\sqrt{d_i^2 A_i d_i} > \epsilon\)

- Based on the updated \(\mu_i = [\{\eta_{ij}\}, \{\xi_j\}]^T\), calculate the priority of each SN for sensing each target \(\rho_{ij}\) based on (36)

While

\(\text{(there is} \ j \in \{1, 2, \ldots, N_i\} \text{with} P_j^D < P_{TH}^D) \&\)

\(\sum_{j=1}^{N_i} \sum_{i=1}^{N} \rho_{ij} < N\)

For \(j = 1:N_i\)

if \(P_j^D < P_{TH}^D\)

sort values of \(\rho_{ij}\) and assign the node with the highest priority to the target \(j\) and remove this SN from the remained SNs list.

End if.

End For.

End While

- calculate the sub-gradient vector and update \(\mu_{i+1}\) and \(A_{i+1}\) based on (22).

End WHILE.
4 | SIMULATION RESULTS

In this section, to evaluate the performance of the proposed algorithm, it is compared with benchmark algorithms in the same scenario using the numerical simulation. It is worth mentioning that these simulations are carried out in the MATLAB.

In addition, it is assumed that the spatial size of the network is considered $100^m \times 100^m$ length in which $N = 50$ SNs are distributed randomly with the uniform distribution. Figure 2 shows how SNs, BS and MSes locate in the network. As can be seen in Figure 2, the network area is divided into four equal squares with $M = 4$ MSes which are initially located at the centre of four squares. Also, the BS is located at the centre of the network. The maximum speed of each MS is taken into account to be $V_{\text{max}} = 20$ m/s. Duration of each round is $\Delta t = 0.5$ s. It should be noted that the SN assignments are performed at each round. The targets are entered into the network based on the Poisson probability distribution. The mean appearance is considered to be 0.03. In other words, on the average, in every 33 rounds, one target enters the network. Each target moves on a straight line at a constant speed. The target entrance position and the target exit position are random on the border of the network. It is assumed that the position of the target from the entrance position to the exit position is known at the BS. Based on the number of the MSes, up to four targets can be presented simultaneously in the network area.

It is presumed that, the initial energy of all SNs is 13 J. Each SN transmits $P = 1$ mw for the target detection. The energy consumption for sensing per round is equal to $C_s = 1^{\text{mw}} \times 0.5^{\text{sec}} = 0.5$ mj. Electrical processing energy per round is assumed to be $C_{\text{elec}} = 55$ $\mu$J. Besides, the coefficient $e_{\text{amp}}$ for the computing transmission energy at the rate of 250 kb/s is $e_{\text{amp}} = 5050$ nJ/m$^2$ [23, 24].

For calculating the target detection probability, the wavelength of the transmitted signal is $l = 12.5$ cm based on the carrier frequency $f_c = 2.4$ GHz. The RCS of all targets is assumed to be $\sigma = 1$ m$^2$. In addition, the bandwidth of the detector filter is $B = 1000$ Hz. The receiver’s antenna gain is assumed to be 0 dBi. The noise figure of the receiver is $N_f = 5$. The threshold of the detector is set based on a given false alarm probability $P_{\text{fa}} = 10^{-6}$. Moreover, it is assumed that the desired target detection probability is $P_{\text{d}}|_{\text{TH}} = 0.9$.

Based on our knowledge, there are no benchmarks in the literature to compare our work with. Therefore, similar to other works, benchmark algorithms are proposed for evaluations and comparisons. These benchmarks consider influential factor(s), which effects on network lifetime. These factors are: (a) the required active SNs to detect targets to meet the desired detection probability and (b) the communication range between each active SN and its assigned MS. (c) the distances between SNs and targets which influences on the local detection probability at the SNs. The considered benchmark algorithms have been presented as follow:

4.1 | Near-sink algorithm

In this algorithm, the sink to target allocation and the sink displacement is the same as our proposed algorithm. The SN selection is done based on the distance from the MSes. The SNs near the MS are allocated to sense the considered target. These SNs consume lower transmission energy than other nodes. It should be noted that the energy consumption for the data transmission is the main parts of the energy consumption at the SNs. Since the MS is located near to the target, the assigned SNs have a good detection performance, too.

4.2 | Near-target algorithm

Since the SNs near the target have the best detection performance, in this algorithm, SNs are allocated regarding to the distance from the target. The MS to target allocation and the MS displacement is the same as our proposed algorithm.

4.3 | Stand (fixed) sink algorithm

In this algorithm, the sinks are not mobile. Indeed, the sinks are fixed at the pre-designed location. For a fair comparison, the SN selection is the same as our proposed algorithm.

Figure 3 shows one of the realisations in which the target is randomly enters the network and passes through it. Besides, the allocated MS and its movement path have been also shown in this figure.

As can be seen, in the initial rounds, the assigned MS moves along the curved path towards the target with the maximum speed. Also, this MS adapts its velocity vector with the target in the next rounds.
In Figure 4, the number of alive SNs versus round number has been shown for the network with size 100 × 100 m². The network lifetime is the time duration in which the number of alive nodes is higher than 90% of the total distributed SNs. As can be seen, the network lifetime of the proposed algorithm is much more than other benchmark algorithms, followed by the fixed-sink algorithm. In the proposed algorithm and stand sink algorithm, the network does not die before 29,000 and 21,000 rounds, respectively. In the both near-sink algorithm and the near-target algorithm, the network dies after 10,000 rounds. It can be concluded that the appropriate SN selection strategy is more effective than sink mobility in the dense network.

The minimum and the maximum remained energy of the SNs versus round number have been shown in Figure 5. Considering the objective function of the network lifetime maximisation problem, the proposed SN selection algorithm tries to balance the remained energy of the SNs. However, the near-sink algorithm and the near-target algorithm try to minimise the energy consumption by reducing communication range and number of selected SNs, respectively. In this strategy, some SNs with the lower energy consumption may be selected in multiple successive rounds. Thus, the selected SNs die while some SNs may not be activated yet. It can be easily realized that the MSes can considerably improve the balance between the minimum and the maximum remained energy.

Furthermore, the simulations have been repeated for a network with size 200 × 200 m² and 100 distributed nodes. The number of alive SNs versus round number has been shown in Figure 6. The minimum and the maximum of the SNs’ remained energy versus round number have been shown in Figure 7. Besides, as can be seen from the Figure 6, the trend of the results is similar to the previous test. The main difference between these two tests, which can be found from Figures 4 to 7, is that the performance of the fixed sink algorithm significantly decreases in comparison with the proposed algorithm. In fact, in the fixed-sink algorithm, as the network’s size increases, the distance between the SNs near the target and the fixed sink increases. Consequently, the transmission energy consumption increases and leads to a shorter network lifetime.

The minimum and the maximum remained energy of SNs are shown in Figure 7. Increasing the network’s size leads to a decrease in the density of SNs. In this case, a few number of the SNs with a desirable target detection performance can be found. Thus, some SNs may be selected in multiple successive rounds and the balance between minimum and maximum remained energy decreases.
It is worthwhile mentioning that the proposed optimization framework is independent from the model of detection probability versus SNR or the model of SNR versus sensing range. Therefore, we have conducted a simulation in which the SNR value of the SNs’ target detector is modelled based on the free-space path loss as follows:

$$Y_i = \frac{P_i^2 \sigma}{(4\pi d_i^{m-\alpha})^2 K T B N_i}$$  \hspace{1cm} (39)

Note, the parameters of this simulation are the same as the first simulation’s parameters except that $P = 1 \mu W$. The result is given in Figure 8. As can be seen in Figure 8, there is a remarkable distinction between the proposed method and the benchmarks in terms of the performance, that is network’s lifetime. Indeed, the proposed method maintains its performance in comparison with the benchmarks. Thus, the generality of the proposed method will be unchanged by changing the SNR model.

The optimality of the proposed solution can only be assured when we compare it with the complex exhaustive search method. So, we have proposed a simplified exhaustive search algorithm to compare our solution’s effectiveness with it in terms of the network lifetime and the computation time. In this algorithm named ‘simplified exhaustive’, MS assignment and MS displacement are the same as the proposed method. For the SN selection with respect to each target in the simplified exhaustive search algorithm, the $N_d$ nodes with the highest local detection probability are selected as candidates. Then, all possible combinations of candidates are taken into account. For each possible combination of the candidate nodes, the minimum remained energy and the global detection probability are calculated. Based on these calculations, the combination of nodes which maximises the minimum remained energy and satisfies all constraints of the problem (24) is selected as an optimal solution for SN selection. It is clear that the simplified exhaustive is near optimal for the networks with large size and low number of nodes. In these kind of networks, with respect to each target, there are only a few nodes that can cooperate to detect target. So, $N_d$ can be a small number and the computational complexity of the exhaustive search can be tolerable. For the simplified exhaustive, the computational complexity order is reduced to $O(2^{N_d} N_t)$ in which $Nd$ and $Nt$ are small numbers. Based on this fact, to implement the exhaustive search method,

Besides, the comparison of the achieved network lifetime is summarised in Table 1.

The proposed algorithm is superior in the network lifetime maximisation in comparison with other benchmark methods.

**TABLE 1** The comparison of considered methods in terms of the network lifetime

| The Network Lifetime (Rounds) | Network Size is $100 \times 100$ m$^2$ and $N = 50$ | Network Size is $200 \times 200$ m$^2$ and $N = 100$ |
|------------------------------|---------------------------------------------------|---------------------------------------------------|
| Proposed                     | 29,000                                            | 7300                                              |
| Near-sink                    | 10,000                                            | 2800                                              |
| Near-target                  | 10,000                                            | 2800                                              |
| Fix-sink                     | 21,000                                            | 5000                                              |
search to handle the computational complexity. As can be found, in the large-scale networks with low number of nodes, sink's movement on the optimised path is more effective than the optimum SN selection. This issue shortens the lifetime of the fixed sink method. From the complexity view, the average computation time per round of the considered methods has been compared in Table 2. As can be seen, the computation time per round of the simplified exhaustive search method is about 24 times than that of the proposed method. It should be noted that the fixed sink algorithm does not spend time for the MS movement path design. Therefore, the fixed sink algorithm is the fastest method. Although the proposed method takes only twice as long as the near sink and near target methods, it considerably increases the network lifetime.

5 | CONCLUSION

Herein, multiple MSes have been used to access the information of the multiple events may be happened at the different geographical positions. To do that, our approach and results have been as follow:

1. We have formulated the network lifetime optimization problem as the maximising the minimum remained energy of network's nodes.
2. To tackle the complexity of solving the main problem, it has been separated into two sub-problems, that is MS assignment with movement path planning and SN selection.
3. We have solved sub-problems through the convex optimization framework and proposed three algorithms along with each other to manage MSes and SNs.
4. To evaluate our proposed method, we have introduced the benchmarks in which some factors with much more impact on the network lifetime are considered, that is sinks' movement, SNs' detection probability and communication range between SNs and MSes. Simulation results showed that the proposed method efficiently trades off between all of the influential factors while the benchmarks consider one or some of them.
5. The simulation results showed that the proposed algorithm outperforms the benchmark algorithms. The proposed method improved the Fix MS, Near-Target, and Near MS by 38%, 90.4%, and 90.4% in lifetime, respectively, for the network with 100 × 100 m² size and 50 nodes.
6. The superiority of the proposed method is not affected by the type of sensor or the model of network.
7. The proposed method's lifetime is longer than the near-optimal simplified exhaustive search while it reduces the computation time by 24 times.

200 × 200 m² network with 30 randomly distributed nodes is assumed. The other settings are the same as first and second simulations. For simplified exhaustive algorithm, $N_d = 5$ is considered. The number of alive nodes versus round number has been shown in Figure 9. As can be seen, the proposed method offers better lifetime in comparison with the near optimal simplified exhaustive search algorithm. This is due to simplification of the simplified exhaustive search method which has been taken into account on the original exhaustive

### Table 2: The comparison of considered methods in terms of computation time

|          | Proposed | Near-sink | Near-Target | Fixed-sink | Simplified Exhaustive |
|----------|----------|-----------|-------------|------------|-----------------------|
| Average elapsed time (s) | 0.0485 | 0.0217 | 0.0216 | 0.0142 | 1.1605 |
APPENDIX

The problem (8) is a convex problem

Proof: To prove convexity of problem (8), it must be shown that the objective function is a convex function and all constraints make a convex set [19]. The objective function of (8), that is

\[
    f(\{V_x(z)\}, \{V_y(z)\}) = \sum_{t=1}^{\tau} \left( x'(t) - x'(1) - \sum_{z=1}^{\tau-1} V_x(z) \Delta t \right)^2 + \left( y'(t) - y'(1) - \sum_{z=1}^{\tau-1} V_y(z) \Delta t \right)^2
\]

Hessian matrix of objective function is as follow:

\[
    \nabla^2 f = \begin{bmatrix}
        \nabla^2 V_x f & 0 \\
        0 & \nabla^2 V_y f
    \end{bmatrix} = 2 \times I_{(\tau_{end}-1) \times (\tau_{end}-1)} \nonumber
\]

Based on (40), it can be concluded that Hessian matrix is positive definite matrix and objective function is convex with respect to the optimising variables, that is \( \{V_x(t)\} \) and \( \{V_y(t)\} \).

To prove convexity of constraint (8-2), we have defined function \( G_t(t) = V_x^2(t) + V_y^2(t) - \left( V_{max}^2 \right) \). It is intuitively clear that \( G_t(t) \) is a convex function with respect to the optimising variables. Considering (8-2), we can concluded that (8-2) is the sublevel set of a convex function, that is \( g_t \). According to fact that the sublevel set of a convex function is a convex set, constraint (8-2) will be a convex set. Consequently, problem (8) is a convex optimization problem.

How to cite this article: Saadatdoost, L., et al.: Sensing node selection and mobile sink displacement in the environments with multiple targets. IET Wirel. Sens. Syst. 11(4), 147–160 (2021). https://doi.org/10.1049/iet-wss.2012.013

REFERENCES

1. Heidari, M., Nootemannade, A., Naji, H.: Effect of using mobile sink and clustering on energy consumption in wireless sensor networks. Turk. J. Electr. Eng. Comput. Sci. 23, 1–16 (2015)
2. Wang, J., et al.: Energy efficient routing algorithm with mobile sink support for wireless sensor networks. Sensors. 19(7), 1494 (2019)
3. Brindha, L., Muthaiah, U.: Energy efficient mobile sink path selection using a cluster based approach in WSNs. Int. J. Innov. Res. Comput. Commun. Eng. 3, 1936–1943 (2015)
4. Zhang, Y., He, S., Chen, J.: Near optimal data gathering in rechargeable sensor networks with a mobile sink. IEEE Trans. Mobile Comput. 16(6), 1718–1729 (2016)
5. Chang, J., Shen, T.: An efficient tree-based power saving scheme for wireless sensor networks with mobile sink. IEEE Sens. J. 16(20), 7545–7557 (2016)
6. Abo-Zahhad, M., et al.: Mobile sink-based adaptive immune energy-efficient clustering protocol for improving the lifetime and stability period of wireless sensor networks. IEEE Sens. J. 15(8), 4576–4586 (2015)
7. Zhou, Z., et al.: An energy-balanced heuristic for mobile sink scheduling in hybrid WSNs. IEEE Trans. Ind. Inf. 12(1), 28–40 (2015)
8. Weng, W., et al.: EAPC: energy-aware path construction for data collection using mobile sink in wireless sensor networks. IEEE Sens. J. 18(2), 890–901 (2017)
9. Kaswan, A., Nitesh, K., Jana, P.: Energy efficient path selection for mobile sink and data gathering in wireless sensor networks. AEU-Int. J. Electr. Commun. 73, 110–118 (2017)
10. Wang, J., et al.: Particle swarm optimization based clustering algorithm with mobile sink for WSNs. Future Gener. Comput. Syst. 76, 452–457 (2017)
11. Yarimezhad, R., Sabari, A.: Reducing delay and energy consumption in wireless sensor networks by making virtual grid infrastructure and using mobile sink. AEU-Int. J. Electron. Commun. 84, 144–152 (2018)
12. Yalcın, S., Erdem, E.: A mobile sink path planning for wireless sensor networks based on priority-ordered dependent nonparametric trees. Int. J. Commun. Syst. e4396 (2020)
13. Ghosh, N., Banerjee, I.: Application of mobile sink in wireless sensor networks. In: IEEE 10th International Conference on Communication Systems & Networks (COMSNETS), pp. 507–509 (2018)
14. Thomas, S., Mathew, T.: Intelligent path discovery for a mobile sink in wireless sensor network. Proc. Comput. Sci. 143, 749–756 (2018)
15. Heinzelman, W., Chandrakasan, A.P., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor. IEEE Trans. Wireless Netw. 1(4), 606–670 (2002)
16. Ammer, J., Rabacey, J.: The energy-per-useful-bit metric for evaluating and optimising sensor network physical layers. In: Proc. IEEE International Workshop Wireless Ad-Hoc Sensor Network, pp. 695–700 (2006)
17. Skolnik, M.: Introduction to radar systems, 3rd ed. McGraw-Hill, New York (2001)
18. Liang, Y.C., et al.: Sensing-throughput tradeoff for cognitive radio networks. IEEE Trans. Wireless Commun. 7(4), 1326–1337 (2008)
19. Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, Cambridge (2004)
20. Yu, W., Lui, R.: Dual methods for nonconvex spectrum optimization of multicarrier systems. IEEE Trans. Commun. 45(7), 1310–1322 (2006)
21. Seong, K.: Cross Layer resource allocation for Multiuser communication systems, Ph.D. thesis. Stanford University (2008)
22. Najimi, M., et al.: A novel sensing nodes and decision node selection method for energy efficiency of cooperative spectrum sensing in cognitive sensor networks. IEEE Sens. J. 13(5), 1610–1621 (2013)
23. Maleki, S., Pandharipande, A., Leus, G.: 'Energy-efficient distributed spectrum sensing for cognitive sensor networks'. IEEE Sens. J. 11(3), 565–573 (2011)
24. Part 15.4: wireless medium access control (MAC) and physical layer (PHY) Specifications for low-rate wireless personal area networks (WPANs). IEEE 802.15.4 Standard (2006)