



HAL
open science

A Distributed Energy-Efficient Algorithm for Cellular Target Tracking in Wireless Sensor Networks

Zohreh Alizadeh, Mohsen Afsharchi, Armin Ghasem Azar

► **To cite this version:**

Zohreh Alizadeh, Mohsen Afsharchi, Armin Ghasem Azar. A Distributed Energy-Efficient Algorithm for Cellular Target Tracking in Wireless Sensor Networks. 11th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI 2015), Sep 2015, Bayonne, France. pp.75-90, 10.1007/978-3-319-23868-5_6. hal-01385346

HAL Id: hal-01385346

<https://inria.hal.science/hal-01385346v1>

Submitted on 21 Oct 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

A Distributed Energy-Efficient Algorithm for Cellular Target Tracking in Wireless Sensor Networks

Zohreh Alizadeh¹, Mohsen Afsharchi², Armin Ghasem Azar¹

¹ Department of Computer Science and Information Technology
Institute for Advanced Studies in Basic Sciences, Zanjan, Iran
{z.m.alizadeh.b, armin.ghasemazar}@gmail.com

² Department of Electrical and Computer Engineering
University of Zanjan, Zanjan, Iran
afsharchim@znu.ac.ir

Abstract. Energy efficiency is one of the most challenging issues in wireless sensor networks, particularly in the target tracking. The main purpose of these networks is to preserve the distributed important sites from the targets who intend to destroy the sites in an environment. This paper proposes a distributed energy-efficient mechanism for sleep scheduling the sensors followed by a dynamic cellular clustering algorithm for tracking the targets. Probabilistic positions of the targets are predicted based on an improved particle filter procedure consecutively. A cell, including a subset of sensors, is constructed over time considering the predicted positions of targets. This cellular algorithm not only decreases the number of awake sensors, but also increases the tracking quality. Moreover, a concept named communication base has been proposed for alleviating the communication volume of the sensors. Utilizing communication bases, sensors are able to exchange information in maximum three hops. The experimental results demonstrate the capability of the provided algorithms and mechanism in optimizing the energy of sensors and increasing the tracking quality.

Keywords—*Wireless sensor networks, target tracking, energy efficiency.*

1 Introduction

A wireless sensor network is a network of spatially-distributed autonomous sensors for monitoring physical or environmental conditions to cooperatively disseminate their data through the network. Sensors are generally equipped with the capabilities of sensing, processing, and communicating. One of the most important issues in wireless sensor networks is energy consumption of sensors since these networks consist of a large number of battery-limited sensors which are typically used in the high-risk areas and thus the battery replacement is practically impossible [1]. The importance of energy efficiency of sensors is further significant in target tracking, as one of the most important applications of wireless sensor networks [2]. Target tracking in wireless sensors networks consists of accumulating information about target movement by sensors and delivering related information to other sensors or a base station to make an appropriate decision about the targets. Idle listening is a major source of energy waste for target detection and measurement [1]. In particular, sleep scheduling is one of the most commonly-used

mechanisms for reducing the energy consumption during idle listening in order to extend the network lifetime [3]. The idea of sleep scheduling is to put sensors in the sleep state for most of the time and only wake them up periodically. In certain cases, the sleep pattern of sensors may also be explicitly scheduled (i.e. forced to sleep or awaken on demand). Additionally, sensor clustering has been demonstrated to be an energy-efficient structure in sensor networks since information routing and relaying are only operated by special sensors namely cluster heads [4]. In each time step, a number of the sensors create a cluster and a sensor with a specific qualification becomes cluster head. Cluster heads can aggregate and process the information sent by cluster members, thus reduce the network load and alleviate the bandwidth.

This paper gives an account for utilizing a multi-objective optimization technique for localizing the sensors in the environment as a precursor for the main proposed algorithm. The assumed rectangular-shape environment is composed of some entrance gates and several solid obstacles followed by a set of important sites. Minimizing the number of sensors while maximizing the network coverage in the environment are the assumed objectives of the used technique which are obviously in conflict with each other. Furthermore, a distributed energy-efficient algorithm for sleep scheduling the sensors has been proposed followed by a dynamic cellular clustering procedure for effectively tracking the targets. In naive clustering approaches, a cluster head is designated to each cluster which is responsible for gathering the information from sensors and reporting it to other cluster heads. However, considering the fact that cluster heads must be awake continuously, their limited energy will not last long. Hence, this paper has substituted the cluster head with the concept of cell while the role of cluster head has been modified with constant communication bases. Furthermore, an improved particle filter algorithm based on the radial basis function networks has been used for predicting the trajectory of the targets through the time [5]. In each time interval, a cell is created based on the information gathered by the sensors and probabilistic positions of the targets in the previous time interval.

The rest of this paper is organized as follows. Section 2 contains a review of the related work. In Section 3, the comprehensive description of the system model is presented. The proposed algorithm followed by its required procedures are members of Section 4. Section 5 demonstrates experimental results. Finally, conclusion and future work are drawn in Section 6.

2 Related Work

Recent developments in target tracking have heightened the need for effectively maximizing the target tracking quality [1, 2]. Jiang *et al.* [1] applied a probability-based prediction mechanism in order to increase the tracking accuracy and reduce the sensors' active time by predicting the position of each target more precisely. Zhuang *et al.* [2] proposed an energy-efficient algorithm to accurately minimize the energy consumption of sensors in wireless sensor networks taking the probabilistic distance distributions into account. However, the number of hops needed to track the targets was high.

As one of the sleep scheduling approaches, proactive wake-up has been studied for awakening sensors pro-actively to prepare for the approaching target [3, 4]. Wang *et al.*

[3] represented a low-energy communication structure while the energy efficiency of wireless sensor networks is enhanced by cluster-based dynamic energy management. However, the role of cluster head was idle in the cluster which resulted in reducing the energy of cluster head and target loss. Furthermore, the sleeping policy introduced by Fuemmeler *et al.* [4] performed well in energy conservation and tracking the location of the target within the accuracy of the radius of the sensor, but there was a trade-off between the energy savings and the tracking errors.

Particle filtering for target tracking in wireless sensor networks has already attracted some attention [6, 7]. Recently, Djuric *et al.* [6] have proposed a class of interacting particle filters for multi-target tracking. This class of algorithms relied on splitting the state-space into lower dimensional sub-spaces in order to become computationally tractable, but did not guarantee that the particles were assigned proper weights. Additionally, the velocity of the targets has been assumed to be fixed. In real world, most of the times, the targets move in the environment with different velocities depending on the situations. DELTA was the distributed algorithm proposed by Wälchli *et al.* [7] which tracked the target solely at constant speed by dynamic clustering and selection of cluster head based on the light measurement. The advantage of DELTA was that the communication range of the sensors was higher than their sensing range. However, the main challenge of this approach was that it could only deal with targets' constant velocity.

As a result, according to the significant issues mentioned in the related work, followings are the main contributions of this paper:

- Proposing a new sensor cluster formation based on a new concept named cell;
- Providing a cellular algorithm for optimizing the energy consumption of sensors;
- Combining a multi-objective evolutionary technique for optimally localizing the sensors with the proposed energy-efficient target tracking algorithm;
- Employing a set of communication bases for effectively decreasing the communication volume of sensors by minimizing the hops in information exchange;

3 System Model

In this section, the proposed system model including its elements and running procedures will be described. It is assumed that there is an environment, e.g., airport, city, etc. The dimension of the environment is $M \times N$, such that $M, N \in \mathbb{R}^+$. This environment has $G \in \mathbb{N}$ entrance gates and $O \in \mathbb{N}$ various-shaped planar obstacles in this environment which it is not possible to access into them. Moreover, there are $T \in \mathbb{N}$ important sites such as school, hospital, etc. In this paper, individuals or objects, including minatory purposes, are named as the targets.

Wireless Sensor Networks: The main purpose in this paper is to track and then, destroy the targets which are trying to reach the important sites. Sensors are scattered in the environment such that targets are tracked with a high accuracy and low loss rate. In this paper, sensors have been assumed as radar-based ones which invisibly sense the targets. One sensor begins to sense in its sensing radius and communicate with other sensors within its communication radius. However, each sensor has a cost and limited

amount of energy. Therefore, it is desirable to use the minimum number of sensors. Here, due to simplicity, the purchasing cost of sensors has not been considered. Equation (1) refers to the first objective which is reducing the number of sensors.

$$f(s) = \min S. \quad (1)$$

$S \in \mathbb{N}$ denotes the number of sensors. Each sensor s_i , $i \in (1, 2, \dots, S)$, has a sensing radius $R_i^s \in \mathbb{R}^+$ and a communication radius $R_i^c \in \mathbb{R}^+$. This paper assumes that communication radius of each sensor $R_i^c \in \mathbb{R}^+$ is at least twice its sensing radius $R_i^s \in \mathbb{R}^+$. Next, (2) indicates the total overlapping area based on the pairwise overlap between each two independent sensors.

$$E = -\frac{1}{2} \times (|L_i^s - L_j^s|) \times \sqrt{(4 \times (R_i^s)^2) - (|L_i^s - L_j^s|)^2} - (2 \times (R_i^s)^2) \times \tan^{-1} \left(\frac{|L_i^s - L_j^s|}{\sqrt{(4 \times (R_i^s)^2) - (|L_i^s - L_j^s|)^2}} \right) + \pi \times (R_i^s)^2. \quad (2)$$

In this paper, it has been assumed that all sensors have the same sensing and communicating radius. L_i^s is the location of sensor s_i . As a result, (3) represents the second objective in the sensor placement problem, which implies maximizing the covered area. o_i is the obstacle o_i in the environment, where $|o_i|$ is its area.

$$g(s) = \max \left(\left(\sum_{i=1}^S \pi \times (R_i^s)^2 \right) + \left(\sum_{i=1}^O |o_i| \right) - E \right). \quad (3)$$

For expressing the function related to the second objective which is increasing the area covered by sensors, the main challenge is the distance between locations of each two sensors. Fig. 1 reflects the status of sensors with various possible assumptions. Fig. 1(a) demonstrates the distance between two sensors which is twice the sensing radius. It is obvious that these two sensors have no intersection with each other. Now, if another sensor is also placed next to them, a status similar to Fig. 1(b) is obtained. It is clear that the space between these three sensors is a blind space or non-covered space and ultimately increases the risk of losing the target. Furthermore, it is assumed that placing the sensors inside/on obstacles and important sites is not allowed. As a result, based on the experiments done and according to Fig. 1(c), the distance between each two independent sensors should not be less than 1.5 times the sensing radius.

Since satisfying the mentioned objectives and constraints at the same time is impossible, multi-objective optimization techniques have been used [8]. Multi-objective optimization is an area of multiple criteria decision-making, where mathematical optimization problems involving more than one objective function should be optimized, simultaneously. Optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives. Solving a multi-objective optimization problem necessitates computing all or a representative set of Pareto-optimal solutions. Here, each Pareto-optimal solution includes locations of a random number of sensors.

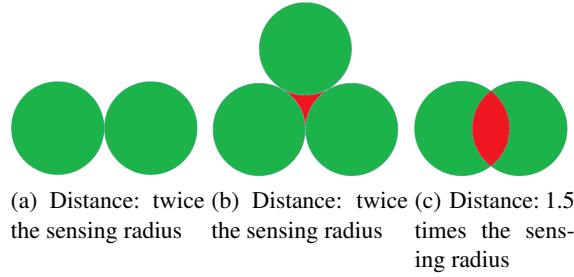


Fig. 1: The status of sensors with different distances from each other

Evolutionary algorithms are one of the most well-known meta-heuristic search mechanisms utilized for multi-objective optimization problems since their structure is free of search space and goal capacities [9]. Evolutionary algorithms involve biological mechanisms such as reproduction, mutation, recombination, natural selection and survival of the fittest. The main advantage of evolutionary algorithms, when applied to solve multi-objective optimization problems, is the fact that they typically generate sets of solutions, allowing computation of an approximation of the entire Pareto-front. Each Pareto-front includes a set of Pareto-optimal solutions. Currently, most evolutionary multi-objective optimization algorithms apply Pareto-based ranking schemes such as the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [10]. Applying this algorithm, S sensors with the same sensing radius are dispensed in the environment. Fig. 2 presents the arrangement of the sensors obtained from applying the NSGA-II including four obstacles and two important sites. The difference between Fig. 2(a) and Fig. 2(b) is the sensing radius and number of sensors. Increasing the sensing radius, the number of sensors decreases and consequently, the covered area also decreases. However, if the radius is too high or too low, the total covered area is close to the maximum. Due to the fact that the implementation and procedure of the NSGA-II is a pre-process for the main proposed algorithm, its description has been disregarded [10].

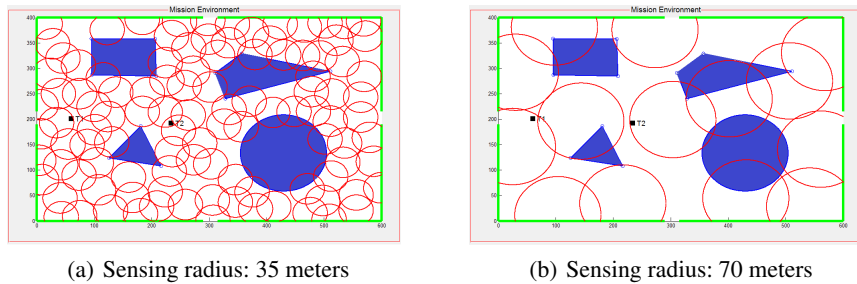


Fig. 2: An example of localizing the sensors through the NSGA-II

Energy Consumption of a Sensor: Each sensor is able to do a sort of actions and consumes some energy for performing each of the actions. Table 1 includes the energy each sensor consumes for performing the actions. Each sensor is equipped with a battery which has a limited energy. The reason for using battery-powered sensors instead of battery-less sensors is their high prices. As the battery goes out of energy, the sensor will break down and result in network failure and information loss. The actions 1 to 5 outlined in Table 1 are the ordinary actions each sensor performs in the environment. Two additional actions have been taken into consideration in this paper, namely, waiting for receiving the messages and delay in doing the actions 1-5.

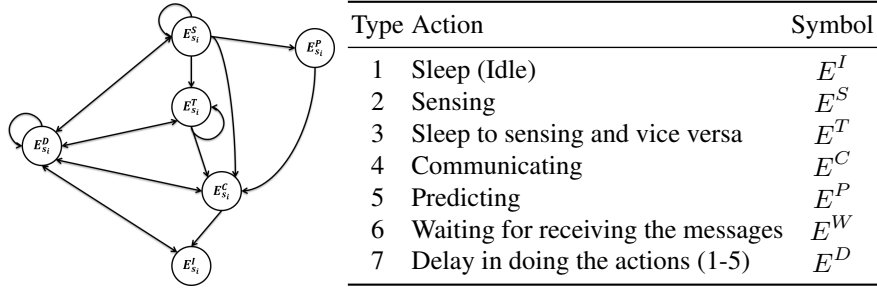


Fig. 3: The state diagram of en- **Table 1:** Actions and corresponding energy consumption variations for each sensor of the sensors

$E^I \in \mathbb{R}^+$ (mJ/s) denotes the energy the sensor consumes in sleep (Idle) state. Each sensor spends $E^S \in \mathbb{R}^+$ (mJ/s) when sensing the environment. $E^T \in \mathbb{R}^+$ (mJ/s) is the energy the sensors employ for changing their states from sleeping to sensing mode and vice versa. In addition, each sensor consumes an amount of energy to communicate with other sensors which corresponds to $E^C \in \mathbb{R}^+$ (mJ/s). Finally, $E^P \in \mathbb{R}^+$ (mJ/s) demonstrates the energy that sensors use for predicting the targets' location, velocity, and direction. Although doing each action types 1 to 5, mentioned in Table 1, has only a little delay, however, aggregation of these little delays may cause a tracking gap in the system. $E^D \in \mathbb{R}^+$ (mJ/s) implies the energy each sensor consumes when it has a delay in performing the mentioned actions. Furthermore, each sensor waits a limited time for receiving any message from its neighbors regarding the wireless communication and transmission delays. Sensors apply the $E^W \in \mathbb{R}^+$ (mJ/s) when they are waiting for receiving information from their neighbor sensors. The state diagram of energy variations for each sensor with respect to Table 1 can be seen in Fig. 3.

The loop in $E_{s_i}^D$ states that the sensor may remain in delay state continuously. This state is also applicable to $E_{s_i}^T$ and $E_{s_i}^S$. The edge between $E_{s_i}^S$ and $E_{s_i}^P$ also means that the sensor can predict the target after sensing the environment. As another instance, according to the edge between $E_{s_i}^T$ and $E_{s_i}^C$, the sensor begins to communicate with other neighbor sensors after waking up. As mentioned before, most of the energy of sensors is consumed for sending messages and communicating between sensors. Since sensors have a limited sensing and communication radius, therefore, they need to send the information they have collected via other sensors in the environment. To decrease this communication volume, a new concept namely communication base is added to the environment as follows.

Communication Bases: Sensors are not able to communicate with all other sensors due to their limited communication radius. In other words, if a sensor needs to communicate and send messages to other sensors which are far from it, it has to send the message through other neighbor sensors that probably results in wasting the energy and losing the packets across the path. For this purpose, $B \in \mathbb{N}$ communication bases is utilized. The bases are connected in a complete graph. As Fig. 4 illustrates, communication bases are scattered in the environment as follows: the environment is divided into $K \in \mathbb{N}$ equal parts and one communication base is randomly allocated to each part. Then, it is checked if all the sensors in each part are in communication radius of their related communication base or not. If not, another communication base is randomly assigned to that part and this process is repeated until all sensors are connected to the communication bases.

The next step is to merge the communication bases to have the minimal number. For each two communication bases, their average Euclidean distance is calculated and a new communication base is placed on that point. It is checked whether all sensors, related to two previous communication bases, can be connected to the new one or not. If so, previous bases are replaced by the new one. This process continues for each two communication bases in the same part. Afterwards, the merging process will be applied on the all survived and newly obtained bases in different parts until the minimal number of bases is obtained. As a result, in each area, depending on the type and number of obstacles as well as the communication radius of the sensors, a number of sensors are connected to each communication base. The role of communication bases in energy conservation is critical. Utilizing the communication bases, messages including the information of targets can be exchanged among sensors in at most three hops.

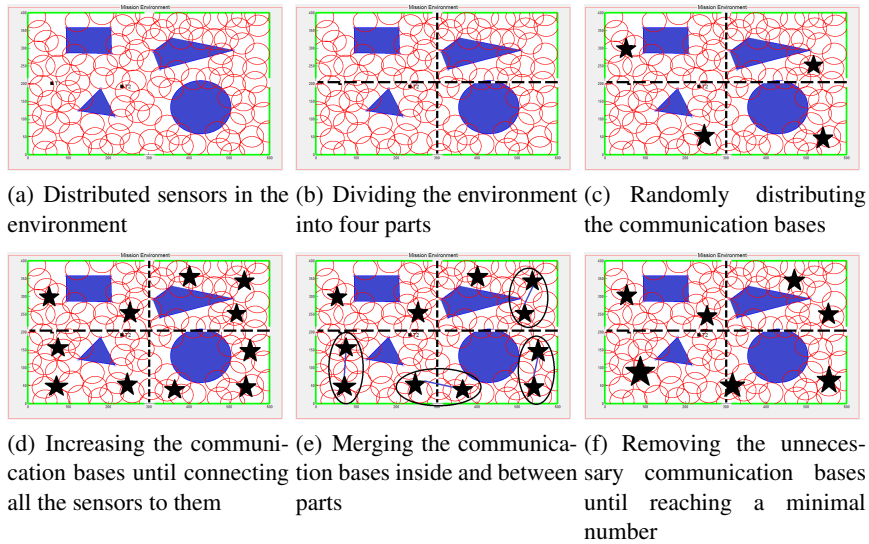


Fig. 4: An example of placing the communication bases in the environment

Unmanned Aerial Vehicles: As mentioned before, there are some entrance gates in the environment. Targets enter the environment from these gates and attempt to destroy one or more of the important sites as well in unknown time intervals. The task of wireless sensors is to predict the location, direction, and velocity of the targets with a probability. As previously mentioned, the main goal is to have all the sites at full health. Therefore, the Unmanned Aerial Vehicle (UAV) technology is used to protect them and destroy the targets [11]. For this purpose, a constant and independent UAV station is considered which is connected to all communication bases. Suppose the information including the presence of a target is delivered to the UAV station in time interval t . Depending on the information sent, this station orders to one of UAVs inside to fly over the environment. Since all the information received by the UAV station is approximate and probabilistic, the UAV flying will be also probabilistic. The UAV flies with the maximum speed until it reaches near the target. Reaching near the target, it decreases its velocity and integrates its spatial information with the information sent to the UAV station before and the new information obtained during the flight. Since the target's predicted position is probabilistic and also, there are many sensors around the target, the destruction should be done with a high accuracy. It is assumed that the probability of destroying the target after striking the missile is one. This assumption preserves the sensors from the damage.

4 The Distributed Cellular Algorithm

Initializing all the described procedures/elements is known as a pre-process for the main tracking algorithm. As the main parts of the proposed distributed cellular algorithm, two novel concepts named *boundary* sensors and sensors *cell* have been announced.

4.1 Boundary Sensors

As pictured in Fig. 2(a), a number of sensors are close to the entrance gates. Therefore, there is a unique set of boundary sensors in front of each entrance gate. The boundary sensors on the one hand are those which include the relevant gate in their sensing radius. Apparently, to be able to predict the direction of targets, these boundary sensors should always be awake and sense the environment, as there is no information of when and which gate the targets will enter the environment from. On the other hand, keeping all these boundary sensors awake is time and energy consuming. In this paper, it is assumed that the boundary sensors wake up periodically in specific time intervals and start sensing as long as a sensor detects a target. This approach will be a great help to reduce the energy consumption and eventually increase the reliability level of protecting the important bases. Fig. 5(a) is an example of boundary sensors in the environment. To be able to firstly, adapt this periodically sleeping/awaking strategy for boundary sensors and then, track the targets with maximum efficiency and accuracy, a concept called sensors' cell is proposed.

4.2 Sensors' Cell

In this section, the concept of the cell and its applicability in heightening the target tracking accuracy is introduced. As an important note, the targets are unaware of the

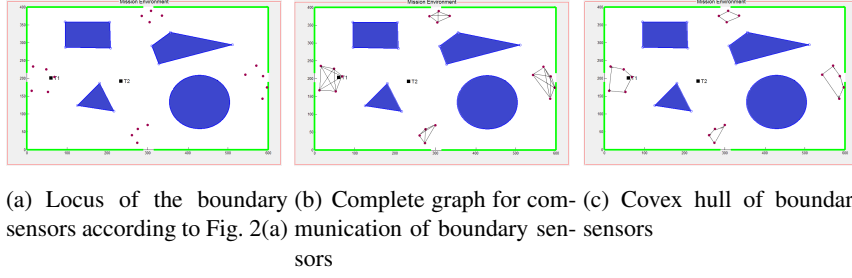


Fig. 5: Forming the preliminary cell among boundary sensors

location of sensors and the running algorithms. Each target moves into the environment from one of the gates. Once it appears in the sensing radius of at least one boundary sensor, this sensor will sense and predict the position, velocity, and direction of the target. Obviously, the target will continue to move and leave the sensor's sensing radius. Therefore, firstly, to predict the position of the target with higher probability, and secondly, not to lose the target, a preliminary cell among corresponding boundary sensors is created. Fig. 5 indicates how the sensors' cell is constructed.

According to Fig. 5(b), sensors, which have a gate in their sensing radius, are connected to each other via their communication radius and they create a complete graph. This graph is represented by $G = (V, E)$. V includes the set of boundary sensors and E is the set of edges which exhibits the communications between these sensors. To reduce the communication volume and have a planar graph, the convex hull of the vertices is created. The convex hull of a set of points in the plane is the smallest convex polygon which includes all points within or on the perimeter [12]. Fig. 5(c) depicts the convex hull of each set of boundary sensors. The target will certainly enter inside one of the preliminary created cells. Additionally, this target will definitely get closer to one of the sensors than the others. In other words, as mentioned before, in the first step, all boundary sensors wake up and sleep periodically until a target is recognized. As soon as a target is detected, the sensor, which is closest to the target, continues its work (sensing) and the others wait and remain in awake mode. In later steps, the target definitely continues its way and goes out of the sensor's sensing radius. As a result, the sensor sends the collected information to other sensors of the cell, and then, next sensor begins to sense based on the received information, and other sensors stay in awake mode. The information tables are updated in each time interval and target path is obtained. Sensors inside the cell utilize the probabilistic direction and inform the other sensors of the neighborhood cell. These neighbor sensors create the new cell to continue tracking in that cell. Next, the procedure of creating the new cells based on the exchanged information will be described in detail.

Creating the New Cells: Considering the fact that the targets move continuously and leave the cells consecutively, the preliminary-created boundary cells should be updated in order that the targets are not lost and the tracking procedure continues. In each time interval, the active sensor in the current cell broadcasts an alert message to the sensors

in the same cell and also its neighbor sensors which are not in the same cell. Fig. 6 indicates how to predict and create a new cell. According to target's estimated position and direction, sensors update their table and compare the prior position with the new estimated position and decide on which sensor the target is going to get close to. Utilizing the target's estimated direction, the new cell is created and this process is repeated until the target is tracked and destroyed.

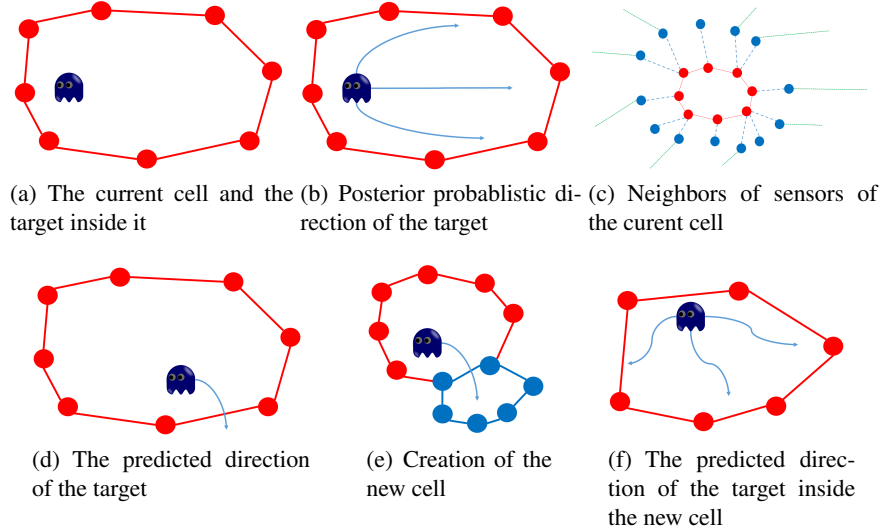


Fig. 6: Constructing a new cell based on the predicted trajectory of the target

4.3 Tracking

Efficiently tracking the targets is an indispensable task in the target tracking problem. To track a target appropriately, it is essential to estimate its position and direction to create the next cell before the target leaves the current cell. Considering the fact that the target's movement is non-linear, it is difficult to estimate its exact position. Hence, an efficient method, which is able to cope with target's non-deterministic movement, should be utilized. Here, the concept of a well-known probabilistic tracking method, i.e. *particle filter* (PF), is employed [13]. PF is a sequential Monte Carlo method based on representing the probability densities. This method uses an online posterior density estimation algorithm that predicts the posterior density of the state-space by directly implementing the Bayesian recursion concept. Nonetheless, the pure PF method cannot overcome the degeneracy of particles and accumulation of estimation errors. Therefore, an improved version of this algorithm using Radial-Basis Function Network (RBFN) has been engaged with the proposed distributed cellular algorithm [5].

This improved version of PF employs the RBFN in the sampling step for dynamically constructing the process model from observing and updating the value of each par-

title. With the sampling step in RBFN, the PF can give an accurate proposal distribution and maintain the convergence of a sensor system. RBFN is a three-layer feed-forward neural network which is embedded with several radial-basis functions. This network is characterized by an input layer, a single layer of non-linear processing neurons, and an output layer. The reason for applying RBFN is to dynamically construct the process model from observations and update the value of each particle. With the RBFN sampling step, this approach provides an accurate proposal distribution of a sensor. Fig. 7 demonstrates the changes in target tracking accuracy compared to the changes in the number of particles used in the PF.

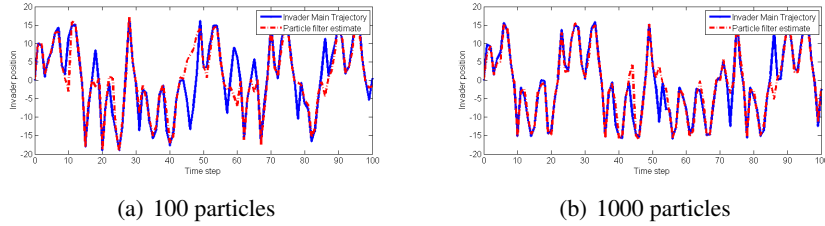


Fig. 7: Prediction accuracy versus number of particles

At each step of the algorithm, in which a cell is created and sensing and tracking operations are done inside that cell, all the information is circulated among communication bases. The reason is that since only communication bases are in contact with the UAV station, therefore, it is essential to send all information regarding the direction and velocity of all the targets to the UAV station. For instance, suppose tracking is being done in a cell and simultaneously a UAV flying over the environment is tracking the target. If another target enters the environment at that moment, a new instance of the tracking algorithm will be created. Hence, the communication bases should send the new information to UAVs station in order that new UAVs fly or the flying UAVs change their directions. In this paper, for simplicity, no constraint has been considered on the number of UAVs. Moreover, considering the communication protocols between all elements of the assumed problem is out of scope of this paper. As a final assumption for the tracking procedure, the targets are unable to destroy the UAVs.

5 Experimental Setup and Analysis

In this section, first, the experimental setup will be described, and subsequently, the experimental results will be clarified precisely.

5.1 Experimental Setup

The system model and proposed algorithms have been implemented with Matlab R2014b on a personal computer with an Intel Core i7 2.0 GHz CPU and 6 GB of memory. The

dimension for the assumed environment has been done in the context of $M = 600 m$ and $N = 400 m$. It has been assumed that there are $G = 4$ entrance gates (one in each side). Moreover, there are $O = 8$ two-dimensional planar and solid obstacles, and $T = 10$ important sites in the environment. The sensing and communication radius of all the sensors have been considered as $R^s = 35$ meters and $R^c = 70$ meters, respectively. Considering the communication radius twice the sensing radius represents the capability of the direct communication for the sensors which have a common sensing area. As the setting of evolutionary algorithm applied for sensors placement, the tournament selection operator (tournament size=2), linear crossover (probability=80%) and exchange mutation (probability=20%) have been set. Moreover, the generation and population sizes are equal to 400 and 100, respectively. The selected numbers here are the average of various tests. Therefore, $S = 83$ sensors are dispensed in the environment followed by $B = 11$ communication bases with respect to the assumed setting.

According to the intended purposes of this paper, the concept and characteristics of a sensor, namely Intel Imote2 [14] has been utilized. The reason for using this sensor is its capabilities in facilitating the tracking accuracy and quality. This sensor includes a CPU, a number of internal memories, and a communication device which uses the Zigbee radio communication to communicate between each two sensors. In addition, the sensor has a tiny OS that allows us to mount various algorithms on it. Followings are the corresponding amounts of energy consumed by each sensor for performing actions mentioned in Table 1.

- Action 1: 1 mJ/s
- Action 2: 13.4 mJ/s
- Action 3: 2.6 mJ/s
- Action 4: 11.5 mJ/s
- Action 5: 16.7 mJ/s
- Action 6: 9.6 mJ/s
- Action 7: 0.5 mJ/s

Obviously, the energy consumption of sensors in off mode is zero. The default amount of energy for each sensor in the environment is equal to 1800 mJ. The amounts mentioned for Types 6 and 7 are based on the same simulation running for the proposed algorithm. In this paper, to simplify the model, the energy of communication bases has been considered infinite. Trying to dispense the communication bases in the environment, the environment is divided into $K = 4$ parts. Furthermore, it has been assumed that the initial places of UAVs are as a point at one corner of the environment and their energy is infinite. Another point about the UAVs is their velocity. In this study, the velocity for each UAV has been selected from the range $[2, 12]$ m/s. Now, selecting the proper velocity depends upon the received information including the target's direction and velocity. However, in general, UAVs upon receiving the information, make their attempt to fly as quickly as possible to reach near the target, and then, reduce their velocity gradually in order to track the target more accurately.

The sensors and UAVs are not aware of when and where the targets enter the environment from. In other words, the targets know how many they are, in which time interval they enter the environment, which gate they enter the environment from, and which important base they will go through. Furthermore, it is assumed that the total number of targets is 25 in the simulation framework. However, the moving direction is randomly determined in each time interval. Finally, the targets select their desired velocity from the range $[0.5, 10]$ m/s.

5.2 Experimental Results

There are some studies in the field of centralized target tracking in which all sensors are usually awake and sense the environment [1, 15, 16]. The aim of those studies is to firstly update the position of targets continuously and then transfer it among all the sensors. The updating cost in centralized mode depends on the network size involving sending/receiving many messages within the network. In contrast to this approach, the proposed distributed algorithm needs to send and receive less messages. There is no need to exchange the information among all the sensors and even the neighbors since the targets choose a dynamic path for moving. This issue is also helpful for energy efficiency of sensors. In other words, since the prediction and tracking in each step is performed inside a cell, other sensors and even its neighbors are able to go to sleep mode which will also reduce the energy consumption. Thus, each sensor only needs to communicate with its neighbors inside the cell or receive information from them.

Suppose a sensor goes out of work for any reason, the algorithm has been designed in a way that it does not have much effect on tracking quality and associated operations. In other words, deleting a sensor from the network as well as inside the cell, one of its neighboring sensors will be replaced and the cell still lasts. Merging two cells, when a sensor or sensors are lost, imposes a cost consisting $O(n_V)$, where n_V is the number of sensors used in the merge operation. In addition, the network will inflict an extra cost for updating the tracking information when the target moves (from one cell to another cell). As a result, the cost equals to the number of cells the targets pass. It is beneficial to note that the 1000 particles have been used in the simulations.

To assess the efficiency of the proposed algorithm, it has been compared with three well-known algorithms, VigilNet [17], Forms [18], and iTOA (improved Time Of Arrival measurement model) [19]. VigilNet algorithm has been chosen for the similar concepts used in this paper, its high speed in tracking operations, and specifications of the sensors. Forms has been selected due to using an idea close to the one proposed in this paper. Finally, the reason for choosing iTOA is its similarity of prediction and tracking parameters used followed by the concept of communication base. The results obtained are the average of ten experiments done on the whole system. Three parameters including total target tracking and destroying time (I_T), energy efficiency of the sensors (EE), and tracking quality (Q_T) have been utilized in order to evaluate the performance of the proposed algorithm. Regarding the tracking quality, it can be noted that the purpose is to investigate how successful the system is against all difficulties including target tracking error, low detection probability, sensor destruction, and eventually the obstacles, which exist in the environment.

Hereinafter, the term DEECTA (Distributed Energy-Efficient Cellular Target tracking Algorithm) will refer to the proposed algorithm. In the paper, Fig. 8(a) represents the total target tracking and destroying time compared with other algorithms in terms of variations in the target tracking velocity. The calculation of I_T will continue since the simulation has started until all targets are destroyed. The numbers obtained for I_T in Fig. 8(a) are the average times consumed for tracking all targets. As a final note for this evaluation, in spite of the fact that assigning a greater velocity to the UAV will decrease the efficiency and even the destruction of target will be harder, however, the proposed algorithm works better than other algorithms even with a large number of target miss-

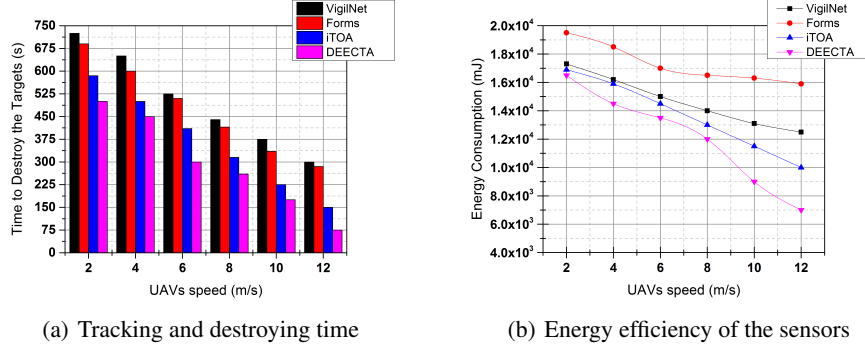


Fig. 8: Evaluations of the algorithm considering the changes in the tracking speed of the UAVs

ing. Fig. 8(b) represents the energy efficiency of sensors after destroying the targets. In this assessment, the performance of algorithm is evaluated with changes in UAVs velocity. It can be seen that when the UAVs' velocity fluctuates between 6 to 12 m/s, the amount of energy consumed by sensors is reduced. Furthermore, Fig. 8(b) indicates that the algorithm Forms utilizes more energy than other algorithms. The reason is the large number of sensors used in its tracking operation. The reason for superiority of the DEECTA algorithm is that at each step of the algorithm, a few number of sensors, that is, only one sensor in each cell, is in communication with others.

In Fig. 9(a), the impact of the sensor density on the performance of the algorithm in relation to energy consumption has been discussed with different sensing radius. If the sensing radius is large, then the density will be low. This in turn could increase the error rate of localization and target tracking since many small number of sensors are involved in tracking. Now, comparing the impact of density with the amount of energy consumed, the authors come to this conclusion that when the sensors density increases, energy consumption declines. The reason is the limited participation of sensors due to the idea of cell for tracking the targets. However, the proposed algorithm saves large quantity of sensor energies because of the reduction in the amount of communication involved. Finally, Fig. 9(b) represents the tracking quality compared to the different UAVs' speeds. In this case, the number of sensors involved in tracking will be important. Note that when the number of sensors is too high or low, tracking will be done slower than if average number of sensors are involved. In addition, changing the sensing radius of the sensors, as depicted in Fig. 9(a), influences the number of utilized sensors and consecutively the number of communication bases.

As a final point, the effectiveness of using communication bases on energy efficiency and tracking accuracy will be noted. Using these bases, energy efficiency of sensors increases by 48% compared to the case in which communication bases are not used. Also, the communication volume decreases by 79% using these bases. If communication bases did not exist in the environment, sensors had to communicate more in order to transmit their information to the UAV base which would also result in delay in tracking, rapid reduction of sensors' energy and increase in information loss. As a

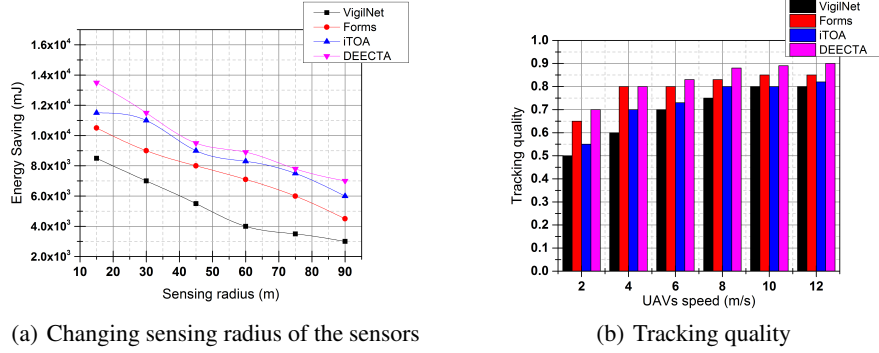


Fig. 9: Energy efficiency of the sensors in a basis of increasing the number of sensors combined with the tracking quality evaluation

result, in general, the existence of communication bases increases the security of information transmitted and makes the detection and destruction of targets faster and easier. To analyze the computation time required for running the algorithm, the experiments show that the Matlab uses only 300 megabytes with 35% of the CPU.

6 Conclusion and Future Work

In this paper, the problem of multi-sensor multi-target tracking using dynamic clustering and sleep scheduling in wireless sensor networks is investigated. As the main contribution, a distributed cellular cluster formation algorithm has been proposed to tackle with the problem of tracking and destroying the targets to protect the important sites in the environment. As a pre-process for the proposed algorithm, a multi-objective evolutionary algorithm has been used for the sensor placement with two conflicting objectives including minimizing the number of sensors and maximizing the area coverage. Next, a new concept named cell is proposed to minimize the number of awake sensors participated in tracking targets. Applying this method, only the sensors in the cell are turned on periodically and there is no need to wake up all sensors. Moreover, some communication bases have been used in order to decrease the sensors' amount of communications and also number of messaging hops. Maximizing the tracking quality motivates the authors to use an improved version of the particle filter algorithm based on radial basis function networks. Finally, in order to be able to destroy the targets moving in the environment, a UAV station including a set of UAVs, has been lodged in the environment. The results include reduction in communication volume, energy consumption, and also improvement of reliability and tracking quality.

As multiple works which can be done in the future, the possibility of applying the learning approaches for UAVs and sensors, considering a finite energy source for communication bases in order to propose a better and robust cluster formation and tracking mechanisms, improved target tracking procedure, and sleep scheduling algorithm for sensors can be mentioned.

References

1. Jiang, B., Ravindran, B., Cho, H.: Probability-Based Prediction and Sleep Scheduling for Energy-Efficient Target Tracking in Sensor Networks. *IEEE Transactions on Mobile Computing* **12**(4) (2013) 735–747
2. Zhuang, Y., Pan, J., Cai, L.: Minimizing Energy Consumption With Probabilistic Distance Models in Wireless Sensor Networks. In: *Proceedings of the IEEE INFOCOM*. (2010) 1–9
3. Wang, X., Ma, J.J., Wang, S., Bi, D.W.: Cluster-Based Dynamic Energy Management for Collaborative Target Tracking in Wireless Sensor Networks. *Sensors* **7**(7) (2007) 1193–1215
4. Fuemmeler, J.A., Veeravalli, V.V.: Smart Sleeping Policies for Energy-Efficient Tracking in Sensor Networks. In: *Networked Sensing Information and Control*. Springer (2008) 267–287
5. Wang, X., Wang, S., Ma, J.J.: An Improved Particle Filter for Target Tracking in Sensor Systems. *Sensors* **7**(1) (2007) 144–156
6. Djuric, P.M., Lu, T., Bugallo, M.F.: Multiple Particle Filtering. In: *Proceedings of the International IEEE Conference on Acoustics Speech and Signal Processing (ICASSP)*. (2007) 1181–1184
7. Wälchli, M., Skoczylas, P., Meer, M., Braun, T.: Distributed Event Localization and Tracking With Wireless Sensors. In: *Wired/Wireless Internet Communications*. Springer (2007) 247–258
8. Deb, K.: Multi-Objective Optimization. In: *Search methodologies*. Springer (2014) 403–449
9. Deb, K.: *Multi-Objective Optimization Using Evolutionary Algorithms*. Volume 16. John Wiley & Sons (2001)
10. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6**(2) (2002) 182–197
11. Chandler, P.R., Pachter, M., Rasmussen, S.: UAV Cooperative Control. In: *Proceedings of the IEEE Conference on American Control Conference*. Volume 1. (2001) 50–55
12. De Berg, M., Van Kreveld, M., Overmars, M., Schwarzkopf, O.C.: *Computational Geometry*. Springer (2000)
13. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A Tutorial on Particle Filters for Online Nonlinear/non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing* **50**(2) (2002) 174–188
14. Nachman, L., Huang, J., Shahabdeen, J., Adler, R., Kling, R.: Imote2: Serious Computation at the Edge. In: *International IEEE Wireless Communications and Mobile Computing Conference (IWCMC)*. (2008) 1118–1123
15. Zhang, W., Cao, G.: DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks. *IEEE Transactions on Wireless Communications* **3**(5) (2004) 1689–1701
16. Cao, D., Jin, B., Das, S.K., Cao, J.: On Collaborative Tracking of a Target Group Using Binary Proximity Sensors. *Journal of Parallel and Distributed Computing* **70**(8) (2010) 825–838
17. He, T., Krishnamurthy, S., Luo, L., Yan, T., Gu, L., Stoleru, R., Zhou, G., Cao, Q., Vicaire, P., Stankovic, J.A., et al.: VigilNet: An Integrated Sensor Network System for Energy-Efficient Surveillance. *ACM Transactions on Sensor Networks (TOSN)* **2**(1) (2006) 1–38
18. Sarkar, R., Gao, J.: Differential Forms for Target Tracking and Aggregate Queries in Distributed Networks. *IEEE Transactions on Networking* **21**(4) (2013) 1159–1172
19. Xu, E., Ding, Z., Dasgupta, S.: Target Tracking and Mobile Sensor Navigation in Wireless Sensor Networks. *IEEE Transactions on Mobile Computing* **12**(1) (2013) 177–186