An Optimize-Aware Target Tracking Method Combining MAC Layer and Active Nodes in Wireless Sensor Networks

Amir Javadpour

Published online: 25 April 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract
Target tracking applications are very popular in wireless sensor networks (WSN) as they can be used in industrial, healthcare, smart home, etc. This study proposes a method for reducing energy consumption in WSNs, considering target tracking. A prediction method is also presented using a combination of Medium Access Control layer and active modes to track the target while the rest of nodes are sleep. Nodes in network are categorized into target detection, target monitoring, and off modes. When target enters the environment, other available nodes sense it and change their situation into detection mode. Proposed method reduces energy consumption by decreasing number of involved nodes in target tracking and activating merely a limited number of necessary nodes. Target tracking continues as long as target is visible through representative nodes which are two closest nodes to the target. The conducted experiments using NS2 simulator and MATLAB represent higher performance of proposed algorithm compared to similar studies. The evaluation metrics are energy consumption, number of involved nodes and system throughput. This means the proposed method is superior and more efficient.

Keywords Active nodes · MAC layer · Energy efficiency · Target tracking · Wireless sensor networks · Target tracking

1 Introduction
Various solutions have been proposed for the long-standing problem of tracking mobile targets by means of radars, robots, and the Personal Communication Networks (PCN). Studies on distributed tracking started with Distributed Sensor Networks (DSN) program in Defense Advanced Research Projects Agency (DARPA) in early 80s. The project was initially defined as “Tracking with Large Numbers of Small and Inexpensive Sensors with Wireless Communication Capabilities.” This seemed an ambitious program considering the technology level of the time and they virtually had to study networks with a small number of nodes, which were large sensors (such as radars). By the advances of the technology...
in recent years, the initial program defined for distributed sensor networks activated and today large numbers of small sensor nodes are applicable for distributed tracking as Wireless Sensor Network [1–5]. WSNs contain large numbers of wireless sensor nodes that can sense one or more physical phenomena. These networks have drawn much attention due to large range of various applications they cover. One such application is target tracking [6], which in turn, consists of a wide range of various applications, including tracking of the enemy, animals, human targets, cars on the highways, and many more. On the other hand, sensor nodes of WSNs usually use batteries as energy source, recharging and replacing of which is very costly and sometimes even impossible. Therefore, maximizing the life span of the network through reduction of energy consumption of each node and balancing energy consumption of all nodes is the main challenge that research on target tracking in WSNs are faced with [4, 5, 7].

Different methods have been studies, considering various criteria such as scalability, communication overheads, energy consumption, and target tracking accuracy [5, 6, 8]. Have classified target tracking methods into cluster-based methods, tree-based methods, Mobicast message-based methods, prediction-based methods, and hybrid methods. In cluster-based tracking algorithms, node members of the cluster identify the target and send the information to the cluster head. The cluster head collects all the information, calculates the location of the target and finally sends the information to the sink node (central receiver) [9–11]. Given the cluster forming and calculation of the target location in each iteration of the algorithm, this method has a high time and calculation complexity and can be improved. Similarly, in tree-based algorithms, nodes form the tree structure before the network starts working or as soon as the target is identified and each node sends its information to the parent node. As the information is cumulated in the root node, the location of the target is identified and the corresponding report is sent to the sink node [12–16]. Similar to cluster-based method, this algorithm also requires much time and formation of leaves and transmission of information from leaves to parent and from parent to sink takes long periods of time. If the target is mobile, then the process will be more complex. Mobicast message-based methods depend on prediction. In these methods, a message is sent to a group of nodes that are changing based on the estimated speed of the target [17]. This message contains the time and location of the discovered object. Transmission of a group message leads to traffic increase and as a result, the speed of target finding decreases [8]. Prediction-based algorithms predict the next location of the target based on its recent speed and direction. Consequently, nodes of other areas, which are mainly neighbor areas where the target is more likely to go, activate before the target reaches them and switch back to sleeping mode after the target passes them. In these methods, sleeping mode is where the sensor is not tracking and sensing any targets and thus all its components are off except for the radio section of the sensor, which is in idle mode and awaits a signal from other nodes to start tracking and monitoring targets again [5]. Additionally, other nodes, which are not in the path of the target, are also kept in sleeping mode so that energy consumption is reduced and the network’s life span is increased. A challenge prediction-based algorithms face is activation of large numbers of nodes during their implementation. That is because the predicted next location of the target can cover a large area with a large number of sensor nodes. However, given the direction of the movement of the target, activation of many of these nodes is not necessary [18, 19]. Hybrid algorithms use a combination of the above-mentioned methods. Hybrid tracking methods are those tracking algorithms that meet the requirements of more than one type of target tracking methods. Distributed Predictive Tracking (DPT) [20] and Hierarchical Prediction Strategy (HPS) [21] are two examples of such algorithms. Due to the fact that problems of the existing methods reduce the quality of the network and thus reduce the accuracy of target
tracking, providing methods to increase the accuracy and optimize energy consumption is very important. This is what this paper is trying to do. The Sect. 2 of this paper reviews previous works in the literature. The Sect. 3 explains the proposed method and describes the problem. The Sect. 4 evaluates the results of simulation. Finally, the Sect. 5 presents conclusion and future works.

2 Literature Review on Target Tracking

The [22] has used Rumor method for tracking and routing. This method is a moderate state between announcing the existence of new information in the sensor node and requesting the information to be sent from the destination to the sensor node. Therefore, it functions as an interface between two networks that use flooding routing method, where the request to send information is distributed in one network while announcement of receiving new information is distributed in the other.

The [23] has used Multi Sink method to track several objects. In this method, each node sends a copy of information to all the neighbor nodes in order to distribute information. Upon receiving the information, every node resends the information to its neighbor nodes, save for the node it has received the information from. The time needed for a node to receive information and send it to other nodes is called a Round. The algorithm ends when all nodes of the network have received a copy of the information. The [24] has used Energy-Efficient Routing Protocol (EERP) for tracking and routing. This method uses geographical information for target tracking. It sends requests, which often include geographical properties. This protocol aims to limit and exclude the number of interests in release and direct distribution and to assign a specific area to sending interests to the entire network. In GEAR, each node holds an estimated cost and a learning cost in case of reaching the destination through its neighbors. The estimated cost is a combination of the remaining energy and the distance from the destination. The learning cost is a refinement of the estimated cost that is calculated for routing around the holes of the network. A hole is created when the node has no neighbor closer to the target area than itself. The estimated and learning costs are equal where there is no hole. When a packet reaches the destination, it jumps back so that to arrange the launch and create path for the next packet. That is how the learning cost is calculated. The [17] has used method Energy-Aware target tracking by reducing active nodes in Wireless Sensor Networks. This paper proposes a new tracking algorithm (EATT) reducing the number of active nodes in both positioning and tracking by predicting the target deployment area in the next time interval according to some factors including the previous location of the target, the current speed and acceleration of the target without reducing the tracking performance. There is not scheduling activates for transmuting data and all sensor nodes available in the target area by predicting the target position in the next time slot interval. The [25] has used hybrid method of Geographic Adaptive Fidelity (GAF) for tracking and routing. GAF is an energy-aware position-based routing algorithm that is primarily designed for mobile networks, but may also be equally applicable to sensor networks. GAF appropriately preserves energy by shutting down unnecessary nodes of the network without any change in the level of routing. It creates a virtual grid for the covered area in which each node uses its position, identified with GPS, to integrate itself with a target. Nodes integrated with the same nodes are considered equivalent packets in terms of routing costs. This equivalence is applied to some nodes in a unique network environment in the sleeping mode to store energy correctly.
Therefore, GAF can significantly increase the life span of the network as much as the number of nodes increases. The [26] has presented Hybrid Clustering for Multi-Target Tracking (HCMIT) in WSNs. In this paper, a dynamic clustering algorithm is presented for tracking distributed objects with the help of audio information. A hierarchical sensor network system is a combination of high-capacity and low-dispersion nodes with limited but dense power. High-performance nodes play as cluster heads and low-performance ones as cluster members. Cluster heads closer to the target are more likely to be activated than those far from the target are. Similarly, the probability that a cluster member sends data to the cluster head depends on its distance from the target. The [27] has used target tracking method based on data-driven protocols. In most target tracking applications of WSNs, the extended absolute number has made assigning global identifiers to each node impossible. Lack of a global identifier, along with the random expansion of sensor nodes, makes it more difficult to select a unique set of studied sensor nodes. Furthermore, sensor node data is transmitted with significant redundancy within one area. Therefore, it is ineffective considering energy consumption and routing protocols that will be able to select a set of sensor nodes and use the data density that is addressed during data recovery. This careful consideration leads to data-driven routing that is different from address-based routing. In data-driven routing, the base station sends the query to specified areas and waits for data to be received from sensors located in selected areas. This data-driven protocol addresses data transfer between nodes to remove additional data and save energy. Later, a direct release is created, which is an unexpected development in data-driven routing. Most other direct-release-base protocols are proposed based on a similar concept. The [28] has used Hybrid Energy-Efficient Distributed (HEED) clustering algorithm for tracking and routing. HEED is different from the previous algorithms in that it uses several parameters (rather than one) to form its subnets, which was mentioned in the previous section. The objective here is to allow each node to use its lowest level of power (as much as possible) in transmitting the data and store higher levels of its power to communicate with cluster heads, as well as sending data to the base station. The [25] has used hierarchical protocols for tracking and routing. Nodes with different capabilities can be used in WSNs. Some nodes have higher levels of energy, higher computing power and more memory compared to other nodes. The protocols used in this kind of sensor networks prioritize nodes. The network is divided into subnets; each consists of subgroups of nodes inside. Using specific algorithms, each node should transfer its information to its subgroup. The subgroup then sends this information directly or as a multiplex to the base station. Hence, we have a multi-layer protocol. The [29] has used P-LEACH (Low-energy adaptive clustering hierarchy) protocol for tracking and routing. This is a LEACH-based protocol that uses timing for adaptive mobility and adaptive traffic. In this protocol, cluster heads assign time intervals based on traffic and mobility of sensors. Each time interval, thus, is given two owners: An Original Owner and an Alternative Owner. The protocol keeps and serves new mobile sensor nodes in the database table, whether free or idle time intervals exist. The [30] has used Support Vector Machines (SVM) algorithm for multi-target tracking. This algorithm is based on Flooding and Support Vector Machine methods but it selected one node out of neighbor nodes randomly and sends information only to that node. Moreover, the receiver node can send the information to the sender node again, albeit if that node is selected through the random selection. Table 1 is illustrated summarize of different methods in target tracking.
Table 1 Summarize of different methods in target tracking

| Protocol | Description | Disadvantages | Advantages | Simulator |
|----------|-------------|---------------|------------|-----------|
| Rumor    | An interface between two networks that use flooding routing method, where the request to send information is distributed in one network while announcement of receiving new information is distributed in the other. All nodes can send an agent message, which increases the spatial complexity. | It contains a table of all routes, which is sent to all nodes so that they update their routes. | MATLAB |
| SINK     | In order to distribute information in the network, each node sends a copy of the information to all its neighbor nodes. Each node then resends the information it has received to all its neighbor nodes. | The nodes do not need initial information about the structure of the network and neighbor nodes. | MATLAB |
| EERP     | This protocol limits and excludes the number of interests in release and direct distribution and assigns a specific area for sending interests to entire network. If the packet is reached the destination area, it can be released through Recursive Geographic Forwarding or Restricted Flooding in that area. | A neighbor is selected to forward the packet based on learning cost function. This selection can be updated according to convergence of learning costs during packet delivery. | NS2 |
| GAF      | An energy-aware position-based routing algorithm that is primarily designed for mobile networks. This protocol has delay and relatively high spatial complexity. | Neighbors arrange their sleeping time so that they preserve routing in the system correctly. | NS2 |
| HCMT     | High-performance nodes play as cluster heads and low-performance ones as cluster members. Cluster heads closer to the target are more likely to be activated than those far from the target are. This protocol uses estimation algorithm for finding the target. Given the mobility of the targets, the algorithm cannot show the predicted location accurately. | Low delay for finding targets. | NS2 |
| EATT     | Energy-Aware target tracking by reducing active nodes in Wireless Sensor Networks. Cost Reduction, Extra Overhead. | Reducing the number of active nodes in both positioning and tracking by predicting the target deployment area. | NS2 |
| Data-driven protocols | The extended absolute number has made assigning global identifiers to each node impossible. | Computational Complexity | Optimal Energy Consumption | NS2 |
| Protocol               | Description                                                                 | Disadvantages                                       | Advantages                                                                 | Simulator |
|----------------------|------------------------------------------------------------------------------|----------------------------------------------------|---------------------------------------------------------------------------|-----------|
| Hybrid sensory       | Uses clustering and hybrid sensory method for tracking and routing.          | This algorithm does not balance the load            | Cost Reduction                                                            | NS2       |
| clustering           |                                                                              |                                                    |                                                                           |           |
| Hierarchical protocols | Uses hierarchical protocols for tracking and routing.                       | Extra Overhead                                      | Expandable, Easy management of information transfer routs and nodes       | NS2       |
| F-LEACH              | Uses timing for adaptive mobility and traffic. Cluster heads assign time intervals based on traffic and mobility of sensors | Extra Overhead                                      | Increases the packet delivery ratio and decreases energy consumption     | NS2       |
| SVM                  | Uses flooding and support vector machine methods for target tracking        | A node can receive the same information from different routes, which causes energy loss | The information overhead problem on neighbors of the tracked node is solved by SVM method | MATLAB    |
3 Proposed Method

This section presents the proposed method for finding node’s location and studies its tracking. Sensors with identifiers are used for implementing the tracking protocol. The aim of the protocol proposed in this chapter is to implement a routing method for tracking mobile objects in the network. Scalability and energy consumption reduction are criteria that are taken into account in the proposed algorithm. Desirable performance of a target tracking method depends on the following [12]:

- Detector nodes can identify the target location using locating methods and the locations of detector nodes.
- The route the target takes over time can be tracked through frequent reports.
- Given the limited energy of the nodes, minimum numbers of sensor nodes are active at each point of time.

The details of the proposition are as follows:

M is a mobile target entering the network at time k and point \((x_k, y_k)\) (Fig. 1a).
i and j are the two sensor nodes closest to M at time k.
i and j nodes can sense the target and their distances from M are \(d_i\) and \(d_j\), respectively.
The speed of M is \(V_k\) (Fig. 1b).
Assuming that the speed of the target does not exceed \(R_s/T\) during \([k, k + T]\) (\(V_k \leq R_s/T\)).

The proposition’s overall view is as follows:

The target’s location area at the next time interval is a circle centered on the current target coordinates and a radius proportional to the current target speed (Fig. 2). A specific limit is set for the speed of the target, which is equal to \(V_k \leq R_s/T\) based on proposition presented in [12].

![Fig. 1](image)

**Fig. 1** Details of proposition. 
(a) i and j are the two sensor nodes closest to M at time k, (b) the speed of M is \(V_k\)
3.1 Problem Statement

In the field of target tracking, we aim to track the target that enters an area of sensor nodes and we want the target to be accessible until it exits the area. The proposed method is based on prediction and thus it attempts to predict the next location of the target. This section explains the problem. When a target enters an area, the aim is to track it with the least amount of energy consumption. On the other hand, we need to ensure that the target is tracked. We are first going to explain the assumptions of the proposed method. As shown in Fig. 3, Rc is the communication radius of a sensor node, which is its communication range. The sensing radius, which is the range in which the node senses the environmental factors, is shown with Rs. It is assumed that $R_c \geq 2 \times R_s$. In the proposed method, if the nodes involved in target tracking can identify the target in the time slot, energy consumption is reducing. Obviously, reduction of the number of active nodes involved in tracking, as long as it does not damage the quality of tracking, leads to increase of life span of the network.

3.2 Proposed Algorithm

In the proposed algorithm, nodes in the network are categorized into target detection, target monitoring, and off modes. When the target node enters the environment, other available nodes, based on their Rc and Rs being in the range of the target, sense it and change into detection mode. In order to identify the target and given the fact that the target is mobile, nodes that have sensed the target send waking messages to neighbor nodes. Upon receiving these messages, neighbor nodes are activated and switch into detection mode to identify
the target. In our scenario, all nodes contain IDs (identification), which allow us to find out which nodes are involved in tracking. When the existing nodes observe the target in their radius, they notify each other of observation of the target through identification messages. The node with the least amount of numerical data is considered the representative node. The nodes that have not detected the target will switch into off mode in order to optimize energy consumption. Assuming that a slotted system is available, nodes involved in tracking use the same time slot model for sending and receiving data. At the beginning of each time slot the node with data to be sent is selected for sensing the channel and it is granted access to the channel based on the result of this sensing (Fig. 4). Different algorithms are used for sensing and accessing the channel to maximize throughput, as well as reducing interference in order to reduce energy consumption. At the end of the time slot, the receiver notifies the sender of successful receipt of the information by means of ACK (Acknowledgement). The time slot structure in the method proposed by this study is presented in Fig. 5. As shown in flowchart in Fig. 6, the target first enters the sensing area. Sensors that observe the target switch into detection mode and immediately start to communicate through sending packets. They cooperate with one another in this step in order to find the two nodes closest to the target and select a representative node. The representative node sends a message to the two closest nodes, which were identified and activated in the previous step, informing them that the target was observed. This message is sent in this step’s time slot through metastasis and the algorithm moves to the next step after its acknowledgement. The acknowledgement validates transfer of the packet. The two nodes closest to the target wake their neighbor nodes. If no node observes the target, the representative sends no message and the two nodes closest to the target in the previous step realize that the target is lost. In this case, the newly activated nodes switch back to sleeping mode. However, if the target is observed, the tracking operation continues.

4 Evaluation

This section compares the proposed method with other algorithms. The most important parameter in WSNs is energy consumption. Here, we compare the proposed method with method (A Novel Energy-Aware Target Tracking Method by Reducing Active Nodes in Wireless Sensor Networks) EATT presented in [17]. In EATT, all nodes are sensing and no

---

**Fig. 4** Finding the target and sending and receiving in the time slot
A pre-designed strategy exists. The simulation is conducted using NS2 software (Network Simulator2 [31, 32]) and output charts are created using MATLAB. Table 1 presents the simulation parameters and Table 2 contains network parameters of NS2. An example of the positioning of nodes in the simulation is also presented in Table 3.

4.1 Performance Criteria

Throughput is calculated by Eq. (1):

\[
\text{Throughput} = \frac{\mu}{t}
\]  

(1)

Throughput means, the amount of information moved from one place to another place in a given stipulated time slice, generally, it is measured in bits per second [33, 34]. Where \( \mu \) is the number of received bits and \( t \) is time (for the entire network, the result will be equal to average throughput of all node.) (Table 4).

End-to-End delay is calculated through Eq. (2):

\[
D = T_d - T_s
\]

(2)
### Table 2  Simulation parameters

| Parameter                                      | Value           |
|------------------------------------------------|-----------------|
| Initial energy of the node                    | 5 J             |
| Sensing radius (Rs)                           | 25 m            |
| Communication radius (Rc)                     | 50 m            |
| Network dimensions                            | 500 × 500       |
| Number of nodes                               | 250             |
| Length of control packets                     | 512             |
| Mobility model                                | Random Waypoint |
| Node distribution strategy                    | Random          |
| Energy consumption in sleeping mode           | 0.00027 J       |
| Energy consumption in sensing mode            | 0.012 J         |
| Energy consumption in communicating mode      | 0.0378 J        |
| Sensing range radius 25–30 m                  |                 |
| Communication range radius 50–60 m            |                 |
| Length of control packets 32 bite             |                 |
| Node initial energy 5 J                       |                 |
| Energy consumption during sleep amp 0.0013 J  |                 |
| Energy consumption during sensing amp 0.0000000000001 J | |
| Energy consumption during communicating amp 50 J |                 |

### Table 3  Simulation parameters of NS2

| Parameter                                      | Value           |
|------------------------------------------------|-----------------|
| Network Interface Type                         | Phy/WirelessPhy |
| Channel type                                   | Channel/WirelessChannel |
| Propagation/TwoRayGround                       | Radio-propagation model |
| MAC type                                       | Mac/802_11      |
| Antenna model                                  | Antenna/OmniAntenna |
| Interface queue type                           | Queue/DropTail/PriQueue |
| Max packet                                     | Random Waypoint |
| Queue type                                     | LL              |
| Number of nodes                                | 250             |
| Max packet                                     | 512             |
| Routing protocol                               | DSDV            |

### Table 4  Example of positioning of nodes in the simulation area

| Node (0) | Set X_5.0 |
|----------|-----------|
| Node (0) | Set Y_2.0 |
| Node (0) | Set Z_0.0 |
| Node (1) | Set X_390.0 |
| Node (1) | Set Y_385.0 |
| Node (1) | Set Z_0.0 |
| Node (2) | Set X_265.0 |
| Node (2) | Set Y_209.0 |
| Node (2) | Set Z_0.0 |


End-to-end delay is the time taken for a packet to be transmitted through a network from source to destination, it is called as delay [33, 34]. Where $D$ is the delay of one packet, $T_d$ is the receiving time of the packet at the destination and $T_s$ is the sending time at the origin.

Definitions of energy consumption and the tracking rate are given as follow:

Energy consumption is closely related to the number of sleeping sensors, sensing sensors and the number of sensors switched from communicating to sensing. Energy consumption of the node is calculated through Eq. (3) (Energy Consumption Model) [12]:

$$\text{NodeRemEng} = \text{NodeEt} - (E_{tx} \times N_t + E_{rx} \times N_r + E_{ix} + E_{sx})$$  \hspace{1cm} (3)$$

where $E_t$ is the total energy for each node, $E_{tx}$ is the energy consumption for sending the packet, $E_{rx}$ is the energy consumption for receiving the packet, $E_{ix}$ is the initial energy consumption, $E_{sx}$ is the cumulated energy consumption of all nodes, and $N$ is the number of nodes. Figure 7 presents an example of target tracking simulation in a space with random distribution of nodes. The target route is indicated with dashed line and the target stopping point is specified with a green star-like symbol. Sensor nodes in sleeping mode are specified with blue hollow circles, nodes that the target is located in their sensing radius are specified with $\ast$ and nodes that switch into monitoring mode after activation are specified with $\circ$. The target is examined as soon as it enters the simulation area and it is continued to be tracked until it exists the area.

4.2 Analysis of Results

The first experiment addressed energy consumption. This experiment examined the changes in sensing and communication radiuses of the existing nodes in the simulation.

![Fig. 7 Target movement in a space with random distribution of nodes](image)
Energy consumption in the proposed method was optimized with the changes of communication radius compared to EATT method [17]. Considering that the information of the time slot was used for tracking the target node, repeated transmissions were avoided and eventually, the energy consumption was optimized. In the proposed method, the energy consumption with changes in communication radius was optimized by less involvement of target observing nodes. In this experiment (Fig. 9), the energy consumption in the network was studies by changing the number of nodes in the simulation with a communication radius of 50 meters. By increasing the number of nodes in the communication radius areas the proposed method became optimum compared to EATT and the energy consumption rate changed with a moderate slope. The experiment in Fig. 10 shows examination of nodes involved in target tracking process. The number of nodes involved in target tracking is shown based on communication radius in the proposed method and in EATT. As the changes in communication radius increase, the number of involved nodes in the proposed method is less than EATT. This indicates decrease of energy consumption in the network.
The next experiment studies the energy consumption where the number of nodes in the network equals 250.

### 4.3 Calculating Total Energy

The energy consumed by the nodes consists of the energy used for saving the data and data recovery. Figure 11 shows a comparison between consumed energy by steps in the proposed method and EATT. When the network operates based on EATT method, a large amount of energy is lost due to nodes being permanently active and unnecessary listening during the idle time of the channel. The energy consumption of the proposed protocol is optimum considering its infrastructure. As shown in Figs. 11, the energy consumption of the proposed method optimum by repeatedly sending the message on the route between the sender and the receiver.
4.4 Calculating the Throughput

A counter is set in the data entry section for the node parameter of NS2 in order to calculate the proposed method’s throughput. This counter sends 3000 packets with specific length (byte) and different data rate (byte/s). This information moves between the specified origin and destination nodes. By each iteration of the proposed algorithm, sections related to Cyclic Redundancy Check (CRC) and the acknowledgment of the sent frame become inactive. The results show that changes in setting of data rate speed lead to different throughputs in the implementation system. The throughput changes by changing the data rate (data transfer speed) set in the sender node. As shown in Fig. 12 parts (A–D), the throughput of the proposed method increased compared to EATT by increasing the bandwidth rate and data transfer speed. The experiments were conducted based on the number of iterations (for x bytes) between the sender and the receiver. When the transfer rate is in 1 MB/s, the output of proposed method will have higher throughput. As the data rate is increase, the throughput of the proposed method is still higher than EATT. This indicates that the network remains stable and in good condition when the throughput increases. The proposed method was also efficient (compared to EATT) when the data rate increased in 3 and 4 MB/s.

Fig. 12  Network throughput (data rates 1–4 MB/s)
5 Conclusion

Reducing energy consumption and optimization are two important topics in target tracking. That is because the nodes are scattered in the network and it is hard to access them for battery replacement. Therefore, the energy-centric protocol in the network should be designed carefully. As mentioned above, sensors with IDs and time slots have been used to implement the tracking protocol. In the proposed algorithm, scalability and energy consumption reduction for WSNs was considered in tracking. In this method, the detector nodes determine the location of the target using locating methods and locations of detector nodes. To reduce energy consumption at any given moment, the least number of sensor nodes were activated in the tracking area. The proposed method was implemented and compared with the basic method of EATT through simulation. As discussed, the proposed method has a better energy consumption than the basic method. In addition, the proposed method had high throughput due to using time slots and this evaluation criterion has been investigated in number of iterations. The number of bits sent by the node and the number of bits reached in the network (with different rates) were reviewed. The high throughput of the proposed method is due to the use of time slots. This paper only focuses on movement of the target in one rout and its exit from the environment. In order to enhance the accuracy of prediction of the target movement, the records of each target can be kept, so that they can be used for better tracking of the target the next time it returns to the environment. In this regard, two type of environment can be examined: environments where one type of target enters and environment where several types of target enters. In the later environments, the target type is identified and it is recognized based on its acceleration, speed, and direction. Based on the proposed method in this study, after identification of the target, a more accurate prediction with fewer errors can be made considering the previous information about the target pattern. Future works are summarized such as 1-Changing the medium access layer and reassessing target tracking, 2-Examining metaheuristic methods for example ACO (Ant Colony Optimization) and PSO (Particle Swarm Optimization) algorithms for improving target tracking, 3-Examining delays in the presented working scenario and Using time slots considering the Markov forecast algorithm.

Acknowledgements This work is supported in part by the National Natural Science Foundation of China under Grants 61632009 and 61472451, in part by the Guangdong Provincial Natural Science Foundation under Grant 2017A030308006 and Hgh-Level Talents Program of Higher Education in Guangdong Province under Grant 2016ZJ01.

References

1. Bhatti, S., & Xu, J. (2009). Survey of target tracking protocols using wireless sensor network. In Fifth international conference on wireless and mobile communications, 2009. ICWMC’09 (pp. 110–115).
2. Chi, Y.-P., & Chang, H.-P. (2013). A tracking-assisted routing scheme for wireless sensor networks. Wireless Personal Communications, 70(1), 411–433.
3. Javadpour, A., Memarzadeh-Tehran, H., & Saghafi, F. (2015). A temperature monitoring system incorporating an array of precision wireless thermometers. In 2015 international conference on smart sensors and application (ICSSA) (pp. 155–160).
4. Li, L., Yang, W., & Wang, G. (2016). Intelligent fusion of information derived from received signal strength and inertial measurements for indoor wireless localization. AEU: International Journal of Electronics and Communications, 70(9), 1105–1113.
5. Zheng, J., Bhuiyan, M. Z. A., Liang, S., Xing, X., & Wang, G. (2014). Auction-based adaptive sensor activation algorithm for target tracking in wireless sensor networks. *Future Generation Computer Systems*, 39, 88–99.

6. Sharif, H. R., & Kavian, Y. S. (2015). Technological breakthroughs in modern wireless sensor applications. *Pennsylvania: IGI Global*.

7. Liu, Q., Guo, Y., Wu, J., & Wang, G. (2017). Effective query grouping strategy in clouds. *Journal of Computer Science and Technology, 32*(6), 1231–1249.

8. Safavi, S. M., Meybodi, M. R., & Eshaashari, M. (2014). Learning automata based face-aware mobile and improved differential evolution. *Wireless Personal Communications*, 77(3), 1923–1933.

9. Zhang, S., Choo, K.-K. R., Liu, Q., & Wang, G. (2017). Enhancing privacy through uniform grid and caching in location-based services. *Future Generation Computer Systems, 86*, 881–892.

10. Javadpour, A., Kazemi Abharian, S., & Wang, G. (2017). Feature selection and intrusion detection in cloud environment based on machine learning algorithms. In 2017 IEEE international symposium on parallel and distributed processing with application 2017 IEEE international conference on ubiquitous computing and communications (pp. 1417–1421).

11. Bhuiyan, M. Z. A., Wu, J., Wang, G., Cao, J., Jiang, W., & Atiquzzaman, M. (2017). Towards cyber-physical systems design for structural health monitoring: Hurdles and opportunities. *ACM Transactions on Cyber-Physical Systems, 1*(4), 19:1–19:26.

12. Rostami, A. S., Badkoobe, M., Mohanna, F., Keshavarz, H., Hosseinabadi, A. A. R., & Sangaiah, A. K. (2018). Survey on clustering in heterogeneous and homogeneous wireless sensor networks. *Journal of Supercomputing, 74*(1), 277–323.

13. Yin, X., Ling, Z., Guan, L., & Liang, F. (2014). Minimum distance clustering algorithm based on an improved differential evolution. *International Journal of Sensor Networks, 15*(1), 1–10.

14. Javadpour, A., & Memarzadeh-Tehran, H. (2015). A wearable medical sensor for provisional health-care. In *ISPTS 2015: 2nd international symposium on physics and technology of sensors: Dive deep into sensors, proceedings* (pp. 293–296).

15. Bhuiyan, M. Z. A., Wu, J., Wang, G., Wang, T., & Hassan, M. M. (2017). e-Sampling: Event-sensitive autonomous adaptive sensing and low-cost monitoring in networked sensing systems. *ACM Transactions on Autonomous and Adaptive Systems, 12*(1), 1:1–1:29.

16. Peng, Z., & Wang, G. (2017). An optimal energy-saving real-time task-scheduling algorithm for mobile terminals. *International Journal of Distributed Sensor Networks, 13*(5), 1550147717708791.

17. Shokouhi Rostami, A., Mohanna, F., & Keshavarz, H. (2017). A novel energy-aware target tracking method by reducing active nodes in wireless sensor networks. *Wireless Personal Communications, 86*(4), 3585–3599.

18. Chen, J. C. J., Cao, K. C. K., Sun, Y. S. Y., & Shen, X. S. X. (2012). Adaptive sensor activation for target tracking in wireless sensor networks. In 2012 IEEE international conference on communication (vol. 2012).

19. Khaksar, S., Pazzi, R. W., & El-Khatib, K. (2014) A prediction based clustering algorithm for target tracking in vehicular ad hoc networks. *Proceedings of the fourth ACM international symposium on development and analysis of intelligent vehicular networks and applications* (pp. 39–46).

20. Wang, Z., Lou, W., Wang, Z., Ma, J., & Chen, H. (2010). A novel mobility management scheme for target tracking in cluster-based sensor networks. In *R. Rajaraman, T. Moscibroda, A. Dunkels, & A. Scaglione (Eds.), Distributed computing in sensor systems sE—13* (Vol. 6131, pp. 172–186). Berlin: Springer.

21. Banaezadeh, F., & Haghighat, A. (2015). Evaluation ARIMA modeling-based target tracking scheme in wireless sensor networks using statistical tests. *Wireless Personal Communications, 84*(4), 2913–2925.

22. Sann, Z., & Minn, K. T. (2011). Simulation of the rumor routing algorithm in sensor networks. In 2011 3rd international conference on computer research and development (ICCRD) (vol. 3, pp. 10–14).

23. Liu, D., & Zhang, J. (2011). A multi-sink and multi-object tracking strategy for wireless sensor networks. In 2011 international conference on electrical and control engineering (ICECE) (pp. 4273–4276).

24. Ghaffari, A. (2014). An energy efficient routing protocol for wireless sensor networks using A-star algorithm. *Journal of Applied Research and Technology, 12*(4), 815–822.

25. Wang, Z., Wang, Z., Chen, H., Li, J., Li, H., & Shen, J. (2013). HierTrack: An energy-efficient clustered-based target tracking system for wireless sensor networks. *Journal of Zhejiang University SCIENCE C, 14*(6), 395–406.

26. Campos, A. N., Souza, E. L., Nakamura, F. G., Nakamura, E. F., & Rodrigues, J. J. P. C. (2012). On the impact of localization and density control algorithms in target tracking applications for wireless sensor networks. *Sensors, 12*, 6930–6952.
27. Hajiaghajani, F., Naderan, M., Pedram, H., & Dehghan, M. (2012). HCMTT: Hybrid clustering for multi-target tracking in wireless sensor networks. In 2012 IEEE international conference on pervasive computing and communications workshops (PERCOM Workshops) (pp. 889–894).

28. Naderan, M., Dehghan, M., Pedram, H., & Hakami, V. (2012). Survey of mobile object tracking protocols in wireless sensor networks: A network-centric perspective. International Journal of Ad Hoc and Ubiquitous Computing, 11(1), 34–63.

29. Cho, S., Han, L., Joo, B., & Han, S. (2014). P-LEACH: An efficient cluster-based technique to track mobile sinks in wireless sensor networks. International Journal of Distributed Sensor Networks, 10, 803656.

30. Kim, W., Yoo, J. H., & Kim, H. J. (2012). Multi-target tracking using distributed SVM training over wireless sensor networks. In 2012 IEEE international conference on robotics and automation (ICRA) (pp. 2439–2444).

31. Issariyakul, T., & Hossain, E. (2012). Introduction to network simulator NS2. SpringerUS. https://books.google.nl/books?id=_VkTzFLnwD4C.

32. Moussa, N., Nasri, F., & Tourki, R. (2014). NOC architecture comparison with network simulator NS2. International Journal of Engineering Trends and Technology, 13(7), 340–346.

33. Kumar, K. S. A., & Balakrishna, R. (2017). Comparative analysis of delay and throughput using IEEE 802.11 and receiver centric-MAC protocol in wireless sensor networks. In 2017 innovations in power and advanced computing technologies (i-PACT) (pp. 1–5).

34. Javadpour, A., Adelpour, N., Wang, G., & Peng, T. (2018). Combing fuzzy clustering and PSO algorithms to optimize energy consumption in WSN networks. In 2018 IEEE SmartWorld, ubiquitous intelligence & computing, advanced & trusted computing, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (pp. 1371–1377).

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

Amir Javadpour received the B.Sc. in ICT in 2010 from Tamishan University of Behshahr, Iran, and received his M.Sc. degree in 2014 in Medical Information Technology Engineering from University of Tehran, Iran. He is currently a Ph.D. Candidate at Guangzhou University, China. His research interests include cloud computing, Steganography, Wireless Target Tracking, Wireless Body Area Network (WBANs), Big Data, Intrusion Detection Systems (IDS), Healthcare Real time monitoring measurement, Machine Learning, Wireless Sensor Network, SDN network and Secure Communication.