

Review Article

A Comparative Study of Target Tracking Approaches in Wireless Sensor Networks

Asmaa Ez-Zaidi and Said Rakrak

Laboratory of Applied Mathematics and Computer Science (LAMAI), Faculty of Science and Technology, Cadi Ayyad University, 40000 Marrakesh, Morocco

Correspondence should be addressed to Asmaa Ez-Zaidi; asmaa.ez.zaidi@gmail.com

Received 27 May 2015; Revised 25 September 2015; Accepted 28 September 2015

Academic Editor: Wentao Zhang

Copyright © 2016 A. Ez-Zaidi and S. Rakrak. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks have been the subject of intense research in recent years. Sensor nodes are used in wide range of applications such as security, military, and environmental monitoring. One of the most interesting applications in wireless sensor networks is target tracking, which mainly consists in detecting and monitoring the motion of mobile targets. In this paper, we present a comprehensive survey of target tracking approaches. We then analyze them according to several metrics. We also discuss some of the challenges that influence the performance of tracking schemes. In the end, we conduct detailed analysis and comparison between these algorithms and we conclude with some future directions.

1. Introduction

Recent advances in MEMS technology have given rise to low power, cost, and multifunctional sensor nodes [1] which are densely deployed inside a phenomenon or close to it and in charge of monitoring physical phenomena like temperature, humidity, vibrations, and seismic events. Wireless sensor networks (WSNs) can be used in different areas such as precision agriculture [2, 3], military [4], security, environmental control, healthcare [5], intelligent transport [6] for smart parking lots, and accident avoidance.

Target tracking (shown in Figure 1) is one of the most important applications in WSN in which sensor nodes monitor and report the positions of moving objects to the application's user with a minimum latency. In fact, target tracking has many real life applications such as battlefield surveillance, detection of illegal borders crossing, gas leakage, fire spread, and wildlife monitoring.

Target tracking can be performed using a single node or through the collaboration between different sensors. However, using a single node may result in power loss and induces heavy computation burden on that node, while using multiple sensors gives better results in terms of accuracy and energy saving, due to the cooperation between nodes.

Various taxonomies of target tracking algorithms were proposed in the literature. So, there is no standardized or predefined classification. Some works have studied tracking algorithms according to security aspect [7] and others according to energy efficiency [8], fault tolerance, mobility, accuracy, and so on. In this paper, we investigate tracking algorithms from different angles. We also highlight the challenges that can affect the target tracking process.

The rest of the paper is organized as follows. Section 2 discusses target tracking challenges. Section 3 presents the metrics according to which we will evaluate the target tracking algorithms introduced in the paper. We also give a detailed description of each metric. Section 4 reviews the existing approaches of target tracking. Comparison, analysis, and discussion are drawn in Section 5. Section 6 concludes the paper.

2. Localization and Target Tracking Challenges

Many challenges can affect the target tracking quality in wireless sensor networks:

- (i) Node failure: nodes in WSNs are prone to failures due to battery exhaustion, physical disasters, hardware

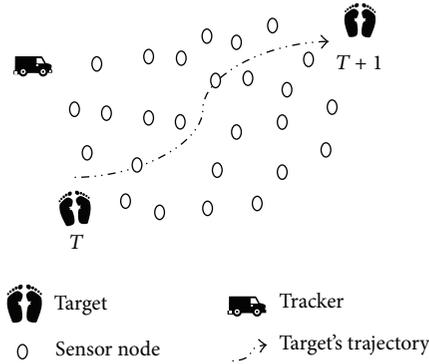


FIGURE 1: Scenario of target tracking in wireless sensor networks.

failures, external attacks, and so on. Consequently, proposed target tracking protocols should cope with these challenges.

- (ii) Target missing and recovery: prediction errors, obstacles, sudden changes in target trajectory, and speed cause loss of target. To tackle this challenge, robust tracking algorithms should be proposed to decrease the probability of missing targets. Moreover, recovery mechanisms have to be considered in case the target was lost.
- (iii) Coverage and connectivity: there is a mutual relationship between coverage and target tracking. High coverage results in high tracking accuracy. However, the performances of target tracking algorithms are degraded in the case of sparse networks, or in the presence of coverage holes.
- (iv) Data aggregation: the aggregation mechanism [9, 10] aims to eliminate data redundancy. In a clustered network, the nodes transfer their sensed data to their associated cluster head, which performs data aggregation and eliminates redundancies. So, the data aggregation should be accomplished in an accurate manner, with minimum data latency while preserving energy.
- (v) Tracking latency: the execution of target tracking algorithms must be performed rapidly while preserving positioning accuracy. When the operation of tracking takes too long, the moving node may change its location.
- (vi) Energy consumption: since sensor nodes run on batteries which are, in most cases, nonrechargeable, energy efficiency is a critical issue in wireless sensor networks especially in sensitive applications like target tracking.

3. Taxonomy of Tracking Methods

Target tracking algorithms were studied in the literature from several angles, so there is no standardized classification. Target tracking can be classified according to different

aspects: security, energy efficiency, network structure, accuracy, mobility of the target, fault tolerance, and so on.

In this section we will discuss the metrics according to which we will evaluate several recent algorithms of target tracking. As shown in Figure 2, there are six metrics: network structure, prediction-based mechanisms, type of chased object, type of sensor, number of targets, and recovery strategies.

3.1. Network Structure. In [11], authors have classified the network structure into three main classes: leader-based, tree-based, and cluster-based protocols. They have combined each one of these protocols with prediction strategies to reduce energy consumption and the probability of losing the target.

Meanwhile in [12] authors have categorized target tracking approaches into two main architectures: hierarchical and peer to peer. The former is composed of five schemes: tree-based, cluster-based, prediction-based, hybrid tracking, and activation methods. The latter includes the embedded filter based consensus and other networks.

We have classified the network structure into three categories: tree structure [13], cluster structure, and face structure.

Tree Structure. In the tree structure, sensor nodes are organized in a hierarchical tree or represented as a graph in which vertices represent sensor nodes and edges are links between nodes that can directly communicate with each other.

Face Structure. It is where the sensor network is divided into faces. Each face contains a set of sensor nodes organized in a ring model. In a face, each member node knows its spatial neighbors and their locations. The face structure is built using the Relative Neighborhood Graph (RNG) algorithm [14].

Cluster Structure. The cluster architecture is used to facilitate the target tracking. It provides a significant impact on the implementation of the object tracking systems. Sensors are organized into clusters, where each cluster contains a cluster head and member nodes. We can distinguish two types of clustering:

- (i) Static clustering: in static clustering, clusters are formed at the time of network deployment and remain unchanged until the end of network lifetime. Despite its simplicity, the static clustering [15] has several negative sides: in case a cluster head dies due to its power depletion, the whole cluster becomes useless. Furthermore, sensor nodes from different static clusters do not have the possibility to collaborate or exchange information.
- (ii) Dynamic clustering: the clusters are formed dynamically as the target moves. The use of dynamic clustering has many benefits. This is due to its flexibility; that is, new clusters are established as the need arises. Furthermore, only one cluster is activated when the mobile target passes by, which implies the mitigation of data redundancy and interference issues.

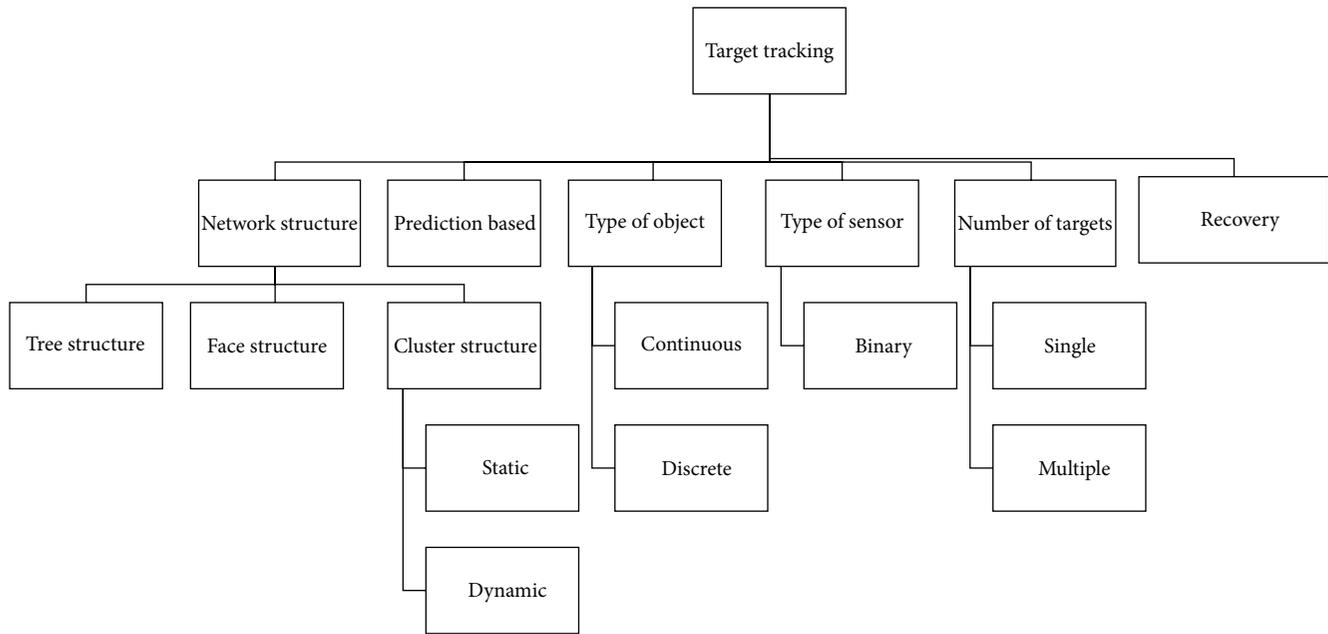


FIGURE 2: Target tracking possible metrics.

3.2. Prediction-Based Tracking. Prediction methods are used to predict the future position of the mobile object. Only sensor nodes located near this position are turned on to detect the target, while the other nodes remain in sleep mode to conserve energy. Prediction techniques are often integrated with face, cluster, or tree structure. Different techniques are used to predict the next position of the target such as kinematics, Kalman filter (KF) [16], extended Kalman filter (EKF) [17], and particle filter (PF) [18].

3.3. Number of Targets. Target tracking approaches are divided into two types: single and multiple target tracking.

- (i) **Single target:** in general, tracking a single object is energy efficient and consumes less power. This is because only a low traffic load is generated in the network while chasing the target.
- (ii) **Multiple targets:** tracking multiple objects is a very challenging task, especially when the number of targets arises. It becomes more complex because of the different directions and speed variations of the tracked objects. In cluttered environment, and in the presence of multiple targets, a sensor node can obtain more than one measurement. It is difficult to know which observation belongs to which target. This uncertainty in measurements results in data association problem.

3.4. Type of Object. Most of the proposed approaches in target tracking aim to track individual (called also discrete) objects such as vehicles, people, and animals. However, few research efforts have been done on tracking continuous objects such as forest fires, gas leakage, biochemical material

diffusion, and oil spills. In contrast to discrete objects, the continuous ones occupy a large area, tend to diffuse, change in shape, increase in size, and even split into smaller objects.

3.5. Type of Sensors. Sensor node can be either binary or ordinary. Binary sensors detect the presence or absence of the target in its sensing range by generating one-bit information (0 or 1). They are low-cost and consume less energy. By using binary outputs, the size of data sent to the base station is reduced.

3.6. Recovery. Prediction algorithms may suffer from localization errors. This is why a recovery mechanism should be included while designing target tracking approaches in order to relocate the missing object.

3.7. Other Metrics. Target tracking can also be studied from the network architecture aspect. We can distinguish two types of architecture.

Centralized. When sensor nodes detect the target, they send the information to the base station. This latter analyzes and processes the received data and then takes the appropriate decisions. In the centralized approaches [19], more messages are exchanged in the network with the base station which consumes large bandwidth and communication overhead.

Distributed. Distributed systems [20, 21] are more robust than the centralized one; this is because sensor nodes are autonomous in decision and collaborate between each other to track the target. Therefore, the traffic overload is reduced and the network lifetime is enhanced.

4. Overview of Target Tracking Approaches in WSN

In this section, we survey the state of the art of some target tracking protocols in wireless sensor networks, and then we will evaluate them according to the metrics defined in Figure 2.

In [22], authors proposed three algorithms for target tracking in WSN using clustering architecture: adaptive head, which is based on dynamic clustering, and static head and selective static head, both of which are based on static clustering. These algorithms enhance the tracking accuracy and energy efficiency by selecting sensor nodes that are nearby the target to participate in the tracking process while the other nodes are put in sleep mode. For each algorithm, a state transition model was introduced in order to describe the behavior of the sensors in the tracking process. Moreover, the three algorithms include a recovery phase in case the target has been lost. For example, in the adaptive head algorithm, the master node broadcasts an alarm message to all network sensors. The node that detects the target in its sensing area responds with a reply message and enters the master election phase, while, in the static and selective static heads algorithms, the message alarm is forwarded from a cluster head to another until finding the object.

The static head algorithm performs worse than the adaptive head in terms of energy efficiency. This goes back to the fact that a large number of control messages are exchanged during each cluster formation. Whenever a target is detected, slave nodes and neighboring cluster heads send their distance readings to the master node which congest the network.

The rate of tracking errors in the adaptive head algorithm increases as the target speed increases. This is due to error accumulations produced when estimating the position of the target as well as the noise added to TOA [23, 24] measurements.

One of the biggest issues in static clustering is the boundary problem. Sensor nodes in different clusters are not able to collaborate or exchange information with each other, which leads to a boundary problem when the mobile target moves along the boundaries of these clusters. To solve this issue, authors in [25] proposed a hybrid cluster-based target tracking (HCTT) for efficient target tracking in a large scale cluster-based WSN. Their approach combines features of static and dynamic clustering. After their deployment, nodes are organized into static clusters, where the members of each cluster cooperate to track the mobile target. Once the target approaches the cluster boundaries, an on-demand dynamic clustering process is triggered to form a dynamic cluster so as to solve the boundary problem (shown in Figure 3). By doing so, nodes from different static clusters can temporarily share information. Finally, when the target moves away, the dynamic cluster is dismissed.

The dynamic clustering employed in HCTT approach induces too much overhead while forming, maintaining, and dismissing the clusters. These operations will consume a lot of energy and lead to the premature death of sensor nodes.

In [26] authors proposed an energy-efficient clustering algorithm for detecting unexpected events in WSNs such

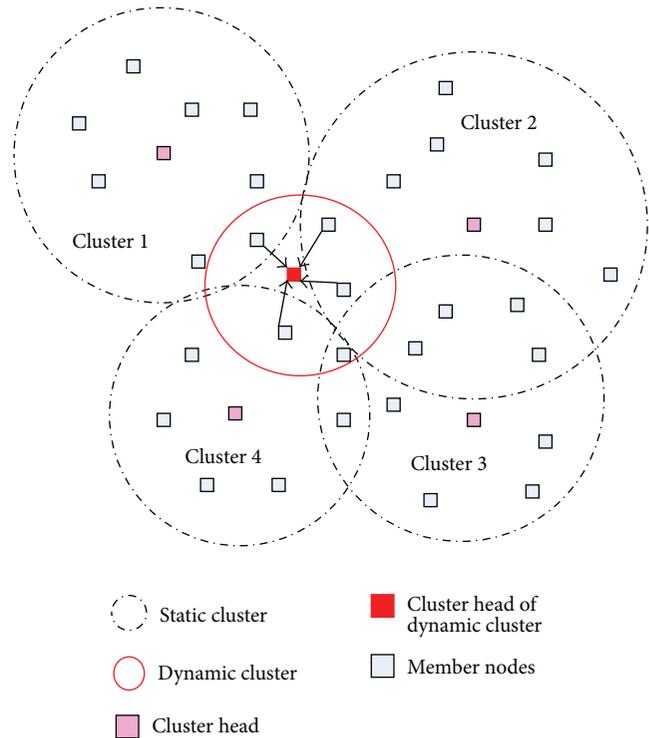


FIGURE 3: Boundary problem.

as fires and intrusions. Their approach consists in creating dynamic clusters repeatedly along the movement of the target using a competition scheme. Sensor nodes having detected an event in the network create their CHEW (Cluster Head Election Window) and start competing to be cluster heads. The CHEW function uses both residual energy and distance from the target as parameters. The node with high energy and closest to the target (i.e., having a small CHEW) wins the competition and creates a cluster for the target. This process is repeated along the movement of the target. Clusters that no longer detect the target are destroyed to maximize energy efficiency as shown in Figure 4. Cluster disconnections have been considered while creating clusters. The approach handles both the single and multiple target tracking.

The dynamic clustering in this approach brings extra overhead. A reclustering process is triggered whenever the target changes its location. Furthermore, each time a new cluster head is elected, it broadcasts advertisement messages to the other sensor nodes to announce its election. As a consequence, these advertisements reduce drastically the network lifetime and its performances.

Upon its movement, the target may enter into a hole [27] and gets temporarily lost. Such a scenario could be critical, especially in sensitive applications such as border crossing and enemy tracking. This is why it is important to integrate a recovery mechanism while designing tracking algorithms. Unfortunately, this approach lacks a recovery mechanism to capture the missed target.

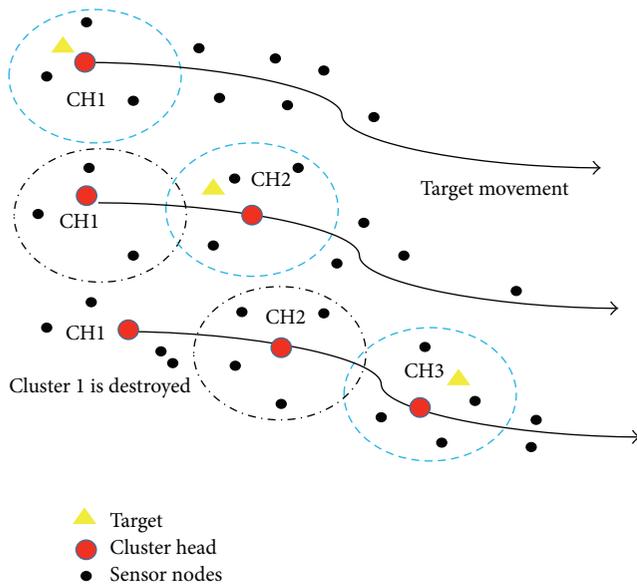


FIGURE 4: The cluster lifetime.

Authors in [28] presented a prediction-based clustering algorithm for target tracking. In their approach, they considered two parameters, distance from predictable location and energy of nodes, to select tracker sensor nodes. Their algorithm is divided into two main phases: clustering and tracking. In the clustering stage, CHs form their clusters by sending hello messages to the other nodes. In the second stage, the CH having detected the target in the network selects three nodes of its members to perform the sensing task. These sensors send their distance from the moving object. Based on these distances, the CH calculates the position of the target using the trilateration [29] algorithm. Then, it predicts its next location and reselects and activates three sensor nodes in the next iteration before the target reaches that location. The selection is made based on the distance to the predicted location and the remaining energy of the node in the network.

This approach consumes more energy, since, in each measurement period, sensor nodes transmit a large amount of packets towards their cluster head.

A variational filtering (VF) approach was proposed in [30] to reduce the interclustering communications and to precisely track the target. A nonmyopic cluster activation rule was introduced to activate the cluster located near the target and reduce the occurrence of the handoff operations. A binary proximity observation model (BPOM) was used to minimize the communication burden by quantifying the detected information to a single bit.

Binary sensors do not provide accurate estimation of the target's location, direction, or velocity. They only generate one-bit information regarding the presence or absence of the target.

Authors in [31] proposed a Face-Based Object Tracking (FOTP) scheme to track mobile objects. They combined a face-based architecture with a hexagon algorithm to achieve high accuracy of tracking and reduce the number of waking faces. By using hexagon algorithm, the target's next position

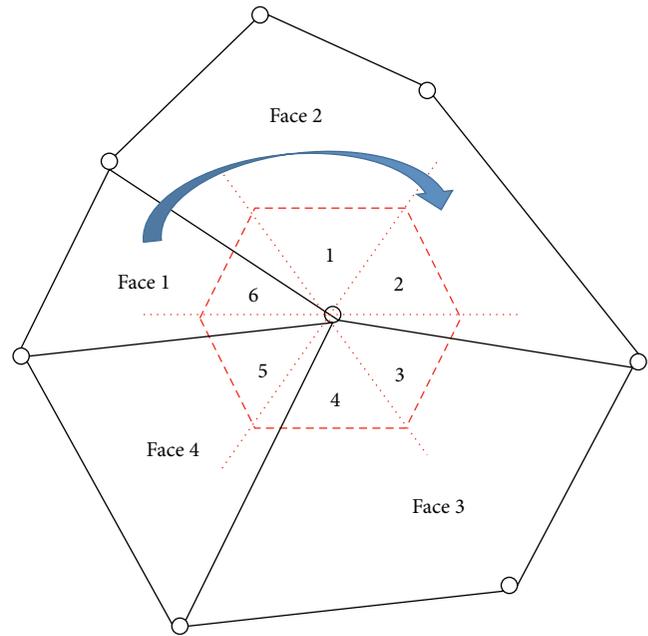


FIGURE 5: FOTP approach.

is reduced to 1/6 of face area in comparison with the other works (shown in Figure 5).

In FOTP approach, when the target gets lost and could not be detected in a predefined time T , all faces in the network are activated to capture the object. As a result, all sensor nodes are turned on, which leads to their battery wastage.

Hsu et al. have proposed a Prediction-Based Optimistic Object Tracking (POOT) strategy [32] that aims to track efficiently a moving target. POOT adopts a face routing and a linear mobility prediction to predict the next face that will be visited by the target. Authors also proposed two schemes, a Time-Efficient Object Recovery Scheme (TORS) and a Communication-Efficient Object Recovery Scheme (CORS), to improve object recovery. The former is designed to find a lost object with the least amount of elapsed time. It consists in waking up all sensor nodes within a circular area that is centered on the lost position and its radius is equal to the distance travelled by the object moving at a maximum speed. The latter uses the least amount of communication to find the lost object using its direction of movement and distance. CORS searches in a sequential manner for the lost object based on the probability of it being in certain faces.

CORS algorithm saves more power in comparison with TORS. This is because CORS activates only faces where the missed target most possibly resides. Such behavior reduces the amount of communication but affects the recovery accuracy since only few nodes are activated to track the object. On the contrary, in TORS, more sensor nodes are involved in the recovery process, which implies that more energy is dissipated and high accuracy is provided.

A prediction-based energy-efficient target tracking protocol (PET) was proposed in [33] to accurately predict the future location of the target, using the two-dimensional Gaussian distribution. In the authors' approach, the network

was configured in face structures obtained by planarization technique to reduce the amount of active faces and thus maximize the lifetime of the sensor network. Moreover, a wakeup mechanism was introduced to activate in advance the appropriate sensor nodes located near the target's predicted position in order to save energy.

PET scheme suffers from a high missing rate when the node's sensing range is small. By way of explanation, the smaller sensing range is, the more uncovered zones appear in the network and the higher probability of losing the object is.

Authors in [34] proposed an object tracking scheme based on prediction mechanisms. The process of object tracking in this scheme is as follows: initially, only nodes at the border of the network are activated in order to detect an intruder. Once the target is found, all the faces that are constructed using the Relative Neighborhood Graph (RNG) algorithm are set to sleeping mode, except the one where the object was discovered in. The face containing the object records the coordinates of this latter and calls the prediction function LSM (least square method) [35] to predict the path of the target. Based on this prediction, network activates only the face to which the object is intended to move.

In this approach, the frequency at which the prediction mechanism is invoked depends on the object's movement. When the object moves at high speed, the prediction algorithm is called several times to estimate the target's position. The problem with this scheme is that it uses a very complex prediction algorithm that consumes larger energy.

A Dynamic Lookahead Spanning Tree Algorithm (DLSTA) was proposed in [36] to decrease the target miss ratio by preconstructing lookahead clusters along the predicted trajectory of the target before its arrival. In DLSTA, the node closest to the target is elected as a root node. The latter constructs its clustered tree and estimates the speed, location, and direction of the target. The more the object's velocity increases the more the preconstructed trees increase. Different prediction techniques, such as linear, extended Kalman filter (EKF), and particle filters (PF), were used in the DLSTA approach to raise the prediction accuracy.

The object tracking tree performance is affected when the movement behaviors of the object are different from the mobility profile already predefined. To tackle this issue, authors in [37] proposed the message tree adaptive (MTA) procedure to dynamically adapt the object tracking tree. The MTA procedure is triggered whenever a sensor node detects that the movement behaviors of the object do not match the predefined mobility profile. MTA reconstructs the tree using either TAP or DAB [38] approaches and corrects the events rate. Because of the high cost of the MTA procedure, the reconstruction of the tree is not performed all the time, so authors have determined a ratio of change: when this ratio is under a threshold value, the MTA procedure is not necessary to perform.

MTA procedure incurs very high overhead to enhance the update cost of the object tracking tree. The situation gets worse when the size of the tree increases, which means that more adaptive messages will be transmitted to the sink along the object tracking tree.

Probability-Based Target Prediction and Sleep Scheduling (PPSS) protocol [39] was introduced to improve the efficiency of proactive wakeup. The PPSS proposes a prediction scheme based on kinematics rules and theory of probability to determine the expected displacement of the target. Based on the prediction results, PPSS schedules awakened nodes by controlling their time activity to enhance the energy efficiency.

PPSS assumes that the object moves in a smooth curvilinear trajectory. However, this approach suffers from a main limitation which is the incapacity of handling abrupt changes in target direction. As a result, PPSS's performances are degraded since it will make a long delay to compute the target's position.

Authors in [40] proposed a minimal contour tracking algorithm (MCTA) to improve the energy efficiency by reducing the number of working nodes. The tracking contour is constructed based on the vehicular kinematics to exclude the most unlikely areas where the target cannot traverse within a limited time.

However, MCTA keeps all sensor nodes in the contour turned on without using a sleep scheduling mechanism to alternate between active and sleep modes. As a consequence, more sensor nodes are awakened which means more energy is consumed.

Since MCTA focuses on vehicle's kinematics, it seems obvious that it will fail to track the other types of targets like tanks, unmanned aerial vehicles (UAV), and so on.

Authors in [41] used the concept of kinematics in order to reduce the tracking area so that less number of sensors will be active, which prolongs the network lifetime. In their approach, the whole tracking area is mapped to Voronoi polygons [42] and contains three categories of sensor nodes, that is, worker nodes, border nodes, and computational ones. Authors studied how overlapped polygons could influence the performance of tracking multiple targets. This situation occurs when more than one target comes close. Larger overlapped area between polygon leads to smaller number of working nodes in the polygon.

Chen et al. [43] proposed a practical distributed sensor activation algorithm (DSA2) for reliable tracking using binary-detection sensors to catch mobile targets with large acceleration. The concept of the algorithm consists in activating each sensor node with a probability of detecting targets or sleeping to reduce energy dissipation according to the behaviors of its one-hop neighboring nodes.

DSA2 is not efficient when it is about recapturing the target after it gets lost. The missed object is located by sensor nodes that happen to be in the sensing mode.

Data association (shown in Figure 6) is a big challenge in multiple target tracking. It consists in finding the correct correspondences between measurements and their origins. In a cluttered environment, sensor node can obtain wrong measurements which leads to target losses. To solve the problem of data association, Parmar and Zaveri [44] used the Munkres algorithm.

Authors in [45] addressed the problem of tracking a group of targets having a correlated motion using binary sensor nodes. Two algorithms were proposed to estimate

TABLE 1: Classification of target tracking approaches.

	[47]	[25]	[32]	[26]	[33]
Network structure					
Face	—	—	✓	—	—
Tree	—	—	—	—	—
Cluster					
S: static	S&D	S&D	—	D	—
D: dynamic					
Prediction	—	—	Linear mobility	—	Two-dimensional Gaussian distribution
Type of target					
Continuous	✓	—	—	—	—
Discrete	—	✓	✓	✓	✓
Number of targets					
Single	—	✓	✓	✓	✓
Multiple	—	—	—	✓	—
Binary sensor	—	—	—	—	—
Recovery	—	—	TORS & CORS	—	✓

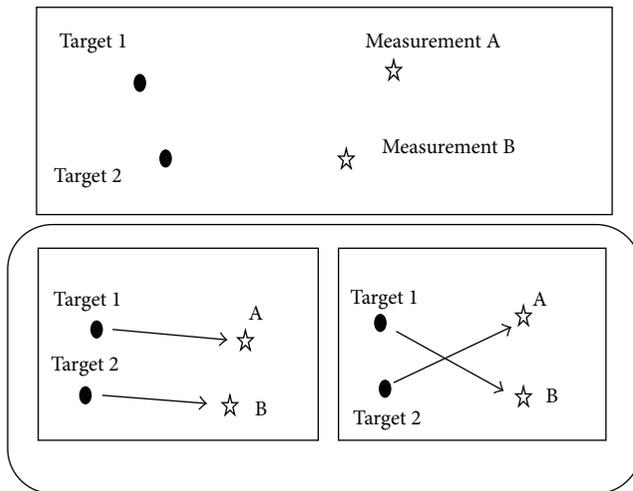


FIGURE 6: Data association problem.

the trajectory of target group: Hull and Cir algorithms. The former, which is very accurate, computes the convex hull using the Graham scan algorithm; however it is complex and expensive. The latter creates a circle covering the convex hull; nevertheless it is not accurate. A reporter node selection algorithm was also proposed to select the node closest to the plus sensors in order to reduce the energy consumption. A fault tolerant algorithm target detection was introduced to handle the problem of measurement errors that occurs in the network.

This approach focused on tracking a group of targets that move collectively and have correlated motion, speed, and direction. It is exactly the case of a group of wild animals that migrates from one habitat to another. The problem with this scheme is that it does not handle scenarios where these sets of targets (animals) split, merge, or even overlap.

A Collaborative Boundary Detection and Tracking of Continuous Objects in WSNs was proposed in [46] to detect and monitor the inner and outer boundaries of the tracked object. It adopts local communication between nodes located near the holes or edges of the phenomenon in order to find boundary nodes. An interpolation algorithm is used to find the boundary points and thus obtain an accurate boundary shape.

CODA [47] was proposed to detect and track continuous objects. It is a combination between static and dynamic clustering. The CODA scheme is divided into 5 steps: in the first step the whole network is divided into static clusters. The cluster head determines sensor nodes located in the boundary of its cluster by resolving the convex-hull problem using the Graham scan algorithm. The second step consists in distinguishing the object boundary within each static cluster. Four scenarios were depicted depending on the number of static clusters in which the object spreads. In the third step, each CH organizes the boundary sensors into a dynamic cluster, fuses data, and then relays it to the sink. The last step is boundary tracking; it consists in updating boundary sensors each time the object moves away.

When sensor nodes detect the presence of a target, they transmit their detection information to their immediate cluster head. Based on this information, the CH identifies the boundary sensors within its cluster. This process will cost additional energy, especially when the target overlays a large number of nodes.

5. Comparison and Analysis

A comparison between different target tracking approaches is drawn in Tables 1, 2, 3, and 4. We analyze them according to the metrics cited in Section 3.

Balancing the tradeoff between energy conservation and tracking accuracy is a big challenge. The more sensors are involved in tracking target, the more accurate the tracking is.

TABLE 2: Classification of target tracking approaches.

	[28]	[39]	[31]	[43]	[36]
Network structure					
Face	—	—	✓	—	—
Tree	—	—	—	—	✓
Cluster					
S: static	S	—	—	—	D
D: dynamic					
Prediction	✓	Kinematics & theory of probability		✓	—
Type of target					Linear, EKF, and PF
Continuous	—	—	—	—	—
Discrete	✓	✓	✓	✓	✓
Number of targets					
Single	✓	✓	✓	—	✓
Multiple	—	—	—	✓	—
Type of sensors (binary)	—	—	—	✓	—
Recovery	—	—	✓	—	✓

TABLE 3: Classification of target tracking approaches.

	[34]	[37]	[40]	[22]
Network structure				
Face	✓	—	—	—
Tree	—	Tree combined with Vornoi Graph		—
Cluster				
S: static	—	—	—	S&D
D: dynamic				
Prediction	LSM least square	—	Kinematics	—
Type of target				
Continuous	—	—	—	—
Discrete	✓	✓	✓	✓
Number of targets				
Single	✓	✓	—	✓
Multiple	—	—	✓	—
Type of sensors (binary)	—	—	—	—
Recovery	✓	—	—	✓

TABLE 4: Classification of target tracking approaches.

	[45]	[44]	[41]	[30]	[46]
Network structure					
Face	—	—		—	—
Tree	—	—	Uses Voronoi polygons		—
Cluster	—	S		S	—
Prediction	—	KF	Kinematics		—
Type of target					
Continuous	—	—	—	—	✓
Discrete	✓	✓	✓	✓	—
Number of targets					
Single	—	—	—	✓	—
Multiple	✓	✓	✓	—	—
Type of sensors (binary)	✓	—	—	✓	—
Recovery	—	—	—	—	—

However the use of a large number of sensor nodes in the tracking process causes high energy dissipation.

Most of the approaches introduced in this paper try to minimize the energy consumption while enhancing the tracking accuracy by using several techniques such as clustering, prediction of target's movement, adopting specific network structures, and utilizing binary sensors.

There are several reasons that hamper the tracking and lead to target loss:

- (i) Communication failures, presence of obstacles, and coverage holes.
- (ii) Abrupt change in target's velocity and direction.
- (iii) Sensor nodes driven by batteries and having low-energy resources (when they remain in active mode for a long period, they deplete their energy faster causing the so-called "energy hole" problem [48]).
- (iv) Inaccuracy while estimating the target's actual position.
- (v) The long delay that a predicted algorithm may take to compute the future position of the target.

Unfortunately, HCTT, MTA, [25, 47], and many other approaches may experience target loss due to the reasons stated above. These algorithms do not integrate recovery mechanisms to capture the missing object. They are not practical especially in sensitive applications such as enemy tracking or terrorist tracking.

In terms of energy efficiency, PPSS, MCTA, and [41] perform better because they use kinematics to predict the next location of the moving target. They eliminate areas that the object cannot reach within a specific time. As a result, the tracking area is reduced and thus the number of awakened sensor nodes is minimized.

Tree-based approaches are a good choice for target tracking. However, they induce a high overhead cost. In tree architecture, lots of messages are exchanged in the network from leaf nodes toward the sink. This situation becomes more complicated, especially when trees are large. Furthermore, sensor nodes located near the root consume much energy in comparison with the other nodes. For instance, in the MTA scheme, many messages are exchanged between the sink (root node) and leaf nodes along the tree, causing a high overhead. MTA approach is more practical when the size of the network is small.

FOTP, POOT, and [34] are more accurate because they combine face structure with prediction algorithms as well as recovery mechanism. In face structure, nodes inside the same face cooperate to track the target which increases the tracking accuracy. In addition, using prediction algorithms coupled with recovery mechanisms decreases the probability of losing targets; hence the accuracy is improved.

The VF approach achieves high energy efficiency since it uses binary sensor nodes in tracking. These so-called binary sensors generate one bit of information regarding the presence or absence of the target in their sensing range which reduce the amount of data transmitted in the network. However, this type of sensors cannot provide enough information about the chased target (velocity, direction, etc.).

DLSTA scheme consumes less energy and provides accurate tracking because it considers targets that move at high speed and make nonlinear and linear motions. In contrary to the other approaches, it employs three types of prediction techniques, linear, PF, and EKF, which make the DLSTA a strong approach. Moreover, since DLSTA have a smaller miss ratio, the recovery algorithm is executed rarely and thus more power is saved. By contrast, the recovery process in the adaptive head [22] involves a large number of sensor nodes to capture the missed target. Therefore, more energy is consumed; as a consequence the network lifetime is shortened. CORS algorithm [32] estimates the potential faces where the missed object could reside. However in large networks, this recovery process may take a long time to retrieve the exact location of the target.

Much additional research effort will be needed

- (i) to handle the problem of data association in multiple target tracking,
- (ii) to treat the problem of inaccurate measurements and false alarms that may occur in the network,
- (iii) to track mobile objects in presence of obstacles,
- (iv) since sensor nodes are prone to failures due to energy depletion, communication errors, and malicious attacks. More research efforts should be done to deal with fault tolerance issue in target tracking. None of the protocols listed in the tables below have considered it except the approach in [45].

Tables 1, 2, 3, and 4 summarize the classification of approaches introduced in this paper according to the metrics presented above.

6. Conclusion

Target tracking has gained considerable attention in recent years for its application in different fields such as military, civilian, and wildlife monitoring. The massive research in this field has inspired us to present this survey. In this paper, we investigated some of the target tracking algorithms currently used in wireless sensor networks. We compared and analyzed them from different angles. It is clear that all protocols presented in this paper share one common objective, which is ensuring a high target tracking accuracy while maintaining the energy.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] K. Yang, *Wireless Sensor Networks*, Springer, Berlin, Germany, 2014.
- [2] M. Srbinska, C. Gavrovski, V. Dimcev, A. Krkoleva, and V. Borozan, "Environmental parameters monitoring in precision agriculture using wireless sensor networks," *Journal of Cleaner Production*, vol. 88, pp. 297–307, 2015.

- [3] H. Sahota, R. Kumar, and A. Kamal, "A wireless sensor network for precision agriculture and its performance," *Wireless Communications and Mobile Computing*, vol. 11, no. 12, pp. 1628–1645, 2011.
- [4] M. P. Durišić, Z. Tafa, G. Dimić, and V. Milutinović, "A survey of military applications of wireless sensor networks," in *Proceedings of the 1st Mediterranean Conference on Embedded Computing (MECO '12)*, pp. 196–199, IEEE, June 2012.
- [5] M. Al Ameen, J. Liu, and K. Kwak, "Security and privacy issues in wireless sensor networks for healthcare applications," *Journal of Medical Systems*, vol. 36, no. 1, pp. 93–101, 2012.
- [6] M. Tubaishat, P. Zhuang, Q. Qi, and Y. Shang, "Wireless sensor networks in intelligent transportation systems," *Wireless Communications and Mobile Computing*, vol. 9, no. 3, pp. 287–302, 2009.
- [7] A. Oracevic and S. Ozdemir, "A survey of secure target tracking algorithms for wireless sensor networks," in *Proceedings of the World Congress on Computer Applications and Information Systems (WCCAIS '14)*, pp. 1–6, IEEE, Hammamet, Tunisia, January 2014.
- [8] O. Demigha, W.-K. Hidouci, and T. Ahmed, "On energy efficiency in collaborative target tracking in wireless sensor network: a review," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1210–1222, 2013.
- [9] M. Bagaa, Y. Challal, A. Ksentini, A. Derhab, and N. Badache, "Data aggregation scheduling algorithms in wireless sensor networks: solutions and challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1339–1368, 2014.
- [10] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, "In-network aggregation techniques for wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 14, no. 2, pp. 70–87, 2007.
- [11] M. Naderan, M. Dehghan, and H. Pedram, "Mobile object tracking techniques in wireless sensor networks," in *Proceedings of the International Conference on Ultra Modern Telecommunications & Workshops*, pp. 1–8, IEEE, St. Petersburg, Russia, October 2009.
- [12] K. Ramya, K. Kumar, and V. Rao, "A survey on target tracking techniques in wireless sensor networks," *International Journal of Computer Science and Engineering Survey*, vol. 3, no. 4, pp. 93–108, 2012.
- [13] C.-Y. Lin, W.-C. Peng, and Y.-C. Tseng, "Efficient in-network moving object tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 1044–1056, 2006.
- [14] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.
- [15] X. R. Zhu, L. F. Shen, and T.-S. P. Yum, "Hausdorff clustering and minimum energy routing for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 990–997, 2009.
- [16] M. Di, E. M. Joo, and L. H. Beng, "A comprehensive study of kalman filter and extended kalman filter for target tracking in wireless sensor networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '08)*, pp. 2792–2797, Singapore, October 2008.
- [17] M. Di, E. M. Joo, and L. H. Beng, "A comprehensive study of kalman filter and extended kalman filter for target tracking in wireless sensor networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '08)*, pp. 2792–2797, IEEE, Singapore, October 2008.
- [18] A. Smith, A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer Science & Business Media, 2013.
- [19] J. Singh, R. Kumar, U. Madhow, S. Suri, and R. Cagley, "Multiple-target tracking with binary proximity sensors," *ACM Transactions on Sensor Networks*, vol. 8, no. 1, pp. 1–26, 2011.
- [20] R. Rahman, M. Alanyali, and V. Saligrama, "Distributed tracking in multihop sensor networks with communication delays," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4656–4668, 2007.
- [21] H. Chen and K. Sezaki, "Distributed target tracking algorithm for wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '11)*, pp. 1–5, IEEE, Kyoto, Japan, June 2011.
- [22] K. A. Darabkh, S. S. Ismail, M. Al-Shurman, I. F. Jafar, E. Alkhalder, and M. F. Al-Mistarihi, "Performance evaluation of selective and adaptive heads clustering algorithms over wireless sensor networks," *Journal of Network & Computer Applications*, vol. 35, no. 6, pp. 2068–2080, 2012.
- [23] L. Cheng, C. Wu, Y. Zhang, H. Wu, M. Li, and C. Maple, "A survey of localization in wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 962523, 12 pages, 2012.
- [24] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Localization systems for wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 6–12, 2007.
- [25] Z. Wang, W. Lou, Z. Wang, J. Ma, and H. Chen, "A hybrid cluster-based target tracking protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 494863, 16 pages, 2013.
- [26] J.-I. Kong, J.-W. Kim, and D.-S. Eom, "Energy-aware distributed clustering algorithm for improving network performance in WSNs," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 670962, 10 pages, 2014.
- [27] J. Kanno, J. G. Buchart, R. R. Selmic, and V. Phoha, "Detecting coverage holes in wireless sensor networks," in *Proceedings of the IEEE 17th Mediterranean Conference on Control and Automation (MED '09)*, pp. 452–457, Thessaloniki, Greece, June 2009.
- [28] F. Deldar and M. H. Yaghmaee, "Designing a prediction-based clustering algorithm for target tracking in wireless sensor networks," in *Proceedings of the International Symposium on Computer Networks and Distributed Systems (CNDS '11)*, pp. 199–203, IEEE, Tehran, Iran, February 2011.
- [29] M.-S. Huang and R. M. Narayanan, "Trilateration-based localization algorithm using the lemoine point formulation," *IETE Journal of Research*, vol. 60, no. 1, pp. 60–73, 2014.
- [30] J. Teng, H. Snoussi, and C. Richard, "Decentralized variational filtering for target tracking in binary sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 10, pp. 1465–1477, 2010.
- [31] X. Ji, Y.-Y. Zhang, S. Hussain, D.-X. Jin, E.-M. Lee, and M.-S. Park, "FOTP: face-based object tracking protocol in wireless sensor network," in *Proceedings of the 4th International Conference on Computer Sciences and Convergence Information Technology*, pp. 128–133, IEEE, November 2009.
- [32] J. M. Hsu, C. C. Chen, and C. C. Li, "Short-term prediction-based optimistic object tracking strategy in wireless sensor networks," in *Proceedings of the 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 78–85, IEEE, July 2011.

- [33] M. Z. A. Bhuiyan, G.-J. Wang, L. Zhang, and Y. Peng, "Prediction-based energy-efficient target tracking protocol in wireless sensor networks," *Journal of Central South University of Technology*, vol. 17, no. 2, pp. 340–348, 2010.
- [34] Y. Shen, K. T. Kim, J. C. Park, and H. Y. Youn, "Object tracking based on the prediction of trajectory in wireless sensor networks," in *Proceedings of the IEEE 10th International Conference on High Performance Computing and Communications & IEEE International Conference on Embedded and Ubiquitous Computing (HPCC-EUC '13)*, pp. 2317–2324, Zhangjiajie, China, November 2013.
- [35] J. Wolberg, *Data Analysis Using the Method of Least Squares: Extracting the Most Information from Experiments*, Springer, 2006.
- [36] A. Alaybeyoglu, A. Kantarci, and K. Erciyes, "A dynamic lookahead tree based tracking algorithm for wireless sensor networks using particle filtering technique," *Computers and Electrical Engineering*, vol. 40, no. 2, pp. 374–383, 2014.
- [37] M.-X. Chen, C.-C. Hu, and W.-Y. Weng, "Dynamic object tracking tree in wireless sensor network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, Article ID 386319, 2010.
- [38] H. T. Kung and D. Vlah, "Efficient location tracking using sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '03)*, vol. 3, pp. 1954–1961, New Orleans, La, USA, March 2003.
- [39] B. Jiang, B. Ravindran, and H. Cho, "Probability-based prediction and sleep scheduling for energy-efficient target tracking in sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 735–747, 2013.
- [40] J. Jeong, T. Hwang, T. He, and D. Du, "MCTA: target tracking algorithm based on minimal contour in wireless sensor networks," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 2371–2375, IEEE, Anchorage, Alaska, USA, May 2007.
- [41] A. A. U. Rahman, M. Naznin, and M. A. I. Mollah, "Energy-efficient multiple targets tracking using target kinematics in wireless sensor networks," in *Proceedings of the 4th International Conference on Sensor Technologies and Applications (SENSORCOMM '10)*, pp. 275–280, Venice, Italy, July 2010.
- [42] W. Alsalih, K. Islam, Y. Rodríguez, and H. Xiao, "Distributed voronoi diagram computation in wireless sensor networks," in *Proceedings of the 20th Annual Symposium on Parallelism in Algorithms and Architectures (SPAA '08)*, p. 364, ACM, Munich, Germany, June 2008.
- [43] J. Chen, K. Cao, K. Li, and Y. Sun, "Distributed sensor activation algorithm for target tracking with binary sensor networks," *Cluster Computing*, vol. 14, no. 1, pp. 55–64, 2011.
- [44] P. Parmar and M. Zaveri, "Multiple target tracking and data association in wireless sensor network," in *Proceedings of the 4th International Conference on Computational Intelligence and Communication Networks (CICN '12)*, pp. 158–163, Mathura, India, November 2012.
- [45] D. Cao, B. Jin, S. K. Das, and J. Cao, "On collaborative tracking of a target group using binary proximity sensors," *Journal of Parallel and Distributed Computing*, vol. 70, no. 8, pp. 825–838, 2010.
- [46] C. S. Hussain, M.-S. Park, A. K. Bashir, S. C. Shah, and J. Lee, "A collaborative scheme for boundary detection and tracking of continuous objects in WSNs," *Intelligent Automation and Soft Computing*, vol. 19, no. 3, pp. 439–456, 2013.
- [47] W.-R. Chang, H.-T. Lin, and Z.-Z. Cheng, "CODA: a continuous object detection and tracking algorithm for wireless ad hoc sensor networks," in *Proceedings of the 5th IEEE Consumer Communications and Networking Conference (CCNC '08)*, pp. 168–174, Las Vegas, Nev, USA, January 2008.
- [48] G. Ma and Z. Tao, "A nonuniform sensor distribution strategy for avoiding energy holes in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 564386, 14 pages, 2013.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

