Hindawi Publishing Corporation
Journal of Computer Networks and Communications
Volume 2016, Article ID 7631548, 10 pages
http://dx.doi.org/10.1155/2016/7631548

Research Article

Dynamic Clustering Algorithm for Tracking Targets with High and Variable Celerity (ATHVC)

Mohamed Toumi, Abderrahim Maizate, Mohammed Ouzzif, and Med Said Salah

RITM-ESTC/CED-ENSEM, University Hassan II, Casablanca, Morocco

Correspondence should be addressed to Mohamed Toumi; m_toumy@yahoo.fr

Received 20 July 2016; Accepted 23 October 2016

Academic Editor: Yun Liu

Target tracking with the wireless sensors networks is to detect and locate a target on its entire path through a region of interest. This application arouses interest in the world of research for its many fields of use. Wireless sensor networks, thanks to their versatility, can be used in many hostile and inaccessible to humans environments. However, with a limited energy, they cannot remain permanently active, which can significantly reduce their lifetime. The formation of a cluster network seems an effective mechanism to increase network lifetime. We propose to build optimal dynamic clusters on the target trajectory. For increasing energy efficiency, our algorithm integrates for the first time, to our knowledge, strategies to avoid overlapping clusters and a model to wake up the sensors, adapting to the context of targets with large and variable speed.

1. Introduction

There are many areas of use of the wireless sensor networks; we can mention military applications, environmental and industrial monitoring, applications related to smart cities, and others. We are interested in monitoring applications and in particular the tracking of moving targets. Target tracking is to perform two main functions: the detection and the monitoring of the target along its path through sensors deployed in an area of interest. However, the deployment of these networks depends on the constraints imposed by the miniaturized structure of the sensors: small storage capacity, limited life of the battery, communication range, bandwidth, and so forth. Particularly, as in most cases, the nodes are not serviceable after their deployment. Forming a clustered network seems an effective mechanism to increase network's lifetime. Many researchers focused on the development of efficient energy clustering algorithms [1–5]; however, these algorithms are not suitable for applications specifications related to the targets tracking. Indeed, the cited protocols offer clustering schemes that form and maintain a hierarchical network without considering the fact that the target itself seems rare and in specific places in the network. The optimization of the clustering algorithms consists in activating only the nodes which are in the path of the target when it is within their detection ranges. All other sensors must be in sleep mode. Numerous proposals have been published in recent years [6–10] which propose dynamic clustering algorithms; these form temporal clusters depending on the evolution of the target across the network. However, despite the efficiency of their energy, these algorithms are not adapted to an environment where the velocity of the target can become extremely large and variable. Tracking, for example, the tsunami waves whose velocity $V \approx 870\sqrt{(h * 6)}$ is a function of the single parameter water depth h (km) in the case of sufficiently long period of tsunamis, typically about ten minutes, which is the case of most tsunamis of tectonic origin. This means that the speed is 940 km/h for a depth of 7 km and 158 km/h for a depth of 200 m. The variability of the speed of a tsunami is clearly identifiable with the approach of the coast. Tracking wave of an earthquake or a wildfire driven by the torrential winds are examples of applications where speed of the target is important and variable. In this case, the target can be moved, while the cluster configuration's messages are just in the neighbor discovery phase and sharing information. The time required for the selection of cluster heads, based on different metrics and recruitment of member nodes, can be detrimental to ensure the function of tracking for this type of applications. We propose an efficient algorithm that is well suited to target tracking applications with significant and variable speed; in our algorithm, we use a new metric for forming optimal clusters according to the evolution of the target, with minimal time waiting and clusters overlap, which allows for better energy efficiency and especially having a minimum response time needed to track high-speed targets. The selection of cluster head is not the only problem to be raised in the dynamic clustering algorithms. Indeed, the nodes are a priori in the sleep state; only the sensors that are on the target path must be active. So how and by what criteria should we activate these sensors to form the active cluster to retrieve the data on the target to a base station?

The prediction schemes have been proposed in recent years to predict the position of the target which enables activating only the nodes which are on the trajectory of the target. The extended Kalman filter [11] combined with detection mechanisms for changes of direction as CuSum [12] can effectively calculate future coordinates of the target and wake up the sensors accordingly. The prediction filters require message exchanges and sometimes complex calculations by a central entity, which consumes a lot of time, particularly as more often these algorithms use a correction step and lead to additional calculations and thus a precious time is wasted which is necessary to effectively meet the real time constraint imposed by high-speed target. Our prediction system must necessarily take account of this constraint. We propose to use an activation based on overhearing the members of the active cluster to wake the other nodes on the way to the target. The sensors that are awakened must return to the sleep state once they no longer detect the target after a system time Δt. So the question is how to properly adjust this time in the context of variable speed targets. Poor timing could create what is called "the reflected wake waves." Although several solutions using this method of activation exist in the literature [13], our solution is, to our knowledge, the first to consider a prediction scheme offering two effective solutions to prevent the problem of reflected wake waves described in Section 3 and to minimize the overlap of the generated clusters during the movement of the target.

This article is organized as follows: in Section 2, we present some related work on tracking of the targets. Then, in Section 3, we will detail the proposed protocol. In Section 4, we present the simulation results. Finally, we end this article with a conclusion in Section 5.

2. Related Work

An important number of researches concerning tracking moving targets in the context of wireless sensor networks with the use of clustered network architectures exist in the literature. This type of architecture provides advantages such as scalability, traffic reduction, and energy efficiency. These works can be classified into three main categories:

(i) Static clustered architectures

(ii) Dynamic clustered architectures

(iii) Hybrid clustered architectures

2.1. Static Clusters. Algorithms [1–5] are trying to statically create clusters at the time of the network deployment based on different criteria for the CH (cluster head) selection, such as the position, density, security, computing power, or the residual energy of the sensors. However, these static clustering algorithms are not suitable for targets tracking, because they hold all the nodes in activities, even if the target appears only rarely. It is inefficient to create groups of the nodes in advance regardless of the target's movement in the sensor field; this can waste inefficiently the sensors' energy.

2.2. The Dynamic Clusters. In this category of algorithms, the target's trajectory is predicted, allowing only the activation of the sensors in the path and thus saving energy. This prediction can be performed using predictive models including the Kalman filters [11] or using probabilistic mechanisms such as Markov chains [14].

The authors of [7] presented the prediction protocol and sleep scheduling nodes based on the probability (SSPP) to improve energy efficiency. The approach provides a target prediction method based on kinematics and probability.

An approach is proposed in [8] to awaken the sensors that form clusters along the planned trajectory to reduce the probability of missing the target. The solution provides that active cluster members identify the target and send the data to their cluster head. The base station collects all the data members, determines the possible location of the target, and sends wake-up messages to the nodes in the path of the target.

In [15], there is a provided dynamic clustering algorithm coupled with the Kalman filter to predict the position of the single moving target. The Kalman filter is a prediction model with two stages: prediction and correction. It allows estimating recursively the process status based on its earlier statements; it aims to estimate the future target position based on the current position. It is described using a state evolution model and a measurement model that assumes linear with Gaussian errors.

The authors of [13] propose a distributed algorithm for constructing clusters dynamically and measuring the displacement of the target. Sensors that detect the target enter a regional competition regime. Literally, every such candidate must calculate a parameter called CHEW. The node with the lowest value will start the operation of the recruitment of member's knots. The equation of CHEW parameter is defined by

$$\text{CHEW} = \text{INT}\left(\frac{D_{\text{target}}}{E_{\text{batry}}}\right) + \text{random}(). \quad (1)$$

The authors want to promote the node with a high residual energy and that closest to the target. The second part decimal random() is a random value between 0 and 1 whose role is to prevent the collision of data transmission when two or more nodes define the same window size.

CHEW creates an optimal clustering scheme depending on the target of the movement as shown in Figure 1 and selects from its neighboring sensors that participate in the monitoring of the target. The predictive solutions have the advantage

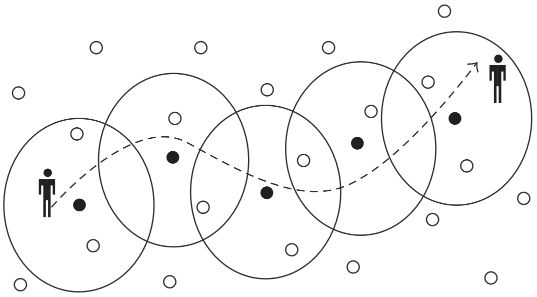


Figure 1: Diagram clustering of CHEW.

to exploit with the best way the available information to save energy by activating only nodes that are in the path of the target. However, they remain unsuitable to the contexts of the target with very large and variable speeds. Indeed, the predictions calculation time can be disadvantageous to respond with a real time for tracking applications associated with a great target speed.

2.3. Hybrid Solution. As its name suggests, this category includes solutions with several combined approaches. For example, in [16, 17], the authors propose algorithms hybrid clustering in which dynamic reactive clusters are formed in collaboration with static proactive clusters. The major objective of this research is to continue to monitor targets in the cluster border regions with energy efficiency; however, these algorithms still suffer from energy inefficiency due to having a priori static cluster structure.

In our work, we have offered a global vision by focusing on collaboration between nodes, including how to wake them up one after the other to follow the target throughout its evolution in the area of interest, and managing at best the energy consumption by minimizing the overlap between the clusters and avoiding the problem of the reflected wake waves; then we have proposed a metric for selecting clusters head adapting to the context of large and variable speed targets. The next paragraphs will be applied to describe in detail this solution.

3. The Proposed Algorithm

3.1. The System Model and the Assumptions. We assume that nodes are initially in the state of sleep which guarantees minimal energy consumption. Indeed, in this state, all equipment units, which make up the sensor, are off, except for a processing unit and a low power channel for receiving the activation-up messages. Upon receiving a wake-up message, each node has to start all these hardware units.

It is also assumed that the nodes have knowledge of their geographical positions and the first target detection is done. This task is beyond the scope of this project.

In general, the signal received by node N_i from node N_j is decreased with the distance between the two nodes.

We are adopting the mitigated disk detection model [17] to estimate the distance $d(N_i, N_j)$ from the received signal. The model equation is written as follows:

$$r_i = \begin{cases} \dfrac{\beta}{d^{\alpha}\left(N_i, N_j\right)} & \text{if } d\left(N_i, N_j\right) \le r_s \\ 0 & \text{else.} \end{cases} \quad (2)$$

r_i is the signal received from the node N_j. β is the original strength of the signal transmitted by a node. α is attenuation coefficient depending on the environment. r_s is the detection range of the node. $d(N_i, N_j)$ is the Euclidean distance between the two nodes N_i and N_j.

To facilitate the description of the protocol, we adopt the following notations:

(i) R_t is node communication range.

(ii) r_s is the node detection radius.

(iii) $R(n_i, r_s)$ is the detection region of node N_i with the detection range r_s.

(iv) G is the wireless sensor network $\mathbf{G} = (\mathbf{V}, \mathbf{E})$.

(v) E is the set of all links between nodes. The set E is defined by $E = e = (v_i, v_j) \mid \{v_i, v_j \in V^2 \cap d(v_i, v_j) \le R_t \cap i \ne j\}$.

(vi) $\mathbf{V}$ is the set of all nodes: $\mathbf{V} = \{\mathbf{v_1}, \mathbf{v_2}, \dots, \mathbf{v_n}\}$.

(vii) The cluster $\mathbf{C_i}$ is defined by $\mathbf{C_i} : \{\mathbf{v_j} \mid \mathbf{d}(\mathbf{v_i}, \mathbf{v_j}) \le \mathbf{R_t}\}$; $\mathbf{v_i}$ is the cluster head node.

(viii) $\mathbf{L_{(t)}}$ is the location of the target at time t.

3.2. The Activation of Nodes in Sleep State (Prediction). There are two types of targets in relation to their ability to communicate with the network: targets that can communicate and targets that lack this ability.

A target that can communicate is a target equipped with a communication module allowing it to transmit signals or periodic messages (hello messages, for example). On the other hand, the no-communicating targets are devoid of this capacity; they are more realistic especially when it comes to model natural phenomena. We are interested in this kind of target in the following: the detection process becomes more complex as these targets are not cooperative. In our contribution, we propose an algorithm that uses the activation of sensors located on the way to the target by overlistening nodes belonging to the active cluster. The activation process is described as follows.

Let $C_K : \{v_j, d(\text{CH}, v_j) \le R_t\}$ be the set of the CH (cluster head) node neighborhoods of the active cluster, which is defined as

$$N_{\text{Neig}} = \left\{v_i \in V \cap \overline{C_K} \mid \exists v_j \in C_K,\ d\left(v_i, v_j\right) \le R_t\right\}. \quad (3)$$

N_{Neig} is the set of all neighboring nodes to the active cluster C_K.

$D_K = \{v_j \in C_K \mid \exists t_0,\ d(L_{(t_0)}, v_j) \le r_s\}$ is the set of the nodes belonging to the active cluster which detected the target at some point. It also defines the set of the nodes that

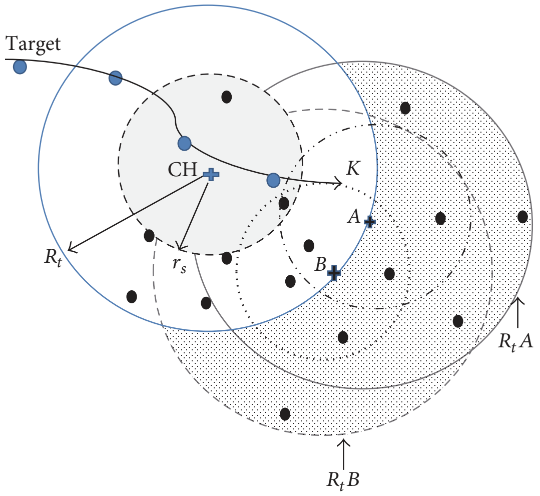


Figure 2: Activation of nodes in sleep state.

will be activated by the prediction process based on three overhearing messages by

$$N_{\text{Activ}} = \left\{ v_i \in V \cap \overline{C_K} \mid \exists v_j \in D_K,\ d\left(v_i, v_j\right) \leq R_t \right\}. \quad (4)$$

It is clear that $N_{\text{Activ}} \sqsubset N_{\text{Neig}}$; that is to say, the set of the nodes that will be activated is only a subset of the active cluster neighbors. This improves energy efficiency by activating just the nodes that potentially will detect the target.

In Figure 2, the active cluster is defined by the transmission range R_t of the CH. All nodes of the chopped area will move to the active state; nodes A and B which have detected the target and which are at the edge of the active cluster mark the wake zone by their respective transmission range: R_tA and R_tB.

3.3. Clusters Overlap Problem. The majority of the dynamic clustering algorithms suffer from clusters overlap problem. Figure 3(a) is showing that cluster 1 is still operational because the target is in range of node B, while another cluster will be formed when node A detects the target at point K.

To minimize the overlap time, we define an area where nodes, awakened by the overhearing messages, will have a status CH_Forb; that is to say, they cannot be CH, even if they are the first to detect the target. This field is the set defined by

$$\mathbf{N}_{\text{Forb}} = \left\{ \mathbf{v_i} \in \mathbf{N}_{\text{Activ}} \mid \exists \mathbf{v_j} \in \mathbf{D_k},\ d\left(\mathbf{v_i}, \mathbf{v_j}\right) \leq r_\mathbf{s} \right\}. \quad (5)$$

In Figure 3(b), the nodes belonging to the chopped area defined by the detection ranges r_sA and r_sB, respectively, of nodes A and B will not participate in the selection of the cluster head; they will have necessarily a member status.

It therefore sets a requirement that node v will have the status Cluster_Ready; that is to say, it could be after cluster head.

The set of the neighbors that detected the target of node $\mathbf{v}$ belonging to the active cluster is defined by

$$\mathbf{N}_{\text{Node-Act}}(\mathbf{v}) = \left\{ \mathbf{v_i} \in \mathbf{D_k} \mid d\left(\mathbf{v_i}, \mathbf{v}\right) \leq \mathbf{R_t} \right\}. \quad (6)$$

Node $\mathbf{V}$ is CH_Ready if and only if

$$\mathbf{V}\ \mathbf{v_i} \in \mathbf{N}_{\text{Node-Act}}(\mathbf{v}) \qquad \min\left\{ d\left(\mathbf{v_i}, \mathbf{v}\right) \right\} > r_s. \quad (7)$$

This simply can be translated by imposing the following condition.

Each node V that is not member of the active cluster receiving a message Msg-data from a member node $\mathbf{v_i}$ must evaluate the distance $d(\mathbf{v_i}, \mathbf{v})$ separating the two nodes, if the condition $d(\mathbf{v_i}, \mathbf{v}) \leq r_s$ is verified; node v cannot be CH and passes to CH_forb status.

3.4. The Selection of CHs. The target tracking phase is triggered by building optimal clusters after the target has been detected. The cluster-building process is illustrated in Figure 5. Recall that a node cannot become a CH unless it has the status CH_Ready; in other words the node belongs to $\mathbf{N}_{\text{ready}}$ group which is defined as

$$\mathbf{N}_{\text{ready}} = \left\{ \mathbf{v_i} \in \mathbf{N}_{\text{Activ}} \mid \mathbf{V}\ \mathbf{v_j} \in \mathbf{D_k},\ d\left(\mathbf{v_i}, \mathbf{v_j}\right) > r_s \right\}. \quad (8)$$

If this is not the case, the node can only become a member pending an invitation from a CH or returns to the sleep state after a system time.

Each node v having the status CH_Ready and detecting the target must generate ΔT timer competition after which it passes to the node cluster head state and subsequently send the messages MSG_Inv to recruit member's nodes.

ΔT is comprised of two parts, as shown in (9). The first term of ΔT aims to create repulsion effect between adjacent clusters and therefore allows distances between clusters one another.

Indeed, the more the candidate node to be CH is far from the active cluster nodes, the more it will have a small competition timer and therefore a greater probability of being a cluster head. This metric reduces overlap and at the same time the number of generated clusters and therefore better energy efficiency:

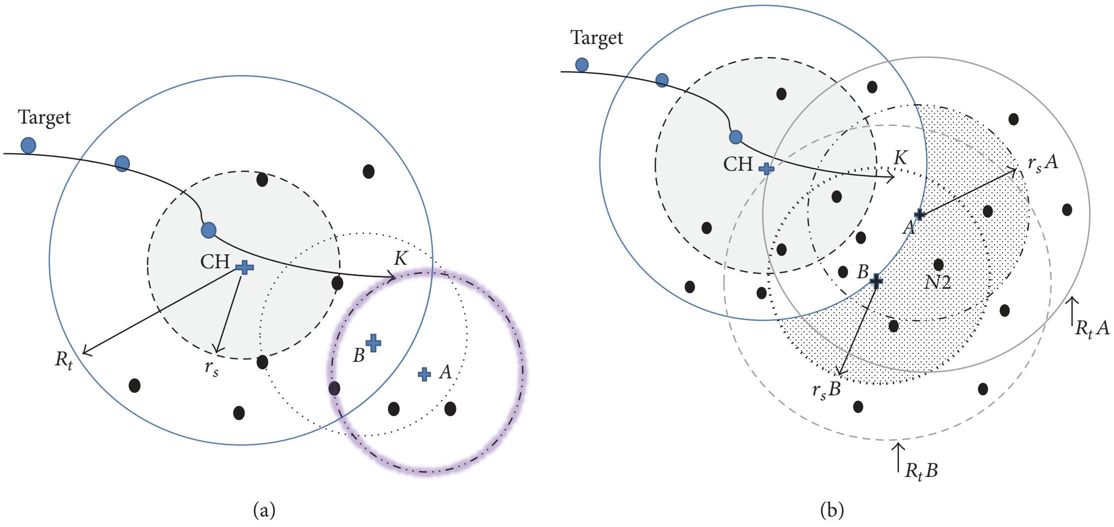
$$\Delta T = \frac{r_s}{\min(v)} + \text{random}(), \quad (9)$$

$$\min(v) = \left\{ \min\left\{ d\left(v_i, v\right) \right\},\ V\ v_j \in N_{\text{Node-Act}}(v) \right\}. \quad (10)$$

The second term of ΔT is a random number between 0 and 1; its role is to prevent two or more candidates nodes generating the same ΔT.

It is easily shown that $\Delta T < 2$ ms, which is not a considerable time, because the target would not have left the scope of detecting even with very high speeds. Figure 4 shows the distance traveled by a target at different speeds in a 2 ms time. At a speed of 1000 km/h, for example, the target would have traveled 5.5 meters against 16 meters at a speed of 30,000 km/h.

The proposed solution reduces the number of clusters formed during the displacement of the target. Indeed, without the new proposed metric, we form more compact clusters with a considerable overlap as shown in Figure 6(a). Our



(a)

(b)

Figure 3: Overlap problem between clusters.

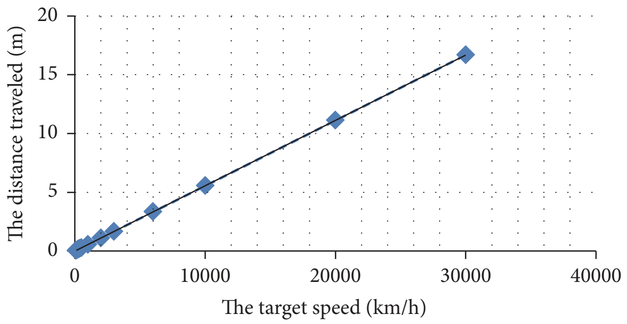


Figure 4: The traveled distance in 2 ms.

solution is trying to acquire a scope increase in each cluster so as to minimize the overlap as shown in Figure 6(b).

Fewer clusters will be formed to the same movement of the target, which will necessarily save total energy consumption.

3.5. The Problem of Reflected Wake Waves. When the target moves through the active cluster (cluster 1 in the diagram), the nodes in this cluster use data messages (Msg_Data) to collect and send the data to their cluster head, waking at the same time the nodes in the outside of the cluster, to achieve in a predictive manner the arrival of the target. A new cluster 2 will be created according to the algorithm proposed in Section 3 once the target is detected by other nodes outside the active cluster. Reference [13] proposes to return all member nodes of the active cluster in sleep state once they no longer detect the target after a system time $\Delta t'$. The problem is to provide a way to properly adjust the system time when the target speed is variable to ensure that the cluster nodes 2, by Msg_data messages, do not wake up the cluster members 1 that have just been put to sleep. These messages can be designed as the reflected wake waves to the origins clusters, which represents a considerable loss of energy by the sleep process and repetitive activation. In Figure 7, node A belonging to cluster 1 is not detecting the target at time $t1$ and triggers a timer $\Delta t'$ after which this the node will return to the sleep state, while, at time $t2$ and time $t3$, MSG_Data messages wake up node B belonging to cluster 2, which also wakes node A. It is obvious that the proper adjustment of $\Delta t'$ depends on the speed of the target and the density of nodes in the network. How then can one adjust this setting in an environment where the velocity of the target and the density of nodes are variables?

Our algorithm solves this problem by synchronizing the transfer to the sleep mode on listening posts MSG_data and not on the moving target. A node cannot return to the sleep state if it does not receive after time $\Delta t'$ MSG_data messages, the messages of predictions, or the reflected wake waves. This method of synchronization will not therefore be affected by the variable speed of the target or by the density of nodes.

3.6. The Transition State Diagram

(i) Five possible states have been defined in the algorithm as shown in Figure 8: sleep, CH-ready, CH, CH-for, and member.

(ii) Initially or when there is no target activity in the network, all nodes are in the state of "sleep."

(iii) When node i is in "sleep" state, it receives Msg-data packet from another node j belonging to the active cluster; node i goes to the CH-For state if $dij < r\_s$ or CH-ready state otherwise.

(iv) dij is the distance between the two nodes i and j.

(v) r_s is the detection range.

(vi) Only the nodes with the CH-ready state can become CH after their detections of the target; contrariwise, the nodes with the CH-For state can only become members.

(vii) A node i in CH-ready state and receiving MSG-data to another node j, which is belonging to the active

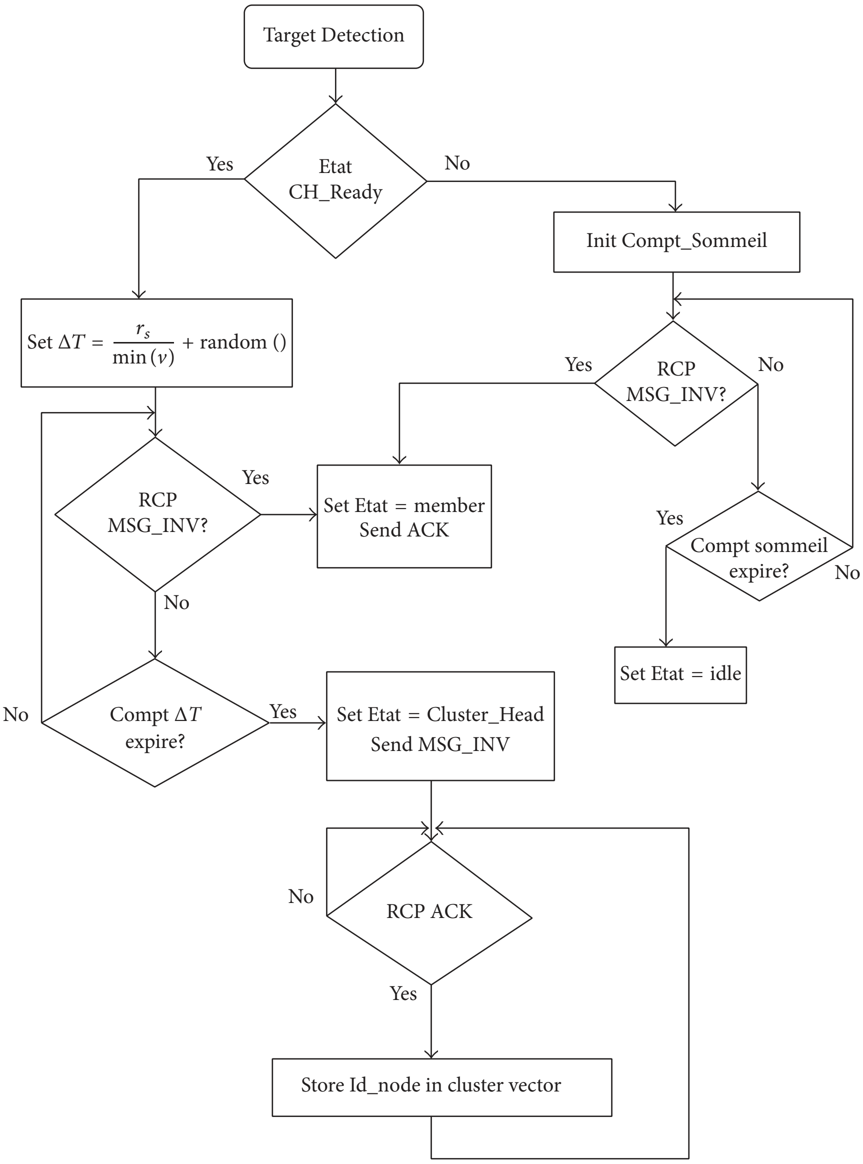


Figure 5: The organization chart of ATHVC.

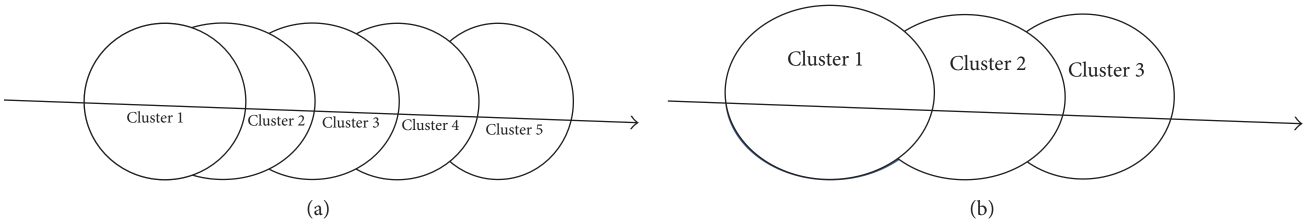


Figure 6: Overlapping of the clusters.

cluster with $dij < r\_s$, must pass to the CH-for state and remains there until becoming a *member of a cluster* or return to the sleep state, if the sleep counter timer expires.

(viii) A node i in CH-ready state and detecting the target triggers timer ΔT calculated from (9).

(ix) If node i receives a MSG-inv, another CH before the expiry of the ΔT timer, node i switches to the member state.

(x) A node i in CH-ready state or CH-for, receiving MSG-inv invitation message, switches automatically to the state member.

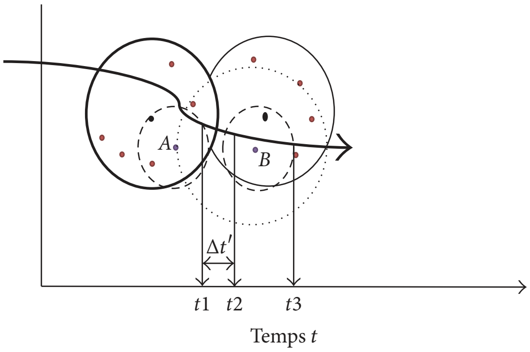


Figure 7: The problem of reflected wake waves.

Table 1: The simulation parameters.

Parameter	Values
Deployment surface	(0, 0) × (10000 m, 10000 m)
Displacement of the target	Random
Eelec	50 nJ/bit
Eamp	10 pJ/bit/m^2
Eamp2	0.0013 pJ/bit/m^4
Initial energy of a sensor	10 J
Sleep mode	0.005 J/second
Active mode	0.078 J/second

(xi) A member which is not receiving MSG-data messages after a system time t sends a cluster destruction message to his CH and returns to the sleep state.

(xii) A CH, receiving cluster destruction messages, removes those members of the member vector until having no members; it returns itself ultimately to the sleep state.

4. The Performances Evaluation

In this section, we are evaluating the proposed solution performance. We are using the NS-2 simulator [18] to perform this task. In the next experiments, the S-MAC IEEE protocol is used as a model for the data link layer. We are also fixing the bandwidth of all nodes to 2 Mbit/s and R_t scope nodes to 200 m while r_s to 50 m. The remaining parameters are presented in Table 1. The results presented in this section are corresponding to an average value of the results obtained by simulation of several scenarios chosen in a random manner. The proposed clustering algorithm was designed with the objective to design a solution to save energy consumption by minimizing the duplication of clusters built for tracking targets moving with great and variable celerity. The evaluation metrics will be measured for relatively large values of the speed of the target. To highlight the contributions of this algorithm, we will focus on the following metrics:

(i) The number of the cluster generated during the tracking process

(ii) Energy consumption

(iii) Tracking precision

We are comparing our algorithm ATHVC with two dynamic clustering schemes for the target tracking cited in Section 2: CHEW [13] is using, as the prediction method, the overhearing messages from the active clusters; the second solution, DKF DC [15], is using Kalman filters to prevent the trajectory of the target.

4.1. The Number of Generated Clusters. This metric determines the effectiveness of the algorithms relative to their ability to generate optimal clusters depending on the target's movement; the more the number of clusters is reduced, the more effective the algorithm is. Figure 9(a) is showing the number of clusters generated during the target tracking based on the simulation time which is proportional to the distance traveled by the target. This test is performed with 1000 nodes distributed randomly, with a low speed of 10 m/s, to have the opportunity to compare our algorithm with CHEW and DKF DC algorithms that support only low speeds. Our algorithm provides optimum generation of clusters.

Our algorithm provides an optimum generation of clusters according to the movement of the target; the results bring up an average of 27% of improvement compared to CHEW and 33% against DKF DC in number of clusters generated for the same movements of the target.

We explain these results by using a metric for selecting cluster heads to create a repulsion effect between adjacent clusters and also the techniques of avoidance of duplication, which greatly reduces the number of clusters generated for the same displacement of the target.

Figures 10(a) and 10(b) are showing two examples of simulations, respectively, with ATHVC and CHEW algorithms. We have ensured that the execution environments of the two algorithms are the same (the same distribution nodes and the same trajectory of the target). It is clearly noted that ATHVC has built larger clusters and therefore less overlap than CHEW. We have also noted the independence of the number of generated clusters compared to the density of nodes and the target velocity.

We almost get the same previous results in Figure 9(b) with a speed of 30 m/s and 2000 nodes.

4.2. Energy Consumption. We rely on the energy model whose parameters are summarized in Table 1. To evaluate the energy consumed during the monitoring process, we record the energy cost of deferent tasks such as the sensors startup, the time of activity, location, and message exchanged with the CH. The simulation time will be set to 150 seconds. The results presented in this section are corresponding to the mean value of the results obtained by simulation of several scenarios selected in a random manner. The proposed clustering algorithm has been designed with the aim of providing a solution to save energy consumption and increase the life of large scale networks and high speeds of targets. The evaluation metrics are therefore measured for large values of the number of nodes in the network and for sufficiently large values of

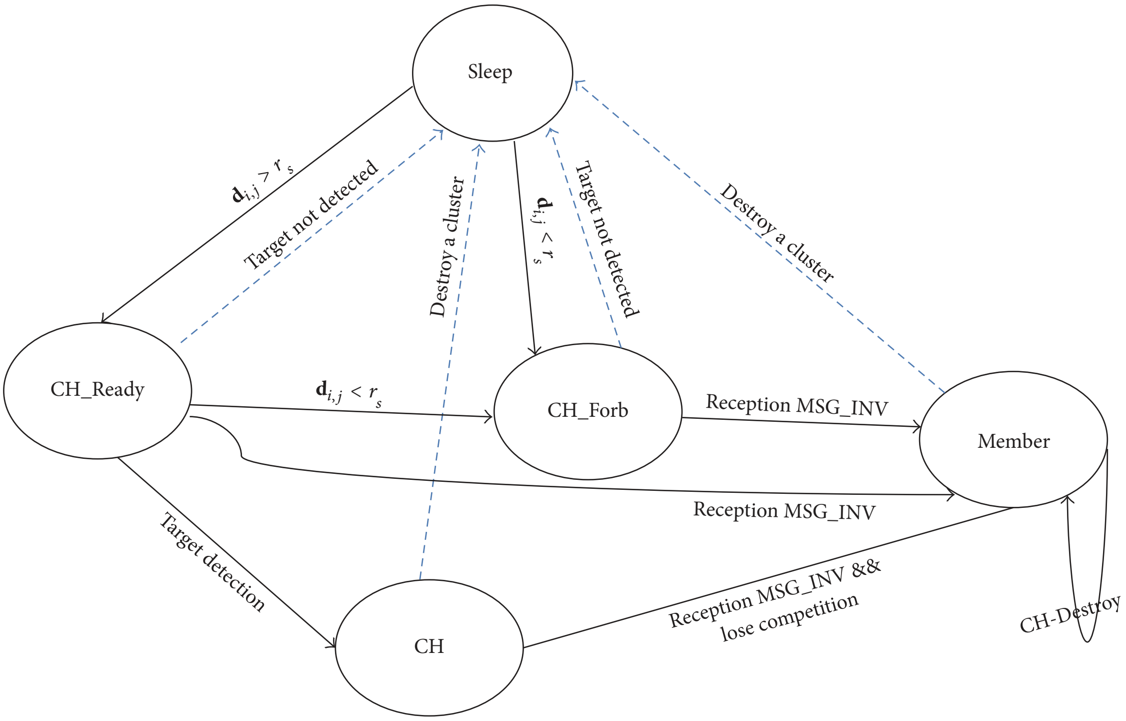
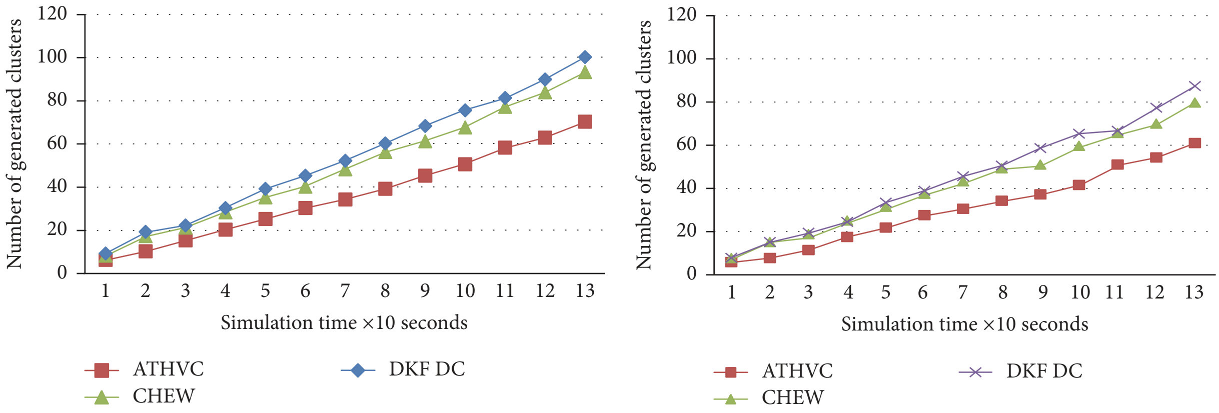


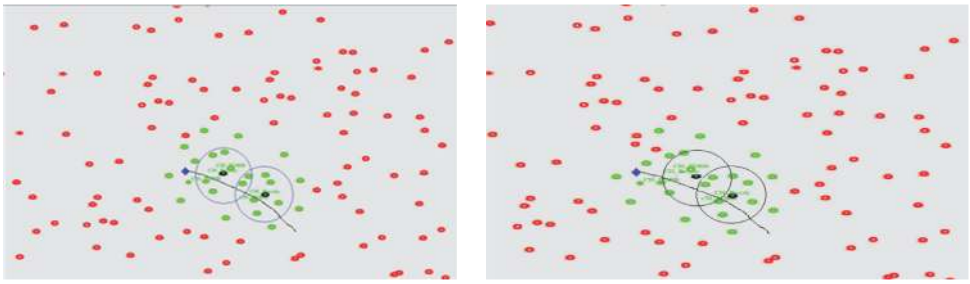
Figure 8: State diagram of the algorithm.



(a) Target speed 10 m/s with 1000 nodes

(b) Target speed 30 m/s with 2000 nodes

Figure 9: The number of clusters based on the simulation time.



(a)

(b)

Figure 10: (a) Simulation with ATHVC. (b) Simulation with CHEW.

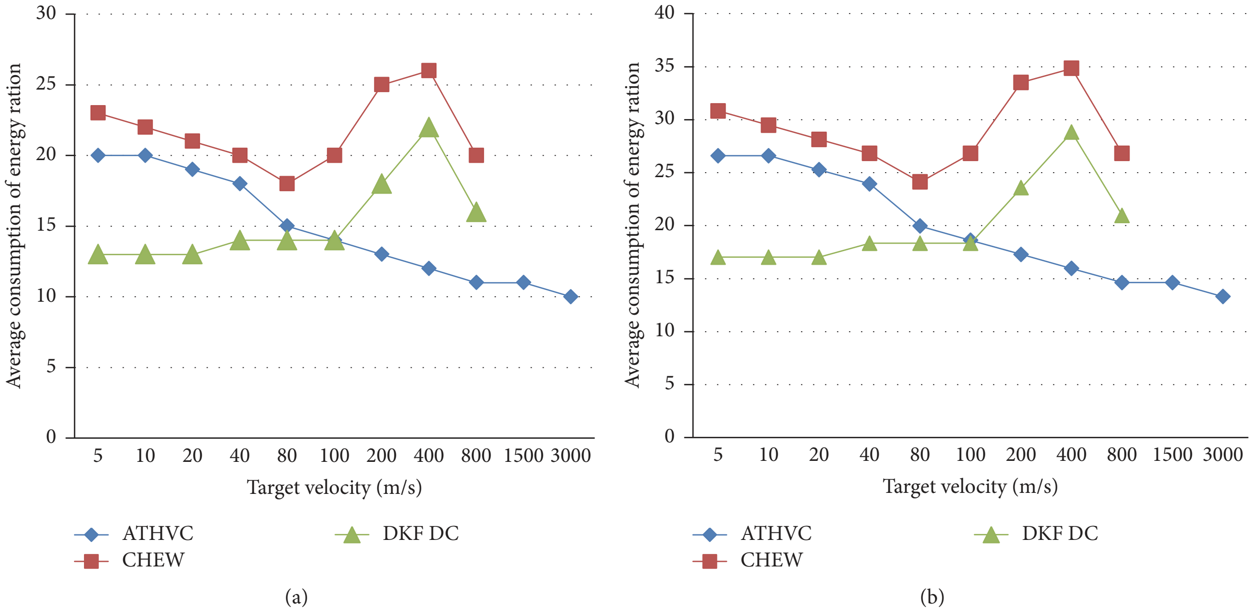


FIGURE 11: (a) Energy consumption with 1000 nodes. (b) Energy consumption with 2000 nodes.

the speed of the target. Figure 11(a) is showing the average energy consumption of the network. As expected, DKF DC shows good performance for low speeds thanks to the predictive model. Only the sensors on the target's trajectory are activated, but from speed of 100 m/s, the consumption has been increased significantly before falling on, reaching the velocity $v = 390$ m/s.

The first increase is explained by unnecessary activations of the sensors due to the prediction errors; in fact, these errors are more frequent when the speed of the target is increased. But from a critical velocity V there is a relapse of the energy consumed, because the high-speed target simply eludes most of the formed clusters. The energy consumed declines but at the expense of tracking accuracy since we are collecting little information about the target.

ATHVC algorithm outperformed CHEW even at low speeds thanks to the clusters overlap avoidance mechanism before CHEW creates unnecessary awakenings due to the problem of reflected wake waves reaching speed of 120 m/s; energy consumption, therefore, has been increased in a considerable way before collapsing for the same reasons as DKF DC.

ATHVC ensures the best energy efficiency that increases with speed. ATHVC ensures the best energy efficiency that increases with speed, since the nodes commissioning time decreases with increasing the speed of the target. The prevalence of the algorithm ATHVC with respect to the other algorithms is ensured even with different deployments of the nodes as shown in Figure 11(b) which was conducted by a simulation with 2000 nodes. The energy consumption is increased for the three algorithms; indeed with a high density of nodes the size of the formed clusters is increased. The deployment of nodes should strongly affect the energy consumption.

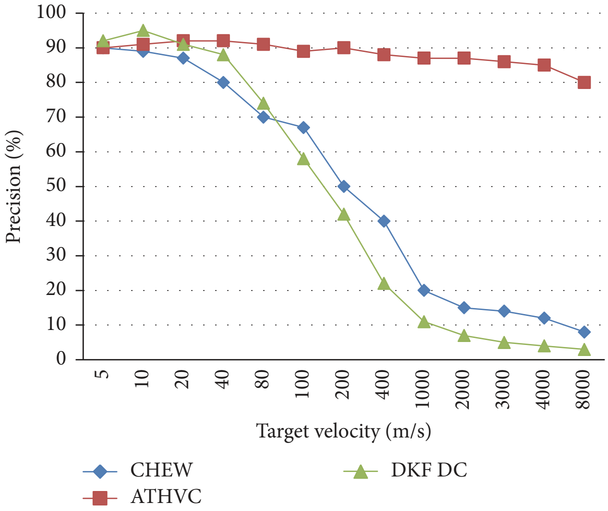


FIGURE 12: The tracking accuracy.

4.3. The Tracking Accuracy. This metric is dependent on the number of coordinates obtained on a given path. This will allow us to measure the ability of algorithms to ensure the function of monitoring with a high speed of the target.

Figure 12 is showing the results, with a low target velocity up to 70 m/s, DKF DC gets the best performance, and the algorithm reaches up to 95% accuracy. We explain these results by the efficiency of the prediction mechanism. Indeed, by coupling FKE (extended Kalman filter) and CuSum, we can capture the trajectories realistic targets. As soon as the speed exceeds 80 m/s for Chew and 100 m/s for DKF DC, the target simply eludes to the most clusters formed; therefore they are collecting little data from the target and the accuracy collapses to 10% and 7%, respectively, for CHEW and DKF DC for a speed of 8000 m/s. Also note the impact of the deployment strategy on performance and tracking accuracy, but, in any case even with totally random deployment strategies, our ATHVC algorithm achieves good performance (up to 80%) even with colossal speed of the target.

5. Conclusion

In this paper, we have proposed clustering algorithm for a target tracking. Our algorithm has built dynamic clusters on demand and included a phase to activate the sensors based on overhearing messages of the nodes in the active cluster. This phase implements, for the first time, two strategies; the first technique has minimized the overlaps between clusters formed during the movement of the target and the second one has avoided the reflected wake waves in the case of variable speeds, which is the most common for the real targets. Then we have proposed a new metric for selecting cluster heads adapted to the targets moving with great and variable celerity. The simulation results have demonstrated the effectiveness of the proposed models and the prevalence of our algorithm compared with other algorithms, cited in related work, especially in the context of the high-speed targets.

As a future work, we are planning to design a variant of ATHVC algorithm to adapt to an environment with multiple targets. This type of monitoring is complex and involves additional steps including identification to help differentiate and count targets.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

References

[1] P. Nayak and A. Devulapalli, "A fuzzy logic-based clustering algorithm for wsn to extend the network lifetime," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 137–144, 2016.

[2] L. Zheng, L. Gao, and T. Yu, "An energy-balanced clustering algorithm for wireless sensor networks based on distance and distribution," in *Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation: Innovation and Practice of Industrial Engineering and Management (volume 2)*, pp. 229–240, Atlantis Press, 2016.

[3] C. Vimalarani, R. Subramanian, and S. N. Sivanandam, "An enhanced PSO-based clustering energy optimization algorithm for wireless sensor network," *The Scientific World Journal*, vol. 2016, Article ID 8658760, 11 pages, 2016.

[4] S. Sen, D. Karmakar, and S. K. Setua, "An power efficient algorithm for distributed ad-hoc cluster based Wireless Sensor Networks," in *Proceedings of the 3rd International Conference on Computer, Communication, Control and Information Technology (C3IT '15)*, pp. 1–6, February 2015.

[5] D. Kumar, "Performance analysis of energy efficient clustering protocols for maximising lifetime of wireless sensor networks," *IET Wireless Sensor Systems*, vol. 4, no. 1, pp. 9–16, 2014.

[6] M. Akter, O. Rahman, N. Islam, and A. Habib, "Incremental clustering-based object tracking in wireless sensor networks," in *Proceedings of the International Conference on Networking Systems and Security (NSysS '15)*, pp. 1–6, January 2015.

[7] 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.

[8] 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.

[9] J. Xiong, Z. Liu, G. Tang et al., "A moving target collaborative tracking algorithm based on dynamic fuzzy clustering," in *Electrical, Control Engineering and Computer Science: Proceedings of the 2015 International Conference on Electrical, Control Engineering and Computer Science (ECECS 2015, Hong Kong, 30-31 May 2015)*, p. 73, CRC Press, 2015.

[10] J. Hare, S. Gupta, and J. Wilson, "Decentralized smart sensor scheduling for multiple target tracking for border surveillance," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '15)*, pp. 3265–3270, May 2015.

[11] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proceedings of the Multisensor Fusion, Tracking, and Resource Management II*, vol. 3068 of *Proceedings of SPIE*, p. 182, Orlando, Fla, USA, April 1997.

[12] F. Gustafsson, *Adaptive Filtering and Change Detection*, John Wiley & Sons, New York, NY, USA, 2000.

[13] 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. 10, no. 3, Article ID 670962, 2014.

[14] J. G. Kemeny and J. L. Snell, *Finite Markov Chains*, van Nostrand, Princeton, NJ, USA, 1960.

[15] O. Demigha, H. O. Slimane, A. Bouziani, and W.-K. Hidouci, "Energy efficient target tracking in wireless sensor networks with limited sensing range," in *Proceedings of the 6th International Conference on Systems and Networks Communications*, Barcelona, Spain, October 2011.

[16] M. Akter, M. O. Rahman, M. N. Islam, and M. A. Habib, "Incremental clustering-based object tracking in wireless sensor networks," in *Proceedings of the International Conference on Networking Systems and Security (NSysS '15)*, pp. 1–6, January 2015.

[17] 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. 9, no. 3, Article ID 494863, 2013.

[18] K. Fall and K. Varadhan, The network simulator (ns-2), 2007, http://www.isi.edu/nsnam/ns.

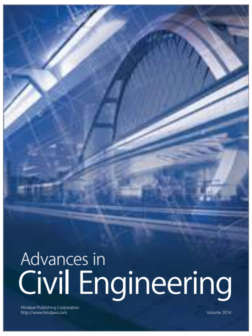
Journal of
Engineering

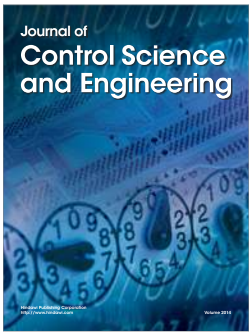
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

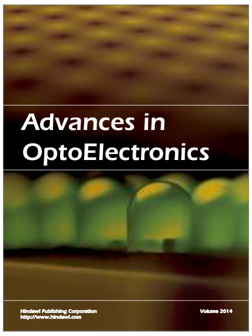
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

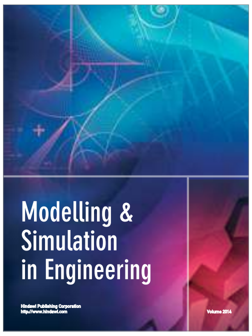
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

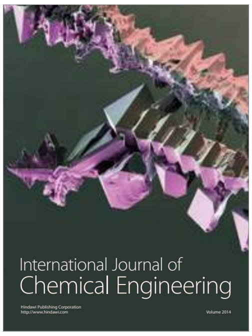

Advances in
OptoElectronics

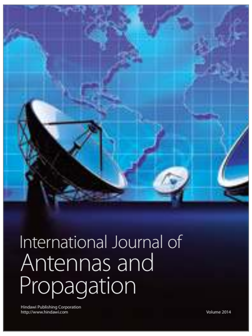
VLSI Design

International Journal of
Navigation and
Observation

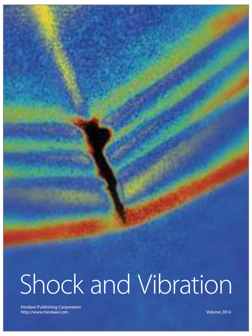

Modelling &
Simulation
in Engineering

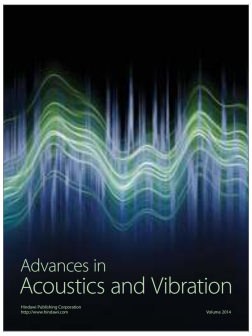
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration