Energy efficient approach for target tracking in Wireless Sensor Networks

Asmaa Ez-zaidi † and Said Rakrak †
Applied Mathematics and Computer science Laboratory (LAMAI)
Cadi Ayyad University
Marrakech, Morocco

Summary
Target tracking is one of the hottest topics of research in wireless sensor networks. It consists in detecting and retrieving the successive locations of single or multiple objects during their movements in the monitored area. Minimizing the energy consumption while providing high accuracy in target tracking is a challenging problem as sensor nodes are constrained in terms of energy.
In this paper, we present an energy efficient approach for object tracking based on data mining techniques to predict the future location of the target. Nodes within this location are activated while the others remain in sleep mode which reduce the energy consumption and prolong the network lifetime. Moreover, we introduce a recovery mechanism to recapture the object in case it was missed. Extensive simulations via NS-2 simulator revealed the effectiveness of our proposed approach.
Key words:
Wireless Sensor Networks, Target tracking; mobile object; tracker; recovery; prediction; energy consumption; target loss

1. Introduction

Wireless sensor networks are made of a large number of low-power, low-cost and multifunctional sensor nodes that monitor physical or environmental conditions and cooperate with each other wirelessly to forward collected data to a central base station for further analysis. WSNs enable novel applications in a wide range of disciplines such as health care, environmental monitoring, industrial and manufacturing automation and so on. Target tracking is one of the most important applications of wireless sensor networks that have received considerable attention.

The main concept of object tracking is to monitor the location, speed and the trajectory of one or multiple moving objects that can be persons, animals, vehicles or even robots [1]. In the tracking process, nodes cooperate among each other in order to not lose the object. They can stay active during the whole tracking process, or switch between active and sleep mode to conserve energy. Depending on the object's velocity, the number of tracking sensors may vary. Binary sensors can be also used in tracking [2] [3], they make binary decision by providing only 1-bit information (1 or 0) regarding the presence or the absence of the target in their range. This type of nodes might be suitable in certain applications since they do not consume energy and the probability of providing erroneous measurements is very low.

Sensor nodes are constrained in terms of energy since they are powered by batteries that are in most cases impossible to replace or recharge. Energy consumption during the tracking process is the key concern for the majority of approaches proposed in the literature. When all the nodes are kept active during the tracking process, more energy is dissipated but high accuracy is achieved. Finding a tradeoff between energy conservation and tracking accuracy is a big challenge. Energy can be saved by putting sensor nodes in sleep mode when the object is not in their vicinity and activating only nodes having the target in their range.

Prediction algorithms are used for this purpose. They forecast the future position of the target and activate only nodes that are located around that position while the remaining nodes are kept in sleep mode. Tracking performances depend strongly on the adopted prediction strategy, which unfortunately does not guarantee 100% of accuracy and there will be always a probability for missing the target due to errors or anomalies, node failures, sharp change in the target's velocity and direction. So a recovery technique should be envisaged to recapture the lost object in a quick manner.

In this paper, we present an energy efficient approach for target tracking using data mining technique to predict the upcoming positions of the target. By doing so, only nodes near the predicted locations will be activated while the others will remain in sleep mode to conserve the network's energy. Our approach includes also a recovery mechanism to retrieve the target in case it was lost.

The remainder of this paper is organized as follows. Section 2 highlights the relevant related work in objet tracking. Section 3 presents our proposed approach; section 4 discusses the simulation results. Conclusion and future works conclude the paper.

Manuscript received September 5, 2017
Manuscript revised September 20, 2017

2. Related Work

In recent years, target tracking in Wireless Sensor Networks has attracted considerable attention [4]. Most of the research works have covered the energy issue in target tracking by using prediction mechanisms to reduce the number of nodes participating in tracking. Prediction consists in forecasting the upcoming positions of the object, so only sensor nodes located near those positions are turned on while the others are kept in sleep mode to save energy. To estimate the target’s trajectory and predict its movement, various methods were used such as particle filter (PF) presented in [5], extended kalman filter (EKF), Kalman Filter in addition to kinematics.

In [6] Kalman filter were combined with machine learning to track the target and estimate instantaneously its positions. Authors in [7] have constructed a tracking contour based on vehicular kinematics to exclude the most unlikely areas where the object cannot traverse within a limited time.

Another interesting work was proposed in [8], in which authors have used face structure to track the object and used the least square method LSM to predict the path of the moving target. Based on this prediction the face to which the target is expected to move to is activated while the other faces are turned in sleep mode.

An energy efficient prediction-based clustering algorithm for object tracking was proposed in [9], it uses a linear prediction method to anticipate the next location of the target. Then, the CH selects three sensor nodes that are near the estimated position to perform target localization locally using the trilateration mechanism.

Authors in [10] studied the problem of sleep scheduling for target tracking in energy harvesting sensor networks and proposed a heuristic sleep scheduling algorithm that chooses the awaked nodes with high residual energy and nearer to the target to perform the tracking task.

In [11] a dynamic cluster and duty cycling mechanism have been used to schedule sensor nodes that will participate in the tracking process. The trajectory of the object is predicted by using Kalman filter. Clusters are formed dynamically as the target moves and only the cluster having detected the presence of the object is activated.

Data mining techniques was also used in target tracking, their major role consists in extracting useful association rules from the historical movement information of the target [12]. These rules help in understanding the behavior of the target and to anticipate its future positions.

As instance, in [13] authors proposed a prediction-based tracking technique using sequential patterns (PTSP), it consists in generating patterns that will be used for predicting the future position of the object. The approach tends to conserve energy while tracking and in the same time maintaining tolerable missing rate levels. As prediction techniques do not guarantee 100% of accuracy and there is always a possibility to lose the target, authors have presented three recovery mechanisms for relocating the missed object.

3. Proposed approach

3.1 System architecture:

Our network consists of a set of stationary sensor nodes scattered randomly over a large area. All the sensor nodes have the same communication range and are aware of their own locations. They have the same capabilities in terms of sensing, computing and data processing. We assume the presence of only one target in the field that crosses the network in a random manner with a constant speed. Sensor nodes monitor periodically the region to detect and track the object. Figure 1 depicts the target tracking scenario.

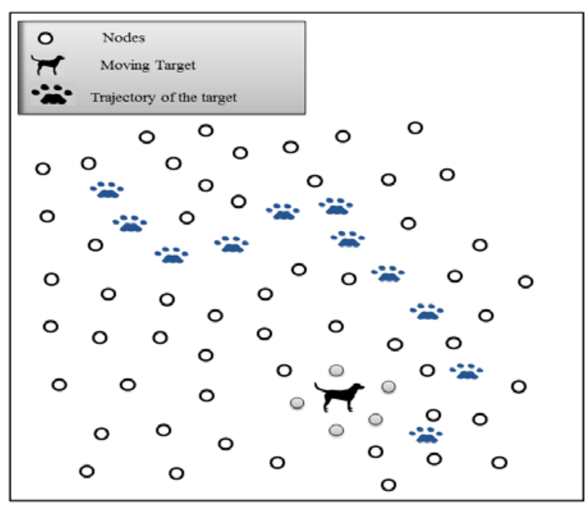


Fig. 1 Our network model.

3.2 Target tracking

At the beginning, all the border nodes are kept active to detect the presence of the object once it enters the sensing area. The other nodes stay in sleep mode to save energy.

In an attempt to reduce the energy consumption while tracking objects, we used the concept of prediction to keep only a necessary set of sensors working, while putting the rest in a sleeping mode.

We try to obtain interesting patterns from the historical movement information of the target. We have applied Eclat algorithm to discover relevant frequent item sets and then induce useful association rules that will be further used for analyzing and predicting the target's behavior and path.

In addition to Eclat, other algorithms have been presented in the literature such as Apriori and FP-growth. From a set of unknown interconnections, these algorithms are able to discover frequent patterns and then conclude interesting rules between them.

Apriori was presented by Agrawal and Srikant [14], it is a well-known algorithm, used to extract frequent itemsets based on candidate generation in a given database and get the association rules. It works in a breadth-first manner. At the beginning, Apriori scans the database and finds the support count of each item. Infrequent items with support count less than the minimum support are removed. Candidate k-itemsets are generated from frequent k-1 itemsets through join and prune. This process is repeated and the database is scanned several times as long as large frequent itemsets are generated.

FP-Growth (Frequent Pattern Growth) was proposed [15] to overcome drawbacks in Apriori. FP-Growth is a popular algorithm for frequent itemset mining without candidate generation. It uses divide and conquer technique and it only needs two database scan to find the frequent item sets. In the first scan of the database, items are found and sorted in descending order while the infrequent ones are discarded. In the second scan of database the FP-tree is constructed.

Equivalence Class Transformation "ECLAT" algorithm was introduced by Zaki [16]. This algorithm uses a vertical data format to store the list of transactions (tid) for each item. Contrarily to Apriori, ECLAT scans the database only once to extract the frequent itemsets. Besides, it traverses the prefix tree in a depth manner which reduces the search space. Eclat is composed of two steps: candidate generation and pruning. The database is scanned in order to construct the tidset of each single item. The support of these items is the length of its tidset. K+1-itemset candidate is generated by intersecting two frequent k-itemsets that share a common prefix then its support is counted and compared with a threshold. If the support is greater than that threshold, then it is a frequent itemset and thus it will be used to generate k+2 itemsets, otherwise it will be discarded. This process is repeated until no candidate itemset are found.

Why choosing Eclat?

- Apriori algorithm scans the database several times, which increases its time complexity and generates huge candidate sets which make Apriori very slow and memory consuming. Deficiencies in the Apriori algorithm led to the development of other algorithms that are more efficient such as Eclat [17].
- Performs quicker.
- Eclat explores the search space in a fast manner in comparison with Apriori [18].

We have applied Eclat algorithm in Wireless Sensor Networks to track and predict future movements of the target based on its behavior.

1. At the beginning, nodes monitor the sensing region to gather the movement sequences of the target. This process is launched several times in order to collect a large volume of data which will be then forwarded to the sink node for further analysis. The data format is as follows: *sensor ID, time*. *Sensor ID* is the node having detected the target, and *time* is the time when the target has been detected.
2. Then the sink, which has high computational performances, aggregates the received movement information (fig 2) and performs Eclat algorithm to find persistent patterns (set of nodes that the target usually passes by) and useful associations that indicate the behavior of the object in the network.
3. These associations are then embedded in all sensor nodes by the sink to allow them predicting the next movements of the tracked object.

Figure 2 depicts an example of the target's recorded movement sequences that the sink will mine. Sni designates sensor node number X. The sink applies Eclat algorithm to find the frequent motion patterns of the object.

TID	Itemset
1	Sn27, Sn29, Sn30, Sn32, Sn33, Sn36
2	Sn32, Sn33, Sn36, Sn37, Sn40
3	Sn27, Sn29, Sn30, Sn32, Sn33, Sn15
4	Sn32, Sn33, Sn36, Sn38, Sn39
5	……..
6	…………

Fig.2 Initial database

Steps of Eclat:

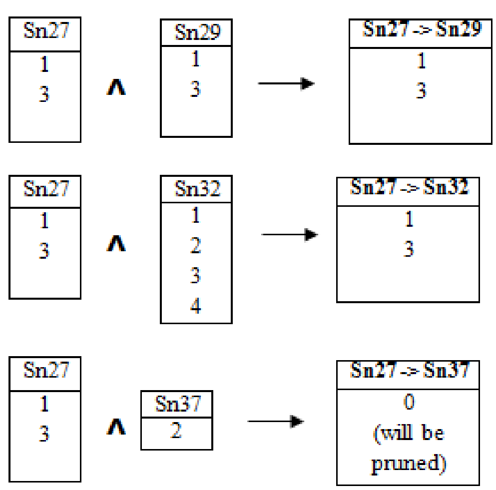
First, the initial database is transformed from horizontal format to vertical one (Fig 3). For each item (sensor node in our case) a transaction list is created. Sensor node 27 (Sn27) for example appears in transaction 1 and 3, Sensor node 32 "Sn32" appears in transaction 1, 2,3 and 4 and so on.

Sn27	Sn29	Sn30	Sn32	Sn33
1	1	1	1	1
3	3	3	2	2
			3	3
			4	4

Sn36	Sn37	Sn38	Sn39	Sn40	Sn15
1	2	4	4	2	3
2					
4					

Fig.3 Vertical layout of the initial database

The second step is to create 2-Itemset candidates. It is performed by intersecting two frequent 1-itemsets sharing the same prefix. For example to create 2-itemset candidates like {Sn27 -> Sn29}, {Sn27 -> Sn30}, {Sn27-> Sn32} etc, we intersect Tidlitst of {Sn27} with all other items as shown in the figure below. The same concept will be applied with {Sn20} giving {Sn20, Sn23}, {Sn20, Sn4}, {Sn20, Sn30} and so on…



After constructing the 2- frequent itemsets, 3-itemsets are constructed using the same logic. Tidlitst of {Sn27, Sn29} is intersected with all the items which generate: {Sn27→ Sn29→ Sn30}, {Sn27→ Sn29→Sn32}, {Sn27→ Sn29→Sn33} etc…

After pruning infrequent itemsets that have a support count less than the minimum support and extracting the association rules, the results are:

P1:
Sn1 → Sn27→ Sn29→ Sn30→ Sn32→ Sn33→ Sn36→ Sn38→ Sn39
P2:
Sn30→ Sn32→ Sn33→ Sn36→ Sn37→ Sn40→ Sn51

Once the computation of the frequent paths by the sink has finished, they are embedded in sensor nodes to allow them predicting the future position of the target and determine which nodes have to be activated.

From the figure below, and according to the association rules extracted P1 and P2. Node Sn32 knows that the target will most likely pass by node Sn33. For this reason, Sn32 activates node Sn33 before the arrival of the object to take over the tracking process and monitor the target. Sn32 will turn to sleep mode in order to save energy.

Both node Sn37 and Sn38 will be activated by node Sn36, because according to the associations that have been found, the target may pass by one of these two paths {Sn36→Sn37→Sn40} or {Sn36→Sn38→ Sn39}.

Finally, in case the target has been lost, a recovery mechanism is triggered.

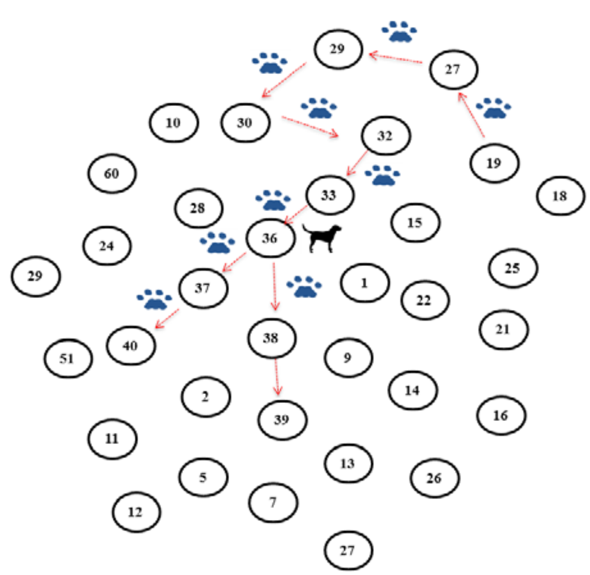


Fig.4 Prediction of the target's future location

3.3 Recovery

Prediction mechanisms do not guarantee 100% of accuracy except if all sensor nodes are awake. Prediction may suffer from errors, noises and anomalies. Therefore, the future position of the moving object will be estimated wrongly. Another scenario may occur is when the target changes its velocity and direction suddenly and becomes undetectable. It is thus mandatory to use a recovery strategy to relocate the missing object in a timely manner.

In our case, we have proposed a recovery mechanism that consists in activating extra sleeping nodes to retrieve the target in case it was lost. Figure 6 illustrates our recovery process. Each node keeps tracking the moving object until it leaves its sensing range (Let’s say Sn32). At this stage, it sends a wakeup message to the next node (let’s say Sn33) that will take over the tracking process.

In case Sn33 has not detected the object within a certain period of time, it will assume that the target has been lost and informs via message the previous node (Sn32) that launches the recovery process in order to recapture the target.

We define a recapture zone Cr and all nodes within this zone are turned on to search the missed object. Cr is the semi-circle that is centered on the position where the target has been lost and its radius is calculated in function of the target’s velocity and the time elapsed since the object was last detected.

The probability that the target makes right turns or left turns is equal. However, to know which semi-circle to activate first, we analyze both behaviors and movement trends of the target based on its recorded history path to define the frequency of its direction changes, more particularly how many times the tracked object makes

turns to the right or to the left. To recapture the target, node Sn33 will activate sensor nodes located in the right semi-circle if the target tends more in its movement to turn right. In case the object could not be retrieved, the left semi-circle is activated.
The recapture zone Cr encompasses also both the cases in which the target either continues moving straight forward or makes a half turn.

4. Simulation results:

To evaluate the performance of our approach, we have used Ns-2 network simulator version 2.35. Our network consists of a set of static sensor nodes deployed in a field area of 100 * 100 m². Each node has as initial energy 5J and has a fixed sensing range with value 15m.

We assume that sensor nodes track only a single mobile target that crosses the network in a random manner with a constant speed.

Simulation metrics:

we have considered three metrics to evaluate our apporach:

i. Energy consumption that represents the quantity of dissipated energy in the network.
ii. Time taken for recapturing the lost target.
iii. Number of times the target was lost in the network.

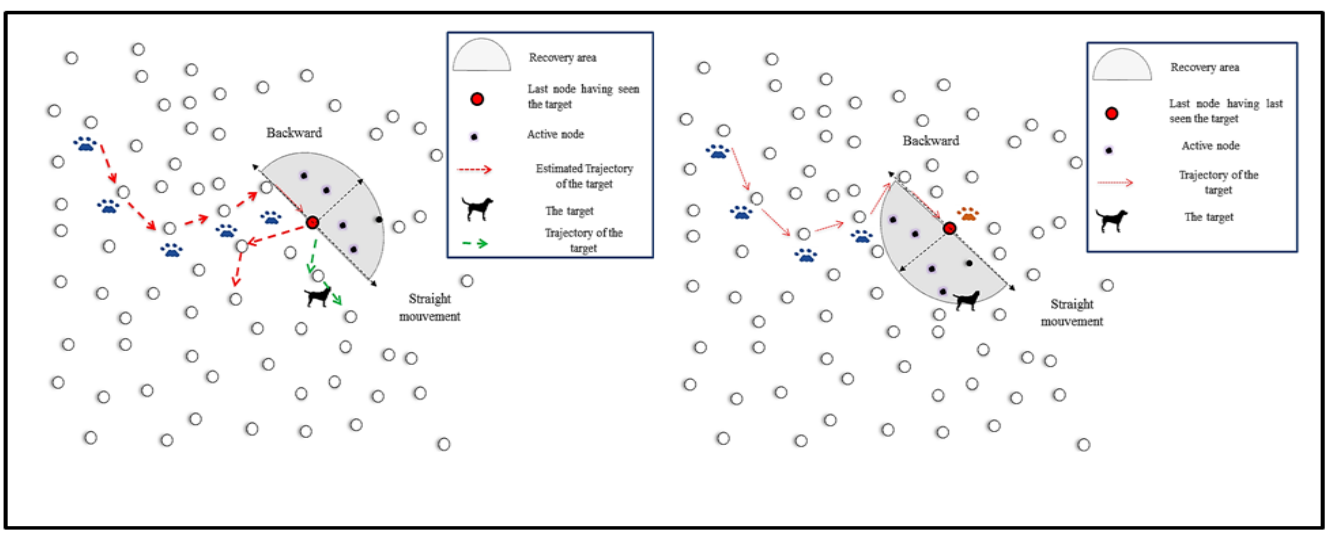


Fig.5 Recovery mechanism.

Simulation parameters	
Area (m²)	100x100
Number of target	1
Number of sensor nodes	50
Simulation time (s)	2000
Target's velocity (m/s)	5, 10, 15, 25
Sensing range	15
Communication range (m)	15
Mac protocol	802.15.4
Energy of nodes (J)	5

4.1 Energy consumption

We evaluate the energy consumption of our approach and compare it with the basic tracking scheme in which all nodes are in active mode to track the moving object. The simulation was carried out by varying the speed of the target between 5 m/s and 25 m/s to evaluate the impact of the speed on energy consumption. Figure 6 shows that as the velocity of the object increases, the energy consumption increases too. This can be explained by the fact that when the target moves at high speed, the probability of losing it becomes high. So, extra nodes have to be activated to recover the object, which means that more energy will be dissipated.

In comparison with the basic tracking scheme, our approach consumes less energy. This is because we anticipate the future location of the target and thus only the node situated near that location is activated while the remaining nodes are kept in the sleep mode to conserve energy.

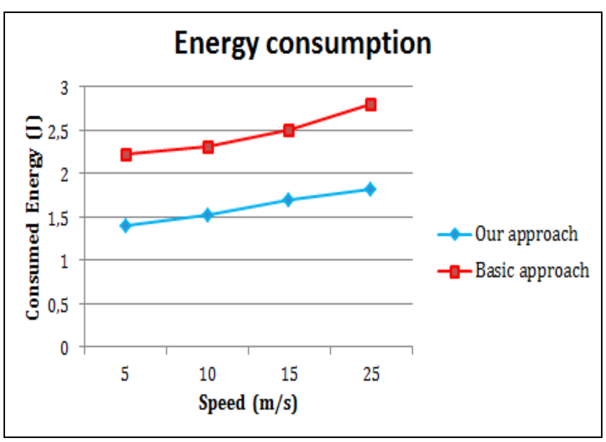


Fig 6. Energy consumption

4.2 Capturing time

In this section, we evaluate the time taken to recapture the lost target. In this experiment, we examine the performances of our recovery mechanism that consists in activating nodes within the right or left semi-circle that is centered on the position where the target has been lost and its radius is calculated in function of the target's velocity and the time elapsed since the object was last detected.

We compute the time taken to relocate the missed target using our proposed recovery mechanism, and we compare it with Eclat-based recovery technique that consists in activating nodes of all frequent itemsets that have been extracted using Eclat algorithm.

We vary the speed of the object from 5 to 25 m/s to examine the impact of the target's speed on the recovery time. We observe from the results that when the speed of the object is low, it is rapidly captured because the distance it will traverse will be relatively short. Conversely, when it moves at high speed, it may make abrupt changes in its velocity and trajectory and may flee the detection area without being detected which prolong the duration of the recapturing.

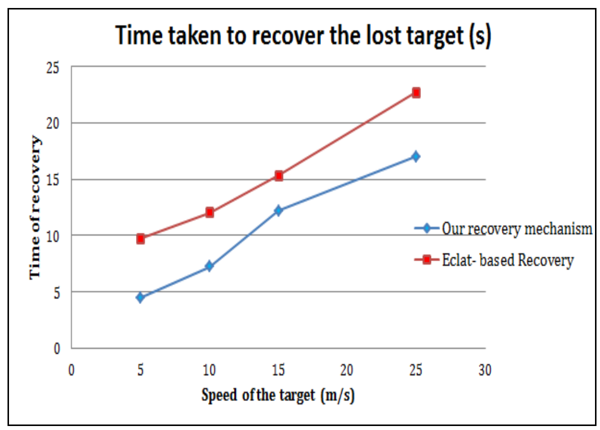


Fig.7 Time taken to recover the lost target

4.3 Number of times the target has been lost

In this section, we evaluate how many times the target has been lost during the simulation. From the figure below, it can be seen that as the speed of the object increases, the number of times the target gets lost increases too. This is because the object may go undetected when passing by sensing nodes with a high speed.

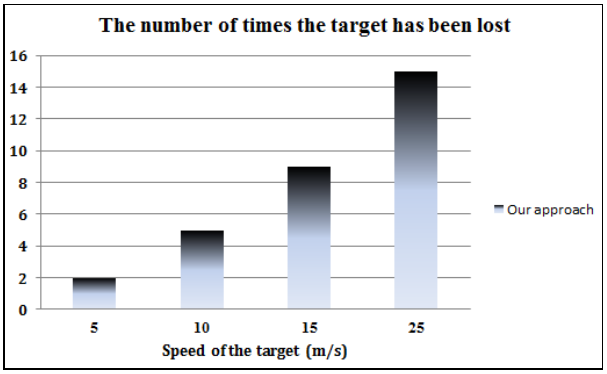


Fig.8 Number of times the target has been lost.

5. Conclusion

Energy conservation plays a major role in target tracking. In this paper we have proposed an energy efficient target tracking approach for wireless sensor networks that uses data mining technique to extract useful association rules from a log of data containing the movement sequences of the target in order to anticipate its trajectory. We have also proposed a recovery strategy in order to relocate the lost object. Simulation results reveal that our scheme can effectively reduce the energy consumption while maintaining accuracy and retrieve the missed object within a very short time. Our future work will mainly consists in testing the performances of our approach using multiple targets.

References

[1] Padmavathi, G., Shanmugapriya, D., & Kalaivani, M. (2010). A study on vehicle detection and tracking using wireless sensor networks. Wireless Sensor Network, 2(02), 173.

[2] Wang, Z., Bulut, E., & Szymanski, B. K. (2010). Distributed energy-efficient target tracking with binary sensor networks. ACM Transactions on Sensor Networks (TOSN), 6(4), 32.

[3] Laoudias, C., Michaelides, M. P., & Panayiotou, C. (2013, June). Sensor health state estimation for target tracking with binary sensor networks. In Communications (ICC), 2013 IEEE International Conference on (pp. 1878-1882). IEEE.

[4] Ez-Zaidi, Asmaa, and Said Rakrak. "A comparative study of target tracking approaches in wireless sensor networks." Journal of Sensors 2016 (2015).

[5] Kalpana, B., & Sangeetha, R. (2013, April). A collaborative target tracking framework using particle filter. In Wireless and Mobile Networking Conference (WMNC), 2013 6th Joint IFIP (pp. 1-4). IEEE.

[6] Mahfouz, S., Mourad-Chehade, F., Honeine, P., Farah, J., & Snoussi, H. (2014). Target tracking using machine learning and Kalman filter in wireless sensor networks. IEEE Sensors Journal, 14(10), 3715-3725.

[7] 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.

[8] 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.

[9] Dayan, V. P., & Vijeyakumar, K. N. (2012). Target tracking in sensor networks using energy efficient prediction based clustering algorithm. Procedia Engineering, 38, 2070-2076.

[10] Zeng, Q., Chen, H., Zhao, F., & Li, X. (2014, October). A sleep scheduling algorithm for target tracking in energy harvesting sensor networks. In Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2014 International Conference on (pp. 323-330). IEEE.

[11] Yan, D., & Wang, J. (2011). Sensor scheduling algorithm target tracking-oriented. Wireless Sensor Network, 3(08), 295.

[12] Peng, W. C., Ko, Y. Z., & Lee, W. C. (2006, May). On mining moving patterns for object tracking sensor networks. In Mobile Data Management, 2006. MDM 2006. 7th International Conference on (pp. 41-41). IEEE.

[13] Samarah, S., Al-Hajri, M., & Boukerche, A. (2011). A predictive energy-efficient technique to support object-tracking sensor networks. IEEE Transactions on Vehicular Technology, 60(2), 656-663.

[14] Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In Proc. 20th int. conf. very large data bases, VLDB (Vol. 1215, pp. 487-499).

[15] Han, J., Pei, J., & Yin, Y. (2000, May). Mining frequent patterns without candidate generation. In ACM sigmod record (Vol. 29, No. 2, pp. 1-12). ACM.

[16] Mohammed J. Zaki. Scalable algorithms for association mining. IEEE Transactions on Knowledge and Data Engineering 12(3):372-390, May/June 2000.

[17] Heaton, J. (2016, March). Comparing dataset characteristics that favor the Apriori, Eclat or FP-Growth frequent itemset mining algorithms. In SoutheastCon, 2016 (pp. 1-7). IEEE.

[18] Vaarandi, R. (2004). A breadth-first algorithm for mining frequent patterns from event logs. Intelligence in Communication Systems, 293-308.

Asmaa Ez-zaidi received his Diploma in Computer Engineering from Cadi Ayyad University, Morocco. She is currently a PhD student in the Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Techniques, Cadi Ayyad University, Marrakesh, Morocco

Said Rakrak Professor and researcher in the Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Techniques, Cadi Ayyad University, Marrakesh, Morocco