Mobile Object Tracking Techniques in Wireless Sensor Networks

Marjan Naderan, Mehdi Dehghan and Hossein Pedram
Computer Engineering and Information Technology Department
Amirkabir University of Technology (Tehran Polytechnic)
Tehran, Iran
E-mail: {naderan, dehghan, pedram}@aut.ac.ir

Abstract— Mobile object tracking in wireless sensor networks (WSNs) has gained much attention during recent years due to the special characteristics of these networks. Considering the wide application of tracking in WSNs, various problems of data aggregation, routing, scheduling and energy conservation have been revisited and new solutions have been proposed recently. Due to the importance of this topic and the availability of a significant body of literature, a detailed survey becomes necessary and useful at this stage. In this paper we present a survey of the state-of-the-art mobile object tracking techniques in WSNs. We give a definition for the general problem of object tracking and introduce challenges in designing an efficient object tracking sensor network (OTSN). We highlight the advantages and performance issues of the existing tracking methods and make a brief comparison based on the design parameters proposed. Finally, we will present open problems and the future works conceivable in this broad field of research.

Keywords—sensor networks; object tracking; multi-target tracking; quality of tracking; OTSN

I. INTRODUCTION

Wireless sensor networks (WSNs) are networks in which nodes have the ability of sensing one (or more) physical phenomenon and communicating wirelessly, with limited capability of computation and battery resources. Two special applications, monitoring and tracking of mobile objects, have been marked as the two major categories [1]. Tracking of mobile targets may include tracking of enemy, animals, humans and cars in highways.

The object tracking problem had been investigated previously for robots and Personal Communication Services (PCS) networks [2]-[5]. Research on distributed tracking via few numbers of large sensors (e.g., airborne radars) began with the DARPA Distributed Sensor Networks (DSN) program early in 1980's [6]. For large WSNs, issues of imperfect transmission, medium access control (MAC) and energy constraints have attracted attention recently. A wireless sensor network used for tracking targets is called an Object Tracking Sensor Network (OTSN) [7].

In this paper, we investigate current challenges in OTSNs and will develop a classification for the techniques proposed in recent years. Our objective is to provide a deeper understanding of the problem of mobile object tracking in WSNs and to show that several important factors have to be considered in designing an object tracking protocol. In addition, we elaborate on some properties of current techniques and will introduce some open research issues that can be further pursued. To the best of our knowledge this is the first work of doing a complete survey in the area of object tracking in wireless sensor networks. Considering the broad range of topics in the field of distributed tracking and due to the increasing trend in relevant works during recent years, providing expository surveys on this area is even more encouraged.

The rest of the paper is organized as follows: in section II we define the general problem followed by principle OTSN design factors. In section III we classify the tracking methods in sensor networks. In section IV we give a comparative discussion of the proposed methods. In section V we propose some interesting open research problems and challenges, and finally in section VI we conclude the paper.

II. PROBLEM DEFINITION AND EVALUATION METRICS

A. Problem definition

Problem scenario: An OTSN consists of \( n \) wireless sensor nodes distributed in a region for tracking \( m (m \geq 1) \) mobile objects. In general, nodes can be deployed across the region with no specific assumption made about their placement. Sensor nodes can detect the presence or absence of these targets by sampling the sensed signals from the objects. Each object enters and exits the region at a random time and at a random place, stays in that place with some probability \( p \) for some duration, moves independently from the others and disappears with probability \( 1-p \). Generally we have no assumptions about the mobility models of targets. A more detailed mathematical definition can be found in [8].

Objectives: It is desirable to determine the locations of the objects. Through performing this operation periodically, we can track the trajectories of the objects in time. Sensing nodes that detect the target(s) are supposed to send reports towards the monitoring application. Only a small number of nodes need to be active during each period so that the energy spent is minimized and consequently the network lifetime is maximized. Furthermore, it is not desirable to miss the trajectories of the targets and in case of a loss, there must be some recovery mechanism with bounded error.
The region including active nodes is referred to as the monitoring region [7][9]. In case of using prediction methods, it is possible to activate nodes in regions that the targets are more likely to move on to, referred to as the forwarding region [10]. In [11], the tracking problem is viewed as forming a region with active sensors that besieges the target at any time, and is maintained as the mobile object moves. Such region is also termed as the envelopment net in [12], and its associated model is dubbed the Frisbee model [11].

The design of tracking protocols in WSNs is influenced by many challenging factors. We divide these factors into two groups which will be described in the following two subsections.

B. Effect of other layers and services on algorithm design

Several assumptions about the functionality of the other layers and services will have a great influence on designing an object tracking protocol and vice versa. A tracking method can be implemented on top of the network layer, as it may need routing information, or alternatively, it may be part of the network layer. Some of the functionalities of the other layers and services are as follow:

Routing and querying protocols: An important factor in the routing protocol used for object tracking is the existence of correlation between data of neighboring nodes. If such a correlation exists, new routing metrics for selecting paths can be proposed. For example, in [49], a new link metric is proposed that characterizes the detection error exponent. In [21], a specific routing mechanism is proposed based on information derived from sensor data. However, most of the existing tracking methods do not consider this correlation between the sensor data. Instead, in most of them, such as [9][12][13][17][18], it is assumed that either a routing algorithm is available for sending reports towards the sink, or in methods such as [7][20][43][44], the information is delivered to the sink simply via one hop from the cluster head (or tree root) to the sink. In some works, such as [9][12][13][17][20], the tracking request is assumed to be flooded from the sink to the network or even no explicit mention of this problem has been made at all.

Aggregation and data fusion strategies: Aggregation is normally done on data of the same type and its goal is to reduce redundancy and to compress data so that the energy consumption is reduced during communication. Data fusion, on the other hand, is a distributed process on several sensors data and its goal is to obtain an inference about the gathered data and to reduce the error by eliminating noisy measurements. A significant body of literature exists on aggregation and data fusion techniques, e.g., surveys proposed in [22] and [62]. Some recent aggregation methods have also been presented in [23] and [24]. However, in most of the existing tracking methods, no specific aggregation or fusion strategy is proposed or none of the existing approaches are incorporated into the final solution.

Mobility model of the target: If the movement model or the velocity of the target is known by the nodes, or at least they can predict the movement trajectories of the targets, as in [7][20][25][26], targets are less likely to be missed. Nevertheless, it is more desirable not to rely on advanced information on the objects movements. It should also be noticed that not every mobility model can be applied to every environment (e.g. [19]).

Node placement or the location model of nodes: The fact that the nodes are placed randomly or are deployed in a pre-planned fashion in the monitoring region is an important consideration in designing the tracking algorithm. Some approaches regarding this factor in object tracking can be found in [18] and [27].

Localization algorithm: Localization algorithms can be applied in two stages in object tracking techniques: to estimate the location of the object based on the locations of detecting nodes, or as well to determine the location of nodes in the initial phase of the network operation. Most tracking protocols draw on simple localization algorithms to determine the location of the targets, such as [12][13][17][18], or it might as well be simply assumed that the target location be approximated by the location of the nearest node [9][19].

Scheduling algorithm: The sleep mode in sensor networks and its management through a scheduling algorithm is considered as a mechanism to conserve energy and increase the network lifetime. Two primary approaches have been proposed till recently for using sleeping sensors [28]: the sleeping sensors can be woken up by external means on an as-needed basis, as proffered in [7][20][29], or as an alternative, modifications have been applied to MAC protocols as outlined in [30] and [31]. In [28], information about the object trajectory is exploited for scheduling purposes. However, it is argued in [28] and [32] that at special times during which nodes are in sleep mode, the tracking error might increase.

MAC protocols: The MAC protocol is more general than the scheduling algorithm which is only in charge of managing the on/off periods of the radio or the sensing part of the sensor nodes. Usually, there are no assumptions about functionality of the MAC layer in object tracking strategies. In fact using cross-layer information can greatly improve the performance of a tracking algorithm at the expense of designing a special-purpose MAC protocol. The work in [13] is the only study which specifically envisions a MAC layer protocol and clearly determines its interactions with the other layers.

C. Metrics related to quality of tracking

In this section, we define metrics dealing with the quality of tracking. Some of these metrics are used for evaluating tracking algorithms also in the form of simulation measures, for instance in [7][12][13][17][19][20][30]:

Scalability of the tracking protocol: This metric mostly includes increasing the number of sensors, tracking multiple targets, increasing the query requests for different regions, different objects or more accuracy, and finally increasing the monitoring region under cover. Algorithms that can track multiple targets are referred to as Multi-Target Tracking (MTT) algorithms. For some algorithms several properties must change if the MTT functionality is desired, e.g., in [13] and [14]. Tracking multiple objects simultaneously is closely related to the data association problem in which measured
data are associated to each target (track) with uncertainty [15]. Here, we do not further discuss this issue as it requires exploring into the signal processing field.

**Tracking precision:** This means how far the estimated location of each target, derived from samples and localization algorithm, differs from its real location. Other related parameters for tracking precision are probability of target missing (or miss probability) [7][20][25], prediction accuracy [27], sensitivity to noisy measurements [17] as well as the ratio of false alarms.

**Tracking delay:** This metric plays a key factor in two stages of the tracking operation: I) the time it takes to estimate the location of a given target from the very moment it appears, and II) the latency of reporting the estimated location to the sink. The delay parameter in the second phase is mostly inversely related to this metric. It is obvious that including or excluding the target, the tracking algorithm does not depend on a specific target mobility model, or that it is able to track higher speed targets while incurring little performance degradation is an important factor in evaluating tracking protocols.

**Energy consumption:** The lifetime of the network is inversely related to this metric. It is obvious that including or not including any metric for quality of tracking can affect energy consumption of the nodes. Thus, there must be some kind of a tradeoff between the desired metric(s) and the consumed energy.

**Adaptability of the tracking algorithm:** The fact that the tracking algorithm does not depend on a specific target mobility model, or that it is able to track higher speed targets while incurring little performance degradation is an important factor in evaluating tracking protocols.

**Degree of coverage:** This means what proportion of the monitoring region is covered by nodes with a given sensing range to track the target efficiently. For example, as shown in [30], it is possible to reach acceptable Quality of Surveillance (QoSv) with a partial coverage of the monitoring region. Several other existing works inspecting this aspect of target tracking have been proposed in [20][30][32][36][37].

**Protocol overhead:** This metric entails the control packets overhead and the preprocessing cost of the algorithm. Usually the protocols that rely on a specific structure (e.g., [9][12][17]-[19]) have more control packets than unstructured protocols (e.g., [13][40]) which is due to the initial cost of the structure setup and/or its maintenance over time.

Table I gives a summary of the considerations and metrics discussed in this section and depicts the relation between factors affecting the algorithm design and their corresponding quality of tracking metrics. As can be seen from the table, protocols in different layers are tightly coupled with several different quality of tracking parameters. For instance, while relying on advanced information about the mobility model of the target(s) may limit the scalability or even adaptability of the tracking algorithm, it might instead help decrease the energy consumption, which is evidently a desirable quality of tracking metric.

<table>
<thead>
<tr>
<th>Layers and services</th>
<th>Quality of tracking metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing, querying,</td>
<td>- scalability of algorithm</td>
</tr>
<tr>
<td>aggregation and data fusion strategies</td>
<td>- tracking precision</td>
</tr>
<tr>
<td>strategies</td>
<td>- tracking delay</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- control packets overhead</td>
</tr>
<tr>
<td>Mobility model of the target</td>
<td>- scalability of algorithm</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- adaptability of algorithm</td>
</tr>
<tr>
<td>Node placement or location model</td>
<td>- tracking precision</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- coverage of the region</td>
</tr>
<tr>
<td></td>
<td>- control packets overhead</td>
</tr>
<tr>
<td>Localization algorithm</td>
<td>- tracking precision</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- control packets overhead</td>
</tr>
<tr>
<td>Scheduling algorithm</td>
<td>- scalability of the algorithm</td>
</tr>
<tr>
<td></td>
<td>- tracking precision</td>
</tr>
<tr>
<td></td>
<td>- coverage of the region</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- control packets overhead</td>
</tr>
<tr>
<td>MAC protocols</td>
<td>- scalability of the algorithm</td>
</tr>
<tr>
<td></td>
<td>- nodes energy consumption</td>
</tr>
<tr>
<td></td>
<td>- control packets overhead</td>
</tr>
</tbody>
</table>

### III. CLASSIFICATION OF MOBILE OBJECT TRACKING METHODS IN WIRELESS SENSOR NETWORKS

Mobile object tracking algorithms in WSNs can be classified into two major groups of algorithms:

- Signal processing algorithms
- Network protocols

Signal processing algorithms or more specifically Collaborative Signal Processing (CSP) are mostly used for detection and classification of targets. In most of these methods, it is assumed that nodes are grouped into clusters and only one node in charge of gathering data and tracking the target, e.g., [6][8][14]-[16][29]. These algorithms reduce energy consumption by activating only a subset of nodes and leaving other nodes in sleep mode [12].

Algorithms of the second group mostly concern the communication pattern among sensor nodes aimed for gathering data and estimating the location of the targets. These protocols can be implemented in the network layer or above within the communication architecture while signal processing algorithms reside at the upper (application) layer. For example, the method discussed in [38] can be implemented on top of routing protocols such as Directed Diffusion [39] as stated in [16] and [40].

Network protocols for mobile object tracking can in turn be classified according to three different aspects: 1) network structure, 2) number of the objects, and 3) type of the objects. Figure 1 depicts this classification.

#### A. Classification with respect to network structure

Regarding the network structure point of view, we can classify network protocols to three main classes:

- Leader-based protocols
- Tree-based protocols
- Cluster-based protocols

Each protocol in the aforementioned classes can optionally be combined with prediction strategies which help reduce both the energy consumption and the probability of target miss. Table II shows the protocols proposed till recently with our classification based on the structure of the network.

**Table II. Classification of object tracking methods according to network structure**

<table>
<thead>
<tr>
<th>Structure type</th>
<th>Method</th>
</tr>
</thead>
</table>
| Cluster-based        | - Dynamic clustering method for acoustic sensors [17]
                        | - DCS [43] and its optimization CODA [44]  |
| Tree-based           | - DAB [19] and its optimizations DAT and Z-DAT in [44]
                        | - DCTC [9] and its optimization in [46]     |
| Leader-based         | - DOT [12]                                  |
                        | - LESOP architecture [13]                    |
                        | - Mobile agent based method [18]             |
                        | - IDSO [38][40]                             |
| Cluster-prediction   | - PES [7] and DRP [27]                       |
                        | - DPT [20]                                  |
                        | - PREMON [26]                               |
| Tree-prediction      | - DCTC [9] (prediction-based scheme)         |

Leader-based methods resemble cluster-based methods with the only difference that no specific structure is setup amongst nodes and only those detecting the target participate in the tracking process.

It is worth mentioning that traditional clustering algorithms, such as [41] as well as those stated in [42], cannot be applied directly for object tracking methods; due to the fact that they do not guarantee uniform distribution of cluster heads throughout the network. Besides, as a requirement for object tracking methods, the cluster head or the leader node must be the one nearest to the target, so that it would be able to provide better estimations of the target locations and thus to reduce the energy consumption.

**C. Classification with respect to the type of the objects**

In this class, object tracking methods can be divided into two groups: continuous object tracking and discrete object tracking. By a continuous object we mean objects such as wild fire or a bio-chemical material [43][44]. Continuous objects are different from ordinary ones in that they continuously extend across the region and usually occupy a large area. They tend to diffuse, increase in size, change in shape or even split into multiple relatively smaller continuous objects. The most challenging problem in tracking continuous objects is the fusion and/or dissemination of local boundary information [48] which must be performed in a dynamically adaptable fashion.

**IV. COMPARISON AND DISCUSSION ON METHODS**

In this section we compare the object tracking methods mentioned in the previous sections according to some of the criteria discussed in sections II.B and II.C. Table III summarizes the results of this comparison. Due to space limitations, we skip giving synopsis of each object tracking method and the interested reader is referred to the citations mentioned in figure 1 and table II.

In general, compared to cluster and leader-based methods, tree-based algorithms have more cost for setting up, for the maintaining the infrastructure as well as for sending information from leaf nodes back to the root (usually along a multi-hop path). For example, DCTC involves heavy message exchanges and broadcasts which are not desired when the data rate and/or target speed is high [23]; moreover, the consumed energy increases with increase in the number of nodes, the monitoring area and the number of targets.

![Fig. 1. Mobile object tracking methods in WSNs: a taxonomy](image-url)
DPT, PES and DPR make use of prediction together with a cluster structure amongst nodes and as a result, operate better from the energy consumption standpoint. DOT and the mobile agent-based method discussed in [18] reserve unnecessary nodes in sleep mode because of using special structures, and thus they operate similarly to cluster-based methods when it is viewed from the energy usage perspective. In LESOP, no specific structure exists amongst nodes; however, by using a prediction-based scheme, the next leader is determined and other nodes are at sleep mode. Thus, we can conclude that to reduce energy consumption of nodes, we have to use special structures among nodes or use prediction schemes.

Losing the target and the recovery mechanism used in response are other parameters to consider in comparing tracking methods. In prediction-based schemes such as DPT, PES and DPR, the probability of losing the target is reduced, although not completely omitted, compared to methods which do not use prediction. In addition, these methods utilize recovery mechanisms in case of losing target. In DOT, increasing the wakeup frequency of nodes decreases the probability of target loss. In methods such as DCTC, DAB and the method proposed in [17], increasing node density in the monitoring area and constructing trees or clusters with dense nodes, is an important factor to decrease the target loss probability.

Another parameter for comparing target tracking schemes is the control overhead. In most tree-based and cluster-based methods, the preprocessing stage consumes energy at the initial phase of the network operation. Among the leader-based methods, LESOP has no preprocessing cost, but DOT and the mobile agent-based method, have a preprocessing stage to compute the underlying structure.

Considering number of control packets in the protocol overhead is another important factor for comparison. As it can be seen from table III, for some methods such as DPT, PES and DPR, no specific clustering algorithm is mentioned and hence, the exact number of control packets can not be specified.

Support for multi-target tracking is another issue to compare object tracking techniques. As shown in table III, most techniques embrace multi-target tracking. Some methods do not directly address this issue but have facilities to enable the system to track multiple targets.

For instance in [13] orthogonal radio channels are needed in the algorithm to support multi-target tracking; which can be provided by TDMA or FDMA methods. In [17], signature information provided by *signature packets*, increase system robustness when one deals with multiple targets. Furthermore in [47] a group formation algorithm is proposed to track multiple targets for the IDSQ method.

The routing algorithm used for sending tracking reports to the sink is another important consideration in assessing target tracking solutions. In the mobile agent-based method in [18] two methods, namely TB and DB, are presented that allow the master node to decide whether to send its report to the gateway (sink) or wait and carry the results until a special condition is met. CADR [21] is a routing strategy for directing queries to the source of information and vice versa. Other tracking methods have not considered this parameter and assume that tracking reports are delivered to sink(s) via one-hop communication or a specific routing algorithm.

### Table III. Comparison of Object Tracking Methods

<table>
<thead>
<tr>
<th>Classification</th>
<th>Metric Method</th>
<th>Deactivating non-necessary nodes</th>
<th>Missing the target</th>
<th>Recovery mechanism</th>
<th>Protocol overhead</th>
<th>MTT support</th>
<th>Routing/Aggregation protocol</th>
<th>Localization algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Leader-based methods</strong></td>
<td>LESOP</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>3</td>
<td>x</td>
<td>Linear combination of sensors locations</td>
</tr>
<tr>
<td>DOT</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td>✓</td>
<td>Trilateration</td>
</tr>
<tr>
<td>DCTC</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td>6</td>
<td>x</td>
<td>Nearest sensor location to the target</td>
</tr>
<tr>
<td>DAB</td>
<td>x</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
<td></td>
<td></td>
<td>x</td>
<td>Nearest sensor location to the target</td>
</tr>
<tr>
<td>DCS &amp; CODA</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td>✓</td>
<td>2 + cluster construction</td>
<td>Continuous object</td>
<td>Sink computes object’s boundary</td>
</tr>
<tr>
<td>Method in [17]</td>
<td>✓</td>
<td>x</td>
<td>x</td>
<td>✓</td>
<td></td>
<td>5</td>
<td>✓</td>
<td>Voronoi diagram based, and nonlinear optimization-based</td>
</tr>
<tr>
<td>DPT</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>8 + cluster construction</td>
<td>✓</td>
<td>Linear prediction</td>
</tr>
<tr>
<td>PES &amp; DPR</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>6 + cluster construction</td>
<td>✓</td>
<td>BS computes target’s location</td>
</tr>
</tbody>
</table>
Two other factors in comparing tracking methods are the location information of nodes as well as the localization algorithm used for estimating locations of targets. In all methods, it is assumed that nodes know their location and they usually exchange this information with their neighbors. Nodes know their locations if this information is set priori in them. Otherwise knowledge of location information can be achieved by a localization algorithm, which its overhead can be added to the preprocessing stage. In some methods such as DCS, CODA, PES and DPR a central node (e.g., "sink" node in DCS and CODA, "BS" in PES and DPR) which knows the locations of all nodes, computes target boundary or locations of targets.

V. OPEN RESEARCH PROBLEMS AND FUTURE WORK

In spite that the field of object tracking has received sizable attention recently, in this section we provide some interesting open research issues which have not been covered yet.

As new object tracking methods are proposed by researchers, classification with respect to a richer set of criteria will become necessary. For example in our ongoing work, classification with respect to the modality of sensors has great importance, as we are exploiting properties of these methods that depend on the type of sensors. Another classification could be with respect to the functioning layer as some methods operate at higher layers, or even as is the case with LESOP, the authors have proposed a two-layered architecture which is specifically intended for object tracking.

In addition, a great number of object tracking methods proposed in the field of signal processing are also in need of classification, which might as well be done on the basis of different important factors, such as their estimation algorithms. Other interesting topics in the field of signal processing algorithms include data fusion strategies and the problem of identity management; the latter has been introduced recently in works such as [8] and [16].

When considering the effects of other layers and protocols, one other issue which comes to mind is the topic of multi-task protocols in sensor networks. In this kind of communication, also referred to as Mobicast, data transmission takes place between groups of nodes. Works discussed in [10] and [60], and more recently in [61] deal with this problem. More in-depth assessment is required on the effect of these protocols on object tracking methods and vice versa.

The problem of multi-target tracking protocols has to be investigated more to consider application level tasks and not only network level functions such as coverage or data dissemination. An important application level task recently introduced in [48], and complemented in [58] and [59], is multiple mission assignment. In these works, a sensor node may be assigned to at most one of the missions that is applicable, but it is legal for a mission to accept utility from multiple sensors. The problem that arises in this kind of multiple mission assignment is node reassignment, i.e. a node that is assigned to one mission may later be reassigned to another mission because it is more useful there. Effects of such node reassignments on missions and the pertaining costs need to be studied. The case of assigning a sensor to more than one mission, as a generalization case of the multiple mission assignment problem in [58] and [59], is yet another important direction of research.

Another issue in the field of multi-target tracking is the deployment of multi-modal sensors. A network consisting of multi-modal sensors is able to detect different targets. In such networks it is desirable to activate the most appropriate modality in a specific sensor and then have it assigned to a mission. To the best of our knowledge this issue has not been approached before.

Another significant topic of interest is optimization of tracking protocols with respect to the network structure as well as application specific requirements (e.g., multi-target tracking). Such optimization might require the mapping of quality of tracking parameters defined in the application level to lower level parameters, so that performance of tracking protocols can be evaluated in terms of these low level parameters. Currently we are working on defining specific quality of tracking parameters such as robust detection and maximum target separability for the MTT problem. Similar ideas are also applicable for clustering techniques especially for MTT or for designing MAC protocols [50].

In addition to the quality of tracking parameters defined in this paper, other parameters of interest are: reliability of tracking in the presence of noisy measurements [51], reliable delivery of a large number of measurements to the sink with minimal human intervention [52], fault tolerant methods in detection and classification of targets [53][54], and finally extracting the target movement patterns in predicting the future movement of target to reduce errors with data mining methods [25][55][48] or with Markov models [57].

Another topic of interest is the performance evaluation of OTSNs with precise mathematical models in tracking targets with respect to quality of tracking parameters. Evaluation or comparisons of tracking methods with different structures such as tree, cluster or leader have also not been studied yet.

As the application of tracking mobile objects using wireless sensor networks moves away from military and industrial areas into crowded urban and natural environments, new problems and issues arise in this field.

VI. CONCLUSION

In this paper we investigated the problem of mobile object tracking methods in wireless sensor networks. We introduced important design factors in the relation to the other layers of the communication architecture as well as parameters concerned with the quality of tracking. Furthermore, we classified tracking methods according to various criteria and made a brief comparative study of these methods according to some of the mentioned criteria. The initial findings drawn from this comparison can be listed as follow:

- Tree-based methods have more cost compared to cluster and leader-based methods.
- Scalability of tree-based protocols is more limited than cluster and leader-based methods as a result of tree expansion.
In order to inactivate unnecessary nodes, it is strongly recommended to exploit either regular structures or prediction-based schemes.

We believe that a broad field of open research issues exist for object tracking in WSNs, in addition to problems of existing methods, some of which we have discussed in this paper.

REFERENCES


