

A Survey on Target Tracking Techniques in Wireless Sensor Networks

K. Ramya¹, K. Praveen Kumar², and Dr. V. Srinivas Rao³

¹Student, Department of CSE, VR Siddhartha Engineering College
kondavetiramy@gmail.com

²Sr. Asst. Professor, Department of CSE, VR Siddhartha Engineering College
praveenman@gmail.com

³Professor & Head, Department of CSE, VR Siddhartha Engineering College
hodcse@vrsiddhartha.ac.in

Abstract:

Target Tracking as it moves through a sensor network has become an increasingly important application in Wireless Sensor Networks. This paper examines some of the target tracking techniques in use today. An analysis of each technique is presented along with its advantages, problems and possible improvements. There are seven main categories explored in this paper. The survey promotes overview of recent research literature along with their performance comparison and evaluation.

Keywords: *Wireless Sensor Networks, target tracking, cluster, prediction, mobicast.*

I. INTRODUCTION:

The continuous evolution in wireless sensor network technology make it possible to implement the wireless sensor network (WSNs) in a variety of scenarios. WSNs consist of thousands of tiny sensor nodes deployed in a physical environment for observation of an event of interest. The sensors in the vicinity of an event must be able to monitor it and report back to the sink. A sink sensor node has capability to communicate with outside world such as laptop, base station. Sensor nodes have been deployed to play significant roles in traffic control, battlefield, habitat monitoring and intruder tracking in recent years. The traditional target tracking methods for Wireless Sensor Networks make use of a centralized approach. As the number of sensors rise in the network, more messages are passed on towards the sink and will consume additional bandwidth. Thus this approach is not fault tolerant as there is single point of failure and lacks scalability. Moreover in traditional target tracking methods, sensing task is usually performed by one node at a time resulting in less accuracy and heavy computation burden on that node. In WSNs each node has very limited power; consequently traditional tracking methods based on complex signal processing algorithms are not useful.

In a target tracking application, the sensor nodes which can sense the target at a particular time are kept in active mode while the remaining nodes are to be retained in inactive mode so as to conserve energy until the target approaches them. To continuously monitor mobile target, a group of sensors must be turned in active mode just before target reaches to them. This group of

active sensors varies depending on the velocity of moving target and schedule from cluster head. Ultimately, target tracking in course of maintaining the balance between network resources like energy, bandwidth, and overheads is challenging.

The purpose of this paper is to introduce, summarize and compare some of the target tracking algorithms currently used in sensor networks.

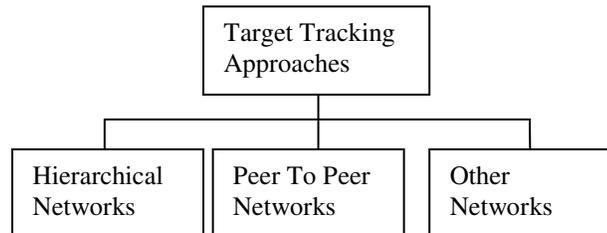


Fig: Types of Target Tracking Approaches

2. TAXONOMY OF TARGET TRACKING

In WSNs Moving object tracking has received Considerable attention in recent years and the solutions can be mainly classified into five schemes, such as tree - based tracking, cluster-based tracking; prediction-based tracking; mobicast message-based tracking and hybrid methods and Activation based methods.

2.1 Tracking methods for Hierarchical networks:

In Hierarchical networks mesh-based systems with multihop radio connectivity among or between wireless nodes are employed. The sensors in the vicinity of an event must be able to monitor the event of interest and report back to the sink. A sink sensor node has capability to communicate with outside world such as laptop, base station. The important characterizations of the WSNs are that (i) sensor nodes can support communications on behalf of other sensor nodes by acting as repeaters; (ii) the forwarding node can support data processing or information fusion on behalf of the sensor nodes. Tree-Based Target Tracking, Cluster Based Target Tracking, Hybrid method and Activation methods are belongs to Hierarchical Networks.

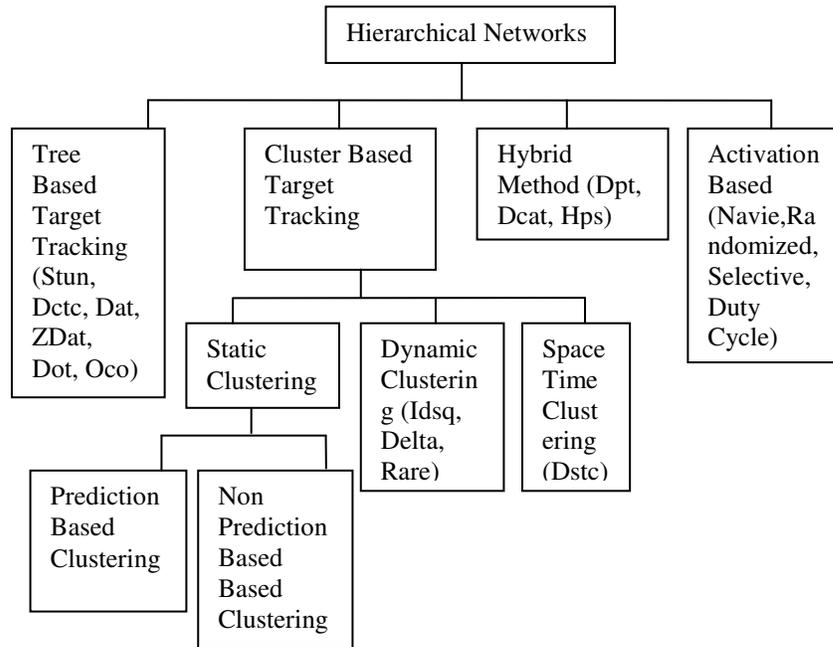


Fig: Classification of Hierarchical Networks

2.1.1 Tree-Based Target Tracking:

In Tree-Based Target Tracking, nodes in a network may be organized in a hierarchical tree structure or represented as a graph. The vertices in the structure represent sensor nodes and edges are links between nodes that can directly communicate with each other.

While tracking a target the nodes that detect the target communicates with each other and selects a root node. The root node collects information from all the nodes via a distributed spanning tree. If the root node is far away from the target, then the tree will be reconfigured. Although the spanning tree based approaches track the moving objects more accurately, tree organizations result in high-energy consumptions.

The centralized target tracking approaches are both time and energy consuming; to avoid this limitation tree-based tracking methods are proposed. They are Scalable Tracking Using Networked sensors (STUN) [1], Dynamic Convoy Tree-Based Collaboration (DCTC) [2],[3], Deviation Avoidance Tree (DAT)[4], and Dynamic Object Tracking (DOT) approach[6], Optimized Communication & Organization (OCO) [7].

2.1.1.1 Scalable Tracking Using Networked sensors (STUN):

Scalable Tracking Using Networked sensors [1] is one of the tree based approach developed by Kung & Vlah. In this the cost is assigned to each link calculated by Euclidean distance between the two nodes. The leaf nodes are used for tracking the moving object and then sending collected data to the sink through intermediate nodes. The intermediate nodes keep a record of

detected object and whenever there is a change in that record, the nodes send updated information to the sink. The performance of STUN is determined by the structure of its message-pruning hierarchy and the tracked objects mobility pattern. STUN relies on a tree structure called “Drain and Balance Tree”. Drain And Balance is a method that constructs the hierarchical structure of STUN based on the expected frequency of object movements over a region [1]. Drain and Balance method constructs the tree in a bottom-up fashion, i.e., from leaves to the root, through a series of DAB steps. Within each DAB step, a subset of the sensors is merged into balanced subtrees, without regard for event rates. The effectiveness of the method comes from properly choosing the nodes to merge in each of these steps. STUN has some drawbacks. First, drain and balance tree does not replicate physical sensor network as it is a logical tree, hence an edge may consist of multiple communication hops. So it may raise communication cost. Second, the construction of DAB tree does not consider query cost.

2.1.1.2 Dynamic Convoy Tree-Based Collaboration (DCTC):

DCTC [2] first detects the target and monitors it by tracking the surrounding area of the target. It relies on a tree structure called convoy tree, that include sensor nodes around the moving object, and the tree is configured to add some nodes and prune some nodes as the target moves. The target first enters the detection region, sensor nodes that can detect the target collaborate with each other to select a root and construct an initial convoy tree. Root node collects information from the sensor nodes and refines this information to obtain more complete and accurate information about the target using some classification algorithms. A big challenge of implementing the DCTC framework is reconfiguration of the convoy tree in an energy efficient way, when the target is moving. Some of the schemes optimize the energy consumption but those are not practical [3]. Some of the practical solutions to implement the DCTC framework are constructing the Initial Tree, Tree Expansion and Pruning, Tree Reconfiguration.

2.1.1.3 Deviation Avoidance Tree (DAT) [4]:

Object tracking typically involves two basic operations: update and query. Location updates and queries may be done in various ways. Generally Updates are initiated when the object moves from one sensor location to another sensor location. For querying one way is Delivering a query to flood the whole network. This process is inefficient because a considerable amount of energy will be consumed when the network scale is large or when the query rate is high. Another method is locating all information in a specific sensor (Sink).Then no flooding is required. One drawback is that when objects move frequently, abundant update messages will be generated. To overcome this “Drain and Balance Tree Structure” is proposed. In this Query messages are not required to be flooded and update messages are not always transmitted to the sink. But it has some drawbacks. DAB tree is a logical tree, not reflecting the physical structure of the sensor network. Communication cost is high due to edges consisting of multiple communication hops. And the construction of DAB tree does not take the query cost into consideration. The proposed methods to overcome these issues are Deviation-Avoidance Tree (DAT) [4] and Zone-based Deviation-Avoidance Tree (Z-DAT) [5].

DAT [4] treats each node as a singleton sub tree. More links can be included to connect all sub trees together. At the end all the subtrees will be connected into one tree. DAT is a two stage approach. The first stage aims at reducing the update cost, while the second stage aims at reducing the query cost.

2.1.1.4 Zone-based Deviation-Avoidance Tree (Z-DAT):

Z-DAT [5] is similar to DAT except that it examines links in a different order. The DAT and Z-DAT almost reduce the update cost but sometimes fails to reduce the query cost. To overcome this problem Query Cost Reduction (QCR) is designed. It reduces the total update and query cost by adjusting the object tracking tree obtained by DAT/Z-DAT.

2.1.1.5 Dynamic Object Tracking (DOT):

DOT [6] is one of the protocol that reports the tracking information of moving object to moving source. The face neighbors are identified by Gabriel graph. In target discovery, source sends request to sensor nodes and the node close to the target replies back. To detect moving target continuously, the spatial neighbors of near sensor nodes are waken up. In target tracking, source send query to beacon node (node keeping track information), which reply back target's next location and the source moves towards next beacon node. The process is repeated until the source catches the target.

CLOUD framework: Cloud framework is mainly used in region based tracking. A tree-based collective structure is formed for each event region in each time slot. In Tree based structure nodes may be organized in a hierarchical tree structure or represented as a graph. But the approach is limited by the dependence on the tree structure for the network topology.

R-tree sensor network topology: R-tree sensor network topology is adopted for the detection and tracking of region-based targets. It consists of Forward-all and forward-description approaches. The problems with this topology are detection of an event boundary, merging the event region obtained from the child nodes in the R-tree, and simplifying and smoothing the event boundary. To overcome these problems three algorithms are proposed. Those are boundary detection algorithm, merging algorithm and description improvement algorithm.

2.1.1.6 Optimized Communication and Organization (oco):

Optimized Communication & Organization [7] is a tree-based method for target tracking. It provides self-organizing and routing capabilities with low computation overhead on sensor nodes. It consists of four steps. Those are position-collecting, processing, tracking, and maintenance. The position-collecting phase involves the base sensor node collecting positions of all the nodes in the network. The processing phase involves the base cleaning up the redundant nodes, detecting the border nodes, and routing. The tracking phase detects all objects coming from outside the perimeter of the sensor network. Normally, only the sensor modules of the border nodes are ON. When a border node detects an object, it periodically sends its position information to the base sensor node by using its father's ID. When it has lost the object, it sends a message to turn on the sensor modules of all its neighbors. If a neighbor detects the object, it will continue sending its position to the base sensor and, right after it has lost the object, it turns the sensor modules of all its neighbors to ON, and so on. If activated neighbors detect nothing, they automatically turn the sensor module OFF after a short interval. This way, the objects are tracked as long as it remains within the perimeter of the network. The maintenance phase is started when the network has a dead node (for example, the node has run

out of its energy). In this case, the base deletes the dead node from the list, and then re-organizes the network by starting the four-step procedure again. The main drawback of this approach is high energy consumption.

2.1.2 CLUSTER BASED:

Cluster-based method divides the network into clusters to support collaborative data processing. A cluster consists of cluster head and member sensor nodes. When a sensor detects an object it volunteers to act as a CH. There is No need of explicit election of leader. So message exchanges are not incurred. If more than one powerful sensor may detect the signal, multiple volunteer nodes may exist. So a decentralized approach has to be applied to ensure that only one Cluster Head is active in the vicinity of a target to be tracked with high probability. Cluster based method is divided into three types. Static Clustering, Dynamic Clustering and Space Time Clustering. Static clustering is divided into Prediction based clustering and Non Prediction based Clustering [10]. Dynamic Clustering consists of some methods like IDSQ [11], DELTA [12], and RARE [13]. DSTC [14] is one of the Space time clustering method.

Low-Energy Adaptive Clustering Hierarchy (LEACH):

Low-Energy Adaptive Clustering Hierarchy [8] is used to reduce energy consumption. In LEACH method sensor nodes formed as clusters and choose one of them as cluster-head. Sensor node first detects the target and sends the data to its cluster-head. Then the cluster head aggregates and compresses the data collected from all the nodes and sends it to the base station. The Cluster head requires more energy than other nodes in the network. So LEACH uses random rotation of the nodes required to be the cluster-heads to evenly distribute energy consumption in the network. LEACH operations can be divided into two phases one is Setup phase and another one is Steady phase. In the setup phase clusters are formed and a cluster head is chosen for each cluster. In the steady phase, data is sensed and sent to the central base station.

Continuous Object Detection and Tracking Algorithm for Wireless Sensor Networks (CODA):

The Continuous Object Detection and Tracking mechanism [9] enables each sensor node to detect and track the moving boundaries of objects in the sensing field. Monitoring continuous objects without incurring excessive communication costs requires an efficient target detection mechanism. In developing the CODA mechanism, a static backbone comprising a designated number of static clusters is constructed during the initial network deployment stage. In each static cluster, any sensor detects the object in their vicinity transmit the detected information directly to the CH. Upon receiving this information, the CH executes a local boundary estimation function to determine the boundary sensors of the continuous object boundary which lies within its cluster. After forming these boundary sensors into a dynamic cluster, it then sends the boundary information of this dynamic cluster to designated sinks. CODA has two principal advantages. First, the boundary sensors of the continuous objects are identified by the CHs of the static cluster rather than via a process of message exchange among the local sensors, and thus the communication overhead is substantially reduced. Second, selecting the CHs in

each dynamic cluster from the boundary sensor set is not necessary. Consequently, an explicit header election scheme is not required and excessive message exchanges are avoided.

2.1.2.1 STATIC CLUSTERING:

In Static Clustering [10] clusters are formed statically at the time of network deployment. The attributes of each cluster, such as the size of a cluster, the area it covers, and the members it possesses, are static. In spite of its simplicity, the static cluster architecture suffers from several drawbacks. First, fixed membership is not robust from the perspective of fault tolerance. If a Cluster Head dies of power depletion, all the sensors in the cluster render useless. Second, fixed membership prevents sensor nodes in different clusters from sharing information and collaborating on data processing. Finally, fixed membership cannot adapt to highly dynamic scenarios in which sensors in the region of high event concentration may be instrumented to stay awake.

Static clustering can be divided into Prediction Based or Proactive Clustering and Non Prediction Based Clustering.

Prediction Based or Proactive Clustering [10]: This scheme is mostly employed in a network of sleep sensors, where most of the sensors stay in the sleep mode. In prediction based clustering when a target moves from the region of one cluster head to the other, the current cluster head has to make an estimation or prediction about where the target is moving and correspondingly wakeup the cluster head.

Non Prediction Based Clustering [10]: This scheme is used in a network of non sleep sensors. Here the energy saving is not an issue instead the proper selection and the life time of cluster head is an issue. So based on the application environment a cluster head selection algorithm is run on each individual node, and the nodes collaboratively select the cluster head.

2.1.2.2 DYNAMIC CLUSTERING:

Dynamic cluster architecture has several desirable features. When a sensor with sufficient battery and computational power detects signals of interest, it volunteers to act as a CH. Compared with the static clustering approaches, dynamic clustering networked sensors do not statically belong to a cluster and may support different clusters at different times. Moreover, as only one cluster is active in the vicinity of a target with high probability, redundant data is suppressed and potential interference and contention at the MAC level is mitigated. Dynamic Clustering consists of IDSQ [11], DELTA [12], and RARE [13].

2.1.2.2.1 Information-Driven Sensor Querying (IDSQ):

Information-Driven Sensor Querying [11] can be considered as a generalization of directed diffusion routing. IDSQ allows sensors to become activated when there are “interesting” events to report and only those parts of the network with the most useful information balanced by the communication cost need to be active.

2.1.2.2.2 DELTA:

DELTA [12] is one of the distributive algorithm. It tracks the object at constant speed by dynamically making a cluster and selecting Cluster Head based on light measurement. The advantage of this method is that the communication range of the sensor node is significantly higher than their sensing range. But the main drawback of this method is that it can only deal with constant speed, whereas, varying speed is not considered.

2.1.2.2.3 RARE:

The target tracking protocol uses two algorithms, RARE-Area [13] (Reduced Area Reporting) and RARE-Node (Reduction of Active Node Redundancy) [13]. The RARE-Area algorithm reduces the number of sensors used for tracking. The RARE-Node algorithm reduces the amount of redundant data in the network.

RARE Area: The RARE area [13] algorithm consists of two folds. First, it limits the number of sensors participating in tracking, and secondly it controls the amount of data to be transmitted to the cluster head. It achieves these two goals using a weighting scheme. The algorithm allocates a weight to the sensor data and only sensors whose weight value is above the set threshold are allowed to participate in tracking. As the distance between the target and the sensor increases, the strength of the signal decreases, and if the target distance increases, the percentage of noise in the received signal increases. The quality can be determined as the percentage of noise within the received target signal.

The RARE-Node algorithm: The RARE-Node algorithm [13] determines whether the data generated by a node is redundant or not. At the time of RARE-node algorithm run on a node it first checks the neighboring sensors within its sensing range. If no sensor is found, then the data is not redundant and it is forwarded to the cluster head. If neighboring sensors exist, the RARE-Node algorithm selects the nearest one to the target and checks whether that neighbor has enough energy to send data to the cluster head or not. If the sensor has enough energy, then that sensor is responsible for sending the data to the cluster head by considering the spatial relationship between them. If the nearest neighbor does not have enough energy to transmit data to the cluster head, then the RARE-Node algorithm checks the energy of the next nearest neighbor to the target until it finds one with enough energy to transmit data to the cluster head.

2.1.2.3 SPACE TIME CLUSTERING:

Dynamic Space-Time Clustering (DSTC):

The Dynamic Space-Time Clustering algorithm [14] is based upon the Closest Point of Approach (CPA) of a target to the nodes in the sensor network. A CPA is a local maximum in time of an acoustic signal arriving at a sensor node. The maximum intensity is broadcast to other nodes nearby, and they broadcast their CPAs as they occur. After a set time, if a node has received at least 4 other CPAs, and is the receiver of the highest intensity CPA in its local area, then it declares itself to be a local cluster head, and estimates the source location as a weighted mean of CPA positions, with the square root of the CPA intensity as the weight. The radius of

the area over which a CPA broadcast is considered to be equal to the range at which the CPA can reliably be detected, and the time window should be set to approximately the same time that a target takes to traverse the region. This constraint is imposed upon the sensor network by the situation, and often cannot be controlled during network deployment. The DSTC algorithm can fail if the density is too low, or if the cluster sizes are too large. And for target tracking not enough target position estimations are formed. In order to guarantee enough detection, the area radius may have to be reduced from maximum CPA detection range, which means that more nodes are required to guarantee 5 nodes per cluster area.

2.1.3 HYBRID METHOD:

Hybrid methods are the tracking algorithms that fulfill the requirements of more than one type of target tracking. Hybrid Clustering consists of some methods like DPT [15], DCAT [16], and Hierarchical prediction strategy (HPS).

2.1.3.1 Distributed Predictive Tracking (DPT):

Distributed Predictive Tracking adopts an approach [15] for scalability and prediction based tracking mechanism to provide distributed and energy efficient solution. It is robust against the prediction failures, temporary loss of the target and recovers from such scenarios quickly and with very little additional energy use.

2.1.3.2 Dynamic Clustering for Acoustic Tracking (DCAT):

Dynamic Clustering for Acoustic Tracking [16] is mainly used for single target tracking. Using Voronoi Diagrams, clusters are formed and only one Cluster Head becomes active when the signal strength detected by Cluster Head exceeds a pre-determined threshold. The Cluster Head then asks the sensors in its vicinity to join cluster by sending a broadcast packet. The sensor based on the probabilistic distance estimates between itself and target, decides whether it should reply to Cluster Head or not. Cluster Head executes a localization method to estimate location of target based on sensor replies and sends result to the sink.

2.1.3.3 Hierarchical prediction strategy (HPS):

Hierarchical prediction strategy is formed using Voronoi division and the target next location is predicted via Least Square Method.

2.1.4 ACTIVATION BASED:

2.1.4.1 Naive activation based tracking:

Naive activation (or direct communication) based tracking scheme [17] is the simplest approach, for which all nodes are in tracking mode all the time. Each node sends the local measurement to the sink node or base station. Then the base station estimates and predicts the

target state according to the received local measurements. Since it offers the best tracking results, it is a useful baseline for comparison. However, this strategy offers the worst energy efficiency and it inflicts heavy communication and computation burden on the base station of sink node. This makes the naïve approach not robust against base station failure especially for the case of link failure and channel congestion.

2.1.4.2 Randomized activation:

Randomized scheduling [18] of sensor nodes provides prolonged coverage in the sensor network. Two important motivations for considering a randomized scheduling algorithm are: (i) The disjoint set problem is NP-complete and hence it is not possible to solve it optimally using any centralized deterministic algorithm in polynomial time, and (ii) Randomized scheduling algorithms could be easily implemented in a distributed network environment, since they require the sensors to maintain minimal or no global state information, and also avoid unnecessary communications overhead for decision making amongst the sensor nodes.

2.1.4.3 Selective activation based on prediction:

In this strategy, only a small subset of all the nodes is in tracking mode at any given point of time. They also predict the “next” position of the target and hand over tracking to nodes best placed to track the target in the “next” position. The rest of the nodes are in communication mode and can switch to tracking mode on being alerted by signals from tracking nodes.

2.1.4.4 Duty-cycled Activation (DA):

In duty-cycled activation [19], the entire sensor network periodically turns off and on with a regular duty cycle. One key feature of duty-cycled activation is that it can actually be used in conjunction with any other activation strategy for target tracking

2.2 Tracking methods for peer-to-peer networks:

For the tree or cluster-based methods, sensing task is performed by several nodes at a time and inflicts heavy computation burden on the root node or the CH. This makes the tree or cluster-based WSN tracking systems lack of robustness in case of root node or the CH failures. On the contrary, another architecture for target tracking is the peer-to-peer WSN. As it can guarantee that sensors obtain the desired estimates and rely only on single hop communications between neighboring nodes, the limitations mentioned above are not encountered in peer-to-peer WSN based target tracking systems. Embedded filter based consensus belongs to Peer-to-Peer Networks.

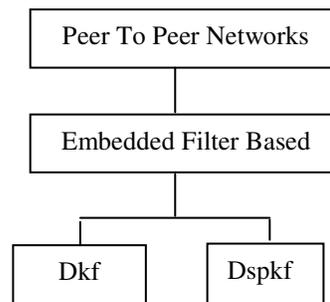


Fig: Classification of Peer To Peer Networks

2.2.1 Embedded filter based consensus:

Distributed estimation using peer-to-peer Wireless Sensor Networks is based on successive refinements of local estimates maintained at individual sensors. Each iteration of the algorithm comprises a communication step where the sensors interchange information with their neighbors, and an update step where each sensor uses this information to refine its local estimate. The DKF algorithm [20] consists of a network of micro-Kalman filters each embedded with a high-gain high-pass consensus filter (or consensus protocol). The role of consensus filters is to estimate of global information contribution using only local and neighboring information. The main idea is to use dynamic consensus strategy to the information form sigma-point Kalman filter (ISPKF) that derived from weighted statistical linearization perspective. Each node estimates the global average information contribution by using local and neighbours' information rather than by the information from all nodes in the network. Therefore, the proposed DSPKF algorithm [21] is completely distributed and applicable to large-scale sensor network.

2.2.1.1 Distributed Kalman filtering (DKF) algorithm:

The DKF algorithm [20] consists of a network of micro-Kalman filters each embedded with a high-gain high-pass consensus. The role of consensus filters is the estimation of global information contribution using only local and neighboring information.

2.2.1.2 Distributed Scalable Sigma-Point Kalman Filter (DSPKF):

A distributed scalable Sigma-Point Kalman filter (DSPKF) [21] is proposed for distributed target tracking in a sensor network based on the dynamic consensus strategy. The main idea is to use dynamic consensus strategy to the information form sigma-point Kalman filter (ISPKF) that derived from weighted statistical linearization perspective. Each node estimates the global average information contribution by using local and neighbor's information rather than by the information from all nodes in the network. Therefore, the proposed DSPKF algorithm is completely distributed and applicable to large-scale sensor network.

2.3 Other Methods:

2.3.1 Prediction-Based Methods:

Prediction-based methods are used to predict the Future movement locations of the moving objects. Some of the methods in this are Distributed Predictive tracking algorithm (DPT) [22], Dual prediction reporting (DPR) [23], Prediction-based Energy Saving scheme (PES)[24], Prediction-based energy-efficient target tracking protocol (PET)[25], Prediction-based tracking technique using sequential pattern (PTSP)[26], Exponential distributed predictive tracking (EDPT)[27].

2.3.1.1 Distributed Predictive tracking algorithm (DPT):

Distributed Predictive tracking algorithm [22] is one of the distributed and scalable prediction based algorithm to accurately track mobile targets using sensor networks. Cluster based architecture is used in this algorithm for more scalability and robustness. The algorithm is used after sensors are deployed and clusters are formed. In this algorithm there is no need of central control point. So it eliminates the single point failures. The tracking task is carried out distributively by sequentially involving the sensors located along the track of the moving target. In DPT several sensors exists in several areas. Some sensors lies in the boarder, some sensors located within a given distance of the border, and non border nodes exists in terms of their operation. Border sensors are required to keep sensing all times in order to detect all targets that enter the sensing region, the non-border sensor's sensing channel hibernates unless it is specifically asked to sense by its cluster head. Since the target is assumed to move from outside into the sensing area, it will be detected by the border sensors when tracepassing the border. When a target is detected a sequence of tasks carried out like sense-predict-communicate. This is the basic idea behind the DPT.

2.3.1.2 Dual prediction reporting (DPR):

Dual prediction reporting [23] minimizes the number of long distance transmissions between sensor nodes and the base station with a reasonable overhead so energy consumption is reduced. The basic idea of DPR is that a sensor node which has an object in its territory first predicts the movement of the object for the next reporting period. Meanwhile, the base station makes the same prediction based on the same objects' movement history. The predictions at the base station, stored locally, are never transmitted to the sensor nodes and thus incur no communication cost. If the observed object's movement matches the sensor node's prediction, no transmission is needed, since the base station has the same predictions as the sensor node. The history stored at the base station and sensor nodes are consistent, and their predictions for the objects' movement are identical all the time. The transmissions from the sensor nodes to the base station are avoided as long as the predictions are correct. On the other hand, to allow the sensor nodes which never saw the object to make predictions, the objects' movement history has to be passed between sensor nodes as the object moves. One hop transmission is done between the neighboring sensors to exchange the movement history. The energy consumption is minor compared with the multi-hop transmissions between the base station and sensor nodes.

2.3.1.3 Prediction-based Energy Saving scheme (PES):

Prediction-based Energy Saving scheme [24] minimizes the sampling frequency and the number of nodes involved in object tracking. It balances the overhead caused by missing of objects. PES consists of three parts: prediction model, wake up mechanism, recovery mechanism. Prediction model anticipates the future movement of an object so some of the sensor nodes expected to discover the object will be activated. Wake up mechanism based on some heuristics taking energy and performance into consideration, sets up which nodes has to be activated and when they should be activated. A recovery mechanism initiated only when the network loses the track of an object. The basic idea of PES is that a sensor node not performing the duty of object tracking should stay in sleep mode as long as possible. A sensor node which

has a moving object in its territory, called current node, should also try to turn to sleep mode as much as it can. Based on the prediction model used, the current node will predict the possible locations of the moving object and determine a group of sensor nodes, called target nodes, to help tracking the moving object. The current node inactivates itself after sending the wake up call to the target nodes.

2.3.1.4 Prediction-based energy-efficient target tracking protocol (PET):

A prediction-based energy-efficient target tracking protocol (PET) [25] was proposed for deriving the target moving path and utilizing the target moving patterns for energy saving prediction-based target tracking sensor networks. The PET tracking protocol is derived from the cooperation of sensors. The prominent characteristics of the cooperative tracking protocol were that it achieved better resolution than that with a lot of sensors being used in tracking and provided an earlier estimation of the target's position, direction and velocity. It uses a linear predictor to predict the target's next location. However, not all sensors or partial sensors may have useful information; hence, a well-versed selection of sensors with the best data for cooperation can conserve power. Whenever a target is detected, an election process is conducted by the current beacon and nearby sensors to choose a sensor, on which a tracker is initiated to observe the movement behavior of the target. The PET first determines the location of the target, and then uses two-dimensional Gaussian distribution to accurately predict the future location of the target. Once the future location of the target is known, the new beacon is computed. Based on coordination among the sensor nodes, a node is selected, which uses direct flooding to communicate the information to the new active face. The PET uses a primary sensor that is in the active state, and the secondary sensors are in the awakened state. The protocol significantly simplified the beacon's computation and minimized the volume of messages exchanged between beacons as well as sensors and trackers.

2.3.1.5 Prediction-based tracking technique using sequential pattern (PTSP):

Prediction-based tracking technique using sequential pattern [26] is one of the object tracking technique that predict the future movements of the objects that track with the minimum number of sensor nodes. PTSP is based on two stages: Sequential pattern generation, Object tracking and monitoring. In the sequential pattern generation stage, the prediction model is built based on a huge log of data collected from the sensor network and aggregated at the sink in a database, producing the inherited behavioral patterns of object movement in the monitored area. Depending upon this data the sink will be able to generate the sequential patterns that will be deployed to the sensor nodes in the network. So the sensor nodes that can predict the future movements of moving objects in their detection area. In the second stage, the actual tracking of moving objects starts. This stage has two parts: Activation Mechanism and Missing Object Recovery Mechanism. The use of the Activation Mechanism is to predict which node should be activated continually to keep track of the moving object. The missing object recovery mechanism is used to find the missing object in case of the activated node is not able to locate an object in its detection area.

2.3.1.6 Exponential distributed predictive tracking (EDPT):

The Exponential distributed predictive tracking [27] protocol is suitable to track target with lower computation complexity. It can also estimate the target position without any prior noise state matrix or target state matrix. This protocol can serve as a tracking protocol with relatively higher precision and lower computation complexity.

2.3.2 MOBICAST Based Method:

A new multicast communication paradigm called a “spatiotemporal multicast” or “Mobicast” was investigated to support spatiotemporal coordination in applications over wireless sensor networks. [28] The distinctive feature of multicast is the delivery of information to all nodes that happen to be in a geographic zone at a particular point in time. The set of multicast message recipients is specified by a forwarding zone which continuously moves and evolves over time. When continuously monitoring an entity, forwarding zones at different time intervals greatly differ. This provides a mechanism for application developers to express their needs for spatial and temporal information dissemination directly to the multicast communication layer. It also offers a new just-in-time multicast delivery paradigm.

Some of the protocols involve in the MOBICAST Based Method are Face-Aware Routing protocol (FAR) [29], Variant-egg-based mobicast (VE-mobicast) routing protocol [30], Hierarchical-variant-egg (HVE)-based mobicast routing protocol [31].

2.3.2.1 Face-Aware Routing protocol (FAR): The aim of the algorithm is that every node that has at least one spatial neighbor that is a Delivery-Zone node, that will forward or locally broadcast the Mobicast packet. So all delivery zone nodes will receive the corresponding packet. This simple rule leads an “as-soon-as-possible” style Mobicast protocol that exhibits a high average slacktime which is not desirable. The face-aware routing algorithm consists of two methods: greedy forwarding and turned forwarding. FAR [29] provides both reliability and scalability at the same time. Its scalability comes from the fact that it does not rely on any global topological information and each node makes local forwarding decisions based on its spatial neighborhood configuration.

2.3.2.2 Variant-egg-based mobicast (VE-mobicast) routing protocol: VE-mobicast protocol [30] builds a new shape of a forwarding zone, called the variant-egg that can adaptively and efficiently determine the location and shape of the forwarding zone to maintain the same number of wake-up sensor nodes. It is a fully distributed algorithm which reduces the communication overhead of determining the forwarding zone and the mobicast message forwarding overhead. And it can improve the predicted accuracy of the forwarding zone by considering the factors of moving speed and direction.

2.3.2.3 Hierarchical-variant-egg (HVE)-base mobicast routing protocol: VEmobicast protocol can adaptively determine the location and shape of the message forwarding zone. This method is not efficient and wastes energy unnecessarily. HVE [31] is proposed to overcome this problem. But each node which participates in forwarding the mobicats message has complex computing.

REFERENCES

- [1] H. T. Kung and D. Vlah Proceedings of 2003 IEEE Wireless Communications and Networking Conference "Efficient Location Tracking Using Sensor Networks".
- [2]" DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks" Wensheng Zhang, TRANSACTIONS ON WIRELESS COMMUNICATIONS, VOL. 3, NO. 5, SEPTEMBER 2004.
- [3] Wensheng Zhang and Guohong Cao Published in the IEEE Transactions on Wireless Communications "DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks.
- [4] Lin, C.Y., Peng, W.C., Tseng, Y.C., "Efficient in-network moving object tracking in wireless sensor networks", Department of Computer Science and Information Engineering-National Chiao Tung University, 2004.
- [5] Chih-Yu Lin and Yu-Chee Tseng "Structures for In-Network Moving Object Tracking in Wireless Sensor Networks" Proceedings of the IEEE.
- [6] H.-W. Tsai, C.-P. Chu and T.-S. Chen, "Mobile Object Tracking In Wireless Sensor Networks," Computer Communications 30, 2007,
- [7] Tran, S.P.M., Yang, T.A. "OCO: Optimized Communication & Organization for Target Tracking in Wireless Sensor Networks", Proceedings of the IEEE International Conference on Sensor Networks.
- [8] Chan, H, Luk, M, Perrig, "Using clustering information for sensor network localization", in: Proceedings of the International Conference on Distributed Computing in Sensor Systems (DCOSS'05), 2005.
- [9] W.-R. Chang; H.-T. Lin and Zong-Zhi Cheng, "CODA: A Continuous Object Detection and Tracking Algorithm for wireless Ad Hoc Sensor Networks," Consumer Communications and Networking Conference, 2008. 5th IEEE, vol., no., pp.168-174, 10-12 Jan. 2008.
- [10] Mohsin Fayyaz , Wireless Sensor Network, 2011, "Classification of Object Tracking Techniques in Wireless Sensor Networks".
- [11] Feng Zhao and Jaewon Shin et al IEEE Signal Processing Magazine, March 2002, "Information-Driven Dynamic Sensor Collaboration for Tracking Applications".
- [12] M. Walchli, P. Skoczylas, M. Meer and T. Braun, "Distributed Event Localization and Tracking with Wireless Sensors," in Proceedings of the 5th international Conference on Wired/Wireless internet Communications, May 23 - 25, 2007.
- [13] E. Olule, G. Wang, M. Guo and M. Dong, "RARE: An Energy Efficient Target Tracking Protocol for Wireless Sensor Networks," 2007 International Conference on Parallel Processing Workshops (ICPPW 2007), 2007.
- [14] SHASHI PHOHA Et al International Journal of Distributed Sensor Networks, 1: 81-99, 2005 "Space-time Coordinated Distributed Sensing Algorithms for Resource Efficient Narrowband Target Localization and Tracking".
- [15] H. Yang and B. Sikdor, "a protocol for tracking mobile targets using sensor network, sensor network protocols and applications", in First IEEE International Workshop on Sensor Network Protocols and Applications, Anchorage, Alaska, pp. 71-81, 2003.
- [16] Wei-Peng Chen, Jennifer 2003 "Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks".

- [17] Proceedings of the second international conference ACM 2003 "Energy quality trade offs for target tracking in wireless sensor networks".
- [18] Nabhendra Bisnik, Neeraj Jaggi 2006 "Randomized scheduling algorithm for wireless sensor networks".
- [19] Zahedi, Etal 2010 "Quality tradeoffs in object tracking with Duty-cycled sensor networks".
- [20] R.Olfati-Saber 2005 Ipsn'05 Proceedings of the 4th international symposium on information "Distributed Kalman Filtering for Sensor Networks".
- [21] Yan Zhou, Jianxun Li, Proceedings of the 2009 IEEE International conference on system "Distributed Sigma Point Kalman Filtering For Sensor Networks: Dynamic Consensus Approach".
- [22] Yang, H., Sikdar, B., "A Protocol for Tracking Mobile Targets using Sensor Networks", Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, pp.71-81, 2003.
- [23] Xu, Y., Winter, J., Lee, W.C., "Dual Prediction-based Reporting for Object Tracking Sensor Networks", First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, pp. 154-163, 2004.
- [24] Yingqi Xu, Winter, J., Wang-Chien Lee., "Prediction-based Strategies for Energy Saving in Object Tracking Sensor Networks", IEEE International Conference on Mobile Data Management, pp. 346-357, 2004.
- [25] Bhuiyan, M Z A, Wang, G J, Peng Yong, Zhang Li., "Prediction-based energy-efficient target tracking protocol in wireless sensor networks", Journal of Central South University of Technology, Vol. 17, pp. 340-348, 2010.
- [26] Samarah, S., Al-Hajri, M., Boukerche, A., "A Predictive Energy-Efficient Technique to Support Object-Tracking Sensor Networks", IEEE Transactions On Vehicular Technology, Vol. 60, NO. 2, pp. 656-663, 2011.
- [27] Liang Xue, Zhixin Liu, Xinping Guan, "Prediction-based protocol for mobile target tracking in wireless sensor networks", Journal of Systems Engineering and Electronics Vol. 22, No. 2, pp. 347-352, 2011.
- [28] Qingfeng Huang, Chenyang Lu, Gruia-Catalin Roman. "Mobicast: Just-in-time multicast for sensor networks under spatiotemporal constraints", in Proc. of the 2nd International Workshop on Information Processing in Sensor Networks, Palo Alto, CA, USA, pp. 442-457, 2003.
- [29] Qingfeng Huang, Chenyang Lu, Roman, G.C., "Reliable Mobicast via Face-Aware Routing", Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 2004.
- [30] Chen, Y. S, Ann, S. Y., "VE-Mobicast: A Variant-Egg-Based Mobicast Routing Protocol for Sensornets", In Proceeding of The 40th IEEE International Conference on Communications, January 2005.
- [31] Yuh-Shyan Chen, Yi-Jiun Liao, Yun-Wei Lin, Ge-Ming Chiu., "HVE-Mobicast: A Hierarchical-Variant-Egg-Based Mobicast Routing Protocol for Wireless Sensornets", Telecommunication Systems, Springer, pp. 121-140, 2009.