Tracking and Predicting Moving Targets in Hierarchical Sensor Networks

Zhibo Wang, Hongbin Li, Xingfa Shen, Xice Sun, Zhi Wang*

***Abstract*—Target tracking is an important application of newly developed Wireless Sensor Networks (WSN). Much work has been done on this topic using a plane network architecture. We propose a scheme, namely Hierarchical Prediction Strategy (HPS), for target prediction in hierarchical sensor networks. The network is divided into clusters, which are composed of one cluster-head and many normal nodes, by Voronoi division. For an existing target, cluster-heads only selectively activate nearby sensor nodes to perform tracking. Moreover, Recursive Least Square technique is used to predict the target trajectory and help activate next-round sensor nodes. Extended simulations show the properties of the proposed network architecture and the efficiency of the prediction scheme.**

***Index Terms*—Wireless Sensor Network, Target Tracking, Prediction.**

I. Introduction

With the fast development of distributed Wireless Sensor Networks (WSN), the scenarios of field surveillance and target tracking by small and populated local sensor nodes become possible. The sensor nodes, integrated with sensing, data processing and wireless communication, are utilized to play significant roles in environment monitoring, traffic control, precise agriculture and battlefield surveillance [1]–[4].

Target tracking is an important application of WSN. However, large-scale wireless networks frequently suffer from packet losses, communication delays and energy limitations etc. How to coordinate a large-scale network to efficiently track a moving target while conserve network resources, namely energy and bandwidth, is a great challenge. A main feature, introduced by tracking in large-scale networks, is only those sensors near target can detect the target and perform sensing. Based on this fact, making all nodes working is not the optimal choice. Our efforts in this research mainly focus on the management of large-scale networks and the effectiveness of target tracking. By managing sensor nodes behaviors, only a small amount of nodes near the target are activated, thus saving energy without losing target. Moreover, activated cluster can perform target prediction and nodes pre-activation, which enable high-quality tracking.

Most recent work focused on totally distributed homogeneous sensor models and dynamic cluster coordination. As a matter of fact, energy resources, computation capacity and bandwidth are restricted in WSN. Such networks, managed by dynamic cluster coordination, are prone to suffer from endless message exchanges among sensor nodes, which drain off network energy quickly. In hierarchical network, however, powerful nodes, which can optimize local resources and efficiently coordinate local network, provide an alternative.

In this paper, we propose a scheme named Hierarchical Prediction Strategy (HPS) for target tracking in sensor networks. The two-tier hierarchy divides sensor nodes as Cluster-Heads (CHs) and Normal Nodes (NNs). We assume that CHs can communicate with each other and send commands/inquiries to NNs, while NNs can transmit observations to CHs but do not talk to other NNs. By constraining NNs' communication and reducing network congestion, wireless bandwidth is saved so that fusion packets formed on CHs can be delivered with better guarantee. Further, we explore the property of the proposed tracking and predicting algorithm in the hierarchical network. Simulations show that our method efficiently coordinates large-scale hierarchical network and achieves a nice tradeoff between energy consumption and tracking quality.

The outline of this paper is as follows. Section 2 briefly presents a survey of related work. In Section 3 we propose a scheme for target tracking in sensor network. Then Section 4 presents and analyzes the simulation results concerning our proposed scheme. Finally, Section 5 states our conclusions and future work.

II. Related Work

The first cluster-based routing approach for dynamic network is introduced in [5]. Composed of a group of neighboring nodes, a cluster manages the whole detection/tracking process, with the cluster head collecting joint observations from the pertaining sensor nodes and eventually producing a final fusion results.

Moving target tracking based on WSN has been an attractive topic in recent years. The information driven sensor querying (IDSQ) mechanism for sensor collaboration in ad hoc sensor networks is proposed in [6]. In IDSQ, at any time, there is a dynamic leader deciding which sensors should be selectively activated in order to obtain the best information about the target. In [7], the authors use cellular automata models to study tracking performance and network data in a unified framework. [8] describe self-organized, distributed target tracking techniques with prediction based on Pheromones, Bayesian, and Extended Kalman Filter. In [9], the authors assume a scenario of totally distributed WSN and the performance of four strategies, namely Naive Activation, Random Activation,

Zhibo Wang, Hongbin Li, Xice Sun and Zhi Wang are with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China.

Xingfa Shen is with the Institute of Computer Application Technology, Hangzhou Dianzi University, Hangzhou, China.

*Corresponding Author. (e-mail: wangzhi@iipc.zju.edu.cn)

Selective Activation and Duty-cycled Activation, are compared in terms of energy consumption and tracking error. They conclude that there is a tradeoff between energy consumption and tracking quality. And Selective Activation shows the best tradeoff among all strategies. In Distributed Predictive Tracking (DPT), proposed in [10], clusters selectively activate the sensor nodes. Simulation shows that DPT is able to track with high quality even the target is moving at high speed.

Inspired by recent work on cluster-based network, we consider the problem of target tracking in hierarchical sensor network. Particularly, the problem of Pursuit Evasion Games (PEGS) [11]–[15], which we find very interesting in both practice and theory, is a suitable future application of hierarchical sensor networks.

By dividing the network into clusters, we shift the cost of endless message exchanges among sensor nodes to cluster heads with high energy resources and computation capacity. Each cluster works in a centralized way — CH manages NNs local behaviors and NNs don't exchange ideas. On the upper layer, however, CHs run in a totally distributed way.

III. Hierarchical Prediction Strategy

A. Hierarchical Network

The network architecture considered in this paper is a two-tier hierarchical network. It differs from previous homogeneous network in the fact that all nodes deployed in the field of interest (FoI) are categorized into 2 levels: CHs and NNs. CHs have stronger computational power and infinite energy resources, while in contrast, NNs serve as small identical sensing components and their computation capacity and energy are both under stringent constraints. The respective roles of CHs and NNs are defined as follows:

1) *NNs collect observations from the environment and transmit to the corresponding CHs directly but do not communicate with each other.*
2) *CHs send orders to NNs directly and different CHs can talk to each other.*

The clusters work under two modes: *Active* and *Idle*. A cluster switches from Idle to Active whenever the occurrence of a target is detected or the future trajectory predicted by neighbor CH intersecting with its own coverage area. For NNs in a cluster, their modes are further divided into *Probabilistic Active* and *Sleep*. When a cluster is *Active*, it works in a centralized fashion. The cluster-head, which manages the global behavior within the cluster, assign *Probabilistic Active* or *Sleep* to NNs, according to the estimated target location. NNs working at *Probabilistic Active* sense data in a probability of p and sleep in a probability of $1-p$ in each sensing circle. The value p is assigned according to the importance of the target, aiming at decreasing unnecessary energy consumption.

Let N_s be the number of NNs and N_H the number of CHs. $\mathbf{R}$ denotes the field FoI. $L_i \in \mathbf{R}$ be the location of node i and $L_{H_i} \in \mathbf{R}$ denotes the locations of CH i. The locations of NNs and CHs are denoted as $L = \{L_i : 1 \leq i \leq N_s\}$ and $L_H = \{L_{H_i} : 1 \leq i \leq N_H\}$, respectively.

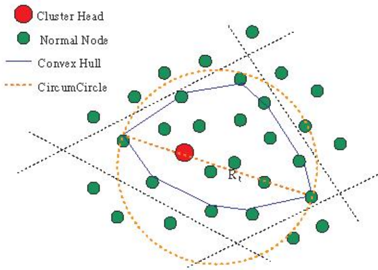


Fig. 1. A Sample of Deciding R_t in A Cluster

We assume the deployment processes of CHs and NNs both conform to uniform distribution throughout the FoI. The cluster assignment is based on Voronoi division [16] resulting from the locations of CHs. Once the deployment is accomplished, the location information and identity of CHs and NNs are assumed known to CHs. Then CHs negotiate with each other to determine the cluster division according to Voronoi division rule: NN i belongs to CH k, $k = min_k||L_{H_k} - L_i||$. Finally CHs recruit calculated NNs and form specific clusters accordingly. In each cluster, the CH assign an identical transmission range R_t that equals to the diameter of the circumcircle of the local convex hull, as shown in Figure 1. The division is completed as a result.

B. Sensor Node Models

We assume the NNs have binary sensing model. Let R_s be the identical sensing range of NNs. When a target T appear at location L_T, the sensing result R_i of i is defined as

$$R_i = \begin{cases} 1, & ||L_i - L_T|| \leq R_s, \\ 0, & ||L_i - L_T|| > R_s. \end{cases} \tag{1}$$

The energy model of the sensor nodes is simplified as follows. One normal node send one message to a distance of 1 meter consumes energy E_k. For sending a message over a distance of d, the energy consumption is $E_k \times d^2$. NNs working in *Probabilistic Active* consume $E_k \times R_t^2$ in each sensing circle. Here we ignore the energy consumption of sampling because it is highly application dependent. NNs working in *Sleep* consume $E_l \approx 0$, because from practical experience, E_l is much smaller than E_k.

C. Localization Algorithm

Localization is a basic component in target tracking. Here we use Centroid Localization algorithm for the binary sensing model. In time period t, we assume k NNs are activated to track the target and $l (l \leq k)$ NNs detect the target and their location is $L_{i_j}(x_{i_j}, y_{i_j}), j = 1, \ldots, k$. The estimated target location is

$$L_e(t) = (\frac{\sum_{j=1}^{l} x_{i_j}}{l}, \frac{\sum_{j=1}^{l} y_{i_j}}{l}) \tag{2}$$

The $(k-l)$ NNs, which are activated but fail to detect the target, do not send the result to CHs. We call this *implicit*

% Tracking and Predicting Process
At Each Sensing Circle T_{sc}
1 *Subleaders* calculate local result and send to *leader*
2 *Leader* $CH_l(t)$ calculates $L_e(t)$
3 Predict $L_p(t+1)$ using RLS
4 Assign new leader $CH_l(t+1)$ according to $L_p(t+1)$
5 Deactivate $NN(t-1)$
6 Activate $NN(t+1)$, each with a probability of p
7 Assign *Subleaders* according to the activated NNs

TABLE I
THE TRACKING AND PREDICTING PROCEDURE

report because only NNs with positive detecting result are taken account into localization. By importing *implicit report*, unnecessary energy consumption due to wireless communication can be saved.

D. Prediction Algorithm

As targets are maneuvering, sometimes escaping, in the field. Predicting the movement is as important as localizing. We propose a prediction algorithm using Recursive Least Squares (RLS) technique [17].

Consider a target appears in the FoI and is being tracked by a cluster with *leader* $CH_l(t)$ in time t. Achieving the $L_e(t)$ from k activated $NN(t-1)$ and $NN(t)$, CH_l predict the next location $L_p(t+1)$ using RLS algorithm. The CH where $L_p(t+1)$ locates in will be assigned as the new $leader$. Then a set of nodes $NN(t+1)$ from origin $L_p(t+1)$ within radius R_a would be activated. If nodes in $NN(t+1)$ do not belong to the current cluster, then the corresponding CH(s) would be activated to be *subleader(s)*. In the next round $t+1$, $NN(t-1)$ would be deactivated while $NN(t)$ are still kept active, then $NN(t)$ and $NN(t+1)$ report their sensing results to their corresponding CH(s). *Subleaders* calculate the local results and send them to leader $CH_l(t+1)$ and the *Leader* $CH_l(t+1)$ achieve a final $L_e(t+1)$. The tracking and predicting procedure is summarized in Table I.

In case of less than 3 NNs report sensing result to CHs, which we consider the target is nearly or already lost, the *tracking recovery* is invoked. Consider an extreme example, as shown in Figure 2,only one activated NN, which located in the overlap region of activated circle and target circle, successfully detects the target. Taken into account the estimated target velocity V_t, the expected radius which is able to cover the moving target in the next sensing circle is $R_a+2R_s+V_t*T_sc$. If none of the NNs in the second activated range can detect the target, we deem the target disappeared and the tracking process is terminated.

E. Reconfiguration under CH failures

Once a CH failures for a certain period, nearby CHs reach a consensus that the corresponding CH no longer exist. A new round of negotiation is reinitiated to calculate a new cluster division. The NNs that lost their CH will be recruited by new CHs. The network reconfiguration is thus finished.

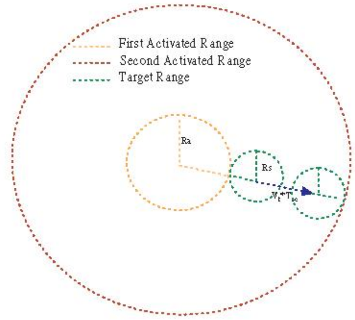


Fig. 2. Illustration of Tracking Recovery

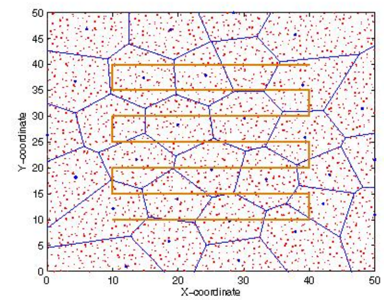


Fig. 3. Designated Target Trajectory

IV. SIMULATION RESULTS AND ANALYSIS

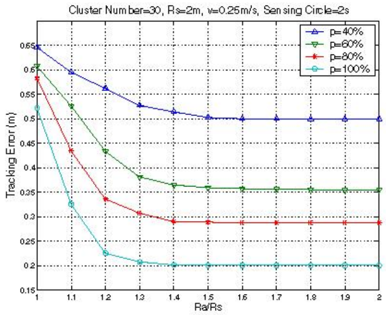
In the simulation, we deploy 2500 normal sensor nodes in a 50m $\times$ 50m area. Each sensor is assigned an identical sensing range $R_s = 2m$. We deploy one target on the simulated field and designate a moving trajectory as shown in Figure 3.

A. Active Radius R_a

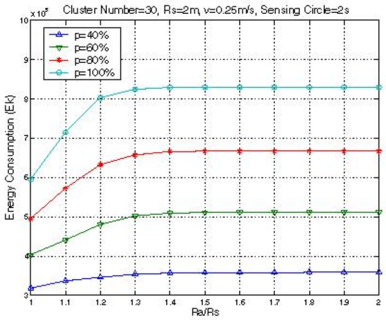
We first study the effects of R_a — the radius within which we activate the NNs near the predicted location.

As shown in Figure 4(a), tracking error drops continuously until R_a reaches $1.5Rs$. Since then, tracking error only drops slightly. Note that since $R_a = 1.5R_s$, curves with smaller p drop more steeply. This is mainly because enlarging R_a brings more NNs involved in tracking, thus more NNs which can sense the target are activated with probability p. When NNs are activated with $p = 100\%$, the tracking error does not change since $R_a = 1.5R_s$. So $R_a = 1.5R_s$ is the optimal active radius.

Figure 4(b) reveals similar facts: Before R_a reaches $1.5R_s$, energy consumption increases steadily. When $R_a > 1.5R_s$, energy consumption almost stays steady. This is because only a set of NNs are able to detect the target regardless of the range of activated NNs. The results of energy consumption, however, do not truly reflect the practical energy consumption, due to the fact that we only consider energy of wireless communication and ignore the energy consumption of sensing, which is greatly application dependent.

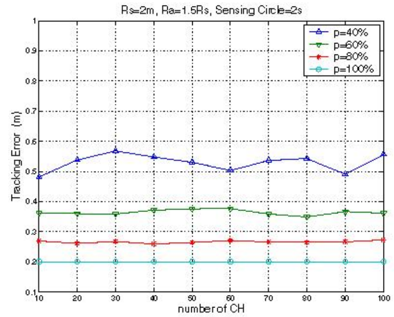


(a) The Effects of R_a on Tracking Error

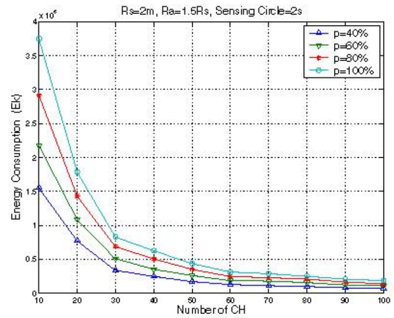


(b) The Effects of R_a on Energy Consumption

Fig. 4. The Effects of Active Radius R_a



(a) The Effects of Number of Clusters on Tracking Error



(b) The Effects of Number of Clusters on Energy Consumption

Fig. 5. The Effects of Number of Clusters

B. Number of Clusters

In this section, we study the effects of number of clusters, which is strongly related with energy consumption and tracking accuracy. The tracking error, as depicted in Figure 5(a), is not sensitive to the number of clusters. When the clusters are able to track the target because of sufficient large S_p and small sensing circle, varying the number of cluster does not affect the number of activated NNs. With smaller active probability p, as observed, the curves of tracking error regarding the number of cluster fluctuate. That is because p introduces an extend of uncertainty to the activated NNs.

On the other hand, as shown in Fig 5(b), increasing the number of clusters results in a dramatic fall in energy consumption. The main reason is we set a transmission range R_t according to the shape of the cluster. That is, NNs in smaller cluster communicate with CH with lower transmission power which results in lower energy consumption. Furthermore, the energy consumption under each setting p is proportional to the value p, which verifies our initial idea of saving energy by activating only partial nodes.

Although saving energy is attractive, our results do not indicate that we should deploy as many as possible clusters in hierarchical sensor networks. Because our results only report energy consumption in terms of wireless communication of NNs, and the hardware cost, energy consumption of the CHs are ignored.

C. Sensing Circle T_{sc}

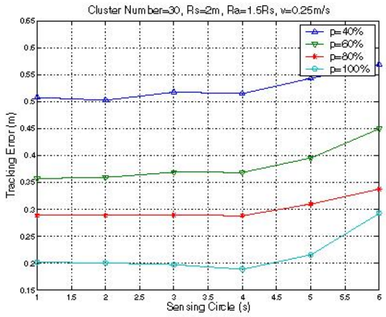
Finally we study the effects of sensing circle, which is related with target motion and tracking accuracy. As depicted in Figure 6(a), tracking error almost stays plane before T_{sc} reaches 4s and starts increasing since then. This reveals useful fact that when target is not running at high speed, we can choose larger sensing circle to achieve energy conservation without considerable loss of tracking accuracy.

Figure 6(b) shows that sensing circle is inversely proportional to energy consumption. But increasing sensing circle induces the risks of deteriorating tracking accuracy and losing target. The tradeoff between quality of tracking and energy consumption is well represented in our system.

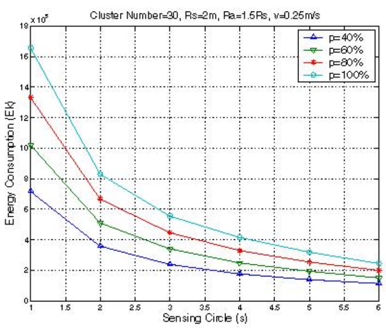
D. The Effectiveness of HPS

Here we present a system snapshot, in order to obtain a better understanding of tracking and predicting a target in hierarchical sensor network. The system setting are as follows: 30 clusters are divided in total. Sensing range $R_s = 2m$ and activated range $R_a = 1.5R_s$. The sensing circle is set 2 second. The target runs at the speed of 0.25m/s and follows a 'M' shaped trajectory.

Figure 7 shows a snapshot of the tracking and predicting process. Green line denotes the designated trajectory; linked red dots denote the estimated target locations and black dots represent the predicted locations. At the upper right corner, black circle denotes the activated range and red circle denotes the range of NNs which successfully detect the target. By comparing the estimated location and designated trajectory, we observed that the network could effectively track and predict the next location. Particularly, in locations where the



(a) The Effects of Sensing Circle on Tracking Error



(b) The Effects of Sensing Circle on Energy Consumption

Fig. 6. The Effects of Sensing Circle

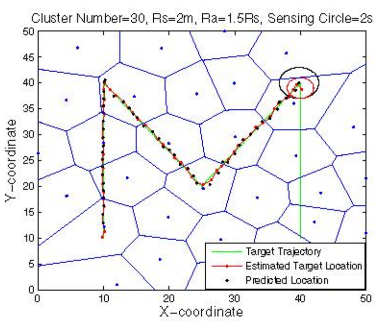


Fig. 7. A snapshot of the tracking and predicting process

target turned sharply, the predicted location encountered slight overrun but it soon went back to the right way.

V. Conclusions

In this paper, we have proposed a cluster-based target predicting strategy, aiming at efficiently coordinate large-scale hierarchical sensor networks. By organizing network behaviors, our scheme is capable of tracking moving targets, as well as recovering from network failures and target losses. Simulations show that our method achieve a nice tradeoff between tracking quality and energy consumption.

Future work includes further studying the relation between target motion and prediction strategy, and implementing a hierarchical sensor network based on Crossbow motes.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant No.60434030 and No.60773181, the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z218.

We are grateful to Jing Yu for her insightful suggestions on this research. Special thanks to Zeyu Liu and Zhaofu Chen for giving us supports on simulation work.

References

[1] D. Estrin, D. Culler, K. Pister, G. Sukhatme. Connecting the Physical World with Pervasive Networks. *IEEE Pervasive Computing.* Volume:1, No.1, 59–69. 2002.

[2] I. F. Akyildiz, S. Weilian, Y. Sankarasubramaniam, E. Cayirci. A Survey on Sensor Networks. *IEEE Communications Magazine.* Volume:40, No.8, 102–114. 2002.

[3] Sensor Networks and Applications. *Proceedings of the IEEE, Special Issue.* Volume:91, No.8, 1151–1256. 2003.

[4] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, D.Estrin. Habitat Monitoring with Sensor Networks. *Communications of the ACM.* Volume:47, No.6, 34–40. 2004.

[5] P. Krishna, N. H. Vaidya, M. Chatterjee, D. K. Pradhan. A cluster-Based Approach for Routing in Dynamic Network. *ACM SIGCOMM Computer Communication Review.* Volume:27, No.2, 49–64. 1997.

[6] F. Zhao, J. Shin, J. Reich. Information-Driven Dynamic Sensor Collaboration for Tracking Applications. *IEEE Signal Processing Magazine.* Volume:19, No.2, 61–72. 2002.

[7] P. R. Brooks, P. Pamanathan, A. M. Sayeed. Distributed Target Classification and Tracking in Sensor Networks. *Proceedings of the IEEE.* Volume:91, No.8, 1163–1171, 2003.

[8] P. R. Brooks, C. Griffin, D. S. Friedlander. Self-Organized Distributed Sensor Network Entity Tracking. *International Journal of High Performance Computer Applications, special issue on Sensor Networks.* Volume:16, No.3, 207–219, 2002.

[9] S. Pattem, S. Poduri, B. Krishnamachari. Energy-Quality Tradeoffs for Target Tracking in Wireless Sensor Networks. *Second Workshop on Information Processing in Sensor Networks.* 2007.

[10] H. Yang, B. Sikdar. A Protocol for Tracking Mobile Targets using Sensor Networks. *IEEE International Workshop on Sensor Networks Protocols and Applications.* 2003.

[11] S. Oh, L. Schenato, P. Chen, S. Sastry. Tracking and Coordination of Multiple Agents using Sensor Networks:: System Design, Algorithms and Experiments. *Proceedings of the IEEE.* Volume:95, No.1, 234–254, 2007.

[12] R. Vidal, O. Shakernia, H. J. Kim, D. H. Shim, S. Sastry: Probabilistic Pursuit-Evasion Games: Theory, Implementation, and Experimental Evaluation. *IEEE Transactions on Robotics and Automation.* Volume:18, No.5, 662–669, 2002.

[13] S. Thrun, W. Burgard, D. Fox. A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots. *Autonomous Robots.* Volume:31, No.5, 1–25, 1998.

[14] A. Antoniades, H. J. Kim, S. Sastry. Pursuit-Evasion Strategies for Teams of Multiple Agents with Incomplete Information. *Proceedings of the 42nd IEEE Conference on Decision and Control.* Volume:1, 756–761, 2003.

[15] J. Kim, R. Vidal, H. Shim, O. Shakernia, S. Sastry. A Hierarchical Approach to Probabilistic Pursuit-Evasion Games with Unmanned Ground and Aerial Vehicles. *Proceedings of the 40th IEEE Conference on Decision and Control.* Volume:1, 634–669, 2001.

[16] F. Aurenhammer. Voronoi Diagrams–A Survey of A Fundamental Geometric Data Structure. *ACM Computing Surveys* . Volume:23, No.5, 345–407, 1991.

[17] S. Haykin. Adaptive Filter Theory (fourth edition). *Beijing: Publishing House of Electronics Industry.* 2002.