

Tracking and Prediction of Mobility without Physical Distance Measurements in Sensor Networks

Yi Jiang, Dulanjalie C. Dhanapala and Anura P. Jayasumana.

Department of Electrical and Computer Engineering,
Colorado State University, Fort Collins, CO80523, USA
{yjiang, dulanjalie.dhanapala, Anura.Jayasumana}@Colostate.edu

Abstract—Existing methods for detection and tracking of mobile nodes and objects via sensor networks depend on physical localization of sensor nodes. Consequently, they suffer from disadvantages associated with localization, which is often costly, error prone, and in some cases even infeasible. Tracking mobile nodes in a topological coordinate domain is proposed as an alternative to that in geographic coordinate domain. Topological coordinates (TCs), the basis of recent developments in Topology Preserving Maps (TPMs), are derived from hop distances from each node to a small subset of nodes. A technique is presented to reduce the distortion closer to the edges of principal component based TPMs, thereby enhancing its accuracy, which in turn facilitates accurate mobility tracking. Topological Coordinate based Tracking and Prediction (TCTP) algorithm is proposed for tracking and predicting the position of mobile nodes in the TC domain. Simulation results presented show that tracking and detecting performance of TCTP is competitive with geographic coordinate based algorithms despite the fact that it relies only on hop distances to a subset of nodes.

Keywords- *Virtual Coordinate System; Wireless Sensor Network; Mobility; Object Tracking; Topology Preserving Maps*

I. INTRODUCTION

Wireless Sensor Networks (WSNs) comprise of low cost and energy limited sensor nodes with limited computational capabilities. Scalable and efficient algorithm design for WSN applications such as target tracking, boundary detection, data dissemination and habitat monitoring, without degrading the life span of the network has received much attention recently. Moving event or object tracking is one main category among WSN applications where current approaches require physical location information obtained using techniques such as Global Positioning System (GPS), Received Signal Strength Indication (RSSI), or time delay. Equipping nodes with GPS is costly, energy inefficient and infeasible for many applications. Localization based on RSSI is error-prone, interference sensitive, and difficult to implement even for small sensor networks [1]. Furthermore, physical localization is infeasible or not useful in certain environments. Eliminating the need of localization will result in simpler nodes and less sensitive algorithms, and facilitate large-scale deployment of WSNs.

This paper proposes a mobile target detection, prediction and tracking algorithm, called Topological Coordinate based Tracking and Prediction (TCTP) algorithm without need for physical location information. This is achieved by carrying out tracking in an alternative coordinate domain based on connectivity while preserving general features such as boundaries and shapes of the network.

Topology Preserving Maps (TPMs) [2] is a recent technique, which generates maps of networks from a Virtual Coordinate System (VCS) without the need of physical distance. A VCS is generated based on subset of nodes called anchors [2][3][4]. Each node in the network is characterized first by a Virtual Coordinate (VC) vector with the minimum hop distances to each of the anchors. VC generation is a many to one transformation from the physical domain, thus the directional information is not available in virtual domain. Using a Singular Value Decomposition (SVD) based dimensionality reduction scheme, Topological Coordinates (TCs) of nodes can be derived, thus regaining the directional information lost in VCS to derive TPMs. Though different from physical maps, TPMs preserve the internal and external boundaries of the network and provide a good substitute for physical maps for applications such as mapping [2], routing [3] and boundary detection [5]. TCTP algorithm proposed here is the first to demonstrate how TC domain can effectively be used for mobility and tracking, which up to now requires physical distance measurements or localization.

Several challenges have to be overcome for implementing object tracking and mobility related applications in the TC domain. As the TCs are derived from topological information only, the relationship between TCs and Geographic Coordinates (GCs) of network nodes is highly nonlinear. In other words, the TPM of a network is a distorted version of physical map, which however preserve the internal and external boundaries of the network and relative positions among the nodes. Comparison of TPMs with physical map indicates that the distortion often involves compression at the edges [2]. Thus in this paper we first propose a scaling method that reduces the compression especially for the nodes at the edges of network. To take into account the nonlinearity between GCs and TCs, we estimate the position and velocity in the TC domain, which to a significant extent compensates for the nonlinearity.

Consider the following two scenarios. First is a mobile node that moves in a sensor field, to be detected or tracked by the sensors. It could be a friendly node that cooperates with sensors or a passive target. Either way its geographic position is found using the GCs of the sensors. This lets the node navigate further (using GCs) or allows sensors to track its position using GCs. Secondly, as a mobile node moves in a sensor field, it may be necessary to predict its position at some time in future. The GCs are used to estimate the velocity and direction, and then using some mobility model, to predict the target's position at a future time. Such predictions may be used to alert nodes downstream, for example. Another situation

occurs when the mobile node cooperates and communicates with other sensor nodes. Suppose there is a Base Station (BS) that wishes to send a message or program updates to rendezvous with the mobile node at a future time. Rather than routing the message along the path that the mobile node followed, a prediction of its future position will allow the BS to send the messages directly to the rendezvous point. Existing algorithms rely solely on GCs to estimate current velocity and direction and to predict its future position via a mobility model. In [6], Distributive Predictive Tracking (DPT) is proposed for cluster-based WSNs. Cluster heads predict the mobile target's future position and wake up a triplet of sensors to wait for mobile target's arrival. Prediction-based Optimistic Object Tracking (POOT) scheme in [7] combines collecting and maintaining tracking information to minimize routing distance for predictive tracking. In [8], dual predictions take place at both BS and sensor nodes to improve the detection accuracy. Such predictions may, for example, be used to alert nodes downstream.

In contrast to existing approaches, proposed TCTP algorithm is a TC based approach for performing target tracking as well as prediction. To our knowledge, this is the first tracking algorithm that operates in such a virtual domain thus not requiring geographic information based on physical distance measurements. The contributions of this paper include a correction scheme to reduce edge distortions in TPMs to enhance its accuracy. The VCs of the mobile node are derived without resorting to additional flooding's by anchors. The detecting performance of TCTP algorithm in mobile target tracking and perdition is compared with the same approach based on geographic (physical) information. Simulation based results demonstrate that TCTP achieves similar performance compared with physical information based approaches. This paper also paves the way for use of TC domain for many other sensor network applications that usually rely on GCs.

II. TOPOLOGY PRESERVING MAPS AND VIRTUAL COORDINATES FOR MOBILE TARGET

A. Virtual Coordinate System and Topolgy Preserving Maps

A VCS is based on a set of anchors, which correspond to a subset of nodes, selected randomly or by an anchor selection strategy [2][9]. Each node in the network, including anchors, is characterized by a VC vector, consisting of shortest hop distances to each of the anchors [2][3][4]. Directional and geographic information are not available in VCS as the VCs propagate radially. A method to extract directional information from VCs in the form of TCs is presented in [2]. TPM of a network based on TCs is a somewhat distorted version of the physical map of the network. It however has been shown to preserve relative position information, and is a good substitute for physical information in applications for WSNs such as routing [3] and boundary detection [4]. Birds' eye view of an area is an example of a distorted physical map, which can still be used for such functions, even though it is not an exact physical map. In this paper, we consider dealing with tracking and mobility related applications directly in the TC domain instead of the physical domain, which is somewhat analogous to using a bird's eye view map.

Consider a network with M anchors and N sensor nodes. Thus each node is characterized by a VC vector of length M , the i^{th} element of which corresponds to the minimum number of hops from the node to the i^{th} anchor. Let P be the $N \times M$ matrix containing VCs of all sensor nodes in the network. Efficient and sensor network-friendly implementation of the SVD computation is addressed in [2]. Considering the extensive computation cost of SVD in deriving TCs from VC set of all sensor nodes which is matrix P , calculating SVD components from anchor set is economical choice for large-scale WSNs. Let A be a $M \times M$ matrix containing VCs of anchor sensor nodes. Generating TCs from anchor nodes set using SVD is presented below [2]:

$$A = U_A \cdot S_A \cdot V_A^T \quad (1)$$

$$P_{SVD} = P \times V_A \quad (2)$$

$$[X_T, Y_T] = [P_{SVD}^{(2)}, P_{SVD}^{(3)}] \quad (3)$$

P_{SVD} is a $N \times M$ matrix containing sensor nodes' principal components. The first column $P_{SVD}^{(1)}$ is the most significant component but contains 1-dimensional radial information which is not sufficient for 2-dimensional TPMs [2]. Meanwhile second column $P_{SVD}^{(2)}$ and third column $P_{SVD}^{(3)}$ contain the topological information that can be translated to angular information and these two columns can be seen as 2-dimensional Cartesian coordinate set for sensor nodes in topological domain [2]. X_T and Y_T , are both $N \times 1$ column vectors and $[X_T, Y_T]$ in eqn. (3) is the TC set for the network. i.e., its i^{th} row corresponds to the TCs of the i^{th} node. We use the Extreme Node Search (ENS) algorithm [9] for anchor selection, which provides better topology maps with a very small set of anchors.

B. Modified Topolgy Preserving Maps

Although $P_{SVD}^{(2)}$ and $P_{SVD}^{(3)}$ reconstruct the lost directional map, ignoring the most significant component $P_{SVD}^{(1)}$ leads to compression, especially for the sensor nodes at the outer boundary of TPM which are far from the center in radial distance, due to the lack of radial distance information. This compression thus introduces significant errors and nonlinear distortion compared with a physical map. The accuracy of tracking performance at the edges of network will suffer from this inaccuracy. Therefore we use the following modification to generate a TPM that is less distorted. We still keep the 2-dimensional angular information obtained from $P_{SVD}^{(2)}$ and $P_{SVD}^{(3)}$ and at the same time take radial information in $P_{SVD}^{(1)}$ into consideration. Consider a node in sensor network with TCs (x_T, y_T) . Note that (x_T, y_T) is the row corresponding to the node in $[X_T, Y_T]$ given by eqn. (3). The corresponding SVD components for this node are $p_{SVD}^{(1)}, p_{SVD}^{(2)}, p_{SVD}^{(3)}$ and etc., which are extracted from this node's corresponding vector in P_{SVD} . We keep the directional information as the angle of sensor node's TCs to the origin in TPM:

$$\theta_T = \tan^{-1}(y_T/x_T) \quad (4)$$

The distance r_T between sensor node and the origin in TPM is:

$$r_T = \sqrt{x_T^2 + y_T^2} = \sqrt{(p_{SVD}^{(2)})^2 + (p_{SVD}^{(3)})^2} \quad (5)$$

We modify r_T by weighting r_T by the radial information in $p_{SVD}^{(1)}$ as followed:

$$r'_T = \sqrt{(p_{SVD}^{(1)})^2 + (p_{SVD}^{(2)})^2 + (p_{SVD}^{(3)})^2} \quad (6)$$

Modified TCs of this node (x'_T, y'_T) can be rewritten as:

$$x'_T = r'_T \times \cos \theta_T \quad (7)$$

$$y'_T = r'_T \times \sin \theta_T \quad (8)$$

We denote by $[X'_T, Y'_T]$ the matrix of modified TCs obtained by applying eqn. (4), (6) – (8) to $[X_T, Y_T]$ in eqn. (3). In this modification, for each node the directional information in $p_{SVD}^{(2)}$ and $p_{SVD}^{(3)}$ is kept and the radial distance in $p_{SVD}^{(1)}$ is also considered in modified r'_T so that the compression at map edges can be reduced. $[X'_T, Y'_T]$ are referred to as TCs in the remainder of the paper

Figure 1 shows the geographic map of the first test network (TN1) used for evaluation of TCTP. TN1 is in 120unit \times 90unit irregular field and consists of 5137 sensor nodes, each with communication range of 1 unit. Figure 2(a) and Figure 2(b) show original TPM generated using eqn. (1) – (3) and modified TPM generated based on eqn. (4), (6) – (8). In Figure 2(b), edges are decompressed significantly compared to those of Figure 2(a). In Figure 1 and Figure 2, triangle nodes in network are the 30 anchors chosen by ENS algorithm.

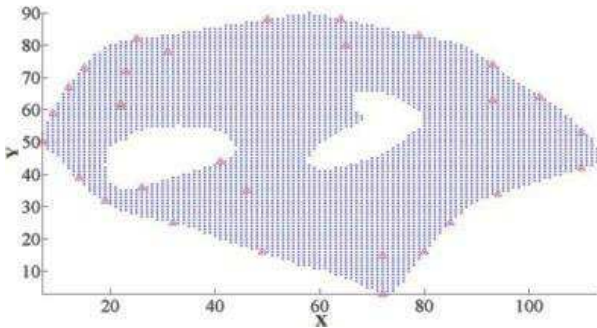


Figure 1. Geographic Map of TN1

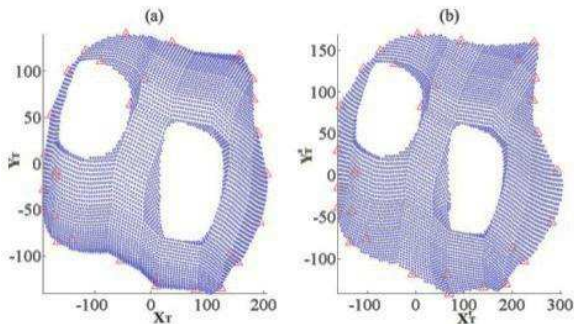


Figure 2. a) TN1's original TPM generated from SVD and b) TN1's modified TPM

C. Virtual Coordinates for Mobile Targets

To do the tracking in TC domain, TCs of mobile target need to be generated. To generate TCs of the mobile node at a point using existing approach, VCs of it need to be evaluated first. The challenge is how to evaluate mobile node's shortest path hop distances to anchors without flooding from anchors. It

is also possible that the mobile node changes the network connectivity thus affecting other VCs.

Next, we present an approximation scheme for getting location of mobile target in TPM with less computation and energy consumption. The mobile target can only communicate with neighbor sensors within communication range. Firstly, VCs of mobile target is obtained by taking average of neighboring sensors' VCs. The averaged VCs provide mobile target's location in VC domain. Secondly by applying SVD in eqn. (3) and modification in eqn. (4), (6) – (8), TCs of mobile target is generated which can be seen as an estimated location in TPM. Note that the accuracy of TCs of the mobile node in the network highly depends on the accuracy of this approximation.

Consider mobile target that is surrounded by n neighbor sensors. Let MT , a $1 \times M$ vector, be the mobile target's VCs and P_{N_i} , a $1 \times M$ vector, be the i^{th} neighbor sensor's VCs. So MT can be obtained by eqn. (9):

$$MT = \sum_{j=1}^n P_{N_i} / n \quad (9)$$

$$MT_{SVD} = MT \times V_A \quad (10)$$

In eqn. (10), MT_{SVD} is a $1 \times M$ vector of mobile target's SVD components. Second and third element of MT_{SVD} are selected as original TCs (x_{MT}, y_{MT}) for mobile target. Modified TC (x'_{MT}, y'_{MT}) can be obtained using eqn. (4), (6) – (8).

III. TOPOLOGICAL COORDINATE BASED TRACKING AND PREDICTION (TCTP) ALGORITHM

This section proposes the TCPC algorithm for mobility tracking and prediction in TC domains. Following terms are used in TCTP algorithm:

1. Sampling time: the time difference between two consecutive sensing locations for mobile target.
2. Detection ellipse: The area surrounding mobile target's predicted position, in which the mobile target is predicted to appear at a given future time. The area is in the shape of an ellipse. Detection ellipse compensates for errors in prediction of the position. The major axis and minor axis of detection ellipse can be adjusted so that it can cover area of different sizes to meet the requirements of different applications.
3. Detecting sensors: Static sensors inside detection ellipse. The number of detecting sensors varies with the size of ellipse.
4. (Prediction) Time Window: Mobile target's track information is sampled at time t_1 and its position is to be predicted at future time $t_f = t_1 + t_p$ where t_p is prediction time. Mobile target is expected to appear at predicted location at time t_f . We set up a time window Δt and we expect mobile target's arrival to be within time $(t_f - \Delta t, t_f + \Delta t)$.
5. Detection failure rate: Detection failure rate is the probability that the mobile target is not detected by any detecting sensor in the time window Δt . Detection failure rate is the main evaluation metric for proposed algorithm.

In TCTP algorithm and related simulation, we make the following assumption:

1. There is one mobile sensor node (target) travelling in the network. BS tries to track and monitor this mobile target in TC domain.

2. The time delay caused by communication among sensors is considered to be negligible compared to the time it takes the node to change its neighborhood.

3. BS possesses the network's TPM, receives mobile target's averaged location in VC domain, calculates its corresponding TCs, tracks its current position in TC domain, predicts the future location and then alerts the sensors in the detection ellipse so that they can wait for the arrival of mobile target.

Traditional tracking and prediction algorithms in geographic domain operate by recording the motion track. Information such as current motion velocity and direction are then used to linearly predict mobile target's position at a future time [6][7][8]. TCTP algorithm follows this basic idea for prediction, but replaces GCs by TCs for current velocity and direction calculation as well as future position prediction. Mobile target's TC position at current time t_i is (x_{MT_i}, y_{MT_i}) and at previous time t_{i-1} is $(x_{MT_{i-1}}, y_{MT_{i-1}})$. The TC domain velocity V_T and direction angle α_T can be calculated using eqn. (11) and (12):

$$V_T = \frac{\sqrt{(x_{MT_i} - x_{MT_{i-1}})^2 + (y_{MT_i} - y_{MT_{i-1}})^2}}{t_i - t_{i-1}} \quad (11)$$

$$\alpha_T = \cos^{-1} \frac{x_{MT_i} - x_{MT_{i-1}}}{\sqrt{(x_{MT_i} - x_{MT_{i-1}})^2 + (y_{MT_i} - y_{MT_{i-1}})^2}} \quad (12)$$

$$x_{MT_{i+1}} = x_{MT_i} + V_T t_p \cos(\alpha_T) \quad (13)$$

$$y_{MT_{i+1}} = y_{MT_i} + V_T t_p \sin(\alpha_T) \quad (14)$$

$(x_{MT_{i+1}}, y_{MT_{i+1}})$ are calculated using eqn. (13) and (14) as the estimated future position at time t_{i+1} for mobile target in TC domain after prediction time t_p from current time t_i .

There are three phases in TCTP algorithm as follows:

1. Estimation of TCs of mobile target (sampling)

Every sampling time, mobile target communicates with neighbor sensors within 1-hop range. VCs of mobile target's current position are calculated using eqn. (9) and sent to BS, e.g., by mobile target. BS uses SVD method to calculate corresponding modified TCs in eqn. (10) and eqn. (4), (6) – (8).

2. BS predicts and sets up detecting sensors

BS receives updated VCs of mobile target every sampling time and records its motion history in TC domain. To predict mobile target's future position, BS calculates current TC velocity and direction using eqn. (11) and (12). Then based on current TC velocity and direction, BS calculates mobile target's position in future using eqn. (13) and (14). After locating the future position in TPM, BS sets the detection area as an ellipse. The predicted position is the center of ellipse and major axis of ellipse is set to be perpendicular to the estimated direction of motion in TC domain.

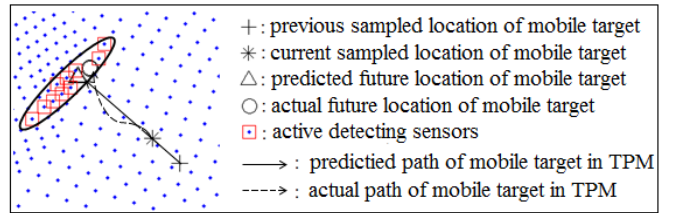


Figure 3. Detecting sensors setup when prediction position is within network's boundary

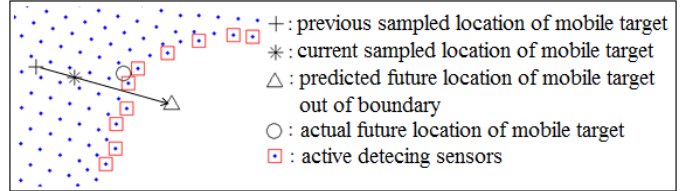


Figure 4. Detecting sensor setup when prediction position is out of network's boundary

Figure 3 shows an example of ellipse detection area. The solid line with arrow is the linear prediction path for mobile target in TC domain, and the dashed line with arrow is its actual motion path. The setup of detecting sensors in ellipse is designed to guarantee the detection of mobile sensor although its track is nonlinear in TC domain. In the case where the predicted position is out of the boundary of TPM, detection ellipse will be adjusted so that the boundary nodes surrounding the projected path are considered to be the candidate sensors downstream for tracking. Figure 4 shows detecting sensors on the boundary that are alerted when the prediction position is out of network's boundary.

3. Detecting sensors wait for mobile target's arrival

Detection ellipse is determined at the BS. Any message that is to be delivered to the mobile node around future time t_f is sent to the detecting sensors inside detection ellipse. Alternatively, in other scenarios of operation, detecting sensors may be woken up by BS to wait for mobile target's arrival.

IV. SIMULATION RESULTS AND DISCUSSION

In this section we evaluate the performance of TCTP algorithm. A simulator was developed using MATLAB@ 2012a. First we use test network TN1 with TPM in Figure 2(b). We evaluate the algorithm using two mobility models to generate the movement in the physical domain, namely, the random direction and random waypoint models [10]. In random direction model, mobile target travels in a random direction until reaching the boundary of the network. After pausing for a certain time, mobile target node continues traveling in a new random direction. In random waypoint model, mobile target randomly selects a physical position as destination to move to, and after reaching it, mobile target randomly selects another position as next destination [10]. The results presented are based on approximately 900 prediction test positions along motion track for each mobility model.

Mobile target's tracks with two models are shown in Figure 5 (a) and (b). Mobility of the node occurs in the geographic domain, i.e., moving in a straight line corresponding to one in geographic domain. The velocity of mobile target in geographic domain is constant at 0.5 units/s.

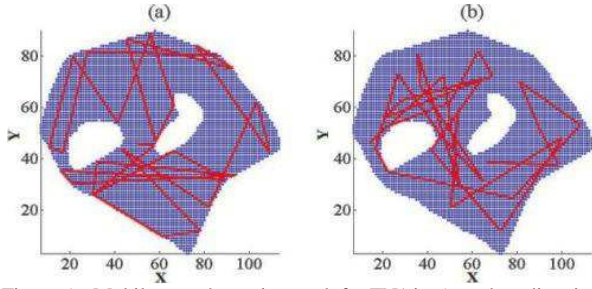


Figure 5. Mobile target's motion track for TN1 in a) random direction mobility model and b) random waypoint mobility model

To evaluate the effectiveness of TCTP algorithm that operates in the TC domain without any physical information, we also use the same tracking algorithm in geographic domain, in which case the tracking and prediction are both based on GCs. This is the existing approach, and as such it serves as the baseline for comparison. Note that the GC based approach has the added advantage of having a constant velocity due to the mobility models used. In our simulation, we sample the position of the mobile node every 1s, 2s, 4s, 6s, 8s and 10s. Every sampling time, the nodes position in the appropriate coordinate system is informed to BS. We set time window for prediction as 0s, 2s, 4s, 6s and 8s. We predict mobile target's position after prediction time 5s, 10s and 20s. For the number of detecting sensors, we choose 10 and 20 per each prediction test position, which corresponds on average to 0.19% and 0.39% of sensors in the network respectively.

Before comparing tracking in TC and GC domains, we point out that the geographic distance measurement provides continuous changes, whereas TCs of the mobile node change only at discrete instances as its neighborhood changes. This is due to the fact that the VCs of the target are obtained by averaging neighbors' VCs. As such tracking in TC domain is not very effective when sampling is done at very fine granularity. This is different from GCs based tracking and prediction algorithm where shorter sampling time leads to optimal detection failure rate [6]. Figure 6(a) shows the variation of detection failure rate vs. sampling time for TN1 in random direction mobility model in time window 4s when there are 20 detecting sensors in detection ellipse. When sampling time is short, i.e., when the travelled distance during the period is shorter than the communication range, this estimated average position is less accurate compared with actual position, due to the average location approximation. If the distance travelled in sampling interval is longer, the accuracy increases. However, if sampling time is too long, the tracking information cannot be updated in time so tracking performance deteriorates in both TC and GC based methods.

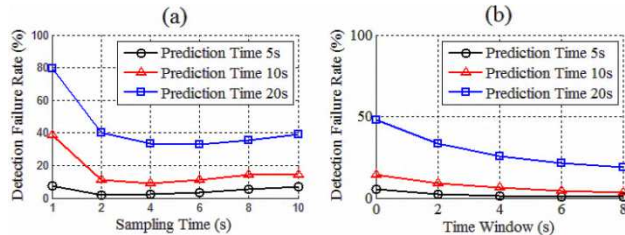


Figure 6. Detection failure rate for TN1 in random direction mobility model with 20 detecting sensors vs. sampling time, a) when time window is 4s, and b) when sampling time is 4s

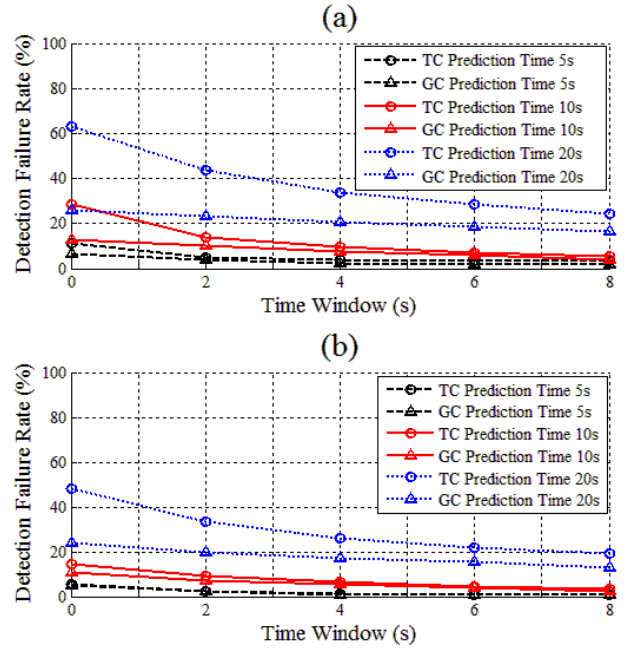


Figure 7. Detection failure rate comparison between TC domain and GC domain for TN1 in random direction mobility model, with a) 10 detecting sensors, and b) 20 detecting sensors

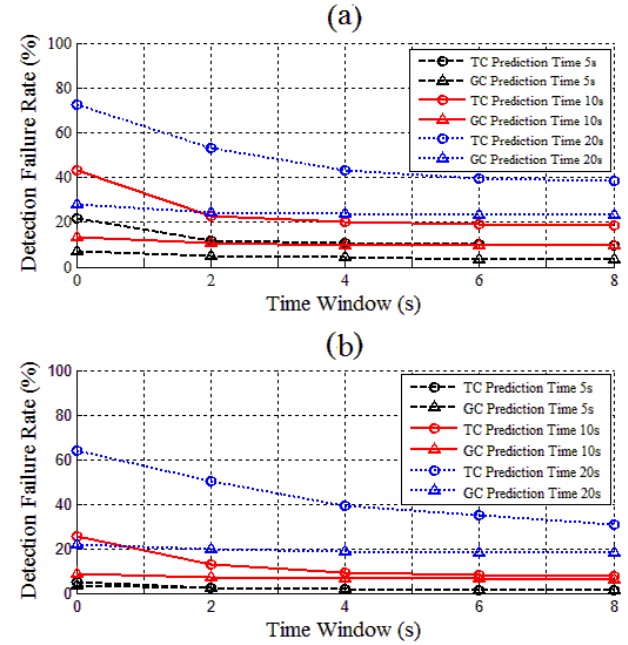


Figure 8. Detection failure rate comparison between TC domain and GC domain for TN1 in random waypoint mobility model, a) with 10 detecting sensors, and b) with 20 detecting sensors

Figure 7 and Figure 8 compare the variation of detection failure rate for TN1 in different time windows for tracking in TC and GC domains when sampling time is 4s. For short-term prediction like 5s and 10s under motion velocity of 0.5 units/s, we can see that the detection failure rate in TC and GC domains are quite close to each other when time window is 2s or longer. For long-term prediction, e.g., 20s into future, the longer the time window the lower detection failure rate will be obtained. It's a challenge to do long term prediction, both in

TC and GC domains, due to the linear prediction model. This weakness can be improved by increasing the number of detecting sensor nodes or area of detection ellipse. However, the important conclusion is that the TC domain tracking is very competitive with that in the GC domain. Furthermore, the lower cost of TC domain tracking, e.g., in terms of power due to absence of GPS or localization algorithms, also means that we may be able to increase the size of the detection ellipse in TC domain, thus enhancing its effectiveness.

We have also evaluated TCTP algorithm in second test network (TN2) with 2500 sensor nodes in random placement, a part of which is shown in Figure 9(a). Unlike TN1, the distances between adjacent nodes in TN2 are not constant. Mobile node follows random waypoint mobility model and the number of detecting sensors is 10. Mobile node's velocity is 0.5 units/s. Figure 10 shows the detection failure rate when sampling time is 10s. TC based approach comes within 20% of the GC based approach for a wide range of time windows, with lower rates at higher time windows (within 10% of GC based approach after time window 6s). TCTP shows competitive tracking and detection performance compared with GC based approach although physical distances among sensor nodes are random and unknown. It's significant that, although without any geographic information, TCTP is able to achieve similar failure rate compared with geographic information based approaches.

V. CONCLUSIONS AND FUTURE WORK

Tracking and prediction of position of mobile targets using sensor networks have hitherto been accomplished using geographic positions and distances. However, measurement of geographic distances is expensive (e.g., needs GPS), unreliable (when distance estimation is based on signal strength or time delay), or simply not feasible (e.g., in harsh environments and indoor environments). We presented a new tracking and prediction algorithm called Topological Coordinate based Tracking and Prediction (TCTP) algorithm for mobile targets, which does not require physical distance information. Instead of doing tracking in geographical domain, it does the tracking in Topological Coordinate (TC) domain. TCs are generated easily from hop-distance for each node to a small subset of nodes. Our simulation results show that even without any geographic information, tracking and prediction mobile target using TCs is effective. A modification was proposed for Topology Preserving Maps (TPMs) generation to enhance the accuracy and provide a 'good' guide map for mobility tracking and prediction. TCTP algorithm has competitive performance compared with same algorithm operating in geographic domain for a wide range of prediction parameters. The number of sensors, size of detection ellipse and window time can be adjusted in different applications. Future work includes the analysis of the algorithm with a view to obtaining the optimum operating parameters.

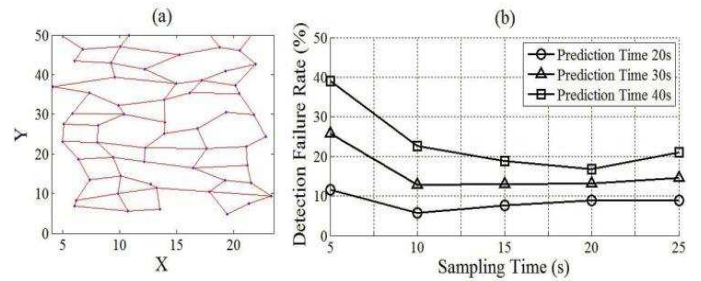


Figure 9. a) Part of physical topology of TN2 with random placed nodes, and b) detection failure rate vs. sampling time for TN2 in time window 6s

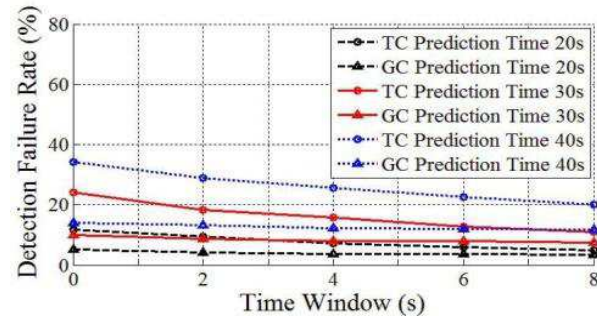


Figure 10. Detection failure rate comparison between TC domain and GC domain for TN2 when sampling time is 10s

REFERENCES

- [1] H. A. B. F. Oliveira, E. F. Nakamura, A. A. F. Loureiro and A. Boukerche, "Error Analysis of Localization Systems for Sensor Networks," Proc. 13th ACM Int. Workshop on Geographic Information Systems, Germany, 2005, pp. 71 – 78.
- [2] D. C. Dhanapala and A. P. Jayasumana, "Topology Preserving Maps from Virtual Coordinates for Wireless Sensor Networks," Proc. 35th IEEE Conf. on Local Computer Networks, 2010, pp. 136–143.
- [3] D. C. Dhanapala and A. P. Jayasumana, "Geo-Logical Routing in Wireless Sensor Networks", Proc. 8th IEEE Conf. on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011, pp. 305 – 313.
- [4] K. Liu and N. Abu-Ghazaleh, "Aligned Virtual Coordinates for Greedy Routing in WSNs," Proc. IEEE Int. Conf. on Mobile Adhoc and Sensor Systems, 2006, pp. 377–386.
- [5] D. C. Dhanapala and A. P. Jayasumana, "On Boundary Detection of 2-D and 3-D Wireless Sensor Networks", Proc. IEEE Global Telecommunications Conference (GlobeCom), 2011, pp. 1 – 5.
- [6] H. Yang, and B. Sikdar, "A Protocol for Tracking Mobile Targets Using Sensor Networks," Proc. First IEEE Int. Workshop on Sensor Network Protocols and Applications, 2003, pp. 71 – 81.
- [7] J. Hsu, C. C. Chen and C. Chi, "Short-term Prediction-based Optimistic Object Tracking Strategy in Wireless Sensor Networks", Proc. Int. Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing, 2011, pp.78 – 85.
- [8] Y. Xu, J. Winter and W. C. Lee, "Dual Prediction-Based Reporting for Object Tracking Sensor Networks," Proc. First Int. Conf. on Mobile and Ubiquitous Systems: Networking and Services, 2004, pp. 154 – 163.
- [9] D. C. Dhanapala and A. P. Jayasumana, "Anchor Selection and Topology Preserving Maps in WSNs — A Directional Virtual Coordinate based Approach", Proc. IEEE 36th Conf. on Local Computer Networks, 2011, pp. 571 – 579.
- [10] T. Camp, J. Boleng and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", Wireless Communication and Mobile Computing: Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2002, vol. 2, no. 5, pp. 483-502.