Deflection-Aware Tracking-Principal Selection in Active Wireless Sensor Networks

Fan Zhou, *Student Member, IEEE*, Goce Trajcevski, *Member, IEEE*, Oliviu Ghica, Roberto Tamassia, *Fellow, IEEE*, Peter Scheuermann, *Life Fellow, IEEE*, and Ashfaq Khokhar, *Fellow, IEEE*

***Abstract*—This paper addresses the problem of energy efficiency balanced with tracking accuracy in wireless sensor networks (WSNs). Specifically, we focus on the issues related to selecting *tracking principals*, i.e., the nodes with two special tasks: 1) coordinating the activities among the sensors that are detecting the tracked object's locations in time and 2) selecting a node to which the tasks of coordination and data fusion will be handed off when the tracked object exits the sensing area of the current principal. Extending the existing results that based the respective principal selection algorithms on the assumption that the target's trajectory is approximated with straight line segments, we consider more general settings of (possibly) continuous changes of the direction of the moving target. We developed an approach based on particle filters to estimate the target's angular deflection at the time of a handoff, and we considered the tradeoffs between the expensive in-node computations incurred by the particle filters and the imprecision tolerance when selecting subsequent tracking principals. Our experiments demonstrate that the proposed approach yields significant savings in the number of handoffs and the number of unsuccessful transfers in comparison with previous approaches.**

***Index Terms*—Moving-object tracking, network coverage, principal selection, sensor network.**

I. Introduction

ENERGY-EFFICIENT tracking of moving objects is one of the canonical problems in wireless sensor network (WSN) research [1], [2], and one specific aspect of the problem is the tradeoff between the energy efficiency and tracking accuracy [3]. In many approaches, the sensor nodes are typically organized in (hierarchical) clusters [4]–[6] according to their geographical positions, and a designated sensor node is elected as the *tracking principal* of each cluster, acting as a temporal data fusion center and a coordinator of the tracking process. To localize the target and to monitor the target's movement information such as speed, acceleration, and moving direction, tracking principals communicate with the member nodes of their respective clusters and obtain the data regarding range/distance measurements [7], which they combine with its own observations.

In addition to coordinating the process, an important responsibility of an on-duty tracking principal, which is the focus of this paper, is *selecting the next principal* and, when needed, handing off the target tracking information to the selected one. There are some fundamental criteria to be obeyed when designing the tracking-principal selection and handoff scheme.

- The sequence of the selected principals should be able to cover the trajectory of the moving object being tracked.
- The number of handoffs between consecutive principals should be minimized to reduce the overhead due to the handoff communication. Conversely, each principal should "cover" a large portion of the tracked object's motion as possible.

An exemplary scenario illustrating the motivation and the main intuition behind this paper is shown in Fig. 1. The current tracking principal P_t at time t needs to select a node that will take the role of the cluster head at the next sampling epoch, i.e., starting time $t+1$. The handoff schemes that predict the trajectory of the tracked target based on the past movement information (cf. [8] and [9]) would consider the location shown with the gray squares along the dashed line in Fig. 1. Based on this, P_t would select the node S_i as the next tracking principal because it yields the largest coverage of the predicted trajectory, regardless of whether the *continuous* straight line segment is used (cf. [8]) or the *discrete* sampling instants are considered (cf. [9]). However, if the moving object deviates a small angle, e.g.,φ from the expected trajectory, as shown with the darker squares along the solid curve in Fig. 1, the node S_i will cover a much smaller portion of the actual trajectory or only two sampling time instants if discrete location sampling is considered. However, in this case, another sensor node, i.e., S_j, will be able to cover a larger portion of the trajectory or four actual locations (denoted as read squares) at subsequent sampling epochs. This qualifies S_j to be a better tracking principal than S_i.

The motivation for this paper is based on the observation that the straight-line-segment-based prediction of the actual position of target may be biased and that even small deviations may significantly decrease the performance of the tracking-principal selection algorithm, a consequence of which would

Manuscript received November 10, 2011; revised April 9, 2012; accepted May 17, 2012. Date of publication May 30, 2012; date of current version September 11, 2012. This work was supported by the U.S. National Science Foundation under Grant CNS-00910952 and Grant CCF-0830149. The review of this paper was coordinated by Prof. A. Jamalipour.

F. Zhou is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: fan.zhou.uestc@gmail.com).

G. Trajcevski, O. Ghica, and P. Scheuermann are with Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208 USA (e-mail: goce@eecs.northwestern.edu; ocg474@eecs.northwestern.edu; peters@eecs.northwestern.edu).

R. Tamassia is with the Department of Computer Science, Brown University, Providence, RI 02912 USA (e-mail: rt@cs.brown.edu).

A. Khokhar is with the Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, IL 60607 USA (e-mail: ashfaq@uic.edu)

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2012.2201188

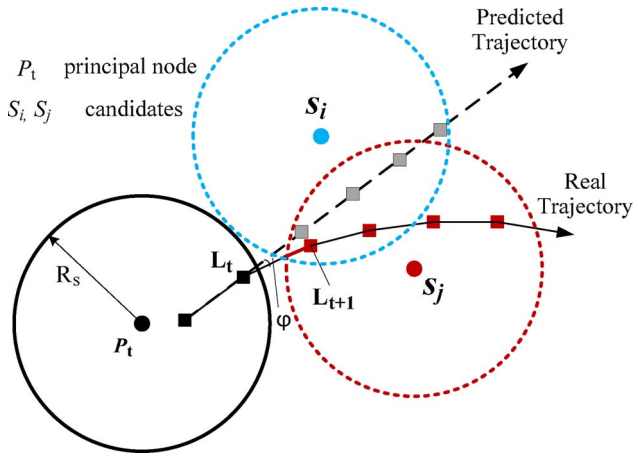


Fig. 1. Motivation.

be an increase in the in-network communication overhead. This happens because of increasing the number of handoffs between successive tracking principals that need to convey data to each other, i.e., along the trajectory of the tracked object. The existing related works on target tracking and coverage often model the trajectory of the tracked object as a sequence of straight line segments [1], [8]–[10] and, consequently, cannot quite capture the changes in the (description of the state) target's motion. Arguably, without taking into proper consideration the target's real location at the handoff step, any location-based precalculation may be inaccurate, particularly in WSN settings, where the problems of time delay/synchronization [11] can be amplified by the drift of the accumulated position error due to mispredictions.

The main objective of this paper is to derive efficient schemes aimed at reducing the number of tracking principals required to cover the moving object's trajectory. We achieve this by incorporating the effects of angular deflection of the target's motion, particularly at the time instants at which the handoff between principals occur. We addressed both *continuous* and *discrete* coverage of the moving object's trajectory by active sensor nodes, considering different target mobility models in which the directions of target continuously change. Our main contributions can be summarized as follows.

- We propose an adaptive algorithm, which is called *deflection-aware selection (DAS)*, for tracking-principal selection, which is aimed at maximizing the spatiotemporal tracking coverage for objects with nonlinear trajectories.
- A particle-filtering-based method, which we will refer to as *dead-reckoning*-based *particle filtering (DRPF)* is presented for recursively calculating the target's angular deflection, which, by leveraging the dead-reckoning prediction within permissible error bound, can significantly reduce the computation and communication overhead of state estimation.
- We analyze the upper and lower bounds on the number of principals required to cover an arc trajectory and demonstrate that DAS yields significant improvement on reducing the number of principal transfers and, thus, in-network communication, compared with existing approaches.
- We present the extensive simulation results to evaluate the proposed approaches and show that our methods outperform existing schemes under a wide range of conditions.

The rest of this paper is structured as follows. In Section II, we introduce the basic settings regarding the network model and present our approach to estimating the direction deviation based on particle filtering. Section III derives the theoretical upper and lower bounds on both continuous and discrete coverage of a mobile target and presents the *DAS* tracking-principal selection algorithm. An extensive experimental comparison of the benefits of our approach on principal selection with respect to the previous methods is presented in Section IV. We discuss the previous works and position our results with respect to the related literature in Section V. The concluding remarks, along with possible future extensions, are discussed in Section VI.

II. Direction Deflection Estimate Using Particle Filters

We first discuss our assumptions regarding the network model and review the localization-related issues in the tracking process, following with the basics of target-state estimation with particle filtering. Subsequently, we present in detail the main steps of the direction deflection estimate in the framework of particle filtering and our *DRPF* methodology.

A. Network Model

We consider a WSN consisting of N homogenous static sensor nodes $S = (S_1, S_2 \ldots S_N)$ deployed over a 2-D surveillance field $F \subset \mathbb{R}^2$. We assume that all the nodes have identical communication range R_c and sensing range R_s. A given sensor node locates the target in its sensing area via some range-based methods, e.g., received signal strength indicator (RSSI) or time difference of arrival [12]. Each node is assumed to be aware of its own location and the locations of its one-hop neighbors, i.e., the nodes within its communication range, i.e., a pair of sensor nodes (S_i, S_j), $i, j \in [1, N]$, can communicate to each other *iff* their Euclidean distance is no greater than the radio communication range, e.g., $\|S_i - S_j\| \leq R_c$. We also assume that the communication range of each node is at least twice of its sensing range, which guarantees both full coverage of the sensing region and network connectivity [13], [14]. Finally, we assume that the network is dense enough to ensure that a moving target can be tracked while it travels in the area of nodes' deployment.

$\mathcal{D}(S_i, R_s)$ denotes the disk centered at the location of sensor S_i with radius R_s, and $\mathcal{N}(S_i)$ denotes the set of one-hop neighboring nodes of S_i, which are distributed within the communication range R_c of S_i.

Contrary to the more passive approaches, e.g., using acoustic sensors to estimate the location of the moving object during collaborative tracking [15], in this paper, we focus more on active scenarios, where tracking sensors also collaborate for estimating the target's state. We reiterate that our main objective is developing criteria for selection of the subsequent tracking principle [8], [9].

The symbols used throughout the rest of this paper are summarized in Table I.

TABLE I
LIST OF NOTATIONS

Notation	Description
S_i	Sensor node i
P_t	Tracking principal at sampling epoch t
R_c	Communication range
R_s	Sensing range
$\mathcal{D}(S_i, R_s)$	Disk sensing area of S_i
$\mathcal{N}(S_i)$	The set of neighborhood nodes of S_i
$\mathbf{x}_t$	Target state at epoch t
$\mathcal{Z}_{0:t}$	The sequence of measurements $\mathrm{z}_0 ... \mathrm{z}_t$
$\mathcal{Z}_t(S_i)$	Measurements of S_i at time t
$\hat{L}_t$	Estimated 2D coordinates $(\hat{x}_t, \hat{y}_t)$
ω_t^i	The weight of i^{th} particle
θ_t	Previous moving direction
φ_t	Angular deflection at t

B. Target-State Estimation With Particle Filters

Particle filtering is a probabilistic framework for sequentially computing the target's state based on the Monte Carlo method. The key idea of particle filtering is to use a number of independent random variables called *particles*, which are sampled directly from the state space, to represent the posterior probability. Then, the posterior can be updated in time using the importance sampling method [16], which is based on target dynamics and the observation likelihood model. It has the flexibility to accommodate arbitrary nonlinear motion patterns and multimodal likelihood models at the computational complexity, which depends largely on the number of particles. However, for our application, even small number of particles works well, as we will show, by utilizing the *DRPF* scheme.

Let $\mathbf{x}_t$ indicate the position distribution of target at time t, and let $\mathcal{Z}_{0:t}$ indicate the observation sequence $\{\mathcal{Z}_0, \mathcal{Z}_1 \dots \mathcal{Z}_t\}$, corresponding to the conditionally independent measurements with respect to target's location sequence $\{\mathbf{x}_0, \mathbf{x}_1 \dots \mathbf{x}_t\}$. Specifically, particle filtering consists of two main processes, *predict* and *update*, which can be, respectively, expressed as follows:

$$P(\mathbf{x}_t|\mathcal{Z}_{t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) p\left(\mathbf{x}_{t-1}|\mathcal{Z}_{0:t-1}\right)\, d\mathbf{x}_{t-1} \quad (1)$$

$$p(\mathbf{x}_t|\mathcal{Z}_{0:t}) = \frac{p(\mathcal{Z}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathcal{Z}_{0:t-1})}{p(\mathcal{Z}_t|\mathcal{Z}_{0:t-1})}. \quad (2)$$

The likelihood $p(\mathcal{Z}_t|\mathbf{x}_t)$ is the measurement equation and noise model, prior distribution $p(\mathbf{x}_t|\mathcal{Z}_{0:t-1})$ represents the knowledge of the mobility model, the denominator $p(\mathcal{Z}_t|\mathcal{Z}_{0:t-1})$ is a constant called evidence, and the expression $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is the target-state transition density and describes the dynamics of the target model.

Equation (1) is commonly referred to as the *predict* process, which computes the prior distribution for the next time t based on the posterior distribution estimation at time $t-1$ and transition density $p(\mathbf{x}_t|\mathbf{x}_{t-1})$. Equation (2), on the other hand, corresponds to the *update* process that calculates the posterior distribution of the target state by involving the measurements $\mathcal{Z}_t$ at time t.

C. Direction Deflection Estimate via Particle Filtering

We now proceed with the details of direction deflection estimate described in the framework of particle filtering.

Let φ_t denote the angular deflection at time t from the previous moving direction θ_t, which is determined by the positions of the target at previous time steps. Further, let $\mathbf{x}_t^1, \mathbf{x}_t^2 \dots \mathbf{x}_t^N$ denote the independent particles, where N is the number of particles and ω_t^i denotes the weight of particle $\mathbf{x}_t^i$.

Our goal is to estimate the posterior distribution of the angular deflection $p(\varphi_t|\mathcal{Z}_{0:t})$, given the observations up to time t. At each time step, when measurements are available, the posterior distribution of φ_t is updated with new observations $\mathcal{Z}_t(S_i)$ $(S_i \in \mathcal{N}(P_{t-1}))$. This process evolves along the time according to the state model, and a detailed discussion of the main steps of estimating the φ_t using a particle filter follows.

1) Sampling: While the choice of $q(\mathbf{x}_t|\mathbf{x}_{0:t-1}^i, \mathcal{Z}_{0:t}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}^i, \mathcal{Z}_t)$ as the importance density minimizes the variance of importance weight ω_t^i conditional upon $\mathbf{x}_{t-1}^i$ and $\mathcal{Z}_{0:t}$, it requires sampling from $p(\mathbf{x}_t|\mathbf{x}_{t-1}^i, \mathcal{Z}_t)$ and evaluating the integral $P(\mathcal{Z}_t|\mathbf{x}_{t-1}^i) = \int p(\mathcal{Z}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{x}_{t-1}^i) d\mathbf{x}_t$ [17]. An intuitive way of implementing this in WSN settings is to use the state transition density $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$ as the importance function. We notice that it does not include the most recent observations, but it is easy to implement and consumes less computation.

In this paper, since the expected moving direction of a given target at current time t is calculated based on its past locations, the state transition probability density function (pdf) is then chosen as the importance density, namely, $q(\mathbf{x}_t|\mathbf{x}_{0:t-1}^i, \mathcal{Z}_{0:t}) = p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$.

Therefore, the ith particle's importance weight ω_t^i can be recursively computed in time with observations and a transition function in the form of

$$\begin{aligned}\omega_t^i(\mathbf{x}_t) &= \frac{p\left(\mathbf{x}_{0:t}^i\right)|\mathcal{Z}_{0:t})}{q\left(\mathbf{x}_{0:t}^i|\mathcal{Z}_{0:t}\right)} \\ &\propto \frac{p\left(\mathcal{Z}_t|\mathbf{x}_t^i\right) p\left(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i\right)}{q\left(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i, \mathcal{Z}_{0:t}\right)} \times \frac{p\left(\mathbf{x}_{0:t-1}^i|\mathcal{Z}_{0:t-1}\right)}{q\left(\mathbf{x}_{0:t-1}^i\right)|\mathcal{Z}_{0:t-1})} \\ &= \omega_{t-1}^i(\mathbf{x}_{t-1}) \times p\left(\mathcal{Z}_t|\mathbf{x}_t^i\right) \end{aligned} \quad (3)$$

where the incremental weight relies only on the likelihood of the observations $p(\mathcal{Z}_t|\mathbf{x}_t^i)$.

Suppose that the angular deflection φ_t follows the uniform distribution in $[-\varphi_m, \varphi_m](\varphi_m \in [0, \pi])$ and the speed of the target is uniformly distributed within $[V_{\min}, V_{\max}]$. Then, the probability of the target being at a given location at time t, based on past locations, can be expressed as

$$p\left(\mathbf{x}_t|\mathbf{x}_{t-1}, \varphi_t\right) = \frac{1}{\varphi_m\left(V_{\max}^2 - V_{\min}^2\right)}. \quad (4)$$

Fig. 2 depicts the sampling area (the shaded sector) at time t, determined by the tracked mobility information, where L_{t-2} and L_{t-1} denote the previous positions of the target.

After specifying the sampling area, an additional random sampling step then draws independent identically distributed samples $\mathbf{x}_t^1, \mathbf{x}_t^2 \dots \mathbf{x}_t^N$ from importance the density $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \varphi_t)$.

2) Updating: At time step t, the location of the target is estimated by assimilating the new available sensor-node range measurements. We assume a Gaussian error model for range

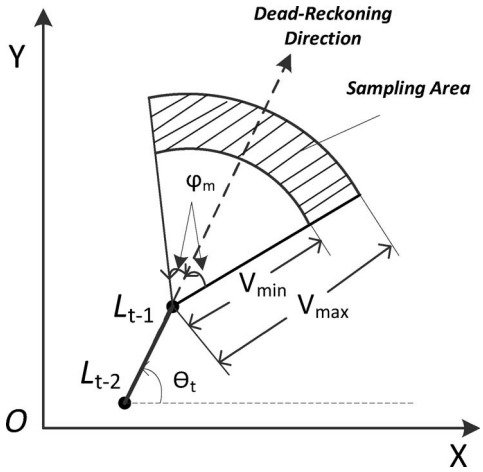


Fig. 2. Sampling area.

measurements of each sensor; then, for particle $\mathbf{x}_t^i$, its weight ω_t^i is recursively calculated as

$$\omega_t^i = \omega_{t-1}^i \prod_{j=1}^{M} \frac{1}{\sqrt{2\pi}\sigma_j} e^{\frac{-\left(z_t(S_j) - z_t^i(S_j)\right)^2}{2\sigma_j^2}}. \tag{5}$$

where M is the number of measurements, and $\mathcal{Z}_t(S_j)$ is the measurement of sensor S_j; σ_j $(j \in [1, M])$ is the standard deviation of measurement noise for S_j, which specifies the confidence of the $\mathcal{Z}_t(S_j)$; and the value $\mathcal{Z}_t^i(S_j)$ denotes the estimated measurement for the ith particle given the location of the sensor S_j. The weight of each particle can be normalized as follows:

$$\omega_t^i = \frac{\omega_t^i}{\sum_{k=1}^{N} \omega_t^k}. \tag{6}$$

The current location $\hat{L}_t$ is updated as the centroid of all the weighted particles as follows:

$$\hat{L}_t(\hat{x}_t, \hat{y}_t) = \left(\sum_{I=1}^{N} \omega_t^i \cdot x_t^i, \sum_{i=1}^{N} \omega_t^i \cdot y_t^i \right). \tag{7}$$

Consequently, the angular deflection φ_t can be estimated as

$$\varphi_t = \arctan\left(\frac{\hat{y}_t - y_{t-1}}{\hat{x}_t - x_{t-1}}\right) - \arctan\left(\frac{y_{t-1} - y_{t-2}}{x_{t-1} - x_{t-2}}\right) \tag{8}$$

where points with coordinates (x_{t-1}, y_{t-1}) and (x_{t-2}, y_{t-2}) are retrieved from the historical trajectory tracking information, which we assume to be contained in a corresponding spatiotemporal buffer C_b.

3) Resampling: One problem of particle filtering methodology is that, after a few iterations, the weights may become highly degenerate because a small number of particles have nearly all of the probability mass. Hence, some resampling may be required to avoid accumulation of the error and to render the particle system more stable. In this paper, the *partial rejection control* (cf. [18]) is being used to reduce the burden of computation while preserving a rejection control.

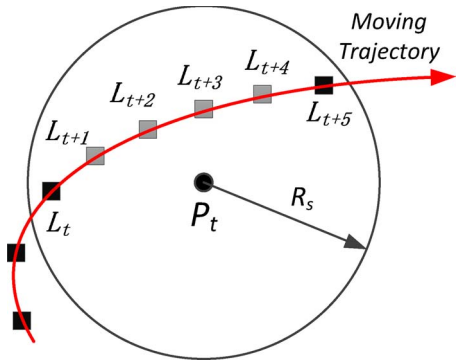


Fig. 3. *DRPF* scheme.

D. Dead Reckoning and Particle Filtering

Equations (4)–(8) specify the process of continuously estimating the deflection of the moving target. However, since particle-filtering-based localization is, to some extent, energy inefficient due to the overhead of intensive computations, it is desirable to derive a lightweight scheme for calculating the target's state. Toward this goal, we present a *DRPF* scheme, which leverages the dead-reckoning prediction of the target's motion pattern to avoid continuous localization at every sampling step. Specifically, *DRPF* uses a threshold δ to control the sampling process of localization and deflection estimation, in which the task of state estimation only happens when the threshold is reached.

Fig. 3 shows the basic idea of *DRPF*, in which the localization tasks in sampling step $t+1, \ldots, t+4$ are saved, and the target's location during this period can be inferred and indexed simply via a dead-reckoning technique based on its moving velocity and direction. In this case, the threshold $\delta = 4$ indicates the number of samplings that tracking principal P_t saves while bounding the target in its monitoring area. After $\delta = 4$ sampling time steps, another target-state estimation task will be triggered.

Thus, *DRPF* is an approach that essentially attempts to provide a tradeoff between the (in)exact position information regarding the target during δ sampling steps and the energy consumption of particle filtering. In other words, *DRPF* dynamically adapts the sampling and state estimation process based on the mobility information. We note that *DRPF* may introduce extra localization error due to expansion of the sampling area. However, as we will show in Section IV, the experimental evaluations verified that substantial computation and communication burden introduced by particle filtering can be saved by employing the *DRPF* scheme.

We note that there may be other definitions of the threshold δ; for example, we can define it as the RSSI value of the tracking principal, below which another process of particle-filtering-based localization will be launched. However, in this paper, we focus on using the dead-reckoning prediction with target's movement information, i.e., velocity and direction, to derive the threshold δ, which is calculated as the number of samples that a tracking principal can tolerate in within a given epoch.

III. Selecting Tracking Principals

We now present our *DAS* algorithm for selecting the tracking principals. We begin this section with reviewing two closely

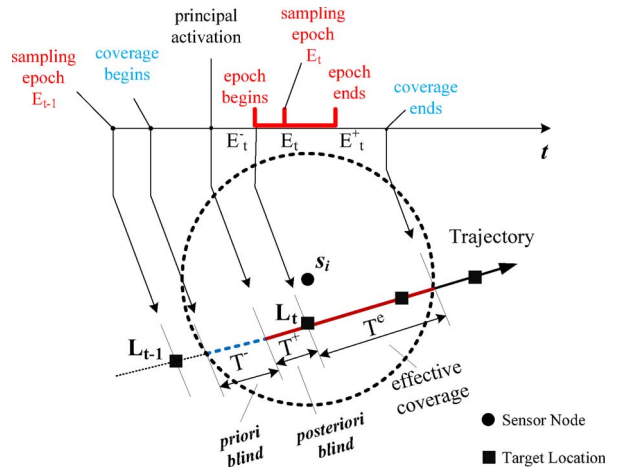


Fig. 4. Spatiotemporal coverage.

related approaches and proceed with discussing the details of the *DAS* algorithm.

A. Principal Selection via Relay Area and Sampling Look-Ahead

A target-movement-prediction-based methodology, which is called *relay-area-based (RAB)* principal selection, is proposed in [8]. The *RAB* method designates an annular sector region, as the *relay area*, from where the next tracking principal is selected. The *relay area* is determined by three tunable parameters φ, $\tilde{D}$, and ω, where φ denotes the central angle of the sector, $\tilde{D}$ is the radius of the outer circle of the sector, and ω defines the width of the annular sector area.

If there is no node residing in the *relay area*, it will be iteratively expanded by increasing the three tunable parameters until at least one sensor node is encompassed within it. It is that particular node that will be selected as the subsequent tracking principal. A lower bound of $\lceil (L/1.1\tilde{D}) \rceil$ on the average number of principals required to cover a line segment L is derived in [8].

To incorporate semantics of the discrete sampling organized in epochs, Ghica *et al.* [9] proposed a tracking-principal selection algorithm, which is called *sampling look-ahead selection (SLS)*. By discriminating between the *blind* coverage (incorporating both *a priori* (T^-) and *a posteriori* (T^+) times), and the *effective* coverage T^e of a given sensor node, *SLS* was able to improve the lower bound proposed by He *et al.* [8] and achieved significant savings on the number of required tracking principals.

An illustration of the *spatiotemporal coverage* by a principal node is shown in Fig. 4, which explains the meaning of the *a priori* T^- and *a posteriori* T^+ *blind* coverage and the one of the *effective* coverage T^e.

While the goal of the *RAB* method is to increase T^e through reducing *a priori blind* coverage, *SLS* also incorporated *a posteriori blind* coverage into the process of tracking-principal selection, thereby reducing the combined impact of T^- and T^+. Following the work in [9], this paper also considers the semantics of the *spatiotemporal coverage* process when selecting the tracking principals.

B. DAS of Tracking Principals

In the sequel, we give a detailed presentation of the main steps of our methodology for selecting tracking principals, separately addressing the issues of continuous coverage and the discrete-time coverage.

1) Trajectory Prediction: Assume that P_{t-1} is the current tracking principal that continuously monitors the target and acts as the fusion center of the mobility information before the target moves out of its coverage disk. When the target moves into the "handoff area," determined both by the target's motion and the sensing range of nodes, the principal P_{t-1} will communicate with its neighbor nodes, requesting them to perform range measurements. Then, at the next tracking epoch starting at time t, the particle-filtering-based localization algorithm *DRPF* is employed to update the location $\hat{L}_t$ and estimate the angular deflection φ_t of the target, using (7) and (8), respectively. More specifically, at time t, the location along the expected trajectory $\mathrm{Traj}(L_k, \tilde{v}I)$ is predicted according to the following formula:

$$\mathrm{Traj}(L_k, \tilde{v} = \begin{cases} \tilde{x}_k = \tilde{x}_t + \tilde{v}I \cdot \cos\left(\theta_t + (k-t)\varphi_t\right) \\ \tilde{y}_k = \tilde{y}_t + \tilde{v}I \cdot \sin\left(\theta_t + (k-t)\varphi_t\right) \end{cases} \tag{9}$$

where the point with coordinates $(\tilde{x}_k, \tilde{y}_k)$ denotes the predicted location of the target trajectory at time step $k (k \geq t+1)$; θ_t is the previous moving direction calculated using the historical movement information, e.g., $\arctan(y_{t-1} - y_{t-2}/x_{t-1} - x_{t-2})$; and $\tilde{v}I$ specifies the expected displacement of the target during the sampling epoch I. We implicitly assumed that $\varphi_t > 0$ if the direction of trajectory is on the left hand of θ_t and $\varphi_t < 0$ if otherwise.

2) Coverage Gain of DAS: As explained in [9], one of the main features of *SLS* is that it minimizes the blind coverage, both *a priori* and *a posteriori* portions, when selecting tracking principals while implicitly increasing the effective coverage. Since it has been demonstrated that *SLS* requires fewer tracking principals when considering the *discrete* nature of sampling epoch, which is also the premise of this paper, we only illustrate the effective coverage gain of *DAS* compared with the *SLS* algorithm.

As a specific example, consider the scenario shown in Fig. 5, involving two candidate tracking principals P_1 and P_2. We have the following observations in order.

- The *SLS* method selects P_1 as the principal node when taking into consideration of the *discrete* nature of the sampling epochs because it exhibits maximal linear predicted effective coverage $\tilde{T}^e$ (see the dashed line in Fig. 5).
- Although *SLS* outperforms the *RAB* method in terms of reducing tracking-principal transfers, the benefit originates from "predicting" the *posterior* blind coverage and hence "forward-looking" the next sampling location of target. One of the problems, as stated in [9], is that it introduces more "unsuccessful" principal transfers in contrast with *CPS* and *RAB* approaches.
- However, we also observe that the object deviates an angle φ_t from the previous moving direction after transmitting the tracking information from the previous principal to P_1. Therefore, unlike *SLS*, *DAS* will select the node P_2 as the next tracking principal when accounting for the

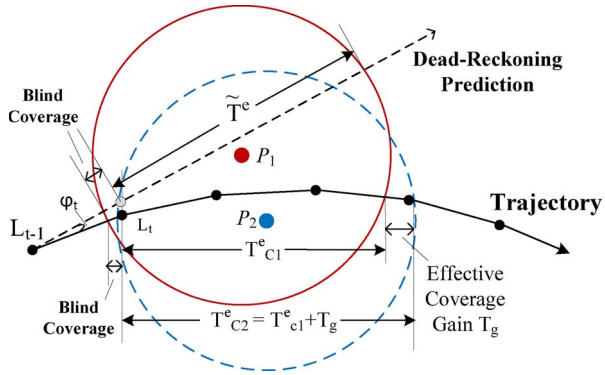


Fig. 5. *DAS* versus *SLS*.

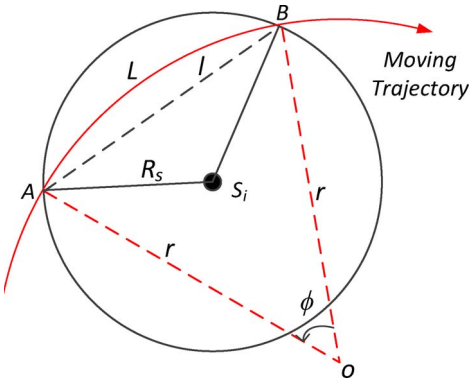


Fig. 6. Trajectory coverage by a tracking principal.

angular deflection of the target at sampling time step t. As shown in Fig. 5, candidate P_2 may yield a coverage gain T_g on the effective coverage compared with candidate P_1. Apparently, we have $T^e_{c2} > T^e_{c1}$, which demonstrates that P_2 is a better choice.

- Finally, as has been stated in the previous section, *DAS* is a safer principal transfer approach compared with *SLS*, due to deferring the location prediction to time step t after involving measurements rather than relying only on linear prediction at $t-1$, which is the case for both the *SLS* and *RAB* methodologies.

3) Continuous Trajectory Coverage: We now derive the bound on the average number of tracking principals required to cover an arc of a given target's trajectory. To help with developing the intuition behind the proposed approach, we use the settings described in Fig. 6, which shows a circle centered at the point o with radius r and its arc segment $\widehat{AB}$ intersecting with the covering disk of a sensor node S_i at intersections A and B.

Let ℓ denote the length of the chord $\overline{AB}$, and assume that ℓ is uniformly distributed in the range $[0, 2R_s]$, where R_s is the sensing range of S_i. Then, angle ϕ ($\angle AoB$ in Fig. 6, measured in radians) and the length of $\widehat{AB}$, e.g.,$\mathcal{L}$, can be expressed as $2 \cdot \arcsin(\ell/2r)$ and $2 \cdot r \cdot \arcsin(\ell/2r)$, respectively. The distribution of $\mathcal{L}$ can therefore be calculated as

$$F(\mathcal{L}) = \int_0^{2r\cdot\sin\left(\frac{\mathcal{L}}{2r}\right)} f(\ell)d_\ell = \frac{r}{R_s} \cdot \sin\left(\frac{\mathcal{L}}{2r}\right) \tag{10}$$

where $f(\ell)$ is the pdf of ℓ, and $f(\ell) = 1/2R_s$, when $0 \leq \ell \leq 2R_s$. Then, the pdf of $\mathcal{L}$, e.g., $f(\mathcal{L})$, can be obtained by differentiating $F(\mathcal{L})$ with respect to $\mathcal{L}$ as follows:

$$f(\mathcal{L}) = F'(\mathcal{L}) = \begin{cases} \frac{1}{2R_s} \cdot \cos\left(\frac{\mathcal{L}}{2r}\right) & 0 \leq \mathcal{L} \leq \pi R_s \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

Therefore, the expectation of the length of the arc segment $\widehat{AB}$ being inside the sensing disk is given by

$$\begin{aligned} E(\mathcal{L})) &= \int f(\mathcal{L}) \cdot \mathcal{L} d_{\mathcal{L}} \\ &= \int_0^{2R_s} 2r \cdot \arcsin\left(\frac{\ell}{2r}\right) \cdot \frac{1}{2R_s} d_\ell \\ &= \frac{r}{R_s}\left[2R_s \arcsin\left(\frac{Rs}{r}\right) + 2\sqrt{r^2 - R_s^2} - 2r\right]. \end{aligned} \tag{12}$$

We now derive the expression for the average length of an arc covered by a sensor node using a method analogous to the one presented in [8] for line segments. Clearly, the mean number of sensor nodes that are needed to cover an arc curve of length L is $\lceil (L/E(\mathcal{L}))) \rceil$, which serves as a lower bound on the required number of tracking principals. On the other hand, the upper bound, derived in [19] on the expected intersections between a straight line and the boundaries of covering disks, is still valid for an arc trajectory. Thus, the average number of nodes N_a required to cover an arc segment of length L satisfies

$$\left\lceil \frac{L}{E(\mathcal{L}))} \right\rceil \leq N_a \leq \lceil 4\lambda L R_s \rceil \tag{13}$$

where λ denotes the node density, which is assumed to be sufficiently large so that the entire area of interest (the network deployment area) is covered with respect to location detection [20].

4) Maximizing Samples Under the Discrete Sampling Process: Previously, we derived the average length of an arc $E(\hat{L})$ covered by a tracking principal under a *continuous* coverage process. Now, we proceed to deriving a formula for quantifying the number of samples covered by a principal, taking the *discrete* nature of the sampling process into account.

More formally, the current principal P_t should select a sensor node, e.g., S_i $(S_i \in \mathcal{N}(P_t))$, which is located in the area of $\mathcal{D}(P_t, R_s)$, as the next tracking principal. The selected sensor node S_i should satisfy that the segment of the expected trajectory $\widehat{AB}$, intersecting with the disk $\mathcal{D}(S_i, R_s)$ at position L_A and L_B, yields the largest number of sampling points along with it.

Given the expected displacement of the moving target during a sampling interval, e.g., $\tilde{v}I$, and the angular deflection φ_t, radius r (see Fig. 6) is given by

$$r = \frac{\tilde{v}I}{2\sin\left(\frac{\varphi_t}{2}\right)}. \tag{14}$$

Hence, the coordinates of the intersections A and B, which are denoted as L_A and L_B, respectively, are given by solving

the following system of equations:

$$\begin{cases} (x-x_j)^2+(y-y_j)^2=R_s \\ (x-x_o)^2+(y-y_o)^2=r \end{cases} \tag{15}$$

where (x_i, y_i) denotes the center of $\mathcal{D}(S_i, R_s)$, and (x_o, y_o) denotes the center of the arc $\widehat{AB}$, which is computed based on the previous locations L_{t-1} and current location L_t of the target, combined with deflection value φ_t, radius r, and expected displacement $\tilde{v}I$.

Let $\gamma(A, B)$ denote the length of the arc segment $\widehat{AB}$, which is the nominal spatial coverage of node S_i, and let $\gamma(A, L_t)$ denote the length of the arc segment between A and the current object's location L_t, representing the *posterior* blind coverage portion of node S_i. Then, the function $\Psi(S_i)$, which calculates the expected number of samples covered by a candidate sensor node $S_i \in \mathcal{N}(P_t)$, can be specified as

$$\Psi(S_i) = \left\lfloor \frac{\gamma(A,B)-\gamma(A,L_t)}{r\cdot\varphi_t} \right\rfloor \tag{16}$$

where the denominator $r \cdot \varphi_t$ is the length of the expected displacement curve during a given sampling epoch I.

The pseudocode of the algorithm that implements the proposed *DAS* approach to selecting tracking principals is presented in Algorithm 1. Note that Algorithm 1 is a heuristic approach, which recursively considers each node $S_i \in \mathcal{N}(P_t)$, and returns the one that can maximize the number of samples along a corresponding segment of the expected trajectory. Obviously, the time complexity of *DAS* is $O(n)$, where n is the number of neighbor nodes of the current principal, which is determined by the node density.

Algorithm 1 Algorithm of *DAS* Tracking-Principal Selection
Require: $current\ principal\ P_t, I, \mathcal{N}(P_t)$
Ensure: $cand \neq Null$
1:Estimate φ_t with (8)
2:Calculate radius r with (14)
3:Time stamp $T \leftarrow t$;
4:$Cand = \mathcal{N}(P_t).first$ //candidate node
5:**while** $true$ **do**
6: $L_{\text{temp}} \leftarrow Traj(L_T, \tilde{v}I)$ with (9)
7: **for** $each\ S_k \in \mathcal{N}(S_i)$ **do**
8: **if** $\|S_k.L_T, L_{\text{temp}}\| > R_s$ **then**
9: $\mathcal{N}(S_i).remove(S_k)$
10: **else if** $\Psi(S_k) > \Psi(Cand)$ **then**
11: $Cand \leftarrow S_k$
12: **end if**
13: **end for**
14: **if** $\mathcal{N}(S_i) == Null$ **then**
15: **return** $Cand$;
16: **end if**
17: $T \leftarrow T + I$;
18:**end while**

IV. Evaluation With Simulations

We implemented three broad groups of simulations comparing our methodologies with the existing related works. Specifically, we present our observations (and corresponding analysis) categorized as follows.

1) The first group of simulations shows our observations regarding the impact of the various parameters on the number of transfers between successive tracking principals.
2) The second group of simulations addresses the impact of the network configuration parameters on localization and deviation estimations.
3) The third group of simulations pertains to the energy consumption and also discusses the energy-related issues related to precision (i.e., minimizing the localization error) via increasing the number of particles.

The evaluations were conducted on the open-source SIDnet-SWANS simulator for WSNs [21]. The network consisted of 800 homogeneous nodes with simulated ranging capabilities that implement the equivalent of an active ultrasonic echo-ranging system, running on a standard MAC802.15.4 link layer protocol.

We adopted a modified *random waypoint* mobility model [22] for moving target. The target traces are generated to represent four types of moving objects: walk (people), bikes, cars, and fast driving cars, calibrated according to the average speed of the real traces: 4, 10, 25, and 50 mi/h, respectively. The only modification of the mobility model is that, at each time step, the target randomly deviates a value, i.e., φ_t, from its previous moving direction θ_{t-1}, where φ_t is uniformly distributed in $[\varphi_{\min}, \varphi_{\max}]$ (in degrees), e.g., $[0^\circ, 15^\circ]$. For simplicity, we set $\varphi_{\min}$ to zero and evaluate the impact of parameter $\varphi_{\max}$ on the performance of tracking-principal selection algorithms.

Every simulation spanned 3.2 h of simulation time consisting of two parts: 1) bootstrapping and neighbor discovery protocols in SIDnet-SWANS and 2) the actual tracking, in the remaining 2.2 h. The simulation configuration space is summarized in Table II, providing 320 distinct configurations. In addition, we conduct ten random runs for each configuration to evaluate the average performance of different schemes.

A. Reduction of Principal Transfers

We first report the results comparing the *DAS* algorithm with the existing principal transfer algorithms, i.e., *CPS*, *RAB,* and *SLS*, in terms of the number of the transfer of data/tracking responsibilities between principals. We note that we employed the *DRPF* scheme in the *DAS* algorithm, unless otherwise specified. Specifically, we investigate the following impact factors on tracking principals required to cover the target trajectory: 1) the angular deviation $\varphi_{\max}$; 2) the moving target's velocity $\tilde{v}$ during each sampling interval; and 3) the average number of measurements at each sampling epoch, which is determined by the node density λ of the networks.

TABLE II
SIMULATIONS CONFIGURATIONS

Communication/ Sensing Range $[m]$	Effective Sensing Range $[m]$	Principal Selection Algorithms	Node Density(λ) [1-hop Neigbors/node]/ Field Area $[m^2]$	Target Type/ Average Speed [mph]/ Sampling Intervals [s]
R_c = 250 m	45m	*CPS*	$8/1,300^2$	"Walk"/4mph/SI=4-8s
R_s = 100 m	40m	*RAB*	$12/1,150^2$	"Bike"/10mph/SI=1.5-3s/
	35m	*SLS*	$16/1,000^2$	"Car"/25mph/SI=0.8-1.5s
		DAS	$24/850^2$	"FastCar"/50mph/SI=0.4-0.8s

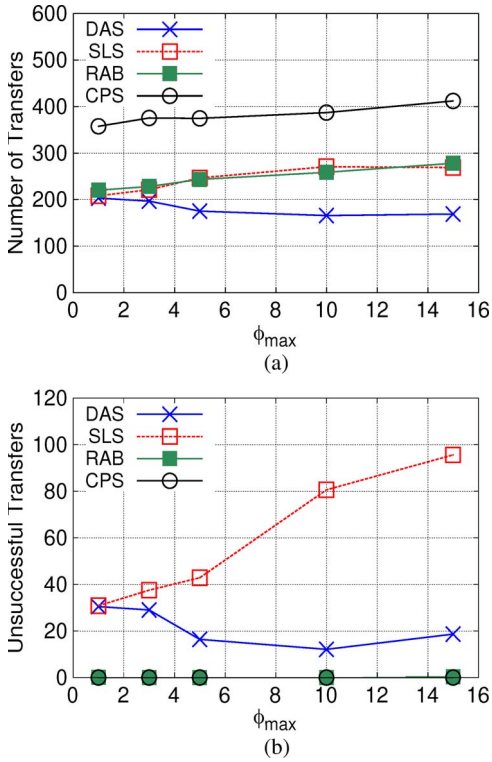


Fig. 7. Impact of angular deviation. (a) Number of transfers. (b) Number of unsuccessful transfers.

1) Impact of the Angular Deviation: Fig. 7(a) and (b) compares the performance of different tracking-principal selection schemes under the impact of parameter $\varphi_{\max}$, increasing from 1° to 15°. The following observations can be established.

- *DAS* achieves improvement by 44%–59% over *CPS*, 8%–39% over *RAB*, and 3%–37% over *SLS* on average. When the angular deviation is small, *DAS* slightly reduces the number of principals compared with *SLS* and *RAB*, due to the fact that there is no significant difference between linear prediction (*SLS* and *RAB*) and the real trajectory. However, as $\varphi_{\max}$ increases, the advantage of *DAS* becomes obvious due to incorporating the angular deviation of the target into trajectory prediction.
- *CPS* is, in a sense, a "risk-free" principal transferring, owing to its conservative sensor selection approach, and *RAB*, which returns the *closest* sensor if there is no node residing in the *relay area* in our experiment. However, *SLS* and *DAS* may produce unsuccessful principal transfers because the selected node may *not* cover the mobile target after handing off the tracking information. Therefore, similar to *SLS*, *DAS* is an "adventurous" tracking-principal selection scheme, which means there exist unsuccessful leader transfers due to inaccurate trajectory prediction. However, as shown in Fig. 7(b), *DAS* can largely reduce the number of unsuccessful principal transfers by accounting for the angular deflection of the mobile target, particularly for larger value of $\varphi_{\max}$.
- The performances of the *SLS* and *RAB* approaches are very similar, which is, to some extent, different from the results reported in [9]. This is because the advantage of *SLS* over *RAB*, through increasing the *posterior blind* coverage, is partially compensated by the "randomly" selection of nodes in the expanded *relay area* of *RAB*. The selected nodes in *RAB* are indeed *sometimes* coincided to or close to the optimal node as chosen by *DAS*, although this kind of selection is *unconscious* regarding the deviation of the moving direction.

We note that due to the localization estimated error, there exists biased angular deviation and generating inaccuracy in terms of the moving object's actual location. This, in turn, is likely to impact the optimal selection of the principals with respect to DAS. The impact of the errors due to estimating angular deviations is analyzed in Section IV-B; however, the complete treatment (qualitative and quantitative) of the impact of the uncertainty due to location errors on the quality of selecting tracking principals is left for our future work.

2) Impact of the Target's Speed: Another factor that affects the performance is the speed of the mobile target. The results, which are shown in Fig. 8(a) and (b), demonstrate that *DAS* reduces 52%–52.5%, 23%–26%, and 22%–24% of handoff size compared with *CPS*, *RAB*, and *SLS*, respectively. While frequent sampling is ideal for accurate localization and robust tracking, desirable sampling intervals for the WSN should be large for the purpose of energy efficiency. To balance the energy saving and quality of tracking, the sampling intervals for different vehicle types vary from $0.4s$ to $8s$, which are empirically chosen.

Intuitively, the faster the mobile target, the more tracking principals are required to cover the moving trajectory if the time is fixed. As shown in Fig. 8(a), the number of tracking principals required, in all schemes, are almost proportional to the target speed. On the other hand, without taking into consideration of the direction deviation, a fast moving object further increases the uncertainty of the prediction and unreliable principal selection, for both *SLS* and *RAB*.

Fig. 8(b) shows the unsuccessful transfers of *SLS* and *DAS*. It demonstrates that *DAS* can keep the unsuccessful transfers in an acceptable level compared with *SLS*.

3) Impact of the Node Density: Next, we present the evaluation of the impact of node density, which we obtained by varying the size of the area of the sensing field, while keeping

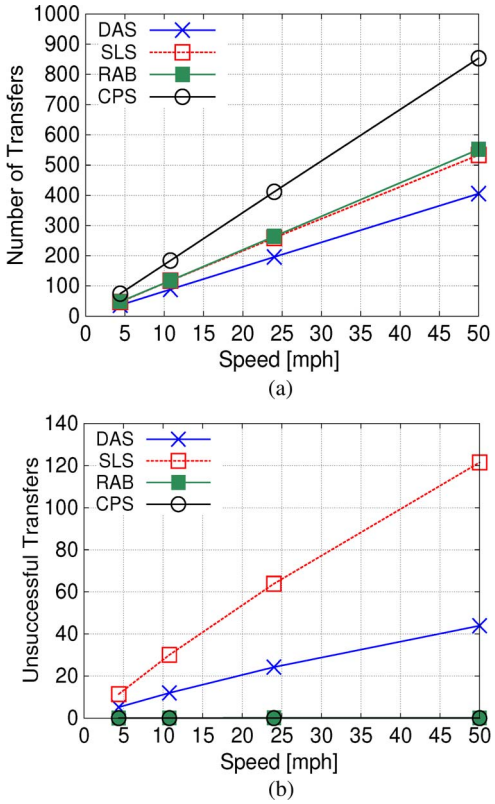


Fig. 8. Impact of target speed. (a) Number of transfers. (b) Number of unsuccessful transfers.

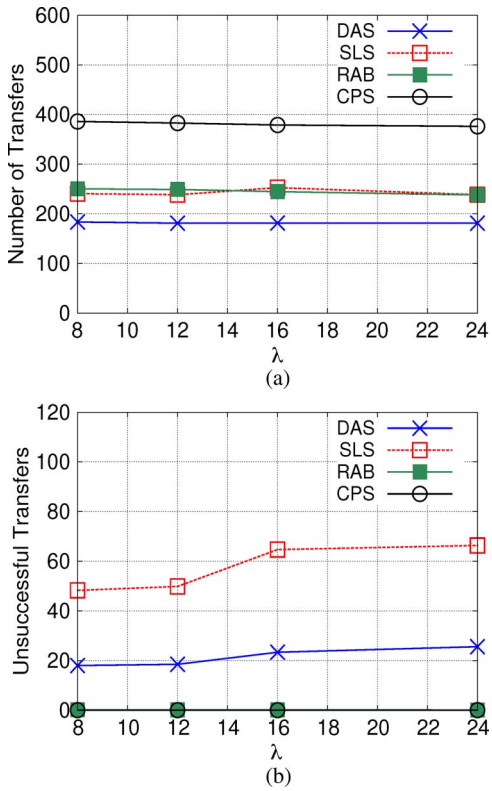


Fig. 9. Impact of node density. (a) Number of transfers. (b) Number of unsuccessful transfers.

the same number of deployed nodes. Node density impacts the performance of tracking-principal selection schemes in two ways. First, the network density has a great impact on the accuracy of location estimation for all schemes. In addition, for *DAS*, the higher the node density is, the more accurate particle-filtering-based localization can be, as can the angular deflection estimation. In addition, from the perspective of trajectory coverage, high node density may provide more qualified nodes that can assume the role of the next tracking principal. However, increasing the node density can largely increase the amount of energy expenditure on sensing, data aggregation, and, particularly, the in-network communication. The energy dissipation issues will be discussed in Section IV-C.

As shown in Fig. 9(a), the *DAS* algorithm still performs the best under all node density configurations. The mean performance gain of *DAS* over *CPS*, *RAB*, and *SLS* is 53%, 26%, and 24%, respectively. Another important observation is that, as shown in Fig. 9(a), a few neighbors are sufficient to reach stable performance for various principal selection schemes.

Fig. 9(b) shows the comparison between *DAS* and *SLS* on unsuccessful transfers, which show the more accurate trajectory prediction of *DAS* compared with *SLS*. An interesting observation is that there are slightly more failed transfers in the higher node density network, which seems counterintuitive. The reason is that there are more nodes distributed in the margin region of the communication area of the current principal in a higher density network, which may cover more trajectory according to (16), and thus are preferable choices. However, these "desirable" nodes also increase the possibility of unsuccessful handoffs because of the presence of prediction errors.

4) General Performance: Fig. 10(a) and (b) shows the general performance of different schemes on the number of principal transfers (successful and failed, respectively) evolving along with the time axis, which is plotted by averaging all runs. As it shows, *DAS* achieves strong principals reduction, on average, compared with other schemes, while keeping the unsuccessful transfers in an acceptable level, which validate the advantage of the proposed algorithm in different scenarios of network configurations.

In conclusion, the simulation results demonstrate that our deflection-aware tracking-principal selection algorithm significantly outperforms the previous schemes under a wide range of conditions.

B. On Localization Accuracy

Since the *DAS* algorithm relies on the particle-filtering-based localization and direction estimation, the next results concern the performance of the localization error and the deviation estimation error, which are denoted as L_e and φ_e, respectively, under the impact of different network configuration parameters.

The localization error L_e is measured as the mean absolute localization error, which is scaled as the percentage of the sensing range $L_e = (1/T)\sum_{t=1}^{T} \|\hat{L}_t, L_t\|/R_s$, where $\|\cdot\|$ denotes the Euclidean distance between the estimated location $\hat{L}_t$ and the real location L_t at time step t, and the results are averaged with the total sampling epochs T. Similarly, the direction deviation estimated error φ_e is calculated as follows: $\varphi_t = (1/T)\sum_{t=1}^{T} |\hat{\varphi}_t, \varphi_t|/\varphi_t$, where $|\cdot|$ denotes the absolute error between the estimated deviation $\hat{\varphi}_t$ and the actual

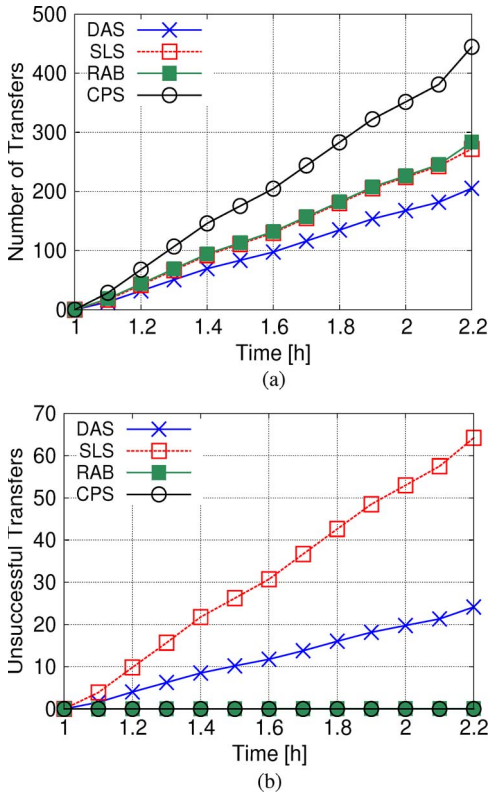


Fig. 10. Comparison of general performance. (a) Number of transfers. (b) Number of unsuccessful transfers.

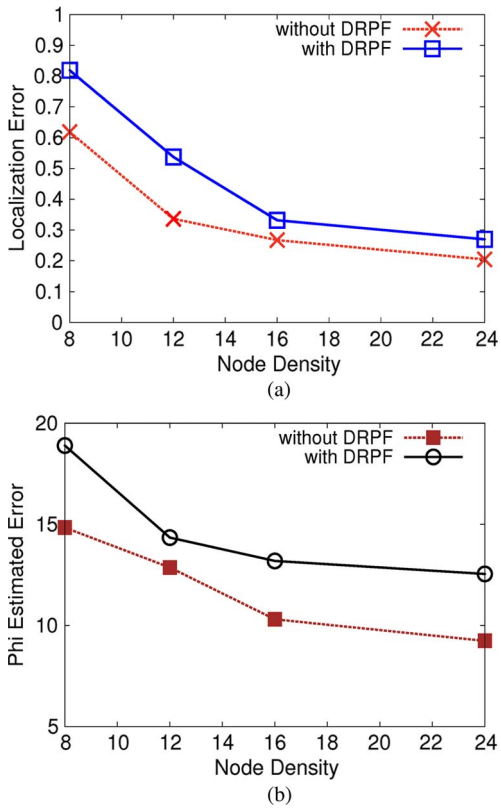


Fig. 11. Impact of λ on estimation error. (a) λ versus L_e. (b) λ versus φ_e.

deviation φ_t. As discussed in Section II, this paper uses the *DRPF* scheme to adapt the sampling process of particle filtering. We compare the state estimation error of the *DRPF* scheme to the original particle filtering calculation, to evaluate the introduced error by *DRPF*. The energy consumption comparison will be held in Section IV-C.

Fig. 11(a) and (b) shows the impact of node density λ on the target-state estimation while evaluating the performance of *DRPF* scheme. Apparently, the accuracy of state estimation, including localization and direction deflection, improves by increasing node density. In some sense, the node density and the particle number affect the accuracy of state estimation in the same way, i.e., through increasing the number of efficient samples and thus approaching the real distribution of state space. On the other hand, it shows that applying the *DRPF* scheme may bring a small estimation error, although it is insignificant in WSNs with a higher node density.

Fig. 12(a) and (b) shows the impact of the number of particles N_p on L_e and φ_e, respectively. As indicated, the expected conclusion is that the more particles there are, the more accurate localization and deviation estimation will be. Note that the *DRPF* scheme slightly reduces the estimation accuracy. However, on the flip side, one can observe that while more particles obtain higher localization accuracy, they incur additional computational overheads and increase the communication levels, thereby demanding more energy. While the problem of striking a good balance between the two is beyond the scope of this paper, we note that [as can be observed in Fig. 12(a) and (b)] using 100 particles may achieve acceptable estimation results.

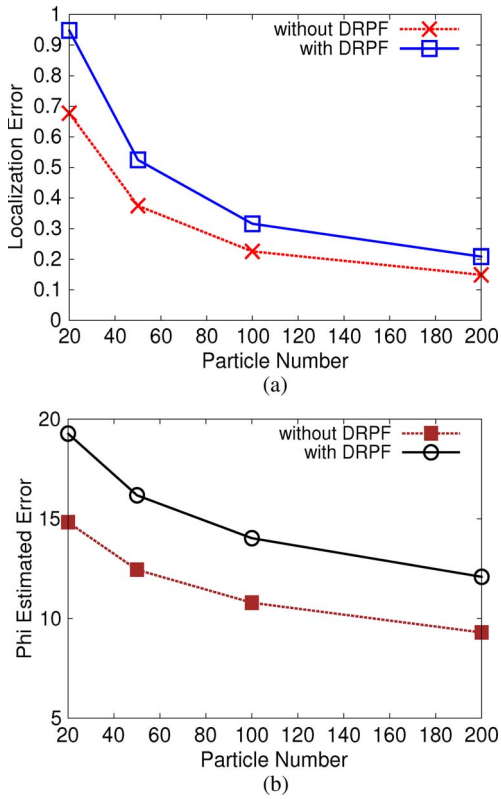


Fig. 12. Impact of N_p on estimation error. (a) N_p versus L_e. (b) N_p versus φ_e.

As shown in Fig. 13, both L_e and φ_e are not too sensitive on the direction deflection $\varphi_{\max}$ due to the expanded sampling area through increasing the values of $\varphi_{\max}$ (cf. Fig. 2). We observe, however, that the deviation estimate significantly

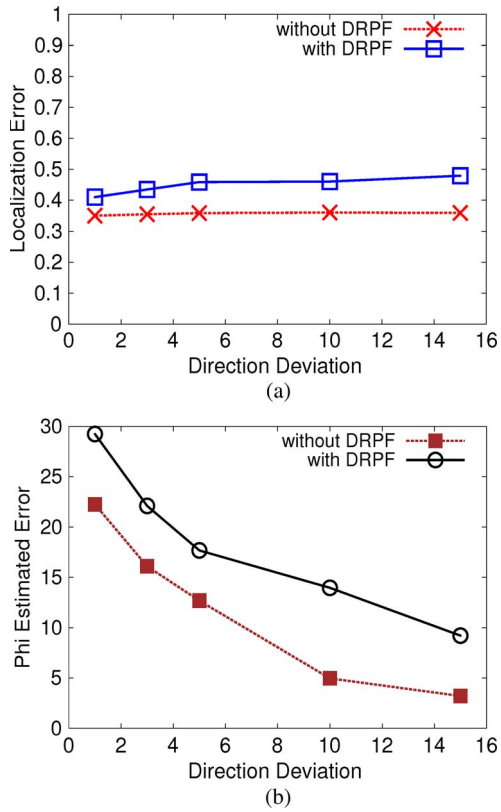


Fig. 13. Impact of $\varphi_{\max}$ on estimation error. (a) $\varphi_{\max}$ versus L_e. (b) $\varphi_{\max}$ versus φ_e.

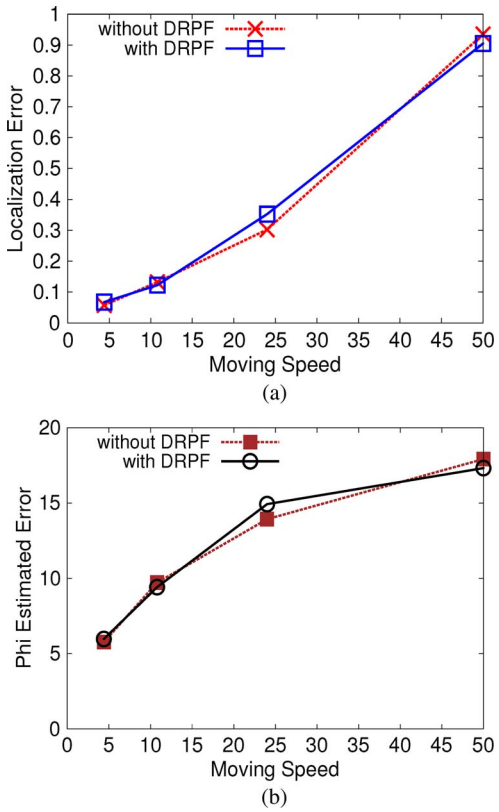


Fig. 14. Impact of $\tilde{v}$ on estimation error. (a) $\tilde{v}$ versus L_e. (b) $\tilde{v}$ versus φ_e.

improves as $\varphi_{\max}$ increases, as shown in Fig. 13(b), which demonstrates that *DAS* performs better with a larger value of $\varphi_{\max}$ (cf. discussion accompanying Fig. 7).

Fig. 14(a) and (b) shows the impact of $\tilde{v}$ on L_e and φ_e, respectively. As can be observed, the performance of *DRPF* is rather insensitive to the parameter $\tilde{v}$, which happens because the sampling period δ tolerated by *DRPF* is inversely proportional to the target's velocity $\tilde{v}$. As explained before, fast moving objects increase the uncertainty of the target's position which, in turn, leads to poorer localization and angular estimation. Hence, from the perspective of the precision of the tracking process, a larger value of particle number and/or more frequent sampling are preferred for a fast moving target.

C. Energy Consumption

The final group of simulation observations that we report addresses the in-network energy consumption aspects.

The nodes are configured to meet the Mica2 Mote energy consumption specifications outlined in Table III. The calculation of the communication overhead of each node is performed using

$$E_c = E_{\mathrm{REQ}} + E_{\mathrm{MES}} + E_{\mathrm{PT}}$$

where we have the following.

1) E_{REQ} denotes the energy expenditure of requesting measurement sent by the tracking principal to its one-hop neighbors.
2) E_{MES} denotes the energy expenditure due to transmitting the measured data.

TABLE III
ENERGY SPECIFICATIONS OF MICA2 MOTE

Node State	Current Draw (mA)	Energy Consumption ($\mu J/bit$)
Sensing Active	10	E_{SA} = 0.781
Sensing Passive	0	E_{SP} = 0
CPU Active	8	E_{CA} = 0.625
CPU Idling	0.015	E_{CI} = 0.001
Radio Transmitting	27	E_{RT} = 2.109
Radio Receiving	10	E_{RR} = 0.781
Radio Listening	3	E_{RL} = 0.234
Radio Off-Mode	0.5	E_{RO} = 0.391
Data Rate(DR): 38.4 kbps		

3) E_{PT} denotes the energy expenditure due to transferring tracking data between consecutive principals.

Specifically, the earlier three components are estimated as

$$\begin{cases} E_{\mathrm{REQ}} = \sum_{t=1}^{T} M_{\mathrm{req}} \cdot \mathcal{N}_t \cdot (E_{\mathrm{RT}} + E_{\mathrm{RR}}) \\ E_{\mathrm{MES}} = \sum_{t=1}^{T} M_{\mathrm{mes}} \cdot N_t \cdot (E_{\mathrm{RT}} + E_{\mathrm{RR}}) \\ E_{\mathrm{PT}} = N_{\mathrm{principal}} \cdot M_{\mathrm{pl}} \cdot (E_{\mathrm{RT}} + E_{\mathrm{RR}}) \end{cases} \quad (17)$$

where M_{req}, M_{mes}, and M_{pl} denote the message (data) size of the request, measurement, and tracking data, respectively, which are set to 24 bits, 128 bits, and 1 KB. $\mathcal{N}_t$ indicates the number of one-hop neighbors, and N_t indicates the number of nodes giving the measurements at each time step. The energy required for aggregating the measurement data and estimating weight of each particle in the principal node is set to 5nj/bit (cf. [23]).

Fig. 15(a) and (b) shows the impact of the number of particles N_p and the node density λ on the overall energy consumption of the *DAS* approach. The results show that increasing N_p or

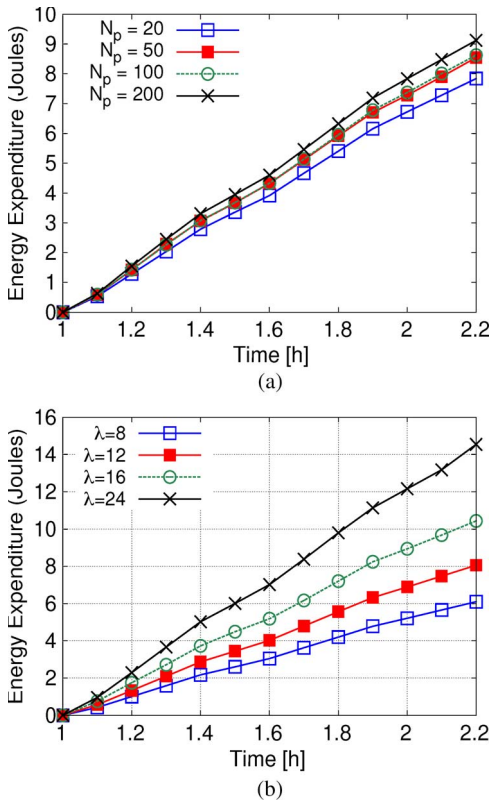


Fig. 15. Energy consumption of *DAS*. (a) Impact of N_p. (b) Impact of λ.

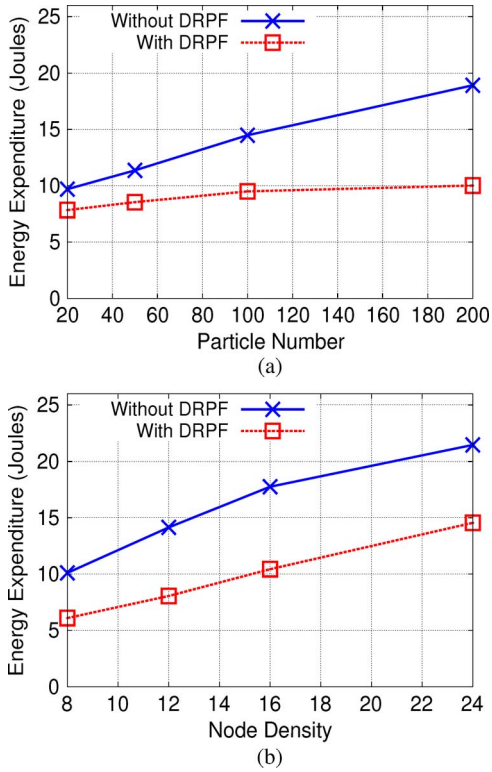


Fig. 16. Energy saving of the *DRPF* scheme. (a) N_p versus E_c. (b) λ versus E_c.

λ also increases the burden of in-network energy dissipation, although it may reduce the localization error, as discussed in Section IV-B. The reason is that *DAS* is a particle-filtering-based localization and an angular deflection estimation methodology, which requires intensive calculation in the principal nodes as λ or N_p increases, although *DAS* achieves fewer transfers of tracking data. There are different ways to conserve energy in high-density networks for *DAS*, one of which is to utilize a threshold parameter ξ to control the number of nodes that can communicate to the principal, e.g., only those nodes detecting the target with signal strength $\geq \xi$ may report the measurements.

Fig. 16 compares the energy consumption of the *DAS* algorithm with the *DRPF* scheme to the conventional particle filter, analyzing the impact of N_p and λ, respectively. As stated earlier, *DRPF* is an approach that is targeting a reduction in the computation cost of the particle filtering method, at the expense of some localization errors. Clearly, significant energy expenditures are saved by employing *DRPF*, particularly in settings in which the values of N_p and λ are larger.

V. Related Work

Research in the field of WSNs has generated a large body of works that have addressed various aspects of the problem of robust and energy-efficient target tracking.

Organizing the nodes in clusters for the purpose of collaborative target tracking has already been considered in the literature [4]–[6], along with some dynamically maintained structures to improve the information fusion, e.g., convoy trees [24]. In addition, various nodes' wake-up strategies and power conservation protocols for target tracking have been investigated [1], [25]–[27]. The cluster formation strategy considered in this paper is similar to that in [4]; however, our approach differs in the aspects of combining the issues of tracking-principal selection and trajectory prediction.

Moving-object trajectory coverage has also been addressed as a problem of interest in the global context of target tracking. Wang *et al.* and Zhang and Hou [13], [14] give the conditions under which a sensor-network deployment region is guaranteed to satisfy both coverage and connectivity. In [1] and [8], the problem of trading off between the energy consumption and the quality of monitoring in WSNs was studied, and analysis was presented for *continuous* target path coverage based on the geometric properties, similar in spirit to that in [19]. However, the works of Gui and Mohapatra [1] and He and Hou [8] only consider the straight line path coverage without concerning the trajectory that is deviating from the expected moving directions, which is one of the features of this paper. In addition, compared with the works in [1] and [8], this paper considered both the *continuous* path and *discrete* segments covered by a sensor node.

The methodologies of indexing and prediction for nonlinear trajectories have been tackled by the researchers from the moving-object database community, e.g., in [28] and [29]. Tao *et al.* [28] proposed the *recursive motion function* and the STP tree for expressing and indexing the nonlinear motion patterns, such as polynomials, ellipses, and sinusoids. Jeung *et al.* [29] studied the problem of discovering trajectory patterns and efficiently answering the predictive query, using a *hybrid prediction algorithm* and an indexing technique called

the *trajectory pattern tree*. Although the idea of prediction and indexing nonlinear motion patterns behind this paper is similar, we focus on addressing the problem of trajectory coverage and efficiency of transmission of the tracking data between successive principals.

Selecting a subset of sensors that will collaborate when executing a certain task is one of the canonical problem in the WSN field. In [30], Zhao *et al.* presented an information driven tracking scheme. Based on [30], several papers such as [31] and [32] have been proposed, addressing the information gain of sensor nodes. The main idea of earlier works are to select the sensor node that can minimize the uncertainty of the moving target's location using mutual information and entropy theory. Compared with the works in [31] and [32], which *predict* the information gain by a sensor node before obtaining the data, our approach focuses on estimating the expected trajectory of the target and, hence, its spatiotemporal coverage by candidate sensor nodes.

The approaches that are most closely related to this paper, in the sense of addressing the problem of reducing the number of tracking information handoffs between principals, have been presented in [8] and [9]. In [8], He and Hou proposed a duty-sensor selection scheme based on the concept of relay areas, which aims at maximizing the *continuous* coverage of the predicted trajectory of a given moving object by a chosen cluster leader. To specifically take into account the *discrete* nature of the sampling process organized in epochs, our previous work [9] addressed the problem of improving the spatiotemporal coverage of the expected moving trajectory through minimizing the *posterior* blind coverage by a given tracking principal. There are two main distinctions of this paper compared with the works of He and Hou [8] Ghica *et al.* [9]. First, when selecting the tracking principals, we consider the angular deflection of the moving object, particularly at the time instant at which the handoff occurs. *Second*, the location "prediction" in this paper is deferred to the time step at which the moving object's location has been updated.

There are also works that have addressed the target or mobile sensor nodes localization, relying upon the Bayesian sequential Monte Carlo methods [32]–[35]. We note that this paper completely relies on the existing particle-filtering-based localization approaches to estimate the moving target, in particular, to estimate the moving angular deflection from the previous moving direction. However, the *DRPF* method proposed in this paper also exploits the movement prediction using the dead-reckoning technique, to balance the position uncertainty regarding the target and the energy consumption of particle filtering estimation. In addition, we focused on improving the accuracy of target trajectory prediction and, most importantly, on reducing the number of tracking principals required for trajectory coverage, by taking into consideration the direction deviation.

Recent work that is similar in spirit, in the sense of tackling the uncertainty of the tracked object's location, but is complementary to our results is presented in [36]. However, while the focus of [36] is on incorporating the imprecision of the location detection in the tracking process, we have addressed the issue of selecting tracking principals' sequence for the purpose of minimizing the energy expenditures.

VI. Conclusion and Future Work

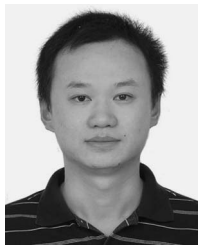
We have addressed the problem of efficiently managing the energy consumption in tracking settings in WSNs, focusing on the aspect of selecting the tracking principals. Specifically, we considered the impact of the deviation of the target's estimated trajectory and efficient adjustments of the principal's selection in response to detecting it. To cater to this dynamics, we have presented a novel tracking-principal selection algorithm, along with a handoff scheme between consecutive principals. The proposed scheme exploits the angular deflection of target and significantly reduces the number of tracking principals required to cover the moving-object trajectory. Both analytical and experimental evaluations, compared with existing tracking-principal approaches, are conducted to verify and validate our scheme in a wide range of scenarios.

As part of our ongoing efforts, we are investigating the issue of load balancing among the participating nodes when there are multiple tracking targets [7] and its impact on the selection of the tracking principal. The instantaneous (location and time) detection in WSNs is typically done via some collaborative trilateration [15], [24], [30], where the distance estimates from the participating nodes are inherently imprecise. While the spatiotemporal data management community has addressed the formalization of the problem of handling uncertainty in various queries [36], [37], incorporating similar treatments when selecting tracking principals is a challenge that we plan to address in the near future, augmenting our recent results [38]. A particular facet of the problem that we plan to address in the future is the incorporation of our proposed scheme into real-time tracking systems similar to VigilNet [39], extending it to the settings in which heterogeneous nodes (static and mobile) are participating in the tracking process.

References

[1] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks," in *Proc. MobiCom*, 2004, pp. 129–143.
[2] J. A. Fuemmeler and V. V. Veeravalli, "Energy efficient multi-object tracking in sensor networks," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3742–3750, Jul. 2010.
[3] S. Pattem, S. Poduri, and B. Krishnamachari, "Energy-quality tradeoffs for target tracking in wireless sensor networks," in *Proc. IPSN*, 2003, pp. 32–46.
[4] H. Yang and B. Sikdar, "A protocol for tracking mobile targets using sensor networks," in *Proc. IEEE Workshop Sens. Netw. Protocols Appl.*, 2003, pp. 71–81.
[5] W.-P. Chen, J. C. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 3, pp. 258–271, Jul.-Aug. 2004.
[6] S. Oh, S. Sastry, and L. Schenato, "A hierarchical multiple-target tracking algorithm for sensor networks," in *Proc. ICRA*, 2005, pp. 2197–2202.
[7] K. Vu, R. Zheng, and Q. Hao, "Multi-target tracking in distributed active sensor networks," in *Proc. MILCOM*, 2010, pp. 1044–1049.
[8] G. He and J. Hou, "Tracking targets with quality in wireless sensor networks," in *Proc. ICNP*, 2005, p. 74.
[9] O. Ghica, G. Trajcevski, F. Zhou, R. Tamassia, and P. Scheuermann, "Selecting tracking principals with epoch awareness," in *Proc. GIS*, 2010, pp. 222–231.
[10] P. Manohar and D. Manjunath, "On the coverage process of a moving point target in a non-uniform dynamic sensor field," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 7, pp. 1245–1255, Sep. 2009.
[11] B. Sundararaman, U. Buy, and A. Kshemkalyani, "Clock synchronization for wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 281–323, 2005.

[12] M. Saxena, P. Gupta, and B. N. Jain, "Experimental analysis of rssi-based location estimation in wireless sensor networks," in *Proc. COMSWARE*, 2008, pp. 503–510.
[13] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proc. SenSys*, 2003, pp. 28–39.
[14] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc Sens. Wireless Netw.*, vol. 1, no. 1/2, pp. 89–124, Mar. 2005.
[15] Y. Zou and K. Chakrabarty, "Advances in target tracking and active surveillance using wireless sensor networks," in *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Network*. New York: Auerbach, 2006.
[16] Z. Chen, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, pp. 1–69, 2003.
[17] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Stat. Comput.*, vol. 10, no. 3, pp. 197–208, Jul. 2000.
[18] J. Liu, R. Chen, and T. Logvinenko, "A theoretical framework for sequential importance sampling and resampling," *Sequential Monte Carlo Methods Pract.*, vol. 13, pp. 225–246, 2001.
[19] P. Hall, *Introduction to the Theory of Coverage Processes*. New York: Wiley, 1988.
[20] S. Shakkottai, R. Srikant, and N. Shroff, "Unreliable sensor grids: Coverage, connectivity and diameter," in *Proc. INFOCOM*, 2003, vol. 2, pp. 1073–1083.
[21] O. Ghica, G. Trajcevski, P. Scheuermann, Z. Bischof, and N. Valtchanov, "Sidnet-swans: A simulator and integrated development platform for sensor networks applications," in *Proc. SenSys*, 2008, pp. 385–386.
[22] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Commun. Mobile Comput.*, vol. 2, no. 5, pp. 483–502, 2002.
[23] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.
[24] W. Zhang and G. Cao, "DCTC: Dynamic convoy tree-based collaboration for target tracking in sensor networks," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1689–1701, Sep. 2004.
[25] Y. Zou and K. Chakrabarty, "Target localization based on energy considerations in distributed sensor networks," *Ad Hoc Netw.*, vol. 1, no. 2/3, pp. 261–272, 2003.
[26] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed, "Detection, classification and tracking of targets in distributed sensor networks," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 17–29, Mar. 2002.
[27] R. Brooks, P. Ramanathan, and A. Sayeed, "Distributed target classification and tracking in sensor networks," *Proc. IEEE*, vol. 91, no. 8, pp. 1163–1171, Aug. 2003.
[28] Y. Tao, C. Faloutsos, D. Papadias, and B. Liu, "Prediction and indexing of moving objects with unknown motion patterns," in *Proc. SIGMOD*, 2004, pp. 611–622.
[29] H. Jeung, Q. Liu, H. T. Shen, and X. Zhou, "A hybrid prediction model for moving objects," in *Proc. ICDE*, 2008, pp. 70–79.
[30] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration for tracking applications," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 61–72, Mar. 2002.
[31] H. Wang, K. Yao, G. Pottie, and D. Estrin, "Entropy-based sensor selection heuristic for target localization," in *Proc. IPSN*, 2004, pp. 36–45.
[32] D. Guo and X. Wang, "Dynamic sensor collaboration via sequential Monte Carlo," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 6, pp. 1037–1047, Aug. 2004.
[33] C. Hue, J. P. Le Cadre, and P. Prez, "Sequential Monte Carlo methods for multiple target tracking and data fusion," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 309–325, Feb. 2002.
[34] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proc. MobiCom*, 2004, pp. 45–57.
[35] M. Rudafshani and S. Datta, "Localization in wireless sensor networks," in *Proc. IPSN*, 2007, pp. 51–60.
[36] Q. Ren, J. Li, and S. Cheng, "Target tracking under uncertainty in wireless sensor networks," in *Proc. IEEE MASS*, 2011, pp. 430–439.
[37] G. Trajcevski, O. Wolfson, K. Hinrichs, and S. Chamberlain, "Managing uncertainty in moving objects databases," *ACM Trans. Database Syst.*, vol. 29, no. 3, pp. 463–507, Sep. 2004.
[38] F. Zhou, G. Trajcevski, and B. Avci, "Tracking coverage throughout epochs with bounded uncertainty," in *Proc. NCA*, 2011, pp. 67–74.
[39] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, and B. Krogh, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM Trans. Sens. Netw.*, vol. 2, no. 1, pp. 1–38, 2006.



Fan Zhou (S'11) received the B.S. degree in computer science from Sichuan University, Sichuan, China, in 2003 and the M.S. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2006. He is currently working toward the Ph.D. degree with the University of Electronic Science and Technology of China.

He is currently working with the DBSN Laboratory, Northwestern University, Evanston, IL, as a Predoctoral Visiting Scholar. His research interests include mobile computing and wireless sensor networks.

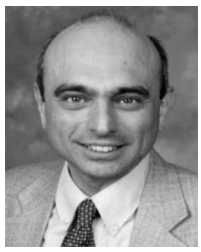
Goce Trajcevski (M'11) received the B.Sc. degree from Ss. Cyril and Methodius of the University of Skopje, Skopje, Macedonia, and the M.S. and Ph.D. degrees from the University of Illinois at Chicago.

He is currently an Assistant Chair with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL. His research has been funded by BEA, Northrop Grumman Corporation, and the National Science Foundation. He is the author or coauthor of over 70 publications in refereed conferences and journals, a book chapter, and three encyclopedia chapters. His main research interests include the areas of spatiotemporal data management, and uncertainty and reactive behavior management, in both moving-object databases and wireless-sensor-network settings.

Dr. Trajcevski was part of the organizing committees of the Association for Computing Machinery (ACM) GIS 2011, ACM SIGMOD 2006, IEEE MDM 2011, and IEEE MDM 2012 and has served on the program committees of numerous conferences and workshops. He received two Best Paper Awards from CoopIS 2000 and MDM 2010, respectively.

Oliviu Ghica received the B.Sc. degree in computer and electrical engineering from the University Politehnica of Bucharest, Bucharest, Romania, in 2002 and the M.Sc. and Ph.D. degrees in electrical engineering and computer sciences from Northwestern University, Evanston, IL, in 2006 and 2011, respectively.

He is currently an active player in open-source communities focusing on simulation methodologies for sensor-network applications. His research interests include cover spatiotemporal data management and routing in large-scale wireless sensor networks with a particular interest in energy efficiency and lifetime benefits of algorithmic implementations.



Roberto Tamassia (M'00–SM'08–F'09) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign in 1988.

He is the Plastech Professor of computer science and the Chair of the Department of Computer Science, Brown University, Providence, RI. He is also the Director with Brown's Center for Geometric Computing. He is the author of six textbooks and more than 240 research articles and books and has given more than 70 invited lectures worldwide. His research has been funded by the Army Research Office, the Defense Advanced Research Projects Agency, the North Atlantic Treaty Organization, the National Science Foundation, and several industrial sponsors. His research interests include information security, cryptography, analysis, design, and implementation of algorithms; graph drawing; and computational geometry.

Dr. Tamassia serves regularly on program committees of international conferences. He is listed among the 360 most cited computer science authors by Thomson Scientific, Institute for Scientific Information. He received a Technical Achievement Award from the IEEE Computer Society for pioneering the field of graph drawing.

Peter Scheuermann (LF'12) is currently a Professor with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL. He has held Visiting Professor positions with the Free University of Amsterdam, Amsterdam, The Netherlands; the University of Hamburg, Hamburg, Germany; the Technical University of Berlin, Berlin, Germany; and the Swiss Federal Institute of Technology, Zurich, Switzerland. During 1997–1998, he served as a Program Director for operating systems with the National Science Foundation (NSF). He is the author of more than 120 journal and conference papers. His research has been funded by the NSF, the National Aeronautics and Space Administration, Hewlett-Packard, Northrop Grumman, and BEA, among others. His research interests include distributed database systems, mobile computing, sensor networks, and data mining.

Dr. Scheuermann is a Fellow of the American Association for the Advancement of Science. He has served on the editorial board of the *Communications of the ACM*, *The VLDB Journal*, and the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING and is currently an Associate Editor of *Data and Knowledge Engineering*.

Ashfaq Khokhar (F'09) received the B.S. degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 1985; the M.S. degree in computer engineering from Syracuse University, Syracuse, NY, in 1989; and the Ph.D. degree in computer engineering from the University of Southern California, Los Angeles, in 1993.

After his Ph.D. years, he spent two years as a Visiting Assistant Professor with the Department of Computer Sciences and School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN. In 1995, he joined the Department of Electrical and Computer Engineering, University of Delaware, Newark, where he first served as an Assistant Professor and then as an Associate Professor. Since 2000, he has been with the Department of Computer Science and Department of Electrical and Computer Engineering, University of Illinois at Chicago, where he currently serves as a Professor. He is the author of over 200 technical papers and book chapters in refereed conferences and journals in the areas of wireless networks, multimedia systems, data mining, and high-performance computing.

Dr. Khokhar served as the Program Chair of the 17th Parallel and Distributed Computing Conference in 2004, as the Vice Program Chair for the 33rd International Conference on Parallel Processing in 2004, and as the Program Chair of the IEEE Golbecomm Symposium on Ad hoc and Sensor Networks in 2009. He received the National Science Foundation CAREER award in 1998, and his paper entitled "Scalable S-to-P Broadcasting in Message Passing MPPs" won the Outstanding Paper Award at the International Conference on Parallel Processing in 1996.