Tracking a moving object via a sensor network with a partial information broadcasting scheme

Jianghai Li *, Qing-Shan Jia, Xiaohong Guan, Xi Chen

Center for Intelligent and Networked Systems, Department of Automation, TNLIST, Tsinghua University, Beijing 100084, China

Abstract

Tracking a moving object is one of the key applications of wireless sensor networks (WSNs) for facility management, logistics, healthcare, etc. The quantitative relationship between tracking accuracy and resource consumption is crucial for designing a WSN-based tracking system. A partial information broadcasting scheme (PIBS) is developed as a decentralized tracking strategy in this paper, where only a part of the nodes broadcast their tracking estimation results to their neighbors. The relationship between tracking accuracy and resource consumption is quantified with this scheme. Two broadcasting policies are proposed by balancing the residual energy among sensors and reducing time delay, where the “Randomly Broadcast” (RB) policy takes energy consumption as a first priority and the “Good Estimates Broadcast First” (GEBF) policy consider time delay first. Both of them can be implemented as a decentralized tracking strategy. It is shown that the PIBS with RB or PIBS with GEBF can reduce required resources with minor degradation in tracking accuracy in comparison with the centralized tracking strategy. The finite sensing range of a sensor node is considered and a node activation scheme with variable activation radius is introduced for energy saving. The activation radius is adjusted to guarantee tracking coverage with trade-off between tracking accuracy and resource consumption. The analytical conditions on tracking coverage and the bounds on tracking accuracy are obtained. The numerical experiments demonstrate that this activation scheme outperforms the existing schemes.

1. Introduction

Tracking a moving object is one of the key applications of emerging wireless sensor networks (WSNs) for facility management, logistics, healthcare, etc. [14]. Two challenging issues arise in developing WSN-based tracking systems with regards to traditional tracking systems, e.g. radar systems. The first issue is how to guarantee to cover the target to be tracked by sensor nodes deployed with geographical distribution. The second one is how to trade off between tracking accuracy with the limited resources of WSNs, such as energy, bandwidth, etc.

The essence of the first issue is that a moving object may not remain in the sensing range or coverage of one sensor all the time. In another word, without the collaboration of WSN nodes, an individual sensor cannot capture the moving target all the time or may lose the target during tracking. Furthermore, the target motion disturbance and the measurement noise may cause failure in sensor activation and lead to target loss in tracking. Finding the conditions for tracking coverage is very important in designing a WSN.
The second issue is due to the resource restriction of WSNs on energy at each node, and limited communication bandwidth among nodes [31]. Energy for computing, sensing, and communicating in WSN is generally supplied by batteries [1] and the communication bandwidth is usually limited. There are many studies on communication protocols [28] and data compression algorithms [19] that enable efficient use of energy and bandwidth. Scheduling the working status of sensor nodes intelligently in the application layer can significantly reduce energy consumption and communication load. Time delay caused by re-transmission and multi-hop packet relay is an important criterion on quality of service (QoS) of WSN. Tracking an object in motion particularly requires a low time delay. However this generally needs high resource requirements. The relationship between tracking accuracy and required resources should be quantified to provide guidelines for WSN.

There are two strategies in terms of data processing mechanism: centralized strategy and decentralized strategy. In the centralized strategy, one sensor is artificially selected as a cluster head [33,36]. The tracking estimation is performed at this node with all data sent to it. With this strategy, many good existing target tracking methods can be applied directly. However, one obvious issue of this strategy is task overloading at the cluster head. In many cases, all sensor nodes have identical computing capability and energy, and thus there is no node with asymmetrically strong processing power to collect and process data. The cluster head rotation schemes are often complicated and would bring in overheads and unreliability [37]. In some other cases, convenient access to tracking results is crucial since some sensor nodes in a particular sub-area are required to provide tracking results for the users nearby [15]. One real life example is to seize a thief in a pursuit-evasion game [35]. Consequently, it is desirable to have a decentralized strategy with high scalability, low energy consumption, light bandwidth requirement, and short time delay. This means that each node updates its tracking estimates based on neighboring and local observations without the aid of a central node. The decentralized estimation can avoid multi-hop relays and reduce communication load at the cost of local computation, and is scalable for large scale WSNs. Furthermore, due to the fact that the energy consumed for wireless communication exceeds that for computation in WSNs [38], the decentralized tracking strategy is more energy efficient.

In this paper, we consider a decentralized state estimation and sensor selection problem subject to communication and energy constraints. A partial information broadcasting scheme (PIBS) is developed as a decentralized tracking strategy and the challenging issues of tracking coverage and resource restriction are addressed. With the PIBS, all the nodes take individual measurements and only a part of the nodes broadcast tracking estimation results to their neighbors. Each node assimilates the public broadcasting data for local estimation. The relationship between tracking accuracy and resource consumption is quantified with this scheme. The resource consumption, including the energy consumption, communication load, and total time delay, is positively related to the number of broadcasting nodes. The tracking accuracy is shown to be related to the selection of broadcasting nodes. Therefore the broadcasting of WSN nodes is scheduled in our PIBS.

Designing the PIBS scheme is formulated as a two-layer optimization problem, where the continuous state estimation is solved at the lower level and the sensor selection is solved at the upper level. The two problems are solved separately. Two broadcasting policies are proposed by balancing the residual energy among sensors and reducing time delay, where the “Randomly Broadcast” (RB) policy takes energy consumption as a first priority, whereas the “Good Estimates Broadcast First” (GEBF) policy considers time delay first. Both of them can be implemented as a decentralized strategy. It is shown that the PIBS with RB or PIBS with GEBF can reduce required energy and bandwidth resources with minor degradation in tracking accuracy in comparison with the centralized tracking strategy.

The finite sensing range of a sensor node is considered and a node activation scheme with variable activation radius is introduced for energy saving. The activation radius is adjusted to guarantee tracking coverage with tradeoff between tracking accuracy and energy consumption. The analytical conditions on tracking coverage and the bounds of tracking accuracy are obtained. The experimental results demonstrate that this activation scheme outperforms some existing schemes.

The rest of this paper is organized as follows. Section 2 surveys the related research work. The general formulation of the tracking problem and assumptions are given in Section 3. Section 4 presents the partial information broadcasting scheme, and derives the distributed estimator based on the PIBS. Section 5 introduces the two-layer optimization problem for sensor selection and presents two broadcasting policies. Theoretical analysis on tracking with finite sensing range is given in Section 6. The numerical testing results are presented in Section 7 followed by the concluding remarks.

2. Literature review

The problem considered in this paper is closely related to the problem of state estimation with sensor selection. In essence, the sensor selection problem is equivalent to the Knapsack problem known to be NP-complete [27,34]. Many heuristic approaches have been developed. Zhao et al. [43] introduced an information-driven sensor query (IDSQ) method which selects the most informative sensor to be the new leader. In the same Bayesian framework, Guo et al. [7] employed the sequential Monte Carlo (SMC) methods to tackle the nonlinear models and improved the approximation of entropy-based information utility. Xiao et al. [41] adapted the sampling interval according to the accuracy and energy to improve the IDSQ using uniform sampling intervals. Xu et al. [42] discussed four heuristics for prediction and wakeup mechanisms. The above policies are greedy in the sense that they reduced the estimation uncertainty by incrementally incorporating additional sensors while not considering the long-term performance.

To improve the above greedy policies, the sensor scheduling problem is formulated as a stochastic optimal control problem. Then the partially observable Markov decision process (POMDP) provides a natural framework to address both
short-term and long-term performances of these control problems. He et al. [9] regarded the state as a combination of target state and sensor on–off state, developed a Monte Carlo method, and provided an approximate optimal policy. However, in order to reduce the action space, they restricted the discussion to that only one sensor can be selected at each time instant. Thus their analysis cannot be applied in situations when multiple sensors can be selected in the same time. Fuummeler and Veeravalli [5] formulated the problem as a POMDP in another way that the state contains the target position and residual sleep time of sensors, and the action is to control the sleep time of each sensor. In order to reduce the state space, the region of interest is divided into cells and the target is restricted to take on one possible location per cell. Williams et al. [40] formulated the problem as a constraint MDP in which the leader sensor and the activated sensor set can be controlled, and developed a dynamic programming approach over a rolling time horizon. These methods, however, can be hardly applied in large-scale practical problems due to the large state and action spaces.

To avoid computational complexity and focus on sensor scheduling, the sequential Bayesian filtering was simplified to Kalman filtering under the assumptions of linear models and Gaussian noises. Kaplan [13] proposed a Kalman-based sensor selection to select the best subset of active nodes out of sensing nodes. However, it did not balance the usage of different sensors thus some sensors may be worn out earlier than the others. Gupta et al. [8] showed a stochastic sensor selection strategy can help to bring down the cost when the observation matrices are incomparable. Shi et al. [29,30] provided a systematic analysis of the tradeoff between the estimation quality and the communication and computation capacity in the background of networked control system [10].

The decentralized implementation of the sensor scheduling algorithm was seldom discussed in the above work. The centralized approach, which often requires a powerful head node and multi-hop data transmission from sensors to the head node, consumes more energy and bandwidth, causes longer time delay, and potentially reduces the survivability of the system when the leader fails [2]. Decentralized estimation is related to multi-sensor data fusion [26] and distributed processing [11]. In WSN, the new challenge is how to reduce the communication load among the nodes while maintaining the required estimation performance.

3. Problem formulation

Tracking a moving target with a wireless sensor network is described by two models. One is the target motion model and the other is the sensor measurement model. The scenario is shown in Fig. 1. For simplicity, a linear model of the target dynamic is considered, i.e.,

$$x(k+1) = Fx(k) + w(k),$$

where \(x(k)\) is the state vector representing the position and the velocity. The state transition matrix \(F\) is known to all the sensors and \(w(k)\) is the motion disturbance. All the mathematical notations are listed in Table 1.

There are \(m\) sensors participating in the tracking task, which means that the moving object is in the union of the sensing ranges of the \(m\) sensors. Assume that each of the \(m\) sensors can detect the moving object and take the measurement of its
position. The sensors here can be represented by positioning sensors such as cameras [4,23], or a logical combination of several bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24]. The set consisting of these general bearing or ranging sensors, such as Cricket [25] or passive infrared (PIR) sensors [20] or RFID tags [21] by some localization techniques [16,24].
Assumption 3. All the sensors in the participating set can hear the broadcasting of other nodes in the same set.

All the \( m(k) \) sensors covering the target are no more than \( 2r_s \) apart, while the communication radius of off-the-shelf sensors are larger than \( 2r_s \). It is reasonable to assume that transmission range of one node covers all nodes within sensing range of target [13]. Besides the packet loss in the communication channel is not considered.

Assumption 4. All the sensors are synchronized.

The WSN is assumed to be time-driven, which means all the sensors in the network have the same clock to schedule the communication. This is a reasonable assumption in the tracking scenario. The time interval of the application layer is usually very large compared with that of the network layer [32].

4. Partial information broadcasting scheme

The partial information broadcasting scheme (PIBS) is developed in this section with consideration of the relationship between the tracking accuracy and the number of broadcasting nodes. The energy consumption, communication load, and total time delay of the WSN are positively correlated with the number of broadcasting nodes. These relationships are characterized by the specified energy models and communication protocols. Reducing the broadcasting nodes can decrease the consumption of resources, but may result in tracking accuracy degradation. The PIBS provides a framework to establish this quantitative trade-off relationship.

In the decentralized estimation scheme, every node calculates an estimate of the moving target without any central node support. The PIBS is executed in three steps. First, all the sensors begin to detect the moving object, take measurements, and do the pre-processing. Next, a part of sensor nodes broadcast the packets containing local estimates (or other data) one by one to avoid collisions and others keep silent. Every sensor can receive the broadcasting data packets from the others under Assumption 3. Finally, the sensor nodes assimilate the data of the received packets into their local estimates. Ultimately, all the sensors have an assimilated estimate of the target’s position. The procedure is listed in Table 2.

The PIBS realizes the idea of partial information of measurements in the previous section, by selecting/suppressing nodes to broadcast their data.

The key to the PIBS is that not all the sensors broadcast at each time step. An indicator vector \( \mathbf{i}(k) \) is introduced to represent the indices of the broadcasting sensors at time \( k \).

\[
\mathbf{i}(k) = [i_j(k)]_{j=1}^{m(k)},
\]

\[
i_j(k) = \begin{cases} 
1 & \text{if sensor } j \text{ broadcasts at time } k, \\
0 & \text{if sensor } j \text{ does not broadcast at time } k.
\end{cases}
\]

4.1. Estimation based on PIBS

The estimation problem with PIBS is formulated as follows. Given a certain partial information broadcasting policy which depicts the constraints on the number of nodes, i.e., \( \mathbf{i}(k) = i(k), k = 1, 2, \ldots, T \), we like to know the optimal estimate for every node i.e., \( \hat{x}_j^p(k) = E[x(k) | Z_j^p(1), \ldots, Z_j^p(k)] \) (the superscript \( p \) stands for partial information broadcasting) that minimizes

\[
J_p[\hat{x}_j^p(k)] = E\left[(\hat{x}_j^p(k) - x(k))^T (\hat{x}_j^p(k) - x(k)) | Z_j^p(1), \ldots, Z_j^p(k)\right],
\]

subject to the target dynamics (1) and the observation model (2), where

\[
Z_j^p(k) = Z(k) \odot \mathbf{i}_j^p(k) = \begin{bmatrix} z_1(k) \\
\vdots \\
z_{m(k)}(k) \end{bmatrix} \odot \mathbf{i}_j^p(k)
\]

(3)

Table 2
The procedure of PIBS.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sensing</td>
</tr>
<tr>
<td>2</td>
<td>Broadcasting</td>
</tr>
<tr>
<td>3</td>
<td>Computing</td>
</tr>
</tbody>
</table>
and
\[
\mathbf{i}_{\text{ip}}(k) = \mathbf{i}(k) \lor \left[ \begin{array}{ccc} \ldots & \frac{1}{\text{th}} & \ldots \end{array} \right]^T .
\]

In the following discussion, we refer this problem as \( \mathbf{P1} \).

The operator \( \lor \) in (4) is an OR operator. The modified indicator vector \( \mathbf{i}_{\text{ip}}(k) \) indicates all the broadcasting nodes as well as the \( \text{th} \) node itself, because the \( \text{th} \) node also knows its own measurement. The resize operator \( \ominus \) in (3) is for selecting the entries whose indicators are 1. It is defined as follows
\[
\mathbf{Z}_{\text{ip}}(k) = \left[ \begin{array}{c} \mathbf{z}_{1}(k) \\ \vdots \\ \mathbf{z}_{m(k)}(k) \end{array} \right],
\]
where \( \mathbf{z}_{\text{ip}}(k) \) is the measurement taken by the \( \text{th} \) broadcasting sensor. Here the \( \text{th} \) broadcasting sensor means \( \sum_{j=1}^{\text{ip}} \mathbf{i}_{\text{ip}}(k) = l - 1 \) and \( \mathbf{i}_{\text{ip}}(k) = 1 \). The resulting matrix has \( n(k) \) entries where

\[
n(k) = \sum_{j} \mathbf{i}_{\text{ip}}(k).
\]

4.2. The optimal distributed estimation

The problem \( \mathbf{P1} \) is a set of multiple optimization problems. The optimal estimator of each problem is determined by the following theorem.

**Theorem 1.** Under Assumptions 1–4, the optimal solution to problem \( \mathbf{P1} \) is

\[
\hat{\mathbf{x}}_{\text{ip}}(k) = \mathbf{P}_{\text{ip}}(k) \left[ \begin{array}{c} \mathbf{P}(k|k-1))^{-1} \mathbf{X}(k|k-1) + \sum_{j=1}^{\text{ip}} (H_{j}(k))^{T}(R_{j}(k))^{-1}Z_{j}(k) \end{array} \right],
\]

where

\[
\hat{\mathbf{x}}_{\text{ip}}(k|k-1) = F\hat{\mathbf{x}}_{\text{ip}}(k-1) .
\]

The corresponding covariance of the error is

\[
\mathbf{P}_{\text{ip}}(k) = \mathbf{P}_{\text{ip}}(k|k) = \left[ \begin{array}{c} \mathbf{P}(k|k-1))^{-1} + \sum_{j=1}^{\text{ip}} (H_{j}(k))^{T}(R_{j}(k))^{-1}H_{j}(k) \end{array} \right]^{-1},
\]

where

\[
\mathbf{P}(k|k-1) = F\mathbf{P}_{\text{ip}}(k-1)F^{T} + \mathbf{Q} .
\]

**Proof.** The proof is given in Appendix. \( \square \)

The distributed estimation is derived from the Kalman filter algorithm. Therefore, the optimality of the estimators can be guaranteed, and (5) is the optimal mean-square estimator for every node based on the partial information.

5. Broadcasting policy optimization

**Theorem 1** states the optimal solution with a given broadcasting policy, i.e., \( \mathbf{i}(k) = \mathbf{r}(k) \), \( k = 1, 2, \ldots, T \). However, the broadcasting policy itself can also be optimized. A broadcasting policy is a sequence of rules for selecting broadcasting nodes. Intuitively we see, when all the sensors are selected to broadcast, i.e., all the elements of \( \mathbf{r}(k) \) are 1, the estimation result is the best. In this case, the communication load is also the heaviest. The objective function should be optimized under some constraints on \( \mathbf{r}(k) \). Therefore, the WSN-based tracking problem turns to a two-layer optimization problem, which is
\[
\begin{align*}
\min_{k(k)} & \left\{ \min_{i}(\mathbf{x}^{T}(k) | \mathbf{x}^{T}(k-1), \mathbf{P}_{i}(k-1), \mathbf{i}(k), \mathbf{z}(j), j = 1, \ldots, m(k)), i = 1, 2, \ldots, m(k) \right\}, \\
\text{subject to} & \quad \Gamma(\mathbf{i}(k)) \leq C,
\end{align*}
\]  

(9)

where (10) indicates the general constraints on \( \mathbf{i}(k) \). Note that this is a multi-objective decision problem. The objective regarding each sensor is different from each other, denoted as \( J_{i} \). The solution to the inner optimization problem is given by Theorem 1, whereas the outer optimization relies on the specific form of the constraint (10). For some special cases, we can find the non-inferior (Pareto-optimal) policy of the problem.

There are two typical forms of constraints on the broadcasting policy. One is for balancing the residual energy among sensors. That is, the average broadcasting ratio for every node should not exceed an upper bound according to their energy storage level, i.e.,

\[
E\left(\frac{\sum_{k} i_{j}(k)}{T}\right) \leq \beta_{j}, \quad k = 1, 2, \ldots, T.
\]

(11)

Differentiated obligations can avoid rapid power-off of some sensors. The other constraint is defined by the bandwidth and time-delay requirement. That is the total number of broadcasting nodes at each time step is restricted by an upper bound determined by the limited bandwidth and time delay, i.e.,

\[
\sum_{j} i_{j}(k) \leq c(k).
\]

(12)

In the following, we refer to the problem formulated by (9) and (11) as \( \textbf{P2} \) and the problem formulated by (9) and (12) as \( \textbf{P3} \).

5.1. RB policy based on residual energy

One feasible policy to \( \textbf{P2} \) is the Randomly Broadcast (RB) policy. With the RB policy, every node randomly broadcasts its data with a certain probability. As long as the broadcasting probability \( \alpha_{j} \) of the \( j \)th node is no more than the limit of the average broadcasting ratio \( \beta_{j} \), the constraint is satisfied.

**Proposition 1** (The feasibility of the RB policy). Suppose \( \mathbf{i}(k) = [i_{j}(k)] \) satisfies \( \Pr(i_{j}(k) = 1) = \alpha_{j}, \) and \( \Pr(i_{j}(k) = 0) = 1 - \alpha_{j} \), where \( \alpha_{j} \leq \beta_{j} \). Then \( \mathbf{i}(k) \) is a feasible policy to \( \textbf{P2} \), i.e.,

\[
E\left(\frac{\sum_{k} i_{j}(k)}{T}\right) \leq \beta_{j}, \quad k = 1, 2, \ldots, T.
\]

**Proof.** For the random variable \( i_{j}(k) \) with Bernoulli distribution, its expectation is

\[
E[i_{j}(k)] = \alpha_{j}.
\]

Aggregating \( i_{j}(k) \) over time, we have

\[
\sum_{k} E[i_{j}(k)] = \alpha_{j} T.
\]

(13)

Due to the linearity of the expectation, Eq. (13) can be rewritten as

\[
E\left[\sum_{k} i_{j}(k)\right] = \alpha_{j} T \leq \beta_{j} T.
\]

\( \Box \)

The advantage of the RB policy is that whether to broadcast depends only on the random seed of the sensor itself, without relying on other nodes. The truly decentralized scheme and simplicity makes the RB policy robust and easy to implement.

5.2. GEBF policy based on the estimation quality

The other policy called Good Estimates Broadcast First (GEBF) policy is more suitable for \( \textbf{P3} \). With GEBF, the sensors with good estimates broadcast first. The quality of the estimates is measured by local estimation error covariance \( \mathbf{P}_{i}(k|k-1) \) or observation noise covariance \( \mathbf{R}_{i}(k) \). The smaller \( \mathbf{P}_{i}(k|k-1) \) or \( \mathbf{R}_{i}(k) \) is, the better estimation quality. The GEBF policy outperforms the RB policy when the sensing and estimation quality vary significantly.

We can show that the GEBF policy with \( \mathbf{R}_{i}(k) \) as the quality measure is a Pareto-optimal policy to \( \textbf{P3} \) under certain assumption. In many cases, sensors or sensor combinations have the same localization accuracy in different directions. Based on this, we make the following assumption.

**Assumption 5.** The observation noise co-variances of sensors are isotropic, that is, the co-variances \( \mathbf{R}_{i} \)'s are comparable.
Theorem 2 (The Pareto-optimality of the GEBF policy). Under Assumption 5, suppose the observation co-variances of the broadcasting sensors selected by $i(k)$, i.e., $\{R_j(k), j : i_j(k) = 1, i(k) = [i_j(k)]\}$, are the top-$c(k)$ minimum among all the observation co-variances $\{R_i(k)\}$. Then the $i(k)$ is a Pareto-optimal policy to $P_3$.

**Proof.** See the proof in the Appendix. □

Note that the proof of the Pareto-optimality suggests the GEBF policy is the optimal policy for only the non-broadcasting sensors, not for the broadcasting sensor. Because for any broadcasting sensor, when the $i$th node does not broadcast any more, i.e., $i \in \text{Out}$,

$$
\left(\tilde{P}^c_i(k)\right)^{-1} - \left(P^c_i(k)\right)^{-1} = \sum_{j \in \text{In}} H_j^T R_j^{-1}(k) H > 0,
$$

where the sets of $\text{In}$ and $\text{Out}$ are defined in Appendix. Eq. (14) indicates that for the broadcasting nodes, any other policy that excludes itself will improve its result. Thus, there is no policy that can be optimal for both sets of sensors. The GEBF policy is the most appropriate one that we can adopt. In practice, the observation covariance is usually negatively correlated with the distance between the sensor and target. The intuition behind this theorem suggests that the nodes closest to the target should broadcast their estimations.

The difficulty in implementing the GEBF policy is that coordination between sensors is required and comparison at one central node takes time and energy. Instead of bringing all the data together or exchanging the data with one another, the broadcasting property provides us a simple way to pick the good estimates. We can assign different broadcasting timings for different nodes according to their quality indicators. The data packets of good quality are assigned with short waiting time for broadcasting. The other data packets of poor quality assigned with long waiting time would receive the packets from the other nodes with good quality and suppress their own packets. The approach of quality-waiting mapping is a decentralized implementation.

6. Tracking with finite sensing range

In the previous sections, the limit of the sensing range is not considered. In practice, the sensing range of a sensor is finite [3,22]. If the moving target is out of the sensing range of one sensor, there is no valid measurement obtained by this sensor.

The scenario of tracking with sensors of finite sensing range is shown in Fig. 2. It is assumed that the sensing area is a circle with radius $r_s$, within which the observation noise covariance $R$ is the same, and out of which no valid measurement is obtained. When the moving target (the dot) arrives in the sensing area of a sensor, the sensor takes measurements. The estimated paths of the target are based on the results of local estimation without combining the data from other sensors. The figure shows that Sensor 2 cannot get correct estimates (the up triangle) until the target moves into its sensing range. On the other hand, once the target moves out of its sensing range, Sensor 1 can only estimate the target location (the down triangle) by the motion model. Each sensor with finite sensing range cannot track the whole path of the target without communication thus collaboration is crucial in this case. If there is a virtual central node collecting all the estimates and corresponding error co-variances, it can obtain good estimates of the whole path.

There are three basic phases in decentralized tracking with finite sensing ranges: detection, activation, and estimation. The first phase is a target detection problem and is not the subject discussed in this paper. The last two phases take turns in Table 3. If all the sensors are active and dense enough to satisfy the coverage requirement, the moving target will be

![Fig. 2. A sample path in the scenario of finite sensing range.](image-url)
tracked from the beginning to the end. When some sensors are turned off into a sleep state for energy efficiency, the moving target could be missed due to sensing coverage holes. Therefore, it is required that the union of sensing coverage of the active sensors should cover the target path dynamically along with its movement. A common approach is to activate the sleeping sensors near the predicted target before it arrives. However, due to the prediction error, not all the activated sensors can sense the target. The active sensors that can sense the target are effective sensing nodes. The problem in the second phase is how to adjust the activation area to satisfy the dynamic coverage requirement with as few activated nodes as possible to save energy. In the third phase, the tracking accuracy requirement is taken into account and tracking accuracy is balanced with total energy consumption. When a group of sensors are activated to participate in the tracking task, we should decide how many of them are going to broadcast the measurements. The number of broadcasting nodes is positively correlated both with the tracking accuracy and with the total energy consumption.

In the following subsection, the WSN is assumed to be deployed densely and uniformly. The densely deployment assumption means that there is at least one sensor in any tiny small area. The uniform deployment assumption means that the number of deployed nodes only depends on the size of an area, and the larger an area is, the more sensors are deployed in the area.

6.1. Sensor activation

The goal of the sensor activation is to activate the appropriate sensors for tracking, and to send them the historical estimation data, including the predicted state and the corresponding covariance. In a global view, the activated sensors are grouped dynamically along with the movement of the target. A subset of the activated sensor group can sense the target and take the measurements. This subset is determined by the intersection of the activation area and the sensing area.

The main difficulty in activating the sensors is that the WSN does not know the actual position of the target in advance. Instead, the WSN can only activate sensors located in the area where the target is predicted to be. The prediction error may lead to the activation of wrong sensors. As shown in Fig. 3, the activation area based on the predicted target

---

**Table 3**
The procedure of PIBS with finite sensing ranges.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Activation</td>
<td>One of the activated nodes in the last period broadcasts the predicted target position ( \hat{x}(k) ) and corresponding covariance ( P^c(k) ). The sensors, whose is within the activation area based on ( \hat{x}(k) ) will be activated. Others remain asleep. The ( \hat{x}(k) ) and ( P^c(k) ) are also passed to the activated sensors.</td>
<td>All the activated nodes begin to sense</td>
<td>Some (or all) of the sensing nodes broadcast their data</td>
<td>All the active sensors assimilate the broadcasting data to obtain the final result. Back to the step 1</td>
</tr>
</tbody>
</table>

---

**Fig. 3.** Activation area based on the predicted position does NOT coincide with the sensing area based on the actual position.
position does not coincide with the sensing area based on the actual position, caused by the prediction error which is a random variable determined by the motion and observation models with noises. The sensing radius is an important parameter of a sensor node and usually fixed. The activation radius on the other hand is adjusted to guarantee that the activation area intersects with the sensing area regardless of prediction error without wasting energy. This technique is called Adjusting Activation Radius (AAR). We will focus on the upper and lower bound of the activation radius.

Assume that activation area and sensing area are the circles with radius of $r_a$ and $r_s$. The prediction error at time $k$ is denoted as $e^*(k)$, i.e.,

$$e^*(k) = \hat{d}^*(k) - d(k),$$

where $d(k)$, $\hat{d}^*(k)$ are the actual and predicted position vectors of the target. So $|e^*(k)|$ represents the distance of prediction error.

(a) The lower bound of $r_a$

$$r_a > |e^*(k)| - r_s.$$  \hspace{1cm} (15)

This lower bound condition guarantees that the WSN is able to track the target. If this requirement is not satisfied, as illustrated in Fig. 4(a), the activation area and the sensing area are totally disjointed. These two areas have no intersection. None of the sensors in the sensing area are activated and all the activated sensors are out of the sensing area. There is a special case in this situation where the prediction error is small enough, i.e., $|e^*(k)| < r_a$, as illustrated in Fig. 4(b). This implies that the predicted position is within the sensing area. Any positive activation radius $r_a$ can satisfy (15) in this case.

(b) The upper bound of $r_a$

$$r_a < |e^*(k)| + r_s.$$  \hspace{1cm} (16)

This upper bound condition prevents the WSN from activating more nodes which cannot detect the target. If the upper bound is exceeded, as illustrated in Fig. 4(c), the activation area totally covers the sensing area. All the sensors that can detect the target are activated and it is not necessary to increase the activation radius $r_a$, because any more sensors involved are out of the sensing area and cannot take valid measurements.

From the discussion above, we can divide the range of the activation radius into three intervals. First, the missing interval, where the activation radius is below $|e^*(k)| - r_s$. No active sensors can sense the target, and the whole WSN may fail to track the target. Second, the saturation interval, where the activation radius is above $|e^*(k)| + r_s$, all the sensing nodes are activated and the number of sensing nodes will not grow as the activation radius increases, and the estimation error will not reduce. Third, the trade-off interval, where the activation radius is between $|e^*(k)| - r_s$ and $|e^*(k)| + r_s$, and optimization on the activation radius can be performed. The three intervals will be illustrated in Section 7.

In summary, the appropriate range of the activation radius can be expressed as follows:

$$|e^*(k)| - r_s < r_a < |e^*(k)| + r_s,$$  \hspace{1cm} (17)

where the lower bound is for the dynamic coverage requirement and the upper bound for not activating nodes more than need.

6.2. Trajectory estimation

In the estimation phase, the tracking accuracy requirement is taken into account. The goal of this phase is to balance tracking accuracy with energy consumption. A new structure for energy consumption is established with the finite sensing range under the above activation mechanism. The power consumption is related to two types of nodes: the activated nodes...
without sensing, and the sensing nodes. The numbers of activated/sensing nodes, rather than the direct energy consumption amount, are considered as energy indices. Because the energy consumption amount varies with different localization methods and different sensing signals such as acoustic, seismic, electromagnetic. Consequently three performance indices are considered in the trade-off problem: the tracking accuracy $P$, the number of activated nodes $n_a$, and the number of sensing nodes $n_s$.

The tracking accuracy is related to the number of broadcasting nodes which can be determined by: selecting broadcasting nodes in the sensing node set, and/or adjusting the activation radius to change the size of the sensing node set. The first selection method, such as the GEBF policy, can be applied when the diversity of sensing capability is taken into account. The approach follows the partial broadcasting framework presented in Section 5. When all the sensors are homogeneous, we do not need select but let all the sensing nodes broadcast. Meanwhile the size of the sensing node set, i.e., the intersection of the activation and sensing areas, is determined by the activation radius $r_a$ given by (17) where the optimization occurs within the trade-off interval. The relationship between the decision variable $r_a$ and the three performance indices, is stated by the following theorem.

**Assumption 6.** The sensors are assumed to be homogeneous, which means that the noise co-variances and the measurement matrices are the same and time-invariant, denoted by $R$ and $H$.

**Theorem 3.** Suppose that sensors are uniformly deployed with the density $\rho$. The tracking procedure is as in Table 3 under Assumption 6. If the activation radius $r_a$ satisfies (17), for the number of activated nodes $n_a$, the number of sensing nodes $n_s$, and the tracking accuracy denoted by the a priori error covariance $P^{-1}(k)$, we have

$$n_a = \pi \rho r_a^2,$$

$$n_i(k) = \rho \left[ r_a^2 (\theta_1(k) - \sin \theta_1(k) \cos \theta_1(k)) + r_a^2 (\theta_2(k) - \sin \theta_2(k) \cos \theta_2(k)) \right],$$

where

$$\cos \theta_1(k) = \frac{r_a^2 + |e^*(k)|^2 - r_s^2}{2 r_a \cdot |e^*(k)|},$$

$$\cos \theta_2(k) = \frac{r_s^2 + |e^*(k)|^2 - r_a^2}{2 r_s \cdot |e^*(k)|},$$

$$P^{-1}(k + 1) = F P^{-1}(k) F^T + Q - F P^{-1}(k) H^T (H P^{-1}(k) H^T + R/n_i(k))^{-1} H P^{-1}(k) F^T,$$

where

$$P^{-1}(k) = \mathbb{E}[(\hat{x}(k|k-1) - x(k))(\hat{x}(k|k-1) - x(k))^T].$$

**Proof.** Eqs. (18) and (19) are obtained by geometry, i.e.,

$$n_a = \pi r_a^2.$$  

In Fig. 5, the area of the half segment A is

$$S_A = S_{\text{sector}} - S_{\text{triangle}} = \frac{1}{2} r_a^2 (\theta_1(k) - \sin \theta_1(k) \cos \theta_1(k)).$$

![Fig. 5. Overlap area of sensing area and activation area.](image-url)
So,
\[ S_{\text{overlap}} = r_1^2(\theta_1(k) - \sin \theta_1(k) \cos \theta_1(k)) + r_2^2(\theta_2(k) - \sin \theta_2(k) \cos \theta_2(k)), \]
\[ n_i(k) = \rho S_{\text{overlap}}. \]

This completes the proof of (19). Eqs. (20) and (21) are derived directly by the cosine theorem of the angle.

The last claim of the a priori error covariance \( P^*(k) \) is proved using the conclusion of Theorem 1. If all the sensors have the same \( R \) and \( H \), when there are \( n_i(k) \) nodes broadcasting at time \( k \), the measurement update for the error covariance is given by (7), i.e.,
\[ (P(k))^{-1} = (P^-(k))^{-1} + n_i(k) H^T R^{-1} H. \]

By applying the matrix inversion lemma \( (A_1 + A_2 A_3^{-1} A_2)^{-1} = A_1 - A_1 A_2 (A_3 + A_2 A_1 A_2)^{-1} A_2 A_1 \) (23) can be expressed as
\[ P(k) = P^-(k) - P^-(k) H^T (H P^-(k) H^T + R/n_i(k))^{-1} H P^-(k). \]
Combining with the time update for the error covariance,
\[ P^*(k + 1) = F P(k) F^T + Q, \]
we can get the iteration of the a priori error covariance of (22). □

The relationship between \( n_m, n_i(k) \), \( P^*(k) \) and the activation radius \( r_a \) given by Theorem 3 will be demonstrated in Section 7.

Unlike the traditional Riccati equation, Eq. (22) has no steady-state solution. However, the bounds can be provided for \( P^*(k) \) by the following two theorems. Define two matrix sequences \( \{\bar{P}(k), k = 0, 1, 2, \ldots\} \) and \( \{\underline{P}(k), k = 0, 1, 2, \ldots\} \)
\[ P(k + 1) = F P(k) F^T + Q - F P(k) H^T (H P(k) H^T + R) H P(k) F^T \]
and
\[ P(k + 1) = F P(k) F^T + Q - F P(k) H^T (H P(k) H^T + R) H P(k) F^T \]
with the same initial value,
\[ P(0) = P(0) = P^*(0), \]
where \( P^*(0) \) and \( R \) are positive definite, \( N_i \) is the number of all the sensors in the sensing area,
\[ N_i = \pi r_a^2. \]

**Theorem 4.** Under the assumption in (15), we have
\[ P(k) \leq P^*(k) \leq \bar{P}(k), \quad k = 1, 2, \ldots. \]

**Proof.**

The proof is given in Appendix. □

**Theorem 4** describes the bounds of \( P^*(k) \) in one estimation step, whereas its bounds in the entire estimation horizon are given by **Theorem 5**.

Let \( P \) and \( \bar{P} \) be a solution of the following equations,
\[ \bar{P} = F \bar{P} F^T + Q - F \bar{P} H^T (H \bar{P} H^T + R)^{-1} H \bar{P} F^T \]
and
\[ P = FPF^T + Q - FPH^T (H P H^T + R/N_i)^{-1} HPF^T. \]

**Theorem 5.** Suppose \( (F, H) \) is reachable, \( (F, H) \) is detectable, and \( R > 0 \). Then for every \( \varepsilon > 0 \), there exists a \( K(\varepsilon) \in N \) such that for every \( k > K(\varepsilon) \), \( P - \varepsilon I < P^*(k) < P + \varepsilon I. \)

**Proof.** If \((F, Q)\) is reachable, \((F, H)\) is detectable, and \( R > 0 \), there is a unique positive-definite limiting solution to (25) which is independent of the initial value [17]. Furthermore, this solution is the unique positive definite solution to (28), i.e., \( P \). So for every \( \varepsilon > 0 \), there exists a \( K(\varepsilon) \) such that for every \( k > K(\varepsilon) \),
\[ P(k) > P - \varepsilon I. \]
Theorem 4 states that if $P(k)$ satisfies (25) with the initial value $P(0) = P(0)$, then
\[ P^+ (k) \geq P(k), \quad k \in \mathbb{N}. \] (30)

By (29) and (30), we have that for every $\varepsilon > 0$, there exists a $K_1(\varepsilon)$ such that for every $k > K_1(\varepsilon)$,
\[ P^+ (k) > P + \varepsilon. \]
Similarly, for every $\varepsilon > 0$, there exists a $K_2(\varepsilon)$ such that for every $k > K_2(\varepsilon)$,
\[ P^-(k) < P - \varepsilon. \]

Let $K(\varepsilon) = \max(K_1(\varepsilon), K_2(\varepsilon))$, which completes the proof. \(\square\)

Theorem 5 states that the bounds are not determined by the initial value, but only related with the target dynamic model and parameters of the WSN, such as the density of sensors $\rho$ and the sensing radius $r_s$. Therefore, the bounds can be determined beforehand. In practice $P$ and $P^+$ provide good approximations for the upper and lower bounds of $P^-(k)$ when $k$ is large enough.

7. Numerical results

In this section we demonstrate the performance of the PIBS in two parts. First, assuming infinite sensing range of each sensor, we compare two PIBS policies, namely RB and GEBF, and show that GEBF achieves higher tracking accuracy than RB under the same energy consumption. Second, a more realistic situation is considered that each sensor has a finite sensing range and a detailed energy consumption model is also incorporated. The impact of activation radius on various performance metrics is demonstrated. Numerical results show that the Adjusting Activation Radius (AAR) method developed in Section 6 outperforms the existing Prediction-based Energy Saving (PES) scheme [42].

7.1. Case study with infinite sensing range: PIBS with RB vs. GEBF

Consider the following well-used one-dimensional nearly-constant-velocity model [18], i.e.,
\[ x(k + 1) = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} \Delta^2 t/2 \\ \Delta t \end{bmatrix} w(k). \]
Recall that $x(k)$ describes the position and velocity of the target; $\Delta t = 1$ is the length of each time interval; $w(k) \sim \mathcal{N}(0,0.02)$ is the state noise. Consider 16 sensors, each of which uses a linear measurement model,
\[ z_i(k + 1) = Hx(k) + v_i(k), \]
where $H = [1 \ 0]$ means that the sensors can only estimate the position of the target, but not the velocity; $v_i(k) \sim \mathcal{N}(0,25)$ is the measurement noise. For a randomly picked $x(0)$, we simulate the performance of PIBS with RB and PIBS with GEBF for 80 steps, and repeat the simulation with 50 replications. We have the following remarks.

Remark 1. The tracking accuracy denoted by $tr(P(k))$ under both RB and GEBF converges asymptotically in the sense of expectation. As an example, $tr(P(k))$’s for Sensor 1 in RB is shown for different $k$’s in Fig. 6.

![Fig. 6. Estimation error covariance $P(k)$ of the RB policy with time.](image-url)
Remark 2. For a given time $k$, the tracking accuracy is improved under both RB and GEBF if more sensors are selected (namely a large value of $\alpha$ in RB or a large value of $c$ in GEBF). The improvement gradually decreases as the number of selected sensors increases.

Remark 3. For a given time $k$, GEBF achieves higher tracking accuracies than RB when the same number of sensors are selected. As an example, $\text{tr}(P(k))$ of Sensor 1 for RB and GEBF under different values of $\alpha$ and $c$ are shown in Fig. 7.

7.2. Case study with finite sensing range: AAR vs. PES

For the case of finite sensing range, the energy model of the Mica2 node (the commercial sensor node of Crossbow Inc.) is introduced. Four metrics are used for evaluation and comparison:

- **Tracking accuracy:** the trace of estimation error covariance.
- **Missing rate:** the time fraction when the target is missed during the tracking process.
- **Total energy consumption:** the total energy consumed both on the activated nodes and effective sensing nodes.
- **Effective node number:** the average number of the effective sensing nodes.

The first two indicate tracking performance and the last two indicate tracking cost.

The scenario is to track a vehicle in a $400 \times 400$ m$^2$ field. The deployment area is sufficiently large to cover the whole target path and eliminate the marginal effect, as illustrated in Fig. 2. The target motion model is a nearly-constant-velocity model extended in 2-D plane, i.e.,

$$
\begin{pmatrix}
    d_x(k + 1) \\
    v_x(k + 1) \\
    d_y(k + 1) \\
    v_y(k + 1)
\end{pmatrix} =
\begin{bmatrix}
1 & \Delta t & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & \Delta t \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{pmatrix}
    d_x(k) \\
    v_x(k) \\
    d_y(k) \\
    v_y(k)
\end{pmatrix} +
\begin{bmatrix}
\Delta t^2 / 2 & 0 \\
\Delta t & 0 \\
0 & \Delta t^2 / 2 \\
0 & \Delta t
\end{bmatrix}
\begin{pmatrix}
w_x(k) \\
w_y(k)
\end{pmatrix}.
$$

The time interval $\Delta t$ is 1 s. The initial state is $[0 7.84 0 7.84]^T$ which means the initial speed is 40 kmph. The state noise accounting for the unpredictable modeling error is characterized by

$$
Q = E[w(k)w^T(k)] =
\begin{bmatrix}
0.03 & 0 \\
0 & 0.03
\end{bmatrix}.
$$

The sensors are deployed uniformly with the density of 1 sensor/100 m$^2$. The limited sensing radius $r_s$ is 15 m. In the sensing range, the measurement model is

$$
\begin{pmatrix}
z_x(k + 1) \\
z_y(k + 1)
\end{pmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\begin{pmatrix}
d_x(k) \\
v_x(k) \\
d_y(k) \\
v_y(k)
\end{pmatrix} +
\begin{pmatrix}
w_x(k) \\
w_y(k)
\end{pmatrix}.
$$

Fig. 7. Comparison of estimation accuracies of RB and GEBF policies.
The parameters of measurement noise covariance are

\[ R_i = \frac{\mathbb{E}(v_x(k)v_y(k))}{\mathbb{E}(v_x(k))^2} = \begin{pmatrix} 400 & 0 \\ 0 & 400 \end{pmatrix}. \]

The correlation of the measurement noise between sensors is introduced to account for the unpredicted sources rather than electronic noise. The correlation coefficients are defined as

\[ \text{coef}_x = \frac{\mathbb{C}(v_u(k), v_w(k))}{\mathbb{C}(v_u(k))\mathbb{C}(v_w(k))}. \]

and \( \text{coef}_y \) is defined similarly. The parameter \( \text{coef}_x \) and \( \text{coef}_y \) are identical in the tests, denoted as \( \text{coef} \).

The energy model is based on the Mica2 platform. The Mica2 sensors can be in four states: sleeping, idle/listening, transmitting, and receiving. The activated nodes are in the idle state, while the effective sensing nodes broadcast data. The radio-triggered [6] circuit for waking up the sleeping sensor consumed negligible energy compared with the normal communication. Therefore, the energy consumption model is

\[ E_{\text{total}} = n_a \cdot P_{\text{idle}} \cdot t_{\text{sen}} + \left| n_a \cdot P_{\text{Tx}} + (n_s - n_a) \cdot P_{\text{Rx}} \right| \cdot t_{\text{com}}. \]

The parameters of the energy model [31] are listed in Table 4. Accordingly, the energy consumptions of three working states are in Table 5. The communication protocol is TDMA, where the time is divided into slots for nodes' broadcasting. As the previous setting, the total number of nodes in the sensing region \( N_s \) is

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
Component & Mode & Power (mW) \\
\hline
CPU & Active & 24 \\
CPU & Idle & 9.6 \\
Sensor board & Active & 2.1 \\
Radio & Rx & 21 \\
Radio & Tx (+10dBm) & 64.5 \\
\hline
\end{tabular}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
 & \( P_{\text{idle}} \) (mW) & \( P_{\text{Tx}} \) (mW) & \( P_{\text{Rx}} \) (mW) \\
\hline
CPU & 9.6 & 24 & 24 \\
Sensor board & 2.1 & 0 & 0 \\
Radio & 21 & 64.5 & 21 \\
Total & 32.7 & 88.5 & 45 \\
\hline
\end{tabular}
\end{table}

Fig. 8. Tracking accuracy vs. activation radius.
We can divide time interval $\Delta t$ into 10 slots, thus $t_{\text{com}}$ is 0.1 s and $t_{\text{sen}}$ is 0.9 s.

We test the activation radius $r_a$ varying from 6 m to 34 m with a step of 2 m. We run 200 trials for each $r_a$ to obtain the average result shown in Figs. 8–11. We make the following remarks:

**Remark 4.** The variation of tracking accuracy is dramatic when the activation radius is small as shown by the error bars in Fig. 8. It means the WSN may lose the target in this case. The effective node number does not increase when the activation radius is large. It means that the expansion of activation radius makes no contribution to the improvement of tracking accuracy, but only rapidly increases the energy cost. These observations are consistent with the analysis on the value range of activation radius in Section 6.1.

**Remark 5.** The measurement correlation hinders the tracking performance by worsening the tracking accuracy and raising the missing rate. The measurement correlation also increases the total energy consumption and decreasing the effective node number. Thus it affects the boundaries of the three intervals of activation radius discussed in Section 6.1.

We also compare our AAR method with the PES scheme [42] and show the results in Fig. 12 and in Table 6. There are three heuristic wakeup mechanisms in PES: DESTINATION, ROUTE, and ALL_NBR. The first one, which wakes up only one node close to the target destination, has the target missing rate of 60% in our test. The last one, which wakes up all the nodes in the network, has the target missing rate of 10% in our test. The second one, which wakes up all the nodes in the network, has the target missing rate of 30% in our test.
**Fig. 11.** Total energy consumption vs. activation radius.

**Fig. 12.** Comparison of PIBS and PES.

### Table 6
The comparison data of PES and PIBS.

<table>
<thead>
<tr>
<th></th>
<th>PES</th>
<th>PIBS ($r_a = 14$)</th>
<th>PIBS ($r_a = 15$)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std</td>
<td>Mean</td>
</tr>
<tr>
<td>Tracking accuracy</td>
<td>53.0</td>
<td>218</td>
<td>57.0</td>
</tr>
<tr>
<td>Missing rate</td>
<td>1.14%</td>
<td>0.075</td>
<td>0.820%</td>
</tr>
<tr>
<td>Total energy</td>
<td>28.3</td>
<td>1.17</td>
<td>23.0</td>
</tr>
<tr>
<td>Number of activated nodes $n_a$</td>
<td>7.71</td>
<td>0.352</td>
<td>6.13</td>
</tr>
<tr>
<td>Number of effective nodes $n_s$</td>
<td>4.91</td>
<td>0.547</td>
<td>4.96</td>
</tr>
<tr>
<td>Effective ratio $n_s/n_a$</td>
<td>63%</td>
<td>81%</td>
<td>78%</td>
</tr>
</tbody>
</table>
neighboring nodes surrounding the route, consumed much more energy than the other two. Therefore, we only compare our AAR method with PES ROUTE, which wakes up the nodes on the route from the current position to the destination. We have the following remark.

**Remark 6.** The AAR method with an appropriate radius outperforms the PES with ROUTE. When the activation radius \( r_a \) is between 14 m to 15 m, the performances of PIBS including the tracking accuracy and missing rate are as good as the ones of PES ROUTE, or even better, and the total energy consumption of PIBS is lower than PES ROUTE. The reason is that the effective node ratio \( n_d/n_a \) is optimized by adjusting \( r_a \). These subfigures also show that the \( r_a \) is critical. If \( r_a \) is improper, the performances of the new method may deteriorate and the cost may increase.

### 8. Conclusions

A partial information broadcasting scheme is presented for the tracking problem via a wireless sensor network. It is formulated as a two-layer optimization problem with the communication and energy balance constraints. Two broadcasting policies, the Randomly Broadcast (RB) policy and the Good Estimates Broadcast First (GEBF) policy, are developed and compared. The GEBF policy with the indicator of measurement covariance is proved to be the Pareto-optimal policy when the number of broadcasting nodes is fixed. The RB policy can balance energy storages of sensors. Both policies can be implemented in a decentralized way. The numerical results show that the GEBF policy performs better than the RB policy. In the case of finite sensing range, the value range of the activation radius is divided into three intervals by both analytical analysis and numerical experiments. We theoretically quantify the relationship among the tracking accuracy, the activation radius and energy consumption, and derive the upper and lower bounds on the tracking accuracy. The comparison demonstrates that our activation scheme outperforms the PES scheme when the activation radius is appropriate.

Note that linear Gaussian models are applied for motion dynamics and estimation. For more general cases where the motion and measurement models could be nonlinear, one may use the extended Kalman filter [39]. However that is beyond the scope of this paper. For non-identical sensors, the measurement matrices \( H_i \)'s of neighboring sensors need to be known. When the topology of the WSN is fixed, the \( H_i \)'s are required to be transmitted only once.

### Acknowledgement

This research presented in this paper is supported in part by the NSFC (60574087, 60704008, 60736027, 60921003, and 90924001), 863 High Tech Development Plan (2008AA01Z415), the MOE New Faculty Member Funding (20070003110), and the National 111 International Collaboration Project (B06002), and Tsinghua-UTC Research Institute for Integrated Building Energy, Safety, and Control Systems.

### Appendix A

**Proof of Theorem 1.** From the Kalman filter [12], it is known that given the system dynamics of (1), the measurement model

\[
z(k) = H(k)x(k) + v(k)
\]

and with the assumption that the system noise \( w(k) \) and measurement noise \( v(k) \) are uncorrelated white Gaussian noise, and \( w(k) \sim N(0, Q(k)) \), \( v(k) \sim N(0, R(k)) \), the optimal MSE estimator is

\[
\hat{x}(k) = \hat{x}(k|k - 1) + K(k)[z(k) - H(k)\hat{x}(k|k - 1)],
\]

where

\[
K(k) = P(k|k)H^T(k)R^{-1}(k),
\]

\[
P^{-1}(k) = P^{-1}(k|k - 1) + H^T(k)R^{-1}(k)H(k),
\]

\[
\hat{x}(k|k - 1) = F\hat{x}(k - 1)
\]

and

\[
P(k - 1) = FP(k - 1)F^T + Q.
\]

Note that (33) can be rewritten as

\[
P(k) = [I - K(k)H(k)]P(k|k - 1).
\]

The measurement that can be obtained by the \( i \)th node is

\[
Z_{i}^k = H_i^T(k)x(k) + V_i^k(k),
\]

where \( H_i^k = [(H_1(k))^T, \ldots, (H_{mk_i}(k))^T]^T \) and \( V_i^k = [(v_1(k))^T, \ldots, (v_{mk_i}(k))^T]^T \). By substituting \( H_i^k \) and \( V_i^k \) into (33), we have
Proof of Theorem 2. It is known that the matrix inversion of an n-by-n matrix has a computational complexity of $O(n^3)$. To reduce the computation, we can explore the block diagonal structure of $R_i^p(k)$ due to the independent property of noises of different nodes by Assumption 1, i.e.,

$$R_i^p(k) = E\left[V_i^p(k)\right] = \text{blockdiag}[R_{11}(k), \ldots, R_{m(k)}(k)].$$

So (37) can be rewritten as

$$(P_i^p(k))^{-1} = P_i^{-1}(k|k-1) + \sum_{i'\in\mathcal{I}(k)} (H_{i'}(k))^T R_{i'}^{-1}(k) H_{i'}(k),$$

which is exactly the same as (7).

Then we substitute $Z_i^p(k)$ and $H_i^p(k)$ into (31) to get the estimator under the PIBS,

$$X_i^p(k) = \hat{x}_i(k|k-1) + K_i^p(k) [Z_i^p(k) - H_i^p(k)\hat{x}_i(k|k-1)] = \left[I - K_i^p(k)H_i^p(k)\right]\hat{x}_i(k|k-1) + K_i^p(k)Z_i^p(k).$$

From (36), we know that

$$P_i^p(k) = \left[I - K_i^p(k)H_i^p(k)\right]P_i(k|k-1).$$

Since $P_i^p(k)$ is invertible, we have

$$(P_i^p(k))^{-1} = (P_i(k|k-1))^{-1}\left[I - K_i^p(k)H_i^p(k)\right]^{-1}.$$  \hspace{1cm} (39)

Combining (38) and (39), we have

$$(P_i^p(k))^{-1}X_i^p(k) = (P_i(k|k-1))^{-1}\hat{x}_i(k|k-1) + (P_i^p(k))^{-1}K_i^p(k)Z_i^p(k).$$  \hspace{1cm} (40)

From (32), we have

$$K_i^p(k) = P_i^p(k)\left[H_i^p(k)^T R_i^p(k)\right]^{-1}.$$  \hspace{1cm} (41)

By Substituting (41) into (40), we have

$$(P_i^p(k))^{-1}X_i^p(k) = (P_i(k|k-1))^{-1}\hat{x}_i(k|k-1) + (H_i^p(k))^T R_i^p(k)^{-1}Z_i^p(k).$$

Under Assumption 1, the above equation can be rewritten as

$$(P_i^p(k))^{-1}X_i^p(k) = (P_i(k|k-1))^{-1}\hat{x}_i(k|k-1) + \sum_{i'\in\mathcal{I}(k)} (H_{i'}(k))^T R_{i'}^{-1}(k) z_{i'}(k),$$

which is exactly the same as (5). Eqs. (6) and (8) can be inferred directly from (34) and (35). \hfill \Box

Proof of Theorem 2. First, the GEBF policy that satisfies (12) is feasible to $P_3$. For convenience, we denote the set $\{i : i_j(k) = 1, i(k) = [i_j(k)]\}$ as $Pub(i(k))$. From (7), we know that for $i \in Pub(i(k))$ we have

$$(P_i^p(k))^{-1} = (P_i(k|k-1))^{-1} + \sum_{j : Pub(k)} H_j^T R_j^{-1}(k) H_j.$$  \hspace{1cm} (37)

for $i \notin Pub(i(k))$, we have

$$(P_i^p(k))^{-1} = (P_i(k|k-1))^{-1} + \sum_{j : Pub(k)} H_j^T R_j^{-1}(k) H_j + H_j^T R_j^{-1}(k) H_j.$$  \hspace{1cm} (37)

For any other feasible policy $\hat{i}(k)$, we denote the non-empty sets $Out$ and $In$ as follows,

$$Out = Pub(i(k)) \setminus Pub(\hat{i}(k)), \hspace{1cm} In = Pub(\hat{i}(k)) \setminus Pub(i(k)).$$

Then the set $Out$ and $In$ have the same size. As $\{R_j(k), j \in Pub(\hat{i}(k))\}$ are the top-$c(k)$ minimum, we have

$$\sum_{j \in Out} H_j^T R_j^{-1}(k) H_j \geq \sum_{j \in In} H_j^T R_j^{-1}(k) H_j.$$  \hspace{1cm} (37)

For $i \in In$, 
Proof of Theorem 4. Under the condition of (15) that guarantees the WSN is able to track the target, there is at least one sensor which can detect the target and take measurements, and \( n_s(k) \) is no more than total number of sensors in the sensing area, that is
\[
1 < n_s(k) \leq N_s.
\]
Since \( n_s(k) \) is always positive and \( R \) is positive definite, we have
\[
\frac{R}{N_s} \leq \frac{R}{n_s(k)} \leq R.
\]
We can prove \( P(k) \leq P^*(k) \) by induction. When \( k = 0 \), the inequality \( P(0) \leq P^*(0) \) is obvious because of (26). Suppose \( P^*(k) \leq P^*(k) \) is satisfied when \( k = i, i \in \mathbb{N} \) that is
\[
P^*(i) < P^*(i).
\]
Now we consider the case when \( k = i + 1 \). If we define
\[
M_1 = P^*(i) - P^*(i)H^T(HP^*(i)H^T + R/n_s(i))^{-1}HP^*(i),
\]
then
\[
P^*(i + 1) = FM_1F^T + Q.
\]
Based on the matrix inversion lemma, this can be expressed as
\[
M_1^{-1} = (P^*(i))^{-1} + n_s(i)H^T R^{-1}H.
\]
Similarly, if we define
\[
M_2 = P(i) - P(i)H^T(HP(i)H^T + R/N_s)^{-1}HP(i),
\]
then
\[
P(i + 1) = FM_2F^T + Q.
\]
\[
M_2^{-1} = (P(i))^{-1} + N_s H^T R^{-1}H.
\]
By subtracting (45) from (47), we have
\[
M_2^{-1} - M_1^{-1} = (P(i))^{-1} - (P^*(i))^{-1} + (N_s - n_s(i))H^T R^{-1}H \geq 0.
\]
Based on (43) which implies
\[
(P(i))^{-1} - (P^*(i))^{-1} \geq 0,
\]
\( R > 0 \), and \( n_s(k) < N_s \), we have
\[
M_1 - M_2 \geq 0.
\]
Combining (44) and (46), we have
\[
P^*(i + 1) - P(i) = FM_1F^T \geq 0
\]
and this completes the proof of \( P(k) \leq P^*(k) \). Accordingly, \( P^*(k) \leq P(k) \) can be proved in the same way. \( \square \)
References


