Energy Efficient Distributed Target Tracking Algorithm in Underwater Wireless Sensor Networks

Chang Ho Yu\(^1\), Kang Hoon Lee\(^2\), Jae Weon Choi\(^2\), Myeong-Kwan Park\(^2\), Taek Lyul Song\(^3\), Tae Il Seo\(^4\), and Young Bong Seo\(^5\)

\(^1\) Department of Intelligent Mechanical Engineering, Pusan National University, Busan, Korea
\(^2\) School of Mechanical Engineering, Pusan National University, Busan, Korea
\(^3\) Department of Control and Instrumentation Engineering, Hanyang University, Gyeonggi-do, Korea
\(^4\) Agency for Defense Development, JinHae, Korea
\(^5\) Innovation Center for Engineering Education, Pusan National University, Busan, Korea

ABSTRACT

The growing interests in underwater wireless sensor networks (UWSNs) for a wide variety of purposes from Tsunami monitoring to commercial oilfield exploration projects have been encouraging. This paper deals with the problem of accurately tracking a single target moving through UWSNs employing acoustic sensors. This paper addresses the issues of estimating the target position, improving energy efficiency by applying a Kalman filter in a distributed architecture. Each underwater wireless sensor node composing the UWSNs is battery-powered, so the energy conservation problem is a critical issue. This paper provides an algorithm which increases energy efficiency of each sensor through a wake-up/sleep (WuS) and a valid measurement selecting (VMS) scheme. Simulation results illustrate the performance of the tracking filter according to the sensor node displacement and sensor detecting area.

KEY WORDS: UWSN, Kalman filter, Wake-up/Sleep (WuS) scheme, Valid Measurement Selecting (VMS) scheme.

INTRODUCTION

Advances in micro electromechanical systems (MEMS) and wireless technologies have allowed for the emergence of inexpensive micro-sensors with embedded processing and communication capabilities (Chong, Zhao, Mori, and Kumar, 2003). Sensor networks consist of many of these micro-sensors communicating with one another over wireless links. Wireless sensor networks (WSNs) are emerging technology for monitoring physical world with a densely deployed network of sensor nodes. The main advantages of WSNs include its low cost, rapid deployment, self-organization, and fault tolerance. It can be used in a variety of applications, such as environment monitoring, traffic monitoring in intelligent transportation systems, industrial sensing and diagnostics, healthcare, navigation and control of mobile robots, and military surveillance.

One of the main constraints for WSNs is energy. It typically is impractical to replace the battery on each sensor node, so the lifetime of the network is tied to the battery life. The authors in (Brooks, Ramanathan, and Sayeed, 2003) argue for the need to study the problem of energy-aware target tracking for wireless sensor networks.

One approach to target tracking is to use a centralized architecture, where all sensor measurements are sent to a central processing station. However, this architecture demands significant bandwidth for communication of every node’s sensor measurements, which can put a strain on the node battery lifetime (Chong, Zhao, Mori, and Kumar, 2003). Other issues with a centralized architecture are the lack of robustness and reliability.

To reduce the communication burden, a distributed architecture is preferred for wireless sensor networks. In a distributed architecture, a sensor node is chosen near the target position as a processing node, and a subset of local sensors in the network is chosen to collect sensor data while the remaining sensors are placed in a power-saving sleep mode. An added advantage to using a distributed architecture is its robustness to network changes and node failures (Chong and Kumar, 2003).

In this paper, we look at an example of a distributed tracking algorithm for a moving target through UWSNs. We consider a single target tracking problem only. Extension to the multiple target tracking problem is not considered. We address the issues of estimating target position and improving energy efficiency by applying a Kalman filter in a distributed architecture. To estimate the states of the target, we propose the energy efficient tracking algorithm using the WuS and VMS scheme for a distributed target tracking.

The rest of the paper proceeds as follows. Section II describes the problem by defining the target motion models, as well as the measurement model of the UWSNs of acoustic sensors. Section III sets up the Kalman filtering framework that will be used for estimating the state of the target and describes a distributed target tracking architecture. In section IV, the energy efficient tracking algorithms, using the WuS and VMS scheme, are presented for distributed target tracking. In section V, simulation
results are presented and the performance evaluations of proposed algorithm are accomplished. Finally, conclusions of this study are drawn in section VI.

PROBLEM FORMULATION

In this paper, we will consider the tracking problem of a single target moving through UWSNs environment and at each time step we will select multiple sensors for detection and generating measurements.

UWSN Architecture

The UWSN architecture is composed of multiple underwater acoustic sensors because radio frequency (RF) signals cannot be transmitted in underwater environments. We consider an UWSN architecture with static acoustic sensor nodes for underwater event monitoring and patrolling unmanned underwater vehicles (UUVs) and submarines as mobile users accessing information from the UWSN anywhere, anytime. Static sensor nodes, settled on the bottom of the underwater, are capable of communicating with each other wirelessly over an acoustic channel and also with the UUVs and submarines. The UWSN may also be wirelessly linked up to a buoy, a mother-ship, and an offshore platform for data analysis. Fig. 1 shows such an architecture for underwater vehicles monitoring application (Heidemann, Li, and Syed, 2005).

State Vector and Dynamic Model

The state vector of a single target consists of position and velocity, in a two-dimensional (2D) Cartesian coordinate system. At the kth time step, the state vector $X(k)$ is defined as

$$X(k) = [x(k) \quad \dot{x}(k) \quad y(k) \quad \dot{y}(k)]^T,$$

whose elements are $x$ position, $\dot{x}$ velocity, $y$ position, and $\dot{y}$ velocity.

The state dynamics equation is given by

$$X(k+1) = \Phi X(k) + \Gamma w(k),$$

where $\Phi$ is the state transition matrix, $\Gamma$ is the process noise matrix, and $w(k)$ is the independent process noise with zero-mean, white, Gaussian probability distribution $N(0,Q(k))$.

Particularly, for a 2D constant velocity (CV) model (Bar-Shalom, Li, and Kirubarajan, 2001), whose parameters are

$$\Phi = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\Gamma = \begin{bmatrix} T^2 / 2 & 0 \\ T & 0 \\ 0 & T^2 / 2 \\ 0 & T \end{bmatrix}.$$

The process noise covariance matrix is

$$Q(k) = \begin{bmatrix} qT & 0 \\ 0 & qT \end{bmatrix},$$

where $q$ is the intensity of the process noise, and $T$ is the time interval between samples.

Measurement Model

UWSNs composed of $N$ acoustic sensor nodes are deployed in a 2D region. The positions of sensor nodes, in Cartesian coordinates denoted by $(x_i, y_i)$, $i = 1, \ldots, N$, are arbitrary but known to the fusion center. Suppose sensors $s_1, s_2, \ldots, s_N$ provide the measurements $z_1(k), z_2(k), \ldots, z_N(k)$ at the kth time step, the measurement model is given by

$$Z(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \\ \vdots \\ z_N(k) \end{bmatrix} = \begin{bmatrix} Hs_1(k) \\ Hs_2(k) \\ \vdots \\ Hs_N(k) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \\ \vdots \\ v_N(k) \end{bmatrix},$$

where $H$ is a measurement function of sensor $s_i$ and $v_i(k)$ is its measurement noise which is assumed to be independent with zero-mean, white, Gaussian probability distributions $N(0,R(k))$ (Xiao, Xie, Lin, and Li, 2006).
algorithm for estimating the state vector. The predicted state \( \hat{X}(k+1 | k) \) of the target can be calculated as
\[
\hat{X}(k+1 | k) = \Phi \hat{X}(k | k),
\]
with the predicted state error covariance
\[
P(k+1 | k) = \Phi P(k | k) \Phi^T + Q(k),
\]
and the predicted fused measurement is
\[
\hat{z}_{\text{fusion}}(k+1 | k) = H(\hat{X}(k+1 | k)).
\]
Then the innovation is given by
\[
\gamma(k+1) = z(k+1) - \hat{z}_{\text{fusion}}(k+1 | k),
\]
with the covariance
\[
S(k+1) = HP(k+1 | k)H^T + R(k+1).
\]
The Kalman filter gain is given by
\[
K(k+1) = P(k+1 | k)H^T S^{-1}(k+1),
\]
and the state estimation will be updated as
\[
\hat{X}(k+1 | k+1) = \hat{X}(k+1 | k) + K(k+1)\gamma(k+1),
\]
with the state error covariance matrix
\[
P(k+1 | k+1) = P(k+1 | k) - K(k+1)HP(k+1 | k).
\]

**Distributed Target Tracking Architecture**

The basic idea of the distributed target tracking architecture is illuminated in Fig. 2, where the dashed circles mean the sensor detection area. At the \( k \)th time step, there exists an ellipse which shows the estimation area of the target. At the \( k+1 \)th time step, a new ellipse which is predicted estimation area of the target can be achieved from the prediction step of the Kalman filter. The node B is selected as the processing node because the node B is the nearest node to the center of the predicted estimation ellipse and the information of the processing node A are transferred to a new processing node B. A new processing node B collects measurements that are inside the predicted estimation ellipse.

Finally at the \( k+1 \)th time step, a new updated ellipse can be achieved from the Kalman filter by using the measurements inside the predicted estimation ellipse. At every time step, the same procedures continue recursively.

**ENERGY EFFICIENT TRACKING SCHEMES**

This paper focuses on the distributed target tracking algorithm, so we propose two energy efficient tracking schemes. From the proposed schemes, the distributed target tracking algorithm can save the energy by reducing the communication burden.

**Wake-up/Sleep (WuS) Scheme**

The basic idea of the proposed WuS scheme is illuminated in Fig. 3. As shown in Fig. 3, the predicted and updated ellipses are transformed into circles because the noise features of \( x \) and \( y \) axes are independent and equal.
where \( \hat{X}_s(k+1|k) \) and \( \hat{X}_f(k+1|k) \) are the predicted \( x \) and \( y \) positions of the target, and \( s_{i,x} \) and \( s_{i,y} \) are the \( x \) and \( y \) positions of the \( i^{th} \) sensor node \( s_i \).

The \textit{WuS} scheme is illuminated in Fig. 4. The \textit{WuS} scheme selects \( n \) waked-up sensor nodes from \( N \) sensor nodes by judging which have the overlapping parts of the predicted \( 3\sigma \) circle.

![Fig. 4 Waked-up and sleeping sensors.](image)

The \textit{WuS} scheme can decide the existence of the overlapping parts if

\[
d_i \leq r_i + 3\sigma,  \tag{16}\]

where \( r_i \) is the detection radius of \( i^{th} \) sensor node \( s_i \), and \( n \) sensor nodes satisfying equation (16) are waked-up. Shown in Fig. 3 and Fig. 4, the waked-up sensor nodes are green circles with black center. And the others, \( N-n \) sensor nodes are changed into the sleeping mode if the overlapping parts do not exist, that is

\[
d_i > r_i + 3\sigma.  \tag{17}\]

A new processing node \( s_p \) whose detection area is yellow circle, is selected by

\[
\min(d_1,\ldots,d_n).  \tag{18}\]

Thus from the \textit{WuS} scheme, we can select only several sensor nodes among all sensor nodes, and the processing node doesn’t have to communicate all of the sensor nodes. This results in increasing the energy efficiency.

\textbf{Valid Measurement Selecting (VMS) Scheme}

From the proposed \textit{WuS} scheme, we only take several sensor measurements within predicted \( 3\sigma \) circle. However, we have to consider the validity of these multiple sensor measurements. The \textit{VMS} scheme, shown in Fig. 5 is related with the detection area of each sensor and the actual target position.

![Fig. 5 Valid Measurement Selecting scheme.](image)

In Fig. 5, the actual target position which is black star symbol, can be achieved from the measurement equation (6). The distance \( D_i \) between the actual target position and position of the \( i^{th} \) sensor node \( s_i \) is

\[
D_i = \sqrt{\left(z_i(k)-s_{i,x}\right)^2 + \left(z_i(k)-s_{i,y}\right)^2}. \tag{19}\]

The \textit{VMS} scheme determines \( m \) valid sensor nodes which can actually detect the target and give measurement data to the processing node. The validity of each sensor node can be calculated by comparing \( D_i \) with \( r_i \). If \( m \) sensors are satisfying

\[
D_i \leq r_i,  \tag{20}\]

these sensors are valid and can give measurements to the processing node. However, the others, \( n-m \) sensors which satisfying

\[
D_i > r_i,  \tag{21}\]

are not valid and cannot detect the target any longer. Thus from the \textit{VMS} scheme, we can reduce the number of sensor nodes among the waked-up sensor nodes. This results in increasing the more energy efficiency.

The valid sensor nodes provide measurement \( z_i(k+1), z_j(k+1), \ldots z_n(k+1) \) to the processing node \( s_p \), and the processing node calculates the fused one measurement \( \overline{z}_{\text{fusion}}(k+1) \) considering the weighting of the all valid sensor measurement covariance matrices. Thus the measurement equation (6) can be modified by

\[
\overline{z}_{\text{fusion}}(k+1) = \frac{R_i(k+1)}{R_i(k+1) + R_j(k+1)} z_i(k+1) \tag{22}
\]

\[
+ \frac{R_j(k+1)}{R_i(k+1) + R_j(k+1)} z_j(k+1),
\]

where \( R_i(k+1) \) is the measurement covariance matrix. And the fused covariance \( R_{\text{fusion}}(k+1) \) can be achieved by

\[
R_{\text{fusion}}(k+1) = \left[R_i(k+1)^{-1} + R_j(k+1)^{-1}\right]^{-1} R_i(k+1) R_j(k+1) \tag{23}
\]

The processing node \( s_p \) updates the state of the target and the state error covariance matrix from the fused measurement \( \overline{z}_{\text{fusion}}(k+1) \) and fused measurement covariance matrix \( R_{\text{fusion}}(k+1) \). At every time step, the same procedures continue recursively, shown in Fig. 6. After the prediction step, the proposed tracking algorithm wakes up the sensor nodes within the \( 3\sigma \) prediction circle, and searches valid sensors within the waked-up sensors. In the case that both the waked-up sensors and valid sensors are existed, the proposed algorithm proceeds the update step II which is derived at Eq. (24)~(26) using the
fused measurement and covariance.
\[
\begin{align*}
\dot{x}(k+1|k+1) &= \tilde{x}(k+1|k) \\
&+ K(k+1) \{z_{\text{fusion}}(k+1) - H\tilde{x}(k+1|k)\} \\
P(k+1|k) &= P(k+1|k) - K(k+1)HP(k+1|k) \\
K(k+1) &= P(k+1|k)H^T \\
&\times \{HP(k+1|k)H^T + R_{\text{fusion}}(k+1)\}^{-1}
\end{align*}
\]  
(24)

(25)

(26)

\[R(k) = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}. \]

(31)

and the initial state error covariance is assumed as

\[P(0) = \begin{bmatrix} 25 & 0 & 0 \\ 0 & 25 & 0 \\ 0 & 0 & 10 \end{bmatrix} \]

(32)

The detection radius of the \(i\)-th sensor node \(s_i\) is assumed to be \(50m\), that is, the detection radius of all sensor nodes is assumed same. And the sensor node distribution is depicted as Fig. 7, and the number or the sensor nodes is 100 and the unit of each axes is meter. And we assume that the target moves at constant velocity.

SIMULATIONS

There are special characteristics that sensor nodes cannot be fixed and move according to the current and wave in the UWSN environment. Thus we can assume that sensor nodes of the UWSN are randomly distributed.

We apply the proposed distributed target tracking algorithm including the WuS and VMS schemes to tracking of a moving target through two kinds of UWSN environments. We evaluate the performance of the proposed algorithm by computer simulations.

Simulation Conditions

We assume the time interval \(T\) as 1sec and initiate the state and the estimate of the state as

\[X(0) = \begin{bmatrix} 20 & 20 & 20 & 20 \end{bmatrix}^T, \]

\[\hat{X}(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T. \]

The intensity of the process noise \(q\) is 10 and the measurement covariance \(R(k)\) is assumed as

Fig. 6 Flow chart of the distributed target tracking filter.

In the case that either the waked-up sensors or valid sensors are not existed, the proposed algorithm proceeds the update step (strictly not update but prediction) which is derived at Eq. (27)–(28).

\[\dot{x}(k+1|k+1) = \tilde{x}(k+1|k) \]

(27)

\[P(k+1|k+1) = P(k+1|k) \]

(28)

Fig. 7 100 sensor nodes randomly distributed

Simulation Results of Dense UWSNs

Fig. 8 2D target trajectory in dense UWSNs.

Fig. 8 shows the 2D trajectory of the target moving in the randomly distributed UWSNs. In these sensor arrangements, each sensor node is not overlapped, thus the target can be detected in only the detection area of each sensor node. Each cross symbol means each sensor node which is randomly
distributed, and each blue cross symbol represents a sleeping sensor node, and each red cross symbol is for a woke-up sensor node. A green solid line represents the actual target trajectory, and a red solid line is for the estimated trajectory of the target. And each large black dashed circle represents the predicted $3\sigma$ circle, and each large red dashed circle is for the updated $3\sigma$ circle. And each small black solid circle represents the valid sensor node which can actually detect a target.

Initially, every sensor node is sleeping. After the prediction step, there can be a predicted $3\sigma$ circle centered at the predicted position of the target. However there are not any sensor nodes, having the detection areas that are overlapped this circle, thus the proposed tracking filter proceeds the update step I. So the predicted $3\sigma$ circles and the standard deviation of the position error are increased until the tracking filter can search both the woke-up and valid sensor nodes. After 2 time step, there is one sensor node, having the detection areas that are overlapped $3\sigma$ circle, thus this sensor node is woke-up from the WuS scheme. And this sensor node is checked whether it can offer a valid measurement or not from the VMS scheme. In this case, the distance $D_3$ is bigger than $r$ and this sensor node is not valid, cannot detect the target. Thus the proposed tracking filter proceeds the update step I. So the predicted $3\sigma$ circles and the standard deviation of the position error are increased until the valid sensor measurement is identified which also depicted as Fig. 9. After another 2 time step, there are three sensor nodes woke-up from the WuS scheme, and from the VMS scheme just one sensor node is valid and the fused measurement $z_{fusion}(5)$ can be achieved. Thus the proposed tracking filter proceeds the update step II and the updated $3\sigma$ circle and the standard deviation of the position error is decreased. The proposed tracking filter can successfully track the target using 2 kinds of update steps recursively.

estimate the states of the target, we have proposed the energy efficient tracking algorithm using the WuS and VMS scheme for a distributed target tracking, and proposed approach have saved the energy by reducing the communication burden. And simulation results have demonstrated that, the proposed approach could guarantee the reliability and tracking accuracy. As the future work, we will apply the proposed distributed target tracking algorithm to various dynamic model of the target, such as constant acceleration(CA) model, Song model, Berg model, etc., and extend to multiple target tracking problem.

ACKNOWLEDGEMENT

This work was supported by Defense Acquisition Program Administration and Agency for Defense Development under the contract UD070054AD.

REFERENCES