



An active beacon-based leader vehicle tracking system

Stanislaw Goll^{1,2}, Elena Zakharova^{1,2}

¹ Department of Information-Measuring and Biomedical Engineering, Ryazan State Radio Engineering University (RSREU), Gagarin Street, 59/1, Ryazan, Russia

² LLC KB Avrora, Skomoroshinskaja Street 9, of 3, Ryazan, Russia

ABSTRACT

This article focuses on mobile robot convoying along a path travelled by a certain leader carrying the active ultrasonic beacon. The robot is equipped with the three-dimensional receiver array in order to receive both the ultrasonic wave and the RF wave marking the beginning of the measurement cycle. To increase measurement reliability, each receiver contains two independent measurement channels with automatic gain control. The distance measurements are pre-processed in order to identify the artefacts and then either remove them or replace them with the interpolated value. To estimate the position of the beacon in the robot's local coordinate system, several methods are used, including the least squares method with subsequent exponential smoothing, the linear Kalman filter, the Rauch-Tung-Striebel smoother, the extended Kalman Filter, the unscented Kalman filter, and the particle filter. The experiments were undertaken in order to estimate the estimation method preferable for following the leader's path.

Section: RESEARCH PAPER

Keywords: ultrasound; mobile robot convoying; ultrasonic beacon; Kalman filter; Rauch-Tung-Striebel smoother

Citation: Stanislaw Goll, Elena Zakharova, An active beacon-based leader vehicle tracking system, Acta IMEKO, vol. 8, no. 4, article 7, December 2019, identifier: IMEKO-ACTA-08 (2019)-04-07

Editor: Yvan Baudoin, International CBRNE Institute, Belgium

Received November 23, 2018; **In final form** June 26, 2019; **Published** December 2019

Copyright: This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Corresponding author: Elena Zakharova, e-mail: zaharova@kb-avrora.ru

1. INTRODUCTION

The convoying scenario is one of the most important tasks in modern robotics. A mobile robot autonomously following a leader can be widely used in such areas as agriculture, transportation, and the military. Many well-known international trials, such as ELROB, include this scenario either as an independent one or as a part of a more complex scenario.

The scenario is focused on the mobile robot autonomously following some leader, which can be either a human operator or another vehicle – either autonomous or remote-controlled. To detect the leader, different systems can be used, including GNSS systems [1], [2], video and infrared cameras [3]-[7], LiDARs [8]-[10], directional antennas [11], radars, ultrasonic rangefinders [12]-[14], and their combinations [15], [16]. The nature of these systems, however, can impose various restrictions on the conditions of their use. GNSS systems perform poorly in urban areas and inside buildings. LiDARs and radars can be used to measure the distance between the vehicle and the surrounding objects, but it is not an easy task to detect the leader based on range data. Cameras, both video and infrared, can help with leader detection but depend greatly on environmental conditions. Moreover, if the obstacle appears

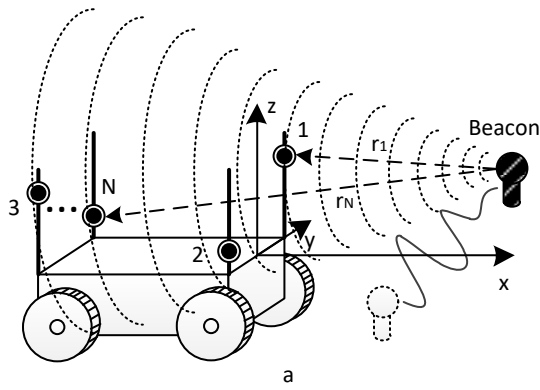
between the leader and the robot, the accuracy of their relative positions' estimate can be severely decreased.

The mathematical methods used for localisation, obstacle detection, and occupancy grid mapping are not reliable enough to build a robust convoying system. Such methods usually require a redundant sensor data of a different nature. The leader detection suffers from the same constraints and, to make things even more difficult, the rough weather conditions, dense vegetation, smoke, and other factors should also be considered.

In section 2, we discuss the implementation of an ultrasonic-based leader detection system that consists of an active beacon carried by the leader and a set of several ultrasonic receivers mounted on the convoyed robot. In section 3, different methods of computing the estimate of the beacon's position are described. Conclusions of this study, experimental results, and dynamic errors for all methods are illustrated in section 4. Finally, recommendations for future work are given in the last section.

2. THE APPROACH

In this article, we propose an ultrasonic-based leader detection system that includes an active beacon carried by the



b

Figure 1. (a) The active beacon and the receiver array mounted on the mobile robot (b) Conveying scenario with the two robots.

leader and a set of N ultrasonic receivers mounted on the conveyed robot (Figure 1).

The active beacon transmits the ultrasonic waves with constant time intervals between the waves. At the same time that the ultrasonic wave is sent, the radio frequency wave marking the beginning of the measurement cycle is also transmitted. For the each of the N receivers, the time interval between the moment of the radio frequency wave and ultrasonic wave arrivals is measured according to the time-of-flight principle. These intervals are proportional to the distances between the beacon and the respective receivers and can be calculated as $r_n = c\tau_n$, $n=1, \dots, N$, where c is the speed of the ultrasound in the air, and τ_n is the time interval measured for the n^{th} receiver.

To estimate the beacon's coordinates in the robot's local coordinate system, the system of N nonlinear equations is solved.

$$r_n^2 = (x^m - x_n)^2 + (y^m - y_n)^2 + (z^m - z_n)^2, \quad (1)$$

where $n=1, \dots, N$, $\vec{u} = [x^m \ y^m \ z^m]^T$ is a vector containing the coordinates of the beacon, and $[x_n \ y_n \ z_n]^T$ are the coordinates of the n^{th} receiver in the conveyed robot's local coordinate system.

During the system operation, an obstacle can cause line-of-sight loss between one or several receivers and the beacon. Moreover, the ultrasonic wave can be reflected by the surrounding objects, causing the problem of multipath propagation. For these reasons, acquired measurements can contain artefact distances r_n (Figure 2(a)), which are identified using the threshold constant p . The distance measurement $r_{n,k}$ is considered an artefact if the following condition is met:

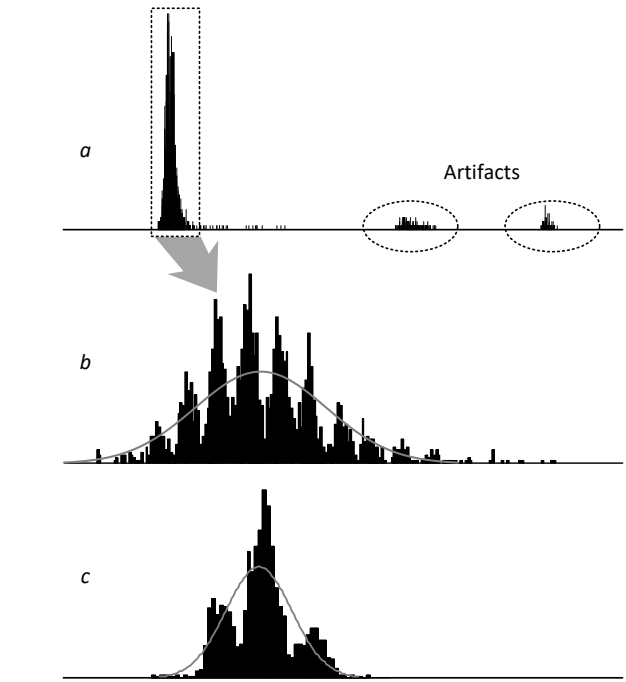


Figure 2. Histogram of centered values for the single receiver's measurements: a – measurement series containing artifacts, b – 11 meter distance between the beacon and the receiver, c – 3 meter distance between the beacon and the receiver.

$$|r_{n,k} - r_{n,k-1}| > p, \quad (2)$$

where p is an adjustable threshold constant, and k is the consequent measurement number.

If the artefact is detected, the linear least squares extrapolation is used to calculate the substitute estimate based on the last V estimates $\{r_{n,k-1}, \dots, r_{n,k-V}\}$.

If the series of M artefacts is detected in the receiver's measurements, or the substitute estimate cannot be calculated due to the unreliable measurements present in the last V estimates, the corresponding equations are excluded from the system (1). When the reliable measurements (i.e. condition (2) is not met for the two consecutive measurements) arrive, the receiver is included back in the system (1), and the artefact removal procedure becomes applicable again. The system must contain at least three reliable measurements from different receivers to calculate the beacon's position estimate \vec{u} . If there are not enough reliable measurements available, the robot stops until the beacon's position can be calculated again.

To make more robust measurements, each receiver contains two independent measurement channels with partly overlapping beam patterns and automatic gain control. If both channels succeeded at the distance measurement at some moment of time k , then the average value is used as a resulting measurement. If one of the channels failed to provide a measurement, then the over channel's measurement is used as a resulting measurement.

3. POSITION ESTIMATION

Many of the effective estimation methods are based on the linear models and use normally distributed values. Figure 2b and 2c show the histograms of centred values for the measured distances r_n between the beacon and the n^{th} receiver with the

beacon positioned along the axis of the receiver's beam pattern at a distance of 11 and 3 metres respectively. Clearly, the histograms are multimodal, with the constant distances $c \cdot \Delta\tau$ between the modes, where $\Delta\tau$ is the period of ultrasound. Since the envelope's shape can be approximated with the Gaussian function, we can assume that the distances r_n are distributed normally, with the variance depending on the beacon's position.

We can use a linear combination of the equations to transform the nonlinear equations of (1) into the linear ones [17]. For each pair of receivers, the difference of the squared distances to the beacon is calculated as follows:

$$r_i^2 - r_j^2 = (x^m - x_i)^2 + (y^m - y_i)^2 + (z^m - z_i)^2 - (x^m - x_j)^2 - (y^m - y_j)^2 - (z^m - z_j)^2,$$

where $i, j = 1, \dots, N$, $i \neq j$, C_N^2 - binomial factor, $L = C_N^2$ - the number of receiver combinations.

As a result, system (1) can be written in the form of the linear system

$$B\vec{u} = \vec{g}, \quad (3)$$

where B is the system matrix, with L rows containing $\begin{bmatrix} 2(x_j - x_i) & 2(y_j - y_i) & 2(z_j - z_i) \end{bmatrix}$, and \vec{g} is the column vector of L constant terms $(r_i^2 - r_j^2 + x_j^2 - x_i^2 + y_j^2 - y_i^2 + z_j^2 - z_i^2)$, $i, j = 1, \dots, N$, $i \neq j$.

Generally, the system (3) is inconsistent due to the measurement noise. Hence, to estimate the position of the beacon, the least squares method can be used:

$$\vec{u} = (B^T B)^{-1} B^T \vec{g}.$$

To compute the estimate \vec{u} of the beacon's position, different methods can be used; therefore, several estimates were computed in the course of this research. During the research, the receiver array of $N = 4$ receivers was used; therefore, the system (3) contains $L = C_4^2 = 6$ equations.

3.1. Least squares method with exponential smoothing

$$\vec{u}_k^s = \alpha \vec{u}_k + (1 - \alpha) \vec{u}_{k-1}^s, \quad \alpha \in (0; 1), \quad (4)$$

where α is the adjustable smoothing factor.

3.2. Kalman filter with the non-stationary measurement noise matrix

The overall filter design is undertaken according to [18]. The stochastic system model can be described as follows:

$$\begin{cases} \vec{x}_{k+1} = A\vec{x}_k + \vec{\xi}_k, \\ \vec{g}_k = C\vec{x}_k + \vec{\eta}_k, \end{cases} \quad (5)$$

where $\vec{x}_k = [x_k^m \ y_k^m \ z_k^m \ v_k^x \ v_k^y \ v_k^z]^T$ is the state vector containing the beacon's coordinates and velocity components;

$$A = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ is the stationary transition}$$

matrix of the dynamic model, where Δt is the time interval between the consecutive measurements;

$\vec{g}_k = [g_{1,k} \ \dots \ g_{L,k}]^T$ is the measurement vector;

C is the stationary measurement model, with L rows $\begin{bmatrix} 2(x_j - x_i) & 2(y_j - y_i) & 2(z_j - z_i) & 0 & 0 & 0 \end{bmatrix}$,

$i, j = 1, \dots, N$, $i \neq j$;

$\vec{\xi}_k = [0 \ 0 \ 0 \ a_k^x \ a_k^y \ a_k^z]$ is process noise,

$\vec{\xi}_k \sim \mathcal{N}(\vec{\xi}_k | \vec{0}, Q)$;

$Q = \text{diag}(0, 0, 0, \sigma_a^2, \sigma_a^2, \sigma_a^2)$ is the stationary process noise covariance matrix;

P_k is the covariance matrix of the state;

$\vec{\eta}_k$ is the observation noise vector, $\vec{\eta}_k \sim \mathcal{N}(\vec{\eta}_k | \vec{0}, R_k)$;

R_k is the non-stationary measurement noise covariance matrix. To estimate the measurement noise covariance [19], we assume that

$$E[\vec{s}_k \vec{s}_k^T] = E[\vec{\eta}_k \vec{\eta}_k^T] - CP_k C^T,$$

where $\vec{s}_k = \vec{g}_k - C\vec{x}_k$ is residual, when

$$R_k = \begin{cases} R_0, & k < D, \\ \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + CP_k C^T, & k \geq D, \end{cases}$$

D is the estimation window size.

The initial state vector is assumed to be

$$\vec{x}_0 = [x_0^m \ y_0^m \ z_0^m \ 0 \ 0 \ 0]^T,$$

```

1  input:   $\vec{x}_0, P_0, R_0, A, Q, C, D$ 
2  output:  $\vec{x}_k, P_k$ 
3  begin
4  |   for  $k = 1 \rightarrow K$ 
5  |   |   Prediction
6  |   |    $\vec{x}_{k|k-1} = A\vec{x}_{k-1}$ 
7  |   |    $P_{k|k-1} = AP_{k-1}A^T + Q$ 
8  |   |   Update
9  |   |    $\vec{s}_{k|k-1} = \vec{g}_k - C\vec{x}_{k|k-1}$ 
10 |   |    $G_k = P_{k|k-1}C^T (R_k + CP_{k|k-1}C^T)^{-1}$ 
11 |   |    $\vec{x}_k = \vec{x}_{k|k-1} + G_k \vec{s}_{k|k-1}$ 
12 |   |    $P_k = (I - G_k C) P_{k|k-1}$ 
13 |   |    $\vec{s}_k = \vec{g}_k - C\vec{x}_k$ 
14 |   |   if  $k < D$  then
15 |   |   |    $R_{k+1} = R_0$ 
16 |   |   |   end
17 |   |   else
18 |   |   |    $R_{k+1} = \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + CP_k C^T$ 
19 |   |   |   end
20 |   |   end
21 end

```

where $\begin{bmatrix} x_0^m & y_0^m & z_0^m \end{bmatrix}^T = (B^T B)^{-1} B^T \vec{g}_0$.

The posteriori error covariance matrix and the measurement noise covariance matrix are written as $P_0 = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2)$ and

$R_0 = \text{diag}(\underbrace{\sigma_g^2, \dots, \sigma_g^2}_L)$ respectively.

3.3. Rauch-Tung-Striebel smoother

The sequence of the beacon's state estimates $\{\vec{x}_k, \vec{x}_{k-1}, \dots, \vec{x}_{k-T}\}$ can be used to set the desired path for the mobile robot. If the corresponding covariance matrices $\{P_k, P_{k-1}, \dots, P_{k-T}\}$ are also known, then this path-to-be can be smoothed using the Rauch-Tung-Striebel smoother. The estimate and covariance sequences are rewritten as $\{\vec{x}_T, \vec{x}_{T-1}, \dots, \vec{x}_0\}$ and $\{P_T, P_{T-1}, \dots, P_0\}$ accordingly. The initial smoothed estimate is $\vec{x}_T^s = \vec{x}_T$ with the covariance $P_T^s = P_T$. Then the smoother is applied from the last time step to the first (i.e., $t = T-1, \dots, 0$) according to the description given in [18].

```

1 input:  $\{\vec{x}_T, \vec{x}_{T-1}, \dots, \vec{x}_0\}, \{P_T, P_{T-1}, \dots, P_0\}, A, Q, T$ 
2 output:  $\{\vec{x}_T^s, \vec{x}_{T-1}^s, \dots, \vec{x}_0^s\}, \{P_T^s, P_{T-1}^s, \dots, P_0^s\}$ 
3 begin
4   for  $t = T \rightarrow 0$ 
5      $\vec{x}_{t+1|t} = A\vec{x}_t$ 
6      $P_{t+1|t} = AP_t A^T + Q$ 
7      $S_t = P_t A^T P_{t+1|t}^{-1}$ 
8      $\vec{x}_t^s = \vec{x}_t + S_t (\vec{x}_{t+1}^s - \vec{x}_{t+1|t})$ 
9      $P_t^s = P_t + S_t (P_{t+1}^s - P_{t+1|t}) S_t^T$ 
10  end
11 end

```

3.4. Extended Kalman filter

According to [18], the stochastic system model can be described as follows:

$$\begin{cases} \vec{x}_{k+1} = A\vec{x}_k + \vec{\xi}_k, \\ \vec{r}_k = \vec{h}(\vec{x}_k) + \vec{\eta}_k, \end{cases} \quad (6)$$

$\vec{h}(\vec{x}_k)$ is the vector-valued function containing N elements

$$\sqrt{(x_k^m - x_n)^2 + (y_k^m - y_n)^2 + (z_k^m - z_n)^2}, \quad n=1, \dots, N;$$

$\vec{r}_k = [r_{1,k} \ \dots \ r_{N,k}]^T$ is the measurement vector;

P_k is the covariance matrix of the state;

R_k is the non-stationary measurement noise covariance matrix. To estimate the measurement noise covariance [20], we assume that

$$E[\vec{s}_k \vec{s}_k^T] = E[\vec{\eta}_k \vec{\eta}_k^T] - H_k P_k H_k^T, \quad \text{where } \vec{s}_k = \vec{r}_k - \vec{h}(\vec{x}_k) \text{ is residual, when}$$

$$R_k = \begin{cases} R_0, & k < D, \\ \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + H_k P_k H_k^T, & k \geq D, \end{cases} \quad (7)$$

D is the estimation window size,

$H_k = d\vec{h}/d\vec{x}_{k|k-1}$ is the Jacobian matrix with N rows

$$\begin{bmatrix} (x_{k|k-1}^m - x_n) \\ \sqrt{(x_{k|k-1}^m - x_n)^2 + (y_{k|k-1}^m - y_n)^2 + (z_{k|k-1}^m - z_n)^2} \\ (y_{k|k-1}^m - y_n) \\ \sqrt{(x_{k|k-1}^m - x_n)^2 + (y_{k|k-1}^m - y_n)^2 + (z_{k|k-1}^m - z_n)^2} \\ (z_{k|k-1}^m - z_n) \\ \sqrt{(x_{k|k-1}^m - x_n)^2 + (y_{k|k-1}^m - y_n)^2 + (z_{k|k-1}^m - z_n)^2} \\ 0 \\ 0 \\ 0 \end{bmatrix}^T, \quad n=1, \dots, N.$$

On the initial step $\vec{x}_0 = [x_0^m \ y_0^m \ z_0^m \ 0 \ 0 \ 0]^T$,

where $\begin{bmatrix} x_0^m & y_0^m & z_0^m \end{bmatrix}^T = (B^T B)^{-1} B^T \vec{g}_0$;

$$P_0 = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2);$$

$$R_0 = \text{diag}\left(\underbrace{\sigma_r^2, \dots, \sigma_r^2}_N\right) \quad (8)$$

```

1 input:  $\vec{x}_0, P_0, R_0, A, Q, \vec{b}, D$ 
2 output:  $\vec{x}_k, P_k$ 
3 begin
4   for  $k=1 \rightarrow K$ 
5     Prediction
6      $\vec{x}_{k|k-1} = A\vec{x}_{k-1}$ 
7      $P_{k|k-1} = AP_{k-1} A^T + Q$ 
8     Update
9      $\vec{s}_{k|k-1} = \vec{r}_k - \vec{b}(\vec{x}_{k|k-1})$ 
10     $H_k = d\vec{b}/d\vec{x}_{k|k-1}$ 
11     $G_k = P_{k|k-1} H_k^T (R_k + H_k P_{k|k-1} H_k^T)^{-1}$ 
12     $\vec{x}_k = \vec{x}_{k|k-1} + G_k \vec{s}_{k|k-1}$ 
13     $P_k = (I - G_k H_k) P_{k|k-1}$ 
14     $\vec{s}_k = \vec{r}_k - \vec{b}(\vec{x}_k)$ 
15    if  $k < D$  then
16       $R_{k+1} = R_0$ 
17    end
18    else
19       $R_{k+1} = \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + H_k P_k H_k^T$ 
20    end

```

3.5. Unscented Kalman filter

In the initial step, the set of sigma points is computed as

$$X_0 = \left[\underbrace{\bar{x}_0 \dots \bar{x}_0}_{2U+1} \right] + \sqrt{U+\lambda} \begin{bmatrix} \bar{0} \\ \sqrt{P_0} \\ -\sqrt{P_0} \end{bmatrix}, \text{ where}$$

$$\bar{x}_0 = \begin{bmatrix} x_0^m & y_0^m & z_0^m & 0 & 0 & 0 \end{bmatrix}^T, \quad (9)$$

$$\begin{bmatrix} x_0^m & y_0^m & z_0^m \end{bmatrix}^T = (B^T B)^{-1} B^T \bar{g}_0.$$

Each sigma point has its own weight, calculated by the formulas

$$W = \left(I - \begin{bmatrix} \bar{w}^m \\ \dots \\ \bar{w}^m \end{bmatrix} \right) \cdot \text{diag}(w_0^c, \dots, w_{2U}^c) \times \left(I - \begin{bmatrix} \bar{w}^m \\ \dots \\ \bar{w}^m \end{bmatrix} \right)^T, \quad (10)$$

```

1 input:  $X_0, P_0, R_0, A, Q, W, \lambda, U, D, \bar{b}$ 
2 output:  $\bar{x}_k, P_k$ 
3 begin
4   for  $k=1 \rightarrow K$ 
5     Prediction
6      $X_k = AX_{k-1}$ 
7      $\bar{x}_{k|k-1} = X_k \cdot \bar{w}^m$ 
8      $P_{k|k-1} = X_k W X_k^T + Q$ 
9     Update
10     $X_{k|k-1} = [\bar{x}_{k|k-1} \dots \bar{x}_{k|k-1}] +$ 
11     $\sqrt{U+\lambda} [\bar{0} \dots \sqrt{P_{k|k-1}} \dots -\sqrt{P_{k|k-1}}]$ 
12     $Y_{k|k-1} = \bar{b}(X_{k|k-1})$ 
13     $\bar{y}_{k|k-1} = \bar{r}_k - Y_{k|k-1} \cdot \bar{w}^m$ 
14     $G_k = X_{k|k-1} W Y_{k|k-1}^T (Y_{k|k-1} W Y_{k|k-1}^T + R_k)^{-1}$ 
15     $\bar{x}_k = \bar{x}_{k|k-1} + G_k \bar{y}_{k|k-1}$ 
16     $P_k = P_{k|k-1} - G_k (Y_{k|k-1} W Y_{k|k-1}^T + R_k) G_k^T$ 
17     $X_k = [\bar{x}_k \dots \bar{x}_k] +$ 
18     $\sqrt{U+\lambda} [\bar{0} \dots \sqrt{P_k} \dots -\sqrt{P_k}]$ 
19     $\bar{y}_k = \bar{r}_k - Y_k \cdot \bar{w}^m$ 
20    if  $k < D$  then
21       $R_{k+1} = R_0$ 
22    end
23    else
24       $R_{k+1} = \frac{1}{D} \sum_{d=k-D}^k \bar{y}_d \bar{y}_d^T + Y_k W Y_k^T$ 
25    end
26  end

```

$$\text{where } \bar{w}^m = [w_0^m \dots w_{2U}^m]^T, w_0^m = \frac{\lambda}{U+\lambda},$$

$$w_0^c = \frac{\lambda}{U+\lambda} + 1 - \alpha^2 + \beta, w_i^m = w_i^c = \frac{1}{2(U+\lambda)}, i=1, \dots, 2U;$$

$U=6$ refers to the number of components in the state vector \bar{x}_k ;

$\lambda = \alpha^2 (U + \kappa) - U$ is a scaling parameter;

α, κ, β are the parameters of the method;

R_k is the non-stationary measurement noise covariance matrix. To estimate the measurement noise covariance [21], we assume that

$$E[\bar{y}_k \bar{y}_k^T] = E[\bar{\eta}_k \bar{\eta}_k^T] - Y_k W Y_k^T, \text{ where } \bar{y}_k = \bar{r}_k - Y_k \cdot \bar{w}^m \text{ is residual,}$$

$$R_k = \begin{cases} R_0, & k < D, \\ \frac{1}{D} \sum_{d=k-D}^k \bar{y}_d \bar{y}_d^T + Y_k W Y_k^T, & k \geq D, \end{cases} \quad (11)$$

D is the estimation window size,

$Y_k = \bar{b}(X_k)$, X_k is the array of sigma points.

The initial value of $P_0 = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2)$,

$$R_0 = \text{diag} \left(\underbrace{\sigma_r^2, \dots, \sigma_r^2}_N \right).$$

The filter implementation complies with [18].

3.6. Particle filter

To describe the beacon's position, the following state vector

$$\bar{x} = [x^m \ y^m \ z^m \ v^x \ v^y \ v^z]^T \text{ is used, where}$$

$$\bar{x} \sim \mathbb{N}(\bar{x} | \bar{x}_0, \Sigma).$$

$$\text{The initial estimate is } \bar{x}_0 = [x_0^m \ y_0^m \ z_0^m \ 0 \ 0 \ 0]^T,$$

$$\text{where } \bar{u}_0 = [x_0^m \ y_0^m \ z_0^m]^T = (B^T B)^{-1} B^T \bar{g}_0,$$

$$\Sigma = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2).$$

Based on the distribution $\mathbb{N}(\bar{x} | \bar{x}_0, \Sigma)_1$, the set of particles X_0 is generated. For each particle $\bar{x}_{j,k}$, the weight $w_{j,0} = 1/J$

is set, with the sum of the weights being $w_\Sigma = \sum_{j=1}^J w_{j,0} = 1$,

$$j=1, \dots, J.$$

```

1 input:  $X_0, A, \beta, N, J, \sigma, \{x_1 \dots x_N\}, \{y_1 \dots y_N\}, \{\tilde{x}_1 \dots \tilde{x}_N\}$ 
2 output:  $\tilde{x}_k$ 
3 begin
4   for  $k=1 \rightarrow K$ 
5      $X_k = AX_{k-1}$ 
6     for  $n=1 \rightarrow N$ 
7       for  $j=1 \rightarrow J$ 
8          $\hat{r}_{n,j,k} = \sqrt{(x_{j,k}^m - x_n)^2 + (y_{j,k}^m - y_n)^2 + (\tilde{x}_{j,k}^m - \tilde{x}_n)^2}$ 
9       end
10    end
11    for  $j=1 \rightarrow J$ 
12       $w_{j,k} = \mathbb{N}(\hat{r}_{1,j,k} | r_{1,j,k}, \sigma) \dots \mathbb{N}(\hat{r}_{N,j,k} | r_{N,j,k}, \sigma)$ 
13    end
14    The particles with the weighs less than  $\beta$  are removed
15     $X_{old} = \{\tilde{x}_{j,k} | w_{j,k} > \beta\}, j=1, \dots, J$ 
16     $\bar{w}_k = \frac{\bar{w}_k}{\sum_{j=1}^{\Omega} w_{j,k}}$ 
17     $\tilde{x}_k = \sum_{j=1}^{\Omega} w_{j,k} \tilde{x}_{j,k}$ 
18    where  $\Omega$  – is the size of the new particle set
19    New particles are added to the set according to the
20    normal distribution  $\mathbb{N}(\tilde{x} | \tilde{x}_k, \Sigma_x)$ 
21     $X_{new} = \{\tilde{x}_{j,k} | \mathbb{N}(\tilde{x} | \tilde{x}_k, \Sigma_x)\}, j=1, \dots, J - \Omega$ 
22     $X_k = X_{old} + X_{new}$ 
23  end
24 end

```

4. RESULTS

During the experiments, an array of $N = 4$ receivers was statically placed, while a mobile robot carrying the active beacon was moved in front of it following a rectangular path (Figure 3) with a width of 4 m and a length of 7 m (1st experiment) or 10.5 m (2nd experiment). The robot moved clockwise, starting and finishing at the coordinates $(-2; 1)$ in the array's local coordinate system.

For each beacon's position, we computed a total of six estimates using the methods described above. The number of particles in the particle filter is $J = 5000$. To compare the performance of these methods, we calculated the average of the Root Mean Square Errors (RMSE).

Dynamic errors in the form of distances between the estimate of the active beacon trajectory and the exemplary trajectory in the two-dimensional $|\Delta_2|$ and three-dimensional coordinate systems $|\Delta_3|$ in time are presented in Figure 3.

The grey area on the dynamic error graphs marks the parts of the active beacon path corresponding to the smallest sides of the rectangle. On the left-hand side, the path estimates and their dynamic errors are shown, which are calculated based on the shifted differences of the square distances between the beacon and each of the receivers g_n . On the right-hand side, the path estimates and their dynamic errors are shown, which

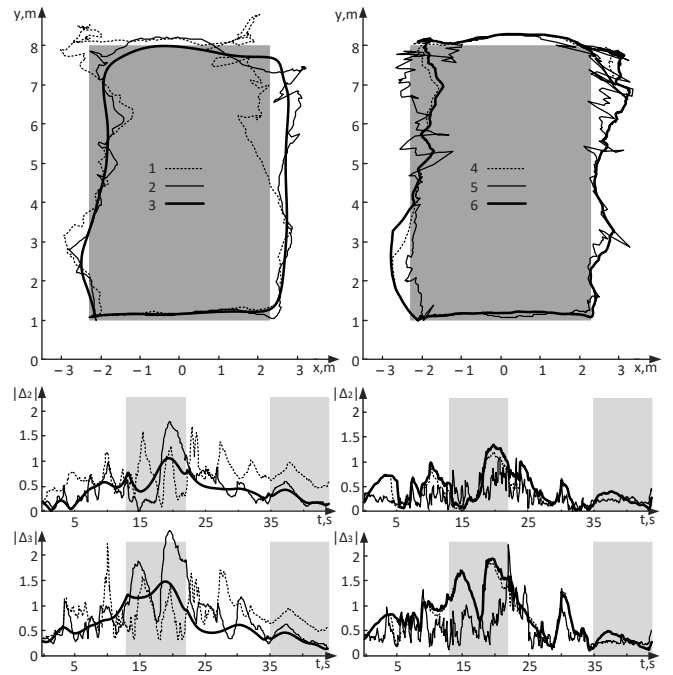


Figure 3. Path estimates and the corresponding dynamic errors in the 1st experiment: 1. Least squares method with exponential smoothing. 2. Kalman filter. 3. Kalman filter with Rauch-Tung-Striebel smoother. 4. Unscented Kalman filter. 5. Particle filter. 6. Extended Kalman filter.

are calculated based on the distances between the beacon and each of the receivers r_n .

Table 1 shows the RMSEs of the beacon's position estimates during the first experiment. Errors $\text{RMSE}(|\Delta_2|)$ and $\text{RMSE}(|\Delta_3|)$ correspond to the estimates made in the two-dimensional and three-dimensional coordinate systems respectively.

The Rauch-Tung-Striebel smoother outperforms other considered methods in terms of RSMEs and dynamic errors. However, the path estimates shown in Figure 3 appear to be more parallelogrammatic than rectangular, altering the shape of the original path. These alterations were caused by the side wind during the experiment.

During the second experiment, no wind was present. Furthermore, the length of the ground truth rectangle was increased to 10.5 m. The path estimates and the corresponding dynamic errors are shown in Figure 4. The RSMEs are shown in Table 2.

Clearly, the Rauch-Tung-Striebel smoother demonstrates the best estimate again, despite changes in the experimental conditions.

Table 1. The root mean square errors of the beacon's position estimates during the first experiment

	RMSE($ \Delta_2 $), m	RMSE($ \Delta_3 $), m
ES	0.7365	0.8948
KF	0.6161	1.0029
RTS	0.4776	0.7410
EKF	0.5569	0.9255
UKF	0.4922	0.8875
PF	0.4830	1.0422

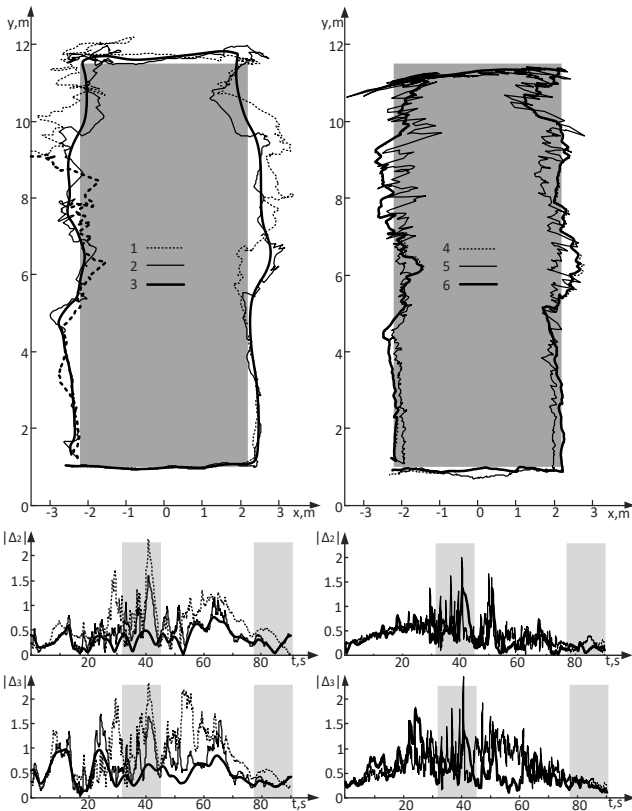


Figure 4. Path estimates and the corresponding dynamic errors in the 2nd experiment: 1 – least squares method with the exponential smoothing, 2 – Kalman Filter, 3 – Kalman Filter with Rauch-Tung-Striebel smoother, 4 – Unscented Kalman Filter, 5 – Particle Filter, 6 – Extended Kalman Filter.

Table 2. The root mean square errors of the beacon’s position estimates during the second experiment.

	RMSE(Δ ₂), m	RMSE(Δ ₃), m
ES	0.7502	1.0236
KF	0.5318	0.7447
RTS	0.3682	0.5431
EKF	0.4386	0.6350
UK	0.4471	0.6552
F		
PF	0.5005	0.6052

5. FUTURE RESEARCH

The main goal of our future research in this area is to eliminate biases caused by the wind and measurement errors during the estimation of the system’s structural parameters. To estimate these parameters, we plan to perform a calibration procedure involving the 3D LiDAR mounted on the robot. Given the 3 cm accuracy of the LiDAR measurements and usage of the machine learning algorithms, such calibration should outperform the current manual measurements made with the slide rule. Moreover, it should also allow for estimating the size and the shape of the receivers.

Currently, the effective measurement range is up to 20 m; however, it can be extended to 100 m. This can be achieved by combining the ultrasonic measurements with the GNSS-based measurements. In this case, it is possible to mitigate the relative

displacement between the robot’s and beacon’s GNSS receivers by mixing the ultrasonic and the GNSS data, while the distance between the robot and the beacon is relatively small.

REFERENCES

- [1] M. Spencer, D. Jones, M. Kraehling, K. Stol, Trajectory-based autonomous vehicle following using a robotic driver, Proc. of the Australasian Conference on Robotics and Automation, 2009, Brisbane, Australia, Australian Robotics and Automation Association, pp. 1-10.
- [2] N. Wong, K. A. Stol, C. Chambers, R. Halkyard, Autonomous vehicle following using a robotic driver, Proc. of the IEEE 15th International Conference on Mechatronics and Machine Vision in Practice, 2-4 December 2008, Auckland, New Zealand, pp.115-120.
- [3] Z. Changchen, C. Weihai, Z. Zhiwen, L. Jingmeng, An RGBD data based vehicle detection algorithm for vehicle following systems, Proc. of the 2013 8th IEEE Conference on Industrial Electronics and Applications (ICIEA), 19-21 June 2013, Melbourne, Australia, pp. 1506-1511.
- [4] R. Vidal, O. Shakernia, S. Sastry, Following the flock: distributed formation control with omnidirectional vision-based motion segmentation and visual servoing. IEEE Robotics & Automation Magazine 11(4) (2004) pp. 14-20.
- [5] J. Krejsa, S. Vechet, Infrared beacons based localization of mobile robot. Electronics and Electrical Engineering 1(117) (2012) pp. 17-22.
- [6] C. Fries, H.-J. Wuensche, Monocular template-based vehicle tracking for autonomous convoy driving, Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2014, pp. 2727-2732
- [7] L. Zhang, T. Ahamed, Y. Zhang, P. Gao, T. Takigawa, Vision-based leader vehicle trajectory tracking for multiple agricultural vehicles, Sensors 16(4) (2016) p. 578.
- [8] T. C. Ng, J. Ibanez-Guzman, J. Shen, Z. Gong, H. Wang, C. Cheng, Vehicle following with obstacle avoidance capabilities in natural environments, Proc. of the IEEE Int. Conf. Robotics and Automation ICRA '04, April 2004, New Orleans, LA, USA (5) pp. 4283-4288.
- [9] C. Fries, H.-J. Wuensche, Autonomous convoy driving by night: the vehicle tracking system, Proc. of the IEEE Conf. Technol. Pract. Robot Appl. TePRA, 2015, pp. 1-6.
- [10] M. Himmelsbach, A. Muller, T. Luttel, H.-J. Wunsche, LIDAR-based 3D object perception, Proc. of the 1st International Workshop on Cognition for Technical Systems, October 2008, Munchen, pp. 1-7.
- [11] B.-C. Min, E. T. Matson, Robotic follower system using bearing-only tracking with directional antennas, Proc. of the 2nd International Conference on Robot Intelligence Technology and Applications (RITA 2013) 18-20 December 2013, Denver, Colorado, USA, pp. 37-58.
- [12] S. Shoval, J. Borenstein, Measuring the relative position and orientation between two mobile robots with binaural sonar, Proc. of the 9th ANS International Topical Meeting on Robotics and Remote Systems, 4-8 March 2001, Seattle, Washington, USA.
- [13] F. Hoflinger, L. M. Reindl, TDOA based localization using interacting multiple model estimator and ultrasonic transmitter/receiver, Proc. of the 9th IEEE International Multi-Conference on Systems, Signals and Devices, 20-23 March 2012, Chemnitz, Germany (1) pp.1063-1069.
- [14] E. S. Ningrum, R. Hakkun, A. Alasiry, 2016, Tracking and formation control of leader-follower cooperative mobile robot based on trilateration data. International Journal of Engineering Technology EMITTER 3(2) (2016) pp. 88-98.
- [15] T. C. Ng, J. I. Guzman, M. D. Adams, 2005, Autonomous vehicle following systems: a virtual trailer link model, Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2-6 August 2005, Edmonton, Alta., Canada, pp. 3057-3062.

- [16] D. Simon, B. Theisen, Convoy active safety technology: environmental understanding and navigation with use of low cost sensors, Proc. of the NDiA Ground Vehicle Systems Engineering and Technology Symposium, 2012.
- [17] A. Montanha, A. M. Polidorio, F. J. Dominguez-Mayo, M. J. Escalona, 2D triangulation of signals source by pole-polar geometric models, *Sensors* 19(5) (2019) p. 1020.
- [18] J. Hartikainen, S. Sumo, Optimal filtering with Kalman filters and smoothers, a manual for the Matlab toolbox EKF/UKF, Department of Biomedical Engineering and Computational Science, Aalto University School of Science, Espoo, Finland, August 2011, p.129.
- [19] J. Wang, W. Ding, J. Wang, Improving adaptive Kalman filter in GPS/SDINS integration with neural network, Proc. of the ION GNSS 2007, 25-28 September 2007, Fort Worth, TX, USA (1) pp. 571-579.
- [20] S. Akhlaghi, N. Zhou, Z. Huang, Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation, arXiv preprint arXiv:1702.00884, 2017, pp.1-5.
- [21] B. Zheng, P. Fu, B. Li, X. Yuan, A robust adaptive unscented Kalman filter for nonlinear estimation with uncertain noise covariance, *Sensors* 18 (2018) p. 808.