ISSN 0005–1144
ATKAAF 45(3–4), 145–154 (2004)

Edouard Ivanjko, Ivan Petrović, Mario Vašak

Sonar-based Pose Tracking of Indoor Mobile Robots

UDK 004.896:681586.4
IFAC 4.5.6

Original scientific paper

In order to perform useful tasks the mobile robot's current pose must be accurately known. Problem of finding and tracking the mobile robot's pose is called localization, and can be global or local. In this paper we address the problem of mobile robot's local localization or pose tracking with prerequisites of known starting pose, robot kinematics and world model. Pose tracking is mostly based on odometry, which has the problem of accumulating errors in an unbounded fashion. To overcome this problem sensor fusion is commonly used. This paper describes a simple odometry calibration method and compares two fusion methods of calibrated odometry data and sonar range data fusion based on a Kalman Filter framework. One fusion method is based on the standard Extended Kalman Filter and another one, proposed in this paper, on the Unscented Kalman Filter. Occupancy grid map is used as the world model, which is beneficial because only sonars' range measurement uncertainty has to be considered. If a feature-based map is used, as the world model, then an additional uncertainty regarding the feature/range reading assignment must be also considered. Experimental results obtained with the Pioneer 2DX mobile robot (manufacturer ActivMedia Robotics) show that better accuracy of pose estimation and smoother robot motion can be obtained with Unscented Kalman Filter.

Key words: non-linear Kalman Filter, mobile robot, localization, occupancy grid map

1 INTRODUCTION

Mobile robot localization is one of the very important tasks in navigation of autonomous mobile robots [1]. In a typical indoor environment with a flat floor plan, mobile robot localization becomes a matter of estimating its pose, i.e. its position given with its x and y coordinates and its orientation given with angle Θ. There are two types of localization: global and local. Using global localization methods mobile robot can solve the unknown start pose and lost-robot problem, i.e. current pose in an a priori known world model can always be determined. On the other hand, using local localization or pose tracking the mobile robot becomes lost when predicted sensor readings become significantly different from real sensor readings. In spite of this drawback of pose tracking methods they are extensively used due to their computational simplicity, which is of particular importance in applications where all algorithms should be computed in onboard computer. A global localization algorithm should also be implemented, but its execution is started only in cases when robot is lost. In such a situation robot interrupts its normal operation and start wandering through the environment trying to find its pose. When it succeeds, execution of global localization algorithm is stopped and robot continues with normal operation.

One of the most important means of solving the pose tracking problem is odometry. This method uses data from encoders mounted on robot wheels and is a simple, inexpensive and easy way to determine the offset from a known start pose in real time. The encoders' data are passed to the central processor that in turns continually updates the mobile robot's pose using geometric equations describing robot kinematics. The disadvantage is its unbounded accumulation of errors due to wheel slippage, floor roughness, discretized sampling of wheel speed data, inaccessibility to the angular velocities of the wheels in some mobile robots etc. So this method can be successfully used only for pose tracking between pose updates using additional sensors [2].

A lot of research has been undergone in order to improve the accuracy of odometry, i.e. to eliminate the systematic error, mobile robot construction constraints and environment influences on the mobile robot pose tracking. A common approach consists of two parts. The first part is about a better error [3] or odometry model [4] and the second part involves usage of additional sensors [2]. Most often used additional sensors are sonar, laser range finder, stereo or mono vision, gyro, compass and GPS (for outdoor mobile robots). Additional sensors can be used for correction of the estimated robot pose,

for online odometry calibration [5], or for both simultaneously. In case of mobile robot pose correction local and global world model matching [6] or sensor fusion techniques are used [7, 8].

In this paper we propose an approach to odometry calibration based on two simple experiments and the fusion of the so calibrated odometry data with sonar range data. For the sake of sensors' data fusion we implemented and tested two versions of nonlinear Kalman filters: (i) standard Extended Kalman Filter (EKF) and (ii) Unscented Kalman Filter (UKF). The idea of their use is to match recent sensory information against prior knowledge of the environment, i.e. world model that is in our case an occupancy grid map. Occupancy grid maps are the dominant paradigm for environment modeling in mobile robotics and they facilitate various key aspects of mobile robot navigation, such as localization, path planning and collision avoidance. Since occupancy grid maps are used, only sonar range measurement uncertainty has to be considered, unlike feature-based maps where an additional uncertainty regarding the feature/range reading assignment must be considered. Thus the numerical complexity is reduced.

The paper is organized as follows. Section 2 presents used mobile robot kinematic model and applied offline calibration procedure. Section 3 describes EKF and UKF based mobile robot localizations. Experimental results obtained using mentioned pose estimation techniques are presented and analyzed in Section 4.

2 MOBILE ROBOT KINEMATIC MODEL AND ITS CALIBRATION

Odometry is one of the most important means for mobile robot pose tracking. This method uses encoders' data and robot kinematic model to calculate its displacement between two time instants.

Main disadvantage of this method is unbounded accumulation of errors due to wheel slippage, floor roughness, discretised sampling of wheel speed data, and inaccessibility to the angular velocities of the wheels in some mobile robots etc. Improved odometry can significantly reduce the rate of errors accumulation, and consequently increase the autonomy range of the mobile robot. This property is vital for many applications, particularly in cases when absolute information of robot pose is temporarily unavailable. Therefore, before the fusion of odometry and any other sensors it is advisable to calibrate the odometry in order to obtain the best possible accuracy of pose estimation. Moreover, the price of the system is a limiting factor in many commercial applications, where it is beneficial to avoid the use of compass/gyro, but at the same time to keep accuracy of the robot pose estimation as high as possible. Below we describe kinematic model of differential drive mobile robots and then the proposed calibration procedure.

2.1 Kinematic model off differential drive mobile robots

Differential drive mobile robots have usually three or four wheels, where two front wheels are motor driven wheels with encoder mounted on them and other wheels are castor wheels needed for robot stability. Driving wheels are controlled independently from each other. The encoders measure the speed of the wheels. The kinematics of the differential drive mobile robots is described with the following equations (Figure 1):

$$x_{k+1} = x_k + D_k \cdot \cos\Theta_{k+1}, \tag{1}$$

$$y_{k+1} = y_k + D_k \cdot \sin\Theta_{k+1}, \tag{2}$$

$$\Theta_{k+1} = \Theta_k + \Delta\Theta_k, \tag{3}$$

$$D_k = v_{t,k} \cdot T, \tag{4}$$

$$\Delta\Theta_k = \omega_k \cdot T, \tag{5}$$

$$v_{t,k} = \frac{v_{L,k} + v_{R,k}}{2} = \frac{\omega_{L,k}R + \omega_{R,k}R}{2}, \tag{6}$$

$$\omega_k = \frac{v_{R,k} - v_{L,k}}{b} = \frac{\omega_{R,k}R - \omega_{L,k}R}{b}, \tag{7}$$

where we denote with: x_k and y_k coordinates of the centre of axle [mm]; D_k, traveled distance between time steps k and $k+1$ [mm], $v_{t,k}$ mobile robot translation speed [mm/s]; T sampling time [s]; Θ_k angle between the vehicle and x-axis in degrees; $\Delta\Theta_k$, rotation angle between time steps k and $k+1$ in degrees, $v_{L,k}(k)$ and $v_{R,k}(k)$ velocities of the left and right wheel, respectively [mm/s]; $\omega_{L,k}(k)$ and $\omega_{R,k}(k)$ angular velocities of the left and right wheel, respectively [rad/s]; R radius of the two wheels [mm],

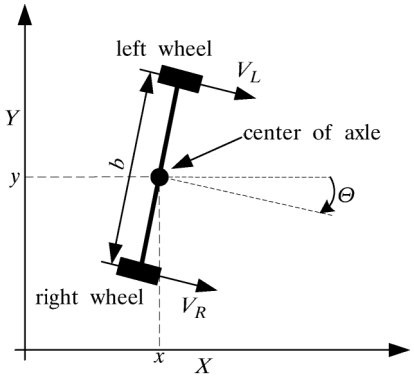

Fig. 1 The mobile robot kinematics

and b vehicle axle length [mm]. It is assumed that the wheels have equal radius. Sampling time T was 0.1 [s]. Equations (1) to (7) describe the basic odometry pose tracking model and pose tracking results obtained by simply propagating these equations through time are marked as uncalibrated odometry (UO).

2.2 Mobile robot kinematics model calibration

Experiments with differential drive mobile robots reveal that pose tracking, which relies on velocities returned by the encoders, produces very large errors in the pose estimates, especially in the orientation estimation, e.g. [2]. As it was said above, there are many factors that deteriorate accuracy of odometry based pose estimation, but the most influential ones are deviations of the wheels radiuses and axle length from their nominal values. These two factors contain the majority of the systematic errors and it is worthy to compensate their influences [2]. In order to implement these compensations we expand the equations (6) and (7) with three additional parameters:

$$v_{t,k} = \frac{k_1 \cdot v_{L,k} + k_2 \cdot v_{R,k}}{2}, \tag{8}$$

$$\omega_k = \frac{k_2 \cdot v_{R,k} - k_1 \cdot v_{L,k}}{k_3 \cdot b}, \tag{9}$$

where parameters k_1 and k_2 compensate the deviations of the wheel radius and parameter k_3 the deviation of the axle length. Thus replacing (6) and (7) with (8) and (9) defines the calibrated odometry pose tracking technique (CO), which is also used as the motion model for both Kalman filter implementations.

Proposed calibration procedure consists of two steps. First, two simple experiments with the mobile robot are performed and then collected data are used for the parameters optimization with respect to a certain criteria. The choice of the experiments is based on the facts that the parameters k_1 and k_2 affect mostly the position estimation and parameter k_3 the orientation estimation. The experiment used for parameters k_1 and k_2 calibration is a »straight-line« experiment (Figure 2). It was accomplished by providing equal speed references to the both wheels. The experiment used for parameter k_3 calibration is a »turn in place experiment« (Figure 3). The mobile robot had to turn for 180 degrees in place. During the experiments wheel speeds and sampling rate were measured and collected. Both experiments were repeated for five times to improve the accuracy of odometry calibration. These data are then used as input data for the optimization scripts.

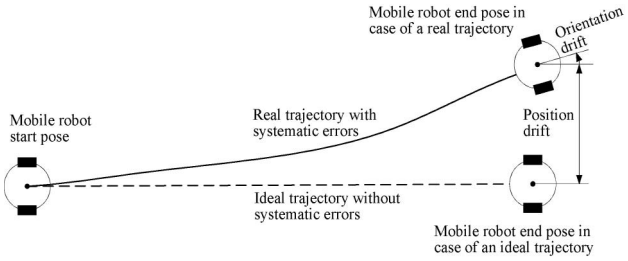

Fig. 2 The straight-line experiment

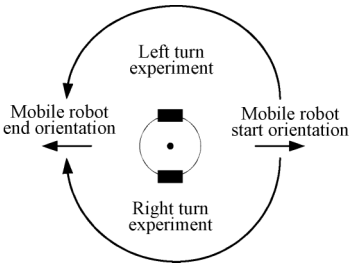

Fig. 3 The turn in place experiment

Optimal values of the compensation parameters k_1, k_2 and k_3 were calculated using the *fsolve* optimization function from the MATLAB Optimization Toolbox [9]. This optimization function uses Gauss--Newton non-linear optimization method to find the optimum for any non-linear function at hand. Optimization of parameters k_1 and k_2 were implemented as *fsolve*("*optimize_k1_k2*", *[k10, k20]*, *options*), and optimization of parameter k_3 as *fsolve*("*optimize_k3*", *[k30]*, *options*), where *k10*, *k20* and *k30* are initial values of corresponding parameters, which were set to 1.0. Pseudo-codes of scripts *optimize_k1_k2* and *optimize_k3* are given in Table 1. Both scripts use the collected data and the mobile robot model, given with equations (1)–(5) and (8)–(9), to calculate the mobile robot pose. Based on the computed mobile robot pose and actual mobile robot pose orientation error was calculated. The goal of the optimization procedure is to minimize the orientation error by adjusting the calibration parameters. This optimization criteria is chosen due to robot orientation Θ is the most significant of the localization parameters (x, y, Θ) in terms of its influence on accumulated odometry errors. The first script is used for the calculation of the parameters k_1 and k_2 from the data collected in the straight--line experiment and the second one for the calculation of the parameter k_3 from the data collected in the turn-experiment. Optimization function *fsolve* invokes these two scripts alternately. When it invokes the first script it uses new value of the parameter k_3 calculated in the previous step, and consequently when it invokes the second script it uses new values of the parameters k_1 and k_2 calculated

Table 1 **Pseudo-codes of optimization scripts**

Function: *optimize_k1_k2* (k1, k2)	**Function:** *optimize_k3* (k3)
Input: new k1 and k2 values	**Input:** new k3 value
File Input: measurement data (wheel velocities and time data, exact start and final mobile robot poses)	**File Input:** measurement data (wheel velocities and time data, exact start and final mobile robot pose)
Output: difference between exact and computed mobile robot orientation	**Output:** difference between exact and computed mobile robot orientation
load collected data of straight-line experiment	**load** collected data of turn experiment
load the parameter k3 value	**load** the parameter k1 and k2 value
compute the final pose using the expanded robot model and the measurement data	**compute** the final pose using the expanded robot model and the measurement data
return difference between exact and computed mobile robot orientation	**return** difference between exact and computed mobile robot orientation

in the previous step. As the stopping criteria for optimization procedure we set the number of iterations for the *fsolve* optimization function to 50.

3 SONAR BASED POSE TRACKING

The challenge of mobile robot pose tracking using sensor fusion is to weigh its pose (i.e. robot's state) and sonar range reading (i.e. robot's output) uncertainties to get the optimal estimate of the pose, i.e. to minimize its covariance. The Kalman filter [10] assumes the Gaussian probability distributions of the state random variable (mobile robot pose in our case) such that it is completely described with the mean and covariance. The optimal state estimate is computed in two major stages: time-update and measurement-update. In the time--update, state prediction is computed on the base of its preceding value and the control input value using the motion model. Measurement-update uses the results from time-update to compute the output predictions with the measurement model. Then the predicted state mean and covariance are corrected in the sense of minimizing the state covariance with the weighted difference between predicted and measured outputs. In succession, motion and measurement models needed for the mobile robot sensor fusion are discussed, and then EKF and UKF algorithms for mobile robot pose tracking are presented. Block diagram of our Kalman filter based pose tracking implementation is given in Figure 4.

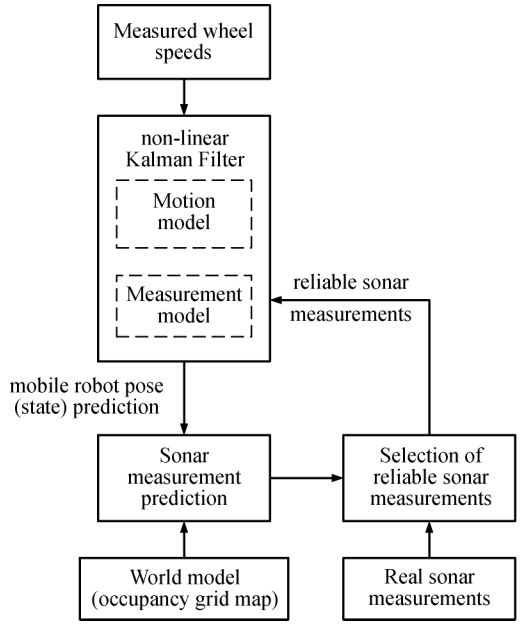

Fig. 4 Block diagram of our nonlinear Kalman filter based mobile robot pose tracking

3.1 Mobile robot motion model

The motion model represents the way in which the current state follows from the previous one. State vector is expressed as the mobile robot pose, $x_k = [x_k \; y_k \; \Theta_k]^T$, with respect to a global coordinate frame, where k denotes the sampling instant. Its distribution is assumed to be Gaussian, such that the state random variable is completely determined with a 3×3 covariance matrix $\mathbf{P}_k$ and the state expectation (mean, estimate are used as synonyms). Control input, u_k, represents the movement commands which are acted upon the mobile robot to move it from time step k to $k+1$. In the motion model control input, $u_k = [D_k \; \Delta\Theta_k]^T$, represents translation through distance D_k followed by a rotation through angle $\Delta\Theta_k$. The state transition function $f(\cdot)$ uses the state vector at the current time instant and current control input to compute the state vector at the next time instant:

$$x_{k+1} = f(x_k, \; u_k, \; v_k), \tag{10}$$

where $v_k = [v_{1,k} \; v_{2,k}]^T$ represents unpredictable process noise, that is assumed to be Gaussian with

zero mean, ($\mathrm{E}\{v_k\} = [0\ 0]^T$), and covariance Q_k. With $\mathrm{E}\{\cdot\}$ expectation function is denoted. Process noise covariance Q_k was modelled on the assumption of two independent sources of error, translational and angular, i.e. D_k and $\Delta\Theta_k$ are added with corresponding uncertainties. The expression for Q_k is:

$$Q_k = \begin{bmatrix} \sigma_D^2 & 0 \\ 0 & \Delta\Theta_k^2 \sigma_{\Delta\Theta}^2 \end{bmatrix}, \tag{11}$$

where σ_D^2 and $\sigma_{\Delta\Theta}^2$ are variances of D_k and $\Delta\Theta_k$, respectively.

3.2 Sonar range measurement model

The measurement model computes the range between an obstacle and the mobile robot's center of axle according to the measurement function [7]:

$$h_i(x, p_i) = \sqrt{(x_i - x)^2 + (y_i - y)^2}, \tag{12}$$

where $p = (x_i, y_i)$ denotes the point (occupied cell) in the world model detected by the i-th sonar. The sonar model uses (12) to relate a range reading to the obstacle that caused it:

$$z_{i,k} = h_i(x_k, p_i) + w_{i,k}, \tag{13}$$

where $w_{i,k}$ represents the measurement noise (Gaussian with zero mean and variance $r_{i,k}$) for the i-th range reading. All range readings are used in parallel, such that range measurements $z_{i,k}$ are simply stacked into a single measurement vector z_k. Measurement covariance matrix $\mathbf{R}_k$ is a diagonal matrix with the elements $r_{i,k}$. It is to be noted that the measurement noise is additive, which is very useful in the noisy sonar range measurements rejection process. Faulty range measurments are rejected by simply removing the discrepant rows from the real and predicted range measurement vectors. Size of affected vectors z_k and h_k is thus effectively reduced, which simplifies the measurement update step in applied non-linear Kalman filters.

3.3 EKF implementation

EKF is the first sensor fusion based mobile robot pose tracking technique presented in this paper. Detailed explanation of used EKF localization can be found in [11] and in the sequel only basic equations are presented. Aforementioned state transition function (10) is used to predict the new mobile robot pose when new wheel speed measurements are available. New pose uncertainty is then computed using the following equation:

$$P_k^- = \nabla f_x P_{k-1} \nabla f_x^T + \nabla f_u Q_k \nabla f_u^T, \tag{14}$$

where

$$\nabla f_x = \nabla f_x(\hat{x}_{k-1}, u_{k-1}, \mathrm{E}\{v_{k-1}\}) \tag{15}$$

is the Jacobian of $f(\cdot)$ with respect to x, and

$$\nabla f_u = \nabla f_u(\hat{x}_{k-1}, u_{k-1}, \mathrm{E}\{v_{k-1}\}) \tag{16}$$

Jacobian to $f(\cdot)$ with respect of u. If new sonar range measurements are available, they are being collected and measurement-update takes place:

$$\hat{z}_{i,k}^- = h_i(\hat{x}_k^-, p_i) + \mathrm{E}\{w_{i,k}\}. \tag{17}$$

Using new range measurements Kalman gain K is computed:

$$K_k = P_k^- \nabla h_x^T (\nabla h_x P_k^- \nabla h_x^T + R_k)^{-1}. \tag{18}$$

Corrected mobile robot pose and pose uncertainty are then updated as follows:

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - \hat{z}_k^-), \tag{19}$$

$$P_k = (I - K_k \nabla h_x) P_k^-, \tag{20}$$

where z_k are real accepted sonar range measurements,

$$\nabla h_x = \nabla h_x(\hat{x}_k, \mathrm{E}\{w_k\}) \tag{21}$$

Jacobian of the measurement function with respect to the predicted state. If no new sonar range measurements are available, corrected mobile robot pose and pose uncertainty are updated using the predicted values:

$$\hat{x}_k = \hat{x}_k^-, \tag{22}$$

$$P_k = P_k^-. \tag{23}$$

3.4 UKF based pose tracking

The second sensor fusion based mobile robot pose tracking technique presented in this paper uses UKF. It utilizes the unscented transformation that approximates the true mean and covariance of a Gaussian random variable propagated through non-linear mapping accurate to the inclusively third order of the Taylor series expansion for any mapping [12] unlike the EKF that approximates only to the first order of Taylor series. Thus a better estimation performance is expected from the UKF estimation framework.

Estimated state and uncertainty prediction is done by propagating so called pre-sigma points through the state transition function. To achieve this the state vector is augmented with means of process noise $\mathrm{E}\{v_{k-1}\}$:

$$\hat{x}_{k-1}^a = \mathrm{E}\left[x_{k-1}^a\right] = \left[\hat{x}_{k-1}^T \quad \mathrm{E}\{v_{k-1}\}^T\right]^T. \tag{24}$$

Pose uncertainty must also be augmented:

$$P_{k-1}^{a} = \begin{bmatrix} P_{k-1} & 0 \\ 0 & Q_{k-1} \end{bmatrix}. \quad (25)$$

For pre-sigma points generation pose uncertainty matrix square root or lower triangular Cholesky factorization has to be computed and scaled with factor γ:

$$\gamma = \sqrt{L+\lambda}, \quad (26)$$

where L represents the augmented state, x_{k-1}^{a}, dimension, and λ scaling parameter computed as follows:

$$\lambda = \alpha^2 (L+\kappa) - L. \quad (27)$$

Parameter α is chosen within range [10^{-4},1], and κ is usually set to 1. Pre-sigma points are now computed as follows:

$$\chi_{k-1}^{a} = \left[\hat{x}_{k-1}^{a} \quad \hat{x}_{k-1}^{a} + \gamma\sqrt{P_{k-1}^{a}} \quad \hat{x}_{k-1}^{a} - \gamma\sqrt{P_{k-1}^{a}} \right], \quad (28)$$

where $\chi_{k-1}^{a} = \left[(\chi_{k-1}^{x})^T \quad (\chi_{k-1}^{v})^T \right]^T$ represents the ma-matrix whose columns are pre-sigma points. All the obtained pre-sigma points are processed by the state transition function thus obtaining the $\chi_{k|k-1}^{x}$ matrix of predicted states (named sigma points) for each pre-sigma point:

$$\chi_{k|k-1}^{x} = f\left[\chi_{k-1}^{x}, u_{k-1}, \chi_{k-1}^{v} \right]. \quad (29)$$

Prior state mean is now computed as weighted sum of acquired points:

$$\hat{x}_{k}^{-} = \sum_{i=0}^{2L} W_i^{(m)} \chi_{i,k|k-1}^{x}, \quad (30)$$

where $\chi_{i,k|k-1}^{x}$ denotes the i-th column of $\chi_{k|k-1}^{x}$ and $W_i^{(m)}$ the weights for state mean computation obtained by the following equation:

$$W_0^{(m)} = \frac{\lambda}{L+\lambda}, \quad (31)$$

$$W_i^{(m)} = \frac{1}{2(L+\lambda)}, \quad i = 1,\ldots,2L. \quad (32)$$

Prior uncertainty matrix is now:

$$P_{k}^{-} = \sum_{i=0}^{2L} W_i^{(c)} \left[\chi_{i,k|k-1}^{x} - \hat{x}_{k}^{-} \right] \cdot \left[\chi_{i,k|k-1}^{x} - \hat{x}_{k}^{-} \right]^T, \quad (33)$$

where $W_i^{(c)}$ represents the weights for pose covariance matrix values computation:

$$W_0^{(c)} = \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta), \quad (34)$$

$$W_i^{(c)} = \frac{1}{2(L+\lambda)}, \quad i = 1,\ldots,2L. \quad (35)$$

In case of a Gaussian distribution like ours $\beta = 2$ is optimal. If new real sonar range measurements are available, sonar range measurement prediction is done for every sigma point and accepted predicted measurements are grouped in a vector:

$$\mathrm{Z}_{k|k-1} = h\left[\chi_{k|k-1}^{x}, p \right] + \mathrm{E}\{w_k\}, \quad (36)$$

where p denotes the series of points in the occupancy grid map predicted to be hit by the sonar beam. Predicted sonar range measurements are now:

$$\hat{z}_{k}^{-} = \sum_{i=0}^{2L} W_i^{(m)} \mathrm{Z}_{i,k|k-1}. \quad (37)$$

To prevent the sonar readings that hit near the corner of obstacles to negatively influence on the measurement correction, since their probability distribution cannot be approximated with Gaussian, another threshold comparison is made. These problematic sonar readings are recognized with mean $\hat{z}_{i,k}^{-}$ that differs from $z_{i,k}$ more than the acceptation thres-hold amounts, and those are being discarded. Ac-cepted sonar range measurements uncertainty becomes now:

$$P_{z_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[Z_{i,k|k-1} - \hat{z}_{k}^{-} \right] \cdot \left[Z_{i,k|k-1} - \hat{z}_{k}^{-} \right]^T, \quad (38)$$

and estimated state output covariance matrix is:

$$P_{x_k z_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\chi_{i,k|k-1}^{x} - \hat{x}_{k}^{-} \right] \cdot \left[Z_{i,k|k-1} - \hat{z}_{k}^{-} \right]^T. \quad (39)$$

Kalman gain is now:

$$K_k = P_{x_k z_k} P_{z_k}^{-1}, \quad (40)$$

and prior state covariance:

$$P_k = P_k^{-} - K_k P_{z_k} K_k^T. \quad (41)$$

Predicted state correction is done in the same way as in the EKF localization case, using Equation (19).

4 EXPERIMENTAL RESULTS

The described pose tracking techniques are tested on a Pioneer 2DX mobile robot from ActivMedia Robotics. Two series of experiments were made to evaluate described localization approaches. The first one was performed in our department corridor, and the second one in a larger room of our

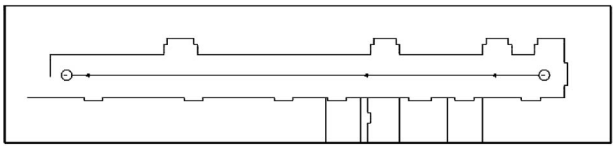

Fig. 5 Model of our department corridor used for the first pose tracking experiment

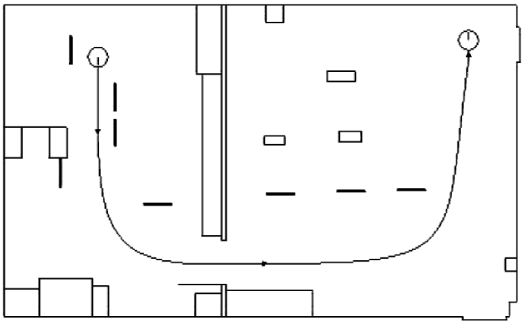

Fig. 6 Model of our department room used for the second pose tracking experiment

ment world, shown in Figure 5, is that it has little features along the axis through the corridor middle and the used occupancy grid model is accurate. The mobile robot starts in the right corridor end and travels to the left end. Main characteristic of the second one, shown in Figure 6, is that it has enough features along both axis but used occupancy grid model is less accurate because this room is full of easily movable furniture (like chairs, tables, trash baskets, etc.) which never stays for long time in the same place. In this experiment setup the mobile robot starts in the left room and has to reach the right room. These two setups enable us to test the local localization algorithm performance in an environment with little features and in a badly/incompletely-modelled environment. Traversed paths for both experiment setups (Figure 6 and 7) are presented by a line with arrows showing mobile

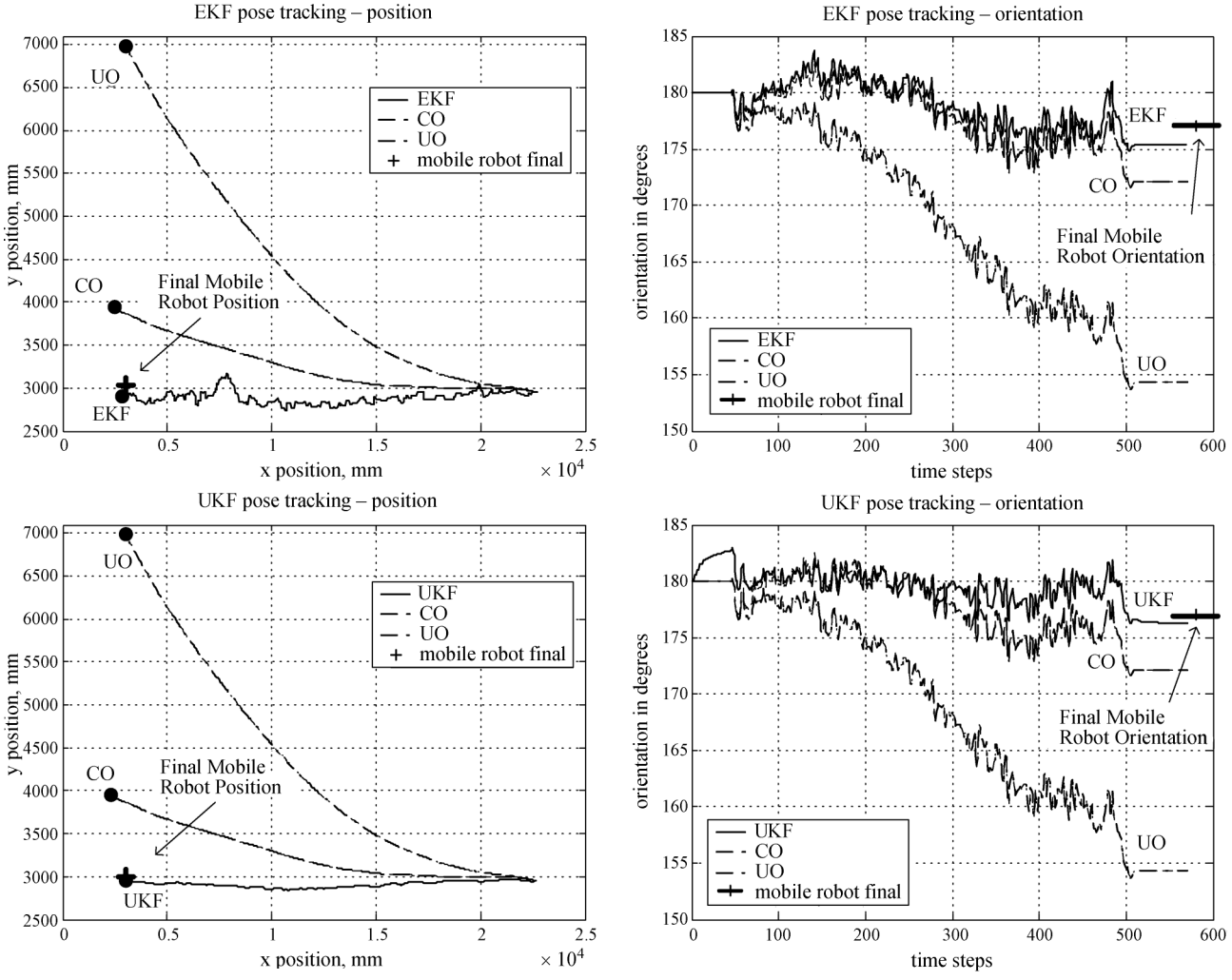

Fig. 7 Position and orientation estimation performance obtained in the first experiment

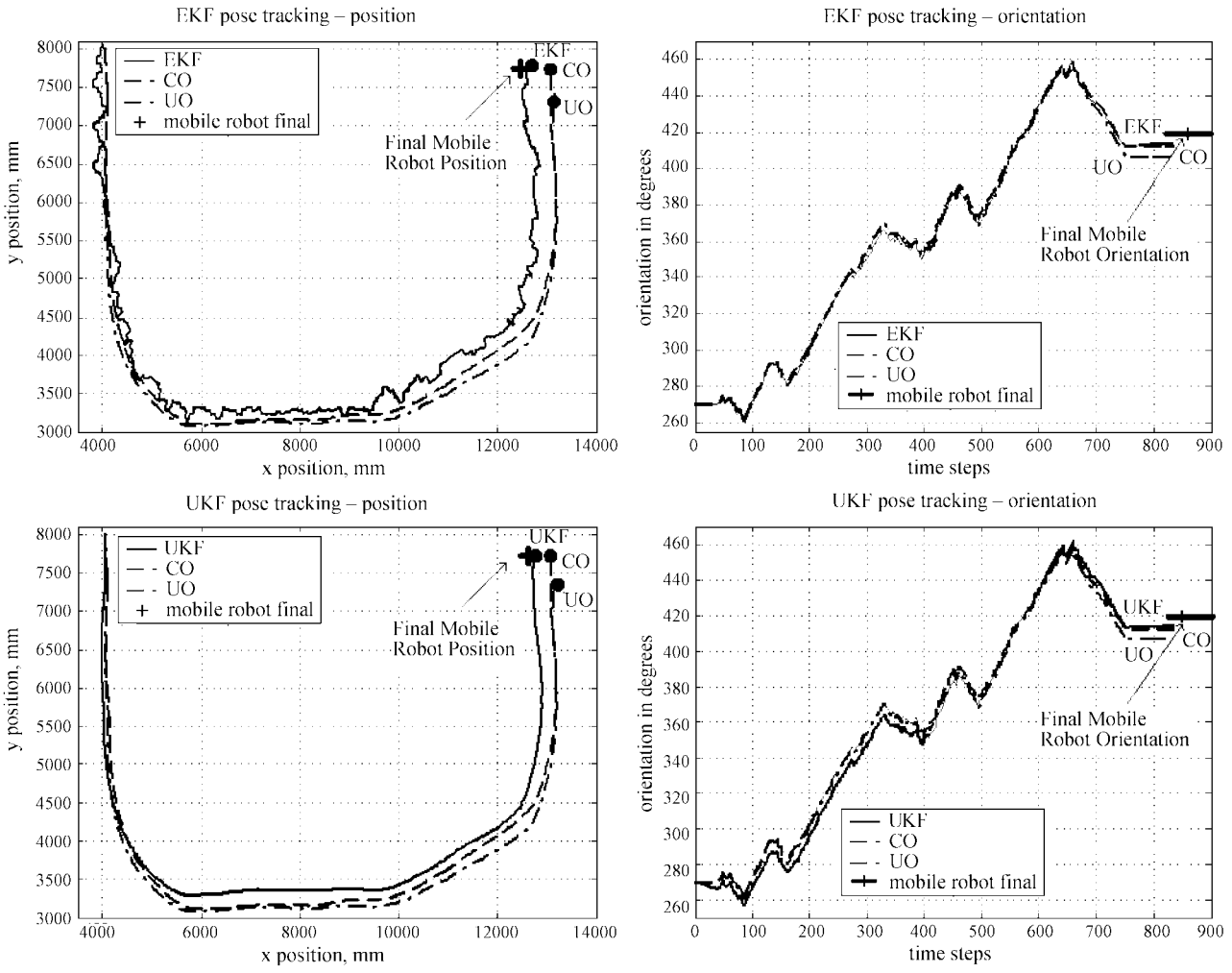

Fig. 8 Position and orientation estimation performance obtained in the second experiment

robot motion succession. A gradient navigation module [13] is used for mobile robot control, i.e. for path planning and local obstacles avoidance. Regarding navigation module's mobile robot pose input; two experiments in each of the environments were made. For navigation module pose input, EKF pose estimation is used in the first one and UKF in the second one.

New sonar range measurements z_k are available every three time steps on this mobile robot. Raw sonar data z_k and predicted readings h_k differ because the real mobile robot pose is not exactly known and due to sonar measurement noise, occlusions, specular reflections, outliers, and used occupancy grid model inaccuracy. So z_k are first compared to h_k and only those readings with difference under a certain threshold are accepted. Used threshold is set to 5 cells. That means, in a grid map with cell size of 100 [mm] $\times$ 100 [mm], as in our case, readings with difference less then 500 [mm] are accepted. Note that this procedure is repeated once more in UKF case when prediction $\hat{z}_{k,k}$ was obtained.

Obtained results are presented in Figure 7 for the first experiment setup and in Figure 8 for the second experiment setup. The left figure side presents position estimation performance, and the right side orientation estimation performance for each Kalman filter implementation separately in both experiment setups. Uncalibrated odometry and calibrated odometry pose estimation where working simultaneously with here described non-linear Kalman filter localization. Exact final mobile robot pose was manually measured at experiment end. In presented figures final mobile robot position part is denoted by a black cross, and final mobile robot orientation part by a short thick line.

Estimated final position for each tested technique is denoted by a black dot.

Position error is calculated as:

$$PositionError = \frac{Pos_{act} - Pos_{est}}{Dist} \cdot 100\,\%, \qquad (39)$$

where Pos_{act} denotes the actual final position, Pos_{est} denotes the estimated final position and *Dist* total distance traversed by the mobile robot. Orientation error is calculated as:

$$OrientationError = \frac{Orient_{act} - Orient_{est}}{Orient_{act}} \cdot 100\,\%, \quad (40)$$

where $Orient_{act}$ is actual final orientation and $Orient_{est}$ estimated final orientation. Average error is calculated as the average value between the position and orientation error. Table 2 summaries the results of all performed experiments.

Table 2 **Error comparison of different pose tracking techniques**

	EKF	UKF	CO	UO
Position error [%]	1.53	1.35	3.71	11.07
Orientation error [%]	2.08	1.39	4.72	10.71
Average error [%]	1.81	1.37	4.22	10.89

5 CONCLUSION

Two non-linear Kalman filter based local localization techniques are implemented and experimentally compared using a real mobile robot. The here presented approach uses an occupancy grid map as the world model to avoid an additional uncertainty calculation. Extended and unscented Kalman filter based mobile robot local localization techniques are implemented and they are used for calibrated odometry and sonar range measurements fusion. So the problem of unbounded odometry error increase is solved. Two implementations of the Unscented Kalman filter are described. The first one is a straightforward implementation and the second one uses the additive properties of obtained sonar range measurements. It can be seen from the presented results that the UKF localization results in a more smoother and accurate mobile robot pose estimate making that approach more suitable for navigational tasks. The EKF localization correction results in bigger correction values reacting thus faster to increased real and estimated mobile robot pose discrepancy but aggravating simultaneously the navigational tasks. UKF has a better localization performance while the EKF localization framework is somewhat computationally simpler because it predicts the sonar range measurements only for the predicted state and not like the UKF localization for every pre-sigma point. UKF approach performs better in an accurately modeled environment. Both approaches can cope with an environment with little features and a badly modeled environment. Future work on this topic will include a dual estimation of both robot pose and calibration parameters in kinematic model.

REFERENCES

[1] J. Borenstein, H. R. Everett, L. Feng, **Where am I? Sensors and Methods for Mobile Robot Positioning**. Ann Arbor, MI 48109: University of Michigan. 1996.

[2] P. Goel, S. I. Roumeliotis, G. S. Sukhatme, **Robust Localization Using Relative and Absolute Position Estimates**. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1999.

[3] J. Borenstein, L. Feng, **Measurement and Correction of Systematic Odometry Errors in Mobile Robots**. IEEE Transactions in Robotics and Automation, Vol. 12, No. 2, 1996.

[4] E. Ivanjko, I. Petrović, N. Perić, **An Approach to Odometry Calibration of Differential Drive Mobile Robots**. Proceedings of International Conference on Electrical Drives and Power Electronics EDPE'03, 2003, pp. 519–523.

[5] N. Roy, S. Thrun, **Online Self-calibration for Mobile Robots**. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 1999.

[6] K. Konolige, **Markov Localization Using Correlation**. In International Joint Conference on Artificial Intelligence, Stockholm, Sweden, July, 1999.

[7] D. Lee, **The Map-Building and Exploration Strategies of a Simple Sonar-Equipped Robot**. Cambridge University Press, 1996.

[8] G. Dudek, M. Jenkin, **Computational Principles of Mobile Robotics**. Trumpington Street, Cambridge CB2 1RP: Cambridge University Press, 2000.

[9] ..., **Optimisation Toolbox For Use with Matlab, User's Guide**. The MathWorks Inc. 2000.

[10] R. E. Kalman, **A New Approach to Linear Filtering and Prediction Problems**. Transactions of the ASME, Journal of Basic Engineering, Vol. 82, pp. 35–45, March 1960.

[11] E. Ivanjko, I. Petrović, **Extended Kalman Filter Based Mobile Robot Pose Tracking using Occupancy Grid Maps**. Proceedings of The 12th IEEE Mediterranean Electro-technical Conference – MELECON 2004, May 12–15, 2004, Dubrovnik, Croatia, pp. 311–314.

[12] S. Haykin, **Kalman Filtering and Neural Networks**. **Ch. 7. The Unscented Kalman Filter.** John Wiley and Sons, 2001.

[13] K. Konolige, **A Gradient Method for Realtime Robot Control**. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000), Kagawa University, Takamatsu, Japan, 2000.

Praćenje položaja mobilnog robota ultrazvučnim osjetilima. Mobilni robot mora u svakome trenutku znati svoj položaj, da bi mogao obavljati korisne zadaće. Problem pronalaženja i praćenja položaja mobilnog robota naziva se lokalizacijom, koja može biti globalna ili lokalna. U ovome se radu obrađuje lokalna lokalizacija, koja podrazumijeva praćenje položaja mobilnog robota uz pretpostavku da su poznati njegov početni položaj, kinematički model te model radnog prostora. Praćenje položaja se najčešće temelji na odometriji, kod koje je glavni problem neograničena akumulacija pogreške. Za rješavanje toga problema uobičajeno se koristi fuzija informacija većeg broja osjetila. Ovaj članak opisuje jednostavnu metodu kalibracije odometrije i uspoređuje dvije metode fuzije odometrijskih podataka s podacima iz ultrazvučnih osjetila (sonara) koji predstavljaju udaljenosti robota do okolnih prepreka. Primijenjene metode fuzije temeljene su na teoriji Kalmanova filtra. Jedna metoda koristi već standardni prošireni Kalmanov filtar, a druga, predložena u ovome radu, nederivacijski tzv. »Unscented« Kalmanov filtar. Za modeliranje prostora primijenjena je mrežasta karta popunjenosti, jer je u tom slučaju dovoljno uzeti u obzir samo nesigurnost mjerenja udaljenosti do najbližih prepreka za razliku od karata temeljenih na značajkama prostora kod kojih se mora uzeti u obzir i nesigurnost dodjeljivanja značajki izmjerenim udaljenostima. Eksperimenti napravljeni mobilnim robotom Pioneer 2DX (proizvođač *ActivMedia Robotics*) pokazuju da se veća točnost estimacije položaja i glađe gibanje mobilnog robota postižu primjenom nederivacijskog Kalmanova filtra.

Ključne riječi: nelinearni Kalmanov filtar, mobilan robot, lokalizacija, mrežasta karta zauzeća prostora

AUTHORS' ADDRESSES

Edouard Ivanjko, B.Sc.E.E., assistant
Faculty of Electrical Engineering and Computing/Department of Control and Computer Engineering in Automation, University of Zagreb, Unska 3, HR-10000 Zagreb

Ivan Petrović, Ph.D., associate professor
Faculty of Electrical Engineering and Computing/Department of Control and Computer Engineering in Automation, University of Zagreb, Unska 3, HR-10000 Zagreb

Mario Vašak, B.Sc.E.E., assistant
Faculty of Electrical Engineering and Computing/Department of Control and Computer Engineering in Automation, University of Zagreb, Unska 3, HR-10000 Zagreb

Received: 2004–11–30