

Infrared Beacons based Localization of Mobile Robot

J. Krejsa, S. Vechet

*Institute of Thermomechanics AS CR, v.v.i., Department of Mechatronics,
Technická 2, 616 69, Brno, Czech Republic, phone: +420 541142885, e-mail: krejsa@fme.vutbr.cz*

crossref <http://dx.doi.org/10.5755/j01.eee.117.1.1046>

Introduction

A number of methods solving mobile robot localization problem exist, mostly based on sensor information of various kinds fused together with an estimator [1]. While advanced types of sensors bring data of high precision and in high rates, the cost of such sensors is high either financially (laser rangefinders [2, 3]) or computationally (vision based approaches [4–6]). In order to decrease the cost the approaches based on beacons can be utilized with the drawback of necessity to install the beacons in the environment. The information from the beacons (usually the range and/or bearing) is commonly processed with Bayesian filters to obtain correct estimate of robot pose.

The beacons can be of various physical principles, from laser pointer emitter [7], through ultrasonic [8–10] to chirp-spread-spectrum (CSS) ranging based on the time of flight of radio frequency signals [11].

Our goal was to find the way to localize the robot in indoor environment with as low cost as possible and as few sensors as possible in order to be able to reduce the dimensions of the robot in future development. This goal can be met with the proposed localization system based on the infrared beacons. Infrared beacons cost is very low, but on the contrary to commonly used beacons the infrared beacons provide only the information about the bearing, not the ranging. However, bearing information is proved sufficient for localization using Extended Kalman filter (EKF).

Localization using bearing only measurements

Localization problem is stated as determination of the robot position $\mathbf{x}_k = [x_k^R, y_k^R, \varphi_k^R]^T$ towards global coordinate system in given time step k , as depicted in Fig. 1. The inputs to the localization algorithm are:

1) The motion model of the robot describing the change of the position due to actions $\mathbf{u}_k = [u_k^l, u_k^r]^T$ – the

translational and rotational velocities of the robot. The actions can either be extracted from motion controller or recalculated from odometry readings when available;

2) The measurement. In the case of infrared beacons it is the measurement of relative angles between the robot and the beacons placed in known fixed positions;

3) The initial position of the robot.

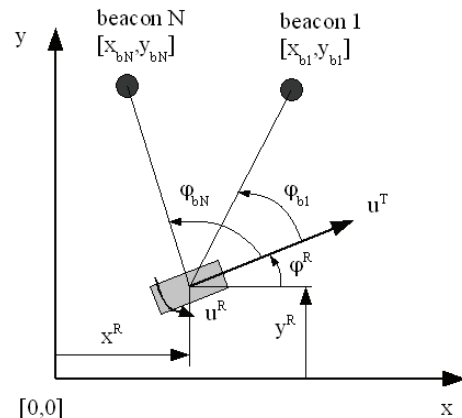


Fig. 1. The state of the robot is given by its position and orientation in the 2D plane with the respect to global coordinate system. The beacons are in known fixed positions (given by xy coordinates), measurement is represented by the relative angle between the robot and the beacon

Due to the imperfections of the motion model and noise in the measurement the position is represented in a probabilistic manner, using simple unimodal Gaussian approximation. The description of the underlying model can therefore be defined as:

$$\begin{cases} \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, k) + \mathbf{v}_k, \\ \mathbf{y}_k = h(\mathbf{x}_k, k) + \mathbf{w}_k, \end{cases} \quad (1)$$

where \mathbf{v}_k is white Gaussian process noise (representing the imperfections of motion model) with zero mean and covariance matrix \mathbf{V}_k , \mathbf{y}_k is system output, \mathbf{w}_k is white

Gaussian measurement noise with zero mean and covariance matrix \mathbf{W}_k and f and h are continuously differentiable nonlinear functions. The state transition function f defines how the state (robot position) changes when action is applied

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k = \begin{bmatrix} \cos \varphi_k^R u_k^t \Delta t + x_k^R \\ \sin \varphi_k^R u_k^t \Delta t + y_k^R \\ u_k^t \Delta t + \varphi_k^R \end{bmatrix} + \mathbf{v}_k. \quad (2)$$

Regarding the measurements, the output \mathbf{y}_k is the set of relative angles between the robot and beacons, as only the bearing information is available. The length of this set depends on number of beacons detected. There is a maximum of N beacons generally available (however, there is no guarantee, that all the beacons will be detected), their positions are (Fig. 1)

$$\mathbf{x}_{bi} = [x_{bi}, y_{bi}], i = 1, 2, \dots, N. \quad (3)$$

For a single beacon the output equation is

$$\mathbf{y}_{1k} = [h_1(\mathbf{x}_k, \mathbf{x}_{b1})] + [\mathbf{w}_{1k}], \quad (4)$$

where

$$h_1(\mathbf{x}_k, \mathbf{x}_{b1}) = [\text{atan2}(y_k^R - y_{b1}, x_k^R - x_{b1}) - \varphi_k^R]. \quad (5)$$

(5) gives the system output (relative angle between the robot and beacon depending on robot position, orientation and position of the beacon in environment. For N beacons the measurement equations are simply added one by one depending on what beacon was measured. In general

$$\mathbf{y}_k = \begin{bmatrix} h_1(\mathbf{x}_k, \mathbf{x}_{b1}) \\ h_2(\mathbf{x}_k, \mathbf{x}_{b2}) \\ \vdots \\ h_N(\mathbf{x}_k, \mathbf{x}_{bN}) \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{1k} \\ \mathbf{w}_{2k} \\ \vdots \\ \mathbf{w}_{Nk} \end{bmatrix}. \quad (6)$$

In (5) all the beacons are considered, when only some beacons are detected, appropriate rows are omitted. In order to use correct beacons positions, the identification of what beacon belongs to what measurement must be provided.

The position of the robot can not be determined exactly, instead the state estimate with mean $\hat{\mathbf{x}}_k$ and covariance matrix \mathbf{P}_k is produced by Bayesian filter [12-14]. There is a number of Bayesian filters suitable for the problem in hand, however when considering computational cost, the simplest one - Extended Kalman Filter (EKF) - gives the best performance/cost ratio. EKF represents the estimate by random vector with Gaussian distribution, producing the estimate with mean $\hat{\mathbf{x}}_k$ and covariance matrix \mathbf{P}_k . It uses the Taylor Expansion to cope with nonlinear functions f and h from (1). EKF therefore requires partial derivatives of the functions.

The EKF estimator works in two stages. First the motion model is applied (estimate changes when action is performed) and new position estimate is predicted. In

second stage the measurement is taken into account (beacons relative angles) and position estimate is corrected. Therefore the input is the current estimate (at the beginning of the whole process this estimate is equal to initial estimate of robot position), the action and the measurement. Necessary relations are given by following equations:

Prediction (state estimate change when action is applied):

$$\begin{cases} \hat{\mathbf{x}}_{k+1|k} = f(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k, k), \\ \mathbf{P}_{k+1|k} = \mathbf{F}_k \mathbf{P}_{k|k} \mathbf{F}_k^T + \mathbf{V}_k. \end{cases} \quad (7)$$

Update (correction of the estimate using measured data):

$$\begin{cases} \hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \tilde{\mathbf{y}}_{k+1}, \\ \mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{H}_{k+1} \mathbf{P}_{k+1|k}, \end{cases} \quad (8)$$

where $\mathbf{F}_k = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k}}$, $\tilde{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - h(\hat{\mathbf{x}}_{k+1|k}, k+1)$,
 $\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T \mathbf{S}_{k+1}^{-1}$, $\mathbf{S}_{k+1} = \mathbf{H}_{k+1} \mathbf{P}_{k+1|k} \mathbf{H}_{k+1}^T + \mathbf{W}_{k+1}$,
 $\mathbf{H}_{k+1} = \left. \frac{\partial h}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k+1|k}}$.

EKF requires the partial derivatives of f and h functions. For the motion model, the partial derivatives forming Jacobian \mathbf{F}_k are given by

$$\mathbf{F}_k = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k}} = \begin{bmatrix} 1 & 0 & -\sin \hat{\varphi}_{k|k}^R u_k^t \Delta t \\ 0 & 1 & \cos \hat{\varphi}_{k|k}^R u_k^t \Delta t \\ 0 & 0 & 1 \end{bmatrix}. \quad (9)$$

Jacobian for a single beacon is (the matrix is transposed to fit within the text)

$$\mathbf{H}_{k+1,1} = \begin{bmatrix} \frac{1}{1 + \left(\frac{y_{b1} - \hat{y}_{k+1|k}^R}{x_{b1} - \hat{x}_{k+1|k}^R} \right)^2} \frac{y_{b1} - \hat{y}_{k+1|k}^R}{(x_{b1} - \hat{x}_{k+1|k}^R)^2} \\ \frac{1}{1 + \left(\frac{y_{b1} - \hat{y}_{k+1|k}^R}{x_{b1} - \hat{x}_{k+1|k}^R} \right)^2} \frac{1}{x_{b1} - \hat{x}_{k+1|k}^R} \\ -1 \end{bmatrix}^T. \quad (10)$$

As the output equation for N beacons is formed according to (7), the Jacobians for more beacons are added the same way

$$\mathbf{H}_{k+1} = \begin{bmatrix} \mathbf{H}_{k+1,1} \\ \vdots \\ \mathbf{H}_{k+1,N} \end{bmatrix}. \quad (11)$$

Infrared beacons

The beacons are placed in environment in known positions according to (3). The measurement provides the

relative angle between the robot and the beacon (5). Such measurement is performed by beacons scanner device, that consist of a ring of M receivers, each i -th receiver returns Boolean value D_i indicating whether the beacon is detected or not. Such set of Boolean values is used to calculate the measured angle φ_{rel} in a following way. The angles between robot heading direction and position of the i -th receiver on scanner circle φ_{ri} are transformed to Cartesian coordinates, the mean is calculated individually for each axis (12) and then resulting angle is recalculated back (13). The validity val is calculated as the length of the resulting angle vector (14), as illustrated on Fig. 2:

$$c = \sum_{i=1}^M D_i \cos \varphi_{ri}, s = \sum_{i=1}^M D_i \sin \varphi_{ri}, \quad (12)$$

$$\varphi_{rel} = \text{atan2}(s, c), \quad (13)$$

$$val = \sqrt{s^2 + c^2}. \quad (14)$$

The measurement noise estimate, required by the estimator, is set according to measurement validity coefficient in (14).

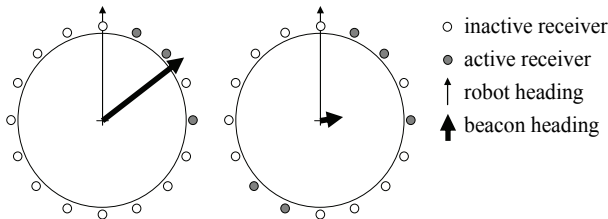


Fig. 2. Beacon relative angle determination. The length of resulting vector determines the validity of measurement. Left case shows typical valid measurement, right case shows the influence of reflections

The beacons used for localization are based on IrDA (Infrared Data Association) principle. Each beacon has its own power-supply and the infrared light is emitted via six infrared LEDs giving the flaring angle of approximately 120 degrees. These beacons can be placed anywhere in the robots environment because of small size/weight. The reliable range of the beacons is about 10m. The signal from the beacons is detected by beacon scanner device placed on the top of the robot chassis. The scanner, shown uncovered in Fig. 5., consists of 16 IrDA receivers, placed on a circle, covering whole 360 degrees range. The scanner detects every beacon within the range which is not hidden behind an obstacle.

In order to reduce the power consumption the beacons are not in operation continuously, but the light is emitted only upon request from the scanner. During beacon detection the ID of the beacon is also retrieved in the following way. The beacon scanner and the emitters communicate with each other using combined radio and infrared communication protocols. The low power consumption radio modules with free 433MHz modulation are used for one way data transmission, in the direction from the scanner to the beacons. This way the proper

beacon is selected for transmission and thus beacon identification problem is solved. For further power consumption reduction only the beacons within the operating range from estimated position are requested to respond. The beacon itself is shown on Fig. 3. with AA size battery to illustrate its dimensions, while in reality it is powered by single LiPol 1100mAh cell that can last for 55 hours of continuous emitting. As the beacon is used approximately twice per second, the total beacon endurance for fully charged battery is about 14 days.



Fig. 3. Infrared beacon with LED protective cover removed

The beacon starts with data transmission immediately after the proper identification number is received. The transmission from selected beacon to the scanner is also one way and it is based on infrared principle. The beacon uses infrared LED (880nm light-wave) with carrier frequency 38kHz. This signal is detected by infrared receivers on the scanner and resulting measured angle φ_{rel} and validity is calculated according to (12)–(14).

Robot navigation

The actions as one of the estimator inputs can be taken directly from the motion controller, or from robot odometry measurements. In later case the action must be recalculated to generalized velocities, depending on particular chassis type, so the estimator can remain intact and can be used with different types of robot chassis, from common to special types [15]. The overall simplified view of robot navigation scheme is shown on Fig. 4.

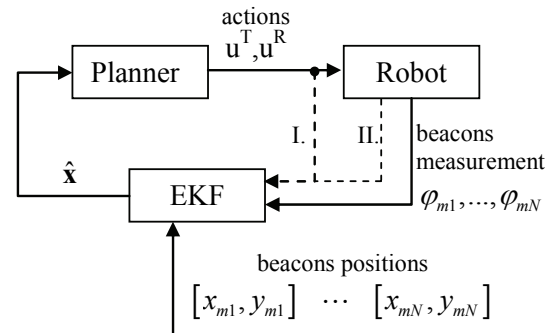


Fig. 4. Robot navigation scheme. Actions are taken either from the controller (I.) or transformed robot odometry measurements (II.)

The motion planner/controller produces desirable values of translational and rotational component of

velocity, the robot performs the motion and provides the set of measurements (relative angles towards the beacons together with the identification of the beacons). The data are fed to EKF estimator that produces the estimate of robot position, further used by the planner (together with obstacle detectors, goal position, etc. – those planner inputs are omitted on the figure). The planner can work in reaction mode or it can incorporate additional data from the surrounding, see e.g. [16].

Simulation

The localization system was first verified using simulations in Matlab. The motion model of the robot necessary for the correct settings of process noise in EKF in the simulations was identified experimentally on real test robot. The response of the robot to the set of actions was repeatedly measured and those data were used to obtain probabilistic model of the motion. The simulations included other implementations of Bayesian filter, Unscented Kalman filter (UKF) and Particle filter (PF) with 50 particles in particular, to compare the EKF based localization with filters. All implementations exhibited similar results with UKF and PF slightly surpassing EKF regarding the localization precision for the cost of higher computational expenses, see Table 1. showing the values obtained for 4 beacons visible.

Table 1. Comparison of EKF, UKF and PF estimators

	EKF	UKF	PF
Average localization error [m, degrees]	0.32±0.09 6.11±1.98	0.29±0.08 5.74±2.52	0.28±0.12 4.35±2.08
Computational cost [instruction/cycle]	684	3247	48724

Further simulations were performed to determine the influence of beacons measurement character. Beacons providing bearing only information, distance only and both distance and bearing were compared. For those simulations only EKF estimator was considered. As can be seen on Table 2. the bearing only requires higher number of beacons to obtain the same precision of localization. However, the cost of additional beacon does not overpass the cost of beacons capable of providing the distance information. Further more the error in robot orientation is of higher importance as it essentially influences the robot position in the long term.

Table 2. Influence of measurement character. The values in the table state localization error [m, degrees]

Measurement character	bearing only	distance only	bearing + distance
3 beacons	0.89±0.49 8.11±4.28	0.52±0.08 16.44±6.24	0.49±0.11 8.06±3.57
4 beacons	0.32±0.09 6.11±1.98	0.27±0.08 14.96±5.84	0.33±0.13 7.58±2.47
5 beacons	0.30±0.08 4.84±1.95	0.21±0.05 13.47±4.78	0.20±0.08 4.36±1.86

Verification experiment with real robot

The algorithm was first implemented into test robot Leela, shown on Fig. 5. Leela has common differential

drive chassis and is equipped with beacons receiver and obstacle detectors. Obstacle detectors are used solely for the path planning and not for localization. The robot is also equipped with odometry readings (IRC sensors on both drives), however during the tests the odometry was ignored, or used for comparison purposes only, as our goal was to reduce the number of necessary sensor to bare minimum.



Fig. 5. Test robot Leela with detail of beacons receivers ring around the neck and the uncovered beacons scanner

The estimator was implemented in ANSI C and all the code runs on 8-bit AVR family processor. ATmega128 is a low cost 8-bit processor with programmable 128kB flash memory and two serial communication interfaces. The processor acts as the main control unit and it runs both the localization algorithm and path planner. As the processor is a low cost, it has some limitations that must be overcome in implementation, e.g. goniometric functions are approximated using look-up table, whenever possible the calculations are transferred into integers, etc. While the processor runs on low frequency (16 MHz), it can operate the localization system with sufficient speed.

The tests were performed on a number of locations, Fig. 6. shows an example: entrance halls of our University. In this particular experiment 4 beacons were used, placed on the ground, with the view often blocked by passing people. The true trajectory was extracted from the videofeed and is shown compared to estimated position (and orientation) determined by EKF. One can see that there is a difference between real trajectory and estimated one, however the localization is stable.

Number of other tests was performed with different number of beacons, different number of students moving in the hall (representing dynamic obstacle) and different planned trajectories. This can be illustrated on Fig 7., where planned trajectory goes beyond the range of most of the beacons. While the certainty of the position is reduced (larger confidence ellipses), the localization is still stable. The influence of reflections and therefore the robustness of the beacon scanner filter was also examined. As we are interested in the use of the system in real environment, the exact dependencies on number of obstacles, reflections, etc. are impossible to measure and the only criteria is whether the robot loses its position or not. The tests approved that the method successfully localized the robot

in real environment with the presence of dynamic obstacles (people) blocking the beacons for extended period of time, when the maximum time depends on precision of motion control / odometry readings. The recovery of correct position after blocked beacons operation is fast and takes about 5 steps of the algorithm (approx. 1 second).

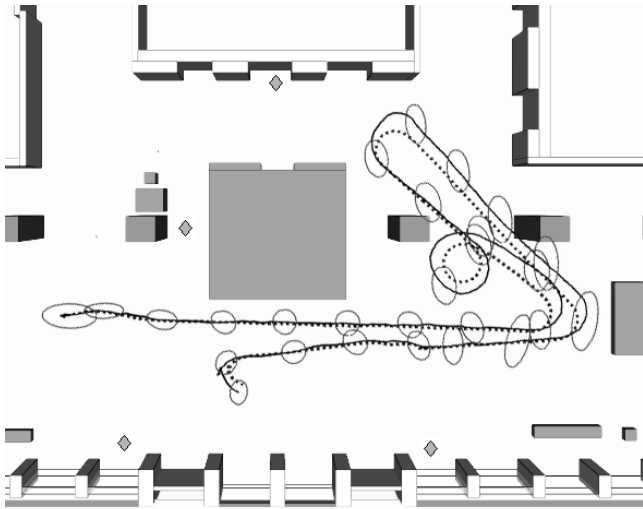


Fig. 6. Evaluation experiment example. Dotted line represents real trajectory, solid line the estimate. Beacons are denoted by diamond symbols. Ellipses denote 95 % confidence interval for the estimate. Path traveled started on the left side of the figure

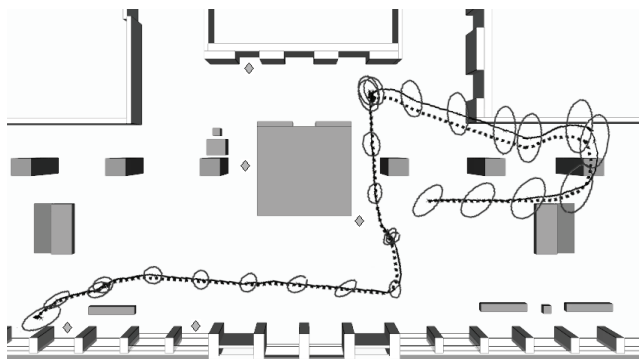


Fig. 7. Evaluation experiment example. Path traveled started on the left side of the figure. The path on the right side is out of the beacons range

Real world operation

As the proposed localization system proved to be capable of successful localization, it was used in larger scale robot – the prototype of commercial robot Advee (Fig. 8). The robot chassis differs from previously described robot as it uses Ackerman steering principle [17], resulting in more reliable odometry. Therefore the odometry readings from IRC sensors on front wheels were used instead of controllers commands (velocities) as the inputs to the estimator. The odometry readings were recalculated into velocities from simple trigonometry and measurements of given time interval. The implementation of the estimator was ported from AVR processor to TS-7800 embedded computer based on ARM family processor running OS Linux, that handles the lower levels of robot

software architecture (interaction with hardware devices, path planner and position estimator).

The beacon scanner is installed on top of the head of the robot. As can be seen from the figure, the robot height is sufficient to keep the scanner high enough for most of dynamic obstacles not to block the direct view between the scanner and the beacons installed close to the ceiling.



Fig. 8. Robot Advee with proposed localization system installed

The tests performed confirmed the usability of proposed solution, as the robot was capable of wandering in a crowded hall for 5 hours without losing its position. Currently the robot has over 300 hours of operation in real environment with no issues regarding the localization.

Conclusions

When selecting the proper localization system for mobile robot, several matters must be considered. The key factors are the application of the robot, computational power available and the characteristics of its environment, such as the accessibility. Proposed localization technique is focused on low cost both in financial, spatial and computational domains. Both simulations and real robots verifications proved, that localization using bearing only beacons as the only source of the information from the outside world can be done with low cost sensors and computational unit. The overall cost of sensor equipment for the robot falls dramatically and the possibility to use the beacons as the only sensors for localization enables to reduce the dimensions of the robot. The prototype cost for the beacons is about 30,-EUR each, the scanner price is about 100,-EUR.

Reduction of the cost may seem not as important as the performance. However, there are numerous possibilities of application of the localization system when its cost is on reasonable levels. What is more of an issue is the essential drawback of any localization based on beacons, no matter the principle, and that is the necessity to install and service

the beacons. Presented beacons cope with the service necessity by prolonged live of the beacons. As the beacons are operating only upon request, the power consumption is low and the beacon can run on battery power for sufficiently long. Thus the installment does not require any wiring and can be done relatively easily. Beacons are reliable and battery replacement/charging is the only service necessary. Further reduction of power consumption by selection of the beacons to respond based on the position estimate of the robot reduces the time required for processing the beacons data in larger areas where higher total number of beacons placed in environment is expected.

Regarding the estimation method, certainly the EKF is not the best available estimator in the terms of precision, mainly when highly nonlinear functions are considered. However, the computational requirements are extremely low and therefore the estimator can run on microcontroller with sufficiently low estimation times that allow the localization procedure to be run in reasonably short intervals. When computational power is sufficient, the Unscented Kalman filter or Particle filters can be used for estimation, but according to our experiences the improvement of localization is not substantial.

Acknowledgements

The research was funded by the Academy of Sciences of the Czech Republic under the research plan AV0Z20760514.

References

1. **Bartkevičius S., Urniežius R., Žebrauskas S.** Robot Localization under Dynamic Uncertainty // *Electronics and Electrical Engineering*. – Kaunas: Technologija, 2009. – No. 7(95). – P. 3–6.
2. **Krejsa J., Věchet S.** Dealing with Sensor Errors in Scan Matching for Simultaneous Localization and Mapping // *Engineering Mechanics*, 2008. – Vol. 15. – No. 5. – P. 337–344.
3. **Arras K., Tomatis N.** Improving Robustness and Precision in Mobile Robot Localization by Using Laser Range Finding and Monocular Vision // *Third European Workshop on Advanced Mobile Robots (EUROBOT '99)*. – Zurich, Switzerland, 1999.
4. **Davison A.** Real-time simultaneous localisation and mapping with a single camera // *9th ICCV*, 2003. – Vol. 2. – P. 1403–1410.
5. **Bekris K., Glick M., Kavradi L.** Evaluation of Algorithms for Bearing-Only SLAM // *IEEE Intl. Conf. on Robotics and Automation*. – Orlando, FL, 2006. – P. 1937–1944.
6. **Xu D., Han L., Tan M., Li Y. F.** Ceiling-based visual positioning for an indoor mobile robot with monocular vision // *IEEE Trans. Ind. Electron.*, 2009. – Vol. 56. – No. 5. – P. 1617–1628.
7. **Hernandez S., Torres J. M., Morales C. A., Acosta L.** A new low cost system for autonomous robot heading and position localization in a closed area // *Autonomous robots*, 2003. – Vol. 15(2). – P. 99–110.
8. **Gonzalez J., et al.** Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach // *Robotics and autonomous systems*, 2009. – Vol. 57(5). – P. 496–507.
9. **Yun J., et al.** Robust positioning a mobile robot with active beacon sensors // *Knowledge-based intelligent information and engineering systems*, 2006. – Vol. 4251. – P. 890–897.
10. **Yenilmez L., Terneltas H.** Ultrasonic positioning of a mobile robot using single external beacon // *Dynamics of Continuous Discrete and Impulsive Systems*, 2006. – Vol. 13E. – P. 2837–2841.
11. **Cho H., Kim S. W.** Mobile robot localization using biased chirp-spread-spectrum ranging // *IEEE Trans. Ind. Electron.*, 2010. – Vol. 57. – No. 8. – P. 2826–2835.
12. **Meng Q. H., Sun Y. C., Cao Z. L.** Adaptive extended Kalman filter (AEKF) based mobile robot localization using sonar // *Robotica*, 2000. – Vol. 18. – No. 5. – P. 459–473.
13. **Cong T. H., Kim Y. J., Lim M. T.** Hybrid extended Kalman filter based localization with a highly accurate odometry model of a mobile robot // *Int. Conf. on Control, Automation and Systems*, 2008. – P. 738–743. <http://dx.doi.org/10.1109/ICCAS.2008.4694596>.
14. **Woo J., Kim Y. J., Lee J., Lim M. T.** Localization of mobile robot using particle filter // *SICE-ICASE International Joint Conference*, 2006. – P. 3031–3034.
15. **Grepel R., et al.** Development of 4WS/4WD Experimental Vehicle: platform for research and education in mechatronics // *IEEE Int. Conf. on Mechatronics, ICM'2011*, 2011.
16. **Baranauskas V., Derviniene A.** Rationalization of the Path Search Algorithm // *Electronics and Electrical Engineering*. – Kaunas: Technologija, 2010. – No. 5(101). – P. 79–82.
17. **Hrbáček J., Ripel T., Krejsa J.** Ackermann mobile robot chassis with independent rear wheel drives // *Proceedings of EPE-PEMC'2010 conference*, 2010. – P. T5/46–T5/51

Received 2011 03 28

Accepted after revision 2011 05 30

J. Krejsa, S. Vechet. Infrared Beacons based Localization of Mobile Robot // *Electronics and Electrical Engineering*. – Kaunas: Technologija, 2012. – No. 1(117). – P. 17–22.

The paper introduces the method for indoor localization of a mobile robot based on infrared beacons positioned stationary in the environment. The beacons provide bearing only information, fused together with either motion controllers commands or robot odometry via Extended Kalman filter to estimate the location of the robot. Simulation results include comparison with UKF and PF based localization and comparison with distance, distance/bearing beacons measurement. The verification on the test real robot with differential chassis and commercial robot with Ackerman chassis is provided. The main advantage of the method is the low cost of whole localization system. Ill. 8, bibl. 17, tabl. 2 (in English; abstracts in English and Lithuanian).

J. Krejsa, S. Vechet. Mobiliojo roboto lokalizavimas naudojant infraraudonųjų spindulių švyturėlius // *Elektronika ir elektrotechnika*. – Kaunas: Technologija, 2012. – Nr. 1(117). – P. 17–22.

Pristatomas metodas mobiliam robotui lokalizuoti patalpoje naudojant stacionarius infraraudonųjų spindulių švyturėlius, išdėstyti aplinkoje. Švyturėlių informacija, kartu su judesio kontrolierių komandomis arba roboto odometrijos duomenimis susieta per išplėstinį Kalmano filtrą, naudojama roboto vietai nustatyti. Imitavimo rezultatuose pateikiamas palyginimas su necentruoto Kalmano filtro ir dalelių filtro lokalizacijos metodais. Pateikiama verifikacija, gauta atliekant naudojant tikro roboto su diferencine važiuokle ir komercinio roboto su Akermano važiuokle bandymus. Pagrindinis siūlomo metodo pranašumas – maža visos lokalizacijos sistemos kaina. Il. 8, bibl. 17, lent. 2 (anglų kalba; santraukos anglų ir lietuvių k.).