



Review Paper

Enhanced SLAM for a Mobile Robot using Unscented Kalman Filter and Radial Basis Function Neural Network

Panah Amir

Young Researchers Club, Qazvin Branch, Islamic Azad University, Qazvin, IRAN

Available online at: www.isca.in

Received 31st October 2012, revised 17th November 2012, accepted 12th December 2012

Abstract

This paper presents a Hybrid filter based Simultaneous Localization and Mapping (SLAM) for a mobile robot to compensate for the Unscented Kalman Filter (UKF) based SLAM errors inherently caused by its linearization process. The proposed Hybrid filter consists of a Radial Basis Function (RBF) and UKF which is a milestone for SLAM applications. A mobile robot autonomously explores the environment by interpreting the scene, building an appropriate map, and localizing itself relative to this map. The proposed approach, based on a Hybrid filter, has some advantages in handling a robot with nonlinear motions because of the learning property of the RBF neural network. The simulation results show the effectiveness of the proposed algorithm comparing with an UKF based SLAM and also it shows that in larger environments has good efficiency.

Keywords: SLAM, UKF, RBF, Hybrid filter SLAM.

Introduction

Research efforts on mobile robotics have mainly focused on topics such as obstacle detection, autonomous navigation, path planning, exploration, map building, etc., and many algorithms have been proposed for these purposes¹. Currently SLAM is one of the most widely researched major subfields of mobile robotics. In order to solve SLAM problems, statistical approaches, such as Bayesian Filters, have received widespread acceptance. Some of the most popular approaches for SLAM include using a Kalman filter (KF), an extended Kalman filter (EKF) and an unscented Kalman filter (UKF) and a particle filter². As in any UKF based algorithm, the UKF SLAM makes a Gaussian noise assumption for the robot motion and its perception. In addition, the amount of uncertainty in the posterior of the UKF SLAM algorithm must be relatively small; otherwise, the linearization in the UKF tends to introduce unbearable errors. The UKF uses the unscented transform to linearize the motion and measurement models. Especially, the UKF is usually used in order to compensate for the EKF's drawbacks which inherently results from linear approximation of nonlinear functions and the calculation of Jacobian matrices³. RBF neural network, adaptive to the changes of environmental information flowing through the network during the process, can be combined with an UKF to compensate for some of the disadvantages of an UKF SLAM approach, which represents the state uncertainty by its approximate mean and variance⁴.

Houshangi and Azizi⁵ integrated the information from odometry and gyroscope using UKF. To improve the performance of odometry, a fiber optic gyroscope is used to give the orientation information that is more reliable. This method is simple to

implement, needless to frequent calibration and applicable to different situations likely EKF. The results show that the UKF estimates the robot's position and orientation more accurately than the EKF.

Ronghui Zhan and Jianwei Wan⁶ presents a robust learning algorithm for an multilayered neural network based on UKF is derived. Since it gives a more accurate estimate of the link weights, the convergence performance is improved. The algorithm is then extended further to develop a neural network aided UKF for nonlinear state estimation. The neural network in this algorithm is used to approximate the uncertainty of the system model due to mismodeling, extreme nonlinearities, etc.

Choi et al⁷ approached the SLAM problem with a neural network based on an extended Kalman filter (NNEKF). When the robot is trained online by a neural network, the NNEKF can capture the unmodeled dynamics, and adapt to the changed conditions intelligently. According to the research results, the NNEKF SLAM, shows better performance than the EKF SLAM.

In this paper, we present a Hybrid approach using RBF neural network and UKF based SLAM problem for decreasing uncertainty in compare to SLAM using UKF.

We also discuss the effectiveness of RBF algorithm to handle nonlinear properties of a mobile robot. Some related algorithms on SLAM are described in section 2, and the Hybrid SLAM algorithm is presented in section 3. Section 4 shows the simulation results of the SLAM based on UKF, and Hybrid filter. Concluding remarks, discussion and further research are discussed in section 5.

Related Algorithms for SLAM

Radial Basis Function Neural Network: Design of artificial neural networks is motivated by imitating human brain and thinking activities as a mechanical tool for various purposes.

Frequently, neural networks are used especially in modeling and simulation of nonlinear systems. Neural networks have two fundamental characteristics of learning based on experimental data and structural parallel. RBF neural network is proposed as a neural network model by Moody and Darken. In RBF neural network, originally, RBF, approximation density function and level fitting technique, are considered as probability functions⁸. RBF neural network typically have three layers, namely, an input layer, a hidden layer with a nonlinear RBF activation function, and a linear output layer. Input layer transfers all data information to hidden layer. This transfer is done through a series of communications with unknown weights. This hidden layer consists of local basis functions. Often, a function is used which is called Gaussian function and does a nonlinear transition process and it is completely local⁸.

Network training is divided into two stages: first, the weights from the input to the hidden layer are determined; then, the weights from the hidden layer to the output layer are determined. The results can be used to simulate the nonlinear relationship between the sensors measurements with the errors, and the ideal output values by using the least squares method⁹.

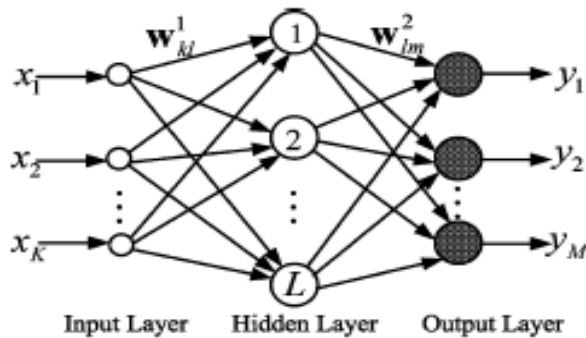


Figure-1

A Radial Basis Function Neural Network Structure

Unscented Kalman Filter : UKF was introduced for the first time by Uhlmann and Julier in 1997. This filter is built based on transformation as unscented transformation.

The base of this transformation is based on this principal which states the estimation a probability is easier than a nonlinear function or a nonlinear transformation. UKF method is more accurate than the EKF for estimation of nonlinear systems¹⁰. EKF for linearization of nonlinear system only uses the first term of Taylor series. Therefore, in cases where the system is fully nonlinear and higher terms in the Taylor series are important, Linearization model by this method is not a good approximation of the real system and this leads to lack of

accuracy in the estimation of states or parameters of these types of systems. In The UKF, there is no need to calculate Jacobian matrix. Since, the process noise in this system is accumulative; therefore the augment state vector is used to implementation this method. In this method, the mean and covariance estimation are calculated with considering the second order of the Taylor series¹⁰. Suppose that a random variable x with mean μ and covariance P_x is given and also a random variable z as with x is associated: $z=f(x)$. Problem calculating the mean and covariance z is, as predicted and corrected the problem in stages UKF a nonlinear system. In the method of unscented transformation, to obtain the mean and covariance random variable z , a set of weighted points called sigma points are used. This sigma points should be selected so that have a mean μ and covariance P_x .

An n-dimensional random variable can be approximated by a set of $2n+1$ weighted sigma points given by the following deterministic algorithm: ($0 < i < n$).

$$X_0 = \mu \quad , \quad W_0 = \frac{\lambda}{n+\lambda} \quad (1)$$

$$X_i = \mu + (\sqrt{(n+\lambda)P_x})_i \quad , \quad W_i = \frac{\lambda}{2(n+\lambda)} \quad (2)$$

$$X_{i+n} = \mu - (\sqrt{(n+\lambda)P_x})_i \quad , \quad W_{i+n} = \frac{\lambda}{2(n+\lambda)} \quad (3)$$

$$\lambda = \alpha^2(n + \beta) - n \quad (4)$$

n number of state variables are augmented. Also, α and β are the coefficients that with their adjustment, the estimation error can be minimized and their values influence on the error rate resulting from higher terms in the Taylor series. In the above relations, $k \in R$ and $(\sqrt{(n+k)P_x})_i$, the i-th row or column of the matrix is the square root of $(n+k)P_x$, W_i is the weight which is associated with the i-th point and k also is used for adjusting the filter more accuracy¹⁰.

Unscented transformation algorithm, first, each point of the set points by a nonlinear function for mapping to a new point and will obtain a new set of sigma points. Then, we can calculate the mean and covariance values of the new random variable. Consider the following nonlinear system:

$$x_k = f(x_{k-1}, u_{k-1}, \varepsilon_k) \quad (5)$$

$$z_k = h(x_k, u_k, \delta_k) \quad (6)$$

Where x is state vector and u is control input and ε , δ , are the system noise and the measurement noise, respectively. In the first phase of implementing this filter, the augment state vector will become as the following form:

$$X_k^s = \begin{bmatrix} X_k \\ \varepsilon \\ \delta \end{bmatrix} \quad (7)$$

In continue, we will see all the formulas which has been implemented in the UKF in which itself includes of two main from sections: Measurement update and Time update¹¹.

The Time Update

$$X_k^a = f(X_k^a, u_k, \varepsilon_k) \tag{8}$$

$$\mu_k = \sum_{i=0}^{2n} w_i X_{i,k}^a \tag{9}$$

$$P_k = \sum_{i=0}^{2n} w_i [X_{i,k}^a - \mu_k][X_{i,k}^a - \mu_k]^T \tag{10}$$

$$z_k = h(x_k, u_k, \delta_k) \tag{11}$$

$$\bar{z}_k = \sum_{i=0}^{2n} w_i z_k \tag{12}$$

The Measurement Update

$$P_{z_k z_k} = \sum_{i=0}^{2n} w_i [z_{i,k} - \bar{z}_k][z_{i,k} - \bar{z}_k]^T \tag{13}$$

$$P_{x_k z_k} = \sum_{i=0}^{2n} w_i [X_{i,k}^a - \mu_k][z_{i,k} - \bar{z}_k]^T \tag{14}$$

$$K_k = P_{x_k z_k} P_{z_k z_k}^{-1} \tag{15}$$

$$\mu_k = \mu_k + K_k (z_k - \bar{z}_k) \tag{16}$$

$$P_k = P_k - K_k P_{z_k z_k} K_k^T \tag{17}$$

Where X_k^a , μ_k , P_k , z_k , \bar{z}_k , $P_{z_k z_k}$, $P_{x_k z_k}$ and K_k , are motion model, predicted mean, observation model, predicted observation, innovation covariance, cross correlation matrix and Kalman gain¹², respectively.

SLAM Using Unscented Kalman Filter: A solution to the SLAM problem using UKF, with many interesting theoretical advantages, is extensively described in the research literature. This is despite the recently reported inconsistency of its estimation because it is a heuristic for the nonlinear filtering problem. Associated with the UKF is the Gaussian noise assumption, which significantly impairs the UKF SLAM's ability to deal with uncertainty. With a greater amount of uncertainty in the posterior, the linearization in the UKF fails¹³. An UKF based on a Bayes filter has two steps, prediction and correct, for SLAM using the measured sensor data of a mobile robot.

SLAM Algorithm Using Hybrid Filter

A new Hybrid filter SLAM with UKF is proposed here, augmented by an RBF acting as an observer to learn the system uncertainty online¹⁴. An adaptive state estimation technique using an UKF and a RBF has been developed. The mean, μ_k which is derived from environmental information values $(x \ y \ \theta \ \varepsilon \ \delta)$ using the RBF algorithm, is entered to the prediction step, as shown in figure-2.

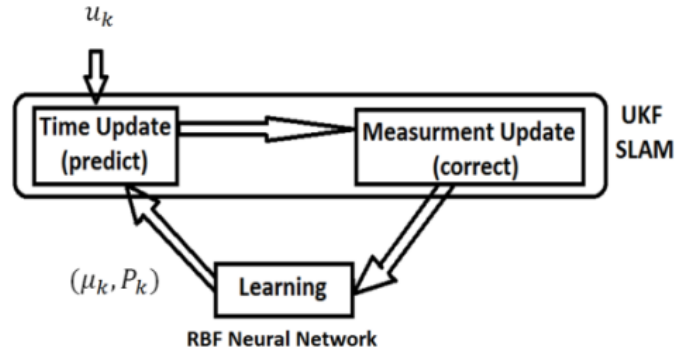


Figure-2
 The architecture of the Hybrid filter SLAM

In this paper, Basic inputs are mean, covariance which are calculated by prior input, u_{k-1} , and present input, u_k . The robot calculates the prior mean and covariance in a prediction step, and then, in an observation step, it calculates a Kalman gain, present mean and covariance and defined features.

RBF neural network is very important, as the kernel of the Hybrid filter is the complementation of errors onto stochastic UKF SLAM processing through the training process. RBF neural network can operate as a fast and accurate means of approximating a nonlinear mapping based on observed data.

Time Update (Predict): The Hybrid filter SLAM algorithm is described using a robot's pose and features, such as the location of landmarks¹⁵. For the SLAM, the basic motion model of the mobile robot needs to be presented. A configuration of the robot with a state equation $X^a = (x \ y \ \theta \ \varepsilon \ \delta)^T$, has the form of equation (18) since it is assumed that the robot is equipped with encoders and exteroceptive sensors.

$$X_k^a = \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ \varepsilon_k \\ \delta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + v_k \Delta t \cos(\theta_k) \\ y_{k-1} + v_k \Delta t \sin(\theta_k) \\ \theta_{k-1} + v_k \Delta t \sin(\frac{\Delta \theta}{L}) \\ \varepsilon_{k-1} \\ \delta_{k-1} \end{bmatrix} \tag{18}$$

$$u_k = v_k + N(0, M_k) \tag{19}$$

Where v_k is velocity of wheels, and L is the width between the robot's wheels, and Δt is the sampling period. Finally, M_k describes the covariance matrix of the noise in control space. The state equation for landmarks, combined with the robot position, is denoted by the vector Y_k , where c is the number of landmarks. ($0 < i < c$)

$$Y_k^a = \begin{bmatrix} X_k^a \\ m \end{bmatrix} = (x_k \ y_k \ \theta_k \ \varepsilon_k \ \delta_k \ m_{k,x}^i \ m_{k,y}^i \ s_k^i \ 00)^T \tag{20}$$

The state transition probability of a Hybrid filter SLAM has the form of Eq. (21):

$$X_k^a = f(X_{k-1}^a, u_{k-1}) + N(0, \varepsilon_k) \quad (21)$$

Under the linearity assumption where f represents the nonlinear functions, ε_k is the process noise, and u_k is control input.

For the Taylor expansion of function, f its partial derivative is used with respect to X_k^a , as shown in equation (22).

$$f'(X_{k-1}^a, u_k) = \frac{\partial f(X_{k-1}^a, u_k)}{\partial X_k^a} \quad (22)$$

f is approximated at u_k and μ_{k-1} . The linear extrapolation is achieved by using the gradient of f at u_k and μ_{k-1} as shown in Eq. (23).

$$f(X_{k-1}^a, u_k) = f(\mu_{k-1}, u_k) + f'(\mu_{k-1}, u_k)(X_k^a - \mu_{k-1}) \quad (23)$$

With the replacement values obtained from equations 1, 2, 3, 4, 5, prior mean and covariance have the following form of:

$$\mu_k = \sum_{i=0}^{2n} w_i X_{i,k}^a \quad (24)$$

$$P_k = \sum_{i=0}^{2n} w_i [X_{i,k}^a - \mu_k][X_{i,k}^a - \mu_k]^T \quad (25)$$

As described in Eq. (26), the observation model, z_k consists of the nonlinear measurement function h and the observation noise δ_k .

$$z_k = h(Y_k^a) + N(0, \delta_k) = \begin{bmatrix} r_k^i \\ \theta_k^i \end{bmatrix} = \begin{bmatrix} \sqrt{(m_{k,x}^i - x_k)^2 + (m_{k,y}^i - y_k)^2} \\ \tan^{-1} \left(\frac{m_{k,y}^i - y_k}{m_{k,x}^i - x_k} \right) - \theta_k \end{bmatrix} + N(0, \delta_k) \quad (26)$$

$$m^i = (m_{k,x}^i \ m_{k,y}^i)^T \quad (27)$$

$$\bar{z}_k = \sum_{i=0}^{2n} w_i z_k \quad (28)$$

The Measurement Update (Correct)

To obtain the Kalman gain K_k , we need to calculate P_{xkxk} and P_{xkyk} in the feature-based maps¹⁶. To obtain the values P_{xkxk} and P_{xkyk} , it is necessary to calculate $X_{i,k}^a, \mu_{i,k}, \bar{z}_{i,k}, z_{i,k}$ that are calculated in equations 18, 24, 26, 28, with replacement of these values, we will have the following equations:

$$P_{x_k x_k} = \sum_{i=0}^{2n} w_i [z_{i,k} - \bar{z}_k][z_{i,k} - \bar{z}_k]^T \quad (29)$$

$$P_{x_k y_k} = \sum_{i=0}^{2n} w_i [X_{i,k}^a - \mu_k][z_{i,k} - \bar{z}_k]^T \quad (30)$$

$$K_k = P_{x_k y_k} P_{x_k x_k}^{-1}$$

In this following, RBF algorithm with UKF are considered to complete SLAM of the mobile robot. RBF algorithm is involved with train through input data and measurement values. In the training process, weights are decided based on the relation of input data and each hidden layers. RBF algorithm need higher weight to objective value on the higher relations between poses and heading angle with comparing to measurement. When applying the other case for RBF, it is the same to substitute inputs. The RBF algorithm generally consists of two weight layers; one hidden layer and the output layer. In addition, the second weight, ω_0 , equals zero because the output offset is zero. Therefore, new estimated mean, can be described as in equation (32): ($0 \leq j \leq J - 1$)

$$\hat{\mu}_k^j = \omega_0 + \sum_{j=0}^{J-1} \omega_j \varphi_k^j(\mu_k^j) \quad (32)$$

$$= \xi \left(\sum_{j=0}^{J-1} \varphi_k^j(\mu_k^j) \right) = \xi \left(\sum_{j=0}^{J-1} \exp \left(-\frac{\|\mu_k^j - d^j\|^2}{2(\sigma^j)^2} \right) \right)$$

μ_k is an n-dimensional input vector and d^j stands for the center of the j-th basis function with the same dimension of the input vector. In the equations considered, τ^j denotes the width of the basis function, N is the number of hidden layer's nodes, $\|\mu_k^j - d^j\|$ describes the Euclidean norm of representing the distance between μ_k^j and d^j and $\varphi_k^j(x)$ means the response of the j-th basis function of the input vector with a maximum value at d^j . The next process to obtain the prior mean and the covariance is to update the results from equation (32). The process described in the above 5 steps repeats until the end of the navigation.

$$\mu_k = \hat{\mu}_k + K_k(z_k - \bar{z}_k) \quad (33)$$

$$P_k = P_k - K_k P_{x_k x_k} K_k^T \quad (34)$$

Simulations

To show the effectiveness of the proposed algorithm, the Matlab code, developed by Bailey¹⁷, was modified. The simulation was performed with constraints on velocity, steering angle, system noise, observation noise, etc., for a robot with a wheel diameter of 1[m] and maximum speeds of 3[m/sec]. The maximum steering angle and speed are 25[°] and 15[°/sec], respectively.

The control input noise is assumed to be a zero mean Gaussian with $\sigma_v (=0.2[m/s])$ and $\sigma_\varphi (=3[°])$. For observation, the number of arbitrary features around waypoints was used. In the observation step, a range bearing sensor model and an observation model were used to measure the feature position and robot pose, which includes a noise with level of 0.1[m] in range and 1[°] in bearing. The sensor range is restricted to 15[m] for small areas and 30[m] for large areas, which is sufficient to detect all features in front of the mobile robot.

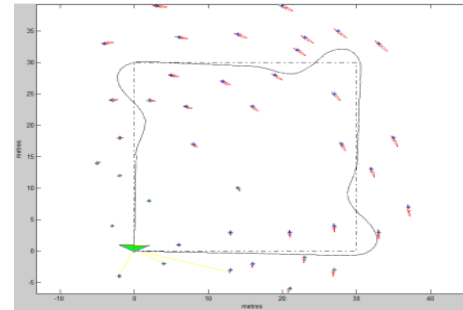
In this research, two navigation cases of the robot are surveyed: a Rectangular navigation, and a Widespread navigation. Specifications of the navigation maps are described in table-1.

Table-1
Fundamental specification for navigation

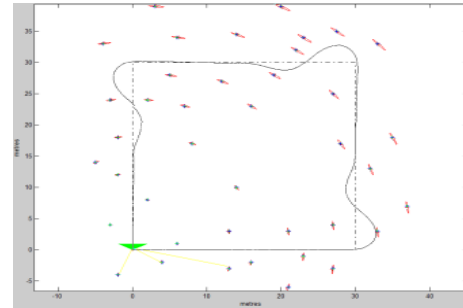
Item	Rectangular	Widespread
Feature	40	35
Waypoint	5	18
Area[m]	30*30	200*160

Navigation on Rectangular map: In the case of rectangular navigation, the UKF based navigation and Hybrid filter based navigation are shown in figure 3, where both results show distortions during navigation at the three edges. The mobile robot decides a direction for the navigation based on the information from the locations of landmarks detected, but it does not instantly turn because it has $1[^\circ]$ in bearing when the robot tries to turn through the edges.

The dashed line, show the paths of robots should traverse and the bold black line is Robot path, based on data described by the actual odometry. In Figure 4, the dashed gray line and the bold black line are the x, y, and θ errors in the case of UKF SLAM and Hybrid filter SLAM with RBF algorithm, respectively.



UKFSLAM



RBF UKFSLAM

Figure-3

Navigation result on rectangular map X Error(meter)

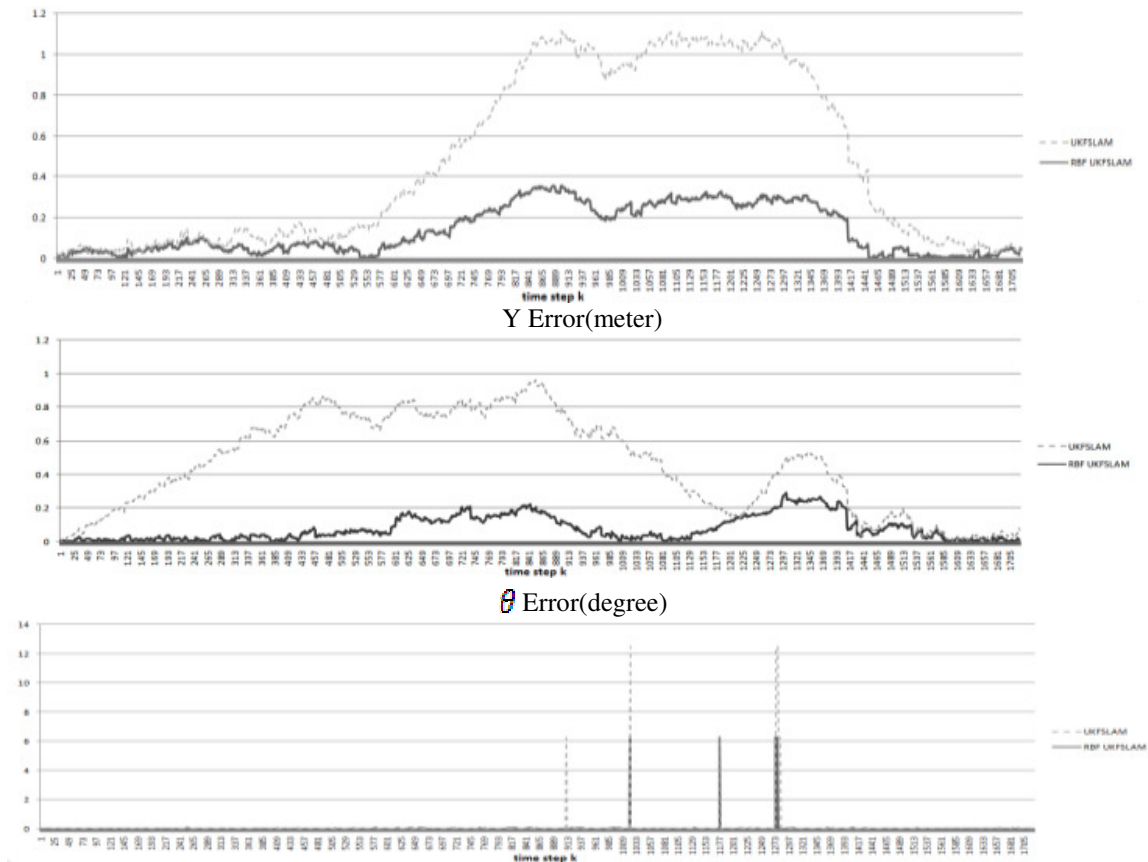


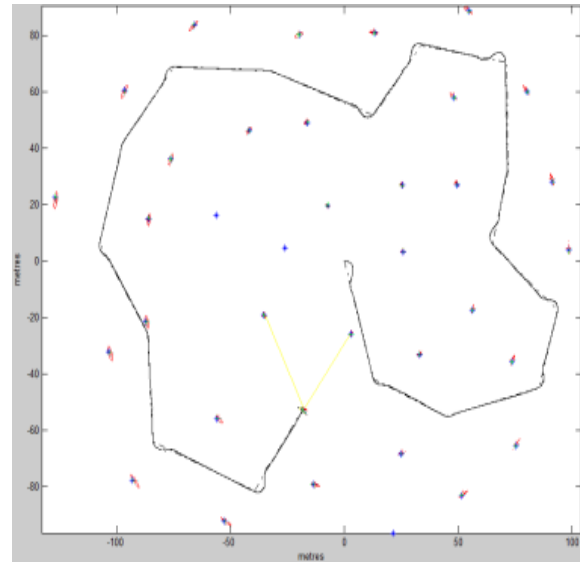
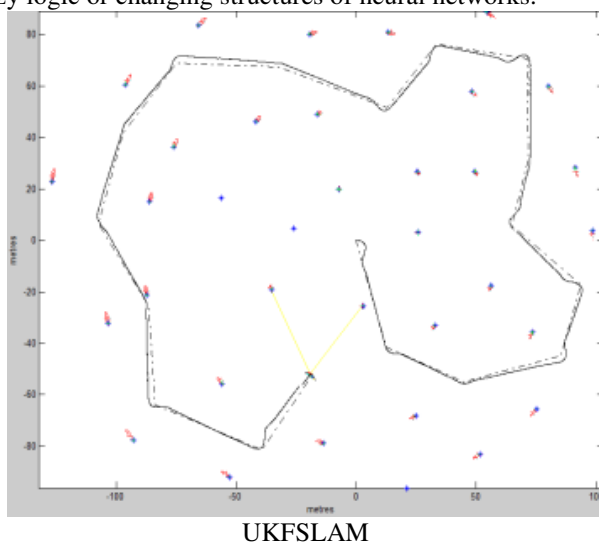
Figure-4

Navigation errors on rectangular map

Navigation on Widespread map: In the case of widespread navigation, the UKF based navigation and Hybrid filter based navigation are shown in figure 5, where more diversions are shown in edges with larger angle during navigation. The mobile robot decides a direction for the navigation based on the information from the locations of landmarks detected, but it does not instantly turn in edges because unpredictable changes in incoming data. The dashed line, show the paths of robots should traverse and the bold black line is Robot path, based on data described by the actual odometry. In figure 6, the dashed gray line and the bold black line are the x, y, and θ errors in the case of UKF SLAM and Hybrid filter SLAM with RBF algorithm, respectively.

Conclusion

The SLAM is one of the most fundamental problems in the quest for autonomous mobile robots since the robot keeps track of its location by maintaining a map of the physical environment and an estimate of its position on that map. This paper proposes Hybrid filter SLAM methods, the RBF SLAM with UKF on a mobile robot, to make up for the UKF SLAM error inherently caused by its linearization process and noise assumption. The proposed algorithm consists of two steps: the RBF Neural Network and the UKF algorithm. The simulation results for two different navigation cases show that the efficiency of the proposed algorithm based on RBF as compared with the UKF SLAM and it also shown that provided algorithms has favorable results in wider environment but we need to use long range sensors. To verify the effectiveness of the proposed algorithm, simulation in Matlab with UKF and RBF are performed. Based on the simulation results, UKF SLAM has more errors than Hybrid filter SLAM. In addition, the results confirm the Hybrid filter SLAM is more stable for robot navigation in the simulation. Research under harsh and real-time condition is under way to verify the robustness of the proposed algorithm by fuzzy logic or changing structures of neural networks.



RBF UKFSLAM

Figure-5

Navigation result on widespread map X Error(meter)

References

1. Kim J.M., Kim Y.T. and Kim S.S., An accurate localization for mobile robot using extended Kalman filter and sensor fusion, IEEE International Joint Conference on Neural Networks, 2928-2933 (2008)
2. Kyung-Sik Choi, Suk-Gyu Lee.: Enhanced SLAM for a Mobile Robot using Extended Kalman Filter and Neural Networks, *International Journal of Precision Engineering and Manufacturing*, **11(2)**, 255-264 (2010)
3. Zhu J., Zheng N., Yuan Z., Zhang Q. and Zhang X., Unscented SLAM with conditional iterations, 2009 IEEE Intelligent Vehicles Symposium, 134-139 (2009)
4. Vafaesefat A., Optimum Creep Feed Grinding Process Conditions for Rene 80 Supper Alloy Using Neural network, *Int. J. Precis. Eng. Manuf.*, **10(3)**, 5-11 (2009)
5. Houshangi N. and Azizi F., Accurate mobile robot position determination using unscented Kalman filter, 2005 Canadian Conference on Electrical and Computer Engineering, 846-851 (2005)
6. Zhan R. and Wan J., Neural Network-Aided Adaptive Unscented Kalman Filter for Nonlinear State Estimation, *IEEE Signal Processing Letters*, **13(7)**, 445-448 (2006)
7. Choi M.Y., Sakthivel R. and Chung W.K., Neural network aided extended Kalman filter for SLAM problem, IEEE International Conference on Robotics and Automation, 1686-1690 (2007)
8. Hu Y.H. and Hwang J.N., Handbook of Neural Network Signal Processing. CRC Press, 3.1-3.23. (2001)

9. Zu L., Wang H.K. and Yue F., Artificial neural networks for mobile robot acquiring heading angle. Proceedings of the Third International Conference on Machine Learning and Cybernetics, 26-29 (2004)
10. Julier S.J. and Uhlmann J.K., A New Extension of Kalman Filter to Nonlinear Systems, Proceedings of AeroSense: The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Control, (1997)
11. Scott F., Page.: Multiple-Object sensor Management and optimization, PHD thesis, in the faculty of Engineering, Science and mathematics School of Electronics and Computer science (2009)
12. Pathak Sunil, Turbocharging and Oil Techniques in Light Motor Vehicles, *Research Journal of Recent Sciences*, 1(1), 60-65 (2012)
13. Farshid Hemmati, Influence of Internal Waves on Underwater Acoustic Propagation, *Research Journal of Recent Sciences*, 1(1), 73-76 (2012)
14. Patil Pallavi and Ingle Vikal, Obtaining a high Accurate Fault Classification of Power Transformer based on Dissolved Gas Analysis using ANFIS, *Research Journal of Recent Sciences*, 1(2), 97-99 (2012)
15. Agbo G.A., Ibeh G.F. and Ekpe J.E., Estimation of Global Solar Radiation at Onitsha with Regression Analysis and Artificial Neural Network Models, *Research Journal of Recent Sciences*, 1(6), 27-31 (2012)
16. Nagadeepa N., Enhanced Bluetooth Technology to Assist the High Way Vehicle Drivers, *Research Journal of Recent Sciences*, 1(8), 82-85 (2012)
17. Bailey T., <http://www-personal.acfr.usyd.edu.au/tbailey/software/index.html>, June (2008)

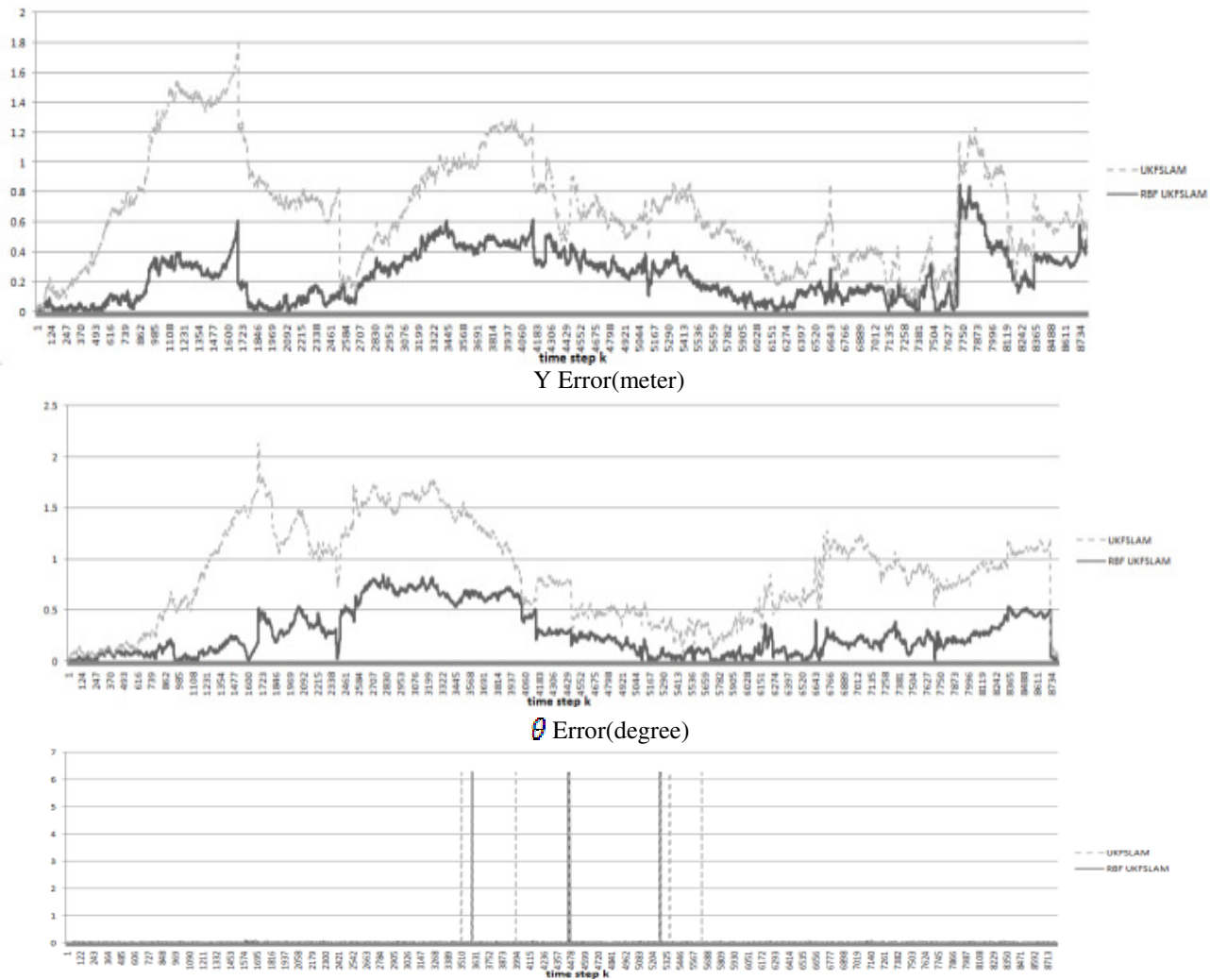


Figure-6
 Navigation errors on widespread map