4th Conference on Technology In Electrical and Computer Engineering (ETECH 2019)
Information and Communication Technology (ICT) Tehran, Iran

Discrete Extended Kalman Filter for the Localization of Noisy Discrete Nonlinear Stochastic State-Space System

Hamed Keshmiri Neghab
Faculty of control engineering
Ferdowsi University of Mashhad
Mashhad, Iran
Keshmirineghab.hamed@mail.um.ac.ir
Tel Number: +989125020161

Naser Pariz
Faculty of control engineering
Ferdowsi University of Mashhad
Mashhad, Iran
n-pariz@um.ac.ir
Tel Number: +989155159162

***Abstract*- This study presents a nonlinear observer design for noisy discrete Nonlinear Stochastic State-Space Systems using the Discrete Extended Kalman Filter to provide accurate Trajectory Tracking. This goes through using system's Jacobian linearization along the current best estimate of its trajectory. The efficiency of Discrete Extended Kalman Filter is implemented on a Trajectory Tracking Problem with sufficiently sampling periods. Here, it is shown that changing principal diagonal of the Initial Estimation Error Covariance Matrix, the trajectory tracking is can be improved.**

Keywords—discrete exended kalman filter; localization; stochastic; jacobian

I. INTRODUCTION

Extended Kalman Filter (EKF) is applied to decrease the noise in corrupted measurements. According to the theory of estimation, the EKF is the nonlinear version of the Kalman Filter which linearizes about an estimate of the current mean and covariance. Sine in the real world, most systems are nonlinear, some attempt was immediately done to apply this method of filtering to nonlinear systems. EKF is used in several systems such as Trajectory Tracking problem and are generally preferred due to its advantages such as accurate target tracking and low cost. In this paper, we simulated a noisy discrete nonlinear stochastic plant model applying Discrete Extended Kalman Filter (DEKF) to track a specific position of autonomous car landmark even if measurement is corrupted by noise(1).

In the real world, controllers typically receive measurements that are corrupted by noise. Moreover, we typically do not or cannot measure all state values. If we want to calculate state values, the only information we have is these noisy measurements and the known inputs. In this situation, we can use a Kalman filter to estimate the state values given noisy sensor measurements.

Accordingly, we implement a DEKF for a noisy discrete nonlinear stochastic state-space model. DEKF calculates the filtered state estimate using only known inputs and noisy measurements of the plant(2).

The DEKF returns the filtered state estimate, which is defined as $\hat{x}(k|k)$. This notation translates as the estimated state vector at time k given all measurements up to and including k. We calculated the filtered state estimate applying a gain matrix $M(k)$ to the difference between the measured output and the estimated output. The DEKF calculates and returns the value of $M(k)$ that minimizes the covariance of the estimation error. This covariance is a matrix $P(k|k)$. The DEKF also calculates the predicted state estimate $\hat{x}(k+1|k)$.

Moreover, with the DEKF, we calculated the predicted state estimate applying a gain matrix $L(k)$ to the difference between the measured output and the estimated output. DEKF calculates and returns the value of $L(k)$ that minimizes the covariance of the prediction estimation error. This covariance is a matrix $P(k+1|k)$.

We can assist the DEKF by specifying the Initial State Estimate xhat(0|–1). This parameter specifies the state values we think the stochastic model returns at the first time step $k = 0$. Providing this function with initial state estimates helps this function converge on the true state values quickly.

We also can specify the Initial Estimation Error Covariance P(0|–1). This parameter defines the covariance of the estimation error at the first time step. A low value of this parameter indicates we have a high degree of confidence in any Initial State Estimate xhat(0|–1) we provide, and vice versa.

II. METHODS

First, we simulate a nonlinear stochastic state-space plant with noise on the states and outputs. Then, we use the DEKF to estimate the position of landmark in distance and angle from the tip of autonomous car. The autonomous car has the reference at the front and, as it moves, the fixed landmark changes position related to the car. Noise corrupts the measurement of the sensors, which in this case affects localization. We use the DEKF to track the landmark position. Since DEKF can give a more accurate measurement of the landmark location (3).

A. Stochastic System Models

Stochastic system models represent the effects of noise on the plant, actuators, and/or sensors. In addition to the state-space matrices *A*, *B*, *C*, and *D*, stochastic models contain the following variables: Vectors w and v represent process noise and measurement noise, respectively. Matrices *G* and *H* relate *w* to the states and outputs, respectively. The following equations define discrete stochastic state-space models:

$$x(k+1) = Ax(k) + Bu(k) + Gw(k) \qquad (1)$$

$$y(k) = Cx(k) + Du(k) + Hw(k) + v(k)$$

Moreover we specify the mean vectors $E\{w\}$ and $E\{v\}$, the auto-covariance matrices $Q = Cov\{w, w\}$, $R = Cov\{v, v\}$, and the cross-variance matrix $N = Cov\{w, v\}$.

$$Q = E\{w.w^T\} - E\{w\}.E^T\{w\} \ (2)$$

$$R = E\{v.v^T\} - E\{v\}.E^T\{v\}$$

$$N = E\{w.v^T\} - E\{w\}.E^T\{v\}$$

B. Simulating Nonlinear Stochastic Models

Before we deploy a controller to a Real Time target, we can test that the controller performs as expected in the presence of noise. To perform this test, we use the Second-Order Statistics Noise Model to generate values of $w(k)$ and $v(k)$ to simulate the behavior of a noisy nonlinear stochastic state-space model. In this situation, we can use the Correlated Gaussian Random Noise to generate Gaussian-distributed values of $w(k)$ and $v(k)$.

C. Using DEKF to Estimate Model States

In the real world, controllers typically receive measurements that are corrupted by noise. Also, we typically cannot measure state values ($\hat{x}$). If we want to calculate all state values, we can use a DEKF. DEKF linearize the nonlinear system around the current estimate by computing a matrix of partial derivatives called a Jacobian matrix. The DEKF evaluates the Jacobian matrix with the current estimated states at each time step (4). The function $f(x, u, k)$ can be used to compute the predicted state from the previous estimate and similarly the function $h(x, u, k)$ can be used to compute the predicted measurement from the predicted state that defined by the following equations:

$$x(k+1) = f(x, u, k) + w(k) \ (3)$$

$$y(k) = h(x, u, k) + v(k)$$

The following equations illustrate the computations that the DEKF performs to produce the estimated states $\hat{x}(k|k)$ of the plant (5).

$$(4) \hat{x}_{k|k} = \hat{x}_{k|k-1} + M_k[y_k - \hat{y}_k]$$

$$\hat{y}_k = h(\hat{x}_{k|k-1}, u_k, k)$$

$$\hat{x}_{k+1|k} = f(\hat{x}_{k|k}, u_k, k)$$

$$M_k = P_{k|k-1} H_x^T(\hat{x}_{k|k-1}, u_k, k) \left[H_x(\hat{x}_{k|k-1}, u_k, k) P_{k|k-1} H_x^T(\hat{x}_{k|k-1}, u_k, k)\right]^{-1}$$

$$P_{k+1|k} = F_x(\hat{x}_{k|k-1}, u_k, k) P_{k|k-1} F_x^T(\hat{x}_{k|k-1}, u_k, k) + Q_k$$
$$-[F_x(\hat{x}_{k|k-1}, u_k, k) P_{k|k-1} H_x^T(\hat{x}_{k|k-1}, u_k, k) + N_k]$$
$$\left[H_x(\hat{x}_{k|k-1}, u_k, k) P_{k|k-1} H_x^T(\hat{x}_{k|k-1}, u_k, k) + R_k\right]^{-1}$$
$$[F_x(\hat{x}_{k|k-1}, u_k, k) P_{k|k-1} H_x^T(\hat{x}_{k|k-1}, u_k, k) + N_k]^T$$

$$F_x(\hat{x}_{k|k-1}, u_k, k) = \left.\frac{\partial f(x_k, u_k, k)}{\partial x_k}\right|_{x_k = \hat{x}_{k|k-1}}$$

$$H_x(\hat{x}_{k|k-1}, u_k, k) = \left.\frac{\partial h(x_k, u_k, k)}{\partial x_k}\right|_{x_k = \hat{x}_{k|k-1}}$$

Where

- $\hat{x}(k|k)$ is the current estimated state at time *k*
- $\hat{y}(k)$ is the estimated output at time *k*
- $\hat{x}(k+1|k)$ is the predicted state estimate calculated at time *k* for the next time step *k*+1
- $P(k+1|k)$ is the prediction error covariance matrix calculated at time step *k*+1
- $M(k)$ is the Kalman Filter gain
- $F_x(\hat{x}(k|k-1), u, k)$ is the Jacobian matrix of $f(x, u, k)$ with respect to x evaluated at $\hat{x}(k|k-1)$
- $H_x(\hat{x}(k|k-1), u, k)$ is the Jacobian matrix of $h(x, u, k)$ with respect to x evaluated at $\hat{x}(k|k-1)$

Fig. 1, is show that the block diagram a DEKF.

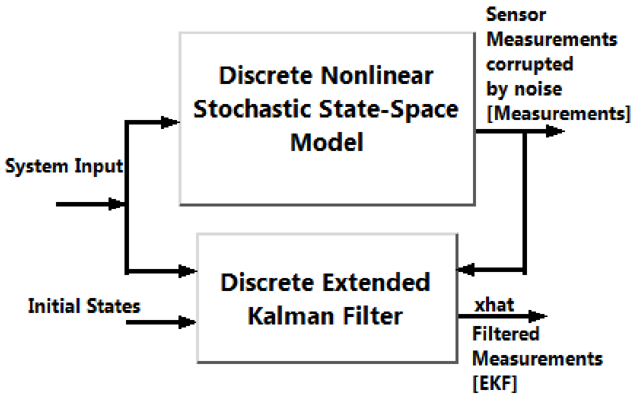


Fig. 1. The DEKF algorithm represented by a block diagram.

To use the DEKF, first we define the Discrete Nonlinear Stochastic Model whose states we want to estimate (6). For instance, consider a plant model of a vehicle moving with constant velocity and whose acceleration is slightly perturbed. Also, assume a sensor measures the range r and the bearing angle θ of the vehicle, both corrupted by additive noise. The following equations describe this plant model.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & .5 \\ 0 & 1 & .5 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ w_1(k) \\ w_2(k) \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} \sqrt{x_1^2(k) + x_2^2(k)} \\ \tan^{-1}\left[\frac{x_2(k)}{x_1(k)}\right] \end{bmatrix}$$

Where x_1 and x_2 are the *x* and *y* position in the Cartesian frame system, x_3 and x_4 are the *x* and *y* velocities, and y_1 and y_2 are the range r and bearing angle θ, respectively. We define the system model by calculating the vector-valued functions $f(x,u,k)$ and $h(x,u,k)$ for the discrete plant model, where:

$$f(x,u,k) = \begin{bmatrix} 1 & 0 & 1 & .5 \\ 0 & 1 & .5 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \quad (6)$$

$$h(x,u,k) = \begin{bmatrix} \sqrt{x_1^2(k) + x_2^2(k)} \\ \tan^{-1}\left[\frac{x_2(k)}{x_1(k)}\right] \end{bmatrix}$$

After we define the plant model, we added noise to the model and construct Discrete Nonlinear Noisy Plant Model to simulate the plant model corrupted by additive noise (7).

The Discrete Nonlinear Noisy Plant simulates the plant dynamics according to the following equations:

$$x(k+1) = f(x,u,k) + w(k) \quad (7)$$

$$y(k) = h(x,u,k) + v(k)$$

Where *w* is the process noise vector and *v* is the measurement noise vector.

Also we specify Initial State Estimate xhat(0|-1)=[30,25,0,0] and $Q = diag[0,0,0.01,0.01]$, $R = diag[0.15,0.01]$, $N = 0$.

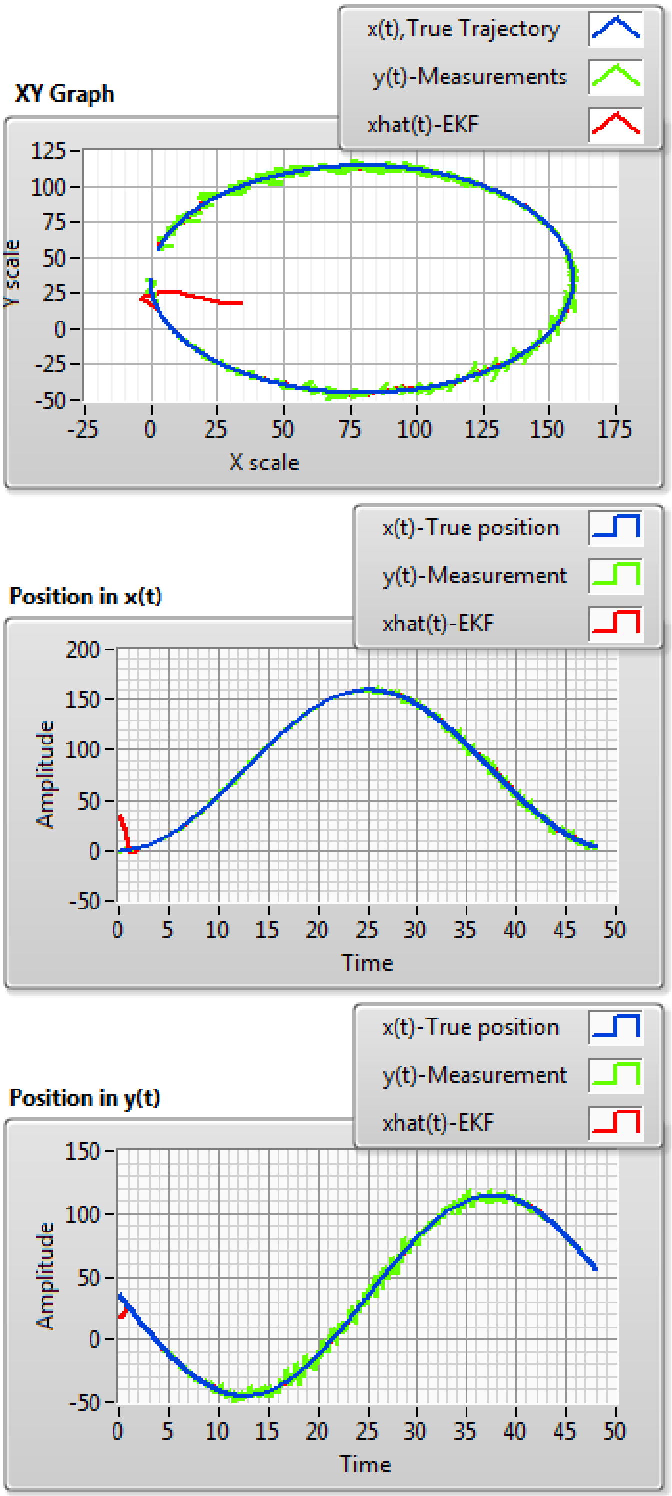
After we define the plant model and add noise to the model, we can use the DEKF to estimate the states of the model. The DEKF takes measurements made on the plant that the Discrete Nonlinear Stochastic Noisy Plant represents.

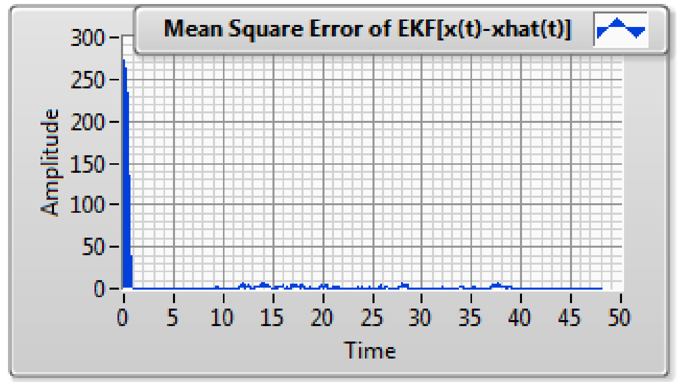
III. RESULT AND DISSCUTION

Generally, the following figures illustrates the implementation of the DEKF for Discrete Nonlinear Stochastic Noisy Plant Model. Also the DEKF calculates the filtered state and output estimates using only known inputs and noisy measurements of the plant. The figures show the resulting graph of the filtered state and output estimates. The state dynamics are linear but the measurements are nonlinear. The x and y positions of the estimated states closely match the true system states.

In this study, we changed the Initial Estimation Error Covariance Matrix to see how it affects the DEKF. Further, changing the Initial Estimation Error Covariance Matrix, its impact on the position of landmark in distance and angle from tip of autonomous car were investigated. The results show that changing principal diagonal of the Initial Estimation Error Covariance Matrix, improve the trajectory tracking.

As shown in fig. 2, with $P(0|-1)$=diag[1,1,0.01,0.01], trajectory tracking gained more accuracy and MSE was significantly decreased as equal as 250 amp. Notably, tracking in angle and position x(t) showed highly accuracy.



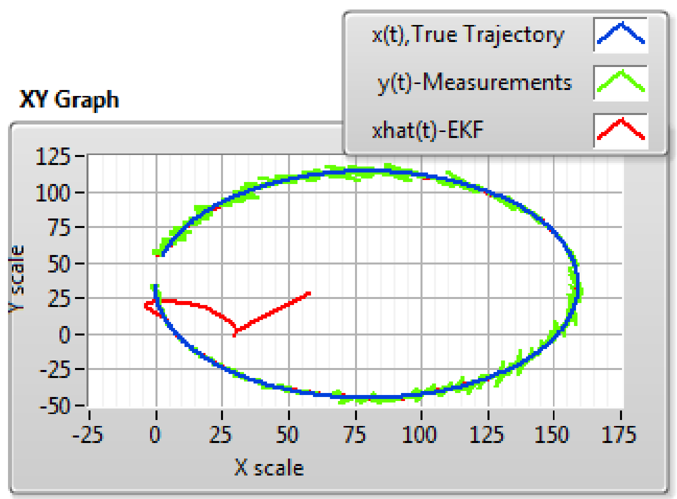


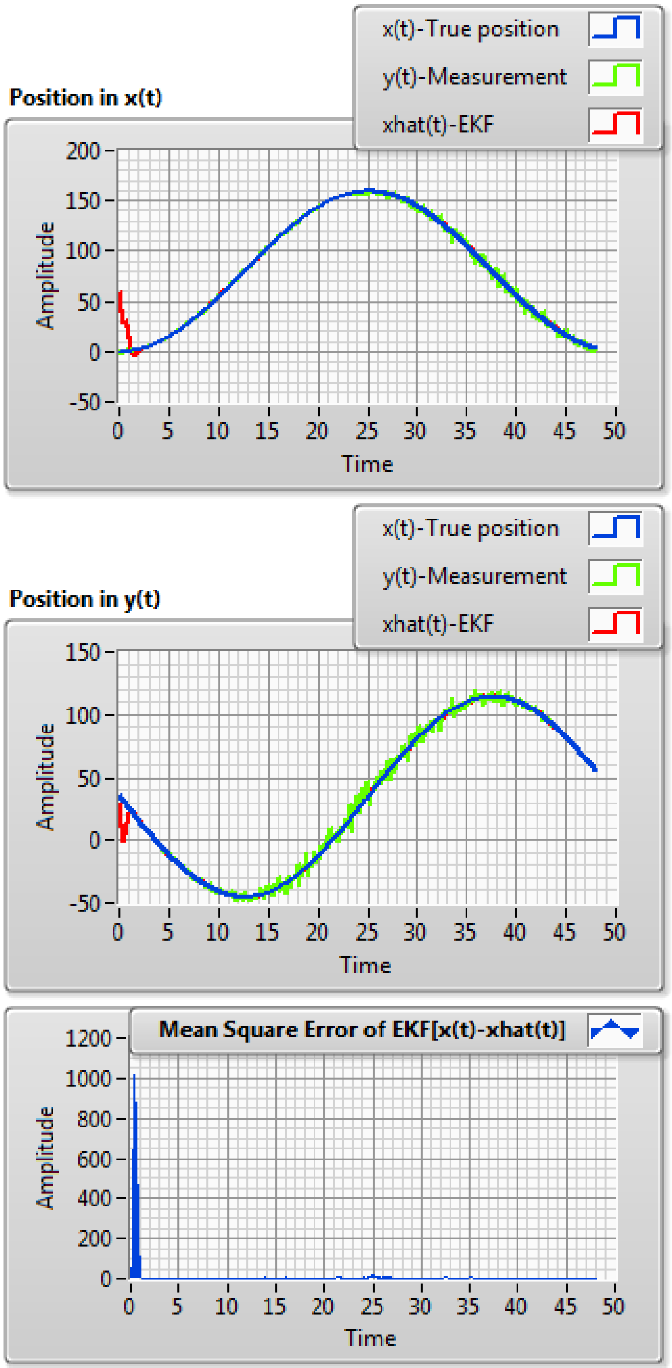
Initial Estimation Error Covariance P(0|-1)

1	0	0	0
0	1	0	0
0	0	0.01	0
0	0	0	0.01

Fig. 2. Localization with DEKF with Initial Estimation Error Covariance matrix $P(0|-1)$=diag[1,1,0.01,0.01]

As shown in fig. 3, with $P(0|-1)$=diag [0.01,0.01,0.01,0.01], trajectory tracking did not have significant accuracy and also MSE was highly increased.



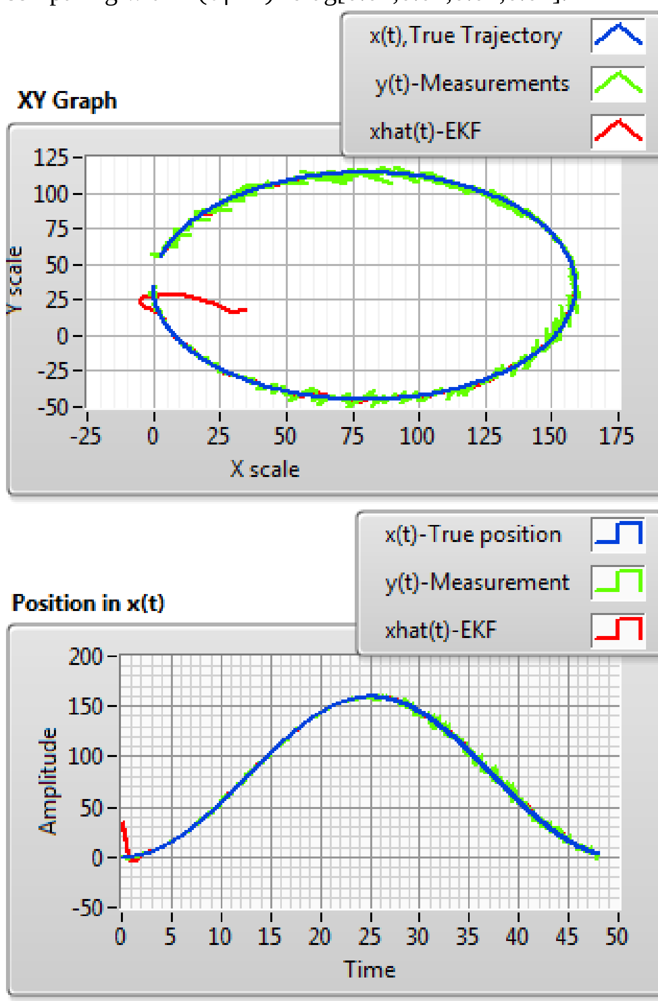


Initial Estimation Error Covariance P(0|-1)

0.01	0	0	0
0	0.01	0	0
0	0	0.01	0
0	0	0	0.01

Fig. 3. Localization with DEKF with Initial Estimation Error Covariance matrix $P(0|-1)$=diag[0.01,0.01,0.01,0.01]

As demonstrated in fig. 4, with $P(0|-1)$= diag [1,1,1,1], the accuracy of trajectory tracking reduced in comparison with $P(0|-1)$=diag [1,1,0.01,0.01] but had more improvement comparing with $P(0|-1)$=diag[0.01,0.01,0.01,0.01].



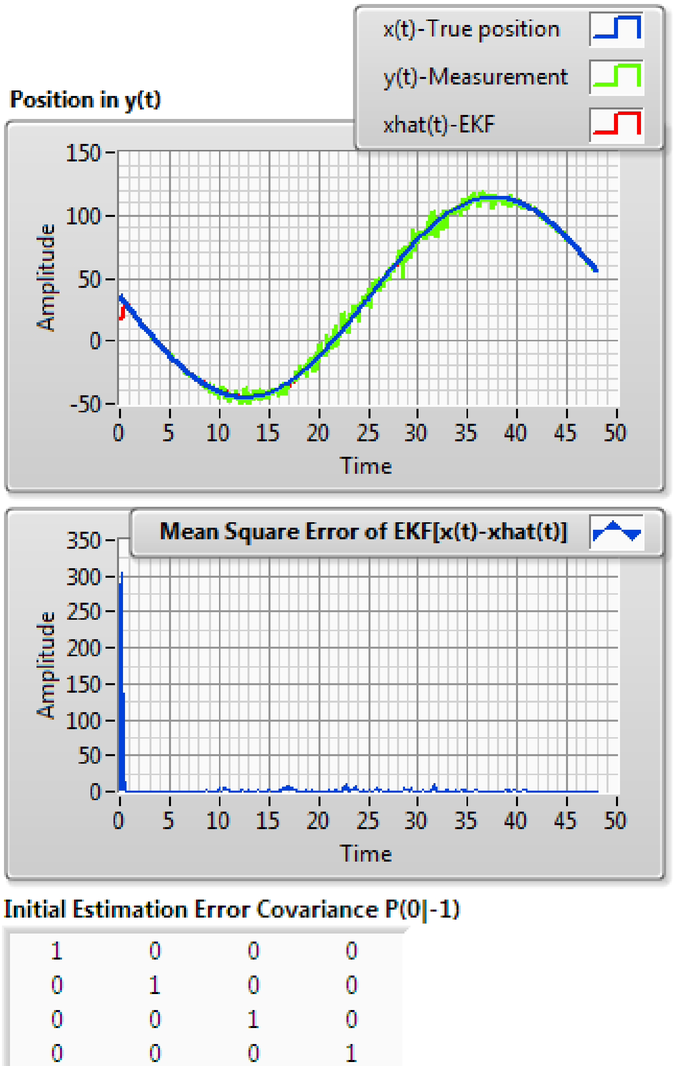


Fig. 4. Localization with DEKF with Initial Estimation Error Covariance matrix P(0|−1)=diag[1,1,1,1]

Results demonstrated that trajectory tracking with DEKF with Initial Estimation Error Covariance matrix $P(0|-1)$=diag[1,1,0.01,0.01] is more accurate than $P(0|-1)$=diag[0.01,0.01,0.01,0.01] and $P(0|-1)$=diag[1,1,1,1]. That is due to the fat that MSE with $P(0|-1)$=diag[1,1,0.01,0.01] is close to 250 amp, that is less than two other matrixes. This can be observed in XY graph.

IV. CONCLUSION

This study presented a performance of DEKF for localization of Trajectory Tracking Problem, using measurements that corrupted by noise. Through simulations, it was observed that, the DEKF technique significantly more preferable and gives more precise result than the Kalman filter and other Trajectory Tracking optimization techniques. Results show that the best method that can be used for Discrete Nonlinear Stochastic Noisy State-Space Systems with Nonlinear or Non differentiable observations, is the DEKF that more flexible and robust and accurate state estimation.

REFRENCES

[1] D. Chen,Y. Yu, L. Xu and X. Liu," Kalman Filtering for Discrete Stochastic Systems with Multiplicative Noises and Random Two-Step Sensor Delays", Discrete Dynamics in Nature and Society, vol., 2015., pp. 1-11, 2015.
[2] J. Pomárico-Franquiza, S. H. Khan and Y.S.Shmaliy, “Combined extended FIR/Kalman filtering for indoor robot localization via triangulation”, Measurement, vol. 50, pp. 236-243, 2014.
[3] L. Jetto, S. Longhi and G. Venturini, “Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots”, IEEE Transactions on Robotics and Automation, vol. 15, pp. 219 – 229, 1999.
[4] A. UmaMageswari, J.J. Ignatious and R. Vinodha, "A comparitive study of Kalman filter, extended kalman filter and unscented Kalman filter for harmonic analysis of the non-stationary signals", International Journal of Scientific & Engineering Research, vol.3, pp.1-9, 2012.
[5] H. Nobahari and A. Sharifi, "A hybridization of extended Kalman filter and Ant Colony Optimization for state estimation of nonlinear systems", Applied Soft Computing, vol.74, pp. 411-423, 2019.
[6] M. Shen, J. Sun, H. Peng and D. Zhao, "Improving localization accuracy in connected vehicle networks using rao-blackwellized particle filters: Theory, simulations, and experiments", IEEE Transactions on Intelligent Transportation Systems, 2018.
[7] J. Qi, K. Sun, J. Wang and H. Liu, "Dynamic state estimation for multi-machine power system by unscented Kalman filter with enhanced numerical stability", IEEE Transactions on Smart Grid, vol.9, pp. 1184-1196, 2018.