# Two New Methods of Kalman Filters Family for Nonlinear SLAM

Farhad Farhadi Yadkuri
Department of Electrical Engineering
Sahand University of Technology
Tabriz, Iran,
Email: farhad.frdi92@yahoo.com

Mohammad Javad Khosrowjerdi
Department of Electrical Engineering
Sahand University of Technology
Tabriz, Iran,
Email: khosrowjerdi@sut.ac.ir

*Abstract*— **In this paper, reducing the linearization error of Kalman filters family for nonlinear simultaneous localization and mapping (SLAM) problem is investigated and two new methods are presented to reduce the linearization error. Inaccuracy of the formula used to calculate the slope of linear approximation of observation function ($H_k$) leads to severe linearization error in Kalman filters family. These new methods named Mean Iterated Extended Kalman Filter (MIEKF) and Mean Stepwise Extended Kalman Filter (MSEKF) lessen the linearization error by revising the formula used to compute $H_k$. Simulation results, utilizing 'Car Park Dataset' demonstrate the effectiveness and reliability of our proposed methods. They illustrate that the best estimation accuracy belongs to MIEKF-SLAM and the method named MSEKF-SLAM comes to the second place in estimation accuracy point of view. In addition, our proposed methods are computationally efficient. Thus, as to both linearization error and computational complexity, MIEKF and MSEKF are more effective than other mentioned Kalman filters in this paper.**

***Keywords- EKF-SLAM; Kalman Filter; slope of linear approximation ($H_k$); linearization error;***

## I. INTRODUCTION

The Simultaneous Localization and Mapping (SLAM) has attracted a lot of attention in the autonomous robot community for two decades. The SLAM algorithm allows robot to operate in an unknown environment by building a map of this environment while simultaneously localizing itself within this map. In the other word, it provides a means to make a robot truly autonomous. SLAM is seen in numerous application domains from landing of spacecraft to operation of underwater robots to remove humans from vehicle operation in particular environments [1-5].

Extended Kalman Filter (EKF) is the most widely used solution to the SLAM problem [6,7] and it uses Taylor series expansion up to first order term to linearize the motion model and observation model. EKF-SLAM suffers from linearization error because of ignoring the second order and higher order terms of the Taylor series expansion and inaccuracy of the formula used to compute the slope of linear approximation of observation function ($H_k$). In order to reduce the linearization error Multilevel-EKF-SLAM and Adaptive Iterated Square-Root Cubature Kalman Filter based SLAM was proposed in [8] and [9] respectively. In [10], it was indicated that Augmented EKF-SLAM increases the accuracy of the feature map from EKF-SLAM by adding systematic parameters to the state vector of EKF-SLAM and decreasing odometry error of the robot. In addition, Unscented Kalman Filter (UKF) based SLAM and Improved UKF-SLAM were presented in [11] and [12]. As these approaches do not use Taylor series approximation to linearize the motion model and observation model, they are more accurate than EKF-SLAM, but their computational complexity is more than EKF-SLAM.

In [13] and [14], Iterated EKF (IEKF) and Stepwise EKF (SEKF) lessen the linearization error of EKF by applying the idea of the iteration of observation updating step and stepwise increment to EKF. In IEKF and SEKF $H_k$ is equivalent to $\partial h / \partial X \big|_{\hat{X}_k^-}$, but MEKF has much better performance in linearization error than IEKF and SEKF because of using the mean of $\partial h / \partial X \big|_{\hat{X}_k^-}$ and $\partial h / \partial X \big|_{X_k}$ as $H_k$ [14].

In this paper, two new methods including Mean Iterated Extended Kalman Filter based SLAM (MIEKF-SLAM) and Mean Stepwise Increment based SLAM (MSEKF-SLAM) are presented to reduce the linearization error. These two methods which are obtained by revising the calculation method of $H_k$ in IEKF-SLAM and SEKF-SLAM and using the mean of $\partial h / \partial X \big|_{\hat{X}_k^-}$ and $\partial h / \partial X \big|_{X_k}$ as $H_k$ have less linearization error than other mentioned approaches in this article. Simulation results with 'Car Park Dataset' validate the effectiveness of proposed methods and they demonstrate that MIEKF is the most suitable filter for nonlinear SLAM among all mentioned Kalman filters on linearization error and computational complexity.

The rest of the paper is organized as follows: the EKF-SLAM problem is described in section II and existing solutions for SLAM are reviewed in section III. In section IV two new methods are proposed. Section V is allocated to simulation done with 'Car Park Dataset' and section VI concludes the paper.

## II. EKF-SLAM ALGORITHM

The goal of SLAM is to estimate the state vector of the vehicle and landmarks $X=[X_v^T, X_l^T]^T$. State vector of the vehicle $X_v=[x_v, y_v, \theta_v]^T$ represents the coordinates and bearing of the vehicle and $X_l=[x_1, y_1, ..., x_i, y_i, ..., x_N, y_N]^T$ is the state vector of the landmarks which $(x_i, y_i)$ represents the coordinates of $i$ th landmarks. In addition, process model is described as:

$$\begin{bmatrix} X_{k,v} \\ X_{k,l} \end{bmatrix} = \begin{bmatrix} f_v(X_{k-1,v}, u_{k-1}) \\ X_{k-1,l} \end{bmatrix} + \begin{bmatrix} W_{k-1,v} \\ 0 \end{bmatrix} \quad (1)$$

where $f_v(.)$ are the motion model of the vehicle. $W_{k-1,v}$ and $u_{k-1}$ represent the motion noise and control input. Besides, observation model is as:

$$Z_k = h(X_k) + V_k \quad (2)$$

$$Z_k = [r_k, \beta_k]^T \quad (3)$$

where $h(.)$ and $V_k$ represent the observation function and observation noise. $r_k$ is the distance between vehicle and landmark and $\beta_k$ represent the bearing of the landmark relative to the vehicle. Motion model and observation model are shown in Fig. 1.

In prediction step the state vector at next time step is predicted as follows:

$$\hat{X}_k^- = \begin{bmatrix} \hat{X}_{k,v}^- \\ \hat{X}_{k,l}^- \end{bmatrix} = \begin{bmatrix} f_v(\hat{X}_{k-1,v}^+, u_{k-1}) \\ \hat{X}_{k-1,l}^+ \end{bmatrix} \quad (4)$$

where $\hat{X}_{k-1,v}^+$ and $\hat{X}_{k-1,l}^+$ represent the state estimate of the vehicle and landmarks and predicted covariance at time $k$ is described as follows:

$$P_k^- = \begin{bmatrix} P_{vv,k}^- & P_{vl,k}^- \\ P_{lv,k}^- & P_{ll,k}^- \end{bmatrix} = \begin{bmatrix} F_{xv} P_{vv,k-1}^+ F_{xv}^T + Q_{k-1} & F_{xv} P_{vl,k-1}^+ \\ P_{lv,k-1}^+ F_{xv}^T & P_{ll,k-1}^+ \end{bmatrix} \quad (5)$$

Where

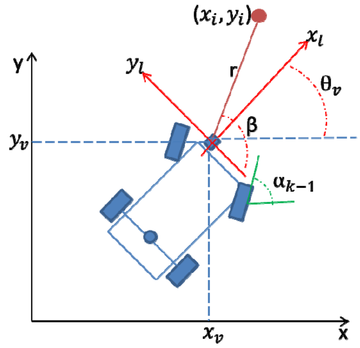


Figure 1. Motion model and observation model

$$P_{lv,k-1}^+ = {P_{vl,k-1}^+}^T \quad (6)$$

where $P_{vv,k-1}^+$ and $P_{vl,k-1}^+$ are the covariance on the state vector of the vehicle and covariance between pose of the vehicle and landmarks at time $k-1$. Moreover, $P_{ll,k-1}^+$ represents the covariance on the landmarks' pose at time $k-1$. $F_{xv} = \partial f_v / \partial x_v \big|_{\hat{X}_{k-1,v}^+}$ and $Q_{k-1}$ indicate the Jacobian of $f_v$ with respect to $X_v$ and covariance of $W_{k-1,v}$ respectively.

In correction step eventual landmarks are first extracted from external sensors like laser range finder [15] and their coordinates are obtained as:

$$\hat{X}_{k,l_{new}}^- = g(\hat{X}_{k,v}^-, Z_k) \Rightarrow \begin{cases} \hat{x}_{k,l_{new}}^- = \hat{x}_{k,v}^- + r_k \cos(\beta_k + \hat{\theta}_{k,v}^- - \frac{\pi}{2}) \\ \hat{y}_{k,l_{new}}^- = \hat{y}_{k,v}^- + r_k \sin(\beta_k + \hat{\theta}_{k,v}^- - \frac{\pi}{2}) \end{cases} \quad (7)$$

where $\hat{X}_{k,l_{new}}^- = [\hat{x}_{k,l_{new}}^-, \hat{y}_{k,l_{new}}^-]^T$ represents the coordinates of extracted landmark and $g(.)$ is the inverse function of $h(.)$. Then, each extracted landmark is associated to a database which has been set up to store all observed landmarks and their iteration numbers. If extracted landmark does not exist in database, it is added to database and its iteration number is set to 1, but iteration number of re-observed landmark is added to 1. If it is lower than a certain number (N), observations of this re-observed landmark is used to correct predicted state vector and its covariance as:

$$H_k = \frac{\partial h}{\partial X}\Big|_{\hat{X}_k^-} \quad (8)$$

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (9)$$

$$\hat{X}_k^+ = \hat{X}_k^- + K_k(Z_k - h(\hat{X}_k^-)) \tag{10}$$

$$P_k^+ = (I - K_k H_k)P_k^- \tag{11}$$

where $K_k$ and $H_k$ are the Kalman gain and the slope of the linear approximation of $h(.)$. $R_k$, $\hat{X}_k^+$ and $P_k^+$ represent the covariance of $V_k$, final estimation of state vector and its covariance at time $k$ respectively. If iteration number is equal to N, corresponding landmark is added to state vector as follows:

$$\hat{X}_{k,new}^- = \left[\hat{X}_k^{-T}, \hat{x}_{k,l_{new}}^-, \hat{y}_{k,l_{new}}^-\right]^T \tag{12}$$

and predicted covariance is extended as follows:

$$P_{k,new}^- = \begin{bmatrix} P_{vv,k}^- & P_{vl,k}^- & P_{vl_{new},k}^- \\ P_{lv,k}^- & P_{ll,k}^- & P_{ll_{new},k}^- \\ P_{l_{new}v,k}^- & P_{l_{new}l,k}^- & P_{l_{new}l_{new},k}^- \end{bmatrix} \tag{13}$$

where

$$P_{l_{new}v,k}^- = G_v P_{vv,k}^- = {P_{vl_{new},k}^-}^T \tag{14}$$

$$P_{l_{new}l,k}^- = G_v P_{vl,k}^- = {P_{ll_{new},k}^-}^T \tag{15}$$

$$P_{l_{new}l_{new},k}^- = G_v P_{vv,k}^- G_v^T + G_V R G_V^T \tag{16}$$

where $G_v = \partial g / \partial X_v \Big|_{\hat{X}_{k,v}^-, Z_k}$ and $G_V = \partial g / \partial V \Big|_{\hat{X}_{k,v}^-, Z_k}$ represent the Jacobian of $g$ with respect to $X_v$ and $V$. Then, predicted state and its covariance are corrected by replacing $\hat{X}_{k,new}^-$ and $P_{k,new}^-$ in Eq. (8) to (11).

## III. Existing Solutions for SLAM Problem

EKF-SLAM suffers from linearization error and different solutions are presented to reduce the linearization error. In this section three solutions including IEKF, SEKF and MEKF are reviewed.

### *A. Iterated Extended Kalman Filter (IEKF)*

IEKF is obtained from iterating the correction step in EKF. In IEKF, observation function is linearized and Kalman gain is calculated repeatedly as:

$$H_{k,i} = \frac{\partial h}{\partial x}\Big|_{\hat{X}_{k,i}^+} \tag{17}$$

$$K_{k,i} = P_k^- H_{k,i}^T (H_{k,i} P_k^- H_{k,i}^T + R_k)^{-1} \tag{18}$$

where $H_{k,i}$ and $K_{k,i}$ represent the slope of the linear approximation of $h$ and Kalman gain for $i$ th re-linearizing step. Then, $\hat{X}_k^-$ and $P_k^-$ are corrected as follows:

$$\hat{X}_{k,i+1}^+ = \hat{X}_k^- + K_{k,i}(Z_k - h(\hat{X}_{k,i}^+) - H_{k,i}(\hat{X}_k^- - \hat{X}_{k,i}^+)) \tag{19}$$

$$P_{k,i+1}^+ = (I - K_{k,i} H_{k,i})P_k^- \tag{20}$$

where $\hat{X}_{k,i}^+$ and $P_{k,i}^+$ represent the state estimate after $i$ re-linearization and its covariance and they start from $\hat{X}_k^-$ and $P_k^-$ ( $\hat{X}_{k,0}^+ = \hat{X}_k^-$ , $P_{k,0}^+ = P_k^-$ ). After N iterations, final state estimation $\hat{X}_k^+$ and its covariance $P_k^+$ are computed.

### *B. Stepwise Extended Kalman Filter (SEKF)*

SEKF [14] is obtained by applying the idea of stepwise increment to EKF. In $i$ th step of correction process the region between $h(\hat{X}_{k,i}^+)$ and $Z_k$ is partitioned into N equal parts and $H_{k,i}$ and $K_{k,i}$ are calculated from Eq. (17) and (18). Then predicted estimate is corrected with following equation [14]:

$$\begin{aligned} \hat{X}_{k,i+1}^+ = \hat{X}_k^- + K_{k,i}(&\frac{i+1}{N}(Z_k - h(\hat{X}_{k,i}^+)) \\ &-H_{k,i}(\hat{X}_k^- - \hat{X}_{k,i}^+)) \end{aligned} \tag{21}$$

Covariance of $\hat{X}_{k,i+1}^+$ is computed with Eq. (20) and Final state estimation $\hat{X}_{k,\mathrm{N}+1}^+$ and its covariance $P_{k,\mathrm{N}+1}^+$ at time $k$ are obtained after N steps.

### *C. Mean Extended Kalman Filter (MEKF)*

MEKF [14] reduces the linearization error of EKF by revising the computation method of $H_k$ and MEKF is different from EKF just on the computation of $H_k$. In EKF $H_k$ is equivalent to $\partial h / \partial X\Big|_{\hat{X}_k^-}$, but in MEKF $H_k$ represents the mean of $\partial h / \partial X\Big|_{\hat{X}_k^-}$ and $\partial h / \partial X\Big|_{X_k}$ [14].

$$H_k = \left(\frac{\partial h}{\partial X}\Big|_{\hat{X}_k^-} + \frac{\partial h}{\partial X}\Big|_{X_k}\right)/2 \tag{22}$$

In addition, for MEKF Kalman gain is computed with Eq. (9) and predicted state $\hat{X}_k^-$ and its covariance $P_k^-$ are corrected with Eq. (10) and (11). Jacobian of $h$ which is obtained from Eq. (33) is as follows:

$$\frac{\partial h}{\partial X} = \begin{bmatrix} A & B & 0 & 0 & 0 & \dots & -A & -B & \dots & 0 & 0 \\ C & D & -1 & 0 & 0 & \dots & -C & -D & \dots & 0 & 0 \end{bmatrix} \quad (23)$$

$$A = \frac{x_v - x_i}{\sqrt{(x_v - x_i)^2 + (y_v - y_i)^2}}, B = \frac{y_v - y_i}{\sqrt{(x_v - x_i)^2 + (y_v - y_i)^2}} \quad (24)$$

$$C = -\frac{y_v - y_i}{(x_v - x_i)^2 + (y_v - y_i)^2}, D = \frac{x_v - x_i}{(x_v - x_i)^2 + (y_v - y_i)^2} \quad (25)$$

Since $X_k$ is unknown, Jacobian matrix of observation function can be calculated with $i$ th landmark's relative coordinates $x_k - x_{k,i}$ and $y_k - y_{k,i}$ as follows [14].

$$x_k - x_{k,i} = -\frac{r_k \times sign(\cos(\beta_k + \theta_k - \pi/2))}{\sqrt{1 + \tan^2(\beta_k + \theta_k - \pi/2)}} \quad (26)$$

$$y_k - y_{k,i} = -\frac{r_k \times sign(\sin(\beta_k + \theta_k - \pi/2))}{\sqrt{1 + \tan^2(\beta_k + \theta_k - \pi/2)}} \times \left|\tan(\beta_k + \theta_k - \pi/2)\right| \quad (27)$$

where

$$sign(\mathrm{A}) = \begin{cases} 1 & A > 0 \\ 0 & A = 0 \\ -1 & A < 0 \end{cases} \quad (28)$$

Besides, $\theta_k$ is unknown in Eq. (26) and (27) and its best estimation $\hat{\theta}^-_{k,v}$ is used instead.

## IV. Two New Solutions for SLAM Problem

Linearization error of EKF-SLAM decreases the estimation accuracy severely and even makes the method to be divergent because of inaccuracy of the formula used to compute $H_k$. IEKF-SLAM and SEKF-SLAM perform better than EKF-SLAM on linearization error because of using the ideas of iterating the correction step and stepwise increment respectively and MEKF-SLAM reduces the linearization error of EKF-SLAM by revising the formula of $H_k$ used in EKF. In [14], it was shown that MEKF-SLAM has less linearization error than three mentioned methods. In this paper, two new methods are proposed to lessen the linearization error by applying the formula of $H_k$ used in MEKF to IEKF and SEKF.

### A. *Mean Iterated Extended Kalman Filter (MIEKF)*

Mean Iterated Extended Kalman Filter is obtained by iterating the correction step in EKF and utilizing the method used in MEKF to compute $H_k$. Thus, MIEKF is different from IEKF just where $H_{k,i}$ is computed and MIEKF can have less linearization error than IEKF because of revising the computation method of $H_{k,i}$. In MIEKF $H_{k,i}$ is equivalent to the mean of $\partial h / \partial X \big|_{\hat{X}^+_{k,i}}$ and $\partial h / \partial X \big|_{X_k}$.

$$H_{k,i} = \left(\frac{\partial h}{\partial X}\Big|_{\hat{X}^+_{k,i}} + \frac{\partial h}{\partial X}\Big|_{X_k}\right)/2 \quad (29)$$

Besides, in the resemblance of IEKF, Kalman gain $K_{k,i}$, state estimate after $i$ th iteration $\hat{X}^+_{k,i+1}$ and its covariance $P^+_{k,i+1}$ are calculated with Eq. (18) , (19) and (20) respectively. In addition, $X_k$ is unknown and Eq. (23) to (28) are used to calculate $\partial h / \partial X \big|_{X_k}$ Similar to MEKF.

In the other word, MIEKF is obtained from iterating the correction step in MEKF, so it has less linearization error than MEKF. As a result, MIEKF has much better performance in comparison with IEKF and MEKF on linearization error because it uses the main idea of IEKF and MEKF simultaneously.

### B. *Mean Stepwise Extended Kalman Filter (MSEKF)*

Applying stepwise increment idea and the approach of computing $H_k$ used in MEKF simultaneously to EKF will lead to new estimator named Mean Stepwise Extended Kalman Filter (MSEKF). Thus, MSEKF is different from SEKF just on the calculation of $H_{k,i}$ and MSEKF can reduce the linearization error of SEKF because of correcting the computation method of $H_{k,i}$ in SEKF. In MSEKF $H_{k,i}$ is described as:

$$H_{k,i} = \left(\frac{\partial h}{\partial X}\Big|_{\hat{X}^+_{k,i}} + \frac{\partial h}{\partial X}\Big|_{X_k}\right)/2 \quad (30)$$

Then, Kalman gain $K_{k,i}$ and state estimate after $i$ th iteration $\hat{X}^+_{k,i+1}$ is calculated with Eq. (18) and (21). Finally, covariance of $\hat{X}^+_{k,i+1}$ is computed with Eq. (20). In the other word, MSEKF is obtained by applying the stepwise increment idea to MEKF. As a result, MSEKF performs better than IEKF and MEKF on linearization error because it uses the main ideas of SEKF and MEKF simultaneously.

In [14], it was indicated that IEKF is more accurate than SEKF, so the idea of iterating the correction step used in IEKF is more effective than stepwise increment idea used in SEKF and MIKEF will be more accurate than MSEKF.

## V. SIMULATION WITH 'CAR PARK DATASET'

Simulation is carried out with 'Car Park Dataset' to evaluate our proposed methods.

### A. Simulation model

The 'Car Park Dataset' was collected by Australian Center for Field Robotics (ACFR) in Sydney with a vehicle shown in Fig. 2 [16]. In this data, ground truth was collected with a GPS and an inertial sensor is used to provide vehicle velocity and steering angle. In addition, a laser range finder is used to measure range and bearing of landmarks relative to the vehicle.

Motion model of the vehicle is described as:

$$\begin{bmatrix} x_{k,v} \\ y_{k,v} \\ \theta_{k,v} \end{bmatrix} = \begin{bmatrix} x_{k-1,v} + \Delta t(v_{k-1}\cos(\theta_{k-1,v}) - \frac{v_{k-1}}{L}\tan(\alpha_{k-1})(a\sin(\theta_{k-1,v}) + b\cos(\theta_{k-1,v}))) \\ y_{k-1,v} + \Delta t(v_{k-1}\sin(\theta_{k-1,v}) + \frac{v_{k-1}}{L}\tan(\alpha_{k-1})(a\cos(\theta_{k-1,v}) - b\sin(\theta_{k-1,v}))) \\ \theta_{k-1,v} + \Delta t\frac{v_{k-1}}{L}\tan(\alpha_{k-1}) \end{bmatrix} + W_{k-1} \tag{31}$$

$$v_{k-1} = \frac{v_{k-1,e}}{1 - \frac{H}{L}\tan(\alpha_{k-1})} \tag{32}$$

where $\Delta t$ represents the sampling time and $v$ is the velocity of the back axle, but $v_e$ which is obtained from inertial sensor represents the velocity of the back left wheel. In addition, the constants L, a, b and H are shown in Fig. 1 and observation model for $i$ th landmark is described as:

$$Z_k = \begin{bmatrix} r_k \\ \beta_k \end{bmatrix} = \begin{bmatrix} \sqrt{(x_{k,v} - x_{k,i})^2 + (y_{k,v} - y_{k,i})^2} \\ \arctan(\frac{y_{k,v} - y_{k,i}}{x_{k,v} - x_{k,i}}) - \theta_{k,v} + \frac{\pi}{2} \end{bmatrix} + V_k \tag{33}$$

### B. Simulation results and analysis

Mean Absolute Error (MAE) and distance between real and estimated state vector are shown in Table 1. $(xe_v, ye_v)$ and $(xe_l, ye_l)$ represent the MAE in the estimation of the vehicle coordinates and landmark coordinates respectively.

$$\begin{bmatrix} xe_v \\ ye_v \end{bmatrix} = \frac{1}{n_t}\sum_{k=1}^{n_t}\begin{bmatrix} |x_{k,v} - x_{k,M}| \\ |y_{k,v} - y_{k,M}| \end{bmatrix} \tag{34}$$

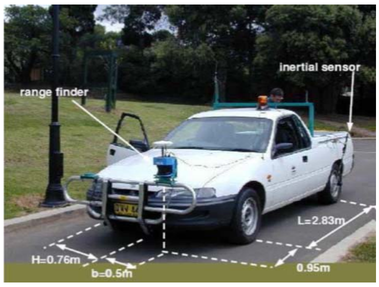


Figure 2. The intelligent vehicle used for data collection

$$X_l\_error = \begin{bmatrix} xe_l \\ ye_l \end{bmatrix} = \frac{1}{n_l}\sum_{i=1}^{n_l}\begin{bmatrix} |x_{i,l} - x_{i,\mathrm{L}}| \\ |y_{i,\mathrm{l}} - y_{i,\mathrm{L}}| \end{bmatrix} \tag{35}$$

where $(x_{k,v}, y_{k,v})$ and $(x_{k,M}, y_{k,M})$ are the estimated coordinates and real coordinates of the vehicle. $(x_{i,l}, y_{i,l})$ and $(x_{i,L}, y_{i,L})$ represent the estimated coordinates and real coordinates of the $i$ th landmark. In addition, $n_t$ is the number of sample data collected with GPS and $n_l$ represents the number of artificial landmarks. Moreover, $d_{v-\max}$ and $d_{v-ave}$ show the maximum and average distance between true path and estimated path. $d_{l-\max}$ and $d_{l-ave}$ represent the maximum and average distance between true coordinates and estimated coordinates of the landmarks.

As shown in Table 1, IEKF-SLAM and SEKF-SLAM improve the estimation accuracy of EKF-SLAM. It means that the idea of iterating the correction step and stepwise increment reduce the linearization error. Moreover, simulation results show that MEKF-SLAM has better performance than EKF-SLAM, IEKF-SLAM and SEKF-SLAM on linearization error. In the other word, revising the formula of computing $H_k$ is more effective than iterating the correction step and stepwise increment. According to Table 1, our proposed approaches are more accurate than other mentioned methods on linearization error. It means that revising the calculation method of $H_k$ in IEKF and SEKF reduces the linearization error of MEKF-SLAM and IEKF-SLAM and SEKF-SLAM. Thus, simulation results shown in Table 1 validate the effectiveness of MIEKF-SLAM and MSEKF-SLAM.

Besides, according to Table 1, it is clear that MIEKF-SLAM performs much better than MSEKF-SLAM on linearization error and it can be claimed that MIEKF-SLAM algorithm comes to the first place among all mentioned methods in terms of estimation accuracy with a view to the simulation results.

The running time of all mentioned algorithms is recorded in Table 2. It should be noted that the running time is not constant for each algorithm and it may vary randomly, so the average running time over 20 simulations of each algorithm is recorded in table 2. It can be concluded from table 2 that EKF-SLAM has the least computational complexity among all mentioned filters in this paper and MEKF-SLAM follows it because the

TABLE I. MAE AND DISTANCE BETWEEN REAL AND ESTIMATED STATE VECTOR (ALL NUMBERS ARE DESCRIBED IN METER)

| Approaches | Vehicle | | | | Landmarks | | | |
|---|---|---|---|---|---|---|---|---|
| | MAE (m) | | Distance (m) | | MAE (m) | | Distance (m) | |
| | $xe_v$ | $ye_v$ | $d_{v-\max}$ | $d_{v-ave}$ | $xe_l$ | $ye_l$ | $d_{l-\max}$ | $d_{l-ave}$ |
| ***EKF-SLAM*** | 0.2225 | 0.1565 | 0.8159 | 0.2916 | 0.3398 | 0.1945 | 0.9483 | 0.4120 |
| ***IEKF-SLAM*** | 0.2194 | 0.1556 | 0.7934 | 0.2887 | 0.3157 | 0.1863 | 0.9370 | 0.3902 |
| ***SEKF-SLAM*** | 0.2201 | 0.1555 | 0.8006 | 0.2893 | 0.3223 | 0.1874 | 0.9445 | 0.3964 |
| ***MEKF-SLAM*** | 0.2068 | 0.1554 | 0.6964 | 0.2782 | 0.2042 | 0.1669 | 0.7928 | 0.2836 |
| ***MIEKF-SLAM*** | 0.1961 | 01562 | 0.6982 | 0.2701 | 0.1433 | 0.1506 | 0.7060 | 0.2263 |
| ***MSEKF-SLAM*** | 0.1966 | 0.1561 | 0.6980 | 0.2704 | 0.1456 | 0.1513 | 0.7110 | 0.2282 |

TABLE II. RUNNING TIME (ALL NUMBERS ARE DESCRIBED IN SECOND)

| **Algorithms** | ***EKF-SLAM*** | ***IEKF-SLAM(N=3)*** | ***SEKF-SLAM(N=3)*** |
|---|---|---|---|
| **Running time** | 1.6278 | 1.7216 | 1.7374 |
| **Algorithms** | ***MEKF-SLAM(N=3)*** | ***MIEKF-SLAM(N=3)*** | ***MSEKF-SLAM(N=3)*** |
| **Running time** | 1.6591 | 1.7833 | 1.8422 |

computation of $H_k$ in MEKF is a little more complex than that of EKF. As to the computational complexity, IEKF-SLAM, SEKF-SLAM, MIEKF-SLAM and MSEKF-SLAM follow MEKF-SLAM in order because they update the state estimate and its covariance N times in one correction step and our proposed methods fall behind IEKF-SLAM and SEKF-SLAM because they use a little more complex method to calculate $H_k$.On the other hand, the running time of our proposed methods is just a little longer than those of other mentioned algorithms and it is small in comparison with the data collecting time (112 s). Thus, with a view to both estimation accuracy and computational complexity, MIEKF and MSEKF are the most appropriate filters for nonlinear SLAM.

## VI. CONCLUSION

In this paper, the reduction of linearization error in Kalman filters family for nonlinear SLAM was investigated and two new methods named MIEKF-SLAM and MSEKF-SLAM was proposed in order to lessen the linearization error. These proposed methods are obtained by revising the calculation method of $H_k$. Simulation results done with ‘Car Park Dataset’ confirmed the less linearization error of our proposed methods than other mentioned methods in this paper and it indicated that MIEKF has less linearization error among all mentioned Kalman filters. Also, these proposed methods are computationally efficient. Thus, with a view to both estimation accuracy and computationally complexity, MIEKF is the most appropriate solution for solving SLAM problem. In addition, computational cost and estimation accuracy increases when the iteration number (N) increases, so it is necessary to set a balance between computational cost and estimation accuracy.

## REFERENCES

[1] Nagappa, Sharad, Palomeras, Narcis, Lee, Chee Sing, Gracias, Nuno, Clark, Daniel E, Salvi, Joaquim, ‘‘Single cluster PHD SLAM: Application to autonomous underwater vehicles using stereo vision’’ in OCEANS-Bergen, 2013 MTS/IEEE, 2013, pp. 1-9.

[2] J. Matthaei, T. Krüger, S. Nowak, and U. Bestmann, "Swarm Exploration of Unknown Areas on Mars Using SLAM," International Micro Air Vehicle Conference and Flight Competition (IMAV), 2013.

[3] F. Le Bars, A. Bertholom, J. Sliwka, and L. Jaulin, "Interval SLAM for underwater robots; a new experiment," NOLCOS, 2010.

[4] A. Mallios, P. Ridao, D. Ribas, and E. Hernández, "Scan matching SLAM in underwater environments," Autonomous Robots, 2013, pp. 1-18.

[5] D. Roy, "Simultaneous Localization and Mapping for an autonomous vehicle using Extended Kalman Filter’’ Indian Institute of Technology, 2004.

[6] A. Burguera, Y. González, and G. Oliver, "RANSAC Based Data Association for Underwater Visual SLAM," in ROBOT2013: First Iberian Robotics Conference, 2014, pp. 3-16.

[7] h S. Oh, M. Hahn, and J. Kim, "Simultaneous Localization and Mapping for Mobile Robots in Dynamic Environments," in Information Science and Applications (ICISA), 2013 International Conference on, 2013, pp. 1-4.

[8] H. Wang, J. Wang, L. Qu, and Z. Liu, "Simultaneous localization and mapping based on multilevel-EKF," in Mechatronics and Automation (ICMA), 2011 International Conference on, 2011, pp. 2254-2258.

[9] Z. Chen, X. Dai, L. Jiang, C. Yang, and B. Cai, "Adaptive Iterated Square-Root Cubature Kalman Filter and Its Application to SLAM of a Mobile Robot," TELKOMNIKA Indonesian Journal of Electrical Engineering, vol. 11, pp. 7213-7221, 2013.

[10] J. G. Kang, W. S. Choi, S. Y. An, and S. Y. Oh, "Augmented EKF based SLAM method for improving the accuracy of the feature map," in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 3725-3731.

[11] f R. Martinez-Cantin and J. A. Castellanos, "Unscented SLAM for large-scale outdoor environments," in Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on, 2005, pp. 3427-3432.

[12] L. Qu, S. He, and Y. Qu, "An SLAM algorithm based on improved UKF," in Control and Decision Conference (CCDC), 2012 24th Chinese, 2012, pp. 4154-4157.

[13] K. Shojaie, K. Ahmadi, and A. M. Shahri, "Effects of iteration in Kalman filters family for improvement of estimation accuracy in simultaneous localization and mapping," in Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on, 2007, pp. 1-6.

[14] s W. Zhou, C. Zhao, and J. Guo, "The study of improving Kalman filters family for nonlinear SLAM," Journal of Intelligent & Robotic Systems, vol. 56, pp. 543-564, 2009.

[15] S. Riisgaard and M. R. Blas, "SLAM for Dummies," A Tutorial Approach to Simultaneous Localization and Mapping, vol. 22, pp. 1-127, 2003.

[16] SLAM summer school, http://www.cas.kth.se/SLAM/, 2002 .