Effects of Iteration in Kalman Filters Family for Improvement of Estimation Accuracy in Simultaneous Localization and Mapping

Khoshnam Shojaie, Kaveh Ahmadi, Alireza Mohammad Shahri, Member, IEEE

Electronic Research Center (ERC), Electrical Engineering Department
Iran University of Science and Technology
Tehran, IRAN
Email: {kh_shojaii|kaveh_ahmadi}@ee.iust.ac.ir, shahri@iust.ac.ir

Abstract— In this paper we investigate the role of iteration in kalman filters family for improvement of the estimation accuracy of states in Simultaneous Localization and Mapping (SLAM). The linearized error propagation existing in kalman filters family can result in large errors and inconsistency in the SLAM problem. One approach to alleviate this situation is the use of iteration in Extended Kalman Filter (EKF) and Sigma Point Kalman Filter (SPKF). We will describe that the iterated versions of kalman filters can increase the estimation accuracy and robustness of these filters against linear error propagation. Simulation results are presented to validate this improvement of state estimate convergence through repetitive linearization of the nonlinear model in EKFSLAM and SPKFSLAM algorithms.

Index Terms— Extended Kalman Filter, Inconsistency, Mobile Robot, Nonlinear Estimation, Sigma Point Kalman Filter, SLAM.

I. INTRODUCTION

The most popular approach for navigation of an autonomous mobile agent is Simultaneous Localization and Mapping (SLAM). This fundamental problem has been investigated by many researchers in mobile robotic community in recent years. A variety of solutions have been introduced by roboticists to the SLAM problem [14]. This problem can be explained as follows: the autonomous vehicle starts from an initial pose without a priori knowledge of the environment and try to get perceptions from its surroundings by its own sensors measurements. The vehicle fuses these measurements to estimate the location of salient features of the environment (mapping process) and simultaneously estimates its own pose in this incrementally built map (localization process) [1], [13].

Kalman filters family has shown its competency to solve this problem that proposed in the literature. The most traditional and dominant approach as a solution of the SLAM problem is Extended Kalman Filter (EKF) which was proposed in a seminal work by Smith, Self, and Cheeseman [12]. Recently, a number of derivative free alternatives to the EKF for nonlinear filtering have been proposed that include Unscented Kalman Filter (UKF), Central Difference Kalman Filter (CDKF) and their square root versions [5], [6] as useful tools for integrated navigation [8]. Some researchers applied these new extensions of the kalman filters family called Sigma Point Kalman Filters (SPKF) to solve the SLAM problem [3], [7]. In spite of superiority of the SPKF over the EKF, both of these filters suffer from analytical and statistical linear error propagation, respectively. The existing linearization in these filters may result in inconsistency in SLAM problem [9], [15]. Because of strong correlation between vehicle and map estimate errors in SLAM algorithm structure, every error caused by linearization of the process and measurement models will be propagated through the whole of the map and vehicle pose. This correlation of the states is essential to maintain consistency of the algorithm [16]. But linear error propagation in EKF and SPKF frameworks will threaten this consistency.

In this paper, we want to investigate if the iteration of linearization in EKF and SPKF can lead to a more robust solution for the SLAM problem. The rest of the paper is structured as follows. Section II summarizes the EKF based SLAM. The Iterated Extended Kalman Filter (IEKF) will be discussed in section III. Then, we will briefly describe the SPKF based SLAM in section IV. In section V, the formulation of the Iterated Sigma Point Kalman Filter (ISPKF) is presented. Then, we compare performance and estimation accuracy of the EKF and SPKF approaches with their iterated versions in section VI. Section VII concludes the paper.

II. EKF BASED SLAM

In this section we summarizes the EKF based SLAM formulation. The structure of a joint vehicle-feature system is as follows. The state vector of the vehicle at time step $k$ is $X_v(k)$. Assuming $N$ stationary features in the environment, the state vector of $i$-th feature is $X_f^i(k)$. The augmented state vector is denoted by $X(k) = [X_v(k)^T , X_f^1(k)^T , ..., X_f^N(k)^T]$, $m = 1,...,N$.
The Extended Kalman Filter estimates the mean and covariance of the posterior probability distribution function (PDF) of the random state variable $X$. Let $\hat{X}(k)$ denote the mean of the posterior PDF at time $k$. The corresponding covariance matrix of the posterior distribution is $P(k)$. The process model for vehicle and features can be written as:

$$\begin{bmatrix}
X(k+1) \\
X_m(k+1)
\end{bmatrix} = f(X(k),u(k+1))$$

$$= \begin{bmatrix}
X_f(k+1) \\
X_m(k+1)
\end{bmatrix} + \begin{bmatrix}
v(k+1) \\
0
\end{bmatrix},$$

(1)

where $f_r(\ldots)$ is the motion model of the vehicle, $u(k)$ is control input and $v(k)$ represents process noise that is zero mean white with covariance $Q(k)$. The EKF fuses the odometry measurements with a sequence of observations from the external sensors with the following observation model:

$$z(k) = h(X(k)) + w(k),$$

(2)

$w(k)$ is zero mean white observation noise with the covariance matrix $R(k)$. Four fundamental stages of the EKF based SLAM are briefly written as follows [1]:

A. Prediction

The mean and covariance of probability distribution of random state variable (RSV) at time step $k+1$ can be predicted from the probability distribution of previous time step as follows:

$$\hat{X}^- = f(\hat{X},u(k+1)),$$

$$\hat{P}^- = \nabla_x f(k) P^+ (k) \nabla_x f^T + Q(k)$$

(3)

(4)

where $\hat{X}$ is the mean of the posterior PDF at time $k$, and $P$ is the corresponding covariance matrix of the posterior distribution at time $k$.

B. Observation

After observation of the $i$th feature, and assuming correct data association, the measurement error is calculated:

$$v(k+1) = z_i(k+1) - \hat{z}_i^- (k+1)$$

(5)

(6)

and its corresponding covariance matrix given by

$$S(k+1) = \nabla_x h \cdot P^- (k+1) \cdot \nabla_x h^T + R(k+1).$$

(7)

C. Update

When a feature is re-observed, the mean and covariance of posterior distribution can be achieved as follows:

$$\hat{X}^+ (k+1) = \hat{X}^- (k+1) + W(k+1) \cdot v(k+1),$$

(8)

where $P^+(k+1) = P^-(k+1) - W \cdot S(k+1) \cdot W^T.$

Kalman gain also given by

$$W(k+1) = \hat{P}^- (k+1) \cdot \nabla_x h^T \cdot S^{-1}(k+1).$$

(9)

(10)

D. Augmentation

When a new feature is detected, its estimate must be appended at the end of the augmented state vector with an augmentation model:

$$\hat{X}_i^+ (k+1) = g_i(\hat{X}_f(k),z(k)).$$

(11)

where $g_i$ is a function to convert the polar observation $z(k)$ to a global Cartesian features location. The covariance matrix also must be updated. Proper calculation of PDF for new detected features is necessary to maintain their consistency [13], [16].

III. ITERATED EXTENDED KALMAN FILTER

The EKF linearizes process and measurement models around the latest state estimates. When models are highly nonlinear, the analytical linearization of models may result in large errors and can cause divergency of the SLAM algorithm. A reasonable solution for this difficulty is iteration of linearization. The Iterated Extended Kalman Filter repeatedly calculates the kalman gain and an intermediate posterior state estimate $\hat{x}_i^k$, where $i$ is the iteration number.

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1},$$

$$\hat{x}_i^{k+1} = \hat{x}_i^k + K_k (z_k - h(\hat{x}_i^k) - H_k (\hat{x}_k - \hat{x}_i^k)).$$

(12)

(13)

The intermediate state estimate starts from the mean of prior PDF (i.e. $\hat{x}_k^0 = \hat{x}_k^-$) and the Jacobian matrix is calculated at the recent value of $\hat{x}_k^i$. After a certain number of iterations, IEKF computes the mean and covariance matrix of the posterior PDF based on the latest value of $\hat{x}_i^k$, $K_k$ and $H_k$. In fact, the IEKF decreases the linearization error by re-linearizing the measurement model and tries to find the best estimate of the state [11]. Note that the IEKF only re-linearizes the observation model. Fig.1 shows re-linearization of the measurement function. Each intermediate state estimate $\hat{x}_k^i$ can be achieved by the true measurement $z_k$ and the calculated line from re-linearization of the function around the previous intermediate state estimate $\hat{x}_k^i$. Note that the IEKF achieve good results if the measurement model is close to linear between the true state $x_k$ and the calculated posterior intermediate state estimate after one linearization. Otherwise, the linearization error increases due to re-linearization and the IEKF fails to improve the state estimate.
IV. SPKF BASED SLAM

This section provides mathematical framework for SPKF based SLAM. By considering assumptions made in section II for the SLAM problem, we define a new augmented state vector that includes the original state vector \( X_k \) and process noise. The new covariance matrix is a block diagonal matrix of \( P_k \) and process noise covariance matrix \( Q_k \):

\[
X^a = [X_k^a, V_k]^T, \quad P^a = \begin{bmatrix} P_k & 0 \\ 0 & Q_k \end{bmatrix}.
\]  (14)

Then, we calculate the sigma points at time \( k \):

\[
\chi^a = [\chi_0^a, \chi_i^a + \sqrt{P^a_k}, \chi_i^a - \sqrt{P^a_k}],
\]  (15)

where \( \gamma \) is a scaling parameter that controls the spread of sigma points around the mean. For efficient calculation of covariance matrix square root, we can use cholesky factorization, if the covariance matrix is positive semi-definite [4], [5]. These sigma points pass through the process model and transformed sigma points will be calculated:

\[
\chi^- (k+1) = f(\chi^+(k), u(k), \chi^v(k)),
\]  (16)

where \( \chi^+ (k) \) is sigma points set for the original augmented state vector and \( \chi^v (k) \) is sigma points set for process noise. The predicted state estimate and its associated covariance matrix can be computed as follows:

\[
\hat{X}_{k+1}^- = \sum_{i=0}^{2N} w_i^m \chi^-_{i, k+1},
\]  (17)

\[
P_{k+1}^- = \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (\chi^-_{i, k+1} \chi^-_{j, k+1})^T.
\]  (18)

The transformed sigma points pass through the measurement model and samples of predicted observation are achieved:

\[
Z^- (k+1) = h(\chi^- (k+1)).
\]  (19)

Then, the predicted measurement and its corresponding covariance matrices can be calculated:

\[
\hat{z}_{k+1}^- = \sum_{i=0}^{2N} w_i^m Z^-_{i, k+1}.
\]  (20)

With assuming correct data association, measurement error and its covariance matrix \( P_{ZZ}^+ \) and cross covariance matrix between state and measurement \( P_{XZ}^+ \) are as follows:

\[
v_{k+1} = z_{k+1} - \hat{z}_{k+1}^-,
\]  (21)

\[
P_{ZZ,k+1}^+ = R + \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (Z^-_{i, k+1} \cdot Z^-_{j, k+1})^T,
\]  (22)

\[
P_{XZ,k+1}^+ = \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (\chi^-_{i, k+1} \cdot Z^-_{j, k+1})^T.
\]  (23)

We supposed that the observation noise is additive. The updated state estimate and its corresponding covariance matrix can be computed:

\[
\hat{X}_{k+1}^+ = \hat{X}_{k+1}^- + K_{k+1} \cdot v_{k+1},
\]  (24)

\[
P_{k+1}^+ = P_{k+1}^- - K_{k+1} \cdot P_{ZZ,k+1}^+ \cdot K_{k+1}^T,
\]  (25)

where kalman gain given by

\[
K_{k+1} = P_{XZ,k+1}^+ \cdot [P_{ZZ,k+1}^-]\^{-1}.
\]  (26)

When a new object \( z_{k+1} \) is detected, the augmenting process is performed easily with the following equations:

\[
X_{k+1}^AUG = [X_{k+1}^a, z_{k+1}^a]^T, \quad P_{k+1}^AUG = \begin{bmatrix} P_{k+1}^a & 0 \\ 0 & R_{k+1} \end{bmatrix},
\]  (27)

\[
X_{i, k+1}^AUG = g_i(\chi_{i, k+1}^AUG),
\]  (28)

\[
\hat{X}_{k+1}^AUG = \sum_{i=0}^{2N} w_i^m \chi_{i, k+1}^AUG,
\]  (29)

Fig. 1. The linearization error will be effectively reduced by re-linearization of the measurement model.

\[
p_{k+1}^- = \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (X^-_{i, k+1} \cdot X^-_{j, k+1})^T.
\]  (18)

With assuming correct data association, measurement error and its covariance matrix \( P_{ZZ}^+ \) and cross covariance matrix between state and measurement \( P_{XZ}^+ \) are as follows:

\[
v_{k+1} = z_{k+1} - \hat{z}_{k+1}^-,
\]  (21)

\[
P_{ZZ,k+1}^+ = R + \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (Z^-_{i, k+1} \cdot Z^-_{j, k+1})^T,
\]  (22)

\[
P_{XZ,k+1}^+ = \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij}^c (\chi^-_{i, k+1} \cdot Z^-_{j, k+1})^T.
\]  (23)

We supposed that the observation noise is additive. The updated state estimate and its corresponding covariance matrix can be computed:

\[
\hat{X}_{k+1}^+ = \hat{X}_{k+1}^- + K_{k+1} \cdot v_{k+1},
\]  (24)

\[
P_{k+1}^+ = P_{k+1}^- - K_{k+1} \cdot P_{ZZ,k+1}^+ \cdot K_{k+1}^T,
\]  (25)

where kalman gain given by

\[
K_{k+1} = P_{XZ,k+1}^+ \cdot [P_{ZZ,k+1}^-]\^{-1}.
\]  (26)

When a new object \( z_{k+1} \) is detected, the augmenting process is performed easily with the following equations:

\[
X_{k+1}^AUG = [X_{k+1}^a, z_{k+1}^a]^T, \quad P_{k+1}^AUG = \begin{bmatrix} P_{k+1}^a & 0 \\ 0 & R_{k+1} \end{bmatrix},
\]  (27)

\[
X_{i, k+1}^AUG = g_i(\chi_{i, k+1}^AUG),
\]  (28)

\[
\hat{X}_{k+1}^AUG = \sum_{i=0}^{2N} w_i^m \chi_{i, k+1}^AUG,
\]  (29)

Fig. 1. The linearization error will be effectively reduced by re-linearization of the measurement model.
\[ P_{k+1}^+ = \sum_{i=0}^{2N} \sum_{j=0}^{2N} w_{ij} \left( \chi_{i,k+1}^{\text{AUG+}} \right) \left( \chi_{j,k+1}^{\text{AUG+}} \right)^T. \]  

(30)

The augmentation model is given by (28). In above equations, the parameters \( w_{ij}^a \) and \( w_{ij}^b \) are scalar positive valued weights. Values of parameters \( w \) and \( \gamma \) are determined by the type of sigma-point approach. These approaches include \textit{Unscented Transformation} and \textit{Central Difference Transformation}.

V. \textbf{ITERATED SIGMA POINT KALMAN FILTER}

The sigma point approach in SPKF can be interpreted as statistical linearization of the nonlinear function by a technique called \textit{Weighted Statistical Linear Regression} (WSLR). The iterated version of the SPKF approach can also improve the state estimate accuracy and filter efficiency in SLAM algorithm. The Iterated Sigma Point Kalman Filter equations can be derived by considering linearized error propagation terms in kalman filter:

\[
\text{cov}(z) = HPH^T, \quad \text{cov}(x,z) = PH^T. \tag{31}
\]

By replacing (31) in (12) and (13), we can compute the kalman gain as follows:

\[
K_k = \text{cov}(\chi, Z)(\text{cov}(Z) + R)^{-1}. \tag{32}
\]

The measurement update equations of the ISPKF can be written as:

\[
x_{k+1}^e = \hat{x}_k + K_k(z_k - h(\hat{x}_k)) - \text{cov}(\chi, Z)P_k^{-1}(\hat{x}_k - \hat{x}_k), \tag{33}
\]

\[
P_{k+1}^- = P_k^- - \text{cov}(\chi, Z)(\text{cov}(Z) + R)^{-1} \text{cov}(\chi, Z)^T. \tag{34}
\]

In fact, the ISPKF approach tries to improve the state estimates by statistical re-linearization of measurement model. In the next section, we compare the performance of EKF and SPKF approaches with their iterated versions for the sake of estimation accuracy and filter efficiency.

VI. \textbf{SIMULATION RESULTS AND DISCUSSION}

In this section, we present some computer simulation results to confirm improving effects of the IEKF and ISPKF relative to non-iterated methods in solution of the SLAM problem. A differential drive vehicle applied for simulation with the following process model:

\[
\begin{bmatrix}
  x_{R,k+1} \\
  y_{R,k+1} \\
  \varphi_{R,k+1}
\end{bmatrix} =
\begin{bmatrix}
  x_{R,k} + \Delta S \cos(\theta_{R,k} + \delta \theta_R/2) \\
  y_{R,k} + \Delta S \sin(\theta_{R,k} + \delta \theta_R/2) \\
  \varphi_{R,k} + \delta \theta_R
\end{bmatrix} +
\begin{bmatrix}
  v_x \\
  v_y \\
  v_{\varphi}
\end{bmatrix}, \tag{35}
\]

where \( \delta \theta_R \) and \( \Delta S \) are defined as:

\[
\delta \theta_R = \frac{\Delta s_R - \Delta s_I}{b}, \quad \Delta S = \frac{\Delta s_R + \Delta s_I}{2}. \tag{36}
\]

The parameter \( b \) is the distance between two wheels of differential drive vehicle. \( \Delta s_r \) and \( \Delta s_l \) are travelled distance for the right and left wheel, respectively. The vehicle equipped with a Polaroid sonar ring to observe simple geometric features of an indoor environment.

The vehicle travels around a 34 \times 29 meters simulated indoor environment and moves at a constant speed along a circular trajectory whose radius is 10 meters. A nearest neighbor standard filter (NNSF) is used for data association in SLAM algorithm. A Technique called \textit{Triangulation Based Fusion} (TBF) is used to extract point features like vertical edges and corners in the environment [10]. The observation model also can be written as follows:

\[
\begin{bmatrix}
  Z_{r,k} \\
  Z_{\theta,k}
\end{bmatrix} = \begin{bmatrix}
  \sqrt{(x_{r,k} - x_{T,k})^2 + (y_{r,k} - y_{T,k})^2} \\
  \frac{\arctan(y_{r,k} - y_{T,k})}{x_{r,k} - x_{T,k}} - \varphi_{r,k}
\end{bmatrix} + \begin{bmatrix}
  w_r \\
  w_{\theta}
\end{bmatrix}, \tag{37}
\]

where \( (x_{T,k}, y_{T,k}) \) is the extracted triangulation point by TBF algorithm. The central difference transformation is also applied to SPKF algorithm. The first simulation is presented for comparison of the EKF approach with the IEKF. The second simulation is also shown to verify the superiority of the ISPKF over the SPKF in SLAM algorithm.

A. \textbf{EKF versus IEKF}

In the first simulation, it took 178 seconds for the vehicle to close a loop. We set a large amount of odometry errors for the vehicle to show the effects of iteration in EKF-SALM so clearly. The estimation error and uncertainty will rise in prior PDF by increasing the odometry errors. Therefore, the kalman gain increases and re-linearization of the measurement model will effectively reduce the errors and uncertainty. Fig. 2 illustrates the path and 2D map of the environment estimated by both extended kalman filter and its iterated version. The estimated path by the IEKF approach is plotted with black dashed line.

The estimated point features by EKF approach are shown by circles and the estimated features by IEKF approach are shown by squares. It is clear that the estimated map by the IEKF is more accurate than EKF. The vehicle pose errors for both filters are also plotted in Fig. 3.

Table I also displays the estimation accuracy in terms of \textit{Mean Squared Error} (MSE) for both filters. It can be seen that the MSE of Vehicle pose for IEKF is less than EKF that shows better accuracy of state estimates in IEKF approach. We repeated this simulation over and over with different initial conditions and all results consistently verified better performance of IEKF algorithm over the traditional EKF.
B. SPKF versus ISPKF

In the second simulation, it took 250 seconds for the vehicle to close a loop. Fig. 4-a illustrates the path and 2D map of the environment estimated by both sigma point kalman filter and its iterated version. The estimated positions of features along with covariance ellipses are plotted by black squares for ISPKF approach. An enlarged portion of the map also plotted in Fig. 4-b to show the superior estimate accuracy of the ISPKF over the SPKF so clearly. We found that if the number of iterations increases in ISPKF, the simulation results will be corrupted. The illustrated results achieved for three iterations in our simulation.

Fig. 4. The path and 2D map of the environment estimated by both SPKF and ISPKF based SLAM (a). An enlarged portion of the map to show differences clearly (b).

VII. CONCLUSION

In this paper, we investigated the effects of iteration in EKF and SPKF frameworks to improve the estimation accuracy in simultaneous localization and mapping. All of simulation results consistently verified better performance and more accurate states estimate of the ISPKF and IEKF approaches over the non-iterated filters.
We highlighted that the re-linearization of the measurement model in non-linear filters can prevent divergence and large estimation errors in the SLAM algorithm. But we must trade off between estimation accuracy and computational cost due to re-linearization. Of course, note that a single program run does not necessarily indicate that the filter is convergent or consistent. The application of some mathematical tools like normalized estimation error squared (NEES) over Monte Carlo runs of the filter seems more reasonable for consistency analysis of applied filters to SALM algorithm.

Future researches may include improving the accuracy of states estimation and consistency in SLAM problem by applying other new Bayesian estimators that will be developed in next researches.

Fig. 5. Comparison of vehicle pose estimate errors for SPKF (solid line) vs. ISPKF (dashed line).

<table>
<thead>
<tr>
<th>MSE</th>
<th>( x_R )</th>
<th>( y_R )</th>
<th>( \varphi_R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF-SLAM</td>
<td>2.7468</td>
<td>0.4121</td>
<td>0.0103</td>
</tr>
<tr>
<td>IEKF-SLAM</td>
<td>0.3012</td>
<td>0.0740</td>
<td>0.0039</td>
</tr>
<tr>
<td>SPKF-SLAM</td>
<td>0.39096</td>
<td>0.13471</td>
<td>0.00313</td>
</tr>
<tr>
<td>ISPKF-SLAM</td>
<td>0.03680</td>
<td>0.02821</td>
<td>0.00116</td>
</tr>
</tbody>
</table>

**REFERENCES**


