# Optimal Mobile Robot Navigation in Unknown Environments using Different Optimization Techniques

Sarah H. Abdulridha[1], Dheyaa Jasim Kadhim[2]

**Authors affiliations:**
1*) Dept. of Computer Eng., Al-Nahrain University, Baghdad, Iraq.
sarah.haider.a@nahrainuniv.edu.iq

2) Dept. of Electrical Engineering, University of Baghdad, Baghdad-Iraq.
dheyaa@coeng.uobaghdad.edu.iq

**Paper History:**

**Received:** 6th Nov. 2023

**Revised:** 24th Dec. 2023

**Accepted:** 2nd Aug. 2024

The delay in this paper's publication resulted from the authors' late submission of their institutional information and their failure to respond to the journal's inquiries.

## Abstract

Mobile robots use simultaneous localization and mapping (SLAM) techniques for generating maps of unknown environments through navigating its. In this work, firstly SLAM technique was considered based on extended Kalman filter (EKF) which it was implemented and evaluated at unknown environments with different number of landmarks to estimate mobile robot's position and build a map for navigated environment at the same time. Then, the detectable landmarks will play an important role in controlling the overall navigation process as well EKF-SLAM technique's performance. After that, three intelligent optimization algorithms are proposed to enhance the performance of the EKF-SLAM trajectory for the mobile robot, these algorithms are: particle swarm optimization (PSO), chaotic particle swarm optimization (CPSO) and genetic optimization (GA). MATLAB simulation results show that CPSO algorithm outperforms PSO and GA algorithms in terms of minimizing the mean square error (MSE1) with increasing the number of landmarks, where MSE1 is the mean square error of EKF-SLAM according to the actual trajectory. The simulation results show also the performance of EKF-SLAM trajectory is better than the performance of the Odometry trajectory and becomes best with using intelligent optimization algorithms.

**Keywords:** Mobile Robot, EKF-SLAM, PSO, GA, CPSO.

## التنقل الأمثل للروبوت المتنقل في بيئات غير معروفة باستخدام تقنيات التحسين المختلفة

ساره حيدر عبد الرضا ، ضياء جاسم كاظم

**الخلاصة:**

تستخدم الروبوتات المتنقلة تقنيات التوطين ورسم الخرائط المتزامنة (SLAM) لإنشاء خرائط لبيئات غير معروفة من خلال التنقل فيها. في هذا العمل، قمنا أولاً بدراسة تقنية SLAM المعتمدة على مرشح كالمان الممتد (EKF) والتي تم تنفيذها وتقييمها في بيئات غير معروفة ذات عدد مختلف من المعالم لتقدير موقع الروبوت المتنقل وبناء خريطة للبيئة الملاحية في نفس الوقت. بعد ذلك، ستلعب المعالم القابلة للاكتشاف دورًا مهمًا في التحكم في عملية الملاحة الشاملة وكذلك أداء تقنية EKF-SLAM. بعد ذلك، تم اقتراح ثلاث خوارزميات تحسين ذكية لتعزيز أداء مسار EKF-SLAM للروبوت المتنقل، هذه الخوارزميات هي: تحسين سرب الجسيمات (PSO)، تحسين سرب الجسيمات الفوضوية (CPSO)، والتحسين الجيني (GA). أظهرت نتائج محاكاة MATLAB أن خوارزمية CPSO تتفوق على خوارزميات PSO وGA من حيث تقليل متوسط مربع الخطأ (MSE1) مع زيادة عدد المعالم، حيث MSE1 هو متوسط مربع الخطأ لـ EKF-SLAM وفقًا للمسار الفعلي. أظهرت نتائج المحاكاة أيضًا أن أداء مسار EKF-SLAM أفضل من أداء مسار Odometry ويصبح أفضل باستخدام خوارزميات التحسين الذكية.

## 1. Introduction

Mobile robots play important roles in new life requirements according to their capabilities and skills in several fields such as salvage search in diverse spaces environments, review of Mars or examine of the sea deepest and several other fields, they have confirmed their aptitudes to traverse in unknown environments. So, the navigation is the main challenge for mobile robots and it has much research care in recent years. The navigation means that the robot should transfer in an environment without colliding landmarks and obstacles, estimate the robot pose itself and then build a map of the environment where it transverses [1].

Navigation can be defined as a collection of self-localization, path planning, and map building issues. The simultaneous localization and mapping (SLAM) is an energetic study area in mobile robotics [2] and has added growing care over the last two decades. The job of the SLAM mechanism is to build a map of an unknown environment whereas estimating the position of the robot according to this map. Since the robots' tracks and maps are together unknown and also the measurements that read from the sensors of the robot are continuously having noise and uncertainties in its locating created by the motion of the robots. Consequently, these errors of the robot's track and map are needed to estimate and correlated [3]. To enhance the robot's navigation numerous approaches occur to address these problems including classical methods such as Particle Filters (PFs) [5] and Kalman Filters (KFs) [6] which they are used to compute combined later distributions over robot position and landmarks. These solutions came out with numerous improvements and developments in have been designated methods such as Rao Blackwellized Particle Filters (RBPFs) and Extended Kalman Filters (EKFs). Furthermore, there is another solution approach such as graph-based algorithms as in [7]. While for stochastic nonlinear systems, the main solution is to suggest using the EKF approach as in works [8][9].

However, our work will consider the estimation performance degradation with EKF SLAM which is mainly caused by the effecting of the noise covariance matrices measurements Q and R, which have a direct impact on the EKF SLAM process. Generally, these noise matrices were firstly adjusted in traditional ways such as using trial and error approaches where they are considered very boring approaches. Therefore, several intelligent optimization algorithms had been proposed to enhance the performance of EKF SLAM based on the position estimation of mobile robots. In recent years, because of the robust and ability to the simultaneous calculations, a number of intelligent approaches, like particle swarm optimization algorithm (PSO), genetic algorithm (GA), and fuzzy logic (FL) algorithm are proposed to solve the navigation problems in robotics.

Authors of [1] proposed suitable solutions for optimizing the navigation of a mobile robot in unknown environments using the PSO technique which examines the solution of space discovery with the proper minimum error value. At [10], Fuzzy based EKF was proposed to deal with the study of diverse Fuzzy membership form performance for Extended Kalman Filter (EKF) based mobile robot navigation to determine the best estimation results for mobile robot and landmarks locations. With [11], a new optimal filter technique genetic algorithm based fuzzy logic (GA-FL) controller was established in its work environment founded on extended Kalman filter (EKF) to enhance the precision of the localization problem in mobile robot and improve the performance of robot localization. In [12], a novel optimal filter named the fuzzy neural network based EKF (FNN-EKF) was proposed to enhance the precision and convergence of the EKF by controlling the noise covariance matrices Q and R for the problem of localization in an unknown indoor environment. In [13], a fuzzy extended Kalman filter (FEKF) technique was proposed and compared to the EKF technique that it had shown that FEKF had better results than EKF and can be further improved if better rules designs are provided. Authors of [14] proposed a particle swarm optimization (PSO) as an alternative approach for the optimization of the covariance matrices Q and R since the PSO algorithm were used a particle collaboration to find the best result. So, our work will consider the PSO algorithm which is an optimization method established by James Kennedy and Russell Eberhart in 1995 [15]. PSO has been used to enhance the precision of the position of the mobile robot pose estimation. The second optimization is chaotic particle swarm optimization (CPSO) which is the optimization of the swarms of chaotic particles proposed in this study since it improves the global search and can achieve the optimal solution with a minimum number of iterations, which depends on the probabilities of chaotic techniques and not on stochastic techniques [16][17]. For performance evaluation purposes, the Genetic algorithms is suggested to apply where it is often used to generate high-quality solutions of optimization and research problems using bio-inspired operators such as mutation, crossing, and selection [18].

The work of this research has been implemented in two steps: firstly, the EKF-SLAM algorithm was implemented and simulated to estimate the robot position and build a map at the same time and comparing the EKF-SLAM trajectory with Odometry trajectory. In the second step, the EKF-SLAM technique is improved by using three different intelligent optimization techniques: PSO, CPSO and GA which also they were implemented and simulated to optimize the navigation at unknown environments. The main contributions come with this research work are: (1) testing EKF-SLAM algorithm for different unknown environments that contain different numbers of landmarks; (2) Enhancing and evaluating the performance of EKF-SLAM technique by using PSO, CPSO and GA where the detectable landmarks will play an important role in controlling the overall navigation process as well EKF-SLAM technique's performance.

## 2. Mathematical Model

To derive the mathematical model proposed for EKF-SLAM, firstly a diagram for the mobile robot was considered that moves with an angle ( $\alpha$ ) where it is the orientation of the mobile robot from the center of rotation, while ($\boldsymbol{w}$) defines here as the width of the mobile robot while $(\boldsymbol{u_t})$ is the control unit which equal to the left and right control movements ($\boldsymbol{l}$ and $\boldsymbol{r}$) of the mobile robot, The motion model can be expressed according to this condition of the left and right control movements. The situation of mobile robot when ($\boldsymbol{r} \neq \boldsymbol{l}$) is shown in Fig. 1 and the situation of mobile robot when ($\boldsymbol{r = l}$) is shown in Fig. 2. The parameter is clearly from Fig. 1 below:

$$\alpha = \frac{r - l}{w} \;, \; R = \frac{l}{\alpha} \qquad ...(1)$$

where $\alpha$ is the radius of mobile robot. Firstly, suppose ($\boldsymbol{l \neq r}$), as shown in Fig. 1. and then the following new position state was produced by:

$$\begin{bmatrix}\bar{x}\\ \bar{y}\\ \bar{\theta}\end{bmatrix} = \begin{bmatrix}x\\ y\\ \theta\end{bmatrix} + \begin{pmatrix}\left(R+\frac{w}{2}\right)(\sin(\alpha+\theta)\text{-}\sin\theta)\\ \left(R+\frac{w}{2}\right)(\text{-}\cos(\alpha+\theta)+\cos\theta)\\ \alpha\end{pmatrix} = \begin{bmatrix}g_1\\ g_2\\ g_3\end{bmatrix} = g(x, y, \theta, l, r) \qquad ...(2)$$

While if $\boldsymbol{r \neq l}$ as shown in Fig. 2. the following new state as the following Eq. 3 where $\alpha$=0

$$\begin{bmatrix}\bar{x}\\ \bar{y}\\ \bar{\theta}\end{bmatrix} = \begin{bmatrix}x\\ y\\ \theta\end{bmatrix} + \begin{bmatrix}l\cos\theta\\ l\sin\theta\\ 0\end{bmatrix} \qquad ...(3)$$

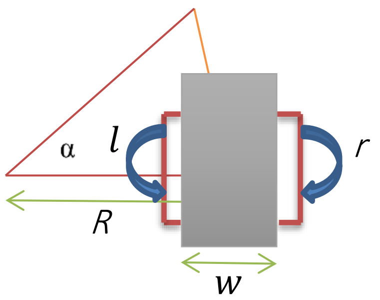


**Figure (1):** The situation of mobile robots when ($\boldsymbol{r \neq l}$).

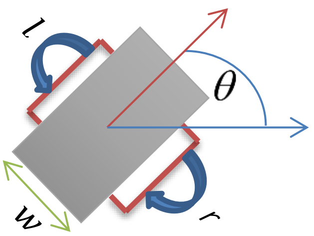


**Figure (2):** The situation of mobile robots when ($\boldsymbol{r = l}$).

where $(\bar{x}, \bar{y}, \bar{\theta})$ is the nonlinear robot position. Then, Extended Kalman Filter (EKF) was proposed to use as a filter in our study. So, EKF uses first-order Taylor series extension to grasp the linearization of nonlinear for both state $(x_t)$and observation ($\boldsymbol{Z_t}$) expressions since it can be expressed by mean $(\mu_t)$ and covariance $(\Sigma_t)$ .Now, the new state of the above state is the next state probability and then the measurement probabilities are ruled by nonlinear functions g and h respectively as in the following Eq. (4).

$$x_t = g(x_{t\text{-}1}, u_t) + \varepsilon_t \qquad ...(4)$$

where $(x_t)$ is the robot position and it is state vector of $(x, y, \theta)$ where g is non-linear function and $\varepsilon_t$ is the Gaussian random vector that models the randomness in the state transition. It is the same dimension of the state vector. Its mean is zero and its noise covariance will be denoted by $R_t$ .

## 2.1 EKF Prediction Stage

The prediction stage can be expressed by mean and covariance as the following:

$$x_t = g(x_{t\text{-}1}, u_t) + \varepsilon_t \qquad ...(5)$$

By take the partial derivative of nonlinear function g and according to the condition of left and right control, when $\boldsymbol{r \neq l}$ the Jacobian matrix is:

$$G_t = \frac{\partial g}{\partial state(x, y, \theta)} = \begin{bmatrix}1 & 0 & \left(R+\frac{w}{2}\right)(\cos(\alpha+\theta)-\cos\theta)\\ 0 & 1 & \left(R+\frac{w}{2}\right)(\sin(\alpha+\theta)-\sin\theta)\\ 0 & 0 & 1\end{bmatrix} \qquad ...(6)$$

$$\text{while if } r = l \text{ then, } G_t = \begin{bmatrix}1 & 0 & -l\sin\theta\\ 0 & 1 & l\cos\theta\\ 0 & 0 & 1\end{bmatrix} \qquad ...(7)$$

## 2.2 EKF Correction Stage

The observation expression $z_t$ can be described as the following:

$$\begin{bmatrix}\bar{x}\\ \bar{y}\\ \bar{\theta}\end{bmatrix} = \begin{bmatrix}x\\ y\\ \theta\end{bmatrix} + \begin{bmatrix}l\cos\theta\\ l\sin\theta\\ 0\end{bmatrix} \qquad ...(8)$$

The distribution of $\delta_t$ is a multivariate Gaussian with zero mean and covariance $Q_t$ . So, Firstly to illustrate the nonlinear function *h*, the mobile robot was assumed stands in the location at the origin point $(x, y)$ while the laser scanner stands at the laser point $(x_e, y_e)$ which was shown in Fig. 3 below :

$$\begin{bmatrix}x_e\\ y_e\end{bmatrix} = \begin{bmatrix}x\\ y\end{bmatrix} + d\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix} \qquad ...(9)$$

The laser scanner was observed a landmark which locates at the point $(x_m, y_m)$.

$$\sqrt{q} = \sqrt{(x_m - x_e)^2 + (y_m - y_e)^2} ,= \sqrt{(\Delta x)^2 + (\Delta y)^2} \qquad ...(10)$$

$$\beta = a\tan\frac{y_m - y_e}{x_m - x_e} - \theta \qquad ...(11)$$

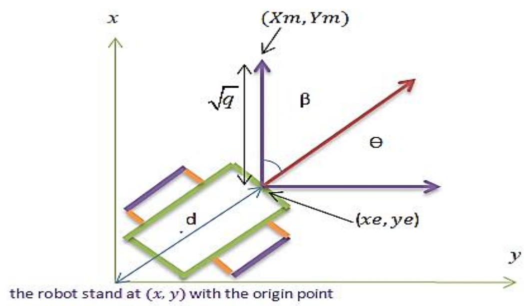


**Figure (3):** The movement of the robot when observed a landmark by a sensor.

Then, the observation function $\hat{z}_t$ can be defined as a vector that equal to nonlinear function h.

$$\hat{z}_t = \begin{bmatrix} \sqrt{q} \\ \beta \end{bmatrix} = h(x, y, \theta) \qquad \ldots(12)$$

Then, the partial derivative of *h* was taken with respect to $(\sqrt{q}, \beta)$ to obtain the Jacobian function:

$$H = \frac{\partial_h}{\partial_{state(\sqrt{q},\beta)}} = \begin{bmatrix} \frac{\partial_{\sqrt{q}}}{\partial_x} & \frac{\partial_{\sqrt{q}}}{\partial_y} & \frac{\partial_{\sqrt{q}}}{\partial_\theta} \\ \frac{\partial_\beta}{\partial_x} & \frac{\partial_\beta}{\partial_y} & \frac{\partial_\beta}{\partial_\theta} \end{bmatrix} \; and \; Q_t = \begin{bmatrix} \sigma^2_{\sqrt{q}} & 0 \\ 0 & \sigma^2_\beta \end{bmatrix} \qquad \ldots(13)$$

Now, the Kalman gain $K_t$ was computed as:

$$K_t = \overline{\sum}_t H_t^T \left( H_t \sum_t H_t^T + Q_t \right)^{-1} \qquad \ldots(14)$$

$Q_t$ is noise covariance. Finally, the mean and covariance of the correction stage are produced as follows:

$$K_t = \overline{\sum}_t H_t^T \left( H_t \overline{\sum}_t H_t^T + Q_t \right)^{-1} \qquad \ldots(15)$$

## 3. Proposed EKF-SLAM Technique

An Extended Kalman Filter was adopted in solving the SLAM problem, and the process of EKF-SLAM technique will be also described by two stages as follows:

### 3.1. EKF-SLAM Prediction Stage

The dimension of EKF-SLAM is considered to be (*2n+3*). So, to map the nonlinear function *g* according to the dimension (*2n+3*) space, the state of the mobile robot will change but the states of landmarks are not changed. Now, to compute the Jacobian matrix $G_t$ and again according to the conditions of mobile robot movement, the Jacobian matrix for the following two cases were obtained:

$$\text{- if } r \neq l \text{ then } G_t = \begin{bmatrix} 1 & 0 & \left(R + \frac{w}{2}\right)\left(\cos(\alpha+\theta) - \cos\theta\right) & 0 & \cdots & 0 \\ 0 & 1 & \left(R + \frac{w}{2}\right)\left(\sin(\alpha+\theta) - \sin\theta\right) & 0 & \cdots & 0 \\ 0 & 0 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \qquad \ldots(16)$$

$$\text{- While if } r = l, \text{ then, } G_t = \begin{bmatrix} 1 & 0 & -l\sin\theta & 0 & \cdots & 0 \\ 0 & 1 & l\cos\theta & 0 & \cdots & 0 \\ 0 & 0 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \qquad \ldots(17)$$

Finally, the mean and covariance of the prediction step of EKF-SLAM are computed according to the previous procedures described above.

### 3.2. EKF-SLAM Correction Stage

One landmark was assumed which situates at the position $(x_m, y_m)$ which was observed by the robot and computes the Jacobin function for one landmark which is denoted by $H_l$ according to individual state vector.

$$H_l = \begin{bmatrix} \frac{\partial_{\sqrt{q}}}{\partial_x} & \frac{\partial_{\sqrt{q}}}{\partial_y} & \frac{\partial_{\sqrt{q}}}{\partial_\theta} & \frac{\partial_{\sqrt{q}}}{\partial_{x_m}} & \frac{\partial_{\sqrt{q}}}{\partial_{y_m}} \\ \frac{\partial_\beta}{\partial_x} & \frac{\partial_\beta}{\partial_y} & \frac{\partial_\beta}{\partial_\theta} & \frac{\partial_\beta}{\partial_{x_m}} & \frac{\partial_\beta}{\partial_{y_m}} \end{bmatrix} = \frac{1}{q}\begin{bmatrix} -\sqrt{q}\Delta_x & -\sqrt{q}\Delta_y & 0 & \sqrt{q}\Delta_x & \sqrt{q}\Delta_y \\ \Delta_y & -\Delta_y & -q & -\Delta_y & \Delta_x \end{bmatrix} \qquad \ldots(18)$$

Now, to obtain the Jacobian function $H_t$ for high dimensional space of EKF-SLAM for *n* landmarks by multiplying the new matrix $F_x$ with previous $H_l$ as

$$F_x = \begin{bmatrix} 1 & \overbrace{0 \quad 0}^{\text{Map for } H_l} & 0 & \cdots & \cdots & 0 & \overbrace{0 \quad 0}^{2j-2} & 0 & \overbrace{\cdots \quad \cdots}^{2N-2j} & 0 \\ 0 & 1 \quad 0 & 0 & \cdots & \cdots & 0 & 0 \quad 0 & 0 & \cdots \quad \cdots & 0 \\ 0 & 0 \quad 1 & 0 & \cdots & \cdots & 0 & 0 \quad 0 & 0 & \cdots \quad \cdots & 0 \\ 0 & 0 \quad 0 & 0 & \cdots & \cdots & 0 & 1 \quad 0 & 0 & \cdots \quad \cdots & 0 \\ 0 & 0 \quad 0 & 0 & \cdots & \cdots & 0 & 0 \quad 1 & 0 & \cdots \quad \cdots & 0 \end{bmatrix} \qquad \ldots(19)$$

Then, the Jacobian of observation function for high dimension was determined by:

$$H_t = H_l F_x \qquad (20)$$

Finally, both mean and covariance for the correction stage of EKF-SLAM are computed by:

$$\mu_t = \bar{\mu}_t K_t \left( z_t - h(\bar{\mu}_t) \right) \text{ and } \Sigma_t = (I - K_t H_t)\overline{\Sigma}_t \qquad (21)$$

The proposed EKF-SLAM technique is simulated and tested in MATLAB software for different number of landmarks. There are three cases of trajectories of the mobile robots are considered: Actual trajectory (ground truth), Odometry robot trajectory (dead reckoning) and EKF-SLAM trajectory (estimated trajectory). Now, because of the circular features of the proposed environment with different number of landmarks, there is an intersection among the measurements that get from run the simulation so it is difficult to see the location of these three cases at each time instance. So this causes inability to notice the efficiency and performance of the proposed EKF-SLAM technique. Therefore, it was needed to determine the distance error between the actual trajectory to Odometry robot trajectory denoted by ($D_1$) and distance error between the actual robot trajectory to EKF-SLAM robot trajectory denoted by ($D_2$) as follows:

$$D_1 = \sqrt{\left(x_{estimate} - x_{actual}\right)^2 + \left(y_{estimate} - y_{actual}\right)^2} \qquad (22)$$

$$D_2 = \sqrt{\left(x_{odo} - x_{actual}\right)^2 + \left(y_{odo} - y_{actual}\right)^2} \qquad (23)$$

where,

$(X_{actual}, y_{actual}), (x_{estimate}, y_{estimate}), (x_{odo}, y_{odo})$ points are defined as the coordinates of the actual robot position and estimated robot position as well as the Odometry robots position. To study the effectiveness of increasing the number of landmarks on the performance of both EKF-SLAM and Odometry trajectories, the mean square error for both previous distance errors $D_1$ and $D_2$ were determined by $MSE_1$ and $MSE_2$ respectively. $MSE_1$ is the mean square error of the distance error between the actual robot trajectory to EKF-SLAM robot trajectory, and $MSE_2$ is the mean square error of the distance error between the actual robot trajectory to Odometry robot trajectory, the expressions of these two metrics are given by:

$$MSE1 = \sum_{i=1}^{k} D_1, \; MSE2 = \sum_{i=1}^{k} D_2 \qquad (24)$$

The number of robot locations is denoted by $k$. Three groups of experiments are tested in MATLAB with a different number of landmarks. According to following expressing, the performance of EKF-SLAM which it is denoted by $Perf.1$ and Odometry performance which it is denoted by $Perf.2$ were computed by:

$$Perf.1 = \left(1 - MSE1\right)\times 100,\ Perf.2 = \left(1 - MSE2\right)\times 100 \tag{25}$$

## 4. Optimization Techniques

At first, the PSO is proposed as an optimization technique. It is the method was built on the conductance of a swarm or colony of bugs with no spearhead, such as ants, bees, the herd of birds and a group of fish. So, the word particle means a bird in a group. Each particle in the swarm depending on its personal experience denoted by ($Pb$) and the group experiences denoted by $(Gb)$. If a particle finds out a good path to the food then, the other particles in the swarm can also follow the good path quickly. It is presumed that the swarm or group is of a specified size denoted by Swarm Size and each particle has a random position in the plan space. The method of PSO represents a random search for finding the maximum or minimum value of the cost objective function until slowly and after several iterations that all birds in a swarm go to the optimal value and based on the following equations [19]. Where the particle has two features, position ($Xpos$) and velocity $(Ve)$.

$$Ve_{i,j}^{n+1} = We \times Ve_{i,j}^{n} + C_1 \times rd_1 \times \left(P_{b_{i,j}}^{n} - Xpos_{i,j}^{n}\right) + C_2 \times rd_2 \times \left(G_{b_j}^{n} - Xpos_{i,j}^{n}\right) \tag{26}$$

$$Xpos_{i,j}^{n+1} = Xpos_{i,j}^{n} + Ve_{i,j}^{n+1} \tag{27}$$

In Eq. (26) $Pb_{i,j}^{n}$ characterizes the personal best $j^{th}$ component of individual, while $Gb_{j}^{n}$ represents $j^{th}$ component of the finest individual of the population up to iteration. The different steps of PSO are as follows [20]. There are different steps of PSO procedures are as follows:

1. Established the constraint (C1 and C2 ) of PSO which are the acceleration factor of PSO, and ( $w_{min}$, $w_{max}$ ,) is then defined as the max and min weight and $rd1$ and $rd2$ are the random numbers.
2. Prepare the inhabitance of the particles with X positions and V velocities and start with the iteration count from n =1.
3. Calculate the objective function of the particles denoted by $F_i^n = f\left(X_i^n\right), \forall i$ and discovery the index of the finest particle C1 and C2, $Wmin$, $Wmax$.
2. Choise $P_{b_i}^{n} = Xpos_i^n, \forall i$ and $G_b^{\ n} = X_b^n$ and calculate the weight $w_e$ as in Eq. (28)

$$w_e = w_{max} - \mathrm{n} \times \left(w_{max} - w_{min}\right) / Maxite \tag{28}$$

3. Upgrade the velocity and the position of particles of both previous Eqs. (26) and (27) and evaluate the new objective function $F_i^{n+1} = f\,(X_i^{n+1})$ for every index $i$. Then discover the finest particle index which is denoted by $b1$.
4. Modernize $(P_b)$ of inhabitants for every index $i$ So, when $F_i^{n+1} < F_i^n$ at this point $Pb_i^{n+1}$ otherwise $Pb_i^{n+1} = Pb_i^n$ and then upgrade $G_b$ of the inhabitants. So, when $F_{b1}^{n+1} < F_b^n$ now $Gb^{n+1} = Pb_{b1}^{n+1}$ then fixed $b$ equal to $b1$,otherwise $Gb^{n+1} = Gb^n$
5. While when $\mathrm{n} < Maxite$ , $\therefore \mathrm{n} = \mathrm{n}+1$ return to the previous six stage otherwise go to the next-stage twelve and
6. Finally typography optimal solution as ( $G_b^{\ n}$ ).

So, the main challenge of using PSO is to enhance the estimations precision and convergence of the EKF by controlling the noise covariance matrices Q and R for the problem of localization in unknown indoor environment. Traditionally, these noise matrices were adjusted in using trial and error approach which leads to many errors in estimations. The wrong choice of these matrices will make the estimation result divergent not convergent or have huge estimate errors. So our work will consider this estimation performance degradation by using a proposed PSO-EKF SLAM approach which it is mainly founded to reduce the effecting of the noise covariance matrices measurements which they have direct impact on EKF-SLAM process. Firstly let's consider the case of increasing the covariance matrix Q which means increasing uncertainties in the mobile robot model and then estimation performance gain will increase. Secondly let's consider the case of increasing covariance matrix R, which causes high noise measurement reading that surely leads to degrade the estimation performance gain. Now, in order to improve the estimation performance, an intelligent approach represented by PSO approach was considered to use to obtain an optimal values of the noise covariance matrices Q and R. So, the noise covariance matrices ($R_t$ and $Q_t$ ) were implemented and simulated in MATLAB with dimensional matrices defined by 3*3 and 2×2 respectively, and they are supposed to be as follows:

$$R_t = diag\left(\left[a\ b\ toRadian\left(c\right)\right]\right).^2 = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & to\ Radian\left(c\right) \end{bmatrix}^2 \tag{29}$$

$$Q_t = diag\left(\left[x \ toRadian\left(e\right)\right]\right).^2 = \begin{bmatrix} x & 0 \\ 0 & toRadian\left(e\right) \end{bmatrix}^2 \quad (30)$$

where a, b, c, x and e, are the parameters. Their values are limited by upper and lower bounds depending on the actual value of its values before the optimization is done. The proposed PSO-EKF SLAM technique is executed in offline mode since the PSO algorithm needs to do numerous iterations to get suitable estimation values with small errors. Through every iteration, the PSO-EKF SLAM algorithm will be simulated and implemented immediately and then accordingly, PSO-EKF SLAM must be executed many times to allowance the enhancement of the parameters Q and R for each measurement [21]. So, the performance of the PSO EKF with various arrangements of Q and R is evaluated by using the MSE1 standard in Eq. (24) between the estimated position of EKF-SLAM and the actual positions of a mobile robot that explained in the simulation results which is used as a minimum objective function.

$$\text{objective function} = \sum_{i=1}^{k} \left(D1 = \sqrt{\left(xEst - xActual\right)^2} + \left(yEst - yActual\right)^2\right) \quad \dots(31)$$

So, Table 1 shows the MSE1 values obtained with our proposed PSO-EKF method with a different number of iterations. After (50) iterations, the MSE1is reduced to be 0.14052. Simulation studies show that the suggested technique provides perfect estimations in the 50 iterations with N = 30 population size and acceleration factors C1 = C2 = 2, and $wmax = 0.9;\ wmin$.that given the optimized parameter a=0.1, b=0.1,c=2.5, d=8, e=27.3.

Then, the chaotic particle swarm optimization (CPSO) was proposed. Here, in order to enrich the search behavior, the chaotic dynamics is incorporated into the above PSO. A well-known logistic equation is employed for the particle swarm optimization. The logistic equation is defined as follows in Eq. (32)

$$Y(k+1) = \lambda Y(k)(1 - Y(k)),\ 0 \le Y\ 0 \le 1 \quad (32)$$

where $\lambda$ is the control parameter, Y is a variable and k=1,2,....S , which K is represent to iteration number of the chaos. To calculate the new weight parameter ( $W_{new}$ ) by sub the $W_e$ of the previous step. 3 in PSO procedure and Eq. (32)

$$Wnew = Y(k+1) * We \quad (33)$$

To improve the overall search capability of PSO, the new inertia weight was needed to add to the speed update equation of PSO , and it became as the follows

$$Ve_{i,j}^{n+1} = We\ new \times Vel_{i,j}^{n} + C_1 \times rand_1 \times \left(p_{bi,j}^{n} - Xpos_{i,j}^{n}\right) + C_2 \times rand_2 \times \left(G_{bj}^{n} - Xpos_{i,j}^{n}\right) \quad (34)$$

The chaotic particle swarm optimization was done in MATLAB with different iterations until it reaches to the minimum objective function of MSE1so Table 2. shows different iteration with MSE1 and with 20 iterations a minimum objective function is obtained to be MSE1=0.1235. So, it was found the CPSO is more accurate and more optimal than PSO because it gives minimum objective function with 20 iterations while the PSO gives the best objective function to be MSE1=0.135 with 50 iterations. to check the effectiveness of CPSO different numbers of environments that contain different landmarks are simulated in MATLAB to enhance the performance of EKF and then it was compared with PSO-EKF. Another technique of optimization was proposed in this work. The genetic algorithm (GA) is used to enhance the performance of EKF and for comparison purposes with the PSO and CPSO techniques. In the same way as the PSO and CPSO optimization, the noise covariance matrices (Q and R) were chosen as optimization parameters for controlling. The parameters that used with the GA technique are followed: the population size is 30; the Probability of exceeding is 0.8; the mutation Probability is 0.01; so, Table 3 shows the generation numbers with their corresponding MSE1 obtained by our proposed approach GA-EKF which show the value of MSE1 is decreased to 0.1531 after 50 generations.

**Table (1):** MSE1 of PSO-EKF SLAM.

| *Iteration* | *MSE1* |
|---|---|
| ***5*** | *0.6583* |
| ***10*** | *0.4521* |
| ***30*** | *0.2841* |
| ***50*** | *0.1405* |

**Table (2):** MSE1 of CPSO-EKF SLAM.

| *Iteration* | *MSE1* |
|---|---|
| ***5*** | *0.5432* |
| ***10*** | *0.3864* |
| ***30*** | *0.2764* |
| ***50*** | *0.1205* |

**Table (3):** MSE1 of GA-EKF SLAM.

| *Iteration* | *MSE1* |
|---|---|
| ***5*** | *0.7462* |
| ***10*** | *0.5341* |
| ***30*** | *0.3872* |
| ***50*** | *0.1526* |

Note that both CPSO-EKF, PSO-EKF give more accurate path estimation compared to the GA method. It is worthy to mention that, our CPSO-EKF method outperforms the PSO-EKF and the PSO-EKF method outperforms the EKF optimized by GA method see the Tables 1, 2 and 3 where the value of MSE1 of PSO that equal 0.14052 of 50 iterations is less than the value of MSE1 in GA that equal 0.1526 with the same iteration where the MSE1 value of CPSO less than MSE1 values of both PSO and GA with 20 iteration which is equal to 0.1205 which supports the supremacy of our methodology.

## 5. Results and Discussion

First, an environment containing one Landmark with a location (x,y) will be implemented in MATLAB and then test the impact of this Landmark on the performance of the EKF-SLAM and our optimization technique PSO, CPSO and GA as shown in Figures. 1, 2, 3 and 4 with three cases of paths of the mobile robots are considered: traditional EKF-SLAM trajectory and the optimization EKF-SLAM trajectory using the three optimization techniques, Odometry trajectory, and actual trajectory.

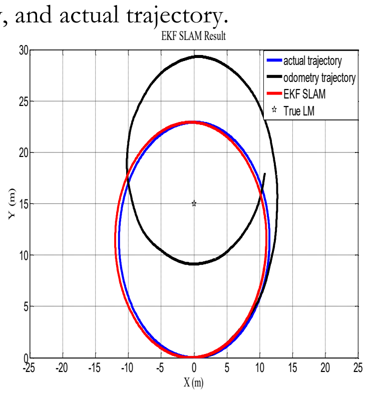


**Figure (1):** The environment one that tested the traditional EKF-SLAM

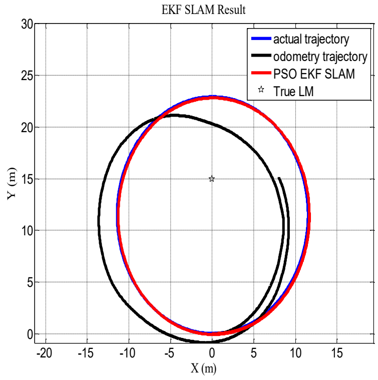


**Figure (2):** The environment one that tested the traditional PSO-EKF-SLAM

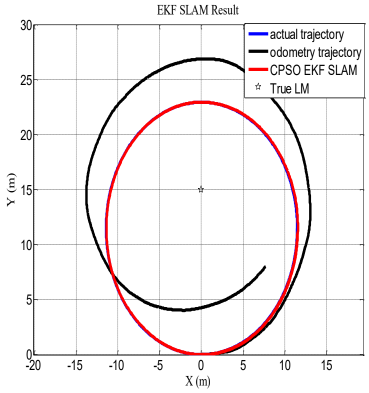


**Figure (3):** The environment one that tested the traditional CPSO-EKF-SLAM

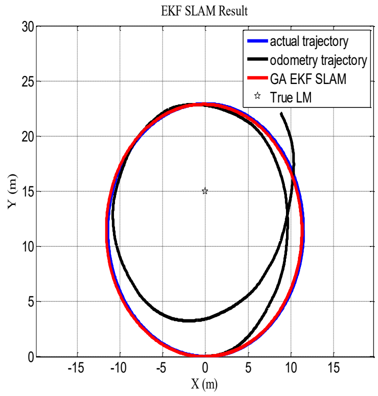


**Figure (4):** The environment one that tested the traditional GA-EKF-SLAM

The distance error D1 and D2 are obtained for the trajectories of EKF-SLAM, PSO EKF-SLAM, CPSO EKF-SLAM and GA EKF-SLAM as shown in Figures. 5, 6, 7 & 8 where MSE1 and MSE2 were computed for both D1 and D2 according to the Eq.(21) and Eq.(22). The results obtained from these figures in this environment are registered in Table 4.

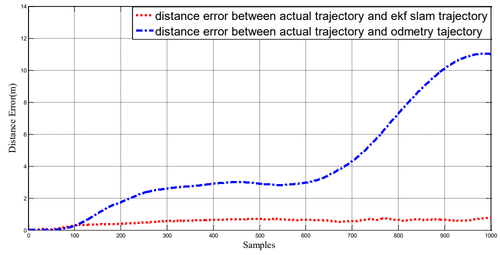


**Figure (5):** D1 and D2 of the first environment for EKF-SLAM.

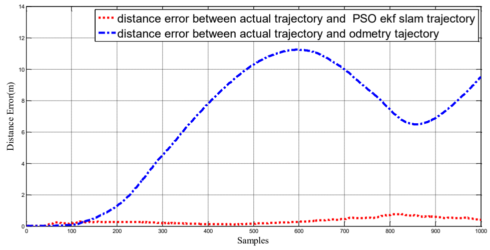


**Figure (7):** D1 and D2 of the first environment for PSO EKF-SLAM

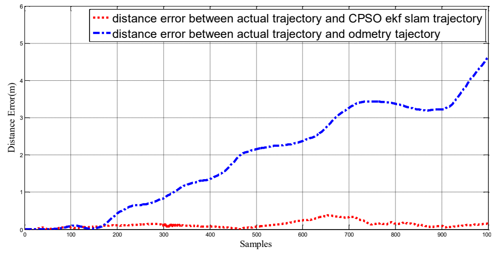


**Figure (7):** D1 and D2 of the first environment for CPSO EKF-SLAM

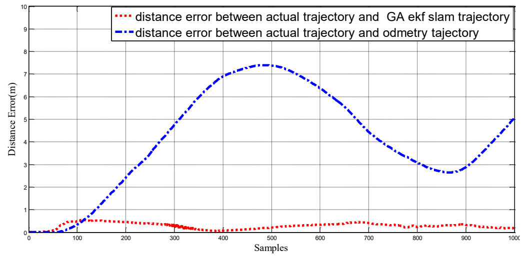


**Figure (8):** D1 and D2 of the first environment for GA EKF-SLAM

Another environment which has eight landmarks organized in rotational form with different location $(x, y)$, the varying of the distribution form of landmarks inside the environment is no impact on the EKF-SLAM performance and other optimization techniques this environment can be expressed as shown Figures 9, 10,11 & 12 in this environment is tested to see the effectiveness of PSO EKF-SLAM, CPSO EKF-SLAM, GA EKF-SLAM and Odometry in the environment which contains eight landmarks and compare it with traditional EKF-SLAM.

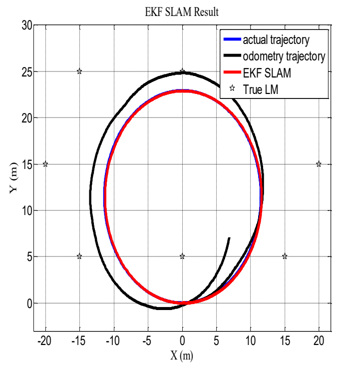


**Figure (9):** The environment that tested the traditional EKF-SLAM

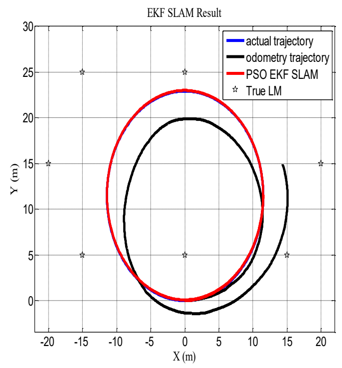


**Figure (10):** The environment that tested the traditional PSO-EKF-SLAM

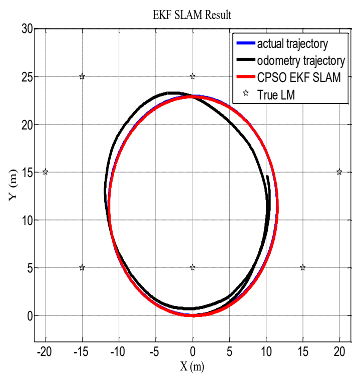


**Figure (11):** The environment that tested the traditional CPSO-EKF-SLAM

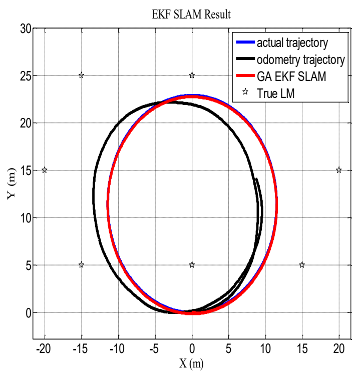


**Figure (12):** The environment that tested the traditional GA-EKF-SLAM

The value of MSE1 remained lower than MSE2 also the performance in the first environment one which contains one landmark is not perfect but the performance of sixth environment that have eight landmarks is very perfect and the performance becomes higher as compared to the previous case when the number of landmarks is increased from one to eight so the distance errors of this environment that test the performance of EKF-SLAM, PSO EKF-SLAM, CPSO EKF-SLAM, GA EKF-SLAM is shown in Figs. 13, 14, 15 &16.

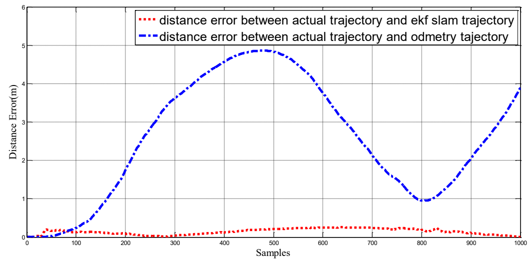


**Figure (13):** D1 Vs. D2 of environment containing six landmarks for EKF-SLAM

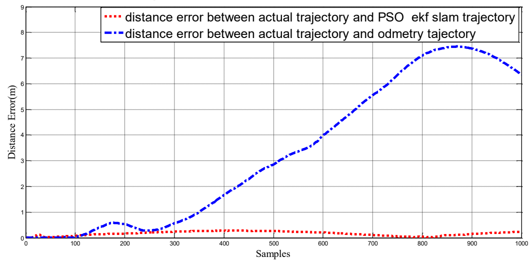


**Figure (14):** D1 Vs. D2 of environment with six landmarks for PSO EKF-SLAM

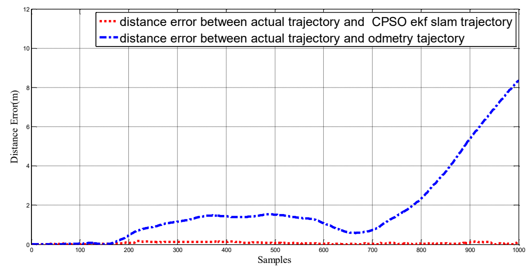


**Figure (15):** D1 Vs. D2 of environment with six landmarks for CPSO EKF-SLAM

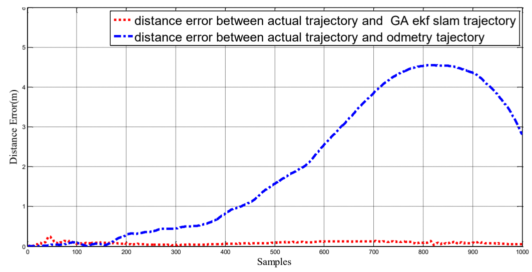


**Figure (16):** D1 Vs. D2 of environment with six landmarks for GA EKF-SLAM

Now, the results of the previous environments are start to describe, in Table 4 the MSE1value of EKF-SLAM is decreased from 0.4023 to 0.1723 when the number of landmarks is increased from one to eight landmarks and also the performance is increased from 64.6 to 86.561.

Also, the MSE1 value of PSO EKF-SLAM is decreased from 0.2923 to 0.0988 and the performance is increased from 70.68 to 90.12 when the landmarks number is increased from one to eight. The PSO results are good when compare it with the GA technique. Also, the performance of PSO EKF SLAM is higher than the performance of EKF SLAM and higher than the performance of the ODOMETRY performance. Further, the MSE1 of CPSO is decreased from 0.2154 to 0.0426 and the performance (perf.1) is increased from 78.4601 to 95.74. The results show CPSO gives good results with few iterations as compare with PSO EKF-SLAM and traditional EKF-SLAM while the MSE1 value of GA in Table 3 is decreased from 0.354 to 0.1344 with increasing the landmarks from one to eight and also the performance is increased from 64.6 to 86.561. The performance of GA is increased when the number of landmarks is increased, the MSE1 is still lower than MSE2 in the whole environment. the performance of GA is lower than the performance of PSO and CPSO. So, The CPSO optimization is more optimal and give the better performance that equal to 95.74 while the performance of PSO equal to 90.12 with the environment that contains eight landmark eight landmarks case.

**Table 4.** Comparison among different optimization techniques

| *Landmark* | *EKF-SLAM Result* | | | | *GA EKF-SLAM Result* | | | |
|---|---|---|---|---|---|---|---|---|
| ***No.*** | ***MSE1*** | ***Perf.1*** | ***MSE2*** | ***Perf.2*** | ***MSE.1*** | ***Perf.1*** | ***MSE2*** | ***Perf.2*** |
| ***1*** | *0.402* | *59.77* | *4.277* | *-327.7* | *0.354* | *64.6* | *3.534* | *-253.4* |
| ***3*** | *0.379* | *62.03* | *5.384* | *-438.4* | *0.313* | *68.65* | *4.168* | *-316.83* |
| ***4*** | *0.295* | *70.44* | *2.281* | *-128.2* | *0.257* | *74.25* | *3.640* | *-264.01* |
| ***5*** | *0.247* | *75.32* | *3.654* | *-364.4* | *0.222* | *77.78* | *3.859* | *-285.88* |
| ***6*** | *0.203* | *79.66* | *4.097* | *-309.7* | *0.157* | *84.22* | *3.123* | *-212.33* |
| ***8*** | *0.172* | *82.77* | *5.187* | *-517.7* | *0.134* | *86.56* | *2.914* | *-191.4* |
| | ***PSO EKF-SLAM Result*** | | | | ***CPSO EKF-SLAM Result*** | | | |
| ***No.*** | ***MSE1*** | ***Perf.1*** | ***MSE2*** | ***Perf.2*** | ***MSE.1*** | ***Perf.1*** | ***MSE2*** | ***Perf.2*** |
| ***1*** | *0.292* | *70.68* | *3.032* | *-203.2* | *0.215* | *78.46* | *2.652* | *-165.21* |
| ***3*** | *0.266* | *73.42* | *2.421* | *-142.2* | *0.182* | *81.77* | *2.695* | *-169.47* |
| ***4*** | *0.213* | *78.70* | *3.580* | *-258.0* | *0.144* | *85.56* | *3.198* | *-219.88* |
| ***5*** | *0.132* | *86.75* | *3.099* | *-209.9* | *0.116* | *88.34* | *2.080* | *-108.02* |
| ***6*** | *0.129* | *87.22* | *2.482* | *-148.2* | *0.088* | *91.13* | *3.215* | *-221.5* |
| ***8*** | *0.098* | *90.12* | *2.529* | *-152.9* | *0.042* | *95.74* | *4.126* | *312.57* |

Figure 17 shows the performance of EKF-SLAM and our optimization techniques are in increase when the number of landmarks is increased while Figure 18 shows the mean square error (MSE1) of EKF-SLAM and our optimization techniques is decreased when the number of landmarks is decreased.

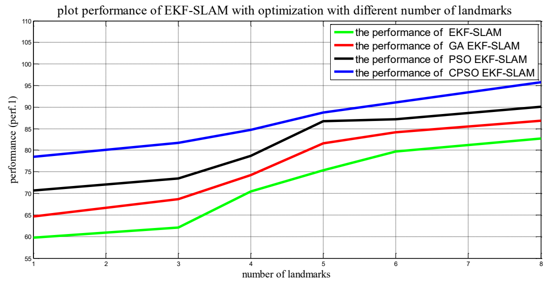


**Figure (17):** The performance of EKF-SLAM and our optimization techniques with different number of landmarks

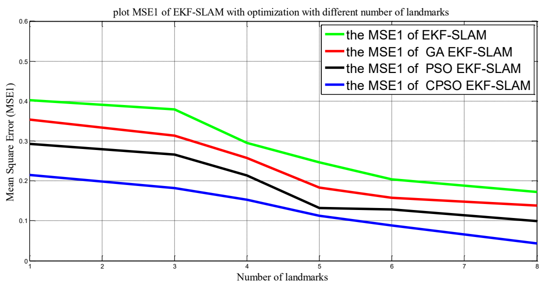


**Figure (18):** The MSE1 of EKF-SLAM and our optimization techniques with different number of landmarks

## 6. Conclusion

The conclusions of this paper in this part will be offered according to their look in this work.

- The first approach proposed in this work to solve the SLAM problem is the Extended Kalman Filter. The simulation results show that the performance of proposed EKF-SLAM is increased when the number of landmarks is increased, where EKF-SLAM algorithm gives an improvement to the path of a mobile robot as compared to the Odometry path according to the actual path of mobile robot, where the performance of EKF-SLAM is higher than the performance of Odometry. The performance of EKF-SLAM for the environment which contains one landmark is 59.77 while the performance for EKF-SLAM for the environment which contains eight landmarks is 82.77.
- Several optimization methods such as GA, PSO, and CPSO based on EKF are used as an optimal solution for the problem of the mobile robot navigation in the unknown environment and to obtain high performance and correct navigation position estimations of a mobile robot. The results show the performance of EKF-SLAM and our optimization techniques are increase when the number of landmarks is increased because the performance is affected by Small noticeable landmarks because the mean square error is decreased when the number of landmarks is increased. It was concluded that the CPSO gives optimal results as compared with GA and PSO with few iterations. The PSO results are good when compare it with the GA technique for 50 iterations. The values of MSE2 and Perf.2 are inaccurate because it depended on the Odometry readings which is inaccurate and give non-optimal path as compared with EKF SLAM and our optimization technique. Also, the performance is affected by Small noticeable landmarks.

## References

1. M. Ghanavati, and S. Ahmadzadeh, " Navigation of mobile robot using the PSO particle swarm optimization," *Journal of Academic and Applied Studies (JAAS)*, 2(1), 32-38, 2012.
2. K.Sun, F.J. Wu, Y.Q.Wang, and L. Sun, "Fading EKF-based adaptive speed observer of induction motor," *Journal of Micromotors*, 43(10), 40-43, 2010.
3. B. Song, J. Xu, and L. Xu, " PSO-based extended Kalman filtering for speed estimation of an induction motor," *Proceedings of the thirty seventh Chinese Control Conference (CCC)*, Wuhan, China, 3803- 3807, Jualy, 2018.
4. H.Chang, W. Yang, H. Zhang, X. Yang, and C.-Y. Chen, "An improved FastSLAM using resmapling based on particle swarm optimization," Proceedings of the eleventh Conference on Industrial Electronics and Applications, ICIEA, IEEE, 229-234,2016.
5. M. Montemerlo, S. Thrun, D. Koller, and Wegbrei. B, " FastSLAM: A factored solution to the simultaneous localization and mapping problem," Proceedings of the National

Conference on Artificial Intelligence. Proceedings of the National Conference on Artificial Intelligence, 593-598, 2010.

**6.** M. W. M. G Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on robotics and automation*, 17(3), 229-241, 2001.

**7.** S. Thrun, and Montemerlo, M. Thrun and M., " The graph SLAM algorithm with applications to large-scale mapping of urban structures," *The International Journal of Robotics Research,* 25(5-6), 403-429, 2006.

**8.** R. Cipriano, and A. "Fuzzy logic based nonlinear Kalman filter applied to mobile robots modelling," *Proceedings of International Conference on Fuzzy Systems IEEE,* 3, 1485 – 1490, 2004.

**9.** M. Barut, S. Bogosyan, and M. Gokasan, "Speed-sensorless estimation for induction motors using extended Kalman filters," *IEEE Transactions on Industrial Electronics*, 54(1), 272-280, 2007.

**10.** H. Ahmad, and N.A Othman, "Fuzzy Logic Based EKF for Mobile Robot Navigation: An Analysis of Different Fuzzy Membership Functions," *Pertanika Journal of Science and Technology*, 25 (S), 189 – 198, , June, 2017.

**11.** H. Wang, W. Liu, F. Zhang, S.X. Yang, and L. Zhang, "A GA-fuzzy logic based extended Kalman filter for mobile robot localization," *Proceedings of the twelfth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Zhangjiajie, China, 319-323, August, 2015.

**12.** J. Ni, C. Wang, X. Fan, and S.X Yang, " A bioinspired neural model based extended Kalman filter for robot SLAM," *Journal of Mathematical Problems in Engineering*, Article ID 905826, 1-11, 2014.

**13.** H. Ahmad, N. Othman, S. Razali, and M.R. Daud, "FEKF Estimation for Mobile Robot Localization and Mapping Considering Noise Divergence," *ARPN Journal of Engineering and Applied Sciences,* 11(6), 3962-3967, 2016.

**14.** N. Kaur, and A. Kaur, "Comparison of hybrid HOD-GSA, HOD and PSO for the tuning of extended Kalman filter," *Proceedings of the fifth International Conference on Reliability, Infocom Technologies and Optimization, ICRITO: Trends and Future Directions*, Noida, India, 107-113, September, 2016.

**15.** M.N Alam, "Particle swarm optimization: Algorithm and its codes in matlab," *ResearchGate*, 1-10, 2016.

**16.** W. Wu, and H. Wang, " Chaotic particle swarm optimization algorithm for hub and spoke systems with congestion," *The Open Automation and Control Systems Journal*, 6(1), 609-615, 2014.

**17.** Z. Ma, X. Yuan, S. Han, D. Sun, and Y. Ma, "Improved Chaotic Particle Swarm Optimization Algorithm with More Symmetric Distribution for Numerical Function Optimization," *Symmetry*, 11(7), 876, 2019.

**18.** C.Lamini, S. Benhlima, and A. Elbekri, "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Computer Science,* 127(C),180–189, 2018.

**19.** K. Jajulwar, and A. Deshmukh, "Design of Mobile Robot Navigation system using SLAM and Adaptive Tracking Controller with Particle Swarm Optimization for Indoor Environment Monitoring" *IOSR Journal of Computer Engineering (IOSR-JCE)*,17(6), 59-63, 2015.

**20.** M.N. Alam, B. Das, and V. Pant, "A comparative study of metaheuristic optimization approaches for directional overcurrent relays coordination," *Journal of Electric Power Systems, Research*, 128, 39-52, 2015.

**21.** Y. Laamari, K. Chafaa, and B. Athamena, "Particle swarm optimization of an extended Kalman filter for speed and rotor flux estimation of an induction motor drive," *Journal of Electrical Engineering*, 97(2),129-138. 2015.