EKF-SLAM with Occupancy Grid: A Comprehensive Framework for Autonomous Navigation in Complex Environments

Yasheerah Yaqoot, Malaika Zafar, Muhammad Ahsan Mustafa, Amber Batool, Roohan Ahmed Khan
Engineering Systems (Robotics)
Skolkovo Institute of Science and Technology
Moscow, Russia

***Abstract*—The lack of synchronization in sensor data streams presents a major barrier for simultaneous localization and mapping (SLAM) in complex situations. This paper solves this SLAM problem on Victoria Park dataset that consists of GPS, odometry and LiDAR data with gaps by utilizing essential localization tool i.e., Extended Kalman Filter (EKF). In the face of data association, a unique method that uses Mahalanobis distances in cost matrix for data linkage guarantees reliable landmark tracking. Additionally, a novel contribution is offered: creating Occupancy Grid Maps using EKF SLAM results, allowing accurate navigation in dynamic situations. The efficiency of our technique is confirmed by experimental findings, which also show its potential applications to real-world autonomous systems.**

***Index Terms*—EKF, SLAM, Occupancy Grid, Data Association, Localization**

I. Introduction

One of the main challenges in robotics research is autonomous navigation in complex situations. To tackle this problem, this research presents an extensive architecture that combines occupancy grid mapping and Extended Kalman Filter (EKF) SLAM. Our approach provides a robust solution for real-time navigation tasks by integrating state estimates and environment representation. We describe our technique, give an overview of relevant work, show the outcomes of our experiments, and talk about the consequences and future directions. By improving autonomy in robotic systems, this paper helps make navigation in difficult areas safer and more effective.

This study uses the Extended Kalman Filter (EKF), a crucial localization method, to solve the SLAM problem on the Victoria Park dataset shown in Fig. 1, which comprises of GPS, odometry, and LiDAR data with gaps [6].Because of its diverse topography, the Victoria Park dataset is a demanding benchmark for testing navigation algorithms and offers a realistic setting for evaluating our method's efficacy.

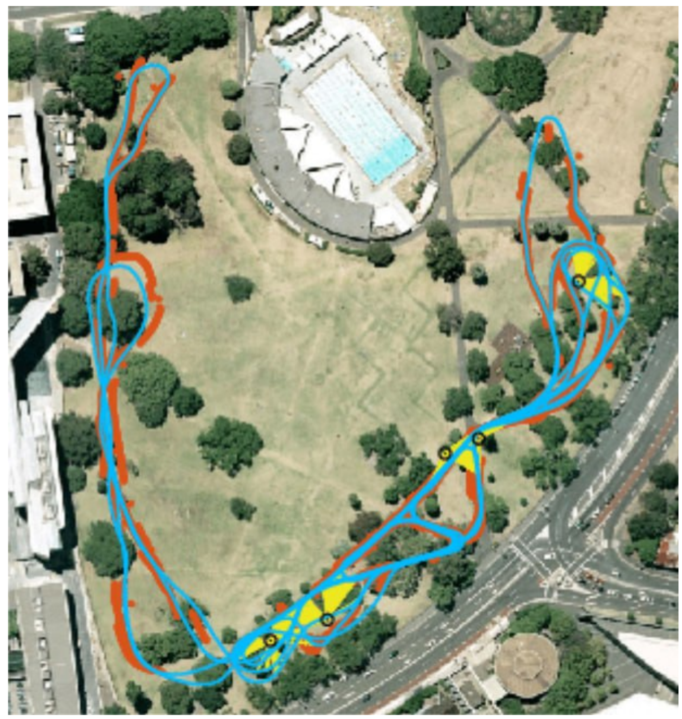

Fig. 1. Trajectory visualized on an aerial view of Victoria Park

II. Literature Review

One of the most important developments in autonomous systems, especially for vehicle state estimation, is the combination of EKF SLAM and occupancy grid mapping. Prominent investigations, including the utilization of EKF SLAM on the Victoria Park dataset, have exhibited the efficacy of this methodology in practical situations [1]. In particular, this research demonstrates how EKF SLAM approaches are resilient and scalable when handling complex and dynamic environments.

Conversely, occupancy grid mapping provides a foundational method for robotics environment representation. The significance of merging EKF SLAM with occupancy grid mapping is further highlighted by recent research [2]. This kind of integration not only improves the precision of vehicle localization, but it also makes reliable navigation easier in difficult terrain.

This combination of approaches not only improves the state-of-the-art in navigation but also lays the foundation for further developments in autonomous systems, leading to safer and more effective robotic navigation in a variety of situations.

III. Methodology

A. Network Model

The methodology of this project is summarized in Fig. 2. We have multiple datasets from GPS, LiDAR and odometry. When conducting multiple events the process follows a sequential procedure. It checks what type of dataset is available at a specific event. If odometry data is available, the trajectory is propagated. However, in the presence of GPS data, GPS update occurs, while in the case of LiDAR data, the data association is performed followed by LiDAR update. This continues for all the events. Once all the events are executed, the updated vehicle trajectory from EKF-SLAM is obtained with all the landmarks.

By using LiDAR data and EKF-SLAM estimates, occupancy grid map was constructed.

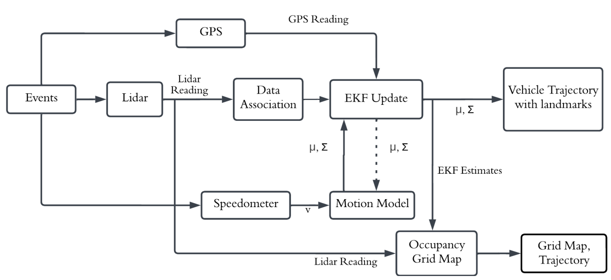

Fig. 2. Network Model

IV. Implementation

A. Motion Model

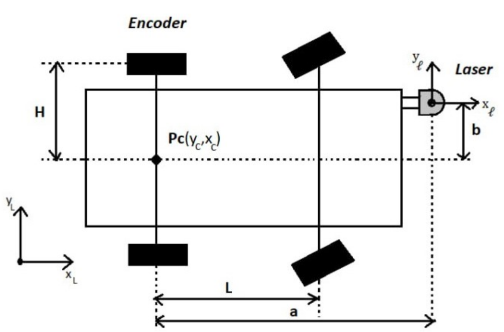

Fig. 3. Vehicle Kinematics used in Victoria Park dataset

The rates of position (x_c, y_c) and angle ϕ are determined using the Ackerman Model [3]:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} v_c \cos(\phi) \\ v_c \sin(\phi) \\ v_c \tan(\alpha) \end{bmatrix} \tag{1}$$

where α is the steering angle.

Now, we translate this model to the GPS and laser point, let us call this point (x_v, y_v)

$$\begin{bmatrix} x_v \\ y_v \end{bmatrix} = \begin{bmatrix} x_c + a\cos(\phi) - b\sin(\phi) \\ x_c + a\sin(\phi) + b\cos(\phi) \end{bmatrix} \tag{2}$$

The velocity v_e is measured with an encoder located in the back left wheel. This velocity is translated to the center of the axle v_c with the following equation:

$$v_c = v_e \frac{1 - \tan(\alpha) H}{L} \tag{3}$$

For our vehicle [3]:

$$L = 2.83m, H = 0.76m, b = 0.5m, a = 3.78m$$

Finally the discrete model in global coordinates can be approximated with the following set of equations [3]:

$$\begin{bmatrix} x_v(t+\Delta T) \\ y_v(t+\Delta T) \\ \phi(t+\Delta T) \end{bmatrix} = \begin{bmatrix} x_v(t) + \Delta T\left(\cdots + b\cos(\phi)\right)) \\ y_v(t) + \Delta T\left(\cdots - b\sin(\phi)\right)) \\ \phi(t) + \Delta T \frac{v_c}{L}\tan(\alpha) \end{bmatrix} \tag{4}$$

where t is the time index and ΔT is the time step between them.

B. Sensor Model

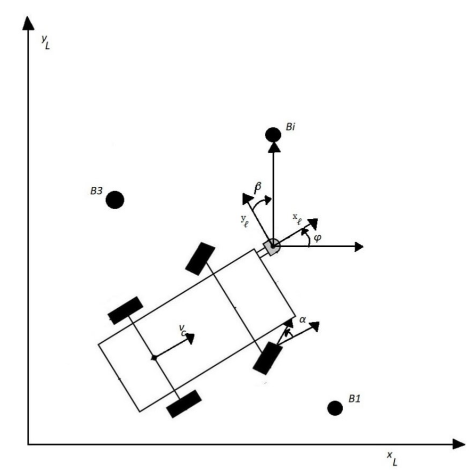

Fig. 4. Vehicle and Laser sensor used in Victoria Park dataset

Furthermore, we have an observational model of the form [3]:

$$\begin{bmatrix} z_r \\ z_\beta \end{bmatrix} = h(x) = \begin{bmatrix} \sqrt{(x_L - x_v)^2 + (y_L - y_v)^2} \\ \arctan\left(\frac{y_L - y_v}{x_L - x_v}\right) - \phi + \frac{\pi}{2} \end{bmatrix} \tag{5}$$

where (z_r, z_β) are the range and bearing, and (x_L, y_L) are the coordinates of a landmark.

Algorithm 1 EKF SLAM with unknown correspondences

Inputs: μ_{t-1}, Σ_{t-1}, *number of landmarks*, u_t, z_t
Initialize events
while filter running **do**
  $e \leftarrow$ next event to process from *Events*
  if e is an odometry event **then**
    Perform EKF propagation with e
  else if e is a GPS measurement **then**
    Perform an EKF update with e
  else if e is a laser scan **then**
    Extract tree range, bearing measurements $\{z\}$ from e
    Perform data association with $\{z\}$
    Perform EKF update with $\{z\}$ using associations
  end if
end while
Define Jacobians, noise covariances, and other required functions for EKF SLAM
Function PERFORM EKF PROPAGATION WITH ODOMETRY EVENT e:
  Compute Jacobians G_t and V_t
  Predict state and covariance: $\bar{\mu}_t = g(\mu_{t-1}, u_t)$, $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + V_t M_t V_t^T$
  Update state and covariance: $\mu_t \leftarrow \bar{\mu}_t$, $\Sigma_t \leftarrow \bar{\Sigma}_t$
Function PERFORM EKF UPDATE WITH GPS MEASUREMENT e:
  Extract GPS measurement z from e
  Compute measurement Jacobian H_t
  Compute innovation covariance $S_t = H_t \Sigma_t H_t^T + Q_t$
  Compute Kalman gain $K_t = \Sigma_t H_t^T (S_t)^{-1}$
  Compute innovation $\delta z = z - h(\mu_t)$
  Update state estimate: $\mu_t \leftarrow \mu_t + K_t \delta z$
  Update covariance estimate: $\Sigma_t \leftarrow (I - K_t H_t)\Sigma_t$
Function PERFORM EKF UPDATE WITH LASER SCAN MEASUREMENTS $\{z\}$:
Perform Data Association (Algorithm 2)
for each observed landmark m_j in $\{z\}$ **do**
  Retrieve the corresponding landmark i from associations
  Compute expected measurement $\bar{z}_t$ for landmark i
  Compute measurement Jacobian H_t for landmark i
  Compute innovation covariance S_t for landmark i
  Compute Kalman gain K_t for landmark i
  Compute innovation δz for landmark i
  Update state estimate: $\mu_t \leftarrow \mu_t + K_t \delta z$
  Update covariance estimate: $\Sigma_t \leftarrow (I - K_t H_t)\Sigma_t$
end for
return μ_t, Σ_t, *number of landmarks*

C. Extended Kalman Filter

The EKF SLAM algorithm is developed for unknown correspondences [5]. Moreover, since each time stamp does not contain data from all the sensors, events are initialized for each data set in order to carry out prediction and update steps smoothly. The process is described in Algorithm 1.

D. Data Association

It is important to identify which observation corresponds to which landmark. There are two possibilities i.e., either an observation can be associated to an already added landmark, or it corresponds to a landmark that currently is not in the map, and needs to be initialized. This is the problem of data association. This is referred to as the *data association* problem [2].

In this project, for Victoria Park dataset, the landmarks are trees. For data association, a cost matrix M is constructed that consists of the Mahalanobis distance; calculated using the residual covariance $\mathbf{S}$ and innovation residual $\mathbf{r}$ [4].

$$d(r; S) = r^T S^{-1} r \tag{6}$$

The data association algorithm looks for the lowest total cost matching in a matrix of costs indicated as M, where m_{ij} reflects the cost associated with assigning entity i to entity j. Measurements that are ambiguous are disregarded, such as those that have costs that are comparable to two known landmarks and those that fall in between matching to a known landmark and initializing a new landmark. All of this is done by using a known χ^2 distribution. The process is described in Algorithm 2.

E. Occupancy Grid Mapping

To validate the results obtained from the EKF-SLAM algorithm, an occupancy grid map was constructed. The environment is discretized into cells that represent the presence of obstacles on this map, which was created using sensor data and EKF estimates. The algorithm's accuracy in mapping complicated surroundings for efficient robot navigation was verified by comparing the map with the ground truth data.

Using Bayesian filtering techniques, each grid cell's occupancy belief is updated in this occupancy grid mapping method in response to sensor measurements. Central to this methodology is the concept of the log-odds ratio (l), which represents the logarithm of the probabilities of a grid cell being occupied relative to being unoccupied.

The log-odds ratio is defined as:

$$l = \log\left(\frac{P(\text{occupied})}{P(\text{not occupied})}\right) \tag{7}$$

where $P(\text{occupied})$ and $P(\text{not occupied})$ are the probabilities of the grid cell being occupied and unoccupied, respectively.

At each time step t, denoted by $l_t(x, y)$, the log-odds ratio for each grid cell (x, y) is updated recursively using Bayes' theorem when a sensor measurement z_t is obtained:

Algorithm 2 Data Association

Initialize empty array *assoc* with size equal to number of measurements
if *number of landmarks* $== 0$ **then**
    return $[-1$ for m in $\{z\}]$
end if
$Q \leftarrow$ covariance matrix of sensor measurements
$M \leftarrow$ initialize cost matrix
for each landmark j **do**
    Compute expected measurement $\bar{z}_t$ and Jacobian H
    for each measurement i **do**
        Compute Mahalanobis distance d between z_i and $\bar{z}_t$
        $M[i,j] \leftarrow d$
    end for
end for
$matches[i,j] \leftarrow$ solve cost matrix M
Initialize thresholds α, β
for each scan i **do**
    if $j \geq$ *number of landmarks* **then**
        if $\min(M[i,j]) > \beta$ **then**
            $associations[i] = -1$
        else
            $associations[i] = -2$
        end if
    else
        $associations[i] = j$
    end if
end for
return $associations$

$$l_t(x,y) = l_{t-1}(x,y) + \log\left(\frac{P(\text{occupied}|z_t)}{P(\text{not occupied}|z_t)}\right) \tag{8}$$

This updating procedure refines the occupancy estimate for every grid cell by integrating sensor results with the prior belief.

V. Results

The trajectory obtained after performing EKF SLAM on the full Victoria Park dataset is shown in Fig. 5. When compared to the actual Victoria Park dataset in Fig. 1, this result seems pretty promising since it matches to the actual trajectory.

The landmarks associated during the EKF SLAM are shown in Fig. 6. It can be seen that the results do not diverge and the landmarks are associated correctly, when compared to Fig. 1.

To further validate the EKF estimates, an occupancy grid map was constructed shown in Fig. 7. It can be observed that the occupancy grid map is mostly grey; this is because there are no such obstacles in the Victoria Park dataset. Since, there are only landmarks and no obstacles, the grid considers it as a grey (unknown) area with a probability of 0.5. Since, the LiDAR is taken as it is in the occupancy grid map construction without omitting the ambiguous measurements, it gives a wider free space at certain intervals.

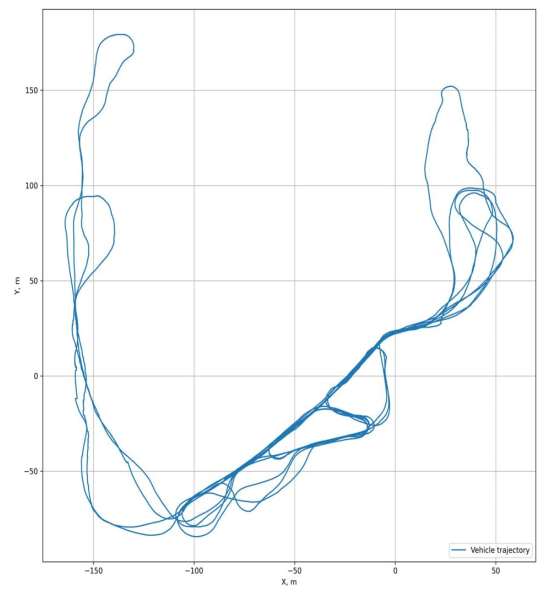

Fig. 5. Trajectory with EKF states

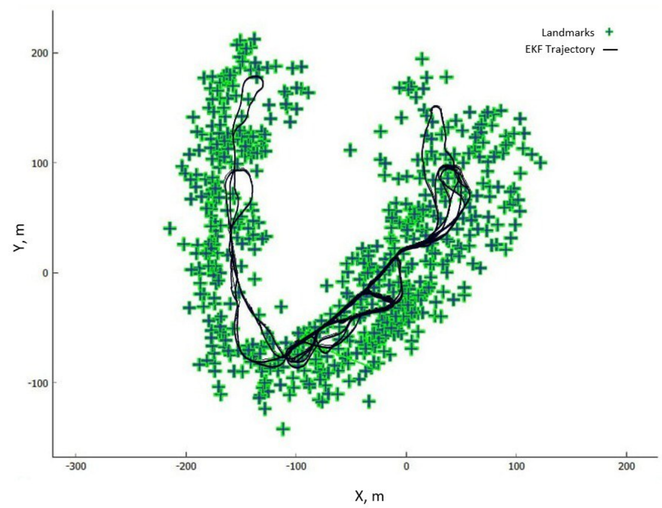

Fig. 6. Trajectory with EKF states and Landmarks

To verify that the EKF SLAM algorithm is working properly, the errors with respect to the ground truth are plotted in Fig. 8 and Fig. 9. It is evident from these plots that the algorithm is working properly since the errors are in the $\pm 3-\sigma$ bounds.

VI. Limitations

Although the suggested method provides promising answers for challenging navigation tasks, a number of drawbacks must be recognized to fully comprehend its applicability.

Sensor data, such as GPS, odometry, and LiDAR, must be accurate and readily available in order for SLAM algorithms to function properly. Moreover, errors, omissions, or noise in these data sources might undermine the effectiveness of

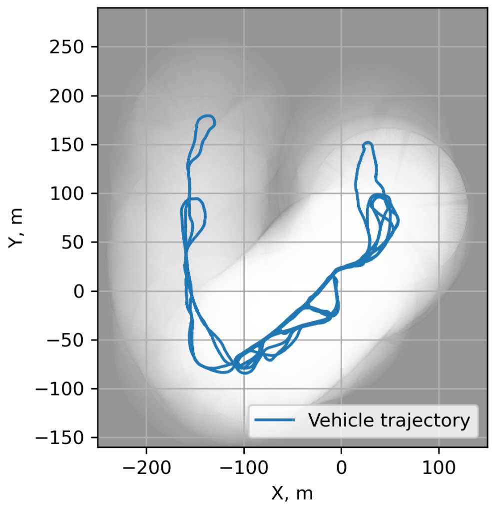

Fig. 7. Occupancy Grid Map with LiDAR data and EKF states

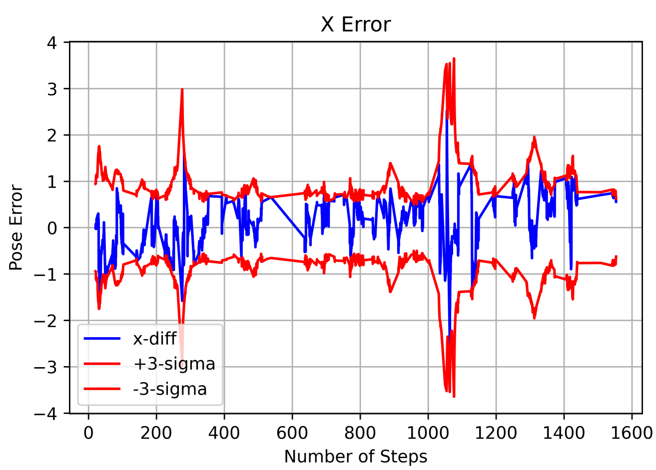

Fig. 8. Error of x-coordinate with $\pm 3 - \sigma$ bounds

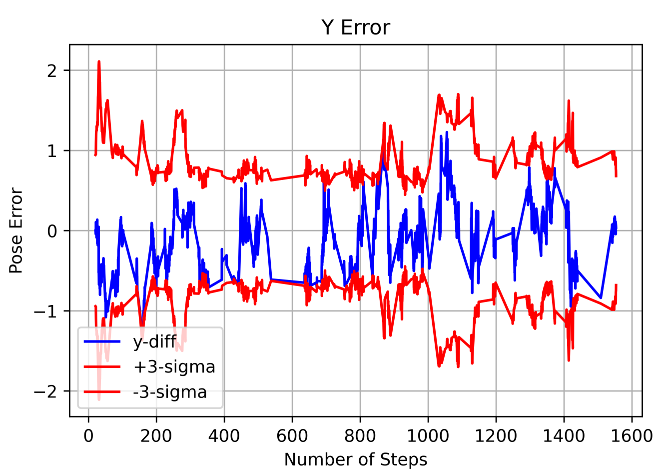

Fig. 9. Error of y-coordinate with $\pm 3 - \sigma$ bounds

the algorithm, affecting the accuracy and dependability of the system's map production and navigation.

Additionally, even though they work well, techniques like occupancy grid mapping and EKF SLAM require a lot of processing power, particularly when used in real-time applications. Implementation is difficult because of this complexity, especially on robotic platforms with limited resources. This would restrict the practicality of these tasks in situations when real-time performance is required.

VII. Conclusion

In conclusion, interesting outcomes have been obtained in the field of simultaneous localization and mapping (SLAM) by the application of Extended Kalman Filter (EKF) SLAM utilizing the Victoria Park dataset. Numerous significant accomplishments have been made possible by this project:

- Through the non-sequential incorporation of GPS, LiDAR, and odometry data, the Victoria Park dataset offered a solid real-world reference for the implementation of EKF SLAM, reflecting the complexity of real-world scenarios.
- When EKF state estimations were compared to ground truth, it was found that the inaccuracies stayed below reasonable ranges, proving that the EKF SLAM method was successful in precisely estimating robot location and mapping the surrounding area.
- Data association was effectively accomplished by utilizing a cost matrix comprising Mahalanobis distances to address the problem of unknown correspondences and ensure robustness in the association process.
- By utilizing EKF outcomes and LiDAR data, an occupancy grid map was created using Bayesian filtering techniques, which successfully localized features like trees and further confirmed the effectiveness of the SLAM framework.

Moving forward, the ranging application of mobile robots demand robust and easy-to-implement tools such as EKF in mapping and localization problems. Therefore, this paper proves the ability of EKF SLAM to adapt the changes to dynamic environment by incorporating real-world data.

References

[1] Civera, J., Grasa, O. G., Davison, A. J. (2010). 1-Point RANSAC for EKF-based structure from motion. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3047-3054).

[2] D. LUNDGREN, "A study of EKF-SLAM for vehicle state estimation.," Dissertation, 2019

[3] "Ute Data Parameters:" https://www-personal.acfr.usyd.edu.au/nebot/experimental_data/modeling_info/Ute_modeling_info.htm

[4] Adarsh Modh, *EKF SLAM Implementation*, GitHub repository, https://github.com/adarshmodh/EKF_SLAM, Last updated: March 6, 2020, Accessed: March 8, 2024.

[5] Thrun, S., Burgard, W., Fox, D. (2005). *Probabilistic Robotics*. MIT Press.

[6] Nebot, E. *Victoria Park Dataset*, https://www-personal.acfr.usyd.edu.au/nebot/victoria_park.htm