Journal of Robotics and Control (JRC)
Volume 5, Issue 6, 2024
ISSN: 2715-5072, DOI: 10.18196/jrc.v5i6.23519 

# Enhancing Autonomous Navigation in GNSS-Denied Environment: Obstacle Avoidance Observability-Based Path Planning for ASLAM

Samaneh Elahian [1], Mohammad Ali Amiri Atashgah [2*], Iswanto Suwarno [3]
[1, 2] Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran
[3] Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Yogyakarta, Indonesia
Email: [1] s.elahian84@gmail.com, [2] atashgah@ut.ac.ir, [3] iswanto_te@umy.ac.id
*Corresponding Author

***Abstract*—Obstacle avoidance (OA) is necessary for any path planning in outdoor environment to prevent any collision with the obstacles in natural environment. In this paper, a quadrotor navigates using Active Simultaneously Localization and Mapping (ASLAM) in GNSS-denied outdoor environment. In ASLAM, the quadrotor path is defined using real-time Observability Based Path Planning (OBPP) method, autonomously. To prepare using of the OBPP in outdoor environment, it is necessary to add the ability of OA to it. So, the OA-OBPP method is introduced which defines the path based on terrestrial landmarks while preventing any collision with the obstacles. This approach is developed by redefining a dataset of in range landmarks while all of the landmark in the vicinity of the obstacles are removed from the in-range landmarks dataset. To evaluate the performance of the proposed method, simulations of the OA-OBPP algorithm are conducted for a simplified 6-Degree of Freedom (DOF) quadrotor using MATLAB. The simulations evaluate the efficiency, accuracy and robustness of the proposed method. Results across various scenarios show that the method effectively avoids collisions with obstacles while simultaneously determining a path to the goal. Additionally, a comparison of the position estimation RMSE with Monte Carlo PP highlights the accuracy of the OA-OBPP. The robustness of the method, tested with varying initial positions, demonstrates its success in real-time path planning (PP) from any starting point to the destination without collisions. The results confirm that the OA-OBPP enhances the robot's capability to perform real-time, autonomous, and robust path planning in outdoor environments, even in the absence of GNSS signals, through visual navigation.**

***Keywords—Obstacle Avoidance; Observability Based Path Planning; Real-Time Autonomous Path Planning; Active Simultaneously Localization and Mapping; GNSS-Denied Environment; Observability Degree.***

## I. INTRODUCTION

The main aim of the paper is to propose a real-time autonomous path planning method for quadrotors operating in unknown, GNSS-denied outdoor environments, with a focus on accurate positioning. This topic addresses several key areas including SLAM for robot and landmark positioning within the environment, OBPP for real-time autonomous motion of the quadrotor, and OA to adapt the OBPP method for practical applications in environments cluttered with obstacles. The following sections discuss these aspects in detail.

Autonomous robot navigation of mobile robots is a significant topic in robotic investigation. They have received special attention in recent research due to constraints in communication link budgets, the limited range of remote control, the complexity of manual operation in challenging environments, and the increasing use of robots in swarm formations [1]–[5]. One of the most challenging issues in autonomous robotics is navigating in outdoor environments where GNSS (Global Navigation Satellite System) signals are unavailable or unreliable [6]–[8]. In this environment, the robot relies solely on its IMU sensors for localization and navigation. Over long distances in outdoor settings, integrating acceleration data from the IMU to compute velocity and position can introduce significant errors that accumulate over time, especially in large environments. Effective navigation in such scenarios necessitates robust localization techniques, real-time autonomous path planning [9], [10], and obstacle avoidance (OA) to prevent collisions with external obstacles [6], [9], [11]. The literature explores each of these challenges from various perspectives. To enhance understanding, we will examine these critical aspects in greater detail.

To prepare a robot to move in such an environment, it must be able to localize itself. SLAM is introduced as a solution for robot localization in an unknown GNSS-denied environment, as it enables the robot to build a map of its surroundings and determine its position using visual sensors to detect terrestrial landmarks [12], [13]. Various issues are investigated in SLAM including sensor types [13]–[15], estimation filters [16]–[20] and path planning [21]–[23]. The trajectory taken by the robot to reach its destination significantly impacts the SLAM accuracy different paths may expose the robot to varying sets of landmarks [24]–[27]. Numerous methods have been proposed for the robots motion planning [28]–[32]. Many of these methods such as A*, D*, RRT, potential field and optimization rely on prior knowledge of the environment. In these approaches, the path is predefined, and the robot navigates through the established map from the starting point to the destination. These traditional methods are often unsuitable for real-time path planning in outdoor environments. Modern approaches, such as neural networks and artificial intelligence, require extensive training to handle the challenges of dynamic and complex environments [33].

The proposed approach for real-time path planning in unknown environments within SLAM applications involves utilizing terrestrial features. These features are essential for accurate SLAM estimation and are also utilized in real-time Adaptive SLAM (ASLAM) for path planning. Key methods in real-time ASLAM include the expected information policy [21], cognitive based adaptive optimization [34], environment information gathering [35] and observability based path planning [36]. As in OBPP, the position estimation is more accurate due to higher observability in the path, so OBPP has been chosen for guiding the robot to its destination.

ASLAM methods are applicable to various types of robots including aerial, space, terrestrial, and marine systems [37]–[41]. For this research, quadrotors are chosen due to their distinct advantages, such as low development and operation cost, ease of use, lightweight and compact design, excellent mobility, and high maneuverability [42]. These advantages enable the quadrotors to be effectively used in a variety of applications including high quality of remote sensing [43], [44], precision agriculture [45]–[47], surveillance [48]–[50] and other fields [51]–[54] contributing to the rapid development of these vehicles. Consequently, this research focuses on applying ASLAM to quadrotors.

The final challenge in autonomous navigation in outdoor environments is ensuring the robot can move safely without colliding with obstacles. Different types of robots encounter specific obstacles based on their intended use, whether in indoor or outdoor settings, as well as their movement capabilities. When a quadrotor operates in an outdoor environment, it encounters obstacles such as tall trees, towers, buildings, other flying devices, or even birds. Therefore, path planning methods must incorporate effective obstacle avoidance capabilities. Several path planning methods are specifically designed with collision avoidance in mind including Cell decomposition and potential field (PF) [23], [55]–[57] and optimization-based approaches. are all categorized within this domain. However, it is crucial to develop collision avoidance capabilities for all path planning methods to ensure safe flight, particularly in outdoor environments. Each path planning method integrates obstacle avoidance according to its own principles. For example, collision avoidance strategies have been applied to methods such as D* [58], RRT [59], [60], optimal path based on Genetic Algorithm (GA) [61], and international regulations for preventing collisions at sea (COLREGs) [62], [63]. Additionally, combination of these methods with other traditional PP techniques have been explored in various studies [64]–[66].

Obstacle avoidance (OA) is a critical consideration in the implementation of ASLAM in various research studies [22], [34], [67]–[71]. ASLAM is employed when a robot autonomously plans its path from the start to the end point in an unknown environment while simultaneously performing SLAM [24], [36], [72]. To enhance robot safety and protect surrounding objects, incorporating effective obstacle avoidance is essential for practical applications of autonomous robot motion in ASLAM. Numerous papers have explored different approaches to obstacle avoidance in this context.

The key aspect of all these methods is that obstacle avoidance (OA) is developed in relation to the underlying principles of the path planning (PP) method. In other words, to effectively prevent collisions with obstacles, the OA approach must align with the principles on which the path planning method is based. For example, in potential field path planning, obstacles are modeled as repulsive forces that affect the robot's path determination by creating repulsion fields around obstacles [73]. In methods like A*, D*, and RRT, path planning involves evaluating a list of open nodes surrounding the robot at each step [74]. Therefore, to define an appropriate OA method for these approaches, obstacles should be represented as nearby nodes to prevent the robot from moving in their direction. In fuzzy logic PP, OA is implemented through the definition of appropriate rules [75], [76]. In optimization-based path planning methods, such as Particle Swarm Optimization (PSO), the path is determined based on a defined cost function [77]. To incorporate obstacle avoidance, penalty terms are added to the cost function to guide the robot's movement away from obstacles and toward the optimal path.

The reviewed literature primarily focuses on pre-planned paths or autonomous path planning in known environments. In response, this paper proposes an enhanced obstacle avoidance method to improve the Observability-Based Path Planning (OBPP) framework for ASLAM in unknown GNSS-denied environments specifically for a quadrotor. OBPP is designed for real-time path planning in such challenging conditions and is based on the concept of observability degree [36]. In an observable system, the input state vector can be accurately estimated by measuring the output [38], [78]–[80]. By increasing the degree of observability, the accuracy of state vector estimation is improved, which in turn enhances the accuracy of the environment map in ASLAM.

The research contributions are:

- **Implementation of EKF-SLAM for a quadrotor in 3D environment**: Extended Kalman Filter (EKF) SLAM technique is employed for autonomous path planning of a quadrotor in unknown outdoor environments. This implementation incorporates simplified six-degree-of-freedom (6-DOF) equations and employs Proportional-Integral-Derivative (PID) controllers for both the internal and external control loops of the quadrotor.
- **Real-time OBPP with Obstacle Avoidance**: Development of a real-time Observability-Based Path Planning (OBPP) method that incorporates obstacle avoidance capabilities for autonomous navigation of quadrotors in unknown outdoor environments.
- **Evaluation of the Proposed Method by Monte Carlo PP**: The assessment of the proposed path planning method confirms its effectiveness in navigating the quadrotor from the starting point to the destination without collisions, and it shows reduced error compared to conventional Monte Carlo (MC) path planning techniques.

In section II, a brief review of ASLAM is provided, highlighting its importance for positioning both the quadrotor

and landmarks in unknown environments. This section also includes an overview of OBPP and its fundamental principles, given that the development of the OA method is closely linked to the underlying path planning approach. Section III details MATLAB simulations used to evaluate the performance of the Collision-Aware OBPP (CA-OBPP) method across three key scenarios: trajectory analysis, accuracy metrics, and the effect of initial positioning. The paper concludes with a discussion of the limitations of the proposed method, suggestions for future work, and a summary of the findings.

## II. METHODS

The motion of a quadrotor in an environment is governed by three main components: guidance, control, and navigation as depicted in Fig. 1. The guidance block calculates the desired destination values for the quadrotor motion. The control block then determines the commands needed to adjust the quadrotor's current position and attitude to match these destination values. In this research, PID controllers are employed for both internal control (which manages the roll, pitch, and yaw angles) and external control (which governs position vectors in the x, y, and z directions). The navigation block measures the quadrotor's current attitude and calculates its position. The EKF-SLAM framework is then employed to estimate the position of both the quadrotor and terrestrial landmarks.

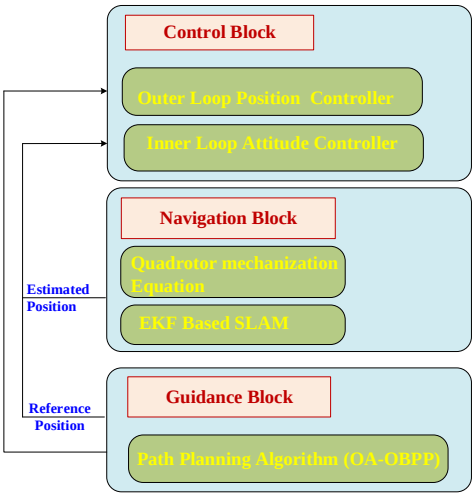


Fig. 1. Quadrotor motion block-diagram with SLAM

This paper focuses on the implementation of the guidance block for the robot path planning incorporating obstacle avoidance capabilities. This block calculates the desired position of the quadrotor using the path planning algorithm while ensuring that obstacles scattered throughout the environment are avoided. We propose an autonomous real-time path planning method that functions independently of GNSS signals and does not require prior knowledge of the environment or the path. This method is based on the concept of observability degree and achieves high accuracy in SLAM estimation, as described below.

### A. ASLAM and OBPP

Autonomous motion of the robot in outdoor environments has been extensively researched [32], [81], [82]. The promising performance of SLAM [83]–[85] in real-world applications has encouraged scientists to utilize it as a basis for the further development. This has led to the creation of fully autonomous systems capable of navigating in unknown GNSS-denied environments without external interference [69], [86], [87]. By incorporating concurrent autonomous path planning with SLAM, known as ASLAM, these systems achieve a high level of autonomy in such challenging environments.

SLAM is a computational problem with various solutions proposed, including the Extended Kalman Filter (EKF) and Particle Filter (PF). Given the promising results of EKF-SLAM, its relatively simple implementation, and its low computational demands for real-time path planning, EKF-SLAM is selected for this research. EKF-SLAM developed based on the state and covariance matrix with two propagation and update steps. The state vector for a quadrotor operating in 3D space is defined as follows (1).

$$X = [x_r\ y_r\ z_r\ u\ v\ w\ \varphi\ \theta\ \psi\ x_{l1}\ y_{l1}\ z_{l1}\ \dots\ x_{lm}\ y_{lm}\ z_{lm}]^T \quad (1)$$

The state matrix in EKF-SLAM includes data of both the quadrotor and the landmarks. For the quadrotor, this data encompasses the position; $(x_r, y_r, z_r)$, linear velocities; $(u, v, w)$ aligned with the coordinate axes, and Euler angles; $(\varphi, \theta, \psi)$ defined in navigation frame. The position of the $i^{th}$ landmarks is represented by $(x_{li}, y_{li}, z_{li})$ in the navigation frame. The prediction of the state (2) and covariance (3) matrices in EKF-SLAM are computed using the process and measurement model ((4), (5)), respectively.

$$\hat{X}_{n+1}^{-} = F\hat{X}_n^{+} + w_{n+1} \quad (2)$$

$$P_{n+1}^{-} = F_{n+1}P_n^{+}{F_{n+1}}^T + Q \quad (3)$$

$$f = \begin{bmatrix}\Delta P^n(k+1)\\ \Delta V^n(k+1)\\ \Delta\psi^n(k+1)\end{bmatrix} = \begin{bmatrix} v^n(k)\Delta t\\ [C_b^n(k)a^b(k) + g^n]\Delta t\\ E_b^n(k)w^b(k)\Delta t\end{bmatrix} \quad (4)$$

$$h = \begin{bmatrix}\rho_{RL}\\ az\\ el\end{bmatrix} = \begin{bmatrix}\sqrt{({\delta_x}^2 + {\delta_y}^2 + {\delta_z}^2)}\\ tan^{-1}(\frac{\delta_y}{\delta_x})\\ tan^{-1}(\frac{\delta_z}{d_{RL}})\end{bmatrix} \quad (5)$$

In this context, w and Q represent the process Gaussian noise and its covariance, respectively. The superscripts n and b denote parameters in the navigation frame and body frame, respectively. Specifically, $a^b$ represent linear acceleration measured in the body frame, $g^n$ denotes gravitational vector in the North-East-Down (NED) navigation frame, and $w^b$ refers to the rotation rates in the body frame. The direction cosine matrix $C_b^n$ and rotation rate transfer matrix $E_b^n$ which transform data from the body frame to the navigation frame, are given by Equations (6) and (7);

$$C_b^n = \begin{bmatrix} C_\psi C_\theta & -S_\psi C_\varphi + C_\psi S_\theta S_\varphi & S_\psi S_\varphi + C_\psi S_\theta C_\varphi\\ S_\psi C_\theta & C_\psi C_\varphi + S_\psi S_\theta S_\varphi & -C_\psi S_\varphi + S_\psi S_\theta C_\varphi\\ -S_\theta & S_\varphi C_\theta & C_\varphi C_\theta\end{bmatrix} \quad (6)$$

$$E_b^n = \begin{bmatrix}1 & S_\varphi t_\theta & C_\varphi t_\theta\\ 0 & C_\varphi & -s_\varphi\\ 0 & S_\varphi sec_\theta & c_\varphi sec_\theta\end{bmatrix} \quad (7)$$

More extensively detailed computations of EKF-SLAM are presented in [88], [89]. For a quadrotor to travel from its initial to final point, a path must be defined at each step of its movement. Therefore, an autonomous real time path planning method for enhancing the positioning accuracy is chosen. OBPP is developed based on the concept of observability degree (OD) [90]–[93], which autonomously determines a path using terrestrial landmarks. The observability of a system reflects its ability to accurately estimate its inputs based on measured outputs [80], [92], [94]. Methods for measuring OD vary: some methods, like the system observability matrix, simply assess whether the system is observable, while others quantify the degree of observability. Utilizing the OD for path determination allows for the comparison of system accuracy in state estimation by measuring different system outputs. This capability is applied in various contexts, such as sensor positioning [78], [95], [96] and navigation [93], [97], [98]. In OBPP, the goal is to determine the next point of motion in a way that maximizes the observability degree of the system at each step.

Among various matrices used for observability measurement—such as the covariance matrix, observability matrix, and Gramian matrix—the eigenvalues and eigenvectors are selected for assessing OD in path planning methods [91], [99]. Eigenvalues and eigenvectors are advantageous because they not only quantify the degree of observability but also indicate the direction in which the system achieves the highest observability. For nonlinear systems, the Gramian matrix is computed by defining the transition and output matrices and then linearizing the nonlinear system through Taylor expansion. The Gramian is calculated as follows (8),

$$W_o \approx O_d^T O_d \tag{8}$$

That $O_d$ is a discrete-time observability matrix (9).

$$O_d = \begin{bmatrix} C_d \\ C_d A_d \\ C_d A_d^2 \\ \vdots \\ C_d A_d^{n-1} \end{bmatrix} \tag{9}$$

The next point for the quadrotor's motion is determined by the eigenvectors corresponding to the maximum eigenvalue of the Gram matrix as given by Equations (10) and (11).

$$\lambda_{timax} = max\ (\lambda_{ti1}. \ldots . \lambda_{tin}) \tag{10}$$

$$v_{timax} = v_{ti}(\lambda_{ti-max}) \tag{11}$$

Further details on the principles of OBPP are discussed in [43] for a terrestrial robot in a 2D environment. In the context of ASLAM, the quadrotor navigates towards the most observable direction using an in-range landmarks dataset, aiming to maximize observability and thereby enhance accuracy in state estimation. The next step is to integrate OA capabilities with OBPP.

### *B. Obstacle Avoidance in OBPP*

In OBPP, the path is defined using terrestrial landmarks within the robot's FOV. This principle also forms the basis for implementing OA. Specifically, to incorporate OA into OBPP, the dataset of in-range landmarks used for path planning is adjusted by excluding landmarks that are close to detected obstacles. The proposed method focuses on the distance between in-range landmarks and obstacles. When obstacles are detected within the quadrotor's FOV, their positions are determined based on the quadrotor's absolute estimated position and the relative distance of the obstacles within the coordinate framework. Landmarks within a specified radius of each detected obstacle are removed from the quadrotor's in-range dataset. In OBPP, the quadrotor's movement direction is determined based on the observability of the remaining landmarks in the dataset. If an area lacks landmarks, the quadrotor avoids moving in that direction. Consequently, by defining detected obstacles as landmark-free regions within a specified radius, the robot effectively avoids paths that would lead to collisions. The OA-OBPP flowchart illustrating this process is shown in Fig. 2.

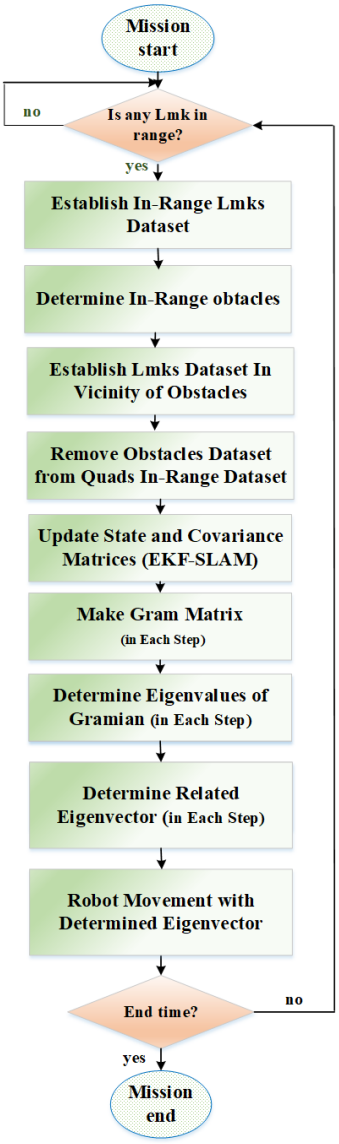


Fig. 2. OA-OBPP process

At each step of the robot's motion, the detected obstacles are treated as a center of a forbidden area. Landmarks within a specified radius around each obstacle are added to the list

of removed landmarks, as described in (12). This forbidden area is modeled as a circle with the obstacle at its center and a radius of 10 meters ($R_{col_{max}} = 10\ m$).

$$R_{col} = \sqrt{\left(x_{obst\ i} - x_{lmk\ j}\right)^2 + \left(y_{obs\ i} - y_{lmk\ j}\right)^2} \quad (12)$$

Landmarks within a radius of less than 10 meters from an obstacle are considered part of the forbidden movement area and are removed from the in-range landmarks dataset as defined by (13) to (14). Additionally, the maximum range for landmarks within the quadrotor's FOV is set to 50 meters ($R_{quad.max} = 50\ m$).

$$R_{quad} = \sqrt{\left(x_{quad} - x_{lmk\ j}\right)^2 + \left(y_{quad} - y_{lmk\ j}\right)^2} \quad (13)$$

$$lmk_{quad} = \begin{cases} \{lmk_i\} \ \ if \ \ R_{lmk_i} < R_{quad.max} \quad for\ i = 1{:}n \quad no\ OA \\ \{lmk_i\} \ \ if \begin{cases} R_{lmk_i} < R_{quad.max} \quad for\ i = 1{:}n \\ \qquad\qquad and \qquad\qquad with\ OA \\ R_{col} < R_{col.max} \quad fori = 1{:}n \end{cases} \end{cases} \quad (14)$$

To clarify the concept of landmark redefinition, an example of the datasets is provided by the the quadrotor's in-range landmarks ($lmk_{quad}$), landmarks near obstacles ($lmk_{obs}$), and the final landmarks dataset for path planning decision-making ($lmk_{dataset}$) as illustrated in (15) to (17).

$$lmk_{quad} = \{L_1.L_7.L_{81}.L_9.L_{11}.L_{46}.L_{53}.L_{21}.L_{34}.L_{74}\} \quad (15)$$

$$lmk_{obs} = \{L_7.L_{88}.L_{46}.L_{53}.L_{13}\} \quad (16)$$

$$lmk_{dataset} = \{L_1.L_{81}.L_9.L_{11}.L_{21}.L_{34}.L_{74}\} \quad (17)$$

The list of landmarks used for PP decisions is then finalized. The algorithm for the OA-OBPP is detailed in Table I and the concept is illustrated in Fig. 3.

TABLE I. OA-OBPP ASLAM ALGORITHM WITH

| | Algorithm 1: Algorithm of the ASLAM with OA-OBPP |
|---|---|
| 1: | **Mission Starts** |
| 2: | **EKF-ASLAM Starts** |
| 3: | **Path-Planning Starts** |
| 4: | **Search** any obstacles in the quadrotor's FOV |
| 5: | Find the landmarks in the defined bound of the obstacle |
| 6: | omit in bound landmarks from the dataset |
| 7: | **Make** Observability dataset |
| 8: | **For** i=1:numLmk **do** |
| 9: | Calculate the observability matrix (O) according to (7). |
| 10: | Calculate observability Gramian ($W_o$) by (6) |
| 11: | Determine max eigenvalue of $W_0$ |
| 12: | **End** |
| 13: | **Search** the most observable direction |
| 14: | Find max eigenvalue of all eigenvalues in each step |
| 15: | Select the eigenvector of the max eigenvalue |
| 16: | $\theta_{rot} = \theta_{MOD} - \theta_{rob}$ (MOD: most observable direction) |
| 17: | **Perform Remaining EKF-ASLAM Tasks** |
| 18: | **Path-Planning Ends** |
| 19: | **EKF-ASLAM Ends** |
| 20: | **Mission Decision Making Tasks** |
| 21: | **Mission Ends** |

As seen in Fig. 3, the landmarks in the defined radius from the center of the obstacle, are depicted as faded landmarks. Means that they can't put into the landmarks dataset for PP decision.

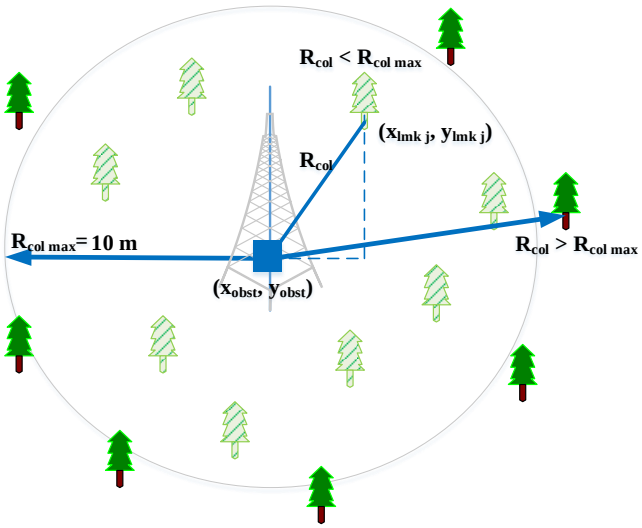


Fig. 3. Forbidden area around a detected obstacle

By integrating OA capability into the OBPP method, it evolves into a comprehensive real-time autonomous path planning (PP) method suitable for diverse outdoor environments for quadrotor motion This enhanced method is capable for various applications such as monitoring vegetation, wildlife, or coastal areas. By distributing the artificial features within the monitoring area, the quadrotor can autonomously assess plant and animal coverage or shoreline conditions. The method is also applicable for planning in unfamiliar plant environments, where specific trees can be used as landmarks to create detailed maps of these areas. Furthermore, it has potential applications in emergency situations; by identifying and mapping key features in such environments, it can aid in locating and providing better identification for injured individuals.

In practical applications, the complexity of feature extraction in real-world scenarios and the requirement for a sufficient number of landmarks within the quadrotor's field of view at all times are significant challenges that limit the application of OA-OBPP. However, there is no limitation on the number of obstacles that can be included.

## III. SIMULATION SCENARIOS AND RESULTS

In this section, we present the simulation results for OA-OBPP with EKF-SLAM across various scenarios. In the simulated environment, the position of the landmarks and obstacles, represented by '+' and '●' symbols respectively, are randomly selected on a 2D plane on the ground with an altitude of zero.

Three main scenarios are defined, each with different obstacle positions. These positions are chosen to ensure that there are always at least ten landmarks within the robot's in-range dataset. Subsequently, the robot's initial position is varied across five different locations on the landmarks plane at ground level. The obstacles considered in these scenarios include communication towers and tall buildings, which are relevant for communication operations.

The quadrotor utilizes internal and external sensors to determine its position and measure the distance and bearing

to landmarks and obstacles. The Specifications of these sensors are detailed in Table II. Each terrestrial landmark is represented as a tree with unique features, which are identified using feature extraction algorithms. The performance of the proposed method is assessed by evaluating the quadrotor's trajectory from the initial point to the endpoint, ensuring that there are no collisions with obstacles along the path and maintaining high accuracy, as illustrated in Fig. 4. In this figure, the quadrotor's path, as determined by the OBPP method, is shown as a dotted line. When obstacles are added and faded landmarks are removed from the main landmark dataset, the updated path, depicted by a solid line, is determined using the OA-OBPP method.

TABLE II. QUADROTOR INTERNAL AND EXTERNAL SENSORS SPECIFICATION

| **IMU** | | **Vision camera specification** | |
|---|---|---|---|
| **Sampling rate** | 200 Hz | Sampling rate | 2 Hz |
| **Acc noise** | 0.12 $m/s^2$ | Max. view range | 55 m |
| **Gyro Noise** | 0.28 deg/s | FOV | $60^{ver}\times180^{hor}$deg |

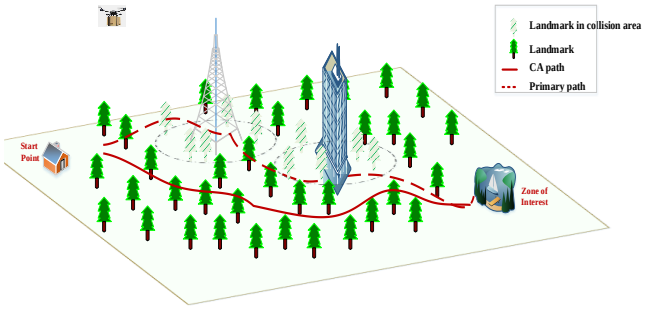


Fig. 4. Quadrotor CA-OBPP test scenarios in the designed outdoor environment

The robot's path is visualized using two trajectories: True Path (TP) and Estimated Path (ESTP) for scenarios with and without obstacles. TP represents the trajectory the quadrotor follows using simplified 6-DOF (Degree of Freedom) equations without any noise in the measured data. This path is derived from measurements of acceleration and angular rates obtained using internal accelerometers and gyroscope sensors

The ESTP is determined by the EKF-SLAM algorithm, which integrates attitude measurements from internal sensors with observations from external sensors, alongside Dead Reckoning (DR) equations. External sensors detect a set of in-range landmarks and obstacles. Using this data, the quadrotor constructs a safe dataset, which includes the in-range landmarks, excluding those near the detected obstacles.

In both TP and ESTP, the next position of the robot's trajectory is calculated utilizing the attitude and estimated current position, along with the OA-OBPP algorithm applied to the established safe dataset.

MATLAB simulations are conducted to analyze the performance of the OA-OBPP method across the designed scenarios. To provide a comprehensive evaluation of the proposed method, simulations are divided into three subsections:

- ***Trajectory Analysis:*** The first subsection investigates the ability of the proposed method to avoid obstacles while guiding the robot from initial to the end point with OA-OBPP method.
- ***Localization Accuracy****:* The second subsection compares the robot localization error between the proposed and Monte Carlo Path Planning (MCPP) method. A key difference is that the MCPP method requires prior knowledge of the environment to determine the path, whereas the OA-OBPP method generates the path autonomously. The accuracy of both methods is evaluated by comparing the RMSE of the robot's position.
- ***Effect of Initial Position:*** Accurate initialization is critical for predetermined path planning (PP) methods in quadrotors, as errors in the initial position can lead to significant deviations in the robot's path. The final subsection examines how variations in robot initialization affect the trajectory from the starting point to the endpoint, evaluating the robustness of the path planning method.

### *A. Trajectory Analysis*

The results are demonstrated for different arrangements of the obstacles in the environment. The path that the quadrotor follows from the start to the end point, in an obstacle-free (OF) and including obstacles (showcase by OA) environment using OBPP for different arrangements of the obstacles in the environment, in three scenarios, is depicted in Fig. 5 to Fig. 7. In these figures, the true path and estimated path by EKF-SLAM are displayed. The error of position in the x and y directions, between the true and estimated values, are depicted in Fig. 8 and Fig. 9, respectively. The Root Mean Square Error (RMSE) of the path for different scenarios is compared in Table III.

TABLE III. POSITION AND ATTITUDE RMSE FOR VARIOUS ARRANGEMENT OF THE OBSTACLES

| Scenario | Pos. RMSE total | Att. RMSE total | x RMSE | y RMSE | z RMSE |
|---|---|---|---|---|---|
| SCN 1 | 2.0078 | 0.1360 | 1.5648 | 1.2579 | 0.0159 |
| SCN 2 | 2.2717 | 0.1339 | 1.7270 | 1.4757 | 0.0162 |
| SCN 3 | 1.4996 | 0.1282 | 1.4687 | 0.3025 | 0.0172 |

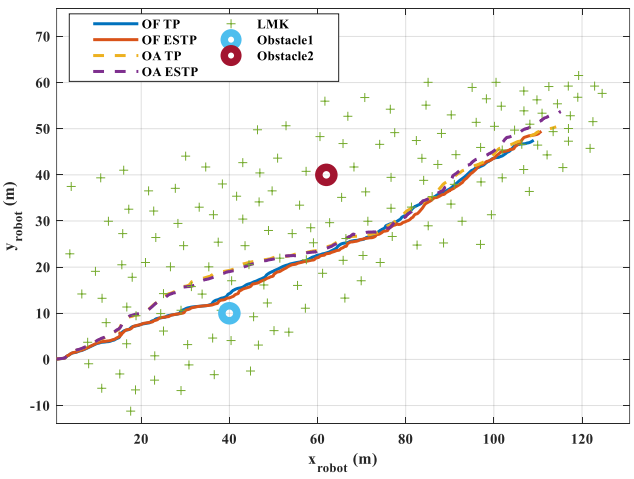


Fig. 5. Quadrotor motion with OA-OBPP for scenario 1

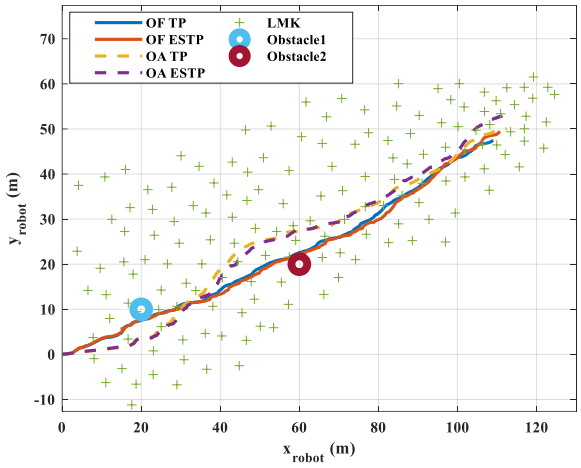


Fig. 6. Quadrotor motion with OA-OBPP for scenario 2

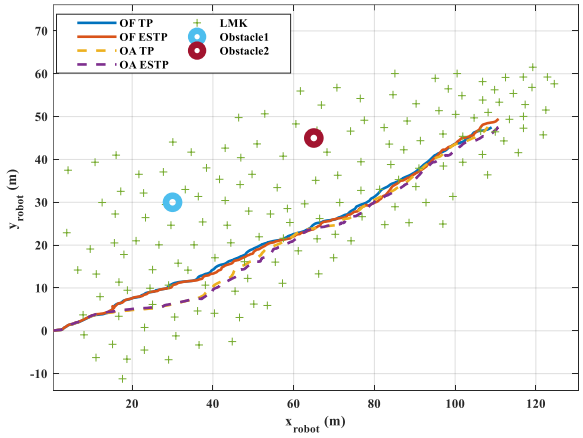


Fig. 7. Quadrotor motion with OA-OBPP for scenario 3

The results demonstrate that with the CA-OBPP method, the quadrotor can autonomously determine its path from the initial to the end point without requiring any prior information about the environment or access to GNSS signals, even in the presence of the obstacles and without any collision with them. So the first goal of the OA-OBPP is achieved. In each of the scenarios the quadrotor defines the path considering the landmarks in the dataset to gain the most observability of the system by passing through the defined trajectory.

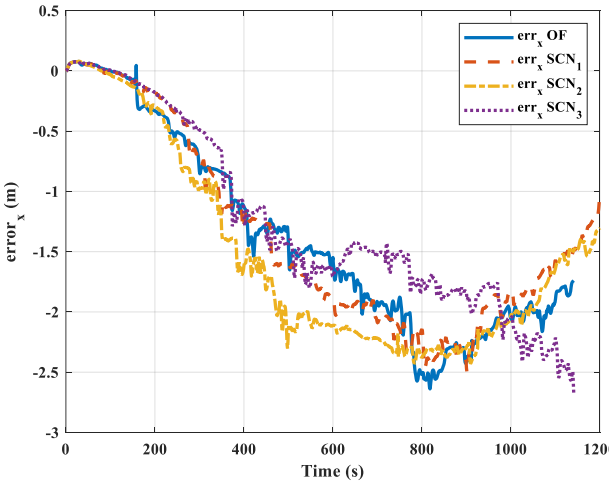


Fig. 8. Quadrotor x-position error with OA-OBPP for scenarios 1-3

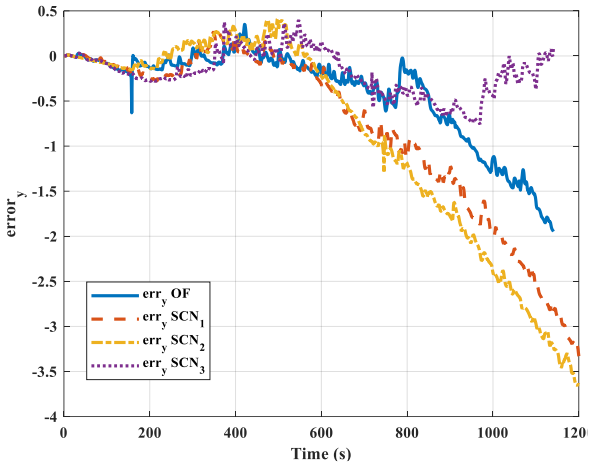


Fig. 9. Quadrotor y-position error with OA-OBPP for scenarios 1-3

The comparison of RMSE for scenarios 1 and 2 indicates that they are nearly identical. This similarity is due to the fact that, in both scenarios, the path is navigated through regions with obstacles. When the quadrotor moves between obstacles, the landmark dataset remains relatively consistent because obstacles reduce the number of landmarks and remove those in their vicinity. In contrast, scenario 3 landmarks a more densely arranged obstacle layout, resulting in a more extensive landmark dataset compared to scenarios 1 and 2. As a result, the RMSE in scenario 3 is lower than in the other scenarios.

### *B. Localization Accuracy*

To provide a comprehensive evaluation of the proposed method, this section compares its accuracy with that of the MCPP method [100] using the same scenario. Given that the OA-OBPP method prioritizes the most observable directions, it is expected to achieve higher estimation accuracy. Consequently, we compare the RMSE of both position and attitude estimation for scenario 2 (identified as having the highest RMSE in subsection A) with the MCPP method. This comparison is illustrated in Fig. 10.

The results demonstrate that with OA-OBPP lower value of RMSE is obtained in comparison with MCPP method and the proposed method gives better accuracy as shown in Table IV. Accordingly, the second goal of the proposed method is passed successfully.

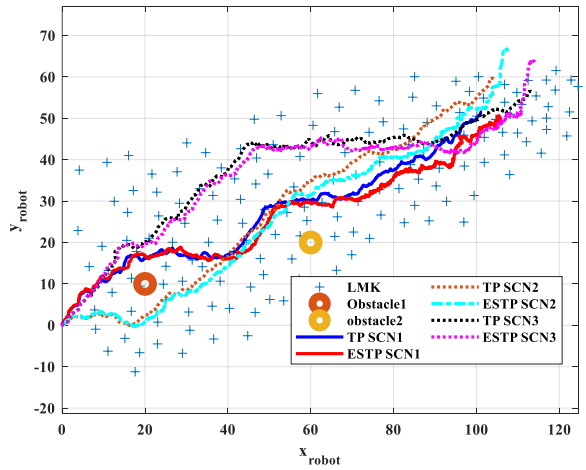


Fig. 10. Quadrotor motion with MCPP for scenario 2

TABLE IV. POSITION AND ATTITUDE RMSE FOR OA-OBPP AND MCPP IN SCENARIO 2

| Scenario | Pos. RMSE total | Att. RMSE total |
|---|---|---|
| SCN 2 | 2.2717 | 0.1339 |
| MC_SCN 1 | 3.4355 | 0.2923 |
| MC_SCN 2 | 2.8683 | 0.2836 |
| MC_SCN 3 | 3.4323 | 0.3052 |
| MC_avg | 3.2453 | 0.2937 |

### C. *Effect of Initial Position*

In pre-planned PP methods if a quadrotor starts from a different initial point, the quadrotor reaches to a different destination. To evaluate the robustness of the proposed method with respect to changes in the initial position, simulations are conducted from various starting points. This provides a comprehensive assessment of the trajectories generated by the proposed method, as shown in Fig. 11, along with the associated errors depicted in Fig. 12 and Fig. 13. In this simulation, the positions of the obstacles are kept constant across all scenarios. However, changes in initial positions impact the landmark dataset within the quadrotor's field of view, particularly at the start of its mission. As a result, the quadrotor follows different paths, leading to varying RMSE values, as illustrated in Table V.

TABLE V. POSITION AND ATTITUDE RMSE FOR VARIOUS INITIAL POSITIONS IN SCENARIO 1

| **initial point (x, y, z)** | **Pos. RMSE total** | **Att. RMSE total** | **RMSE x** | **RMSE y** | **RMSE z** |
|---|---|---|---|---|---|
| (0, 0, 0) | 2.0078 | 0.1360 | 1.5648 | 1.2579 | 0.0159 |
| (-10, 10, 0) | 1.6963 | 0.1209 | 1.4340 | 0.9061 | 0.0149 |
| (-10, 0, 0) | 1.7293 | 0.1206 | 1.5909 | 0.6778 | 0.0138 |
| (-5, 15, 0) | 1.9469 | 0.1164 | 1.9050 | 0.4015 | 0.0142 |
| (5, 30, 0) | 1.5785 | 0.1390 | 1.4590 | 0.5945 | 0.0980 |

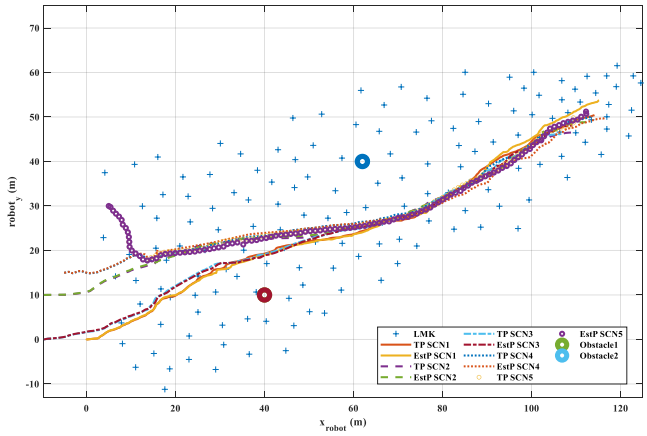


Fig. 11. Quadrotor motion path for various initial positions

The results indicate that variations in initial position do not affect the robot's path to the defined destination. With the proposed method, the quadrotor can determine its trajectory to the endpoint from different starting positions while maintaining optimal observability. Scenarios 2, 4, and 5, which feature initial positions farther from the destination compared to scenarios 1 and 3, benefit from a higher density of landmarks at the start of their paths, resulting in lower RMSE values. However, as the quadrotor progresses, the dataset of in-range landmarks becomes similar across all scenarios, leading to comparable paths in these later stages.

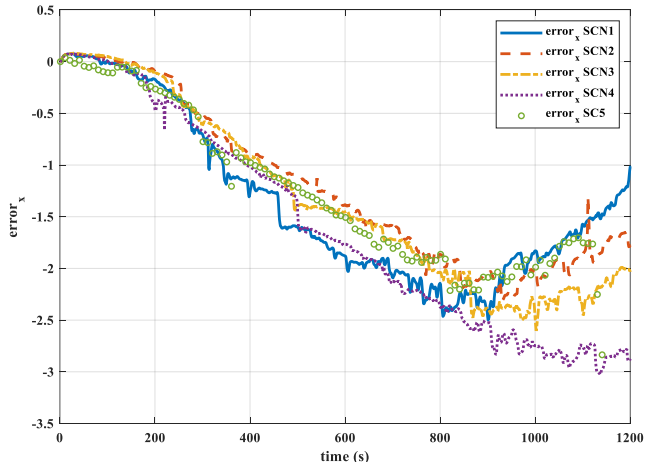


Fig. 12. Quadrotor x-position error with OA-OBPP for various initial position

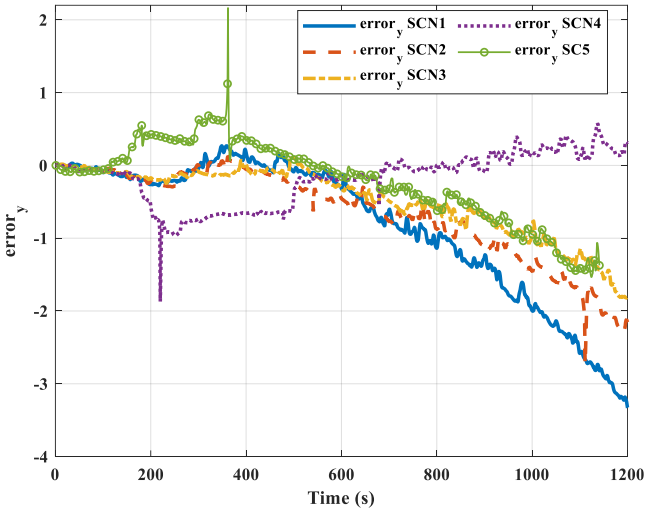


Fig. 13. Quadrotor y-position error with OA-OBPP for various initial position

Although the path during the second part of the quadrotor's motion is nearly identical across scenarios, variations in estimation accuracy during the initial part result in differing overall RMSE for the entire path.

## IV. FUTURE WORK

The proposed method shows promising results in simulations. However, two key considerations are essential for adapting this method to real-world applications. First, feature extraction and matching are crucial. The current research assumes that terrestrial landmarks are readily detected and available to the robot. In practical scenarios, however, these features must be extracted from images captured by visual sensors which requires advanced image processing algorithms and substantial computational power.

Another critical aspect for real-world implementation is the type of obstacles. In this research, it is assumed that obstacles are fixed in the environment. However, in real-world applications, obstacles can be both fixed (such as tall buildings, towers, and mountains) and dynamic (such as other aerial vehicles and birds). In dynamic environments, moving obstacles cause the list of nearby landmarks to change rapidly. This situation affects the obstacle avoidance

performance, as it depends on the quadrotor's velocity, the movement of obstacles, and the range of the quadrotor's vision.

The OA-OBPP method for ASLAM, may face challenges in real-world scenarios with numerous and moving obstacles, potentially leading to performance limitations. Future work should explore solutions for feature extraction and adapt the method to handle dynamic environments effectively to facilitate practical implementation.

## V. CONCLUSION

In this paper, autonomous motion of a quadrotor in a GNSS-denied unknown outdoor environment polluted with several obstacles is implemented utilizing OA-OBPP. The quadrotor navigates through such environment using EKF-SLAM while its path is defined by the OA-OBPP algorithm. In OBPP the quadrotor defines its path in the direction leads to most observability of system for better estimation. Utilizing the concept of observability degree and the landmarks distributed in the environment, quadrotor defines its path in most observable direction achieved by eigenvector of Gram matrix. Accordingly, OA is implemented by redefinition of the landmarks dataset. In this dataset the landmarks in a specified radius of the obstacle are removed from landmark dataset which is used for path planning.

To assess the performance of the proposed method, three categories of analysis is performed including Trajectory analysis, localization accuracy and effect of initial position. In first evaluation the results demonstrated by OA-OBPP quadrotor can autonomously moves from initial to the end point without any collision to the obstacles in the environment. The comparison of RMSE between the proposed method and the MCPP shows that the OA-OBPP method achieves approximately 30% lower localization error. Additionally, the evaluation of the impact of initial position on path planning with the proposed method demonstrates its robustness against variations in starting position. It can autonomously determine the most observable path from the initial point to the endpoint in an unknown, GNSS-denied environment while effectively avoiding collisions with obstacles.

## REFERENCES


[1] S. Bijjahalli, R. Sabatini, and A. Gardi, "Advances in intelligent and autonomous navigation systems for small UAS," *Prog. Aerosp. Sci.*, vol. 115, p. 100617, 2020, doi: 10.1016/J.PAEROSCI.2020.100617.

[2] X. Huang, P. Wang, X. Cheng, D. Zhou, Q. Geng, and R. Yang, "The ApolloScape Open Dataset for Autonomous Driving and Its Application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2702–2719, Oct. 2020, doi: 10.1109/TPAMI.2019.2926463.

[3] S. A. S. Mohamed, M. H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila, "A Survey on Odometry for Autonomous Navigation Systems," *IEEE Access*, vol. 7, pp. 97466–97486, 2019, doi: 10.1109/ACCESS.2019.2929133.

[4] N. Islam, K. Haseeb, A. Almogren, I. U. Din, M. Guizani, and A. Altameem, "A framework for topological based map building: A solution to autonomous robot navigation in smart cities," *Futur. Gener. Comput. Syst.*, vol. 111, pp. 644–653, Oct. 2020, doi: 10.1016/j.future.2019.10.036.

[5] J. Li, W. Zhan, Y. Hu, and M. Tomizuka, "Generic Tracking and Probabilistic Prediction Framework and Its Application in Autonomous Driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3634–3649, Sep. 2020, doi: 10.1109/TITS.2019.2930310.

[6] W. Youn, H. Ko, H. Choi, I. Choi, J. H. Baek, and H. Myung, "Collision-free Autonomous Navigation of A Small UAV Using Low-cost Sensors in GPS-denied Environments," *Int. J. Control. Autom. Syst.*, vol. 19, no. 2, pp. 953–968, Feb. 2021, doi: 10.1007/S12555-019-0797-7/METRICS.

[7] N. Gyagenda, J. V. Hatilima, H. Roth, and V. Zhmud, "A review of GNSS-independent UAV navigation techniques," *Rob. Auton. Syst.*, vol. 152, p. 104069, Jun. 2022, doi: 10.1016/J.ROBOT.2022.104069.

[8] A. Faghihinia, M. A. A. Atashgah, and S. M. M. Dehghan, "Model-Based Cooperative Navigation for a Group of Flying Robots," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 5, pp. 3895–3905, 2022, doi: 10.1109/TAES.2021.3136247.

[9] S. Rezwan and W. Choi, "Artificial Intelligence Approaches for UAV Navigation: Recent Advances and Future Challenges," *IEEE Access*, vol. 10, pp. 26320–26339, 2022, doi: 10.1109/ACCESS.2022.3157626.

[10] Y. Xue and W. Chen, "Multi-Agent Deep Reinforcement Learning for UAVs Navigation in Unknown Complex Environment," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 2290–2303, Jan. 2024, doi: 10.1109/TIV.2023.3298292.

[11] S. Zhang, Y. Li, and Q. Dong, "Autonomous navigation of UAV in multi-obstacle environments based on a Deep Reinforcement Learning approach," *Appl. Soft Comput.*, vol. 115, p. 108194, Jan. 2022, doi: 10.1016/J.ASOC.2021.108194.

[12] D. S. Chaplot, R. Salakhutdinov, A. Gupta, and S. Gupta, "Neural topological SLAM for visual navigation," *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12875-12884, 2020, doi: 10.1109/CVPR42600.2020.01289.

[13] T. Pire, J. Corti, and G. Grinblat, "Online Object Detection and Localization on Stereo Visual SLAM System," *J. Intell. Robot. Syst. Theory Appl.*, vol. 98, no. 2, pp. 377–386, May 2020, doi: 10.1007/s10846-019-01074-2.

[14] J. C. Trujillo, R. Munguia, E. Guerra, and A. Grau, "Cooperative monocular-based SLAM for multi-UAV systems in GPS-denied environments," *Sensors (Switzerland)*, vol. 18, no. 5, p. 1351, Apr. 2018, doi: 10.3390/s18051351.

[15] M. Holder, S. Hellwig, and H. Winner, "Real-time pose graph SLAM based on radar," in *IEEE Intelligent Vehicles Symposium, Proceedings*, pp. 1145–1151, Jun. 2019, doi: 10.1109/IVS.2019.8813841.

[16] M. Bozorg, M. S. Bahraini, and A. B. Rad, "A new adaptive UKF algorithm to improve the accuracy of SLAM," *International Journal of Robotics, Theory and Applications*, vol. 5, no. 1, pp. 35-46, 2019.

[17] Y. Zhang, T. Zhang, and S. Huang, "Comparison of EKF based SLAM and optimization based SLAM algorithms," *Proc. 13th IEEE Conf. Ind. Electron. Appl. ICIEA 2018*, pp. 1308–1313, 2018, doi: 10.1109/ICIEA.2018.8397911.

[18] A. Joukhadar, D. K. Hanna, A. Müller, and C. Stöger, "UKF-Assisted SLAM for 4WDDMR Localization and Mapping," *Mech. Mach. Sci.*, vol. 58, pp. 259–270, 2019, doi: 10.1007/978-3-319-89911-4_19.

[19] Z. L. Ren, L. G. Wang, and L. Bi, "Improved Extended Kalman Filter Based on Fuzzy Adaptation for SLAM in Underground Tunnels," *Int. J. Precis. Eng. Manuf.*, vol. 20, no. 12, pp. 2119–2127, Dec. 2019, doi: 10.1007/s12541-019-00222-w.

[20] G. Zhou, J. Luo, S. Xu, S. Zhang, S. Meng, and K. Xiang, "An EKF-based multiple data fusion for mobile robot indoor localization," *Assem. Autom.*, vol. 41, no. 3, pp. 274–282, 2021, doi: 10.1108/AA-12-2020-0199/FULL/XML.

[21] L. Carlone, J. Du, M. Kaouk Ng, B. Bona, and M. Indri, "Active SLAM and exploration with particle filters using Kullback-Leibler divergence," *J. Intell. Robot. Syst. Theory Appl.*, vol. 75, no. 2, pp. 291–311, Oct. 2014, doi: 10.1007/s10846-013-9981-9.

[22] Y. Chen, S. Huang, and R. Fitch, "Active SLAM for Mobile Robots with Area Coverage and Obstacle Avoidance," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1182–1192, Jun. 2020, doi: 10.1109/TMECH.2019.2963439.

[23] Y. B. Chen, G. C. Luo, Y. S. Mei, J. Q. Yu, and X. L. Su, "UAV path planning using artificial potential field method updated by optimal control theory," *Int. J. Syst. Sci.*, vol. 47, no. 6, pp. 1407–1420, Apr. 2016, doi: 10.1080/00207721.2014.929191.

[24] J. Jiang and Y. Ma, "Path planning strategies to optimize accuracy, quality, build time and material use in additive manufacturing: A

review," *Micromachines*, vol. 11, no. 7, p. 633, 2020, doi: 10.3390/MI11070633.

[25] R. Sharma, R. W. Beard, C. N. Taylor, and S. Quebe, "Graph-based observability analysis of bearing-only cooperative localization," *IEEE Trans. Robot.*, vol. 28, no. 2, pp. 522–529, Apr. 2012, doi: 10.1109/TRO.2011.2172699.

[26] S. Zhang, S. Wang, S. Yu, J. J. Q. Yu, and M. Wen, "Collision Avoidance Predictive Motion Planning Based on Integrated Perception and V2V Communication," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9640-9653, July 2022, doi: 10.1109/TITS.2022.3173674.

[27] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Rob. Res.*, vol. 30, no. 7, pp. 846–894, Jun. 2011, doi: 10.1177/0278364911406761.

[28] I. A. Hassan, I. A. Abed, and W. A. Al-Hussaibi, "Path Planning and Trajectory Tracking Control for Two-Wheel Mobile Robot," *J. Robot. Control*, vol. 5, no. 1, pp. 1–15, 2024, doi: 10.18196/jrc.v5i1.20489.

[29] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Comput. Commun.*, vol. 149, pp. 270–299, Jan. 2020, doi: 10.1016/j.comcom.2019.10.014.

[30] Y. Li, W. Wei, Y. Gao, D. Wang, and Z. Fan, "PQ-RRT*: An improved path planning algorithm for mobile robots," *Expert Syst. Appl.*, vol. 152, p. 113425, Aug. 2020, doi: 10.1016/j.eswa.2020.113425.

[31] N. Abcouwer *et al.*, "Machine Learning Based Path Planning for Improved Rover Navigation," *IEEE Aerosp. Conf. Proc.*, pp. 1-9, 2021, doi: 10.1109/AERO50100.2021.9438337.

[32] N. S. Abu, W. M. Bukhari, M. H. Adli, S. N. Omar, and S. A. Sohaimeh, "A Comprehensive Overview of Classical and Modern Route Planning Algorithms for Self-Driving Mobile Robots," *J. Robot. Control*, vol. 3, no. 5, pp. 666–678, Sep. 2022, doi: 10.18196/JRC.V3I5.14683.

[33] F. Zeng, C. Wang, and S. S. Ge, "A Survey on Visual Navigation for Artificial Agents with Deep Reinforcement Learning," *IEEE Access*, vol. 8. pp. 135426–135442, 2020, doi: 10.1109/ACCESS.2020.3011438.

[34] V. S. Kalogeiton, K. Ioannidis, G. C. Sirakoulis, and E. B. Kosmatopoulos, "Real-Time Active SLAM and Obstacle Avoidance for an Autonomous Robot Based on Stereo Vision," *Cybern. Syst.*, vol. 50, no. 3, pp. 239–260, Apr. 2019, doi: 10.1080/01969722.2018.1541599.

[35] R. Sharma, "Observability based control for cooperative localization," *2014 Int. Conf. Unmanned Aircr. Syst. ICUAS 2014 - Conf. Proc.*, pp. 134–139, 2014, doi: 10.1109/ICUAS.2014.6842248.

[36] S. Elahian, M. A. Amiri Atashgah, and B. Tarverdizadeh, "a Simultaneous Path Planning and Positioning Based on Artificial Distribution of Landmarks in a Gnss Denied Environment," *Aviation*, vol. 27, no. 1, pp. 36–46, 2023, doi: 10.3846/aviation.2023.18461.

[37] J. Cheng, Y. Sun, and M. Q. H. Meng, "Improving monocular visual SLAM in dynamic environments: an optical-flow-based approach," *Adv. Robot.*, vol. 33, no. 12, pp. 576–589, Jun. 2019, doi: 10.1080/01691864.2019.1610060.

[38] J. C. Trujillo, R. Munguia, S. Urzua, E. Guerra, and A. Grau, "Monocular Visual SLAM Based on a Cooperative UAV–Target System," *Sensors*, vol. 20, no. 12, p. 3531, Jun. 2020, doi: 10.3390/S20123531.

[39] R. Buchanan, L. Wellhausen, M. Bjelonic, T. Bandyopadhyay, N. Kottege, and M. Hutter, "Perceptive whole-body planning for multilegged robots in confined spaces," *J. F. Robot.*, vol. 38, no. 1, pp. 68-84, 2020, doi: 10.1002/rob.21974.

[40] A. Bettens, B. Morrell, M. Coen, X. Wu, P. Gibbens, and G. Chamitoff, "Simultaneous localization and mapping architecture for small bodies and space exploration," *Adv. Sp. Res.*, vol. 73, no. 1, pp. 1185–1197, Jan. 2024, doi: 10.1016/J.ASR.2023.10.048.

[41] S. Rahman, A. Quattrini Li, and I. Rekleitis, "SVIn2: A multi-sensor fusion-based underwater SLAM system," *The International Journal of Robotics Research*, vol. 41, no. 11–12, pp. 1022–1042, Jul. 2022, doi: 10.1177/02783649221110259.

[42] N. Elmeseiry, N. Alshaer, and T. Ismail, "A Detailed Survey and Future Directions of Unmanned Aerial Vehicles (UAVs) with Potential Applications," *Aerosp.*, vol. 8, no. 12, p. 363, Nov. 2021, doi: 10.3390/AEROSPACE8120363.

[43] S. D. Apostolidis, P. C. Kapoutsis, A. C. Kapoutsis, and E. B. Kosmatopoulos, "Cooperative multi-UAV coverage mission planning platform for remote sensing applications," *Auton. Robots*, vol. 46, no. 2, pp. 373–400, Feb. 2022, doi: 10.1007/S10514-021-10028-3/METRICS.

[44] X. Xu *et al.*, "A Review of Multi-Sensor Fusion SLAM Systems Based on 3D LIDAR," *Remote Sens.*, vol. 14, no. 12, 2022, doi: 10.3390/rs14122835.

[45] N. Delavarpour, C. Koparan, J. Nowatzki, S. Bajwa, and X. Sun, "A Technical Study on UAV Characteristics for Precision Agriculture Applications and Associated Practical Challenges," *Remote Sens.*, vol. 13, no. 6, p. 1204, Mar. 2021, doi: 10.3390/RS13061204.

[46] N. Amarasingam, A. S. Ashan Salgadoe, K. Powell, L. F. Gonzalez, and S. Natarajan, "A review of UAV platforms, sensors, and applications for monitoring of sugarcane crops," *Remote Sens. Appl. Soc. Environ.*, vol. 26, p. 100712, Apr. 2022, doi: 10.1016/J.RSASE.2022.100712.

[47] M. F. F. Rahman, S. Fan, Y. Zhang, and L. Chen, "A Comparative Study on Application of Unmanned Aerial Vehicle Systems in Agriculture," *Agric.*, vol. 11, no. 1, p. 22, Jan. 2021, doi: 10.3390/AGRICULTURE11010022.

[48] S. Granados-Bolaños, A. Quesada-Román, and G. E. Alvarado, "Low-cost UAV applications in dynamic tropical volcanic landforms," *J. Volcanol. Geotherm. Res.*, vol. 410, p. 107143, Feb. 2021, doi: 10.1016/J.JVOLGEORES.2020.107143.

[49] Z. Ameli, Y. Aremanda, W. A. Friess, and E. N. Landis, "Impact of UAV Hardware Options on Bridge Inspection Mission Capabilities," *Drones.*, vol. 6, no. 3, p. 64, 2022, doi: 10.3390/DRONES6030064.

[50] R. Adade, A. M. Aibinu, B. Ekumah, and J. Asaana, "Unmanned Aerial Vehicle (UAV) applications in coastal zone management—a review," *Environ. Monit. Assess.*, vol. 193, no. 3, pp. 1–12, Mar. 2021, doi: 10.1007/S10661-021-08949-8/METRICS.

[51] L. Xu, X. Shao, and W. Zhang, "USDE-Based Continuous Sliding Mode Control for Quadrotor Attitude Regulation: Method and Application," *IEEE Access*, vol. 9, pp. 64153–64164, 2021, doi: 10.1109/ACCESS.2021.3076076.

[52] M. Polic, A. Ivanovic, B. Maric, B. Arbanas, J. Tabak, and M. Orsag, "Structured Ecological Cultivation with Autonomous Robots in Indoor Agriculture," *Proc. 16th Int. Conf. Telecommun. ConTEL 2021*, pp. 189–195, Jun. 2021, doi: 10.23919/CONTEL52528.2021.9495963.

[53] N. Goyal, R. Aryan, N. Sharma, and V. Chhabra, "Line Follower Cargo-Bot For Warehouse Automation," *Int. Res. J. Eng. Technol*, vol. 8, pp. 1-8, 2021.

[54] M. Idrissi, M. Salami, and F. Annaz, "A Review of Quadrotor Unmanned Aerial Vehicles: Applications, Architectural Design and Control Algorithms," *J. Intell. Robot. Syst. Theory Appl.*, vol. 104, no. 2, pp. 1–33, Feb. 2022, doi: 10.1007/s10846-021-01527-7.

[55] I. Suwarno, W. A. Oktaviani, Y. Apriani, D. U. Rijalusalam, and A. Pandey, "Potential Force Algorithm with Kinematic Control as Path Planning for Disinfection Robot," *J. Robot. Control*, vol. 3, no. 1, pp. 107–114, Jan. 2022, doi: 10.18196/JRC.V3I1.11528.

[56] Y. Rasekhipour, A. Khajepour, S. K. Chen, and B. Litkouhi, "A Potential Field-Based Model Predictive Path-Planning Controller for Autonomous Road Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1255–1267, May 2017, doi: 10.1109/TITS.2016.2604240.

[57] M. Walter and J. Leonard, "An experimental investigation of cooperative SLAM," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 37, no. 8, pp. 880–885, 2004, doi: 10.1016/s1474-6670(17)32091-8.

[58] X. Zhu, B. Yan, and Y. Yue, "Path planning and collision avoidance in unknown environments for usvs based on an improved d* Lite," *Appl. Sci.*, vol. 11, no. 17, p. 7863, Sep. 2021, doi: 10.3390/APP11177863/S1.

[59] J. K. Park and T. M. Chung, "Boundary-RRT* Algorithm for Drone Collision Avoidance and Interleaved Path Re-planning," *J. Inf. Process. Syst.*, vol. 16, no. 6, pp. 1324–1342, Jan. 2020, doi: 10.3745/JIPS.04.0196.

[60] L. Lu, C. Sampedro, J. Rodriguez-Vazquez, and P. Campoy, "Laser-based collision avoidance and reactive navigation using RRT∗ and signed distance field for multirotor UAVs," *2019 Int. Conf. Unmanned Aircr. Syst. ICUAS 2019*, pp. 1209–1217, Jun. 2019, doi: 10.1109/ICUAS.2019.8798124.

[61] W. Rahmaniar and A. E. Rakhmania, "Mobile Robot Path Planning in a Trajectory with Multiple Obstacles Using Genetic Algorithms," *J. Robot. Control*, vol. 3, no. 1, pp. 1–7, Jan. 2022, doi: 10.18196/JRC.V3I1.11024.

[62] L. Hu *et al.*, "A multiobjective optimization approach for COLREGs-Compliant path planning of autonomous surface vehicles verified on networked bridge simulators," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1167–1179, Mar. 2020, doi: 10.1109/TITS.2019.2902927.

[63] B. O. H. Eriksen, G. Bitar, M. Breivik, and A. M. Lekkas, "Hybrid Collision Avoidance for ASVs Compliant With COLREGs Rules 8 and 13–17," *Front. Robot. AI*, vol. 7, p. 475020, Feb. 2020, doi: 10.3389/FROBT.2020.00011/BIBTEX.

[64] H. T. L. Chiang and L. Tapia, "COLREG-RRT: An RRT-Based COLREGS-Compliant Motion Planner for Surface Vehicle Navigation," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2024–2031, Jul. 2018, doi: 10.1109/LRA.2018.2801881.

[65] L. Zhao and M. Il Roh, "COLREGs-compliant multiship collision avoidance based on deep reinforcement learning," *Ocean Eng.*, vol. 191, p. 106436, Nov. 2019, doi: 10.1016/J.OCEANENG.2019.106436.

[66] H. Lyu and Y. Yin, "COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields," *J. Navig.*, vol. 72, no. 3, pp. 588–608, May 2019, doi: 10.1017/S0373463318000796.

[67] S. Wen, X. Chen, C. Ma, H. K. Lam, and S. Hua, "The Q-learning obstacle avoidance algorithm based on EKF-SLAM for NAO autonomous walking under unknown environments," *Rob. Auton. Syst.*, vol. 72, pp. 29–36, Oct. 2015, doi: 10.1016/j.robot.2015.04.003.

[68] I. Maurovic, M. Seder, K. Lenac, and I. Petrovic, "Path Planning for Active SLAM Based on the D∗ Algorithm with Negative Edge Weights," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 48, no. 8, pp. 1321–1331, Aug. 2018, doi: 10.1109/TSMC.2017.2668603.

[69] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to Explore using Active Neural SLAM," *arXiv preprint arXiv:2004.05155*, Apr. 2020.

[70] S. G. Jin, M. U. Ahmed, J. W. Kim, Y. H. Kim, and P. K. Rhee, "Combining Obstacle Avoidance and Visual Simultaneous Localization and Mapping for Indoor Navigation," *Symmetry*., vol. 12, no. 1, p. 119, Jan. 2020, doi: 10.3390/SYM12010119.

[71] T. Małecki and J. Narkiewicz, "A collision avoidance algorithm in Simultaneous Localization and Mapping problem for mobile platforms," *J. Theor. Appl. Mech.*, vol. 60, no. 2, pp. 317–328, Apr. 2022, doi: 10.15632/JTAM-PL/149478.

[72] D. Fethi, A. Nemra, K. Louadj, and M. Hamerlain, "Simultaneous localization, mapping, and path planning for unmanned vehicle using optimal control," *Adv. Mech. Eng.*, vol. 10, no. 1, Jan. 2018, doi: 10.1177/1687814017736653.

[73] J. R. Sánchez-Ibáñez, C. J. Pérez-Del-pulgar, and A. García-Cerezo, "Path Planning for Autonomous Mobile Robots: A Review," *Sensors.*, vol. 21, no. 23, p. 7898, Nov. 2021, doi: 10.3390/S21237898.

[74] K. Katona, H. A. Neamah, and P. Korondi, "Obstacle Avoidance and Path Planning Methods for Autonomous Navigation of Mobile Robot," *Sensors*., vol. 24, no. 11, p. 3573, Jun. 2024, doi: 10.3390/S24113573.

[75] Y. Sun *et al.*, "Local Path Planning for Mobile Robots Based on Fuzzy Dynamic Window Algorithm," *Sensors*., vol. 23, no. 19, p. 8260, Oct. 2023, doi: 10.3390/S23198260.

[76] A. Wondosen and D. Shiferaw, "Fuzzy Logic Controller Design for Mobile Robot Outdoor Navigation," *arXiv preprint arXiv:2401.01756*, 2024.

[77] H. Qin, S. Shao, T. Wang, X. Yu, Y. Jiang, and Z. Cao, "Review of Autonomous Path Planning Algorithms for Mobile Robots," *Drones*., vol. 7, no. 3, p. 211, Mar. 2023, doi: 10.3390/DRONES7030211.

[78] S. Takahashi *et al.*, "Sensor Selection by Greedy Method for Linear Dynamical Systems: Comparative Study on Fisher-Information-Matrix, Observability-Gramian and Kalman-Filter-Based Indices," *IEEE Access*, vol. 11, pp. 67850–67864, 2023, doi: 10.1109/ACCESS.2023.3291415.

[79] W. Xu, D. He, Y. Cai, and F. Zhang, "Robots' State Estimation and Observability Analysis Based on Statistical Motion Models," *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 5, pp. 2030–2045, Oct. 2022, doi: 10.1109/TCST.2021.3133080.

[80] N. Powel and K. A. Morgansen, "Empirical Observability Gramian for Stochastic Observability of Nonlinear Systems," *arXiv preprint arXiv:2006.07451*, 2020.

[81] A. Nemati *et al.*, "Autonomous navigation of UAV through GPS-denied indoor environment with obstacles," *AIAA Infotech@ Aerospace*, p. 0715, 2015, doi: 10.2514/6.2015-0715.

[82] R. J. Prazenica *et al.*, "Vision- aided navigation for a free-flying unmanned robotic system to support interplanetary bodies prospecting and characterization missions," *AIAA Guidance, Navigation, and Control Conference*, p. 0888, 2016, doi: 10.2514/6.2016-0888.

[83] M. W. M. Gamini Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, Jun. 2001, doi: 10.1109/70.938381.

[84] H. Mailka, M. Abouzahir, and M. Ramzi, "An efficient end-to-end EKF-SLAM architecture based on LiDAR, GNSS, and IMU data sensor fusion for autonomous ground vehicles," *Multimed. Tools Appl.*, vol. 83, no. 18, pp. 56183–56206, May 2024, doi: 10.1007/S11042-023-17595-W/METRICS.

[85] L. De Souza Pinto, L. E. S. A. Filho, L. Mariga, C. L. N. Junior, and W. C. Cunha, "EKF-SLAM with Autonomous Exploration using a Low Cost Robot," *15th Annu. IEEE Int. Syst. Conf. SysCon 2021 - Proc.*, pp. 1-7, Apr. 2021, doi: 10.1109/SYSCON48628.2021.9447073.

[86] Y. Song, Z. Zhang, J. Wu, Y. Wang, L. Zhao, and S. Huang, "A Right Invariant Extended Kalman Filter for Object Based SLAM," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 1316–1323, Apr. 2022, doi: 10.1109/LRA.2021.3139370.

[87] S. Rauniyar, S. Bhalla, D. Choi, and D. Kim, "EKF-SLAM for Quadcopter Using Differential Flatness-Based LQR Control," *Electron.*,vol. 12, no. 5, p. 1113, Feb. 2023, doi: 10.3390/ELECTRONICS12051113.

[88] M. Bryson and S. Sukkarieh, "Active airborne localisation and exploration in unknown environments using inertial SLAM," *IEEE Aerosp. Conf. Proc.*, p. 13, 2006, doi: 10.1109/aero.2006.1655801.

[89] M. Bryson and S. Sukkarieh, "Observability analysis and active control for airborne SLAM," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 1, pp. 261–280, Jan. 2008, doi: 10.1109/TAES.2008.4517003.

[90] Q. Ge, J. Ma, S. Chen, Y. Wang, and L. Bai, "Observable Degree Analysis to Match Estimation Performance for Wireless Tracking Networks," *Asian J. Control*, vol. 19, no. 4, pp. 1259–1270, Jul. 2017, doi: 10.1002/asjc.1386.

[91] A. T. Falahati Nodeh and B. E. Mehrababni, "Investigating the Effect of Different Sensors on the Observer Performance in Vehicle Suspension System Based on the Observable Degree Analysis (In Persian)," *Modares Mech. Eng.*, vol. 19, no. 7, pp. 1675–1684, 2019.

[92] Y. Liang, L. Han, X. Dong, Q. Li, and Z. Ren, "An quantitative method for observability analysis and its application in SINS calibration," *Aerosp. Sci. Technol.*, vol. 103, p. 105881, Aug. 2020, doi: 10.1016/j.ast.2020.105881.

[93] L. Huang, J. Song, and C. Zhang, "Observability analysis and filter design for a vision inertial absolute navigation system for UAV using landmarks," *Optik (Stuttg).*, vol. 149, pp. 455–468, 2017, doi: 10.1016/j.ijleo.2017.09.060.

[94] F. M. Ham and R. Grover Brown, "Observability, Eigenvalues, and Kalman Filtering," *IEEE Trans. Aerosp. Electron. Syst.*, no. 2, pp. 269–273, 1983, doi: 10.1109/TAES.1983.309446.

[95] S. Lakshmivarahan, J. M. Lewis, and S. K. Reddy Maryada, "Observability Gramian and Its Role in the Placement of Observations in Dynamic Data Assimilation," *Data Assim. Atmos. Ocean. Hydrol. Appl. (Vol. IV)*, pp. 215–257, 2022, doi: 10.1007/978-3-030-77722-7_9.

[96] Y. Gao, K. Liu, C. Zhu, X. Zhang, and D. Zhang, "Co-Estimation of State-of-Charge and State-of- Health for Lithium-Ion Batteries Using an Enhanced Electrochemical Model," *IEEE Trans. Ind. Electron.*,

vol. 69, no. 3, pp. 2684–2696, Mar. 2022, doi: 10.1109/TIE.2021.3066946.

[97] J. Chen, Z. Liang, Y. Zhu, and J. Zhao, "Improving Kinematic Flexibility and Walking Performance of a Six-legged Robot by Rationally Designing Leg Morphology," *J. Bionic Eng.*, vol. 16, no. 4, pp. 608–620, Jul. 2019, doi: 10.1007/s42235-019-0049-9.

[98] R. Zanetti and C. N. D'Souza, "Observability Analysis and Filter Design for the Orion Earth-Moon Attitude Filter," *J. Guid. Control. Dyn.*, vol. 39, no. 2, pp. 201–213, Feb. 2016, doi: 10.2514/1.G001217.

[99] V. Tchuiev and V. Indelman, "Distributed Consistent Multi-Robot Semantic Localization and Mapping," in *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4649-4656, July 2020, doi: 10.1109/LRA.2020.3003275.

[100] T. Dam, G. Chalvatzaki, J. Peters, and J. Pajarinen, "Monte-Carlo Robot Path Planning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 11213–11220, Oct. 2022, doi: 10.1109/LRA.2022.3199674.

[101] A. Abrishamifar, Ahmad Ale Ahmad and S. Elahian, "Fixed frequency sliding mode controller for the buck converter," *2011 2nd Power Electronics, Drive Systems and Technologies Conference*, pp. 557-561, 2011, doi: 10.1109/PEDSTC.2011.5742481.