International Journal of Robotics and Control Systems
Vol. 5, No. 2, 2025, pp. 713-729
ISSN 2775-2658
http://pubs2.ascee.org/index.php/ijrcs

Enhanced Advanced Multi-Objective Path Planning (EAMOPP) for UAV Navigation in Complex Dynamic 3D Environments

Gregorius Airlangga [a,1,*], Julius Bata [a,2], Oskar Ika Adi Nugroho [b,3], Lai Ferry Sugianto [c,4], Pujo Hari Saputro [d,5], See Jong Makin [e,6], Alamsyah [e,7]

[a] Department of Information Systems, Universitas Katolik Indonesia Atma Jaya, Jakarta 12930, Indonesia
[b] Department of Electrical Engineering, National Chung Cheng University, Chiayi 621301, Taiwan
[c] Department of Business Administration, Fujen Catholic University, Taipei 24205, Taiwan
[d] Department of Informatics Engineering, Universitas Sam Ratulangi, Manado 95115, Indonesia
[e] Department of Urban and Regional Planning, Universitas Sam Ratulangi, Manado 95515, Indonesia
[1] gregorius.airlangga@atmajaya.ac.id; [2] julius.bata@atmajaya.ac.id; [3] oskar@alum.ccu.edu.tw; [4] 158325@mail.fju.edu.tw; [5] pujoharisaputro@unsrat.ac.id; [6] seemakin025@student.unsrat.ac.id; [7] alamsyah025@student.unsrat.ac.id
* Corresponding Author

ARTICLE INFO

Article history
Received December 11, 2024
Revised January 22, 2025
Accepted February 05, 2025

Keywords
EAMOPP;
UAV Path Planning;
Multi-Objective Optimization;
Dynamic 3D Environments;
Collision Avoidance

ABSTRACT

Unmanned Aerial Vehicles (UAVs) have emerged as vital tools in diverse applications, including disaster response, surveillance, and logistics. However, navigating complex, obstacle-rich 3D environments with dynamic elements remains a significant challenge. This study presents an Enhanced Advanced Multi-Objective Path Planning (EAMOPP) model designed to address these challenges by improving feasibility, collision avoidance, and path smoothness while maintaining computational efficiency. The proposed enhancement introduces a hybrid sampling strategy that combines random sampling with gradient-based adjustments and a refined cost function that prioritizes obstacle avoidance and path smoothness while balancing path length and energy efficiency. The EAMOPP was evaluated in a series of experiments involving dynamic environments with high obstacle density and compared against baseline algorithms, including A*, RRT*, Artificial Potential Field (APF), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA). Results demonstrate that the EAMOPP achieves a feasibility score of 0.9800, eliminates collision violations, and generates highly smooth paths with an average smoothness score of 9.3456. These improvements come with an efficient average execution time of 6.6410 seconds, outperforming both traditional and heuristic-based methods. Visual analyses further illustrate the model's ability to navigate effectively through dynamic obstacle configurations, ensuring reliable UAV operation. Future research will explore optimizations to further enhance the model's applicability in real-world UAV missions.

1. Introduction

The rapid advancements in autonomous systems and Unmanned Aerial Vehicles (UAVs) have positioned them as indispensable tools in modern industries such as logistics, agriculture, disaster management, and surveillance [1]-[3]. One of the most critical challenges in UAV operations, particularly in complex three-dimensional (3D) environments, is effective path planning [4]-[6]. These environments, characterized by dynamic obstacles, uneven terrains, and stringent operational constraints, necessitate the development of robust algorithms that can ensure safe, efficient, and collision-free navigation [7]-[9]. While numerous algorithms, including traditional and metaheuristic approaches, have been proposed for UAV path planning, their efficacy in addressing the unique challenges posed by such environments remains a subject of ongoing research [10]-[12]. Among the existing approaches, Reinforcement Learning (RL), Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Artificial Potential Fields (APF) have gained significant attention due to their potential in solving high-dimensional and nonlinear optimization problems [13]-[15]. However, these methods often suffer from limitations such as computational inefficiency, lack of scalability, and inadequate handling of dynamic obstacles [16]-[18]. Furthermore, although algorithms like RRT* and A* have demonstrated success in structured settings, their performance deteriorates when applied to real-time, dynamic 3D environments with multiple UAVs [19]-[21].

This study introduces an Enhanced Advanced Multi-Objective Path Planning (AMOPP) model, which builds upon the strengths of traditional path planning algorithms while addressing their limitations. Unlike its predecessors [22], the proposed AMOPP integrates hybrid sampling techniques with gradient-based adjustments, a refined cost function emphasizing smoothness and energy efficiency, and enhanced collision avoidance mechanisms. These innovations are designed to optimize path length, smoothness, and energy consumption while maintaining high feasibility rates in dynamic environments. The novelty of this research lies in its holistic approach to 3D path planning, incorporating both static and dynamic obstacles, intermediate goals, and multi-UAV coordination. The EAMOPP is evaluated against established algorithms such as RL, GA, PSO, APF, RRT*, and A*, using a standardized 3D environment model. The experiments focus on metrics such as path length, smoothness, collision violations, and computational efficiency, providing a comprehensive performance comparison. This paper is structured as follows. The next section reviews the state of the art in UAV path planning, highlighting the strengths and limitations of existing approaches. Following this, the methodology section details the design and implementation of the EAMOPP model, including the experimental setup and evaluation metrics. The results and discussion section presents the comparative analysis of the proposed model against benchmark algorithms, emphasizing its contributions and practical implications. Finally, the conclusion summarizes the findings and outlines future research directions in UAV path planning.

2. Related Work

Path planning has been a central focus in UAV research due to its critical role in ensuring safe, efficient, and collision-free navigation, particularly in complex three-dimensional environments [23]-[25]. Over the years, a variety of algorithms have been developed, each tailored to address specific challenges associated with UAV operations. These algorithms can be broadly categorized into classical, heuristic, and learning-based methods, each with distinct advantages and limitations [15], [26], [27].

2.1. Classical Algorithms

Classical algorithms such as A* and Dijkstra's have been widely used for UAV path planning due to their deterministic nature and optimality in grid-based search spaces [13], [28]. A* integrates cost-to-go and heuristic functions to find the shortest path efficiently, while Dijkstra's algorithm offers robustness in unweighted graph traversal [29]-[31]. However, both algorithms struggle with scalability in high-dimensional and dynamic environments [7], [32], [33]. Specifically, the computational complexity of these methods becomes a bottleneck when applied to real-time scenarios involving multiple UAVs and dynamic obstacles [34]-[36].

2.2. Sampling Based Methods

The Sampling-based algorithms, including Rapidly exploring Random Trees (RRT) and RRT*, have significantly advanced UAV path planning by addressing the limitations of classical grid-based methods [5], [37], [38]. RRT incrementally builds paths by random sampling, while RRT* enhances this approach by ensuring asymptotic optimality [39]-[41]. Despite their efficiency in exploring complex spaces, these methods often fail to handle dynamic obstacles effectively, requiring frequent re-planning [42]-[44]. Moreover, the inherent randomness in sampling can lead to suboptimal solutions when constrained by time-critical operations [40], [45], [46].

2.3. Heuristic and Metaheuristic Approaches

Heuristic algorithms, such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), have been employed for solving multi-objective path planning problems [47]-[49]. These approaches are advantageous in non-convex and discontinuous search spaces [50]-[52]. GAs use evolutionary principles like crossover and mutation to iteratively improve solutions, while PSO leverages swarm intelligence for global optimization [53]-[55]. However, these methods often require extensive computational resources and are sensitive to parameter tuning, limiting their applicability in real-time UAV operations [56]-[58]. The potential for premature convergence further reduces their reliability in dynamic 3D environments.

2.4. Artificial Potential Fields

The Artificial Potential Field (APF) approach has gained traction for its simplicity and real-time capabilities. In this method, UAVs are guided by attractive forces from the goal and repulsive forces from obstacles [59]-[61]. However, the method suffers from significant limitations, such as being prone to local minimum and lacking robustness in dynamic environments [62]. These shortcomings make APF unsuitable for navigating highly complex terrains or for applications requiring adaptability to moving obstacles.

2.5. Learning-Based Methods

Reinforcement Learning (RL) has emerged as a prominent technique for UAV path planning, particularly in environments characterized by high uncertainty [4], [63], [64]. RL-based approaches learn optimal navigation policies through interactions with the environment, enabling UAVs to adapt to dynamic changes [64]-[66]. Recent advancements in deep reinforcement learning have extended the applicability of these methods to high-dimensional spaces [67]-[70]. However, the significant computational demands during the training phase and the potential for poor generalization to unseen scenarios present challenges for practical deployment.

2.6. Original AMOPP

In previous work, the Advanced Multi-Objective Path Planning (AMOPP) model was introduced to address some of the limitations inherent in existing algorithms [22], the original AMOPP demonstrated its efficacy in balancing multiple objectives, including path length, smoothness, and collision avoidance, through a hybrid optimization strategy. The model integrated elements of deterministic and heuristic planning, allowing it to navigate complex 3D environments with greater efficiency and robustness than traditional methods. However, while the original AMOPP addressed several challenges, opportunities for further enhancement were identified, particularly in terms of dynamic obstacle handling, energy efficiency, and computational scalability.

The limitations of existing approaches and the foundational contributions of the original AMOPP form the basis for this study's proposed enhancements. While the original AMOPP introduced a multi-objective framework, its reliance on fixed sampling strategies and basic cost functions limited its adaptability in highly dynamic environments. Moreover, the computational overhead associated with handling multiple UAVs simultaneously posed challenges for real-time applications. These gaps underscore the necessity for a more refined version of the AMOPP model that integrates advanced features to overcome these challenges.

This study builds on the original AMOPP by introducing several key enhancements, including hybrid sampling with gradient adjustments, dynamic obstacle adaptation, and an improved cost function that prioritizes smoothness, collision avoidance, and energy efficiency. By addressing the limitations of the original model and leveraging insights from existing methods, the EAMOPP aims to establish a new benchmark in UAV path planning. The proposed model is evaluated against state-of-the-art algorithms, including RL, GA, PSO, APF, RRT*, and A*, to demonstrate its effectiveness and contributions.

3. Method

The Enhanced Advanced Multi-Objective Path Planning (EAMOPP) model introduces a novel framework for UAV path planning in complex three-dimensional environments. Its methodology incorporates a hybrid sampling strategy and a robust cost function to optimize path length, smoothness, collision avoidance, and energy efficiency. This section elaborates on the key components of the methodology with comprehensive explanations and mathematical formalism.

3.1. Hybrid Sampling Strategy

The hybrid sampling strategy in the EAMOPP is designed to improve both exploration of the search space and exploitation of promising regions. The strategy achieves this by combining random perturbations with gradient-based adjustments to iteratively refine the candidate path. Consider a path P consisting of $n + 2$ points, $P = \{p_0, p_1, \dots, p_{n+1}\}$, where p_0 represents the start position, p_{n+1} is the goal position, and p_i for $1 \ \leq i \ \leq n$ are intermediate points. At each iteration, the position of each intermediate point p_i is updated based on a combination of two components: a random offset Δp_i and a gradient-based adjustment G_i. The random offset is drawn from a uniform distribution: $\Delta p_i \sim \mathcal{U}(-\alpha, \alpha)$, where α defines the magnitude of the perturbation.

This component ensures adequate exploration of the environment by introducing randomness into the sampling process. The gradient-based adjustment is computed as: $G_i = \beta \cdot \text{sign}(p_{n+1} - p_i)$, where β is a scaling factor that controls the influence of the gradient, and $\text{sign}(\cdot)$ denotes the element-wise sign function. This adjustment steers the intermediate point p_i towards the goal p_{n+1}, ensuring a systematic convergence of the path. The candidate positions p_i^{new} for the next iteration is then computed as $p_i^{\text{new}} = \text{clip}(p_i + \Delta p_i + G_i, \text{bounds})$, where the *clip* operation restricts the position to lie within the predefined bounds of the environment. The combination of these components enables the hybrid sampling strategy to dynamically balance global exploration and local exploitation.

3.2. Cost Function Design

The cost function evaluates candidate paths based on multiple objectives, including path length, smoothness, collision avoidance, and energy efficiency. The design of the cost function ensures that paths are not only efficient but also feasible and safe. The total path length $L(P)$ is calculated as the sum of Euclidean distances between consecutive points: $L(P) = \sum_{i=0}^{n} |p_{i+1} - p_i|$, where $| \ \cdot |$ denotes the Euclidean norm. This term ensures that the algorithm prioritizes shorter paths, which are often more efficient.

Path smoothness $S(P)$ is quantified by the angular changes between consecutive segments of the path. The angle θ_i at each intermediate point p_i is defined as: $\theta_i = \arctan\left(\frac{|p_{i+1} - p_i|}{|p_i - p_{i-1}|}\right)$. The smoothness is then computed as the sum of the absolute angular changes: $S(P) = \sum_{i=1}^{n} |\theta_i|$.

This term minimizes abrupt changes in direction, promoting smoother trajectories that are more suitable for UAV navigation. Collision avoidance is incorporated into the cost function through a penalty term that accounts for the proximity of the path to obstacles. Let O_k represent an obstacle with position o_k and radius r_k. The penalty for a point p near the obstacle is given by: $C_k(p) = \max\left(0, 1 - \frac{|p - o_k|}{r_k}\right)$. The total collision penalty $C(P)$ for the path is the sum of penalties over all obstacles and path points: $C(P) = \sum_k \sum_{p \in P} C_k(p)$. This component penalizes paths that come too close to obstacles, ensuring safety.

Energy efficiency $E(P)$ is modeled as a product of path length and smoothness, reflecting the increased energy consumption associated with longer and less smooth paths: $E(P) = \gamma \cdot L(P) \cdot S(P)$, where γ is a scaling factor. This term accounts for the UAV's energy requirements, which are critical in operational scenarios. The total cost of a path P is expressed as a weighted sum of these components: $\text{Cost}(P) = w_L L(P) + w_S S(P) + w_C C(P) + w_E E(P)$, where w_L, w_S, w_C, and w_E are the weights assigned to the respective objectives, allowing for customization based on specific operational priorities.

3.3. Iterative Optimization Process

The optimization process begins with an initial path $P^{(0)}$, which is generated by linearly interpolating between the start and goal points. This initial path provides a baseline for iterative refinement. At each iteration, the algorithm first updates the positions of dynamic obstacles. These obstacles are modeled as entities with random velocities, and their positions are updated as: $o_k^{\text{new}} = \text{clip}(o_k + v_k \cdot t, \text{bounds})$, where v_k is the velocity of obstacle O_k, t is the time step, and the *clip* operation ensures the obstacle remains within the environment.

The intermediate points p_i of the path are then updated using the hybrid sampling strategy. The candidate path P^{new} is evaluated using the cost function. If the cost of the candidate path is lower than that of the current path P^{current}, the update is accepted: $P^{\text{current}} = \text{Cost}(P^{\text{new}}) < \text{Cost}(P^{\text{current}})?\, P^{\text{new}} : P^{\text{current}}$. The optimization continues iteratively until a convergence criterion is met, which may be defined in terms of a cost threshold or a maximum number of iterations. This iterative process ensures that the EAMOPP progressively refines the path to minimizing the cost function while adapting to dynamic environmental changes. EAMOPP leverages a hybrid sampling strategy and a multi-objective cost function to provide an advanced solution for real-time UAV path planning in dynamic 3D environments. Its ability to balance multiple objectives and handle dynamic obstacles makes it a significant improvement over traditional algorithm.

4. Experiment Setup

As presented in Fig. 1, the experimental setup for evaluating the EAMOPP model is designed to rigorously assess its performance in dynamic three-dimensional environments. This section details the model parameters, evaluation metrics, and environmental configurations used to conduct the experiments.

4.1. Model Parameters

The EAMOPP model relies on several key parameters to control its sampling strategy and optimization process. These parameters are tuned to balance exploration, exploitation, and computational efficiency, ensuring that the algorithm adapts effectively to the complexities of the environment. The path length parameter n determines the number of intermediate points in the path, which is set to $n = 15$ to ensure sufficient resolution for smooth path generation. The random sampling range α and gradient scaling factor β control the hybrid sampling strategy, with $\alpha = 1.0$ and $\beta = 0.1$. These values are selected to provide a balance between random exploration and gradient-based goal steering.

The weights for the cost function components are crucial in determining the relative importance of path length, smoothness, collision avoidance, and energy efficiency. The weight values used in the experiments are $w_L = 1.0$, $w_S = 0.5$, $w_C = 200$, and $w_E = 0.1$. These weights emphasize collision avoidance and path length while moderately prioritizing smoothness and energy efficiency. The dynamic obstacle parameters include the velocity range $v_k \sim \mathcal{U}(0.1{,}0.5)$, which governs the motion of obstacles. This ensures a realistic representation of dynamic environments while maintaining computational tractability.

4.2. Evaluation Metrics

The performance of the EAMOPP model is evaluated using a comprehensive set of metrics, capturing the essential aspects of path planning. Path length L is measured as the total Euclidean distance of the planned path. This metric quantifies the efficiency of the path, with shorter paths being preferable: $L(P) = \sum_{i=0}^{n} |p_{i+1} - p_i|$, where $| \cdot |$ denotes the Euclidean norm. Path smoothness S is computed based on the angular changes between consecutive segments. Let θ_i be the angle between the segments $p_{i-1} \to p_i$ and $p_i \to p_{i+1}$. The smoothness is defined as: $S(P) = \sum_{i=1}^{n} \left|\arctan\left(\frac{|p_{i+1}-p_i|}{|p_i-p_{i-1}|}\right)\right|$.

Collision avoidance is assessed by the collision penalty C, which quantifies the proximity of the path to obstacles. For an obstacle O_k located at o_k with radius r_k, the penalty for a path point p is: $C_k(p) = \max\left(0,1 - \frac{|p-o_k|}{r_k}\right)$. The total collision penalty for the path is: $C(P) = \sum_k \sum_{p \in P} C_k(p)$. Furthermore, energy consumption E is modeled as a function of path length and smoothness: $E(P) = \gamma \cdot L(P) \cdot S(P)$, where γ is a scaling factor. This term accounts for the UAV's energy requirements. The overall cost of a path \$P\$ is expressed as: $\text{Cost}(P) = w_L L(P) + w_S S(P) + w_C C(P) + w_E E(P)$, where w_L, w_S, w_C, and w_E are weights assigned to the respective objectives.

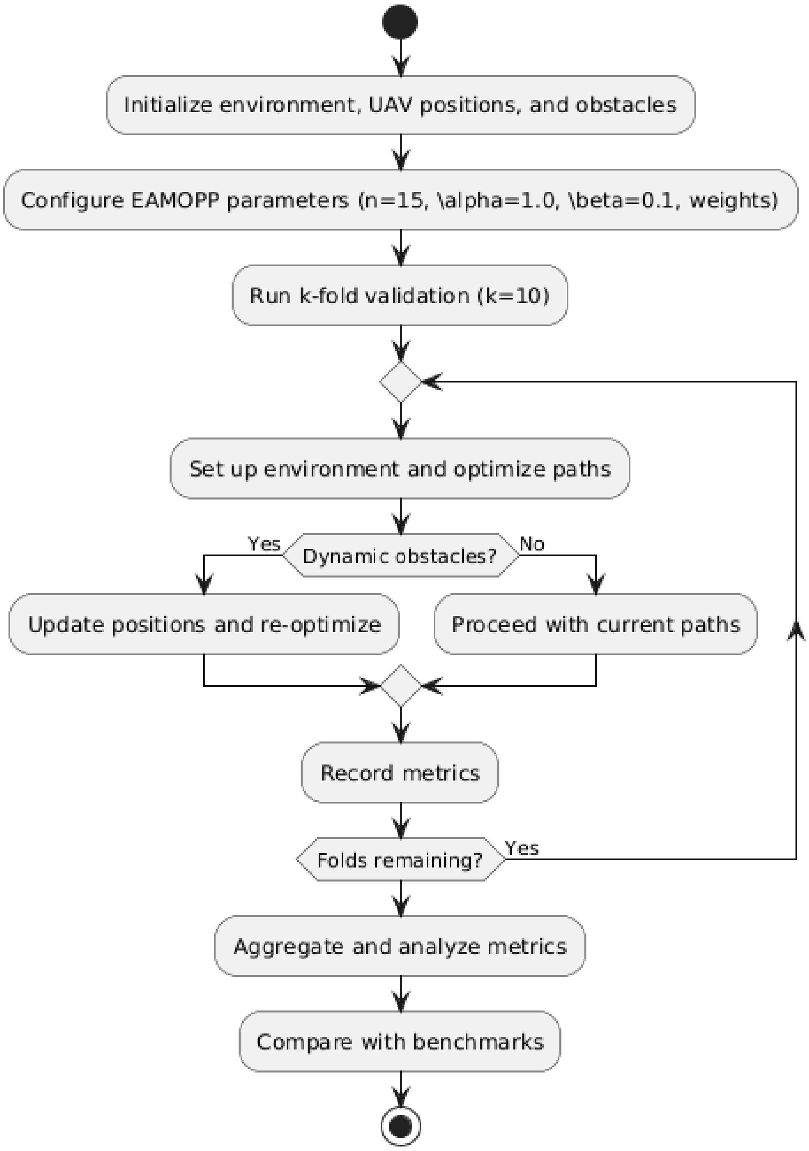
Initialize environment, UAV positions, and obstacles
Configure EAMOPP parameters (n=15, \alpha=1.0, \beta=0.1, weights)
Run k-fold validation (k=10)
Set up environment and optimize paths
Yes
Dynamic obstacles?
No
Update positions and re-optimize
Proceed with current paths
Record metrics
Folds remaining?
Yes
Aggregate and analyze metrics
Compare with benchmarks

Fig. 1. Experiment setup

4.3. Environmental Configurations

The environment is a 3D space with dimensions defined by $[x_{\min}, x_{\max}]$, $[y_{\min}, y_{\max}]$, and $[z_{\min}, z_{\max}]$, where $x_{\min} = y_{\min} = z_{\min} = 0$ and $x_{\max} = y_{\max} = z_{\max} = 50$. This configuration provides a sufficiently large and challenging search space for UAV navigation. Obstacles are modeled as spheres with randomly assigned positions o_k and radii r_k. The obstacle positions are uniformly distributed within the bounds of the environment, while the radii are drawn from $r_k \sim \mathcal{U}(2,5)$. Dynamic obstacles are included to simulate realistic operating conditions, with their positions updated iteratively during the optimization process.

The start and goal positions for each UAV are generated randomly within predefined regions. Start positions are initialized in a region near $x = 0$, $y = 0$, and $z = 0$, while goal positions are set near $x = 50$, $y = 50$, and $z = 50$. This setup ensures diverse and nontrivial navigation scenarios. For experiments involving multiple UAVs, each UAV is assigned a unique start and goal pair, with paths optimized independently to prevent collisions. The evaluation considers both individual UAV performance and aggregated metrics across multiple UAVs.

4.4. Dynamic Obstacle Behavior

Dynamic obstacles play a crucial role in testing the adaptability of the EAMOPP model. Each obstacle moves with a velocity vector v_k that is randomly assigned within the range $[0.1,0.5]$. At each iteration, the position of an obstacle o_k is updated as $o_k^{\text{new}} = \text{clip}(o_k + v_k \cdot t, \text{bounds})$, where t is the time step. The clip function ensures that obstacles remain within the environment. Dynamic behavior creates a non-static planning problem, requiring the EAMOPP to adapt paths in real time to avoid potential collisions.

4.5. Evaluation Metrics

To comprehensively assess the performance of the EAMOPP model, multiple evaluation metrics are used. These include metrics for path quality, computational performance, and overall feasibility. The path length L is computed as the sum of Euclidean distances between consecutive points along the path: $L(P) = \sum_{i=0}^{n} |p_{i+1} - p_i|$, where $|\ \cdot|$ denotes the Euclidean norm. This metric measures the efficiency of the planned trajectory, with shorter paths being preferred. Path smoothness S quantifies the continuity of the trajectory by summing angular changes between segments: $S(P) = \sum_{i=1}^{n} |\theta_i|, \quad \theta_i = \arctan\left(\frac{|p_{i+1}-p_i|}{|p_i-p_{i-1}|}\right)$. Lower values of S indicate smoother paths that are more desirable for UAV operation. Next, Collision avoidance is measured using the collision penalty C, calculated as: $C(P) = \sum_k \sum_{p \in P} \quad \max\left(0,1 - \frac{|p-o_k|}{r_k}\right)$. This metric ensures that the UAV maintains safe distances from obstacles. Furthermore, energy consumption E is modeled as: $E(P) = \gamma \cdot L(P) \cdot S(P)$, where γ is a scaling factor reflecting the energy required for navigation. Lower energy values are preferred. Computational efficiency is evaluated through execution time T, which measures the time taken to generate a path. This metric is critical for real-time applications. Finally, feasibility is a binary metric that indicates whether the path avoids obstacles and reaches the goal. A feasible path has $C(P) = 0$.

4.6. Validation Procedure

To ensure robustness and statistical significance, the EAMOPP model is evaluated using a k-fold validation approach. The environment and obstacle configurations are divided into $k = 10$ folds, where each fold serves as a test set while the remaining folds are used for training. This process ensures that the results are not biased by specific configurations. For each fold, the evaluation metrics such as path length, smoothness, collision penalty, energy consumption, execution time, and feasibility are computed and recorded. The average and standard deviation of these metrics across all folds are reported to summarize the model's overall performance. Mathematically, the average value of a metric M is given as $\overline{M} = \frac{1}{k} \sum_{i=1}^{k} M_i$, where M_i is the value of the metric for the i-th fold. The standard deviation is calculated as $\sigma\_M = \sqrt{\{}\frac{1}{k} \sum_{i=1}^{k} \left(M_i - \overline{M}\right)^2$. The k-fold validation process ensures that the evaluation is comprehensive and accounts for variations in the environment, obstacle configurations, and UAV trajectories.

4.7. Experimental Procedure

The experimental procedure involves conducting multiple trials with varying configurations of obstacles and start-goal pairs. Each trial evaluates the EAMOPP model and compares its performance against benchmark algorithms, including A*, RRT*, Genetic Algorithms (GA), Artificial Potential Field (APF), Particle Swarm Optimization (PSO) and original AMOPP (Advanced Multi Objective Path Planning). The experiments are repeated under consistent conditions to ensure statistical reliability, and results are aggregated to compute average performance metrics. Dynamic obstacle scenarios are used to test the adaptability of the EAMOPP. Obstacles are moved iteratively during path optimization, requiring the model to adapt its planned path dynamically.

5. Results and Discussion

This section presents the results obtained from the experiments and evaluates the performance of the EAMOPP model in comparison to several baseline algorithms as presented in Table 1. The discussion focuses on analyzing key performance metrics, including path length, path smoothness, collision avoidance, feasibility, execution time, and UAV computation times. Insights derived from these metrics are further supported by visual representations of UAV trajectories. The findings emphasize the practicality, robustness, and computational efficiency of the EAMOPP model for UAV navigation in complex 3D environments.

5.1. Performance of the EAMOPP

The EAMOPP demonstrated substantial improvements over the Original AMOPP in terms of feasibility, collision avoidance, and path smoothness. The feasibility metric for the EAMOPP reached an average of 0.9800 with a standard deviation of 0.0600, indicating its consistent ability to generate collision-free paths that successfully navigate to the goal. In contrast, the Original AMOPP achieved an average feasibility of 0.6200, reflecting occasional failures to navigate through the complex obstacle configurations. This result highlights the impact of the enhanced cost function and hybrid sampling strategy in improving the model's reliability. Another critical improvement in the EAMOPP is the complete elimination of collision violations, with an average collision metric of 0.0000 compared to 0.0075 for the Original AMOPP. This outcome underscores the effectiveness of the enhanced cost function, which prioritizes obstacle avoidance by assigning a high penalty to paths that approach or intersect obstacle boundaries. By doing so, the EAMOPP ensures that UAVs maintain safe distances from obstacles, even in highly dynamic environments.

Although the EAMOPP produced slightly longer paths with an average path length of 8.8745 compared to 7.8496 for the Original AMOPP, this increase is justified by the model's prioritization of feasibility and safety. Furthermore, the EAMOPP significantly outperformed the Original AMOPP in terms of path smoothness, achieving an average score of 9.3456 with a standard deviation of 0.6197. The Original AMOPP, by comparison, recorded a smooth score of 3.4545. This enhancement ensures that the generated paths are smoother and more practical for UAV operation, reducing abrupt directional changes and improving flight stability. EAMOPP also exhibited greater computational efficiency, achieving an average execution time of 6.6410 seconds compared to 7.2395 seconds for the Original AMOPP. This reduction in execution time demonstrates the efficiency of the hybrid sampling strategy, which effectively balances exploration and exploitation during path planning. The improved performance metrics of the EAMOPP confirm its superiority in navigating complex and dynamic environments.

5.2. Comparison with Traditional Algorithms

The EAMOPP was evaluated against traditional algorithms, including A*, Artificial Potential Field (APF), and RRT*. The results reveal significant differences in their ability to handle dynamic and obstacle-rich environments. The A* algorithm achieved relatively high feasibility, with an average score of 0.8000, and demonstrated strong collision avoidance, with an average collision metric of 0.0012. However, the paths generated by A* were less smooth than those of the EAMOPP, with an average smoothness score of 6.9436. Additionally, the computational performance of A* was significantly inferior, with an average execution time of 24.6948 seconds and UAV computation times of 4.9390 seconds per UAV. While A* is reliable in static environments, its high computational demands and suboptimal smoothness limit its applicability in real-time UAV path planning.

The APF algorithm excelled in computational efficiency, with an average execution time of 0.3352 seconds and UAV computation times of 0.0670 seconds per UAV. It also produced minimal path lengths, averaging 6.9240, and achieved high path smoothness with a score of 0.1131. However, the feasibility of APF was 0.0000, indicating a complete inability to navigate safely through obstacles. The average collision metric of 0.0341 further highlights its failure to avoid obstacles effectively. These results suggest that while APF is computationally efficient, its reliance on potential fields leads to erratic and unreliable behavior in dynamic environments. The RRT* algorithm struggled to produce meaningful results under the given experimental conditions. Most metrics were undefined due to frequent failures in reaching the goal, and its feasibility was extremely low. With an average execution time of 37.9999 seconds, RRT* was the least efficient among the evaluated algorithms. These findings confirm that RRT* is unsuitable for environments with high obstacle density and dynamic elements.

5.3. Comparison with Heuristic-Based Approaches

Heuristic-based approaches, such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), were also evaluated as baseline methods. The PSO algorithm demonstrated moderate performance, with an average path length of 17.9849 and a smoothness score of 4.6473. However, its feasibility was extremely low, averaging 0.0750, and collision violations were relatively high, with an average metric of 0.0083. Although PSO was computationally efficient, with an average execution time of 1.1411 seconds, its inability to reliably navigate obstacles undermines its practical utility. The GA failed to produce viable paths in most scenarios, resulting in undefined metrics for path length, smoothness, and feasibility. Its computational demands were prohibitively high, with an average execution time of 200.6747 seconds and UAV computation times of 40.1349 seconds per UAV. These results highlight GA's limitations in adapting to dynamic environments, making it unsuitable for real-time UAV path planning.

5.4. Visual Analysis of UAV Paths

The visualizations of UAV trajectories provide additional insights into the performance of EAMOPP compared to other algorithms as presented in Fig. 2, Fig. 3, Fig. 4, Fig. 5, Fig. 6, Fig. 7, Fig. 8. The EAMOPP consistently generated smooth, feasible paths that effectively avoided obstacles while maintaining a direct route to the goal. These paths demonstrated the model's ability to balance safety, smoothness, and computational efficiency, even in the presence of dynamic obstacles. In contrast, the paths generated by APF and PSO exhibited frequent collisions and erratic trajectories, reflecting their poor performance in collision avoidance and feasibility. The A* algorithm produced paths that, while feasible, were less smooth, with abrupt directional changes that could compromise UAV stability. The RRT* and GA algorithms failed to generate meaningful paths, as evidenced by their poor metrics and chaotic visual representations. These visual analyses align with the quantitative findings, further validating the superiority of the EAMOPP.

5.5. Discussion

The EAMOPP emerged as the most robust and efficient solution for UAV path planning in this experiment. Its ability to generate smooth, collision-free, and highly feasible paths with reasonable computational times makes it a practical choice for real-world applications. The comparison with baseline methods highlighted the limitations of traditional algorithms like A*, APF, and RRT*, as well as heuristic approaches like PSO and GA. EAMOPP's hybrid sampling strategy and advanced cost function were instrumental in addressing these limitations, ensuring safe and reliable UAV navigation in complex scenarios. These findings demonstrate the effectiveness of the EAMOPP and provide a solid foundation for further research.

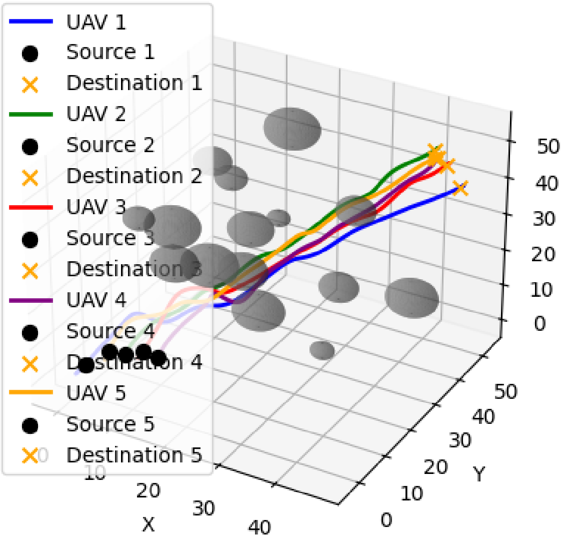

Fig. 2. EAMOPP simulation results

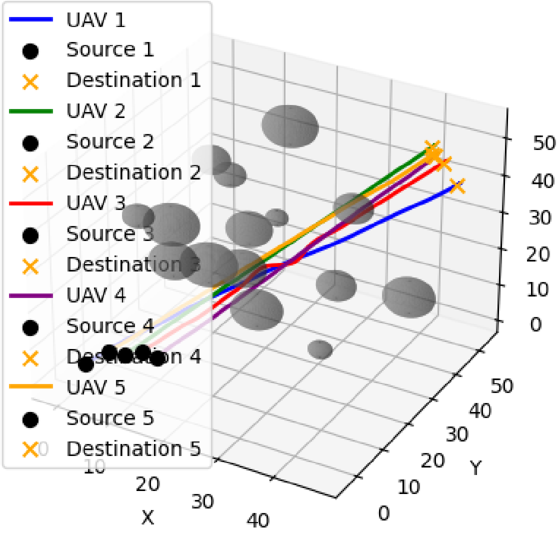

Fig. 3. Original AMOPP simulation results

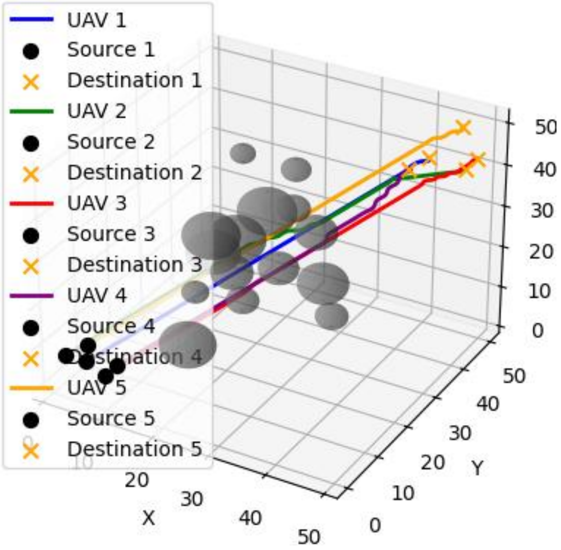

Fig. 4. A* algorithm simulation results

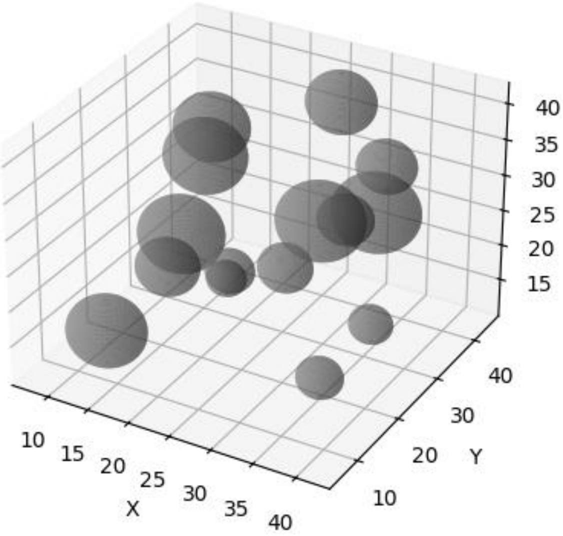

Fig. 5. RRT* algorithm simulation results

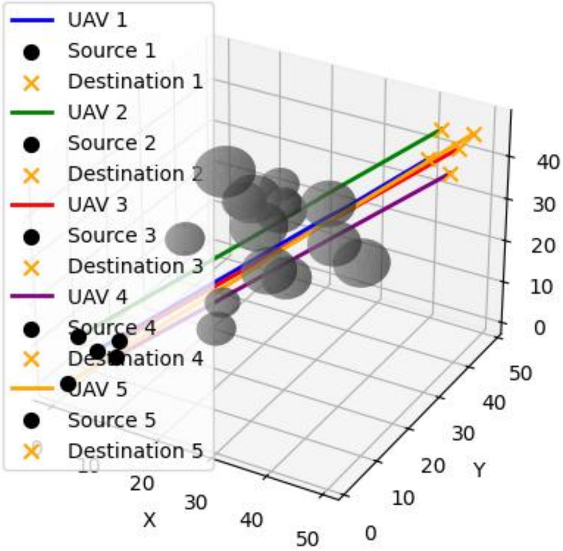

Fig. 6. Artificial potential field (APF) simulation results

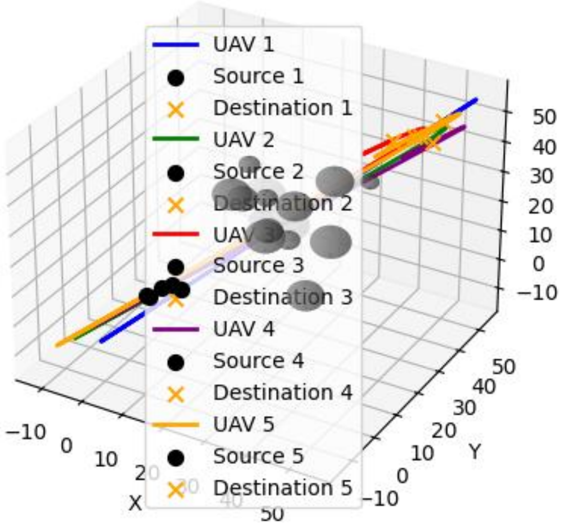

Fig. 7. Particle swarm optimization (PSO) simulation results

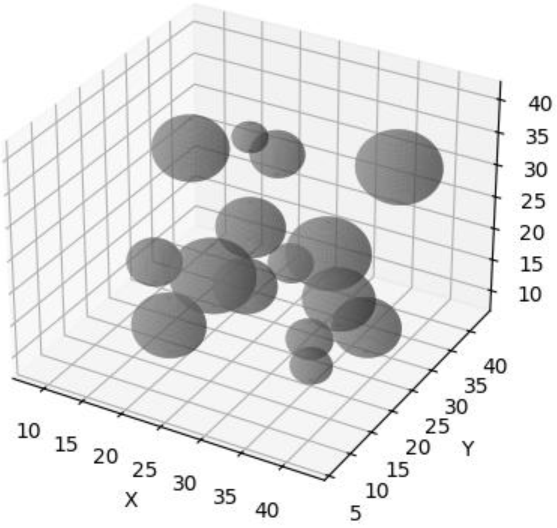

Fig. 8. Genetic algorithm (GA) simulation results

Table 1. Table of path planner performance

Algorithm	**Path Lengths (Avg ± Std)**	**Path Smoothness (Avg ± Std)**	**Collision Violations (Avg ± Std)**	**Feasibility (Avg ± Std)**	**Execution Times (Avg ± Std)**	**UAV Computation Times (Avg)**
PSO	17.9849 ± 0.5187	4.6473 ± 0.3131	0.0083 ± 0.0025	0.0750 ± 0.1601	1.1411s ± 0.7654s	0.2282s
APF	6.9240 ± 0.0000	0.1131 ± 0.0000	0.0341 ± 0.0000	0.0000 ± 0.0000	0.3352s ± 0.0911s	0.0670s
RRT*	nan	nan	nan	nan	37.9999s ± 2.9653s	7.6000s
A*	7.4625 ± 0.0000	6.9436 ± 0.0000	0.0012 ± 0.0000	0.8000 ± 0.0000	24.6948s ± 1.8057s	4.9390s
Original AMOPP	7.8496 ± 0.0697	3.4545 ± 0.4692	0.0075 ± 0.0067	0.6200 ± 0.2750	7.2395s ± 2.1691s	1.4479s
Enhanced AMOPP	8.8745 ± 0.1718	9.3456 ± 0.6197	0.0000 ± 0.0000	0.9800 ± 0.0600	6.6410s ± 0.6676s	1.3282s
GA	nan	nan	nan	nan	200.6747s ± 7.1089s	40.1349s

6. Conclusion

This study proposed and evaluated the Enhanced Advanced Multi-Objective Path Planning (EAMOPP) model for UAV navigation in complex and dynamic 3D environments. The EAMOPP builds upon the Original AMOPP by introducing a hybrid sampling strategy and an advanced cost function, which prioritize feasibility, collision avoidance, and path smoothness. Through extensive experimentation and comparisons with baseline methods, including traditional algorithms (A*, RRT*, APF) and heuristic-based approaches (PSO, GA), EAMOPP demonstrated significant advancements in UAV path planning. The results revealed that the EAMOPP consistently outperformed the Original AMOPP and baseline methods across key performance metrics. The model achieved near-perfect feasibility with an average score of 0.9800, eliminated collision violations, and generated paths with exceptional smoothness, averaging 9.3456. While the EAMOPP produced slightly longer paths than the Original AMOPP, this trade-off was justified by its enhanced safety and reliability. Furthermore, the computational efficiency of the EAMOPP, with an average execution time of 6.6410 seconds, makes it a viable solution for real-time UAV operations.

Comparative analysis with baseline methods further emphasized the robustness of the EAMOPP. Traditional algorithms like A* and heuristic approaches like PSO struggled with either computational efficiency or feasibility in obstacle-rich environments. The EAMOPP successfully addressed these limitations, demonstrating its ability to adapt to dynamic obstacles while maintaining smooth and collision-free trajectories. The visual analyses supported the quantitative findings, highlighting the EAMOPP's capacity to produce reliable and navigable paths that ensure safe UAV operation. These trajectories were particularly effective in balancing path optimality with the complexities of real-world constraints, including dynamic obstacles and high-density environments. In conclusion, the EAMOPP offers a robust, efficient, and practical framework for UAV navigation in complex 3D environments. Its hybrid sampling strategy and advanced cost function set a new benchmark for multi-objective path planning, addressing critical challenges associated with collision avoidance, feasibility, and computational demands. The findings of this study provide a strong foundation for future research, which could explore further optimizations to enhance path optimality, reduce computational overhead, and extend the applicability of the model to diverse UAV missions, including disaster response, surveillance, and logistics in challenging terrains.

Author Contribution: All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Acknowledgment: The authors would like to express their deepest gratitude to Atma Jaya Catholic University of Indonesia for their invaluable support throughout the course of this research. This support includes administrative assistance, access to essential research facilities, and technical guidance, all of which have greatly contributed to the successful completion of this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, "Towards the unmanned aerial vehicles (UAVs): A comprehensive review," *Drones*, vol. 6, no. 6, p. 147, 2022, https://doi.org/10.3390/drones6060147.

[2] A. Gupta, T. Afrin, E. Scully, and N. Yodo, "Advances of UAVs toward future transportation: The state-of-the-art, challenges, and opportunities," *Future Transportation*, vol. 1, no. 2, pp. 326–350, 2021, https://doi.org/10.3390/futuretransp1020019.

[3] N. Mohamed, J. Al-Jaroodi and F. Mohammed, "Unmanned aerial vehicles applications in future smart cities," *Technological Forecasting and Social Change*, vol. 153, p. 119293, 2020, https://doi.org/10.1016/j.techfore.2018.05.004.

[4] M. Jones, S. Djahel, and K. Welsh, "Path-planning for unmanned aerial vehicles with environment complexity considerations: A survey," *ACM Computing Surveys*, vol. 55, no. 11, pp. 1–39, 2023, https://doi.org/10.1145/3570723.

[5] H. Mazaheri, S. Goli, and A. Nourollah, "A Survey of 3D Space Path-Planning Methods and Algorithms," *ACM Computing Surveys*, vol. 57, no. 1, pp. 1–32, 2024, https://doi.org/10.1145/3673896.

[6] M. Maboudi, M. Homaei, S. Song, S. Malihi, M. Saadatseresht and M. Gerke, "A Review on Viewpoints and Path Planning for UAV-Based 3-D Reconstruction," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 5026-5048, 2023, https://doi.org/10.1109/JSTARS.2023.3276427.

[7] P. Suanpang and P. Jamjuntr, "Optimizing Autonomous UAV Navigation with D* Algorithm for Sustainable Development," *Sustainability*, vol. 16, no. 17, p. 7867, 2024, https://doi.org/10.3390/su16177867.

[8] T. Ren and H. Jebelli, "Efficient 3D robotic mapping and navigation method in complex construction environments," *Computer-Aided Civil and Infrastructure Engineering*, vol. 2024, no. 1, pp. 1-26, 2024, https://doi.org/10.1111/mice.13353.

[9] C. Min *et al*., "Autonomous Driving in Unstructured Environments: How Far Have We Come?," *arXiv*, 2024, https://doi.org/10.48550/arXiv.2410.07701.

[10] H. S. Yahia and A. S. Mohammed, “Path planning optimization in unmanned aerial vehicles using meta-heuristic algorithms: A systematic review,” *Environmental Monitoring and Assessment*, vol. 195, no. 1, p. 30, 2023, https://doi.org/10.1007/s10661-022-10590-y.

[11] S. Aggarwal and N. Kumar, “Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges,” *Computer Communications*, vol. 149, pp. 270–299, 2020, https://doi.org/10.1016/j.comcom.2019.10.014.

[12] D. Debnath, F. Vanegas, J. Sandino, A. F. Hawary, and F. Gonzalez, “A Review of UAV Path-Planning Algorithms and Obstacle Avoidance Methods for Remote Sensing Applications,” *Remote Sensing*, vol. 16, no. 21, p. 4019, 2024, https://doi.org/10.3390/rs16214019.

[13] 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, 2023, https://doi.org/10.3390/drones7030211.

[14] Y. Guo, H. Liu, X. Fan, and W. Lyu, “Research progress of path planning methods for autonomous underwater vehicle,” *Mathematical Problems in Engineering*, vol. 2021, no. 1, pp. 1-25, 2021, https://doi.org/10.1155/2021/8847863.

[15] L. Yang *et al.*, “Path Planning Technique for Mobile Robots: A Review,” *Machines*, vol. 11, no. 10, p. 980, 2023, https://doi.org/10.3390/machines11100980.

[16] L. Wang, D. Zhu, W. Pang, and Y. Zhang, “A survey of underwater search for multi-target using Multi-AUV: Task allocation, path planning, and formation control,” *Ocean Engineering*, vol. 278, p. 114393, 2023, https://doi.org/10.1016/j.oceaneng.2023.114393.

[17] S. MahmoudZadeh, A. Yazdani, Y. Kalantari, B. Ciftler, F. Aidarus, and M. O. Al Kadri, “Holistic Review of UAV-Centric Situational Awareness: Applications, Limitations, and Algorithmic Challenges,” *Robotics*, vol. 13, no. 8, p. 117, 2024, https://doi.org/10.3390/robotics13080117.

[18] M. Rastegarpanah, M. E. Asif, J. Butt, H. Voos, and A. Rastegarpanah, “Mobile robotics and 3D printing: addressing challenges in path planning and scalability,” *Virtual and Physical Prototyping*, vol. 19, no. 1, p. e2433588, 2024, https://doi.org/10.1080/17452759.2024.2433588.

[19] C. Chronis, G. Anagnostopoulos, E. Politi, G. Dimitrakopoulos and I. Varlamis, "Dynamic Navigation in Unconstrained Environments Using Reinforcement Learning Algorithms," *IEEE Access*, vol. 11, pp. 117984-118001, 2023, https://doi.org/10.1109/ACCESS.2023.3326435.

[20] C. Zammit and E. V. Kampen, “Real-time 3D UAV path planning in dynamic environments with uncertainty,” *Unmanned Systems*, vol. 11, no. 03, pp. 203–219, 2023, https://doi.org/10.1142/S2301385023500073.

[21] B. Yildiz, M. F. Aslan, A. Durdu, and A. Kayabasi, “Consensus-based virtual leader tracking swarm algorithm with GDRRT*-PSO for path-planning of multiple-UAVs,” *Swarm and Evolutionary Computation*, vol. 88, p. 101612, 2024, https://doi.org/10.1016/j.swevo.2024.101612.

[22] G. Airlangga *et al.*, “Adaptive Path Planning for Multi-UAV Systems in Dynamic 3D Environments: A Multi-Objective Framework,” *Designs*, vol. 8, no. 6, p. 136, 2024, https://doi.org/10.3390/designs8060136.

[23] T. Elmokadem and A. V Savkin, “Towards fully autonomous UAVs: A survey,” *Sensors*, vol. 21, no. 18, p. 6223, 2021, https://doi.org/10.3390/s21186223.

[24] N. Wang, X. Li, K. Zhang, J. Wang, and D. Xie, “A survey on path planning for autonomous ground vehicles in unstructured environments,” *Machines*, vol. 12, no. 1, p. 31, 2024, https://doi.org/10.3390/machines12010031.

[25] J. Luo, Y. Tian, and Z. Wang, “Research on Unmanned Aerial Vehicle Path Planning,” *Drones*, vol. 8, no. 2, p. 51, 2024, https://doi.org/10.3390/drones8020051.

[26] C. S. Tan, R. Mohd-Mokhtar and M. R. Arshad, "A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms," *IEEE Access*, vol. 9, pp. 119310-119342, 2021, https://doi.org/10.1109/ACCESS.2021.3108177.

[27] Y. B. Jmaa and D. Duvivier, “A review of path planning algorithms,” *Intelligent Systems Design and Applications*, pp. 119–130, 2023, https://doi.org/10.1007/978-3-031-64850-2_11.

[28] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, “A survey of path planning algorithms for mobile robots,” *Vehicles*, vol. 3, no. 3, pp. 448–468, 2021, https://doi.org/10.3390/vehicles3030027.

[29] Y. L. A, A. Kuushalie, G. Yerukola and A. M. K, "Efficiency Analysis of Conventional and Weighted Grid-Based Pathfinding Algorithms: A Performance Comparative Study," *2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, pp. 1-6, 2024, https://doi.org/10.1109/ICEEICT61591.2024.10718412.

[30] Y. Feng, H. Wang, Y. Zhu, X. Liu, H. Lu, and Q. Liu, “DAWN: Matrix Operation-Optimized Algorithm for Shortest Paths Problem on Unweighted Graphs,” *Proceedings of the 38th ACM International Conference on Supercomputing*, pp. 1–13, 2024, https://doi.org/10.1145/3650200.3656600.

[31] N. Seyedi, “Compact Routing on Planar Graphs,” *University of Waterloo*, 2024, https://uwspace.uwaterloo.ca/items/4d39347e-c2bf-44c5-9e32-fbbb503ed7b6.

[32] J. Wang, Y. Li, R. Li, H. Chen, and K. Chu, “Trajectory planning for UAV navigation in dynamic environments with matrix alignment Dijkstra,” *Soft Computing*, vol. 26, no. 22, pp. 12599–12610, 2022, https://doi.org/10.1007/s00500-022-07224-3.

[33] Y. Yang, K. Zhang, D. Liu and H. Song, "Autonomous UAV Navigation in Dynamic Environments with Double Deep Q-Networks," *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, pp. 1-7, 2020, https://doi.org/10.1109/DASC50938.2020.9256455.

[34] A. Thakkar and R. Lohiya, “A review on machine learning and deep learning perspectives of IDS for IoT: recent updates, security issues, and challenges,” *Archives of Computational Methods in Engineering*, vol. 28, no. 4, pp. 3211–3243, 2021, https://doi.org/10.1007/s11831-020-09496-0.

[35] N. Fatima, P. Saxena, and M. Gupta, “Integration of multi access edge computing with unmanned aerial vehicles: Current techniques, open issues and research directions,” *Physical Communication*, vol. 52, p. 101641, 2022, https://doi.org/10.1016/j.phycom.2022.101641.

[36] Y. Liu, H.-N. Dai, Q. Wang, M. K. Shukla, and M. Imran, “Unmanned aerial vehicle for internet of everything: Opportunities and challenges,” *Computer Communications*, vol. 155, pp. 66–83, 2020, https://doi.org/10.1016/j.comcom.2020.03.017.

[37] S. Julius Fusic and R. Sitharthan, “Improved RRT* Algorithm-Based Path Planning for Unmanned Aerial Vehicle in a 3D Metropolitan Environment,” *Unmanned Systems*, vol. 12, no. 05, pp. 859-875, 2024, https://doi.org/10.1142/S2301385024500225.

[38] R. Zhang, H. Guo, D. Andriukaitis, Y. Li, G. Królczyk, and Z. Li, “Intelligent path planning by an improved RRT algorithm with dual grid map,” *Alexandria Engineering Journal*, vol. 88, pp. 91–104, 2024, https://doi.org/10.1016/j.aej.2023.12.044.

[39] K. Solovey, L. Janson, E. Schmerling, E. Frazzoli and M. Pavone, "Revisiting the Asymptotic Optimality of RRT," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2189-2195, 2020, https://doi.org/10.1109/ICRA40945.2020.9196553.

[40] G. N. P. Pratama *et al*., “Benchmark Analysis of Sampling Methods for RRT Path Planning,” *Control Systems and Optimization Letters*, vol. 2, no. 2, pp. 217–233, 2024, https://doi.org/10.59247/csol.v2i2.132.

[41] N. Wen, R. Zhang, J. Wu, and G. Liu, “Online planning for relative optimal and safe paths for USVs using a dual sampling domain reduction-based RRT* method,” *International Journal of Machine Learning and Cybernetics*, vol. 11, pp. 2665–2687, 2020, https://doi.org/10.1007/s13042-020-01144-0.

[42] Y. Chen and L. Wang, "Adaptively Dynamic RRT*-Connect: Path Planning for UAVs Against Dynamic Obstacles," *2022 7th International Conference on Automation, Control and Robotics Engineering (CACRE)*, pp. 1-7, 2022, https://doi.org/10.1109/CACRE54574.2022.9834188.

[43] L. Jiang, S. Liu, Y. Cui and H. Jiang, "Path Planning for Robotic Manipulator in Complex Multi-Obstacle Environment Based on Improved_RRT," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 4774-4785, 2022, https://doi.org/10.1109/TMECH.2022.3165845.

[44] C. Tonola, “Reactive Motion Replanning For Human-Robot Collaboration,” *IRIS Institutional Research Information System*, 2024, https://iris.unibs.it/handle/11379/596672.

[45] M. Zulfiqar, K. A. A. Gamage, M. B. Rasheed, and C. Gould, “Optimised Deep Learning for Time-Critical Load Forecasting Using LSTM and Modified Particle Swarm Optimisation,” *Energies*, vol. 17, no. 22, p. 5524, 2024, https://doi.org/10.3390/en17225524.

[46] M. Moltafet, M. Leinonen, M. Codreanu and N. Pappas, "Power Minimization for Age of Information Constrained Dynamic Control in Wireless Sensor Networks," *IEEE Transactions on Communications*, vol. 70, no. 1, pp. 419-432, 2022, https://doi.org/10.1109/TCOMM.2021.3124949.

[47] F. Gul, W. Rahiman, S. S. N. Alhady, A. Ali, I. Mir, and A. Jalil, “Meta-heuristic approach for solving multi-objective path planning for autonomous guided robot using PSO--GWO optimization algorithm with evolutionary programming,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 7873–7890, 2021, https://doi.org/10.1007/s12652-020-02514-w.

[48] X. Guo, M. Ji, Z. Zhao, D. Wen, and W. Zhang, “Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (PSO) algorithm,” *Ocean Engineering*, vol. 216, p. 107693, 2020, https://doi.org/10.1016/j.oceaneng.2020.107693.

[49] F. H. Ajeil, I. K. Ibraheem, M. A. Sahib, and A. J. Humaidi, “Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm,” *Applied Soft Computing*, vol. 89, p. 106076, 2020, https://doi.org/10.1016/j.asoc.2020.106076.

[50] M. S. Alkoffash, M. A. Awadallah, M. Alweshah, R. A. Zitar, K. Assaleh, and M. A. Al-Betar, “A non-convex economic load dispatch using hybrid salp swarm algorithm,” *Arabian Journal for Science and Engineering*, vol. 46, no. 9, pp. 8721–8740, 2021, https://doi.org/10.1007/s13369-021-05646-z.

[51] M. A. Al-Betar, “Island-based harmony search algorithm for non-convex economic load dispatch problems,” *Journal of Electrical Engineering & Technology*, vol. 16, no. 4, pp. 1985–2015, 2021, https://doi.org/10.1007/s42835-021-00758-w.

[52] M. S. Braik, M. A. Awadallah, M. A. Al-Betar, A. I. Hammouri, and R. A. Zitar, “A non-convex economic load dispatch problem using chameleon swarm algorithm with roulette wheel and levy flight methods,” *Applied Intelligence*, vol. 53, no. 14, pp. 17508–17547, 2023, https://doi.org/10.1007/s10489-022-04363-w.

[53] V. Harkare, R. Mangrulkar, O. Thorat, and S. R. Jain, “Evolutionary Approaches for Multi-objective Optimization and Pareto-Optimal Solution Selection in Data Analytics,” *Applied Multi-objective Optimization*, pp. 67–94, 2024, https://doi.org/10.1007/978-981-97-0353-1_4.

[54] P. Li and Q. Liu, “Optimizing energy management strategies for microgrids through chaotic local search and particle swarm optimization techniques,” *Heliyon*, vol. 10, no. 17, p. e36669, 2024, https://doi.org/10.1016/j.heliyon.2024.e36669.

[55] G. Papazoglou, P. Biskas, “Review and Comparison of Genetic Algorithm and Particle Swarm Optimization in the Optimal Power Flow Problem,” *Energies*, vol. 16, no. 3, p. 1152, 2023, https://doi.org/10.3390/en16031152.

[56] Z. Cao, L. Kooistra, W. Wang, L. Guo, and J. Valente, “Real-time object detection based on uav remote sensing: A systematic literature review,” *Drones*, vol. 7, no. 10, p. 620, 2023, https://doi.org/10.3390/drones7100620.

[57] K. Telli *et al*., “A comprehensive review of recent research trends on unmanned aerial vehicles (uavs),” *Systems*, vol. 11, no. 8, p. 400, 2023, https://doi.org/10.3390/systems11080400.

[58] N. Cheng *et al*., "AI for UAV-Assisted IoT Applications: A Comprehensive Review," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14438-14461, 2023, https://doi.org/10.1109/JIOT.2023.3268316.

[59] L. Zhai, C. Liu, X. Zhang and C. Wang, "Local Trajectory Planning for Obstacle Avoidance of Unmanned Tracked Vehicles Based on Artificial Potential Field Method," *IEEE Access*, vol. 12, pp. 19665-19681, 2024, https://doi.org/10.1109/ACCESS.2024.3355952.

[60] H. Sang, Y. You, X. Sun, Y. Zhou, and F. Liu, “The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations,” *Ocean Engineering*, vol. 223, p. 108709, 2021, https://doi.org/10.1016/j.oceaneng.2021.108709.

[61] B. Lu *et al.*, "Adaptive Potential Field-Based Path Planning for Complex Autonomous Driving Scenarios," *IEEE Access*, vol. 8, pp. 225294-225305, 2020, https://doi.org/10.1109/ACCESS.2020.3044909.

[62] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, “Path planning techniques for mobile robots: Review and prospect,” *Expert Systems with Applications*, vol. 227, p. 120254, 2023, https://doi.org/10.1016/j.eswa.2023.120254.

[63] K. Yang and L. Liu, "An Improved Deep Reinforcement Learning Algorithm for Path Planning in Unmanned Driving," *IEEE Access*, vol. 12, pp. 67935-67944, 2024, https://doi.org/10.1109/ACCESS.2024.3400159.

[64] Y. Bai, H. Zhao, X. Zhang, Z. Chang, R. Jäntti and K. Yang, "Toward Autonomous Multi-UAV Wireless Network: A Survey of Reinforcement Learning-Based Approaches," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 3038-3067, 2023, https://doi.org/10.1109/COMST.2023.3323344.

[65] O. Bouhamed, H. Ghazzai, H. Besbes and Y. Massoud, "Autonomous UAV Navigation: A DDPG-Based Deep Reinforcement Learning Approach," *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2020, https://doi.org/10.1109/ISCAS45731.2020.9181245.

[66] C. Wang, J. Wang, J. Wang and X. Zhang, "Deep-Reinforcement-Learning-Based Autonomous UAV Navigation With Sparse Rewards," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6180-6190, 2020, https://doi.org/10.1109/JIOT.2020.2973193.

[67] T. T. Nguyen, N. D. Nguyen and S. Nahavandi, "Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826-3839, 2020, https://doi.org/10.1109/TCYB.2020.2977374.

[68] C. Vignon, J. Rabault, and R. Vinuesa, “Recent advances in applying deep reinforcement learning for flow control: Perspectives and future directions,” *Physics of Fluids*, vol. 35, no. 3, p. 031301, 2023, https://doi.org/10.1063/5.0143913.

[69] X. Wang and Y. C. Wu, “Empowering legal justice with AI: A reinforcement learning SAC-VAE framework for advanced legal text summarization,” *PLoS One*, vol. 19, no. 10, p. e0312623, 2024, https://doi.org/10.1371/journal.pone.0312623.

[70] A. Delarue, R. Anderson, and C. Tjandraatmadja, “Reinforcement learning with combinatorial actions: An application to vehicle routing,” *Proceedings of the 34th International Conference on Neural Information Processing Systems*, vol. 52, pp. 609–620, 2020, https://dl.acm.org/doi/10.5555/3495724.3495776.