IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 6 July 2023, accepted 31 July 2023, date of publication 7 August 2023, date of current version 11 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3303109

RESEARCH ARTICLE

Collision-Free Trajectory Following With Augmented Artificial Potential Field Using UAVs

JURICA GORICANEC, (Graduate Student Member, IEEE), ANA MILAS, (Graduate Student Member, IEEE), LOVRO MARKOVIC, (Member, IEEE), AND STJEPAN BOGDAN, (Senior Member, IEEE)
Faculty of Electrical Engineering and Computing, University of Zagreb, 10000 Zagreb, Croatia

Corresponding author: Jurica Goricanec (jurica.goricanec@fer.hr)

This work was supported in part by the Project VIRTUALUAV-Development of a System of Unmanned Aerial Vehicles (UAVs) Controlled in Virtual Environments funded by the European Regional Development Fund under Grant KK.01.2.1.02.0197, and in part by the Research was a Part of the Scientific Project Strengthening Research and Innovation Excellence in Autonomous Aerial Systems-AeroSTREAM [1] through the European Commission HORIZON-WIDERA-2021-ACCESS-05 Program through Project under Grant 101071270. The work of Ana Milas was supported in part by the ''Young Researchers' Career Development Project–Training of Doctoral Students'' of the Croatian Science Foundation funded by the European Union from the European Social Fund.

ABSTRACT In this paper we address the problem of trajectory following in an unknown environment with an unmanned aerial vehicle (UAV). The main goal is to safely follow the planned trajectory by avoiding obstacles. The proposed approach is suitable for aerial vehicles equipped with 2D or 3D sensors, such as LiDARs. We present a novel algorithm based on the conventional Artificial Potential Field (APF) called Augmented Artificial Potential Field (AAPF) that corrects the planned path to avoid obstacles. Our proposed algorithm uses a combination of two attractive forces and both normal and rotational repulsive forces to avoid obstacles and handle local minima problems. The smooth trajectory following achieved with the MPC tracker allows us to quickly change and re-plan the UAV path. Comparative simulation experiments have shown that our approach solves local minima problems in trajectory following and generates more efficient paths to avoid potential collisions with static obstacles compared to our previously developed algorithm for obstacle avoidance. The laboratory experimental evaluation results indicate that the algorithm can be deployed on a real UAV with limited computational power and real-time processing requirements.

INDEX TERMS Artificial potential fields, obstacle avoidance, UAV, trajectory following, path planning.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) have been recently utilized in various applications, such as agriculture [2], wind turbine inspection [3], inspection of civil infrastructure [4], search and rescue [5], autonomous exploration [6], [7], etc. To successfully accomplish their missions, UAVs need to execute safe and collision-free paths. Usually, the information about the obstacle is not known beforehand and should be detected by the sensors attached to the UAV during the mission. The UAV must be able to avoid the detected obstacle using the information collected online. In other words, obstacle detection and avoidance is one of the most important capabilities for successful UAV application execution.

There are a variety of algorithms to achieve safe path planning [8]. One of the most attractive methods is the Artificial Potential Field (APF) method [9]. The principle by which the APF method navigates the UAV through the environment is the generation of a virtual force vector that dictates the direction of UAV motion. This resulting virtual force vector is the sum of all attractive and repulsive forces acting on the UAV. Attractive forces pull the UAV towards the unknown environment, while repulsive forces push it away

The associate editor coordinating the review of this manuscript and approving it for publication was Cesar Briso.

from obstacles. The APF method offers efficient and effective way for robot obstacle avoidance.

However, the APF method has several major limitations [10], the most common of which is local minima. To overcome the local minima problem and minimize deviations from the originally planned trajectory, we propose Augmented Artificial Potential Field (AAPF) algorithm that follows the planned path while making corrections to it near obstacles. The repulsive potential field is composed of the normal and rotational components of the repulsive force. We also extended the Modified Artificial Potential Field (MAPF) approach described in [11] by adding two attractive forces instead of setting the attractive forces to zero. The attractive forces are based on obstacle points and a goal point to minimize overshoot when avoiding obstacles. To maximize the part where the UAV follows the originally planned trajectory, we extended the algorithm to determine the waypoint on the path closest to the obstacle that the UAV can safely reach after the collision avoidance maneuver. The algorithm results in a fully autonomous trajectory following method with obstacle avoidance.

The contributions of this paper are summarized as follows:

- The augmented potential field algorithm called AAPF based on the normal and rotational repulsive force which solves the problem of local minima.
- Generation of the two attractive forces based on obstacle points and a goal point in order to minimize overshoot while avoiding obstacles.
- Determination of feasibility of the trajectory points based on the generated potential field.
- Comprehensive analysis and validation of the proposed approach in simulation and real-world environments.

In Section II we give an overview of the state-of-the-art in obstacle avoidance methods and position our work in relation to them. Section III and Section IV are the core of the paper and contain a detailed explanation of the proposed method. The results of the simulations performed with a UAV and their analysis are presented in Section V while the real-world experiment analysis is proposed in Section VI. The paper ends with a conclusion in Section VII.

II. STATE OF THE ART

In recent years, considerable research effort has been focused on trajectory planning algorithms utilizing different optimization methods. For example, a path-guided optimization approach was proposed in [12] to develop a robust and perception-aware replanning framework. Similarly, a kinodynamic path searching method based on B-spline optimization was utilized in [13] to navigate the UAV in unknown spaces. In [14], high-speed trajectories for the UAV were obtained by enabling the local planner to optimize in both free and unknown space, generating a feasible backup trajectory in free space if the one in unknown space is obstructed by an obstacle. Although these methods provide fast and safe trajectories for the UAV in unknown spaces, they are primarily focused on reaching a target point rather than following a predefined path which is important in missions where the objective is to collect relevant data along the initially planned trajectory.

The APF method is widely used in path planning as well as real-time collision avoidance. The method is based on the concept of a repulsive and attractive force field that repels a robot from an obstacle and attracts it toward the target [9]. Due to the simple mathematical background and low computational complexity of the algorithm, it is suitable for implementation in mobile robots [15], underwater [16], and aerial autonomous vehicles [17], as well as for controlling swarm formations [18].

Even though the basic approach to obstacle avoidance developed in [9] is efficient and simple, the path plan may not be globally optimal. Several major limitations of the APF method are cited in the literature - local minima, oscillations in a narrow corridor, and unreachable targets [19], [20], [21]. These problems arise when the sum of the attractive and repulsive force vectors becomes zero before the robot reaches the target. To solve these problems, a number of improved approaches based on the APF method have been applied to the path planning of aerial robots. Firstly, integrating other algorithms into the APF method, such as Rapidly-exploring Random Tree [22], [23], Particle Swarm Optimization [24], etc. Secondly, modifying the original APF method, e.g., by extending the repulsion formula to include more terms [25] or by using a different form of the potential function, as described in [26] and [27].

In [28], the authors enumerate the problems of the traditional potential field method and focus on problems with unreachable targets. The Modified Potential Field Method (MPFM) compensates for the repulsive force by adding the Euclidean distance that connects the attractive force to the repulsive force. The authors point out that the potential field method has several inherent limitations among which the non-reachable target problem is the most challenging. Zhu et al. [29] proposed a Modified Artificial Potential Field algorithm, which is able to decompose the total force and estimate the physical barriers on the 3D environment.

Another approach of work on the APF method utilizes a global path planning method to find a desired path to the goal and uses attractive and repulsive concepts of the APF methods for local path planning [30], [31], [32]. Jaillet et al. [32] used a user-defined cost-map to influence node placement in an RRT algorithm. The cost-map specifies a repulsion or attraction factor for each region. Similarly, in [31] Navigation Fields assign a gradient that a robot follows. In the same manner, Scherer et al. [33] combine global and local potential field-based planners to navigate an UAV toward the goal. The global planner is based on the implementation of a Laplace equation that generates a potential function with an unique minimum at the target, while the local planner uses a modification of the conventional potential field method in which the relative angles between the goal and the obstacles are taken into account.

Apart from static obstacles, there is a large number of papers dealing with collision avoidance against a moving obstacle using the APF method. Chen and Zhang [34] present path planning based on the APF method for collision avoidance in a dynamic environment with faster response and high accuracy. The APF method applied in a dynamic environment is also presented in [10] and [25] and is out of the scope of this paper. Furthermore, the APF method can be used not only for the single robot path planning, but also for multi-robot systems, as described in [18], [35], [36], and [37].

However, the main drawbacks of the APF method are oscillations, local minima and unreachable targets. To overcome the problem of local minima, the concept of the curl-free vector field is studied in [10] and [38]. In these papers, the curl-free vector field is utilized instead of the repulsive potential field, although the attractive potential field is the same as the conventional one. Additionally, in [10], the authors provide a solution to the problem of the unreachable target in the conventional APF method.

Motivated by advances in literature and the fact that the APF method has been shown to perform well in obstacle avoidance but typically gets stuck in oscillations and local minima, this paper presents a novel algorithm based on potential fields to enable efficient path planning in a 3D environment. Inspired by the idea in [10], we modified the APF method to resolve oscillations and local minima and focus on obstacle avoidance in a static environment with a single UAV. We have extended the approach from [11] adding attractive potential field forces and introducing the approach for returning to the originally planned path after avoiding obstacles. Similarly to the approach in [30], we use a global trajectory planning method to find a path from the UAV position to the target and then activate the APF method for local path planning.

III. PROPOSED APPROACH

A. PROBLEM FORMULATION

The main goal of our approach is to autonomously and safely plan a trajectory in a 3D environment using a UAV equipped with a sensory set that generates large point clouds, such as LiDARs. LiDARs provide point clouds in a large Field of View (FOV), which is crucial in obstacle avoidance. Autonomous navigation is performed using a UAV that has uncertain prior knowledge of the environment. This means that there is a possibility that the UAV will encounter obstacles on its flight path that are unknown at the time of initial trajectory planning. In this work, we only consider static unknown obstacles in the environment. An example of such a scenario is an infrastructure inspection mission where the layout of the surveyed structure is known, but information about the surrounding environment may be unreliable or unknown. In order to gather relevant data about the surveyed structure, such as photos or laser scans, the initial flight trajectory is planned without considering possible obstacles like trees or other structures in the area. The main objective of the UAV is to follow the originally planned flight trajectory with minimal deviations while avoiding obstacles and reaching the final goal point. Deviations from the planned trajectory are allowed in order to circumvent any obstacles that were previously unknown. However, in contrast to the approach from [11], in this paper, we focus on both obstacle avoidance and optimal trajectory execution. In other words, the deviations from the originally planned flight trajectory are minimized. For instance, if the UAV avoids large obstacles, it can still return to the planned trajectory at the nearest point to the obstacle that is determined safe by the algorithm and continue following its originally planned trajectory.

B. UAV AND SENSOR MODELS

The UAV is represented with a state vector $\mathbf{x} = \begin{bmatrix}\mathbf{q}^T & \psi\end{bmatrix}^T \in \mathbb{R}^4$ that consists of the position $\mathbf{q} = \begin{bmatrix}x & y & z\end{bmatrix}^T \in \mathbb{R}^3$ and the yaw rotation angle around z axis $\psi \in [-\pi, \pi)$. Furthermore, the algorithm assumes a maximum linear velocity $\mathbf{v}_{max} \in \mathbb{R}^3$, a maximum angular velocity around z axis $\dot{\psi}_{max}$, a maximum linear acceleration $\mathbf{a}_{max} \in \mathbb{R}^3$ and maximum angular acceleration around z axis $\ddot{\psi}_{max}$. The algorithm relies on a maximum range of the sensor $R_{max} \in \mathbb{R}$ with horizontal and vertical FOV in range, $\alpha_h, \alpha_v \in (0°, 360°]$, respectively. This allows our algorithm to work with point-cloud-producing sensors with various FOV, such as cameras with limited FOV and LiDARs with limited α_v.

C. GLOBAL TRAJECTORY FOLLOWING

The proposed system consists of a global and a local module. The global part of the system generates the trajectory by passing waypoints to the trajectory planner, while the local planner utilize the AAPF method for the planned trajectory corrections in order to avoid obstacles. Within this paper, we use the Time Optimal Path Parametrization by Reachability Analysis (TOPP-RA) algorithm for planning a trajectory, which is developed in [39]. Apart from the waypoints, inputs for the TOPP-RA are also velocity and acceleration constraints, which are set taking into account the UAV physical limitations.

The UAV simultaneously executes the generated trajectory and processes the data received from the sensor system. The proposed potential field algorithm calculates the total repulsive force F_r, which is described in detail in the next section. Let us define $K^{UAV}_{threshold}$ as a constant that regulates whether the algorithm follows the trajectory point or activates the potential field algorithm. The condition is described as follows:

$$\text{Action} = \begin{cases} \text{Follow Trajectory} & \text{if } F_r < K^{UAV}_{threshold}, \\ \text{Do APF Method} & \text{if } F_r \geq K^{UAV}_{threshold}. \end{cases}$$

An overview of the proposed system is given in Fig. 1. When the AAPF method is not active, the global trajectory points are passed directly to the MPC tracker module. On the other hand, if the AAPF method is active, the modified path is generated and forwarded to the MPC tracker module.

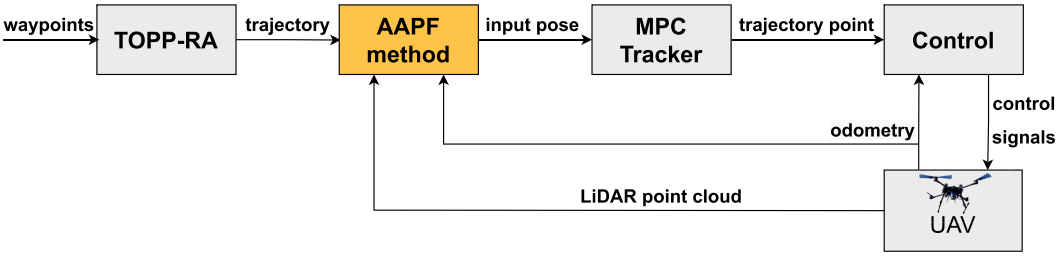


FIGURE 1. Overall schematic diagram of the trajectory planning and obstacle avoidance system. The trajectory generated by TOPP-RA, the LiDAR point cloud, and the odometry data represent inputs to the AAPF method. The MPC tracker module generates a trajectory point to which the UAV navigates.

A standard PID cascade is used to control the UAV, with the inner loop controlling the velocity and the outer loop controlling the position. In our case, the reference for the controller is a trajectory point.

D. LOCAL PATH PLANNING USING A POTENTIAL FIELD-BASED ALGORITHM

Local path planning integrates the potential field algorithm to correct the planned global trajectory and generate a safe and collision-free path. The conventional AAPF is composed of two types of potential fields: attractive potential field and repulsive potential field [9]. Attractive potential field $U_a(\cdot)$ is usually formed by goal location $\mathbf{q}_g$ so that the goal point attracts the UAV in the field. In contrast, repulsive potential field $U_r(\cdot)$ is generated by the position of the obstacle $\mathbf{q}_o$ so that obstacles repel the UAV when it moves within a certain range (d_0) around them. The sum of these two potential fields results in a total potential field $U_t(\cdot)$ that directs the UAV toward the goal point while avoiding obstacles:

$$U_t(\mathbf{q}) = U_a(\mathbf{q}) + U_r(\mathbf{q}). \tag{1}$$

Similarly, our approach consists of attractive and repulsive potential fields. The attractive potential field is generated from obstacle vertex and a goal point to minimize overshoot when avoiding obstacles. The repulsive field is composed of a normal and a rotational components of the repulsive force, which enables the solution of the local minima problem. The UAV follows the planned trajectory and the total potential force corrects it to avoid collision with the obstacle. We also extended the algorithm to determine the waypoint on the originally planned trajectory closest to the obstacle that the UAV can safely reach after the collision avoidance maneuver. Parts of the local module are described in detail in Section IV.

E. MPC TRACKER

An MPC-based tracking method is chosen to generate UAV trajectory points along the corrected collision-free path obtained from the potential fields. The original implementation is presented in [40] while an adapted version of their work is used in this paper. The main motivation for using this tracking method is that it allows us to quickly change and re-plan the UAV trajectory based on the current system state and model dynamics. Furthermore, the tracker enables safe and stable flight, regardless of the goal point resulting from the potential fields.

This tracking method employs a model predictive controller with a constant snap UAV model which controls a virtual UAV using snap commands. Snap commands are used as the input to the linear system, which predicts the next virtual UAV state based on the given model dynamics. The complete state of the virtual UAV is then sampled and used as a referent trajectory point for the real UAV at a rate of $100Hz$.

The definition of the MPC problem is as follows:

$$\begin{aligned} \min_{u_0,\ldots,u_N} & \sum_{i=0}^{N} \left(e_i^T Q e_i + u_i^T P u_i \right) \\ \text{s.t. } & x_{k+1} = A x_k + B u_k \,, \\ & x_k \le x_{max} \,, \\ & u_k \le u_{max} \,, \end{aligned} \tag{2}$$

where N is the horizon length. The error between the predicted virtual UAV state and the reference at the k-th horizon is defined as $e_k = x_k - r_k$. The state and input constraints are denoted by x_{max} and u_{max}, respectively. The matrices A and B represent the well-known constant snap virtual UAV model. The weights Q and P are tuned for smooth trajectory generation wrt the velocity and acceleration.

The CVXGEN solver is used to obtain the optimal snap input u_0^*, which is used as an input to predict the next virtual UAV state and calculate the next referent trajectory point.

IV. AUGMENTED ARTIFICIAL POTENTIAL FIELD ALGORITHM

The AAPF method consists of attractive and repulsive potential fields, which correct the initially planned path and lead to a safe and collision-free path. The repulsive potential field is based on the approach from [11], while the attractive potential field is the novelty of the paper. We also introduce the approach for returning to the originally planned path after avoiding obstacles.

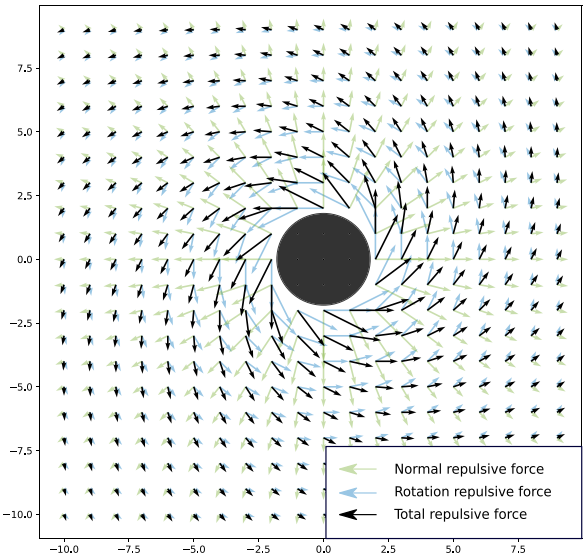


FIGURE 2. Repulsive force generated by the obstacle. The total repulsive force consists of the normal component vector F_{rn} and the rotational component vector F_{rr}, where the rotational potential field is generated in the counterclockwise direction.

A. REPULSIVE POTENTIAL FIELD

The repulsive field U_r is given by [9]:

$$U_r(\mathbf{q}) = \begin{cases} \frac{1}{2}k_{rt}(\frac{1}{\|\mathbf{q}_o - \mathbf{q}\|} - \frac{1}{d_0})^2 & \text{if } \|\mathbf{q}_o - \mathbf{q}\| \leq d_o, \\ 0 & \text{if } \|\mathbf{q}_o - \mathbf{q}\| > d_o, \end{cases} \tag{3}$$

where k_r is repulsive gain coefficient, $\|\mathbf{q}_o - \mathbf{q}\|$ is the relative distance between the position of the UAV $\mathbf{q}$ and the obstacle $\mathbf{q}_o$ and d_0 is the limiting distance of the potential field influence. The corresponding *normal repulsive potential field force*, is derived by computing the negative gradient of the repulsive potential function as follows:

$$\begin{aligned} \mathbf{F}_{rn}(\mathbf{q}) &= -\nabla U_r(\mathbf{q}) \\ &= \begin{cases} k_{rn}(\frac{1}{\|\mathbf{q}_o - \mathbf{q}\|} - \frac{1}{d_0})\frac{1}{\|\mathbf{q}_o - \mathbf{q}\|^3}(\mathbf{q} - \mathbf{q}_o) \\ \qquad \text{if } \|\mathbf{q}_o - \mathbf{q}\| \leq d_o, \\ 0 \qquad \text{if } \|\mathbf{q}_o - \mathbf{q}\| > d_o, \end{cases} \end{aligned} \tag{4}$$

where k_{rn} is the gain of the normal force vector. Since the repulsive potential field is produced for each obstacle, the total potential field for n obstacles is expressed as the sum of the repulsive forces for n obstacles:

$$\mathbf{F}_t = \sum_{1}^{n} \mathbf{F}_{rn_i}. \tag{5}$$

Although the conventional APF method generates an effective path, it encounters several problems, such as oscillations in the presence of obstacles or in narrow passages and getting stuck in local minima [10]. Among the above drawbacks, the problem of local minima is considered in this paper as it leads to an incomplete path [11].

The local minima problem occurs when the total force caused by the potential field becomes zero or small enough that it does not generate any significant displacement of the UAV before it reaches the goal position. This means that the magnitude of the attractive and repulsive potential field forces are equal and their directions are opposite. Conventional APF methods rely on local information, such as the distance and direction to obstacles, to navigate a robot or plan a path. This local information may lead the robot towards a local minimum in the potential field, resulting in suboptimal or inefficient paths. Furthermore, conventional APF methods can be sensitive to the initial conditions and the starting position of the robot. If the robot starts close to a local minimum, it is more likely to converge to that local minimum and struggle to escape from it. To solve those issues and overcome a local minima problem, we propose an extension of the repulsive force $\mathbf{F}_r$ to include both the normal and rotational components of the repulsive force. The proposed repulsive forces ensure the successful navigation of the UAV through challenging environments, avoiding potential collisions, and escaping local minima. The modified repulsive force is formulated as follows:

$$\mathbf{F}_r(\mathbf{q}) = \mathbf{F}_{rn}(\mathbf{q}) + \mathbf{F}_{rr}(\mathbf{q}), \tag{6}$$

where $\mathbf{F}_{rn}(\mathbf{q})$ is the normal part of the repulsive force defined with Eq. 4, while $\mathbf{F}_{rr}(\mathbf{q})$ is the *rotational repulsive potential field force* shown in Fig. 2. The rotational component of the repulsive force is calculated only in the X-Y plane, so we can write position vector of the UAV $\mathbf{q} = \begin{bmatrix}\mathbf{r} & z\end{bmatrix}^T \in \mathbb{R}^3$, position vector of the obstacle $\mathbf{q}_o = \begin{bmatrix}\mathbf{r}_o & z_o\end{bmatrix}^T \in \mathbb{R}^3$, and defined so that the curl of the potential field $U_{rr}(\mathbf{q})$ equals zero:

$$\nabla \times U_{rr}(\mathbf{q}) = 0 \tag{7}$$

$$\mathbf{F}_{rr}(\mathbf{q}) = \begin{cases} k_{rr}(\frac{1}{\|\mathbf{q}_o - \mathbf{q}\|} - \frac{1}{d_0}) \\ \cdot \frac{1}{\|\mathbf{q}_o - \mathbf{q}\|^3} R(\mathbf{r} - \mathbf{r}_o) & \text{if } \|\mathbf{q}_o - \mathbf{q}\| \leq d_o, \\ 0 & \text{if } \|\mathbf{q}_o - \mathbf{q}\| > d_o, \end{cases} \tag{8}$$

where k_{rr} is the gain of the rotational force vector and $\mathbf{R}$ is the rotation matrix. The definition of the matrix $\mathbf{R}$ depends on the direction of the generated rotating repulsive field, which can be clockwise or counterclockwise. The angle ϕ is defined as the angle of the UAV on the trajectory and ρ is the angle of the vector from the position of the UAV to the centroid C_i of the obstacle in the environment (Fig. 3). The angle θ is defined as the difference between these two angles:

$$\theta = \phi - \rho \tag{9}$$

and its sign determines the direction of the potential field around the obstacle. The definition of the matrix R with

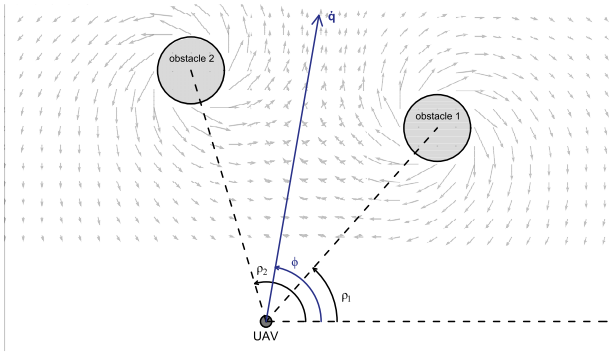


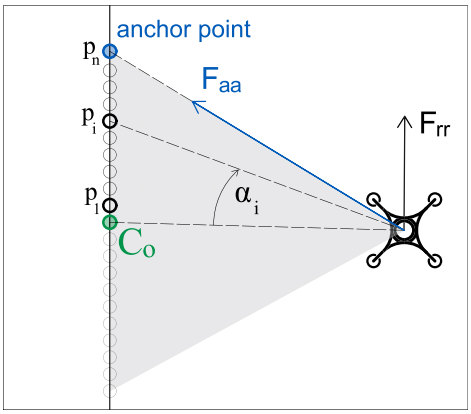
FIGURE 3. Direction of the rotational potential field generated by the obstacle i in the environment is defined by the difference between the angle of the trajectory ϕ and the angle ρ_i, which is an angle of a vector from the UAV to the centroid of the obstacle i.

respect to the θ is given by [10]:

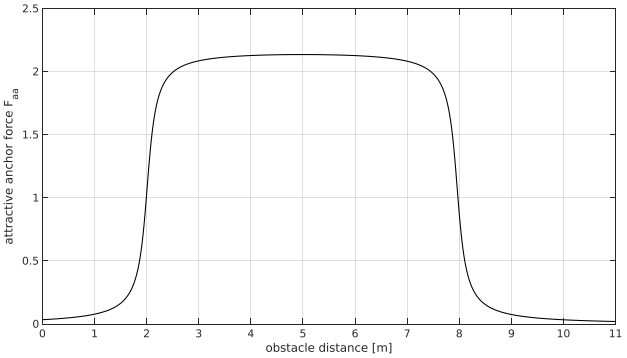
$$\mathbf{R} = \begin{cases} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} & \text{if } \theta \geq 0, \\ \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} & \text{if } \theta < 0. \end{cases} \tag{10}$$

To find the centroid C_i of each obstacle, we use a clustering method and the computation of a 3D centroid from the PCL library [41]. A clustering method is used to divide a disorganized point cloud model of the environment into smaller parts that represent obstacles. The simplest way is to use the Euclidean clustering algorithm, which is a greedy growing region algorithm based on the nearest neighbor. The cluster affinity is based on the distance to each point of a cluster and is defined by the parameter called cluster tolerance $c_{tolerance}$. If the Euclidean distance between the point p_n and any point p_i that is a member of the cluster S_i is below a certain threshold value $c_{tolerance}$, then the point p_n is included in cluster S_i. If that is not the case and the distance exceeds this threshold, point p_n is assigned to a different cluster S_j. The choice of the $c_{tolerance}$ parameter is based on the dimensions of the UAV augmented by a safety margin. This ensures that the UAV can navigate between any two points of separate clusters. If the $c_{tolerance}$ value is set too low, an obstacle that should be considered as a single entity may be perceived as multiple clusters. For instance, a narrow gap in a wall that is too small for the UAV to pass through might be observed as two separate obstacles. On the other hand, if the chosen value is excessively high, multiple obstacles may be treated as a single cluster, thereby neglecting potential pathways between them. The 3D centroid C_i is then computed for each cluster of the given point cloud.

The proposed combination of forces ensures that the UAV always moves away from the obstacle and in the direction of the planned trajectory. This prevents the UAV from getting stuck in an oscillatory motion around the local minimum



(a) Set of detected LiDAR points $P = p_1, ..., p_n$ are selected as candidates for the *anchor point* based on the direction of the rotational repulsive force and position of the obstacle centroid C_o. Angle a_i is calculated for each point p_i and the *anchor point* is then chosen from the set P such that it has the highest value of angle.



(b) Anchor attractive force generated with a bell-shaped function and parameters $k_{aa} = 0.7$, $b_1 = b_2 = 7.5$, $k_1 = 4.8$, $k_2 = 19.0$. The center of the function is at the $5m$ distance from the obstacle.

FIGURE 4. Selection process of the *anchor point* (a) and example of the bell function used to calculate *attractive anchor force* $\mathbf{F}_{aa}$ (b).

created by the normal forces or from orbiting the obstacle at the same distance as caused by the rotational forces.

B. ATTRACTIVE POTENTIAL FIELD

In our previous work [11], we showed that the MAPF algorithm successfully avoids obstacles even in a complex environment. The disadvantage of the MAPF algorithm is that it often generates unnecessarily large deviations from the obstacles. As the attractive forces are set to zero, the repulsive forces drives the UAV to the edge of the field. To minimize the generated deviations when avoiding obstacles, we introduced two attractive forces based on the detected obstacle points and the goal point.

The first attractive force (*attractive anchor force* $\mathbf{F}_{aa}$) was inspired by a mechanical mass-spring system, where the spring determines the maximum deviation of the object from the anchor point where the spring is connected to the environment. In the AAPF algorithm, the anchor point is the

vertex of each detected obstacle (Fig. 4.a). For each obstacle represented by the cluster of detected LiDAR points, the anchor point is calculated separately. Based on the direction of the rotational potential filed generated around the obstacle, a portion of the point cloud cluster is selected in which the angle α_i is calculated for each LiDAR point p_i. The angle α_i is defined as the angle between the vector from the UAV to the centroid of the obstacle and the vector from the UAV to the point p_i. The anchor point is determined by the maximum angle α_i. Since the $\mathbf{F}_{aa}$ attracts the UAV towards the obstacle, the chosen function to generate the force is bell-shaped (Fig. 4.b). The reason for this is that the force needs to have a maximum magnitude at the preferred distance from the obstacle where we want to fly around it, and a minimum magnitude when the UAV is very close or very far from the obstacle. The function for generating the anchor attractive force field is given by:

$$\mathbf{F}_{aa}(\mathbf{q}) = k_{aa}(\arctan(b_1\|\mathbf{q}_a - \mathbf{q}\| - k_1\pi) - \arctan(b_2\|\mathbf{q}_a - \mathbf{q}\| - k_2\pi))(\mathbf{q}_a - \mathbf{q}). \tag{11}$$

The parameter k_{aa} determines the magnitude of the force $\mathbf{F}_{aa}$, and the parameters b_1, k_1 and b_2, k_2 define the rising and falling edges of the function, respectively.

The second attractive force active during the execution of the AAPF algorithm is *attractive goal force* $\mathbf{F}_{ag}$. The direction of the $\mathbf{F}_{ag}$ force is towards the next waypoint on the planned path, which can be considered as the current goal point in the conventional APF algorithm. The magnitude of the $\mathbf{F}_{ag}$ force depends on the distance between the UAV and the current goal point. The main difference between conventional APF attractive force and $\mathbf{F}_{ag}$ is that $\mathbf{F}_{ag}$ is active only in the vicinity of the obstacle (e.g. when the value of repulsive force $\mathbf{F}_r$ is greater than $K^{UAV}_{threshold}$). The attractive goal potential field force is defined as follows:

$$\mathbf{F}_{ag}(\mathbf{q}) = (k_{ag}\|\mathbf{q}_g - \mathbf{q}\| + c_{ag})(\mathbf{q}_g - \mathbf{q}), \tag{12}$$

where k_{ag} is the magnitude gain, c_{ag} is a constant and $d(\mathbf{q}, \mathbf{q}_g)$ is the relative distance between the position of the UAV $\mathbf{q}$ and the current goal point $\mathbf{q}_g$.

All the potential field forces acting on the UAV in the vicinity of the obstacle are shown in Fig. 5. Normal and rotational repulsive forces are polynomial functions, which are dependent on the distance of the UAV from the obstacle (Eq. 4 and 8). Additionally, the anchor attractive force is a bell-shaped function (Eq. 11), while the attractive goal force is a linear function of distance between the UAV and current goal function (Eq. 12). Detection of local minima is possible when the velocity of the UAV falls below a predetermined threshold, the sum of all forces approaches zero and the UAV is not close enough to a current goal position. By periodically checking these three and confirming their validity, we can progressively increase the value of the rotational repulsive force at each step, leading to exponential growth of force $\mathbf{F}_{rr}$. This ensures that the rotational repulsive force will surpass all other forces, leading to the displacement of the UAV out of the local minima.

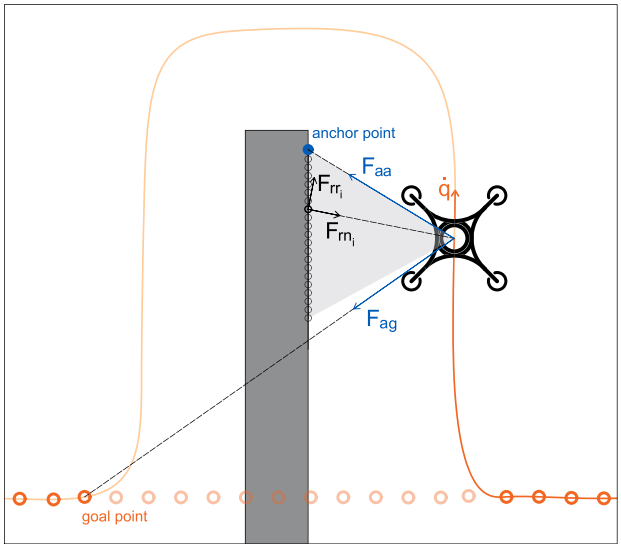


FIGURE 5. When the UAV is in the vicinity of obstacles the AAPF algorithm generates 4 forces that navigate the UAV around the obstacles and back to the planned path. Two repulsive forces are the normal repulsive force F_{rn}, which pushes the UAV away from the obstacle, and the rotational repulsive force F_{rr}, which guides the UAV around the obstacle. Two attractive forces are goal attractive force F_{ag}, which attracts the UAV to the next waypoint on the originally planned path, and the anchor attractive force F_{aa}, which drives the UAV to the vertex of the currently detected point cloud of the obstacle. Planned waypoints are shown with orange markers (solid markers are feasible, transparent ones are not) and the anchor point is marked with a blue circle.

C. RETURN TO INITIALLY PLANNED PATH

In our previous work [11], we presented a path following algorithm in which the UAV, after avoiding the obstacle, returns to the originally planned trajectory. Let the MAPF method trigger at time t_k. While the MAPF method is active, the algorithm keeps track of the time it takes to deactivate the MAPF method, e.g., to avoid the obstacle, denoted as t_o. When the total repulsive force acting on the UAV falls below the predefined threshold $K^{UAV}_{threshold}$, the obstacle is considered bypassed. The UAV continues to fly in the direction of the originally planned trajectory and follows the trajectory point with state $\mathbf{x}(t_k + t_o)$. This provided a good solution for cases where the UAV executed only small deviations from the planned path, but the results were not satisfactory when the UAV flew for a long time t_o under the MAPF method (e.g., when flying around large obstacles). In this case, the UAV may not be able to return to and continue following the originally planned trajectory.

To improve path following and keep the UAV on the planned path as long as possible, we incorporated an online waypoint feasibility determination based on the artificial potential field generated by the obstacles. A waypoint is declared feasible if it is safely reachable (e.g., it is not in or near obstacles). To compute the currently feasible waypoint (the current goal point), we use the same principle as for

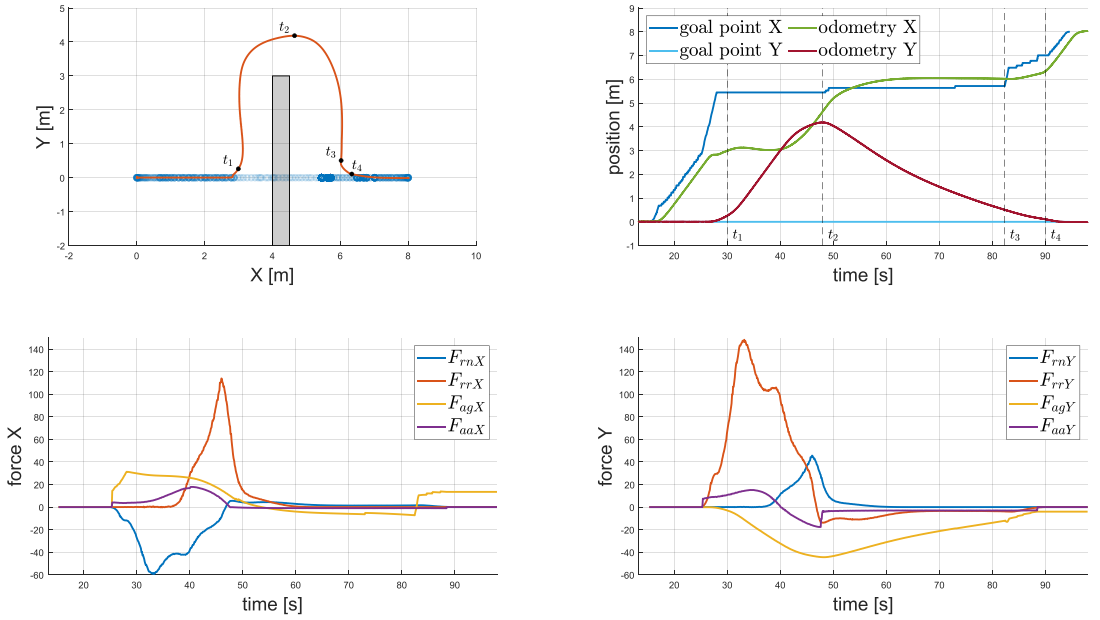


FIGURE 6. The collision-free path for the UAV was generated using artificial potential field forces. The top right graph shows the planned path as a series of goal points (marked in blue) for the UAV to reach, as well as the executed trajectory (orange line). The top left graph represents the time responses from the goal points and the position of the UAV during the simulation. The magnitudes of the forces generated by the AAPF algorithm in the X and Y axes are depicted in the bottom left and right graphs, respectively.

TABLE 1. The results of the experiments in Fig. 7 - 10. The simulations were divided into two scenarios - the first with a wall-like obstacle and the second with a cylindrical obstacle. In each scenario, we tested the MAPF and AAPF collision avoidance algorithms. Path-tracking success is marked with ✓for successfully avoiding the obstacle and reaching the end of the planned path, × for algorithms that got stuck in local minima, and × for experiments that ended with a collision with the obstacle. In addition, for each experiment we provided data for the length of the path during execution of the collision avoidance maneuver l_{dev}, the time interval the UAV deviated from the planned path t_{dev}, and the minimum d_{min} and average d_{avg} distance between the UAV and the obstacle. All experiments were repeated 10 times and results are presented as tuples of mean and standard deviation.

Scenario	Algorithm	Success	$l_{dev}[m]$	$t_{dev}[s]$	$d_{min}[m]$	$d_{avg}[m]$
wall obstacle	MAPF [11]	✓	(37.09, 0.90)	(74.72, 3.01)	(4.12, 0.07)	(5.23, 0.05)
	AAPF	✓	(24.28, 0.11)	**(48.84, 1.34)**	(1.81, 0.07)	(4.78, 0.41)
	AAPF, $F_{ag}=0$	✓	(29.01, 0.52)	(61.33, 3.35)	(2.75, 0.01)	(4.76, 0.04)
	AAPF, $F_{aa}=0$	✓	(26.16, 0.23)	(50.31, 1.36)	(1.89, 0.03)	(5.47, 0.10)
	AAPF, $F_{rn}=0$	×	-	-	0.0	-
	AAPF, $F_{rr}=0$	×	-	-	(1.01, 0.01)	(3.62, 0.84)
	AAPF, $F_{rn}=0, 2\times F_{rr}$	✓	**(23.34, 0.14)**	(55.36, 1.36)	(0.99, 0.01)	(4.28, 0.09)
cylindrical obstacle	MAPF [11]	✓	(35.71, 0.42)	(48.58, 0.37)	(2.02, 0.07)	(3.79, 0.03)
	AAPF	✓	**(27.16, 0.07)**	**(41.26, 0.36)**	(2.22, 0.01)	(3.62, 0.04)
	AAPF, $F_{ag}=0$	✓	(33.10, 0.49)	(43.08, 0.63)	(2.09, 0.07)	(3.90, 0.03)
	AAPF, $F_{aa}=0$	✓	(27.93, 0.09)	(49.19, 0.36)	(2.16, 0.02)	(4.06, 0.05)
	AAPF, $F_{rn}=0$	×	-	-	(1.63, 0.06)	(3.25, 0.12)
	AAPF, $F_{rr}=0$	✓	(62.25, 0.93)	(80.45, 1.56)	(1.07, 0.05)	(4.68, 0.04)
	AAPF, $F_{rn}=0, 2\times F_{rr}$	×	-	-	(1.92, 0.08)	(3.48, 0.18)

calculating repulsive force vectors acting on the UAV. In other words, a force acting on the current goal point is calculated from UAV sensor data. If the force magnitude is smaller than the predefined threshold $K^{goal}_{threshold}$, the current goal point is marked as feasible. Otherwise, the current goal point is not far enough from the obstacle and thus the next waypoint of the planned path is considered as the new current goal point. Now the UAV is able to return to the waypoint that is closest to the obstacle and can be reached safely, and continue to follow the optimal and originally planned trajectory. Potential field U_{gp} acting on the current goal point is defined by:

$$U_{gp}(\mathbf{q}_g) = \begin{cases} \frac{1}{2}k_{gp}(\frac{1}{\|\mathbf{q}_g-\mathbf{q}\|}-\frac{1}{d_0})^2 & \text{if } \|\mathbf{q}_g-\mathbf{q}\| \le d_o, \\ 0 & \text{if } \|\mathbf{q}_g-\mathbf{q}\| > d_o. \end{cases} \tag{13}$$

The corresponding *goal point potential field force* $\mathbf{F}_{gp}$ is derived by computing the negative gradient of the potential

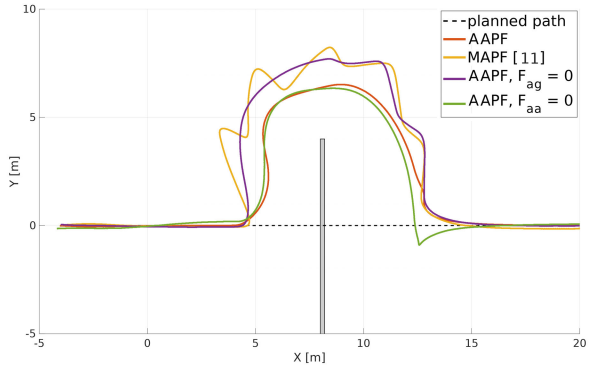


FIGURE 7. Simple scenario with a wall-like obstacle. While the UAV successfully avoided the obstacle using all methods and returned to the planned path, MAPF (yellow) and AAPF without the attractive force towards the goal point F_{ag} (purple) generated a much longer collision avoidance path. AAPF (red) and AAPF with the anchor attractive force F_{aa} set to zero (green) generated a similar obstacle avoidance path without oscillations. In the case where F_{aa} is zero, the UAV overshoots slightly when returning to the planned path.

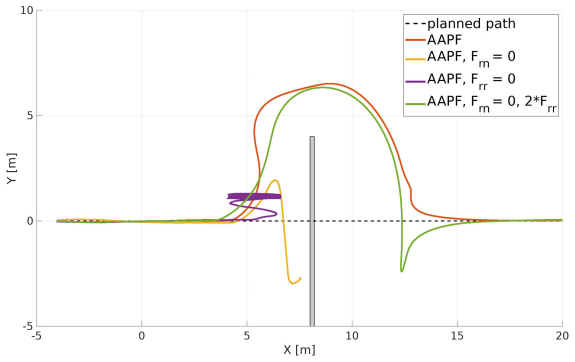


FIGURE 8. Simple scenario with a wall-like obstacle. In a case where there is no rotational repulsive force F_{rr}, the UAV gets stuck in the local minima in front of the wall (purple). This method produces very similar results as the conventional potential field. In a case without the normal repulsive force F_{rn}, the UAV could not successfully avoid the obstacle and the experiment ended with a collision (yellow). To compensate for the lack of F_{rn}, the gain for the force F_{rr} was set to twice the default value (green). This generated a collision-free path for the UAV, but with a significant overshoot when returning to the planned path.

function U_{gp} as follows:

$$\mathbf{F}_{gp}(\mathbf{q}_g) = -\nabla U_{gp}(\mathbf{q}_g) = \begin{cases} k_{gp}(\frac{1}{\|\mathbf{q}_o - \mathbf{q}_g\|} - \frac{1}{d_0}) \cdot \\ \frac{1}{\|\mathbf{q}_o - \mathbf{q}_g\|^3}(\mathbf{q}_g - \mathbf{q}_o) & \text{if } \|\mathbf{q}_o - \mathbf{q}_g\| \leq d_o, \\ 0 & \text{if } \|\mathbf{q}_o - \mathbf{q}_g\| > d_o, \end{cases} \tag{14}$$

where k_{gp} is the magnitude gain. The goal point potential field force is calculated for each new sensor measurement provided by the UAV, which means that the feasibility of the waypoints is constantly checked during the flight.

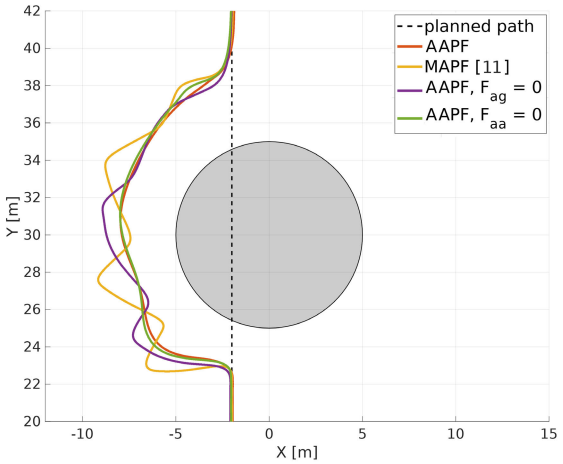


FIGURE 9. Scenario with a circular obstacle. Results are similar to the scenario with a wall-like obstacle - in all 4 cases UAV successfully avoided the obstacle and completed the planned path. MAPF (yellow) and AAPF without the F_{ag} (purple) generated avoidance path with significant oscillations.

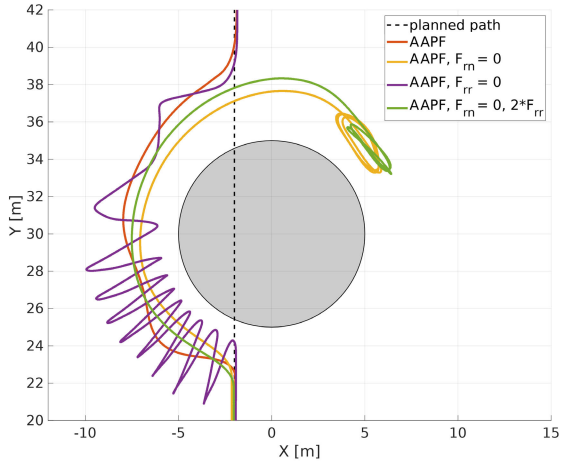


FIGURE 10. Scenario with a circular obstacle. Without the normal force component, the UAV got stuck in a local minima in both cases (yellow and green), regardless of the value of the F_{rr} gain. These are the expected results with a circular obstacle and a rotational repulsive force. In the case without the rotational component, the UAV avoided the obstacle, but with highly oscillating and longer path (purple) than in the case where all forces are active (red).

V. SIMULATION-BASED EVALUATION

Simulations are performed in the Gazebo environment using Robot Operating System (ROS) and a model of the *Kopter-worx* quadcopter. The quadcopter is equipped with a Velodyne VLP-16 LiDAR sensor. All simulations have been run on Intel(R)Core(TM) i7-10750H CPU @ 2.60GHz × 12.

The proposed AAPF algorithm was used to generate a collision-free path for the UAV, and the results of the simulation experiment are presented in Fig. 6. The planned trajectory consists of a series of goal points for the UAV to follow, and the MPC Tracker receives these points as input through the AAPF algorithm when the UAV is flying in an unobstructed environment. However, when an obstacle is

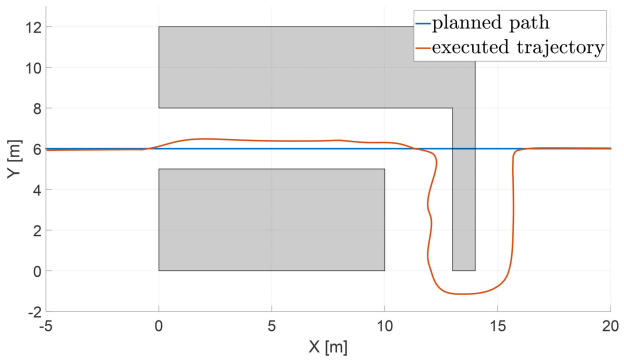


FIGURE 11. Results of a simulation experiment in a narrow L-shaped corridor, where the AAPF algorithm successfully guided the UAV through the middle of the $3m$ wide corridor and avoided collision with the obstacle at the location where the initially planned trajectory would have collided with the wall.

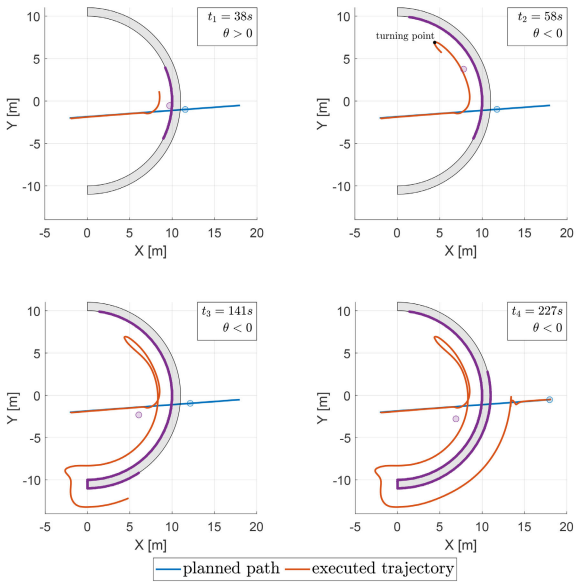


FIGURE 12. Results of a simulation experiment with a concave obstacle are displayed at four distinct time points. The planned flight path for the UAV is depicted as a blue line, while the executed trajectory up to time t_i is represented by an orange line. The purple points show the accumulated point cloud at time t_i. A blue marker indicates the current goal point on the planned path at t_i, while a purple marker represents the position the centroid of the obstacle. Angle θ is defined with the equation 9.

detected, the AAPF algorithm calculates new input points for the MPC Tracker by summing the force vectors generated by the potential forces depicted in the graphs on the bottom row of Fig. 6. This causes the UAV to deviate from the planned trajectory to avoid the obstacle, as can be seen at time point t_1. The goal points that are close to the obstacle, marked with transparent markers in the top left graph of Fig. 6, are discarded as unreachable based on the calculated force $\mathbf{F}_{gp}$. The position of the current goal point remains unchanged until time t_2, when the UAV detects the right side of the wall. At time point t_3, the distance between the UAV and the current goal point is less than the defined parameter ϵ, indicating that the current goal point has been reached. Finally, at time t_4, the repulsive forces $\mathbf{F}_{rn}$ and $\mathbf{F}_{rr}$ fall below the defined threshold $K^{UAV}_{threshold}$, indicating that the obstacle has been cleared, and the UAV resumes following the originally planned trajectory.

To further evaluate the proposed algorithm, we compare it to our previous work [11] and demonstrate how each of the four potential field forces used in the AAPF algorithm affects the UAV's behavior during collision avoidance, we ran five simulation scenarios with different complexity and analyzed the results.

In the first part of the simulation experiments, we tested the path following and obstacle avoidance algorithms in two scenarios with a simple wall-like and cylindrical obstacle and a path initially planned as a straight line. The goal of these experiments was to compare the results of the AAPF algorithm with the MAPF presented in [11], as well as to determine the effect of each of the four potential forces used in the AAPF. To do this, the simulation was run with all of the forces active in the AAPF algorithm and then one of the forces was deactivated in turn for the sequential cases. The procedure was repeated for both scenarios.

The results of the comparison between the MAPF and AAPF algorithms show that the AAPF algorithm has an improvement in the time required to avoid the obstacle and in the length of the avoidance path without a significant decrease in the average or minimum distance between the UAV and the obstacle during flight. From the data in Fig. 7 and Fig. 9, it is evident that the problem of path oscillations at the edge of the potential field has also been solved in the AAPF algorithm.

Both attractive forces had a positive effect on resolving the unnecessarily large deviations from the planned path, while avoiding simple obstacles. In both simulation scenarios, the UAV required the shortest time to fly around the obstacle when all four potential forces were active, as well as the shortest length of the avoidance path in the scenario with the cylindrical obstacle. In the scenario with the wall obstacle, the shortest length of the avoidance path was in the case where $\mathbf{F}_{rn}$ was disabled and the value of the gain for the rotational repulsive force was doubled, but in this case the minimum distance between the UAV and the obstacle was significantly smaller than in the case where all forces are active.

In the first scenario, deactivating the rotational repulsive force caused the UAV to get stuck in the local minima, while in the second scenario, the UAV got stuck in the local minima in the cases where the normal repulsion force was disabled, regardless of the gain value for the $\mathbf{F}_{rr}$ force. The only case in which a collision with the obstacle occurred was when we deactivated the normal repulsive force in the scenario with the wall obstacle.

The results of the simulations show that distance from the obstacle of the point where the UAV started the collision avoidance maneuver is similar to the distance between the obstacle and the waypoint on which UAV returned to the planned path. Exact position or the return waypoint is

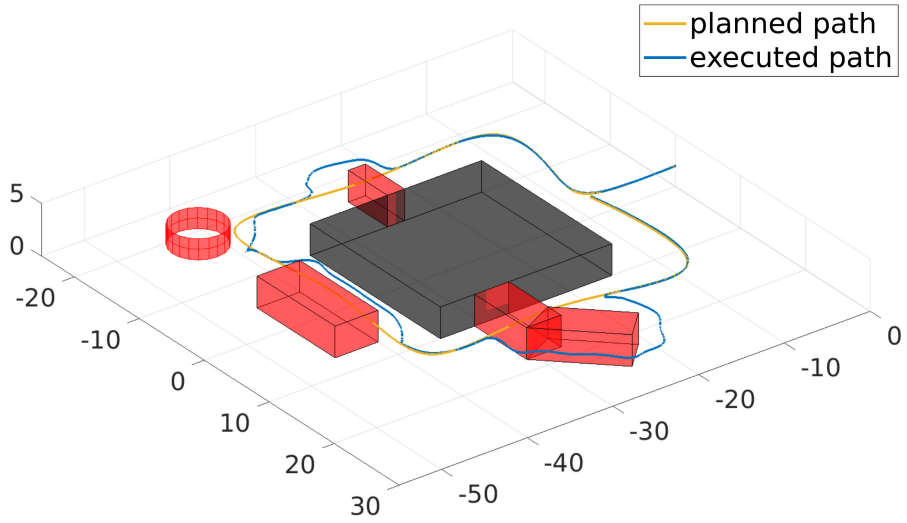


FIGURE 13. Simulation experiment in a complex, maze-like environment. Realistic path is planned around the known object of interest (black object). Red objects are unknown obstacles during the planning phase. Results show successful execution of a flight mission.

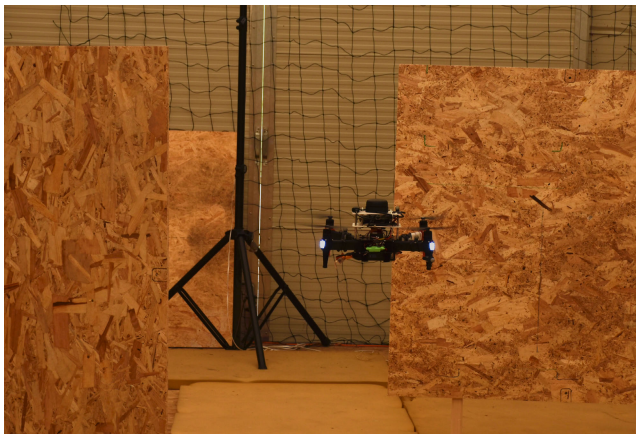


FIGURE 14. *Hexsoon EDU-450* UAV equiped with 2d LiDAR used in experimental evaluation of the AAPF algorithm.

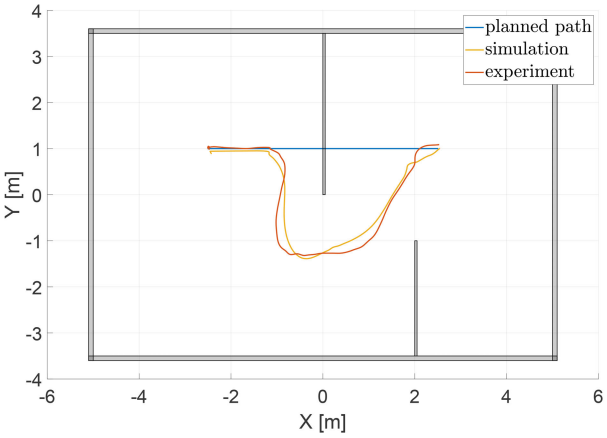


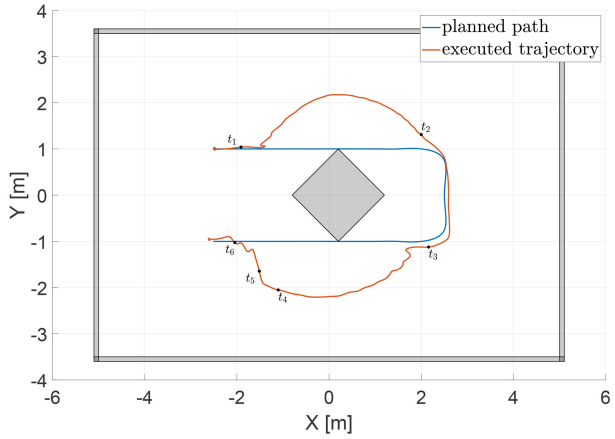
FIGURE 15. The results of the simulation and real-world experiment in the arena with two wall-like obstacles. On the graph are depicted planned trajectory (blue) and executed flight paths in simulation (yellow) and real-world experiment (orange) in the XY-plane.

determined by predefined parameter $K^{goal}_{treshold}$. A larger value of parameter $K^{goal}_{treshold}$ means that the waypoint that is determined feasible is closer to the obstacle. In case where return waypoint is too close for the UAV to reach because of the effect of the repulsive potential field the UAV will be stuck in local minima or fly in an indefinite loop around the obstacle.

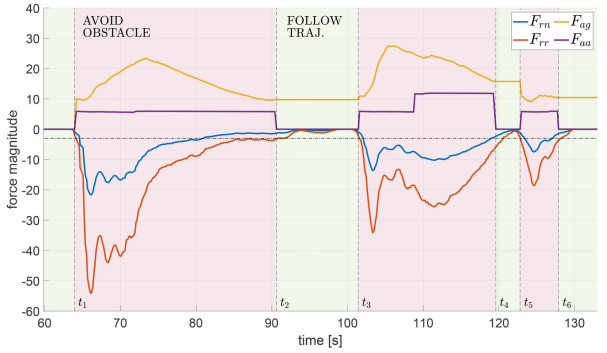
In the third scenario of the simulation experiments, the goal was to navigate the UAV through a narrow L-shaped corridor while following the planned trajectory. Narrow corridors are considered one of the most challenging obstacles for collision avoidance algorithms based on artificial potential fields, according to the literature [21]. Although the initially planned trajectory did not collide with the obstacle in the first part, the AAPF algorithm generated a small deviation to keep the UAV equidistant from both walls in the corridor. As shown in Fig. 11, the AAPF algorithm successfully generated an oscillation-free flight path for the UAV through the corridor and guided it through a sharp turn where the planned trajectory would have collided with the obstacle.

In the fourth scenario, we generated an environment featuring a large concave obstacle, which is widely recognized as a challenging problem for artificial potential field-based algorithms due to the potential for UAVs to become trapped in local minima within the obstacle. Generated obstacle is intentionally larger than the scanning range of the LiDAR so that it is not possible for the UAV to detect the whole obstacle within a single scan with the sensor.

The results of a simulation experiment with a large concave obstacle are presented in Fig. 12 at four discrete time points.



(a) Planned and executed trajectory for the experimental scenario with central obstacle in XY-plane.



(b) Magnitudes of potential field forces during the experiment with central obstacle.

FIGURE 16. Results of the experimental scenario where the initial flight trajectory for the UAV was planned close to the obstacle in the center of the area. In the graph (b) , sections, where the UAV is following the initially planned trajectory, are highlighted with a green background, while sections, where the UAV is avoiding the obstacle, are highlighted with a red background.

At $t_1 = 38$ seconds, the UAV detected only a small portion of the obstacle and initiated a collision avoidance maneuver using a clockwise direction of the rotational potential field. During the flight, the environment was mapped by adding each scan from the LiDAR into the accumulated point cloud. This allowed for the constant updating of the centroid position of the detected obstacle with new information. After a larger portion of the obstacle was detected and the UAV was at the position marked as *turning point* on the top right graph of in Fig 12 angle ρ (angle of the vector from the position of the UAV to the obstacle centroid) became larger than angle ϕ (angle of the vector from the position of the UAV to the current goal point on the planned path). This means that sign of the angle θ (defined with the Eq. 9) was changed and the rotational potential field direction was switched to counter-clockwise. By $t_3 = 120$ seconds, the UAV had successfully navigated out of the concave obstacle without getting stuck in a local minimum. Finally, the results of the experiment, shown in the lower right graph of Fig. 12 at $t_4 = 141$ seconds confirm that the UAV had successfully returned to its planned flight path.

In Fig. 13 the results of the simulation are shown in a complex maze-like environment with several unknown obstacles with different shapes. The original path was planned as a realistic scenario (e.g., inspection of a known infrastructure object in an unknown environment), where the UAV has to fly around the given object. The results show that the UAV successfully avoided all obstacles on the path using the AAPF algorithm, including the narrow corridor and the concave obstacle, which are considered extremely difficult obstacles for algorithms based on artificial potential fields. In addition to successfully generating and executing a collision-free path, the AAPF algorithm forced the UAV to return to the planned path at the first safe waypoint after avoiding the obstacle.

VI. EXPERIMENTAL EVALUATION

The flight experiments were conducted in a $10m \times 7m \times 3m$ flight arena with an *Optitrack* localization system. The *Hexsoon EDU-450* quadcopter was utilized in the experiments, equipped with the *Cube Orange* autopilot system, an *Intel NUC* on-board computer, and a *SLAMTEC RPLIDAR-A3* 2D LiDAR sensor (Fig. 14).

In the first experimental scenario, we set up two wall-like obstacles in the arena and planned the flight trajectory for the UAV to directly collide with one of them. In order to determine the initial set of parameters for the AAPF algorithm in the constructed environment, as well as to get data for analysis and evaluation of the proposed algorithm, we first ran the simulation of the same scenario. Planned and executed trajectories in both simulation and real-world experiment are depicted in Fig. 15. Results of the experiment show that the UAV managed to successfully avoid both walls, safely fly through the gap between them, return and finish the initially planned trajectory.

The second experimental scenario involved setting up a central obstacle, around which the initial flight trajectory was planned. The experimental results, displayed in Fig. 16, show that at the start of the flight, the UAV followed the planned trajectory until the repulsive forces generated by the obstacle exceeded a predetermined threshold of $K^{UAV}_{threshold} = 3$, marked as time t_1. From time point t_1 to t_2, the UAV followed the points generated by the AAPF algorithm to navigate around the obstacle. Once the obstacle was cleared, as indicated by a decrease in the magnitude of repulsive forces below $K^{UAV}_{threshold}$, the UAV resumed following the planned trajectory until encountering the obstacle for a second time at t_3. Between time points t_4 and t_5, the UAV was sufficiently far from the obstacle to move towards the current goal point on the planned trajectory. After time point t_6, the UAV resumes following the planned trajectory in unobstructed space until it reaches the end of the planned path.

In the final experimental scenario, we designed an initial trajectory for the UAV to enter a narrow concave obstacle (a closed corridor). Range of the LiDAR sensor was

TABLE 2. Parameter settings for the potential field forces in experiments with a narrow concave obstacle.

Experiment num.	d_0	k_{rn}	k_{rr}	k_{ag}	c_{ag}	k_{aa}	b_1	k_1	b_2	k_2	k_{gp}	$K_{treshold}^{goal}$
1	1.8	250	550	8.5	6.0	2.0	6.0	1.2	1.7	5.0	200	5.0
2-5	1.8	200	500	8.5	6.0	2.0	6.0	1.2	1.7	5.0	200	5.0

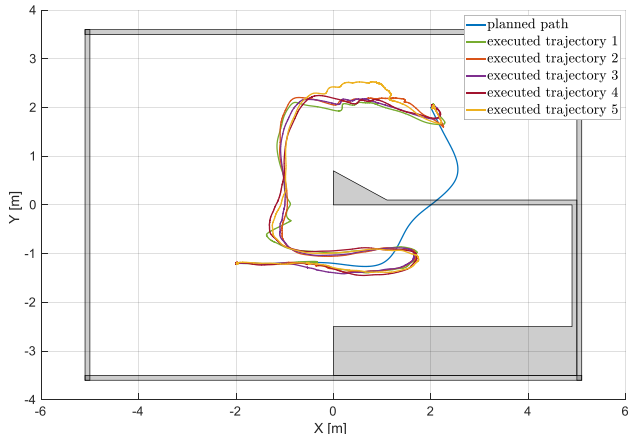


FIGURE 17. The results of the experiment with a narrow concave obstacle, showing the trajectory of the UAV as it followed the planned path into the narrow corridor where obstacles are detected as the two parallel walls. When the UAV detected the closed end of the corridor it changed the perception of the environment to a single concave obstacle with a new centroid, leading to a change in the direction of the rotational potential force. As a result, the UAV was able to successfully exit the corridor and return to the planned trajectory outside of the concave obstacle.

intentionally limited to prevent the UAV from detecting the end of the corridor. As a result, when the UAV entered the corridor, it only detected two parallel walls as separate obstacles, each with its own centroid. However, when the UAV detected the closed end of the corridor, all three walls were perceived as a single concave obstacle with a single centroid. This resulted in a change in the direction of the rotational potential field, guiding the UAV out of the corridor. To assess the AAPF algorithm's repeatability, we conducted the same test five times without making significant changes to the potential force field parameters. In the first test, visible oscillations were observed as the UAV exited the concave obstacle. To resolve this problem, we made slight adjustments to the gain coefficients of the repulsive potential forces, resulting in a reduction of oscillations in subsequent tests. Furthermore, during the fifth test, the algorithm produced different clustering results compared to the other tests. This disparity was a result of changes in the environment, specifically the presence of open doors in the flight arena, which the algorithm detected as a gap in the outer walls. As a consequence, the outer walls were perceived as separate objects instead of a single obstacle, leading to a slightly altered trajectory for the UAV as it returned to its initially planned path. As shown in Fig. 17, the UAV successfully exited the narrow corridor and returned to the planned trajectory in all five tests.

With these experiments, we demonstrated the possibility of the proposed AAPF algorithm to run with limited on-board resources and in real-time in order to navigate the UAV around the obstacles in the environment that were unknown in the planning phase of the mission.

VII. CONCLUSION AND FUTURE WORK

This paper introduces a novel AAPF method for precise path following with obstacle avoidance. The method is capable of an autonomous and safe trajectory following in a static 3D environment. Our algorithm uses augmented repulsive and attractive fields to avoid local minima and minimize deviations from the originally planned trajectory. Furthermore, the algorithm determines the feasibility of the waypoints on the initially planned path to increase the efficiency of the trajectory following. The MPC tracker allows us to correct the UAV trajectory based on the current system state and the model dynamics. The results show an improved behavior in terms of the length of the executed path, the time interval the UAV deviated from the planned path and the resolution of local minima compared to the state-of-the-art strategy. This trajectory following and collision avoidance algorithm has been successfully tested in both simulation and real-world scenarios using a quadcopter equipped with a LiDAR.

The methods and algorithms presented in this paper are aimed at enabling autonomous navigation of UAVs in a warehouse, which aligns with the primary objective of the VIRTUALUAV project mentioned in the Acknowledgement section. One of the project's goals was to ensure the UAV maintains a consistent altitude during autonomous missions. While the obstacles in the environment are detected as 3D objects, our collision avoidance trajectory is currently planned only in the horizontal plane, while the UAV maintains a constant flight altitude. In the future, we plan to perform an analytical analysis of system stability. Additionally, we intend to evaluate our method in a complex outdoor environment, which would require generating collision trajectories in a full 3D space. Furthermore, we aim to extend our algorithm to account for dynamic obstacles and explore its application in a multi-UAV system.

REFERENCES

[1] UNIZG-FER and Larics. (Nov. 30, 2019). *Aerostream Project*. [Online]. Available: https://aerostream.fer.hr/aerostream
[2] D. C. Tsouros, S. Bibi, and P. G. Sarigiannidis, "A review on UAV-based applications for precision agriculture," *Information*, vol. 10, no. 11, p. 349, Nov. 2019. [Online]. Available: https://www.mdpi.com/2078-2489/10/11/349
[3] M. Car, L. Markovic, A. Ivanovic, M. Orsag, and S. Bogdan, "Autonomous wind-turbine blade inspection using LiDAR-equipped unmanned aerial vehicle," *IEEE Access*, vol. 8, pp. 131380–131387, 2020.

[4] W. W. Greenwood, J. P. Lynch, and D. Zekkos, "Applications of UAVs in civil infrastructure," *J. Infrastruct. Syst.*, vol. 25, no. 2, Jun. 2019, Art. no. 04019002.

[5] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48572–48634, 2019, doi: 10.1109/ACCESS.2019.2909530.

[6] A. Batinovic, T. Petrovic, A. Ivanovic, F. Petric, and S. Bogdan, "A multi-resolution frontier-based planner for autonomous 3D exploration," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4528–4535, Jul. 2021.

[7] A. Batinovic, A. Ivanovic, T. Petrovic, and S. Bogdan, "A shadowcasting-based next-best-view planner for autonomous 3D exploration," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2969–2976, Apr. 2022.

[8] M. Radmanesh, M. Kumar, P. H. Guentert, and M. Sarim, "Overview of path-planning and obstacle avoidance algorithms for UAVs: A comparative study," *Unmanned Syst.*, vol. 6, no. 2, pp. 95–118, Apr. 2018.

[9] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous Robot Vehicles*. New York, NY, USA: Springer, 1986, pp. 396–404, doi: 10.1007/978-1-4613-8997-2_29.

[10] K. L. Daegyun Choi and D. Kim, "Enhanced potential field-based collision avoidance for unmanned aerial vehicles in a dynamic environment," in *Proc. AIAA Scitech Forum*, 2020, p. 0487.

[11] A. Batinovic, J. Goricanec, L. Markovic, and S. Bogdan, "Path planning with potential field-based obstacle avoidance in a 3D environment by an unmanned aerial vehicle," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2022, pp. 394–401.

[12] B. Zhou, J. Pan, F. Gao, and S. Shen, "RAPTOR: Robust and perception-aware trajectory replanning for quadrotor fast flight," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1992–2009, Dec. 2021.

[13] B. Zhou, F. Gao, L. Wang, C. Liu, and S. Shen, "Robust and efficient quadrotor trajectory generation for fast autonomous flight," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3529–3536, Oct. 2019.

[14] J. Tordesillas, B. T. Lopez, and J. P. How, "FASTER: Fast and safe trajectory planner for flights in unknown environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 1934–1940.

[15] J. Ren, K. A. McIsaac, and R. V. Patel, "Modified Newton's method applied to potential field-based navigation for mobile robots," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 384–391, Apr. 2006.

[16] S. Saravanakumar and T. Asokan, "Multipoint potential field method for path planning of autonomous underwater vehicles in 3D space," *Intell. Service Robot.*, vol. 6, no. 4, pp. 211–224, Oct. 2013.

[17] T. T. Mac, C. Copot, R. D. Keyser, and C. M. Ionescu, "The development of an autonomous navigation system with optimal control of an UAV in partly unknown indoor environment," *Mechatronics*, vol. 49, pp. 187–196, Feb. 2018.

[18] K. Hengster-Movrić, S. Bogdan, and I. Draganjac, "Multi-agent formation control based on bell-shaped potential functions," *J. Intell. Robotic Syst.*, vol. 58, no. 2, pp. 165–189, May 2010.

[19] Y. Liu and Y. Zhao, "A virtual-waypoint based artificial potential field method for UAV path planning," in *Proc. IEEE Chin. Guid., Navigat. Control Conf. (CGNCC)*, Aug. 2016, pp. 949–953.

[20] A. Ma'Arif, W. Rahmaniar, M. A. M. Vera, A. A. Nuryono, R. Majdoubi, and A. Çakan, "Artificial potential field algorithm for obstacle avoidance in UAV quadrotor for dynamic environment," in *Proc. IEEE Int. Conf. Commun., Netw. Satell. (COMNETSAT)*, Jul. 2021, pp. 184–189.

[21] D. Schleich and S. Behnke, "Predictive angular potential field-based obstacle avoidance for dynamic UAV flights," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 13618–13625.

[22] X. Gao, H. Wu, L. Zhai, H. Sun, Q. Jia, Y. Wang, and L. Wu, "A rapidly exploring random tree optimization algorithm for space robotic manipulators guided by obstacle avoidance independent potential field," *Int. J. Adv. Robotic Syst.*, vol. 15, no. 3, May 2018, Art. no. 172988141878224.

[23] W. Xinyu, L. Xiaojuan, G. Yong, S. Jiadong, and W. Rui, "Bidirectional potential guided RRT* for motion planning," *IEEE Access*, vol. 7, pp. 95046–95057, 2019.

[24] A. A. Ahmed, T. Y. Abdalla, and A. A. Abed, "Path planning of mobile robot by using modified optimized potential field method," *Int. J. Comput. Appl.*, vol. 113, no. 4, pp. 6–10, Mar. 2015.

[25] J. Sun, J. Tang, and S. Lao, "Collision avoidance for cooperative UAVs with optimized artificial potential field algorithm," *IEEE Access*, vol. 5, pp. 18382–18390, 2017.

[26] T. Weerakoon, K. Ishii, and A. A. F. Nassiraei, "An artificial potential field based mobile robot navigation method to prevent from deadlock," *J. Artif. Intell. Soft Comput. Res.*, vol. 5, no. 3, pp. 189–203, Jul. 2015, doi: 10.1515/jaiscr-2015-0028.

[27] P. Sudhakara, V. Ganapathy, B. Priyadharshini, and K. Sundaran, "Obstacle avoidance and navigation planning of a wheeled mobile robot using amended artificial potential field method," *Proc. Comput. Sci.*, vol. 133, pp. 998–1004, Jan. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918310214

[28] T. T. Mac, C. Copot, A. Hernandez, and R. De Keyser, "Improved potential field method for unknown obstacle avoidance using UAV in indoor environment," in *Proc. IEEE 14th Int. Symp. Appl. Mach. Intell. Informat. (SAMI)*, Jan. 2016, pp. 345–350.

[29] L. Zhu, X. Cheng, and F. G. Yuan, "A 3D collision avoidance strategy for UAV with physical constraints," *Measurement*, vol. 77, pp. 40–49, Jan. 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224115004716

[30] H.-T. Chiang, N. Malone, K. Lesser, M. Oishi, and L. Tapia, "Path-guided artificial potential fields with stochastic reachable sets for motion planning in highly dynamic environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 2347–2354.

[31] S. Patil, J. van den Berg, S. Curtis, M. C. Lin, and D. Manocha, "Directing crowd simulations using navigation fields," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 2, pp. 244–254, Feb. 2011.

[32] L. Jaillet, J. Cortés, and T. Siméon, "Sampling-based path planning on configuration-space costmaps," *IEEE Trans. Robot.*, vol. 26, no. 4, pp. 635–646, Aug. 2010.

[33] S. Scherer, S. Singh, L. Chamberlain, and M. Elgersma, "Flying fast and low among obstacles: Methodology and experiments," *Int. J. Robot. Res.*, vol. 27, no. 5, pp. 549–574, May 2008.

[34] X. Chen and J. Zhang, "The three-dimension path planning of UAV based on improved artificial potential field in dynamic environment," in *Proc. 5th Int. Conf. Intell. Human-Machine Syst. Cybern.*, vol. 2, Aug. 2013, pp. 144–147.

[35] E. Oland and R. Kristiansen, "Collision and terrain avoidance for UAVs using the potential field method," in *Proc. IEEE Aerosp. Conf.*, Mar. 2013, pp. 1–7.

[36] T. P. Nascimento, A. G. S. Conceicao, and A. P. Moreira, "Multi-robot systems formation control with obstacle avoidance," *IFAC Proc. Volumes*, vol. 47, no. 3, pp. 5703–5708, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016425036

[37] Z. Zhang, Q. Ling, and Z. Yang, "Formation control with obstacle avoidance of multi-robot systems with second-order dynamics," in *Proc. Chin. Control Conf. (CCC)*, Jul. 2019, pp. 5978–5983.

[38] H. Rezaee and F. Abdollahi, "Adaptive artificial potential field approach for obstacle avoidance of unmanned aircrafts," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2012, pp. 1–6.

[39] H. Pham and Q.-C. Pham, "A new approach to time-optimal path parameterization based on reachability analysis," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 645–659, Jun. 2018.

[40] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 6753–6760.

[41] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 1–4.

JURICA GORICANEC (Graduate Student Member, IEEE) received the B.S.E.E. and M.S.E.E. degrees from the University of Zagreb, Croatia, in 2011 and 2013, respectively, where he is currently pursuing the Ph.D. degree with the Laboratory for Robotics and Intelligent Control Systems (LARICS), Faculty of Electrical Engineering and Computing (FER). After graduation, he worked on development and usage of unmanned aerial vehicles in archaeology excavations, infrastructure surveys, and terrain mapping. He is also a Research Associate with LARICS, UNIZG-FER. He received the Rector's Award, in 2012, for his work on development of self-rotating aerial vehicle.

ANA MILAS (Graduate Student Member, IEEE) received the B.S.E.E. and M.S.E.E. degrees from the University of Zagreb, Croatia, in 2016 and 2018, respectively, where she is currently pursuing the Ph.D. degree. In 2019, she did an internship with the University of Seville, Spain. She is a Research Associate with the Faculty of Electrical Engineering and Computing, University of Zagreb. Her research interests include aerial robotics, autonomous mapping, and exploration. In her undergraduate years, she received the University of Zagreb Rector Award for the Best Student Projects, in 2018.

LOVRO MARKOVIC (Member, IEEE) received the B.S.E.E. and M.S.E.E. degrees from the University of Zagreb, Croatia, in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree with the Faculty of Electrical Engineering and Computing. He is also with the Faculty of Electrical Engineering and Computing, University of Zagreb, as a Research Associate. His latest publication includes *Geometric Tracking Control of Aerial Robots Based on Centroid Vectoring*. His research interest include aerial robotics, geometric control theory, and autonomous operation systems.

STJEPAN BOGDAN (Senior Member, IEEE) received the B.S.E.E., M.S.E.E., and Ph.D.E.E. degrees from the University of Zagreb, Croatia, in1990, 1993, and 1999, respectively.

He spent one year as a Fulbright Researcher with the Automation and Robotics Research Institute, Arlington, USA, under Prof. Frank Lewis with Laboratory. He is currently a Full Professor with the Laboratory for Robotics and Intelligent Control Systems (LARICS), Department on Control and Computer Engineering, Faculty of Electrical Engineering and Computing (FER), University of Zagreb (UNIZG). He is the coauthor of four books and numerous articles published in journals and proceedings. His research interests include autonomous systems, aerial robotics, multi-agent systems, intelligent control systems, bio-inspired systems, and discrete event systems.