OPTIMIZING TRAJECTORIES IN 3D SPACE USING MIXED-INTEGER LINEAR PROGRAMMING

Mirela-Mădălina BIVOLARU[1,2]

Trajectory planning represents a heavily investigated area across robotics, artificial intelligence, and aerospace domains. Predominantly challenging, pre-mission trajectory planning involves navigating through multi-obstacle environments, posing a complex optimization problem. This paper aims to address this challenge by developing a path planner leveraging Mixed Integer Linear Programming, that solves quickly and produces a safe but not very rigid trajectory from the start point to the end point, in a constrained 3D environment.

Keywords: trajectory planning, MILP, obstacles, UAV

1. Introduction

In recent years, many research studies on the designing of unmanned aerial vehicles have been conducted. These systems have important applications in both military and civilian fields. For example, if a task is dangerous or monotonous, replacing human-operated vehicles with unmanned vehicles is desirable.

When a UAV is on mission, it is very important to ensure that it detects obstacles such as mountains, buildings, and other aerial vehicles. No-fly zones should also be avoided.

Many methods have been used to solve the obstacle avoidance trajectory optimization problem. Potential functions are used for space vehicles [1], air traffic management and trajectory planning for UAVs [2], [3]. This method involves replacing the obstacle avoidance constraints with approach penalties in the objective function, thus simpler optimization schemes such as pitch reduction can be used. These schemes offer the operation of problems in a shorter time, some proving to be safe, but not optimal. Random search methods [4] [5] [6] have been designed to rapidly find feasible trajectories among obstacles, again neglecting trajectory optimality. In the problems of UAVs, Voronoi diagrams [7], [8] were used to find the paths between obstacles, these paths being formed by straight segments. Other methods use splines [9] and low-dimensional representations [10]

[1] PhD student, Doctoral School of Aerospace Engineering, University POLITEHNICA of Bucharest, Romania, e-mail: mirela.bivolaru@stud.aero.upb.ro
[2] Scientific Research Assistant/Aerospace Engineer, Romanian Space Agency, Romania

of nonlinear systems to reduce the solution space before performing nonlinear optimization.

Reference [11] adopts Mixed-Integer Linear Programming (MILP) for trajectory planning of UAVs and vehicles with turn and minimum velocity constraints. Path planning for air vehicles based on MILP optimization was first presented by Schoewenaars [12], and this method has been studied and extended by many others. MILP can also be viewed as a reaction in classical control problems [13]. [14] presents a method for tracking icebergs with multiple UAVs, utilizing a MILP approach for path planning.

This study focuses on trajectory planning for fixed-wing UAVs, presenting a simplified model that demonstrates the potential of using MILP for efficient path optimization. The objective of this paper is to create a fast solver for path planning using Mixed Integer Linear Programming, that generates a robust trajectory, considering the obstacles and the acceleration limitation. The complexity of the problem is given by the solution type, the number of decision variables and the number of restrictions.

2. Methodology

This section describes the mixed-integer linear programming (MILP) design methodology and explains the formulation of the core MILP problem used in this paper. Some of the MILP components described in this chapter are the same as those used in other previous implementations [12] [13], but others, such as safety distance, were made by modifying existing techniques, thus being able to improve solver's performance.

2.1 General formulation of the MILP problem

MILP standard formulation is :
Minimize the cost function

$$f(x) = s^T x \tag{1}$$

Given the restrictions

$$\begin{gathered} A_{eq} x = a_{eq} \\ Ax \in [a_{low}; a_{up}] \\ x \in [x_{low}; x_{up}] \end{gathered} \tag{2}$$

where $A_{eq}, A \in \mathbb{R}^{m\times n}$ $are\ matrices,$ $s, x, x_{low}, x_{up}, a_{low}, a_{up} \in \mathbb{R}^n, a_{low}, a_{up} \in \mathbb{R}^m$ $are\ vectors.$

2.1.1 Discretization of time

The initial phase in tackling the mixed-integer linear programming problem involves time discretization. If $t_0=0$ denotes the starting time, N represents the total number of steps, and T_i is the time interval between two consecutive timesteps t_i and t_{i-1}, the total time t_{fin} required to solve the MILP problem is:

$$t_{fin} = t_0 + t_1 + t_2 + \cdots + t_N = \sum_{i=0}^{N} t_i \tag{3}$$

In this paper the time is divided equally, i.e. $\boldsymbol{t_i} = \boldsymbol{T}$ is constant, therefore eq. (3) is equivalent to:

$$\boldsymbol{t_{fin}} = \boldsymbol{N} \cdot \boldsymbol{T} \tag{4}$$

2.1.2 Initial conditions

The initial conditions will be defined next. Knowing the last information about the position and velocity, $\boldsymbol{x_{known}}$ and $\dot{\boldsymbol{x}}_{\boldsymbol{known}}$, the initial conditions can be predicted by linear interpolation with the timestep T:

$$\begin{bmatrix} x_0 \\ \dot{x}_0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} x_{known} + \begin{bmatrix} T \\ 1 \end{bmatrix} \dot{x}_{known} \tag{5}$$

2.2 Restrictions

2.2.1 Restrictions on reaching the destination

The way the UAV gets from the start point to the final point can be expressed through constraints. The conditions for the vehicle to successfully reach its destination point are:

$$\begin{gathered} x_t - x_F \le D(1-\beta_i) \\ x_t - x_F \ge -D(1-\beta_i) \\ y_t - y_F \le D(1-\beta_i) \\ y_t - y_F \ge -D(1-\beta_i) \\ z_t - z_F \le D(1-\beta_i) \\ z_t - z_F \ge -D(1-\beta_i) \end{gathered} \tag{6}$$

$$\sum_{i=1}^{N} \beta_i \leq 1$$

where (x_t, y_t, z_t) is the point where the vehicle is at time t, (x_F, y_F, z_F) is the destination point, β_i is a binary variable for each timestep, and D>0 is an arbitrarily chosen constant, large enough. When $\beta_i = 1,\ x_t - x_F = 0,\ y_t - y_F = 0,\ z_t - z_F = 0$, therefore the UAV has reached its destination.

2.2.2 Restrictions for avoiding collision with an obstacle

2.2.2.1 Increasing obstacles

Although obstacles that are in the operating area of the UAV are included in the MILP problem, the trajectory between time steps can cut the corner of an obstacle. This means that the trajectory between two feasible time steps intersects the obstacle, and this problem must be considered, so that the trajectory calculated by the MILP method does not intersect the obstacle.

In general, the real obstacle, to which the safety margins have been added, should be large enough so that it is not feasible to make a trajectory directly through the obstacle. This determines the fulfillment of the conditions [15]:

$$\begin{aligned} x_{max} - x_{min} &> V_{max}T \\ y_{max} - y_{min} &> V_{max}T \\ z_{max} - z_{min} &> V_{max}T \end{aligned} \tag{7}$$

To simplify the problem, it is assumed that all the obstacles are cubes. The distance from the obstacle limit to the safety margin is denoted by d, and this is the limit that will be implemented in MILP.

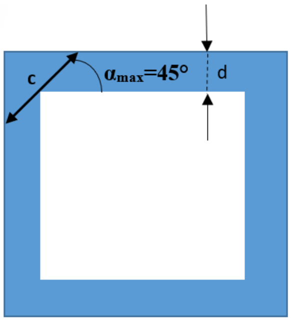

Fig. 1 Obstacle's safety margin

The distance that the UAV can cover for each time step is described by the following inequality:

$$c > V_{max}T \tag{8}$$

where V_{max} is the maximum speed that the UAV can reach, and T is the time step.

Fig. 1 shows that the distance *c* can be described as a function of the margin of safety, d, the dependence described by the relation:

$$c = \frac{2d}{\sin(\alpha)} \tag{9}$$

where α is the angle between the obstacle limit and the line described by the waypoints. In Fig. 1, the angle α is 45°, because it ensures the shortest path, so it is the extreme case for which the safety margin is built.
By substituting equation (9) in the inequality (8), the safety margin should be greater than:

$$d > \frac{V_{max}T}{2}\sin(45°) \tag{10}$$

For a $V_{max} = 36m/s$, and $T = 0.15s$, the safety margin should be greater than 1.91m. Therefore, it is chosen:

$$d = 2m$$

2.2.2.2 Restrictions for avoiding collision with obstacles

The positions of the obstacles will be defined using the upper and lower limits for each dimension: x_{max}, x_{min}, y_{max}, y_{min}, z_{max}, z_{min}. The constraints for each time step are:

$$\begin{aligned} x_t - x_{max} - d &\geq -M\varphi_1^t \\ x_{min} - d - x_t &\geq -M\varphi_2^t \\ y_t - y_{max} - d &\geq -M\varphi_3^t \\ y_{min} - d - y_t &\geq -M\varphi_4^t \\ z_t - d - z_{max} &\geq -M\varphi_5^t \\ z_{min} - d - z_t &\geq -M\varphi_6^t \end{aligned} \tag{11}$$

$$\sum_{i=1}^{6} \varphi_i^t \leq 5$$

In the equations provided earlier, the variable denoted by M serves as a sufficiently large value that dictates the activation of the binary variable φ_i^t in cases where equality fails to hold. This strategic utilization of M, often referred to as the "big M" technique, holds widespread application across mixed-integer linear programming problems and is extensively discussed in [16]. By employing this technique, the system can effectively manage constraints and decision variables, ensuring robust optimization outcomes. Specifically, if the cumulative sum of binary variables remains at or below the threshold of 5, it guarantees that the designated waypoint (x_t, y_t, z_t) remains free from any interference posed by surrounding obstacles.

2.2.3 Vehicle dynamics restrictions

The dynamics equations for any air vehicle are usually non-linear, so the dynamics must be approximated as linear or piecewise linear, in order to be included in the MILP problem. This section describes the system of equations used to represent the vehicle dynamics, which includes limits on acceleration, velocity, takeoff/landing speed, takeoff/landing acceleration rate.

The equations of state used in MILP are linearized and discretized versions of the vehicle model. The general form for a linear time-invariant (LTI) system for a continuous-time model is:

$$\dot{x} = Ax + Bu \tag{12}$$

For the scenario involving the double integrator [11], the dynamics of the UAV may be expressed as follows:

$$\frac{d}{dt}\begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} 0 & I_3 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ u \end{bmatrix} \tag{13}$$

In the three-dimensional scenario, x represents the position vector, defined as $x = [\,x \ y \ z\,]^T$ in the three-dimensional case, v stands for the velocity vector given by $v = [\,v_x \ v_y \ v\,_z]^T$, and u comprises commands for acceleration, denoted as $u = [\,u_x \ u_y \ u\,_z]^T$.

Although the initial approximation might appear rudimentary, the core characteristic of the double integrator's dynamics, which remains unaffected by the heading angle, facilitates the computational efficiency of vehicle simulation within MILP frameworks employing Cartesian coordinates.

Using a zero-order approximation of state changes, one obtains [11]:

$$\begin{bmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \end{bmatrix}_t + \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & \frac{T^2}{2} \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \tag{14}$$

2.3 Objective function

The objective function of the trajectory planning model aims at minimizing the UAV's acceleration. This choice is based on the need to achieve smooth and efficient trajectories for optimal UAV operation. Trajectories characterized by abrupt changes in acceleration often lead to inefficiencies, potentially hindering the vehicle's ability to adhere closely to the desired path. By prioritizing the minimization of acceleration within the objective function, we aim to ensure that the resulting trajectories exhibit smoother transitions, enhancing overall performance and maneuverability. This not only smoothens the trajectory but also implicitly reduces energy consumption by avoiding unnecessary accelerative forces. Therefore, the objective function is:

$$f^T x = \sum_{n=1}^{N} (a_{x,n} + a_{y,n} + a_{z,n}) \tag{15}$$

3. Numerical simulations

This section presents the optimal trajectory calculated using Mixed Integer Linear Programming method when there are restrictions on the dynamics of the UAV and on the environment. Simulation results for trajectory planning with obstacle avoidance will be presented and discussed.

In this paper, the MILP problem will be solved with the MATLAB function *intlinprog.m*, a mixed-integer linear problem solver.

The initial position is given by the coordinates (0,0,5), the final position is (200,150,20), and the optimization is made by minimizing the acceleration. The coordinates of the obstacles are presented in *Table 1*.

Table 1

Obstacles' coordinates for the three scenarios

Obstacle	$(x_{min}, y_{min}, z_{min})$	$(x_{max}, y_{max}, z_{max})$
1	(100, 80, -10)	(135,140,20)
2	(70, 20, -10)	(90,50,10)
3	(150, 80, -10)	(170,120,35)

Table 2 highlights three distinct cases for which the numerical simulations were made, where V_{x0}, V_{y0}, and V_{z0} are the initial velocities on the three axes. The three scenarios reveal the impact of the initial velocity on the trajectory of the UAV.

Table 2

Simulation parameters for the three scenarios

Scenario	V_{x0} [m/s]	V_{y0} [m/s]	V_{z0} [m/s]	t_f [s]	$a_{x,max}$ [m/s^2]	$a_{y,max}$ [m/s^2]	$a_{z,max}$ [m/s^2]
1	17	2	2	20	2	2	2
2	6	15	2	20	2	2	2
3	6	2	14	20	2	2	2

In the first case, Fig. 2 shows that the trajectory was generated by going around the obstacles, through the right side. This is due to the force generated by the X component of the initial velocity. Additionally, it is noteworthy that the trajectory is smooth and does not show sudden changes in the heading angle.

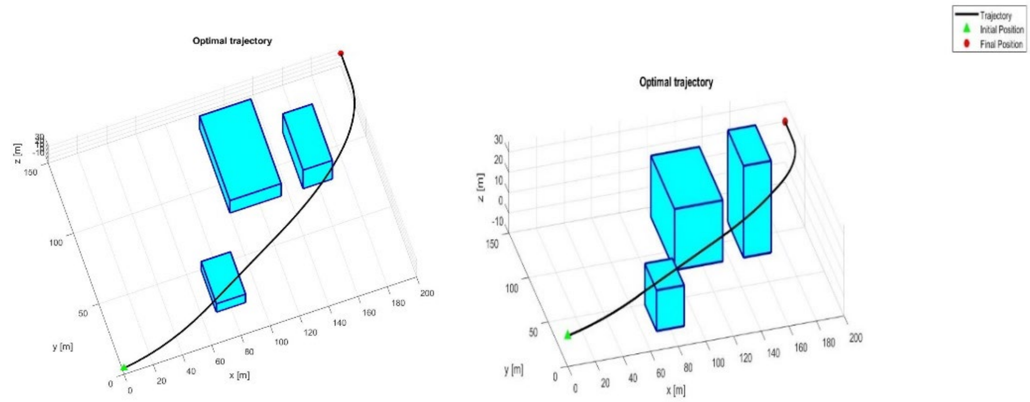

Fig. 2 Trajectory generated in the first scenario

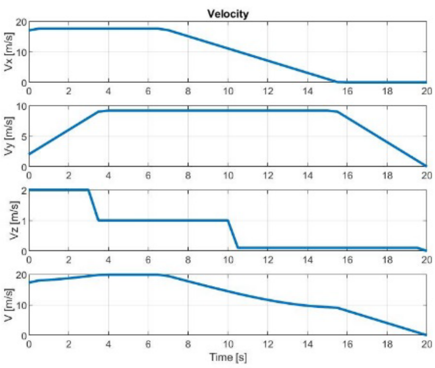

Fig. 3 Velocity of the UAV for the first scenario

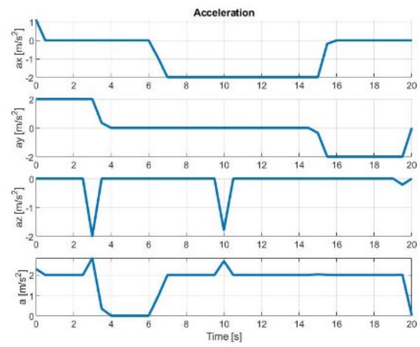

Fig. 4 Acceleration of the UAV for the first scenario

Fig. 5 illustrates the trajectory of the UAV for the second case. It is visible that the path is different from the one in the first case, the trajectory being slightly oriented to the left. Furthermore, as anticipated in Section 2, some waypoints of the trajectory were generated very close to an obstacle, this posed no issue due to the precautionary measure of enlarging the obstacles with a safety margin.

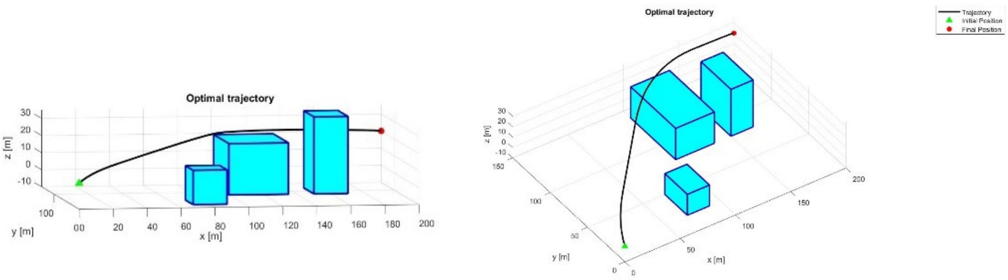

Fig. 5 Trajectory generated for the second scenario

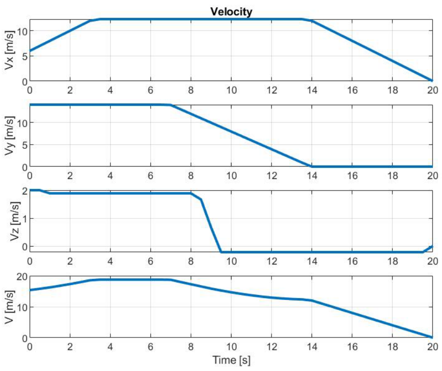

Fig. 6 Velocity of the UAV for the second scenario

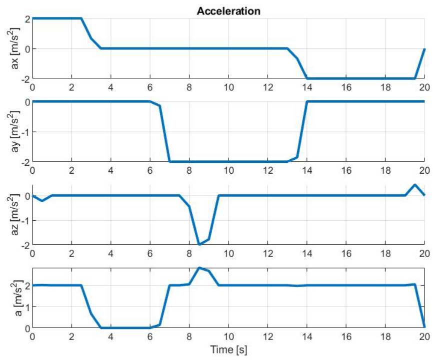

Fig. 7 Acceleration of the UAV for the second scenario

The results for the third scenario are presented in Figure 8-10. As can be seen, the trajectory has a larger amplitude in the yOz plane compared to the other cases. This is due to the high initial velocity, V_{z0}. The speed is unnecessarily high, as illustrated in Fig. 10, prompting the UAV to initiate the trajectory with a significant deceleration along the Z component.

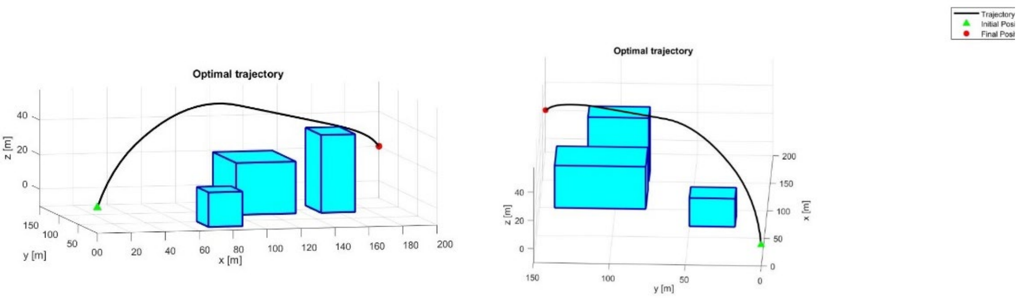

Fig. 8 Trajectory generated for the third scenario

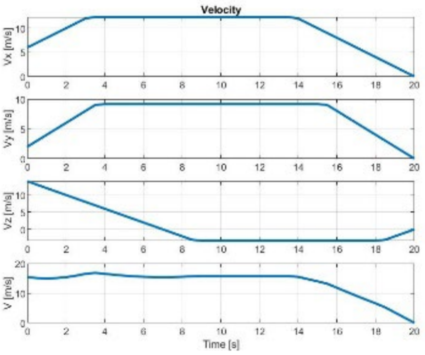

Fig. 9 Velocity of the UAV for the third scenario

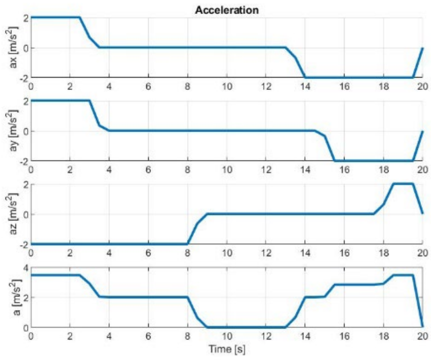

Fig. 10 Acceleration of the UAV for the third scenario

4. Conclusions

This paper presents a trajectory planning methodology employing mixed integer linear programming in a dynamic three-dimensional setting, accounting for obstacles. It undertakes simulation across three distinct scenarios, each characterized by the consistent presence of three obstacles of varying sizes. Notably, these scenarios maintain identical starting and ending points, with the primary variation lying in the velocity of the UAV. To address potential collision risks arising from closely spaced waypoint generation, the sizes of obstacles are intentionally augmented.

The mathematical formulation was established as a linear problem, featuring a minimizable cost function and a set of constraints. The cost function, represented by UAV acceleration, aimed for minimization, while the constraints encompassed two categories: inequality restrictions to ensure obstacle avoidance during UAV navigation from start to end points, and equality restrictions pertaining to vehicle dynamics.

The MILP-based trajectory planner effectively met the specified requirements, generating collision-free trajectories across all scenarios from start to end points. Moreover, analysis of acceleration graphs indicated that acceleration remained within predefined limits. Therefore, it can be concluded that the Mixed-Integer Linear Programming method offers high efficiency in trajectory planning amidst obstacle-laden environments.

These simulations serve as an initial step, demonstrating the feasibility of implementing the MILP-based trajectory planning on a real UAV in a real-world environment. However, for future work, the complexity of the model must be increased by taking into account detailed UAV parameters to ensure more accurate and practical applications.

R E F E R E N C E S

[1] *A. Roger și C. McInnes*, „Safety Constrained Free-Flyer Path Planning at the International Space Station," Journal of Guidance, Control and and Dynamics, vol. **23**, nr. 6, pp. 971-979, 2000.

[2] *A. Batinovic, G. Jurica, M. Lovro și B. Stjepan*, „Path Planning with Potential Field-Based Obstacle Avoidance in a 3D Environment by an Unmanned Aerial Vehicle," International Conference on Unmanned Aircraft Systems (ICUAS), pp. 394-401, 2022.

[3] *G. Hao, Q. Lv, Z. Huang, H. Zhao și W. Chen*, „UAV Path Planning Based on Improved Artificial Potential Field Method," Aerospace, vol. **10**, nr. 6, 2023.

[4] *M. Jiawei, P. Vijay, K. Sebastian și L. Angran*, „UAV Path Planning System Based on 3D Informed RRT* for Dynamic Obstacle Avoidance," 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1653-1658, 2018.

[5] *X. Zhefan, D. Di și S. Kenji*, „Autonomous UAV Exploration of Dynamic Environments Via Incremental Sampling and Probabilistic Roadmap," IEEE Robotics and Automation Letters, vol. **6**, nr. 2, pp. 2729-2736, 2021.

[6] *J. Nieto, E. Slawinski, V. Mut și B. Wagner*, „Online path planning based on Rapidly-Exploring Random Trees," IEEE International Conference on Industrial Technology, pp. 1451-1456, 2010.

[7] *X. Chen și X. Chen*, „The UAV dynamic path planning algorithm research based on Voronoi diagram," The 26th Chinese Control and Decision Conference, pp. 1069-1071, 2014.

[8] *T. Wei, Y.-j. Hu, Y.-f. Zhao și W.-g. Li*, „Mission Planning for Unmanned Aerial Vehicles Based on Voronoi Diagram-Tabu Genetic Algorithm," Wireless Communications and Mobile Computing, 2021.

[9] *K. Judd și T. McLain*, „Spline Based Path Planning For Unmanned Air Vehicles," AIAA Guidance, Navigation and Control Conference and Exhibit, 2001.

[10] *M. B. Milam, K. Mushambi și R. M. Murray*, „A new Computational Approach to Real Time Trajectory Generation for Constrained Mechanical Systems," Proceedings of the 39th IEEE Conference on Decision and Control , vol. **1**, pp. 845-851, 2000.

[11] *K. Culligan*, „Online trajectory planning for UAVs using mixed integer linear programming," Master's Thesis, 2006.

[12] *T. Schowenaars*, „Mixed integer programming for optimal collision-free path planning of autonomous vehicles," Katholieke Universiteit Leuven, Department of Electrical Engineering, 2001.

[13] *A. Richards și J. How*, „ Mixed-integer programming for control," Proceedings of the 2005, American Control Conference, vol. **4**, pp. 2676-2683, 2005.

[14] *A. Albert, F. Leira și I. Lars*, „UAV path planning using MILP with experiments," Modeling, Identification and Control, vol. 38, pp. 21-32, 2017.

[15] *Y. Kuwata*, „Real-Time Trajectory Design for Unmanned Aerial Vehicles using Receiding Horizon Control," 2003.

[16] *A. Hossein*, „Big-M free solution algorithm for general linear programs," International Journal of Pure and Applied Mathematics, vol. **32**, pp. 549-564, 2006.