Model predictive control for path planning of UAV group

D L Vinokursky¹, O S Mezentseva¹, Ph V Samoylov¹ K Y Ganshin¹ and O A Baklanova²

¹ Institute of Information Technologies and Telecommunications, North-Caucasus Federal University, Stavropol, Russia
² Department of Philosophy, North-Caucasus Federal University, Stavropol, Russia

E-mail: fsamoilov@ncfu.ru

Abstract. This paper addresses a predictive control strategy for Unmanned Aerial Vehicles. The goal is to guarantee tracking capabilities with respect to a reference trajectory which is predefined using Gradient descent method. The proposed method provides effective performance validated through flight experiment using micro-UAVs.

1. Introduction
Model Predictive Control (MPC) is an optimization-based control technique, where at every sampling instant a finite-horizon optimal control problem (OCP) is solved. Historically, MPC has its roots in the process industry, where it has been applied successfully for a long time and is considered a mature technology [11]. The method has been developed under many different names throughout the years, and is also known under other names such as Receding Horizon Control (RHC), Dynamic Matrix Control (DMC) [12], or Generalized Predictive Control (GPC) [13]. What unifies these methods is their use of models of the system dynamics to obtain input signals by minimizing an objective function, and that the optimization is done according to the receding horizon principle, as is illustrated in figure 1. Though initially applied primarily to relatively slow processes, recent algorithmic advances and modern hardware have enabled successful implementations of MPC in fast systems that have sampling times of fractions of seconds, for example drones [14] and trucks [15]. Currently, there is a growing interest in group control of multi-agent systems. It is necessary to have a control system that can support the formation of groups, prevent collisions of agents and avoid obstacles by the whole group without losing the integrity of the entire system.

Figure 1. MPC structure.
2. Methods and materials
The use of a single MPC controller to control the entire group is difficult from the view of the fact that the used mathematical model must describe the dynamics of the entire group at once, which also leads to a manifold increase in the number of manipulating, manipulated and output signals. Considering the fact that this controller makes a prediction of system states, then this process can be computationally complex, and the optimizer will not be able to provide the required output states in real time. It should also be considered that in the case of swarm control, each individual agent must independently perform control operations on its own platform moving in space according to general swarm strategy [1].

To solve the problem of group and swarm control, a decentralized MPC controller (DMPC) is most often used [2 - 4]. In work [2], the authors describe a decentralized control system for multiple aircraft based on the Delaunay triangulation algorithm, due to which the agents line up into a regular network-like structure that can be rebuilt in the process of avoiding obstacles, and then returning to their original form. The method of virtual potentials is used to determine the required azimuth and maintain the required speed to keep the grid structure of the group. In this article, the MPC is used for trajectory tracking and obstacle avoidance maneuvers.

Figure 2 describes the architecture of the trajectory control and collision avoidance system. This system works on each UAV and consists of four blocks:

![Figure 2. Architecture of the trajectory control and collision avoidance subsystem. RTG - reference trajectory generator; CAA - a collision avoidance algorithm that limits the MPC when an obstacle is detected; MPC - a predictive controller that calculates the acceleration vector required to maintain a reference path and avoid obstacles; OPP - is a post-processing unit that converts the acceleration vector into a control signal for the aircraft control system, consisting of acceleration and turn rate.](image)

As can be seen from the presented block diagram of the algorithm, at each time moment $k$, the optimal control action $u^*_i(k|k)$ is determined, obtained by solving the optimal control problem on the finite horizon of predictions $[k, k + n_p]$. To generate a reference trajectory, states about the azimuth and velocity of each individual agent in the group are sent to a specific MPC. To obtain the optimal control action, it is necessary to find the optimal sequence $U^*_i(k)$. The numerical solution of this problem is carried out using quadratic programming techniques. In order to prevent collisions, only the closest obstacle is taken into account, due to which a finite set of restrictions is formed.

The performance system checks [2] were carried out under the conditions of different number of aircrafts in the simulation environment. To conclude of the simulation, it was noted that UAVs are formed into a network, and when flying around large obstacles, they are divided into several small groups supporting a mesh structure of triangles, which subsequently tend to merge into a common formation, if other obstacles are not avoided at this time. The most important indicator of simulations is the computational execution time: in the case of the technical equipment of the authors, a maximum of 100 ms is required to perform computations in all MPC simulation blocks, as shown in figure 2.
Thus, the presented approach based on decentralized MPCs can be used on modern low-power embedded computers to implement a real swarm UAVs. The work [3] proposes the use of DMPC for the implementation of group control and obstacle avoidance of many multi-rotor UAVs. This study indicates that each individual agent performs a separate subtask which was obtained by the swarm, and at the same time must implement collision avoidance without disrupting the execution of both its own and others’ subtasks. To implement the control system, a simplified mathematical model of a quadcopter is set, according to which each UAV flies slowly enough not to take into account the influence of the environment and changes in air flows from the propellers, the engines respond instantly to control signals, and the yaw angle is not taken into account. Thus, the model is represented by a fourth-order system in the horizontal plane and a second-order system in the vertical axis.

For the simulation, a system of two followers and one leader quadcopter is presented. Three types of simulation are used: using algorithms for specifying the formation based on consensus; determining whether the multi-agent system is capable of moving around obstacles; calculating the execution time of optimizations. As a result of the first simulation, the maintenance of the formation was revealed, however, at the stage of building into the formation, the possibility of a collision is noted due to the non-compliance of the UAV with the condition of the minimum distances between each other. In the second case, a formation with successful collision avoidance is noted, while the third simulation shows results similar to the results of the second, but with smoother and more optimal trajectories. At the same time, the execution time of the optimization stage for the first and second types of simulations turns out to be approximately equal (0.028 and 0.029 seconds, respectively) and differs from the execution time of the third type by 2.3 times. The authors conclude that the resulting architecture can be extended by adding constraints on the vectors of velocities and Euler angles. In [4], the authors propose to combine MPC with feedback linearization. In this work, the emphasis is placed on the formation of a group of quadcopters of the "leader - follower" type. It is noted that in many cases the control of groups is carried out on the basis of linear controllers, in particular, the maintenance of the group of the UAV formation in work [5] is carried out using a set of PI-controllers. Linear controllers turn out to be easy to understand and easy to synthesize; however, they cannot be used directly to work with constraints, and their performance is not guaranteed in a wide range of values, because they are configured to operate within narrow boundaries around the target point. The use of MPC in this work is due to the following advantages of controlling a group of UAVs:

- Limitations of a dynamic system with multiple signal inputs and multiple outputs of the quadcopter are introduced directly into the mathematical model in the controller cycle;
- Kinematics and dynamics of a quadcopter are an integral system, which makes it possible to represent their group as an integrated flight structure;
- Functions for constructing local trajectories are synthesized by combining the prediction of future states with information about the environment in which obstacles may be located.
Obstacle avoidance is based on mixed integer linear programming and combined with MPC [6]. It is noted that collision avoidance restrictions can lead to a collision due to the fact that the width of the obstacle is less than the predicted distance which can be traveled by the agent at the next iteration of the algorithm. MPC introduces the following limitation to prevent collisions of this kind:

\[
\text{norm}(\hat{x}_{k+1} - x_k) \leq H_i^r
\]

where \(\hat{x}_{k+1}\) and \(x_k\) - next predicted and current positions, respectively.

The operation of a controller with a prediction horizon of 10 steps takes on average 0.14 seconds when using a personal computer based on Intel Core Duo. When a single MPC controller is used for trajectory control, the computation time increased to 1.75 seconds.

MPC control can be used as a local path planner [7]. In this case, the MPC control will generate position and speed commands so that the UAV group follows the general desired trajectory that meets these restrictions [8]. The advantage of this approach is the calculation of coordinates and velocities at each time step, depending on the environment and the drone.

Consider the simplest linear model for determining coordinates and velocities:

\[
\ddot{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T
\]

Control vector:

\[
\ddot{u} = [f_x, f_y, f_z, \dot{\psi}_r]^T
\]

Obstacle flight condition:

\[
|x_j - x_k|^2 \geq R^2 \quad j, k = 1, 2, 3
\]

As a result, we get the following formulation of the MPC control problem:

\[
\min_{\Delta u} x_k(k + 1|k) = AX(k - 1|k) + B\Delta \dot{u}(k - 1|k)
\]

\[
\Delta u_{\min} \leq \Delta u(k + 1|k) \leq \Delta u_{\max}
\]

\[
V_{\min} \leq v_{\min}(k + i|k) \leq V_{\max}
\]

Solving this problem using the gradient descent method, we find new coordinates and speeds of the UAV group.

3. Experiments

For the experiment, the UAV Crazyflie 2.1 was used [9]. Communication between the ground station and the UAV was carried out using the Crazyradio PA module, operating in the 2.4 GHz ISM band with a 2 MHz channel width. The UAVs are equipped with an inertial measuring unit, consisting of a three-axis gyroscope, an accelerometer, and a barometer. The Loco Positioning System (LPS) is used which allows to define position of UAV in the space. LPS consists of 8 radio beacons and receiver modules installed on each UAV. The initial positions of the three UAVs are presented in figure 4.
The following graphs describe current and desired trajectory of the leading UAV:

**Figure 4.** Initial coordinates of three UAVs.

**Figure 5.** Graphs of changes in the leader's position over time. Orange lines – desired coordinate, blue – current coordinate.

4. **Conclusion**
In the future, for minimization, the variable gradient method will be used, as specially developed for solving nonlinear optimization problems.

**Reference**
[1] Beni G and Wang J 1993 Swarm Intelligence in Cellular Robotic Systems Springer **102** 703-12
[2] Bassolillo S R 2020 Decentralized mesh-based model predictive control for swarms of UAVs Sensors **20** 4324
[3] Kuriki Y and Namerikawa T 2015 Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control SICE Journal of Control Measurement and System Integration 8 285-94

[4] Zhao W and Go T H 2014 Quadcopter formation flight control combining MPC and robust feedback linearization Journal of the Franklin Institute 351 1335-55

[5] Proud A, Pachter M and D’Azzo J 2001 Tight formation flight control Journal of guidance, control and dynamics 24 246-52

[6] Eren U 2017 Model predictive control in aerospace systems Journal of Guidance, Control and Dynamics 8 285 - 94

[7] D’Amato E, Mattei M and Notaro I 2017 Bi-level flight path planning of UAV formations with collision avoidance Journal of Intelligent & Robotic Systems 93 193-211

[8] Viana I B, dos Santos D A and Góes L C S 2018 Formation control of multirotor aerial vehicles using decentralized MPC Journal of the Brazilian Society of Mechanical Sciences and Engineering 40 1-12

[9] Vieira Lima G, Monteiro Jorge Alves de Souza R, Silva de Morais A, Oliveira-Lopes L C and Mara Vieira Ladeira G 2019 Stabilization and Path Tracking of a Mini Quadrotor Helicopter IEEE Latin America Transactions 17 485-92

[10] Mansouri S S 2019 Remaining useful battery life prediction for UAVs based on machine learning IFAC-PapersOnLine 50 4727-32

[11] Qin S J and Badgwell T A 2003 A survey of industrial model predictive control technology Control engineering practice 11 733-64

[12] Cutler C R and Ramaker B L 1980 Dynamic matrix control a computer control algorithm Joint automatic control conference 1 (San Francisco, California), 72

[13] Clarke D W, Mohtadi C and Tuffs P S 1987 Generalized predictive control - I. The basic algorithm Automatica 23 137-48

[14] Neunert M 2016 Fast nonlinear model predictive control for unified trajectory optimization and tracking IEEE international conference on robotics and automation (ICRA) (Stockholm, Sweden), 1398-404

[15] Lima P F 2017 Spatial model predictive control for smooth and accurate steering of an autonomous truck IEEE Transactions on Intelligent Vehicles 2 238-50