Control and analysis of quadcopter flight when setting a complex trajectory of motion

N Bogatov¹, A Kostin¹ and N Maiorov²*

¹Laboratory of Unmanned Aerial Systems, Saint Petersburg State University of Aerospace Instrumentation, 67 Bolshaya Morskaia str., 190000, Saint-Petersburg, Russian Federation
²Institute of Aerospace Instruments and Systems, Saint Petersburg State University of Aerospace Instrumentation, 67 Bolshaya Morskaia str., 190000, Saint-Petersburg, Russian Federation
*E-mail: nnm@guap.ru

Abstract. The article presents the solution to the problem of autonomous piloting of a quadcopter for a complex flight task to determine the positioning error. Mathematical model is proposed for the equation of the dynamics of angular motion in a connected coordinate system to solve this problem. The stages of elaboration and creation of a flight task are considered, a model based on directed graphs is presented, reflecting various options for organizing the movement of a quadcopter. Clover quadcopters were taken as the object of research. The study was carried out in the airfield of the laboratory of unmanned aerial systems of the Saint Petersburg State University of Aerospace Instrumentation (SUAI). Fragments of the program code for the implementation of autonomous piloting in Python are given. The flight mission consisted in the formation of the letters SUAI in space. The article discusses the necessary hardware, instrumental systems and camera modes of operation for fixing the movement of the quadcopter. As a result of the research, special photographs of the movement of the quadcopter are taken and the permissible range of positioning errors is determined for performing the flight task.

1. Introduction

Today, there is a significant increase in the practical reconciliation of quadcopters for the industry. Over the past five years, the aviation industry has developed a new ecosystem of Aeronet, which includes manufacturers and operators of unmanned aerial systems, software developers, integrators, service providers, industry associations, specialized research laboratories. The main major segments of the Application of the Aeronet [1,2] market are: agriculture, logistics and transport processes, remote analyzing of the land, construction of digital models based on data taken drones, monitoring tasks. The range of tasks and new solutions is constantly expanding now. New practical examples include tasks of inventory, analysis and monitoring of the urban environment, monitoring of passenger flows. Unmanned aerial systems should be applied to the tasks of passenger traffic research [3], or to monitor the waters in port areas, as they allow to obtain dynamic data on the state of complex technical systems. To perform such tasks, it is required to use unmanned aerial vehicles in autonomous mode.

Research in the field of autonomous micro-aerial vehicles (MAVs) is currently divided into several different approaches: One part of the community focuses on improving the accuracy of quadcopter control, and has led to notable advances such as aggressive flight maneuvers, [4,5], ping pong [6] and...
even collaborative construction tasks [7]. While these developments are certainly impressive, such systems nonetheless require external motion capture systems to perform these demanding tasks properly. Another part of the community focuses on outdoor flights where GPS-based pose estimation is possible. Although complete solutions in this field are already available [8], researchers continue to work on improvements in stabilization [9], photogrammetry, obstacle detection and avoidance.

The positioning method using Aruco-marker was chosen to carry out laboratory tests. Having studied the works of authors [10] and [11], we can conclude that the use of such a positioning system is permissible. As noted by the authors of [10] and [11], Aruco-marker makes it possible to obtain a sufficiently high accuracy of a quadcopter flight and a relatively high accuracy of landing.

In this experiment, a special camera will be used to determine the positioning error, thanks to which you can see how accurately the quadcopter moves along the coordinates.

The base for research was a specialized laboratory of unmanned aviation systems of the Engineering School of Saint Petersburg State University of Aerospace Instrumentation (SUAI). The study was based on "Clover" quadcopters from the industrial company Copter Express Technologies LLC [12]. The list of quadcopters of this company is shown in table 1.

| Name                  | Description                                                                                                                                   | Quadcopers                                      |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|
| Copter Express Ltd.   | Copter Express is a developer and manufacturer of unmanned aerial vehicles and software for their autonomy. The company is a market leader in drones for education and use in training in specialties related to the development, operation, management and programming of UAVs. In the interests of industry, Copter Express develops and manufactures UAVs for autonomous territory monitoring, lightweight delivery and other missions. | AL Clover                                      |
|                       |                                                                                                                                               | COEX Clover 4 Code                              |
|                       |                                                                                                                                               | COEX Clover 4 WorldSkills Russia                |
|                       |                                                                                                                                               | COEX Clover 4 Roy Drones                        |
|                       |                                                                                                                                               | Pelican -autonomous quadcopter with thermal imaging for monitoring |
|                       |                                                                                                                                               | COEX Pelican Mini (delivery of parcels)         |

The feature of Clover quadcopters (figure 1) is that it is a training open-source constructor of a programmable quadcopter, which can be used for solving various problems: cargo transfer, FPV-piloting, autonomous piloting.

![Figure 1. Quadcopter Clover 4 [4].](image)

While moving a quadcopter in a complex environment, you plan and build a trajectory in a complex environment.
1. Determining the basic system of coordinates, launch points, flight field boundaries;
2. Understanding the movement of the quadcopter in space, building the perfect model;
3. Incorporating errors related to the positioning of the copter in space and possible deviations in...
motion caused by both aerodynamic properties and possible vibrations of the quadcopter body.

In this case, the identification [12] of the movement of the quadcopter is based on identification in the flight field by Aruco-marker [13,14]. Well-tested solution is to set the trajectory of the reference points and further represent the trajectory in parametric form.

In this work, the quadcopter should draw the acronym "ГУАП (SUAI)", while the operator will not directly control the device, the flight will take place autonomously. In order to see the painted letters LED tape is used, as well as the camera brand Nikon 3320 with long exposure mode.

2. Experimental part
The position of the copter in space is characterized by coordinates $x, y, z$ of center of the mass of the drone in the stationary Cartesian coordinate system and the three corners of the rotation around the axly associated coordinate system [15-18]. In the initial stages of development, you can also consider the independent part of the apparatus a solid body, and the wind is considered only as external disturbance. When performing a flight mission, you must take into account the hovering of the quadcopter at a certain point (about the Aruco-marker). Figure 2 shows the reciprocal position of the bound $(x_b, y_b, z_b)$ and the normal, corresponding marker,$(x_m, y_m, z_m)$of the hovering coordinate systems and the forces and moments acting on the quadcopter.

![Figure 2. Quadcopter coordinates in hang-mode.](image)

On figure 2 $F_1, F_2, F_3, F_4$ are propeller thrust force, $M_1, M_2, M_3, M_4$ are moments from resistance to the screws. The matrix of conversion of the two $R$ coordinate systems is:

$$\begin{align*}
F_b &= \begin{bmatrix} F_{b1} & F_{b2} & F_{b3} \end{bmatrix}^T = [0, 0, L]^T \\
L &= F_1 + F_2 + F_3 + F_4 = b\left(w_1^2 + w_2^2 + w_3^2 + w_4^2\right) \\
b &= \frac{1}{2}\rho \cdot C_D \cdot A_i \cdot R_i^2
\end{align*}$$

(1)

where $L$ is total thrust, $b$ is traction strength ratio, air density, $C_D$ is lifting ratio, $A_i$ is surface propeller area, $R_i$ is propeller radius, $w_i$ is angular rotation speed of $i$-propeller.

The force of the thrust in the marker coordinate system:

$$F_m = \begin{bmatrix} F_{m1} & F_{m2} & F_{m3} \end{bmatrix} = R \cdot F_b = L[I \cdot s_\varphi s_\psi + c_\varphi c_\psi s_\theta + c_\varphi s_\psi s_\theta - c_\varphi s_\theta, c_\varphi c_\theta]$$

(2)

where is
To build a complete system of equations, it is necessary to include the force of gravity and the force of air resistance (4).

\[
\begin{align*}
\begin{bmatrix}
\mathbf{f}_m \\
\mathbf{G}
\end{bmatrix}
= & \begin{bmatrix}
f_x \\
f_y \\
f_z
\end{bmatrix} \\
= & \begin{bmatrix}
f_x \\
f_y \\
f_z
\end{bmatrix}
+ \begin{bmatrix}
0 \\
0 \\
-mg
\end{bmatrix}
\end{align*}
\]

Taking into account the symmetry of the quadcopter and considering that the center of the mass is located at the beginning of the coordinates of the associated system, the equation of angular movement dynamics in the associated coordinate system can be recorded in the form of:

\[
\begin{align*}
\dot{p} &= \frac{(I_{yy} - I_{zz})}{I_{xx}} qr + \frac{L_2}{I_{xx}} \\
\dot{q} &= \frac{(I_{zz} - I_{xx})}{I_{yy}} pr + \frac{L_2}{I_{yy}} \\
\dot{r} &= \frac{(I_{xx} - I_{yy})}{I_z} pq + \frac{L_4}{I_z} \\
L_2 &= M_{mq} + M_{m_1} + M_{wp} \\
L_3 &= M_{mp} + M_{m_1} + M_{wp} \\
L_4 &= M_{mq}
\end{align*}
\]

where \( p = w_x \), \( q = w_y \), \( r = w_z \) – the projection of the vector of the angular speed of the device; \( I_{xx}, I_{yy}, I_{zz} \) – Axis moments of inertia of the apparatus; \( M_{mq}, M_{mp}, M_{mq} \) – moments created by propellers, \( M_{m_1}, M_{m_1}, M_{wp}, M_{wp} \) – gyroscopic moments of engines and propellers.

Based on the presented equations, the equation of the dynamics of the quadcopter is defined. Figure 3 shows the Aruco-markers field in the Unmanned Aerial Systems Laboratory SUAI.

**Figure 3.** Example of Aruco-marker field and clover quadcopter flight in unmanned aerial systems laboratory.
2.1. Methods and materials

Letters “ГУАП” were chosen as a model of the flight mission, which is an abbreviated acronym for the Federal Autonomous Education Institution “St. Petersburg State University of Aerospace Instrumentation (SUAI)”. The quadcopter must fly on the markers and display the letter in space. Copter will fly with one-letter mission. You need to present a model for movement of quadcopter. Different drawings of letters will depend on the given transitions and changes of heights. Let’s present these variants with the graph theory (figure 4). Let's take the letter "Т" as an example.

![Graph of letter T](image)

**Figure 4.** Model of moving a quadcopter to build a letter in space.

In first version of flight and landing were made at different points of the airfield. At the same time, at motion change points, the copter produces a slight hovering and further movement. In second version of fullness implemented flight in which the quadcopter on the route moves both in the straight and in the opposite direction. In second case, a fully ring route is built through a system of defined points.

Similar to the tasks and transport logistics, this model is a model of pendulum routes [19] in the implementation of one-way and two-way traffic with cargo. Let's take a look at the practical implementation of flight missions.

To organize an autonomous flight of the quadcopter, you need to use a flight controller based on the PX4. This can be a Pixhawk, Pixracer and COEX Pix with special software for the flight controller, as well as a single-board Raspberry Pi computer with an RPi Camera (H) Fisheye Lens and a VL53L1X laser rangefinder.

The Clover platform allows a Raspberry Pi computer to be used for programming autonomous flights. The flight program is typically written using the Python programming language. The program may receive telemetry data (which includes battery data, attitude, position, and other parameters) and send commands like: fly to a point in space, set attitude, set angular rates, and others.

The platform utilizes the ROS framework, which allows the user program to communicate with the Clover services that are running as a clover systems daemon. The MAVROS package is used to interact with the flight controller.

MAVROS (MAVLink + ROS) is a ROS package that allows controlling drones via the MAVLink protocol. MAVROS supports PX4 and APM flight stacks. Communication may be established via UART, USB, TCP or UDP.

MAVROS subscribes to certain ROS topics that can be used to send commands, publishes telemetry to other topics, and provides services.

PX4 uses OFFBOARD mode for autonomous flights. OFFBOARD mode uses an external computer
(like a Raspberry Pi). This mode is used in Clover for autonomous flights. The Clover API can be used to transition the drone to this flight mode automatically.

A drone has to use a positioning system to be able to hover still or to fly from point to point. The system should compute the drone position and feed this data into the flight controller. Clover allows using multiple positioning systems, such as optical flow (requires a camera and a rangefinder), fiducial markers (requires a camera and markers), GPS and others. Our experiment uses a positioning system using Aruco-marker. An Aruco-marker is a synthetic square marker composed by a wide black border and a inner binary matrix which determines its identifier (id). Fiducial markers allow the drone to compute its position relative to these markers. This data may then be transferred to the flight controller.

Given an image where some Aruco-marker are visible, the detection process has to return a list of detected markers. Each detected marker includes:
- The position of its four corners in the image (in their original order).
- The id of the marker.

The marker detection process is comprised by two main steps:
- Detection of marker candidates. In this step the image is analyzed in order to find square shapes that are candidates to be markers. It begins with an adaptive thresholding to segment the markers, then contours are extracted from the thresholded image and those that are not convex or do not approximate to a square shape are discarded. Some extra filtering are also applied (removing too small or too big contours, removing contours too close to each other, etc).
- After the candidate detection, it is necessary to determine if they are actually markers by analyzing their inner codification. This step starts by extracting the marker bits of each marker. To do so, first, perspective transformation is applied to obtain the marker in its canonical form. Then, the canonical image is thresholded using Otsu to separate white and black bits. The image is divided in different cells according to the marker size and the border size and the amount of black or white pixels on each cell is counted to determine if it is a white or a black bit. Finally, the bits are analyzed to determine if the marker belongs to the specific dictionary and error correction techniques are employed when necessary.

To accurately determine the position of the drone, you need to calibrate the camera. Camera calibration can significantly improve the quality of nodes related to computer vision: Aruco-marker detection and optical flow.

Camera calibration process allows to define the parameters reflecting the specific lens installed. These parameters include focal lengths, principal point (which depends on camera lens placement regarding the centre), distortion coefficient D. There are several tools allowing to calibrate the camera and store calculated parameters into the system. Usually they use calibration images, "chessboards" or combinations of "chessboards" and Aruco-marker grids (ChArUco). The Clover 4 drone software has already entered the average calibration values for the Raspberry Pi Camera.

3. Results and discussion
To realize autonomous flight, the quadcopter must perform the following tasks:
- Run autonomously and take off at a given altitude;
- Show a specific color display on the address LED tape;
- Follow a given route;
- Make an autonomous landing and disarm of engines.

The experiment will require: a single-payer Computer Raspberry PI, a NIKON camera with the ability to do long exposure, LED tape and Aruco-marker. The single-payer computer is attached to the bottom deck of the quadcopter (figure 5) and connected to the flight controller with a micro-USB to USB cable. Raspberry PI is required for the Python autonomous piloting program and for navigating the drone using machine vision without GPS.
Based on series of experiments with the NIKON 3320 camera, it was found that exposure of up to 30 seconds is enough to capture the trajectory line of the letter.

The quadcopter uses the lower camera to recognize the marker and match the marker ID with the map file of all marks in flight zone.

The code fragment below (figure 6) takes drone off autonomously, hover’s over the Aruco-marker at an altitude of 1 meter, and then flyie’s to the marker with id 83. Aruco-map is pre-recorded in the memory of the quadcopter.

The Clover 4 open-source platform allows to use the Raspberry Pi to program autonomous drone flight. Most often, the program for autonomous flight is written on the Python. Program can receive telemetry (battery charge, orientation, location, etc.) and send commands: fly to the point, set the oridirection, angular speed and other functions.

In order to indicate coordinates of the point on which the drone should move, the navigate service is used.

Options:
- x, y, z - coordinates (m);
- yaw - angle on the trotting (radian);
- yaw_rate - angular speed on prowling (used when installing yaw in NaN) (rad/s);
- speed - flight speed (setpoint speed) (m/s);
- auto_arm - to transfer the copter to OFFBOARD and to arse automatically (copter will take off);
frame_id is a coordinate system that sets x, y, z and yaw.

To navigate quadcopter during takeoff, landing and flight mission is used frame_id - the coordinate system for the values x, y, z, v_x, v_y, v_z. For take-off, the body coordinate system is used - coordinates relative to the quadcopter without taking into account the slopes on aruco_N the pitch and roll.

3.1. Flight results and analysis

Letters "ГУАП" were chosen as flight mission. Results of copter’s flight are shown on the figure 7.

![Figure 7. Results of the flight of the quadcopter during the flight mission.](image)

The result of the analysis to determine errors of the positioning of copter is given on the figure 8.

![Figure 8. Flight mission study to determine positioning error.](image)
Examples of feature graphs presented in figure 8 were built for each letter.

4. Conclusion
To test the proposed approach, experiments were carried out to move the quadcopter from the initial position with access to the trajectory. The movement of the quadcopter was carried out along a bit-linear trajectory. The quadcopter's positioning error relative to the desired trajectory in the analysis of autonomous flight by all four letters in the laboratory of unmanned aerial systems was 0.1 to 0.2 m. Within the framework of the results obtained, it can be concluded that the positioning of the quadcopter is sufficiently accurate in space. Positioning error is due to the average calibration values of the Raspberry Pi camera, as well as the data from the flight controller sensors. To improve positioning accuracy, it is worth checking the accuracy of applying Aruco-marker to the surface, rechecking the camera and flight controller calibration, and choosing the PID controller values for this quadcopter.

References
[1] AERONET. Association of Operators and Developers of Unmanned Aircraft Systems, available at: https://aeronet.aero
[2] Kostin A S and Bogatov N V 2019 Issues of modern development of the market of unmanned aerial vehicles. System analysis and logistics 4(22) 65
[3] Krile S, Maiorov N N and Fetisov V A 2018 Forecasting the operational activities of the sea passenger terminal using intelligent technologies. Transport Problems. 13(1) 27 doi: 10.21307/tp.2018.13.1.3
[4] Kownacki C and Ambroziak L 2017 Local and asymmetrical potential field approach to leader tracking problem in rigid formations of fixed-wing UAVs. Aerosp. Sci. Tech. 68 465 doi: 10.1016/j.ast.2017.05.040
[5] Kownacki C and Ambroziak L 2017 Flexible structure control scheme of UAVs formation to improve the formation stability during maneuvers. Acta Mech. Automat. 11(3) 178 doi: 10.1515/ama-2017-0026
[6] Gosiewski Z and Ambroziak L 2013 UAV autonomous formation flight experiment with virtual leader control structure. Solid State Phenom. 198 254 doi: 10.4028/www.scientific.net/SSP.198.254
[7] Ambroziak L and Gosiewski Z 2013 Preliminary UAV autopilot integration and in-flight testing. Solid State Phenom. 198 232 doi: 10.4028/www.scientific.net/SSP.198.232
[8] Kondratniuk M and Ambroziak L 2016 Concept of the magnetic launcher for medium class unmanned aerial vehicles designed on the basis of numerical calculations. J. Theor. Appl. Mech. 54(1) 163 doi: 10.15632/jtam-pl.54.1.163
[9] Kondratniuk M and Gosiewski Z 2013 Comparison of simulated and measured signals of the electromagnetic coil launcher for micro aerial vehicles, 2013 International Symposium on Electrodynamic and Mechatronic Systems (SELM) p 57 doi: 10.1109/SELM.2013.6562978
[10] Bożko A, Kamiński A, Ambroziak L and Kondratniuk M 2018 Autonomous multirotor landing with the application of vision systems. AIP Conference Proceedings 2029 020007 doi: 10.1063/1.5066469
[11] Bacik J, Durovsky F, Fedor P and Perdukova D 2017 Autonomous flying with quadcopter using fuzzy control and Aruco-marker. Inte1. Serv. Robotics 10 185 doi: 10.1007/s11370-017-0219-8
[12] Copter Express LLC, COEX, available at: https://ru.coex.tech
[13] Aruco: a minimal library for augmented reality applications based on opencv, Universidad de Cordoba, available at: http://devres.zoomquiet.top/data/20170809231640/index.html
[14] Aruco-marker generator, available at: https://chev.me/arucogen/
[15] Kim J, Kang M S and Park S 2010 Accurate modeling and robust hovering control for a quadrotor VTOL aircraft. J. Intell. Robot. Syst. 57 9 doi: 10.1007/s10846-009-9369-z
[16] Hoffmann G M, Haomiao Huang, Wasland S L and Tomlin C J 2007 Quadrotor helicopter flight
dynamics and control: theory and experiment. *Proc. of the AIAA Guidance, Navigation, and Control Conference* AIAA 2007-6461 doi: 10.2514/6.2007-6461

[17] Engel J, Sturm J and Cremers D 2012 Accurate figure flying with a quadcopter using onboard visual and inertial sensing. *Proc. of the Workshop on Visual Control of Mobile Robots (ViCoMoR) at the IEEE/RJS Intern. Conference on Intelligent Robot Systems (IROS)* (Portugal: Vilamoura, Algarve) 320 43

[18] Mellinger D and Kumar V 2011 Minimum snap trajectory generation and control for quadrotors. *Proc. of the IEEE Intern. Conference on Robotics and Automation* (China: Shanghai) p 2520 doi: 10.1109/ICRA.2011.5980409

[19] Egerstedt M, Hu X and Stotsky A 2001 Control of mobile platforms using a virtual vehicle approach. *IEEE Trans. Automat. Contr.* 46(11) 1777 doi: 10.1109/9.964690