Simulation of Quadcopter's Complex Spatial Motion in Virtual Environment Systems[*]

Evgeny Strashnov[1] [0000-0003-0937-4052], and Mikhail Mikhaylyuk[2] [0000-0002-7793-080X]

Federal State Institution «Scientific Research Institute for System Analysis of the Russian Academy of Sciences», Moscow, Russia
[1] strashnov_evg@mail.ru, [2] mix@niisi.ras.ru

Abstract. The paper considers the task of simulation quadcopter motion control in virtual environment systems. The proposed solution of this task is based on the application of a mathematical model of quadcopter dynamics, in which quaternion and axis-angle representation of the attitude are used. In this work, a feedback linearization method for the equations of quadcopter and its electric motors dynamics is proposed to control the quadcopter. With the application of this approach, quadcopter motion is achieved by the obtained expressions for linear acceleration and the desired quaternion. The proposed methods and approaches for quadcopter simulation were implemented in the software package of virtual environment and tested using examples of the motion of a quadcopter model along a given path.

Keywords: Quadcopter, Simulation, Quaternion, Axis-Angle Representation, Feedback Linearization Method, Virtual Environment System.

1   Introduction

Quadcopter is a special case of an unmanned aerial vehicle with four rotors. Due to the relative simplicity of the construction, such devices are widely used in various fields where it is necessary to carry out aerial photography and video, aerial reconnaissance, cargo delivery, etc. The quadcopter control is performed by changing the rotation speed of the rotors, which allows to realize its vertical take-off, landing and maneuver in the horizontal plane.

The current area of research is the use of virtual environment systems for quadcopter dynamics simulation and control. In this paper, we consider the task of virtual quadcopter model motion along a given arbitrary trajectory. The complexity of this

task consists in the fact that the quadcopter is an under-actuated mechanical system (the control number is less than the number of degrees of freedom). This leads to the fact that the translational and rotational motion of the quadcopter are interconnected. At the same time, for the implementation of complex motions (such as flips), it is necessary to provide arbitrary orientation of the quadcopter.

In recent years several strategies, such as [1-5], based on Euler angles are used to control the rotational motion of the quadcopter. However, this approach leads to the fact that it is impossible to uniquely restore the Euler angles (so-called "gimbal lock" problem) for certain orientations of the quadcopter. Therefore, an alternative solution is that the quaternion can be used to the quadcopter dynamics and control [6-8].

In this work, quadcopter simulation is realized in the virtual environment system with using an approach in which the quaternion and axis-angle representation of the quadcopter attitude are used. The quadcopter control is performed by the feedback linearization method for the equations of quadcopter and its electric motor dynamics. Within this method, simulation of quadcopter motion along a given trajectory is performed by providing relationships in which linear acceleration is associated with the quaternion. Approbation of proposed solution was carried out in the virtual environment software package using the example when quadcopter moving along a complex acrobatic figure.

2   Quadcopter Dynamics

The quadcopter is an under-actuated system, since the number of actuators (four rotors) is less than the number of its degrees of freedom (six degrees of freedom for a rigid body). Figure 1 shows a model of a quadcopter with a structure of type "X", according to the arrangement of the rotors. To describe the motion of the quadcopter, we introduce the following coordinate systems: the inertial frame E_I, the body-fixed frame E_B, and the desired frame E_D. The quadcopter's position is defined with radius vector $\mathbf{r} = \mathbf{OO}_B = (x, y, z)^\mathrm{T}$, and the attitude is defined with unit quaternion $\mathbf{q} = (q_w, q_x, q_y, q_z)^\mathrm{T}$, where $\|\mathbf{q}\| = \sqrt{q_w^2 + q_x^2 + q_y^2 + q_z^2} = 1$.

To obtain a linear model of quadcopter motion, consider the logarithmic mapping [6] of a quaternion $\mathbf{q}$:

$$\ln(\mathbf{q}) = \begin{cases} \dfrac{\mathbf{q}_v}{\|\mathbf{q}_v\|} \arccos(q_w), & \|\mathbf{q}_v\| \neq 0; \\ 0, & \|\mathbf{q}_v\| = 0, \end{cases} \tag{1}$$

where $\mathbf{q}_v = (q_x, q_y, q_z)^\mathrm{T}$ is the vector part of quaternion.

Then the vector $\boldsymbol{\alpha} = 2\ln(\mathbf{q})$ defines the rotation of the quadcopter around the unit axis $\boldsymbol{\alpha} / \|\boldsymbol{\alpha}\|$ by an angle $\|\boldsymbol{\alpha}\|$. In this case, the kinematics of the quadcopter is defined using the linear equations

$$\dot{\mathbf{r}} = \mathbf{v} \,; \quad \dot{\boldsymbol{\alpha}} = \boldsymbol{\omega} \,, \tag{2}$$

where $\mathbf{v} = \left(v_x, v_y, v_z \right)^{\mathrm{T}}$ denotes the quadcopter's linear velocity in the inertial frame E_I, $\boldsymbol{\omega} = \left(\omega_x, \omega_y, \omega_z \right)^{\mathrm{T}}$ the quadcopter's angular velocity in the body-fixed frame E_B.
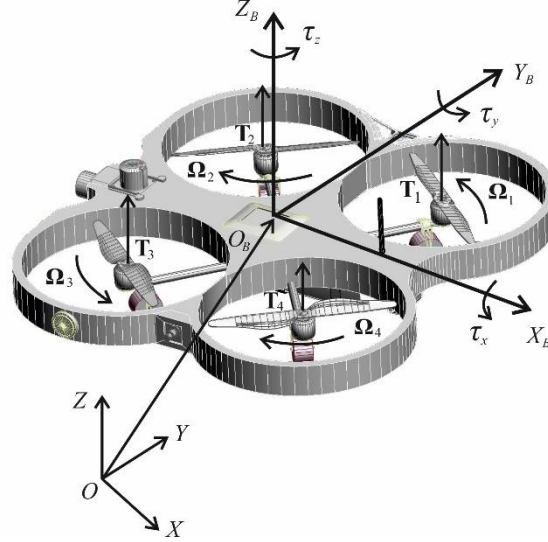

Fig. 1. Virtual quadcopter model.

Rotation of rotors create thrusts $T_i = k\Omega_i^2$, $i = \overline{1,4}$, applied in the attachment points and are directed parallel to the axis $O_B Z_B$ of the body-fixed frame E_B, where Ω_i and k denotes, respectively, angular velocity of rotor i and the lift constant. Also, opposite the rotor rotation there are torques $\tau_i = b\Omega_i^2$, where b denotes the drag constant of rotor. Hence, the four rotors create total thrust $F = \sum T_i$ and torque $\boldsymbol{\tau} = \left(\tau_x, \tau_y, \tau_z \right)^{\mathrm{T}}$, which are computed as

$$F = k(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) ; \quad \tau_x = kl_y (\Omega_1^2 + \Omega_2^2 - \Omega_3^2 - \Omega_4^2) ;$$
$$\tau_y = kl_x (-\Omega_1^2 + \Omega_2^2 + \Omega_3^2 - \Omega_4^2) ; \quad \tau_z = b(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) , \tag{3}$$

where l_x and l_y are the rotor coordinate's positions in the body-fixed frame E_B.

The mathematical model of quadcopter's motion is described as the Newton-Euler equations, which, taking into account the kinematic relations (2), become

$$m\ddot{\mathbf{r}} = R(\mathbf{q}) \begin{pmatrix} 0 \\ 0 \\ F \end{pmatrix} + m \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} ; \tag{4}$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} + \boldsymbol{\tau} , \tag{5}$$

where m is the quadcopter mass, $\mathbf{I}$ the inertia matrix in the body-fixed frame E_B, $R(\mathbf{q})$ the quaternion rotation matrix, g the gravity acceleration.

Formulas (4) and (5) are supplemented by the rotor dynamics equations [5] based on their passport parameters:

$$I_m\dot{\Omega}_i = M_S(U_i - \frac{\Omega_i}{\Omega_{idle}}), \ i = \overline{1,4}, \tag{6}$$

where U_i denotes the voltage supplied to armature of the motor, I_m the motor inertia, M_S the motor starting torque, Ω_{idle} the motor idling speed.

Equations (4), (5) and (6) define a mathematical model of quadcopter and rotor dynamics, where control variables F, τ_x, τ_y and τ_z are given by formulas (3).

3　　Quadcopter Control

To control the quadcopter, we apply the feedback linearization method for equations (4) - (6). This method consists in the fact that the control variables are selected in a such way that the closed-loop system of quadcopter motion equations will be linear with given characteristics. The control task is to compute the thrust force F and torques $\boldsymbol{\tau} = \left(\tau_x, \tau_y, \tau_z\right)^{\mathrm{T}}$ in order to realize translational and rotational motion of the quadcopter. These control variables are used to solve the system of equations (3) with respect to Ω_i. The obtained solution is then used to calculate the voltages U_i supplied to the rotor electric motors.

3.1　　Attitude Control

Here we will consider a more convenient and intuitive approach in which the desired attitude of the quadcopter is specified using Euler angles. Let the desired frame E_D be obtained relative to the inertial frame E_I by a sequence of rotations, where first performed around Z-axis on the jaw angle ψ_d, then around Y-axis on the roll angle θ_d and finally around X-axis on the pitch angle φ_d. With this sequence of rotations, the desired quaternion is derived through the Euler angles as follows [8]:

$$\mathbf{q}_d = \begin{bmatrix} \cos(\varphi_d/2)\cos(\theta_d/2)\cos(\psi_d/2) + \sin(\varphi_d/2)\sin(\theta_d/2)\sin(\psi_d/2) \\ \sin(\varphi_d/2)\cos(\theta_d/2)\cos(\psi_d/2) - \cos(\varphi_d/2)\sin(\theta_d/2)\sin(\psi_d/2) \\ \cos(\varphi_d/2)\sin(\theta_d/2)\cos(\psi_d/2) + \sin(\varphi_d/2)\cos(\theta_d/2)\sin(\psi_d/2) \\ \cos(\varphi_d/2)\cos(\theta_d/2)\sin(\psi_d/2) - \sin(\varphi_d/2)\sin(\theta_d/2)\cos(\psi_d/2) \end{bmatrix}. \tag{7}$$

Then the desired attitude is defined using the vector $\boldsymbol{\alpha}_d = \ln(\mathbf{q}_d)$, which is given by the formula (1).

Let's transform equation (5) of quadcopter's rotation motion to the linear form. To do this, we select torque $\boldsymbol{\tau} = \left(\tau_x, \tau_y, \tau_z\right)^{\mathrm{T}}$ as

$$\boldsymbol{\tau} = \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} - K_{pr}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_d) - K_{vr}(\boldsymbol{\omega} - \boldsymbol{\omega}_d), \tag{8}$$

where $K_{pr} > 0$ and $K_{vr} > 0$ are the diagonal matrices with positive linearization coef-

ficients.

The control law (8) leads to the fact that the system of equations (5) becomes linear and $\boldsymbol{\alpha}(t)$ tends to $\boldsymbol{\alpha}_d(t)$ as $t \to \infty$.

3.2 Trajectory Control

This task is to ensure the motion of the quadcopter along a given path $\mathbf{r}_d(t)$. To solve this problem, we apply the feedback linearization method for equations (4) so that linear accelerations $\mathbf{a} = \ddot{\mathbf{r}} = \left(a_x, a_y, a_z\right)^{\mathrm{T}}$ satisfy the following equation

$$\mathbf{a} = -K_{pt}(\mathbf{r} - \mathbf{r}_d) - K_{vt}(\dot{\mathbf{r}} - \dot{\mathbf{r}}_d),\tag{9}$$

where $K_{pt} > 0$ and $K_{vt} > 0$ are the diagonal matrices with positive linearization coefficients.

Expression (9) is used to compute the thrust F and the desired angles φ_d and θ_d, which are involved in the implementation of the horizontal motion of the quadcopter. After substituting (9) into (4), we obtain that the thrust F is expressed through accelerations $\mathbf{a}$ as

$$F = m\sqrt{a_x^2 + a_y^2 + (a_z + g)^2}.\tag{10}$$

In turn, the angles φ_d and θ_d are derived from (4) by expressing the matrix $\mathbf{R}$ in terms of Euler angles. Applying some transformations in (4), we can show [5] that the following relations are valid:

$$\varphi_d = \arcsin\left(\frac{a_x \sin(\psi_d) - a_y \cos(\psi_d)}{\sqrt{a_x^2 + a_y^2 + (a_z + g)^2}}\right);\tag{11}$$

$$\theta_d = \arctan\left(\frac{a_x \cos(\psi_d) + a_y \sin(\psi_d)}{a_z + g}\right).\tag{12}$$

Formulas (11) and (12) express the relationship of rotational and translational motion of the quadcopter. The angles φ_d and θ_d are provided using the torque are given by the formula (8). Thus, when selecting the thrust F and angles φ_d and θ_d, we obtain accelerations (9) that make equations (4) linear such that $\mathbf{r}(t)$ tends to $\mathbf{r}_d(t)$ as $t \to \infty$.

3.3 Actuator Control

Solving the system of equations (3), we obtain the values $\Omega_{i,d}$ of rotor speeds. To ensure these speeds, we apply the feedback linearization method for equations (6). Selecting voltage U_i in the following form

$$U_i = \frac{1}{\Omega_{idle}}\left((1 - t_c k_{m,i})\Omega_i + t_c k_{m,i}\Omega_{i,d}\right),\tag{13}$$

we get that equations (6) become linear such that Ω_i tends to $\Omega_{i,d}$ as $t \to \infty$. When deriving (13), the relation $I_m = M_s t_c / \Omega_{idle}$ was used, where t_c is the motor time constant.
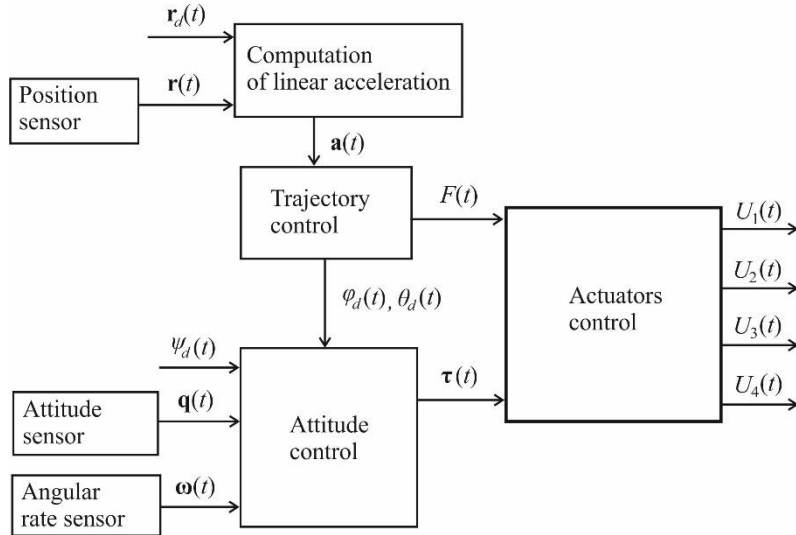

Fig. 2. Structure of the quadcopter control scheme.

4   Simulation Results

The quadcopter simulation was carried out in the virtual environment system, which was developed at the SRISA RAS. This software package includes parts for control, dynamics and visualization of virtual objects. The quadcopter control algorithm is defined by constructing a block diagram scheme. In the control subsystem, this scheme is computed and at the output of which we obtain new values of the voltages supplied to the rotor motors. These voltages are sent to the dynamics subsystem, which computes the new coordinates of the quadcopter and the sensors readings. The resulting coordinates are sent to the visualization subsystem that renders the virtual scene.

The coefficients of the feedback linearization method used were the following: $K_{pr,i} = 100$, $K_{vr,i} = 20$, $K_{pt,i} = 0.2$, $K_{vt,i} = 0.75$, $i = \overline{1,3}$. The quadcopter control scheme is conventionally shown in Figure 2. The scheme inputs are the desired path $\mathbf{r}_d(t)$, the yaw angle $\psi_d(t)$, as well as the virtual sensors readings (position, orientation and angular velocities). According to this scheme, in order to implement the quadcopter trajectory control, linear accelerations are first computed by formula (9). These accelerations are used to compute the thrust F, the roll θ_d and pitch φ_d angles

by formulas (10)-(12). The angles are sent to the block, which implements control of the quadcopter attitude with the computation of the output torque $\boldsymbol{\tau}$. The obtained thrust F and torque $\boldsymbol{\tau}$ are used to compute the rotor speeds $\Omega_{i,d}$ and the voltages U_i supplied to the quadcopter actuators.

Fig. 3. Quadcopter motion around the robot.

Two examples were considered for simulation of the quadcopter motion. In the first example (see Fig. 3), a task is realized in a virtual scene in which the quadcopter flies around an object (MF-4 robot) from different sides.

The path $\mathbf{r}_d(t)$ of the quadcopter motion is defined as

$$x_d(t) = C_x + R\cos\left(\frac{2\pi t}{T}\right); \ \ y_d(t) = C_y + R\sin\left(\frac{2\pi t}{T}\right); \ \ z_d(t) = H = const \ ,$$

where $t \in [0,T]$, C_x and C_y are the object coordinates, H the flight altitude, T the total flight time. At the same time, the yaw angle changes according to the $\psi_d(t) = 2\pi t / T$ so that the virtual camera is directed towards the robot.

Figure 3 shows the different positions of the quadcopter during its motion, where the position with number 1 corresponds to a point in time $t = 0$, with number 2 corresponds to $t = T / 6$, etc.

In the second example (see Fig. 4), quadcopter performs a complex aerobatics figure in the form of a closed loop (corkscrew loop). The quadcopter motion is performed in its longitudinal plane and includes two stages. At the first stage, the quadcopter is accelerated to the required velocity along the Y axis of its body-fixed frame. At the second stage, the quadcopter performs a total rotation around the pitch axis. To implement this motion, the desired pitch angle is defined as

$$\varphi_d(t) = \begin{cases} -\pi/9, & 0 \le t < T_1 \\ 2\pi t/(T_2 - T_1), & T_1 \le t \le T_2 \end{cases},$$

where $t \in [0, T_2]$, T_1 is the time of acceleration, T_2 the time of the flip.
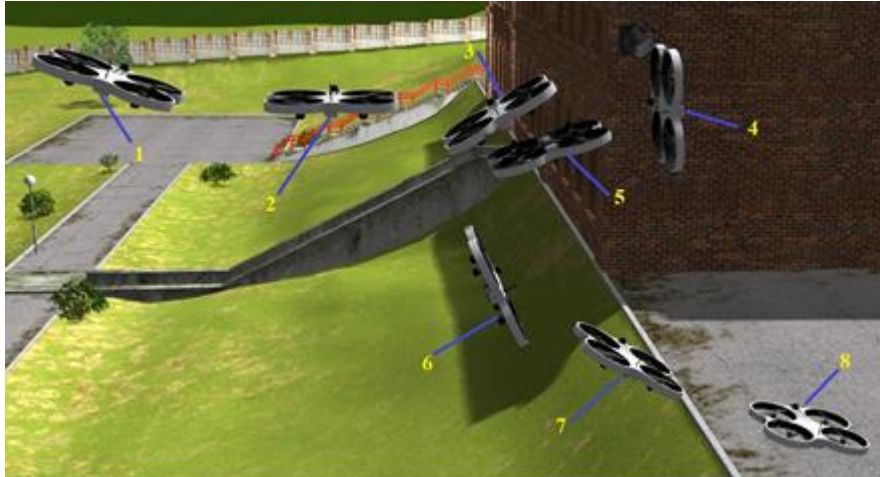

Fig. 4. Quadcopter motion along a complex path.

Figure 4 shows the different positions of the quadcopter when performing a cork-screw loop. The numbers indicate the sequential positions of the quadcopter, where numbers 1 and 2 corresponds to the quadcopter positions during acceleration, and numbers from 3 to 8 corresponds to the positions when the quadcopter performs a flip. Note that the position with number 4 corresponds to the quadcopter rotation on the pitch angle about of 90 degrees around the X axis. With such a rotation, it is impossible to uniquely restore the quadcopter attitude, if Euler angles are used.

5   Conclusion

In this paper, methods and approaches for quadcopter dynamics simulation and control in virtual environment systems were developed. The proposed solution is based on a mathematical model of the quadcopter's dynamics, in which the quaternion and the axis-angle representation are used to define the quadcopter attitude. Testing the examples of a virtual model showed that the proposed methods and approaches make it possible to realize the trajectory motion of a quadcopter with any angle of rotation. It is assumed that the obtained results can be used to control a real quadcopter model. For this in the future, it is planned to improve the developed methods and approaches for quadcopter simulation taking into account external disturbances, wind and noise in the virtual sensors readings.

References

1. I.C. Dickmen, A. Arisoy, and H. Temeltas: Attitude control of quadrotor. In: 4th International Conference on Recent Advances in Space Technologies, pp. 722-727 (2009).
2. Z. Zuo: Trajectory tracking control design with command-filtered compensation for a quadrotor. IET Control Theory Appl., **4**(11), 2343-2355 (2010).
3. T. Luukkonen: Modeling and control of quadcopter. Independent research project in applied mathematics, Espoo: Aalto University (2011).
4. F. Sabatino: Quadrotor control: modeling, nonlinear control design and simulation. Master's Degree Project, Stockholm, Sweden (2015).
5. M.V. Mikhaylyuk, and E.V. Strashnov: Simulation of quadcopter motion control in virtual environment systems. Mathematica Montisnigri, **XLIV**, 60-72 (2019).
6. J. Carino, H. Abaunza, and P. Castillo: Quadrotor quaternion control. In: International Conference on Unmanned Aircraft Systems (ICUAS), Denver, USA (2015).
7. A. Chovancova, T. Fico, P. Hubinsky, and F. Duchon: Comprasion of various quaternion based control methods applied to quadrotor with disturbance observer and position estimator. In: Robotics and Autonomous Systems (2016).
8. E. Fresk, and G. Nikolakopoulos: Full quaternion based attitude control for a quadrotor, In: Proc. ECC, Zurich, CH, pp. 3864-3869 (2013).