DOI 10.70286/EOSS-10.11.2025.009

MATLAB-BASED QUATERNION SIMULATOR FOR FPV DRONE DYNAMICS

Serhii S. Nikulin
Postgraduate Student
Oleksandr V. Zabolotnyi
Doctor of Technical Science, Professor
Intelligent Instrumentation Systems and Quality Management Department
National Aerospace University “Kharkiv Aviation Institute”
Kharkiv, Ukraine

Abstract. This paper presents the development of an interactive MATLAB-based software simulator designed for modeling and visualizing the rotational dynamics of FPV (First Person View) drones. The simulator employs a nonlinear mathematical model that includes the effects of motor thrust, gyroscopic coupling, reactive torque, and aerodynamic damping. To ensure robust numerical stability and eliminate singularities inherent in Euler-angle parameterization, the system uses quaternion-based orientation platform integrated with a fourth-order method of Runge–Kutta. A distinctive feature of the simulator is the inclusion of the inertial influence of a three-axis gimbal used as a test stand, which allows accurate replication of laboratory Hardware-in-the-Loop (HIL) experiments. The graphical user interface (GUI) enables real-time interaction, parameter configuration and visualization of orientation angles, angular velocities and thrust dynamics.

Keywords: FPV drone, quaternion dynamics, MATLAB GUI, Runge–Kutta method, three-axis gimbal, interactive simulation.

Introduction. FPV drones represent a class of highly maneuverable unmanned aerial vehicles (UAVs) that are widely used in military and special missions. Their nonlinear dynamic behavior and strong coupling between control channels make the design and testing of stabilization systems a challenging engineering task. Real-world experiments often involve high risk of equipment damage and limited ability to reproduce extreme flight modes. For these reasons, accurate and physically grounded simulation becomes a vital tool in the research and design of control systems. A realistic mathematical model allows algorithms testing, physical parameters optimization, and safe validation of control responses.

Aims and Objectives. This study focuses on developing a high-fidelity simulation environment that not only represents the rotational dynamics of an FPV quadcopter but also accounts for the inertia of the gimbal frame on which real drones are tested. Such integration creates a bridge between purely virtual models and physical experiments, contributing to the development of hybrid Hardware-in-the-Loop (HIL) systems for UAV research.

Methodology. The FPV drone is treated as a rigid body with distributed masses. The rotational motion around the center of mass is described by Euler’s equation:

$$I\dot{\omega} + \omega \times (I\omega) = \tau \quad (1)$$

where I is the inertia tensor, ω is the body angular velocity vector, and τ is the resultant torque vector produced by the propulsion system, aerodynamic damping, and gyroscopic effects.

To overcome the singularity issues of Euler angles, the drone's attitude is expressed through unit quaternions $q = [q_0, q_1, q_2, q_3]$. The quaternion derivative is given by:

$$\dot{q} = \frac{1}{2}\Omega(\omega)q \quad (2)$$

where $\Omega(\omega)$is the angular velocity matrix. This representation ensures continuity and accuracy even under large-angle rotations and high angular speeds typical for FPV acrobatic maneuvers.

The total torque τ is composed of four parts:

a) thrust torque (τ_T) generated by rotor thrusts applied at lever arms l_i;

b) reactive torque (τ_R) due to the resistance to propeller rotation, using torque coefficient k_M;

c) gyroscopic torque (τ_G) arising from precession effects of rotating propellers;

d) damping torque (τ_D) representing aerodynamic drag proportional to angular velocity.

e) the combined torque defines the instantaneous rotational acceleration.

Unlike conventional UAV models, the proposed system incorporates the inertia of a three-axis gimbal used in laboratory testing. The total inertia tensor is expanded as:

$$I_{total} = I_{drone} + I_{gimbal} \quad (3)$$

where I_{gimbal} includes the moments of inertia of roll, pitch, and yaw frames. Each frame's inertia is computed analytically from its geometry and mass, allowing accurate modeling of the coupled drone–gimbal dynamics. The quaternion and angular velocity equations are integrated using the fourth-order Runge–Kutta method (RK4). This method provides both stability and precision required for stiff nonlinear systems. The quaternion is normalized at each iteration to maintain unit magnitude, ensuring accurate orientation propagation.

The simulator is implemented as a MATLAB script built around a finite-state machine structure. It consists of the following core modules:

a) initialization module: defines drone and gimbal physical parameters (mass, geometry, motor coordinates, inertia tensor);

b) torque computation module: calculates total moments based on motor thrusts and angular velocities;

c) RK4 integrator module: performs quaternion and angular velocity updates;

d) GUI module (App Designer): provides interactive sliders, control buttons, and real-time plots (fig. 1);

e) visualization module: displays Euler angles, angular rates, and motor thrusts dynamically.

The user can choose between two simulation modes: manual control – direct adjustment of motor thrusts via sliders; preset scenario – automated thrust profiles for preprogrammed maneuvers such as roll or pitch tests.

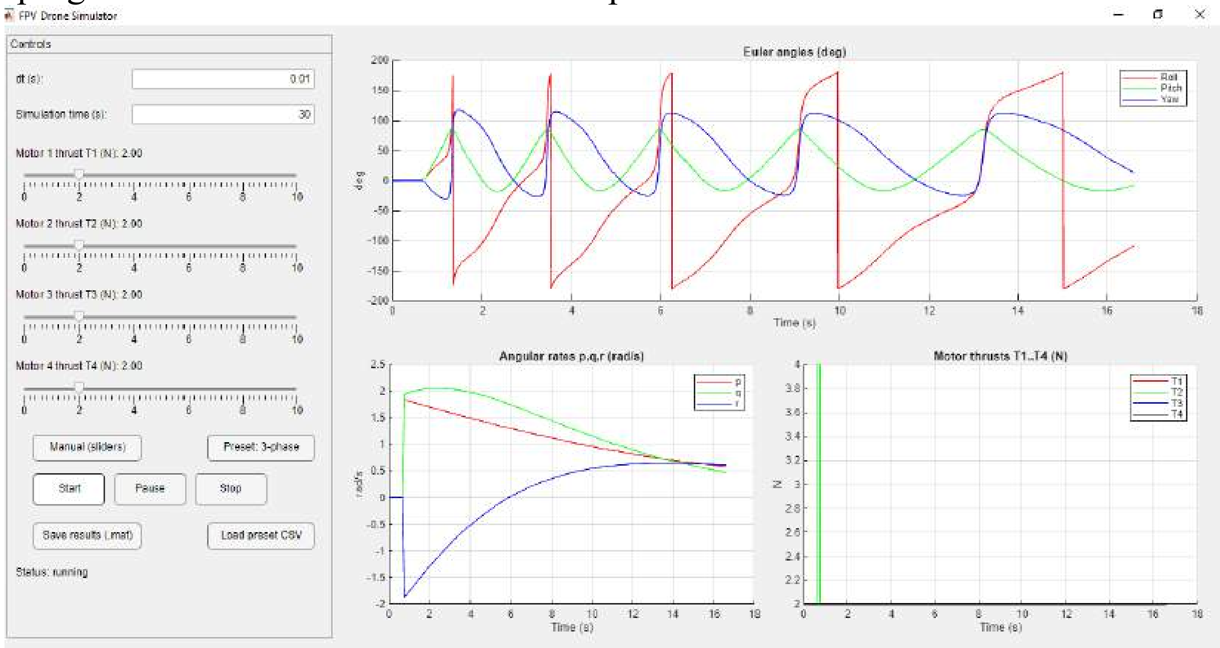
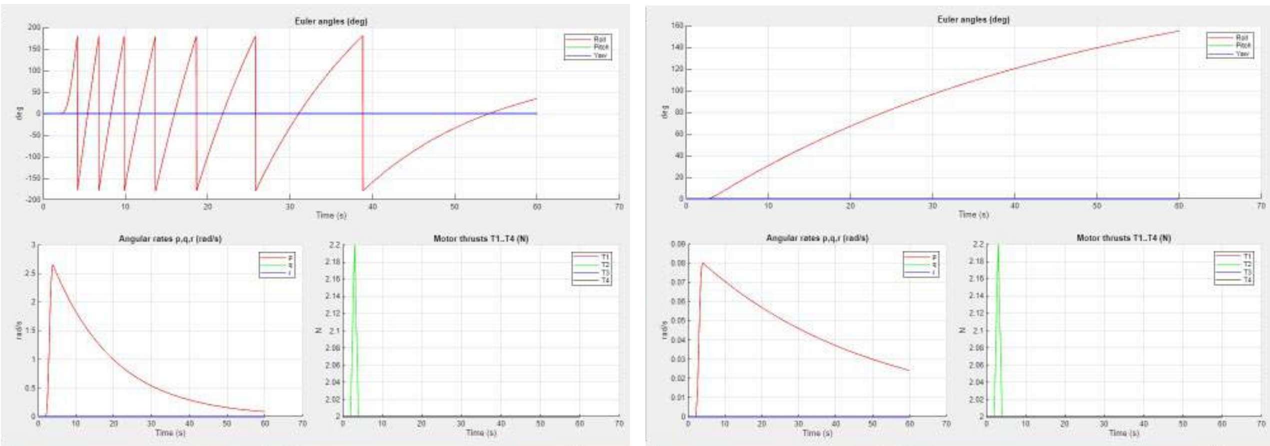


Figure 1 – FPV drone simulation module interface

All physical constants are parameterized, allowing easy adjustment of drone configuration (mass distribution, lever arms, battery mass, damping factors) to match experimental data.

Simulation Results and Discussion. The simulation was validated using real FPV drone parameters and gimbal geometry obtained from a laboratory test stand.

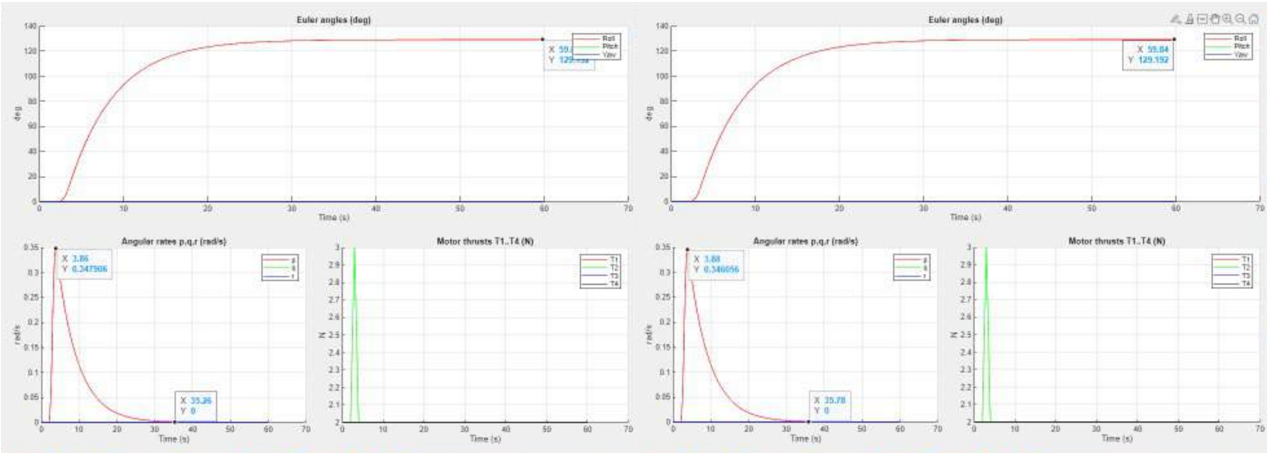
Including gimbal frame inertia significantly affects the system's dynamic response (fig. 2). Without gimbal inertia, the roll response exhibits damped oscillations after control input removal. When gimbal inertia is considered, the response becomes monotonic with higher damping – indicating that the gimbal acts as an inertial stabilizer suppressing short-term oscillations.



a) b)

Figure 2 – Drone response to stepwise action on roll angles, a) without taking into account the inertial moments of the gimbal frames, b) taking into account the inertial moments of the gimbal frames

Increasing battery mass results in higher total inertia, reducing angular acceleration and prolonging stabilization time (fig. 3). Conversely, increasing motor arm length enhances the torque arm, yielding faster roll responses and improved maneuverability. The model accurately reproduces these effects, confirming its physical adequacy.



a) b)

Figure 3 – Change in the system's response to roll control inputs when the battery mass is decreased from 1.320 grams (a) to 0.910 grams (b)

The presence of gyroscopic torques leads to observable cross-coupling between control channels – for instance, rapid pitch commands may induce minor yaw deviations (fig. 4). The RK4 quaternion integrator remains numerically stable even under such fast-changing dynamics, maintaining accurate orientation tracking throughout the simulation.

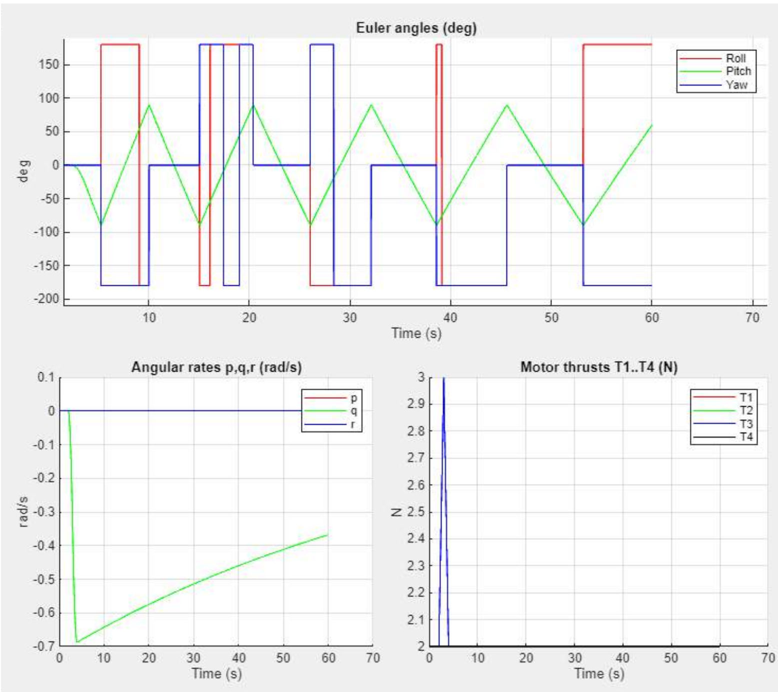


Figure 4 – Manifestation of cross-couplings in the system during a sharp change in the pitch control input

Conclusions. The developed MATLAB-based quaternion simulator provides a high-fidelity virtual environment for studying FPV drone rotational dynamics. Its main advantages include:

a) accurate nonlinear modeling of thrust, gyroscopic, and damping effects;

b) quaternion-based orientation free from singularities;

c) real-time interactive GUI with adjustable physical parameters;

d) inclusion of gimbal inertia enabling realistic HIL test emulation.

The simulator serves as an effective tool for engineering education, research on control algorithms, and pre-flight validation of FPV drone behavior. Future work will focus on integrating the simulator with physical IMU sensors and Arduino-based flight controllers to create a complete Hardware-in-the-Loop platform.

References

1. Zabolotnyi, O. V., & Nikulin, S. S. (2024). Development of a hardware–software complex for combat quadcopters testing. Scientific Notes of Taurida National V. I. Vernadsky University, Series: Technical Sciences, (4), 83–90. https://doi.org/10.32782/2663-5941/2024.4/13
2. Liu, Y. (2024). Modeling and simulation of quadcopter UAV based on PID control. Applied Computational Engineering, 75(1), 202–210. https://doi.org/10.54254/2755-2721/75/20240539
3. Saif, E., & Eminoğlu, İ. (2022). Modeling of quad-rotor dynamics and hardware-in-the-loop simulation. Journal of Engineering, 2022(8), 1–9. https://doi.org/10.1049/tje2.12152
4. Sveier, A. M., & Egeland, O. (2019). Applied Runge–Kutta–Munthe-Kaas integration for quaternion kinematics. Journal of Guidance, Control, and Dynamics, 42(12), 2747–2754. https://doi.org/10.2514/1.G004578
5. Altinörs, A., & Kuzu, F. (2021). Dynamic analysis of a quadcopter using PID, adaptive and LQR control methods. International Journal of Innovative Engineering Applications, 1(2), 45–53. https://doi.org/10.46460/ijiea.929552
6. Nikulin, S. S., & Zabolotnyi, O. V. (2025). Technical aspects of implementing a hardware–software complex for FPV drone testing. Automation of Technological and Business Processes, 17(1), 70–77. https://doi.org/10.15673/atbp.v17i1.3045