Open Source Autopilots for Quadrotor Unmanned Aerial Vehicle

Anitha G[1], Karthikeyan M[2], Karthic V[3]
[1, 2, 3] Department of Avionics, Madras Institute of Technology, Chennai, India
[1] anitha_g@annauniv.edu, [2] karthihss@gmail.com, [3] karthicv26@gmail.com

Abstract— **This paper describes the different steps of designing, building, simulating, and testing an intelligent flight control module for an increasingly popular unmanned aerial vehicle (UAV), known as a quadrotor. It presents an in-depth view of the modeling of the kinematics, dynamics, and control of such an interesting UAV. A quadrotor offers a challenging control problem due to its highly unstable nature. An effective control methodology is therefore needed for such a unique airborne vehicle. The different open source projects available for quadrotor are also discussed along with the selection criteria recommended for the selection of a particular project to a suit and application.**

***Index Terms*—Quadrotor, UAV, control methodologies, fuzzy logic**

I. INTRODUCTION

The quadrotor is a UAV that has four rotors driven by brushless DC motors mounted on the edges of two perpendicular arms. The propellers have a constant pitch unlike normal helicopter which have a variable pitch rotors. The lift is varied varying the speed of the motors. The speed control of the motor is implemented by varying the average voltage supplied to the motor using pulse width modulation (PWM). A quadrotor like an helicopter and unlike a fixed wing airplane, is an inherently unstable system. It can be operated only in a closed loop that includes some control law. The controller that is usually used in most quadrotors is the conventional proportional integral derivative (PID) controller. The PID controller is implemented as a program and loaded on the program memory of the onboard microcontroller. The inherent instability of the quadrotor presents a challenging problem for control design. Moreover at higher speeds of rotation of the propeller, significant aerodynamic effects come into play resulting in high level of nonlinearities that complicate the process of an effective control design.

II. QUADROTOR DYNAMICS AND KINEMATICS

Mathematical modelling provides a description of the behaviour of a system. The flight behaviour of a quadrotor is determined by the speeds of each of the four motors, as they vary in concert, or in opposition with each other. Hence, based on its inputs, a mathematical representation of the system can be used to predict the position and orientation of the quadrotor. The same can further be used to develop a control strategy, whereby manipulating the speeds of individual motors results in achieving the desired motion. To derive the full mathematical model of the quadrotor, the kinematics and dynamics have to be identified first. The kinematic equations provide a relation between the vehicle's position and velocity, whereas the dynamic model defines the relation governing the applied forces and the resulting accelerations.

A. Reference Frames

Before getting into the equations of kinematics and dynamics of the quadrotor, it is necessary to specify the adopted coordinate systems and frames of reference, as well as how transformations between the different coordinate systems are carried out. The use of different coordinate frames is essential for identifying the location and attitude of the quadrotor in six degrees of freedom (6 DOF). For example, in order to evaluate the equations of motion, a coordinate frame attached to the quadrotor is required. However, the forces and moments acting on the quadrotor, along with the inertial measurement unit (IMU) sensor values, are evaluated with reference to the body frame. Finally, the position and speed of the quadrotor are evaluated using GPS measurements with respect to an inertial frame located at the base station. Thus, three main frames of reference are adopted [1].

1) The inertial frame, $\vec{\mathcal{F}_i} = (\vec{x_i}\ \vec{y_i}\ \vec{z_i})$, shown in fig.1, is an earth-fixed coordinate system with the origin located on the ground, for example, at the base station. By convention, the x-axis points towards the north, the y-axis points towards the east, and the z-axis points towards the center of the earth.

2) The body frame, $\vec{\mathcal{F}_b} = (\vec{x_b}\ \vec{y_b}\ \vec{z_b})$ with its origin located at the center of gravity (COG) of the quadrotor, and its axes aligned with the quadrotor structure such that the x-axis $\vec{x_b}$ is along the arm with front motor, the y-axis $\vec{y_b}$ is along the arm

with right motor, and the z-axis, $\vec{z_b} = \vec{x_b} \mathbf{X} \vec{y_b}$ where '$\mathbf{X}$' denotes the cross product.

3) The vehicle frame, $\vec{\mathcal{F}_v} = (\vec{x_v}\ \vec{y_v}\ \vec{z_v})$, is the inertial frame with the origin located at the COG of the quadrotor.

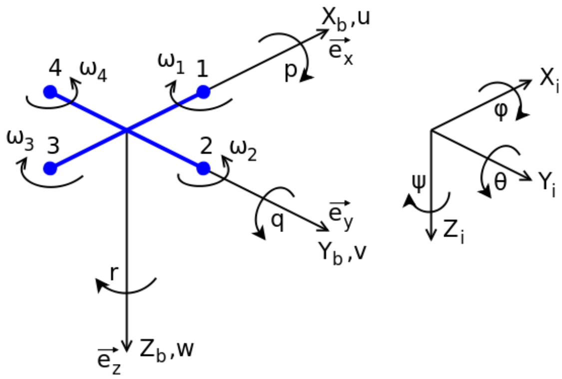

Fig.1. Quadrotor frame of reference

Translation and rotation matrices are used to transform one coordinate reference frame into another desired frame of reference. For example, the transformation from $\vec{\mathcal{F}_i}$ to $\vec{\mathcal{F}_v}$ provides the displacement vector from the origin of the inertial frame to the center of gravity (COG) of the quadrotor. Also, the transformation from $\vec{\mathcal{F}_v}$ to $\vec{\mathcal{F}_b}$ is rotational in nature, therefore yielding the roll, pitch and yaw angles.

B. Quadrotor's Kinematics

Let $P_{\mathcal{F}}^T = [p_x\ p_y\ -p_z]$ and $\Omega_{\mathcal{F}}^T = [\phi\quad\theta\quad\psi]$, denote the quadrotor's position and orientation within a given frame $\mathcal{F}$. The rotation matrix is used to transform vehicle frame entities on to the body frame as in Eq.1.

$$\begin{pmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{K}\end{pmatrix}_{F_v} = \begin{pmatrix}1 & s\phi\tan\theta & c\phi\tan\theta\\ 0 & c\phi & -s\phi\\ 0 & s\phi\sec\psi\text{-}\sin\phi\cos\psi & c\phi\sec\theta\end{pmatrix}\begin{pmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{K}\end{pmatrix}_{F_b}$$
(1)

where $c\phi$ stands for $\cos\phi$ and $s\phi$ for $\sin\phi$.

C. Quadrotor's Dynamics

To build the dynamic model of the quadrotor we will use Newton-Euler formalism, while adopting the following assumptions:

- The quadrotor structure is a rigid body.
- The quadrotor frame is symmetrical.
- The COG of the quadrotor coincides with the center of the rigid frame.

The translational dynamic model is given by Eq.2.

$$\begin{pmatrix}\ddot{p}_x\\ \ddot{p}_y\\ \ddot{p}_z\end{pmatrix} = \begin{bmatrix}\dot{\psi}\dot{p}_y - \dot{\theta}\dot{p}_z\\ \dot{\phi}\dot{p}_z - \dot{\psi}\dot{p}_x\\ \dot{\theta}\dot{p}_x - \dot{\phi}\dot{p}_y\end{bmatrix} + \frac{1}{M}\begin{bmatrix}f_x\\ f_y\\ f_z\end{bmatrix}$$
(2)

The rotational model is given by Eq.3.

$$\begin{pmatrix}\ddot{\phi}\\ \ddot{\theta}\\ \ddot{\psi}\end{pmatrix} = \begin{pmatrix}\frac{[(J]_Y - J_Z)\dot{\theta}\dot{\psi}}{J_X}\\ \frac{[(J]_Z - J_X)\dot{\phi}\dot{\psi}}{J_Y}\\ \frac{[(J]_X - J_Y)\dot{\phi}\dot{\theta}}{J_Z}\end{pmatrix}_{\mathcal{F}_b} + \begin{pmatrix}\frac{\tau_\phi}{J_X}\\ \frac{\tau_\theta}{J_Y}\\ \frac{\tau_\psi}{J_Z}\end{pmatrix}_{\mathcal{F}_b} \qquad (3)$$

III. OPEN SOURCE PROJECTS FOR QUADROTORS

Some open source projects for quadrotor UAVs are explained in this section.

A. Arducopter

Arducopter is a quadrotor autopilot project based on the Arduino framework developed by individual engineers worldwide, which is described earlier. A graphical user interface (GUI)-based software ground control software (GCS) is provided to tune control gains and display flight information. This project shares the same avionics platform with Ardupilot, which is a fixed-wing aircraft autopilot OSP. It uses the GNU Lesser GPL (LGPL) [2].

B. Openpilot

Openpilot is an OSP led by RC hobbyists using GPL. This project features a real-time operating system modified from FreeRTOS, which is an open-source operating system. Openpilot supports fixed wing aircraft and helicopters with the same autopilot avionics. A GUI based GCS is provided to tune gains and receive flight data.

C. Paparazzi

Paparazzi is an autopilot system oriented toward inexpensive autonomous aircraft of all types. It has been in development since 2003 [3]. Originally a fixed wing autopilot, it now supports quadrotor configurations by modifying the control mixing rule. Nine different autopilot hardware systems are developed under the lead of the Paparazzi team at ENAC University. Paparazzi provides GUI based GCS with flight scripting that makes mission planning convenient in outdoors. The project uses GPL for hardware and software.

D.Pixhawk

Pixhawk [4] uses onboard computer vision algorithms developed by ETHZ computer vision group. It is the only project that contains computer vision equipment among the projects introduced here. It also uses a GUI based GCS known as QGroundControl.

E. Mikrokopter

Mikrokopter is a quadrotor autopilot system developed by a subsidiary of HiSystems GmbH in 2006. GUI-based software for gain tuning and health monitoring is provided. Mikrokopter is operated in well-organized Internet shops for their autopilot boards. In 2010, the University of Tasmania and the Australian Antarctic Division made use of Mikrokopter to monitor moss beds in Antarctica.

F. KKmulticopter

KKmulticopter is contributed by 20 people around the world. This project has targeted hobbyists who want to capture aerial photographs using quadrotors. The autopilot hardware of this project is the most basic among the projects described in this article. It is equipped only with a triaxisS gyroscope for inertial measurement and an 8-b microcontroller for control. No GCS is provided and gains are tuned by variable resistors on board.

G.Multiwii

Multiwii is a quadrotor autopilot system developed by RC hobbyists. This project uses an Arduino board as a main processor while the sensor system can vary. This project aims to make the fabrication of electronics easy. It uses gyroscopes and accelerometers of the commercial off-the-shelf Wii motion controller from Nintendo, which needs less soldering. GUI based GCS is provided. GPL is used for this project.

H.Arduino Platform

Although the Arduino platform is not a quadrotor auto-pilot, it is introduced here because many open source quadrotor projects all use it. Arduino is the name of both the open-source single-board microcontroller circuit and the integrated development environment (IDE). Arduino has a well organized device driver library for different sensors and actuators. It is frequently used for rapid prototyping because of the following advantages.

- IDE is easy to install and the firmware can be easily downloaded via USB or RS-232 without an extensive JTAG interface.
- There are more than 100 libraries related to hardware peripherals and signal analysis on the Arduino platform.
- The Arduino is ported on MAC OS, Windows and Linux.

IV. COMPONENTS OF OPEN SOURCE PROJECTS FOR QUADROTOR UAVS

The various components of an open source project for quadrotor UAVs are described in this section.

A. Flight Avionics

Most of the introduced projects provide electronic schematics for self-production. Typically, flight avionics consists of a processor, input/output (I/O) pins, and sensors. The I/O pins connect an off-the-shelf electronic speed controller (ESC) and RC receiver to the flight controller. The sensor suite consists of a gyroscope, accelerometer, barometer, magnetometer, and Global positioning system (GPS). Most flight avionics are full fledged with six degrees of freedom (6 DoF) inertial measurement unit (IMU), magnetometer and barometer. Most flight controllers implement proportional-integral-derivative (PID) control for stabilization of the quadrotor, although the structure of the PID controllers between the projects varies slightly.

B. Radio Transmitters and Receivers

Recently, some groups have modified off-the-shelf RC transmitters to fit their requirement such as complex control mixing or curve shaping of a stick. As a result, custom firmware for a few RC transmitters have been released as open source. In addition open source RC transmitters and receivers are emerging. The OSRC project has developed not only a radio part but also controller hardware. These projects are useful when a flight avionics package needs to be more compact without additional hardware such as an RC receiver.

C. Communication Systems

XBee is a popular communication system because of its simple setup, low cost, and reasonable communication range when compared with its size. All the projects ad- dressed here use Xbee. Arducopter and Pixhawk implement the MAVLink protocol for ground control.

V. OPEN SOURCE QUADROTOR PROJECT INTERNALS

The various algorithms commonly used in the open source quadrotor project internals are explained in this section.

A. Attitude Estimation

A sensor suite is typically composed of a three-axis gyroscope and a three-axis accelerometer, which provide linear accelerations and angular rates only, a proper estimation algorithm has to be employed.

B. Controllers

It is well known that the open-loop rotational dynamics of a quadrotor are unstable as studied in [5]. The identified model reveals that poles are located in the right-half plane of the real-imaginary axis and damping ratio is negative. Therefore, it needs to be stabilized by a feedback control algorithm [6]. SAS, which makes the aircraft stable via the rate measurement in the feedback loop, is popular in aircraft control. SAS is shown in Figure 8 with a dotted-line box. It consists of rate feedback with gain. If SAS is applied to a quadrotor, damping is increased. As a result, the quadrotor becomes controllable by a user. The KKmulticopter has only three gyroscopes. Because the SAS only provides rate regulation, an autopilot is required to maintain the attitude of a quadrotor. The Pixhawk project implements single feedback.

C. Controller Parameters

KKmulticopter is the simplest one, which has only one gain for tuning. Among many controller configurations, PI+P is dominant. P is for the inner loop (rate feedback), and PI is for the forward attitude error compensation.

D. Controller Evaluation

Among these projects, Arducopter, Paparazzi, and Multiwii share the same controller composition. For qualitative evaluation, we mount markers on a quadrotor to acquire ground truth data from the Vicon system. The desired angle is transmitted to the Arducopter-based quadrotor while quadrotor attitude from the Vicon and the transmitted commands are recorded simultaneously.

VI. SELECTION CRITERIA FOR QUADROTOR PROJECTS

The various selection criteria for the quadrotor open source projects are discussed in this section.

A. Availability of flight avionics

All the projects described provide electronic schematic and bill of materials to reproduce their flight avionics. However, it takes high initial cost to manufacture electronics individually. Only five projects among them are available for purchase now: Arducopter, Paparazzi, Mikrokopter, KKmulticopter, and Aeroquad. It is recommended to start with these projects if a reader prefers to avoid electronics fabrication.

B. Attitude estimation algorithm development

For attitude estimation tests, Arducopter and Paparazzi will be a good choice. The other projects are equipped with two or more gyro chips, which are hard to be calibrated for alignment. Only Arducopter and Paparazzi are equipped with 6-DoF IMU. The dynamic range is the best among the all the accelerometers and gyroscopes.

C. Minimalistic configuration

An open-loop model can be easily identified when control input is fully known and no integrators exist in a controller as mentioned in the section “Attitude Estimation Algorithm Development.” Because SAS is implemented to identify the open-loop dynamics, KKmulticopter is a good choice to this end. The system is simple to understand and modify.

D. GPS based navigation

For GPS-based outdoor missions (e.g., waypoint navigation and hovering): Arducopter, Openpilot, Paparazzi, or Mikrokopter will be a good choice. Only these projects support GPS-based navigation. Although Multiwii has GPS, it only supports a homing capability to move a quadrotor back to the initial position.

E. Vision based navigation

Only the Pixhawk project supports vision-based navigation capability. It can synchronize an IMU and a camera in hardware level, which allows tight integration of IMU measurements into the computer vision pipeline.

VII. QUADROTOR PROJECTS IN RESEARCH

A. Vision based navigation

The Pixhawk UAV is designed to be a research platform for computer vision based autonomous flight. The Pixhawk team has constructed a localization test setup using augmented reality Toolkit+. They successfully performed waypoint navigation using a camera on the localization test bed as shown. In [7], adaptive image-based visual serving (IBVS) was integrated with adaptive sliding mode control based on Arducopter. Real time vision-based localization was performed on a quadrotor system based on Arducopter [8]. This quadrotor is equipped with a frontal-view gray- scale USB2.0 camera with 640X480 pixel resolution. Image data from the camera are transferred to a single- board computer and processed in a real time to obtain the vehicle location based on a map created in advance and is shown in Fig.2.

B. Indoor Flight

A Mikrokopter-based quadrotor flew autonomously using a laser range finder (LRF) [9]. Equipped with LRF, Gumstix, and external IMU, it successfully performed autonomous indoor navigation without external localization sensors. Indoor position control based on an onboard LRF was performed on the Mikrokopter based quadrotor platform. An autoregressive moving average with exogenous terms model of the stabilized Mikrokopter was identified. Recently, the quadrotor platform with shared autonomy was investigated for infrastructure inspection.

Multiagent related research can be easily performed on the indoor quadrotor flight system. Especially, as the communication topology between agents can be user defined within the GCS, various settings and algorithms can be exploited. Each quadrotor is equipped with an onboard controller to track input commands sent by the GCS that collects position and/or attitude data of the

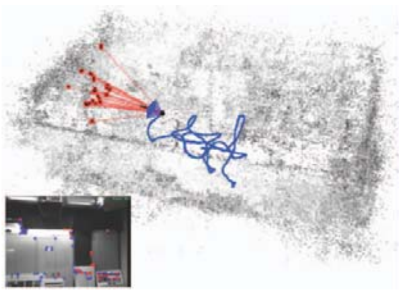

Fig.2. Real time vision base localization

quadrotors from the Vicon motion capture system. Data from the onboard vision sensors are sent to the GCS using a dedicated communication link.

VIII. CONCLUSIONS

This paper has presented eight quadrotor OSPs with descriptions of their avionics, sensor composition, analysis of attitude estimation and control algorithms, and comparison of additional features. To bring out continued improvements based on communities' work, objective evaluations of OSPs remain an important open problem. The meaning of OSP had been more about software, but it is expanding to hardware and even products. There is already a project that has open hardware blueprints and a 3-D model of the quadrotor airframe that can be ordered from 3-D printing services. Sharing the same platform will become easier with such services. It is expected that more OSPs for UAV will be initiated in the future.

REFERENCES

[1] Bouabdallah, S.; Murrieri, P.; Siegwart, R. , "Design and control of an indoor micro quadrotor," *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol.5, no., pp. 4393- 4398 Vol.5, 26 April-1 May 2004.

[2] Free Software Foundation, Inc. (2007, June 29). GNU lesser GPL [Online]. Available: http://www.gnu.org/licenses/lgpl.html.

[3] P. Brisset, A. Drouin, M. Gorraz, P. Huard, and J. Tyler, "The paparazzi solution," in Proc. Micro Aerial Vehicle, Sandestin, Florida, 2006.

[4] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," in Proc. Int. Symp. Experimental Robotics, Dec. 2010.

[5] D. S. Miller, "Open loop system identification of a micro quadrotor helicopter from closed loop data," Master's thesis, Univ. Maryland, College Park, 2011.

[6] F. Lewis and B. Stevens, Aircraft Control and Simulation. Hoboken, NJ: Wiley, 2003.

[7] D. Lee, H. Lim, H. J. Kim, and Y. Kim, "Adaptive image-based visual servoing for an under-actuated quadrotor system," AIAA J. Guid. Control Dyn., vol. 35, no. 4, pp. 1335–1353.

[8] H. Lim, S. N. Sinha, M. Cohen, and M. Uyttendaele, "Real-time image-based 6-dof localization in large scale environments," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), June 2012, pp. 1043–1050.

[9] S. Grzonka, G. Grisetti, and W. Burgard, "A fully autonomous indoor quadrotor," IEEE Trans. Robot., vol. 28, no. 99, pp. 1–11, 2012.