Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online
CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 49-6 (2016) 093–098

A modular virtual laboratory for quadrotor control simulation

Michael D. Murphy

Department of Process, Energy and Transport Engineering,
Cork Institute of Technology, Ireland, (michaeld.murphy@cit.ie)

Abstract: The development of a Virtual Laboratory for control systems simulation is presented in this paper. A mechanistic quadrotor model was created using existing theory and a feedback controller was applied for altitude and coordinate control. The simulation environment was built on a transparent modelling platform that allowed the users to easily modify the controller parameters. The human-machine interface employed both two and three dimensional graphic displays enabling real-time monitoring of the quadrotor's trajectory. The Virtual Laboratory was modulated for usage by multiple student groups. Two example experiments are presented in order to demonstrate the type of control simulations that may be carried out in this Virtual Laboratory. The results from these experiments indicate that this Virtual Laboratory may be easily integrated into control and simulation courses and projects for undergraduate engineering students.

Keywords: Virtual Laboratory, Simulation, Control, UAV, Quadrotor, Education.

1. INTRODUCTION

1.1 Background

Virtual Laboratories (VL) have become an increasingly popular topic for engineering education research (Vrána *et al.,* 2015). Next generation VLs are particularly useful in control education as they allow theoretical systems to come to life through mathematical simulation and give the student a far more interactive learning experience (Chacón *et al.,* 2015). This paper details the development and demonstration of a VL for simulation and control systems education.

1.2 Virtual Laboratory Application

In this study an Unmanned Arial Vehicle (UAV) was selected as the system model. A UAV was chosen for this application as autonomous UAVs (or drones) have become an increasingly common application for control and simulation research. A quadrotor system in particular was selected due to its straight forward design and widespread employment across a multitude of applications (Emelianov *et al.,* 2014; Do Nascimento *et al.,* 2012). A scalable mechanistic model built from first principles theory of mechanics and fluid dynamics was required to allow the student to gain an understanding of the model operations, and also make modifications to the system. A feedback control system was needed for both altitude and coordinate control. The majority of quadrotor controllers have been developed using state-space theory. In order to make the control system in this VL accessible to undergraduate engineering students it was deemed that only classical control theory be applied. The VL in this study was designed to be an intuitive and interactive tool for undergraduate engineering students to better their understanding of mathematical modelling and control systems design.

1.3 Aims and Objectives

- Develop a scalable, mechanistic model of a quadrotor.
- Create a control system for the quadrotor model using classical control methods only.
- Design a VL environment for the quadrotor control simulator with a straight forward human-machine interface to allow easy access for model and control system modification.
- Modularise the VL so as to allow introductory to advanced levels of control systems education for multiple student groups.
- Demonstrate the functionality of the VL for educational purposes.

2. MATERIALS AND METHODS

2.1 Virtual Lab Environment

The VL was built using Matlab and Simulink. The model and control system were parameterised using Matlab while the model operations and algorithms were executed in Simulink. This allowed rapid changes in the system configuration, while also providing a transparent and interchangeable platform for making model and controller modifications (see Appendix A). Both two dimensional and three dimensional plotting systems were employed for visualisation of the dynamic trajectory of the quadrotor (see section 3). The

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.159

mechanistic quadrotor model and control system were created without the use of any Matlab or Simulink toolboxes, or add-ons. This provided both extra transparency to the system operations and also accessibility to students who may not have the latest versions of this software.

2.2. Quadrotor Configuration

The quadrotor is free to rotate around the three axes: pitch (θ) around the y axis, yaw (ψ) around the z axis and roll (ϕ) around the x axis. These rotations are produced by moments about the axes (see figure 1). On a quadrotor, these moments are generated by alternating the rotational speed of the rotors and in doing so produce a net sum of forces that create movements about the quadrotor's centre of gravity. As the quadrotor in this simulation is symmetrical across the x and y axis; the centre of gravity is equidistant from each rotor. The quadrotor parameters used in this study were on based on the OS4 system (Boudabdallah *et al.,* 2007).

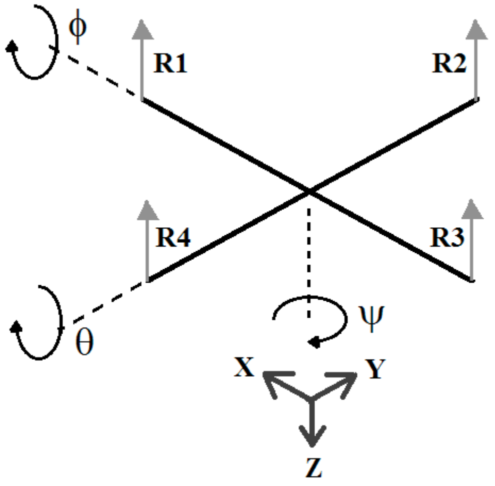

Figure 1. Roll Pitch and Yaw (θ, ψ, ϕ)

The configuration of the quadrotor is as follows (see figure 2): There are two rotor groups; clockwise (2,4) and counter clockwise (1,3) in order to cancel out the moments (torque) around the z axis.

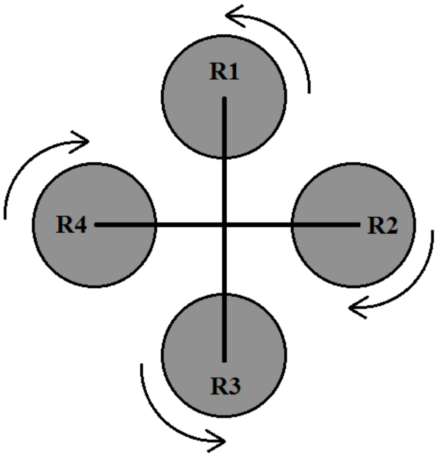

Figure 2. Rotor rotational configuration

2.3 Quadrotor Theory

To control the position of the quadrotor in flight it is necessary to independently modify the thrust of each rotor by altering its rotational speed (Rads/s). The thrust (*T)* can be defined as follows:

$$T_n = b\Omega_n^2 \tag{1}$$

Where Ω is the rotor rotational speed of the rotor, and *b* is the rotor thrust coefficient. The global thrust (T_t) is the summation of the thrusts generate by each rotor.

$$T_t = b(\Omega_4^2 + \Omega_2^2 + \Omega_1^2 + \Omega_3^2) \tag{2}$$

To pitch up; thrust in the front rotor is increased by a set quantity while thrust in the rear rotor is decreased by the same quantity. The torque acting around the y axis (τ_y) is dependent on this thrust differential and *l* is the length of the rotor arm:

$$\tau_y = lb(\Omega_3^2 - \Omega_1^2) \tag{3}$$

Roll control is performed in a similar manner; increasing thrust in one of the side rotors and decreasing in the other to generate torque around the x axis:

$$\tau_x = lb(\Omega_4^2 - \Omega_2^2) \tag{4}$$

Yaw control is achieved by breaking the balance of torque around the z axis. This is done in such a way that the global thrust remains unchanged (increasing rotational speed of one set of rotors while decreasing the corresponding set).

$$\tau_z = d(\Omega_4^2 + \Omega_2^2 - \Omega_1^2 - \Omega_3^2) \tag{5}$$

Where *d* is the rotor drag coefficient.

2.4 Quadrotor Model

The rate of change in angular velocity around each axis ($\ddot{\phi}, \ddot{\theta}, \ddot{\psi}$) is dependent on the torque acting around the axis (τ), the angular velocities around the corresponding axes ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) and the moments of inertia around each axis (I_x,I_y,I_z), (Boudabdallah *et al.,* 2004).

$$\begin{cases} \ddot{\phi} = \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) - \left(\frac{J_r}{I_x}\right)\dot{\theta}\Omega_r + \left(\frac{1}{I_x}\right)\tau_x \\ \ddot{\theta} = \dot{\phi}\dot{\psi}\left(\frac{I_z - I_x}{I_y}\right) + \left(\frac{J_r}{I_y}\right)\dot{\phi}\Omega_r + \left(\frac{1}{I_y}\right)\tau_y \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) + \left(\frac{1}{I_z}\right)\tau_z \end{cases} \tag{6-8}$$

Where Ω_r is the difference in rotational speed between the two rotor groups;