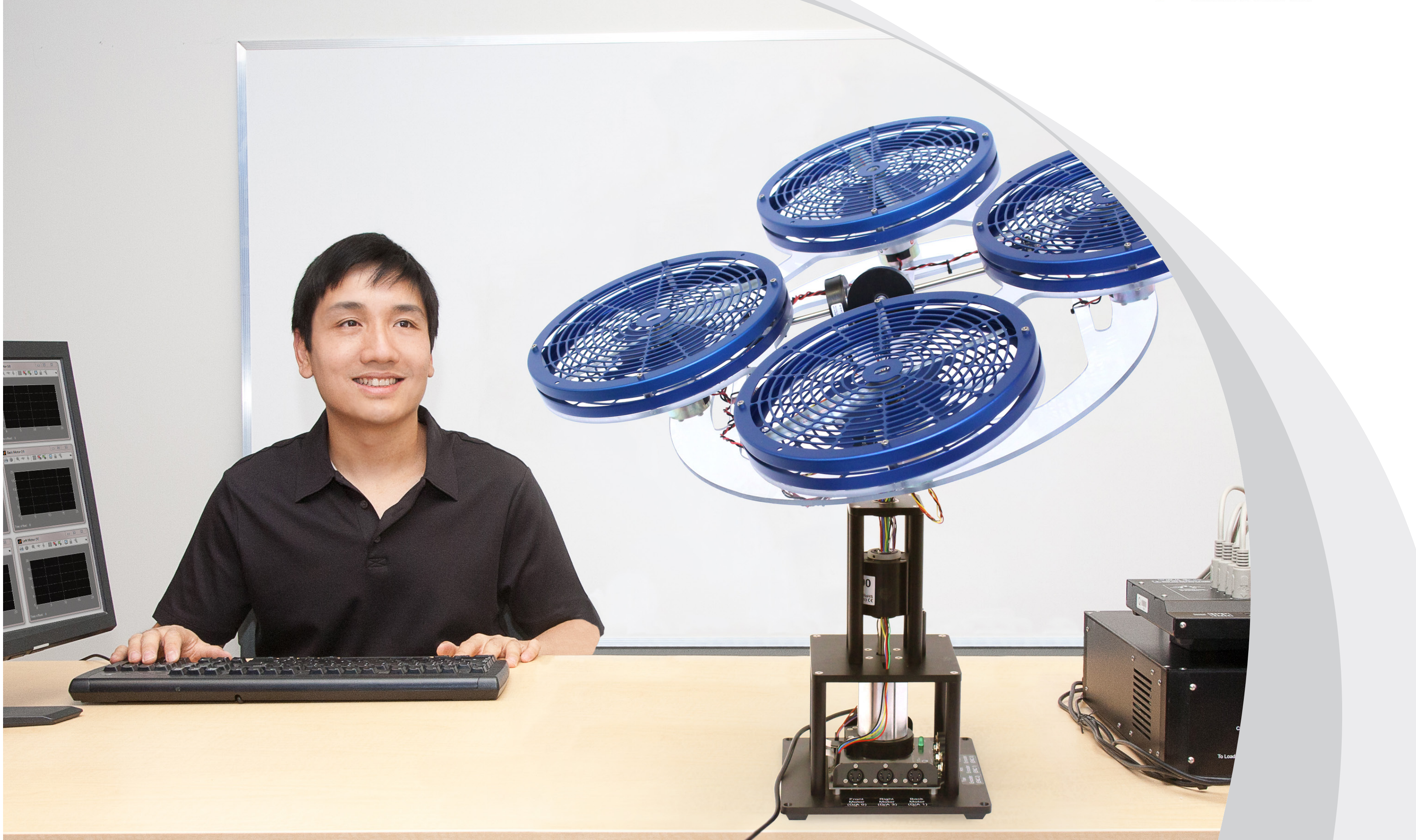
QUANSER
INNOVATE·EDUCATE

LABORATORY GUIDE

3 DOF Hover Experiment for MATLAB®/Simulink® Users

Developed by:
Jacob Apkarian, Ph.D., Quanser
Michel Lévis, M.A.SC., Quanser

Quanser educational solutions are powered by:
MATLAB
SIMULINK



CAPTIVATE. MOTIVATE. GRADUATE.

Quanser Inc.
119 Spy Court
Markham, Ontario
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at:
http://www.quanser.com

CONTENTS

1 INTRODUCTION

This laboratory guide summarizes a simple, linear dynamic model of the Quanser 3 DOF Hover and designs a state-feedback control system. The goal is control the position of pitch, roll, and yaw axes of the 3 DOF Hover with a smooth control signal that does not saturate the actuators.

Topics Covered

- Obtain a state-space representation of the open-loop system.
- Design and tune an LQR-based state-feedback controller satisfying the closed-loop system's desired design specifications.
- Simulate the system and ensure it is stabilized using the designed state-feedback control.
- Implement the state-feedback controller on the 3 DOF Hover system and evaluate its actual performance.

1.1 Prerequisites

In order to successfully carry out this laboratory, the user should be familiar with the following:

- See the System Requirements given in Section 4 for the required hardware and software.
- Modeling and state-space representation.
- State-feedback design using Linear-Quadratic Regular (LQR) optimization.
- Basics of Simulink®.
- How to use basic operations in the Quanser QUARC® software. Please see the QUARC demo and help page for more information.

2 BACKGROUND

2.1 Modeling

2.1.1 Free-Body Diagram

The free-body diagram of the Quanser 3 DOF Hover is illustrated in Figure 2.1 and it accompanies the Maple worksheet named *3 DOF Hover Equations.mws* or its HTML equivalent *3 DOF Hover Equations.html*. The equations can be edited and re-calculated by executing the worksheet using the Maple 9 software.

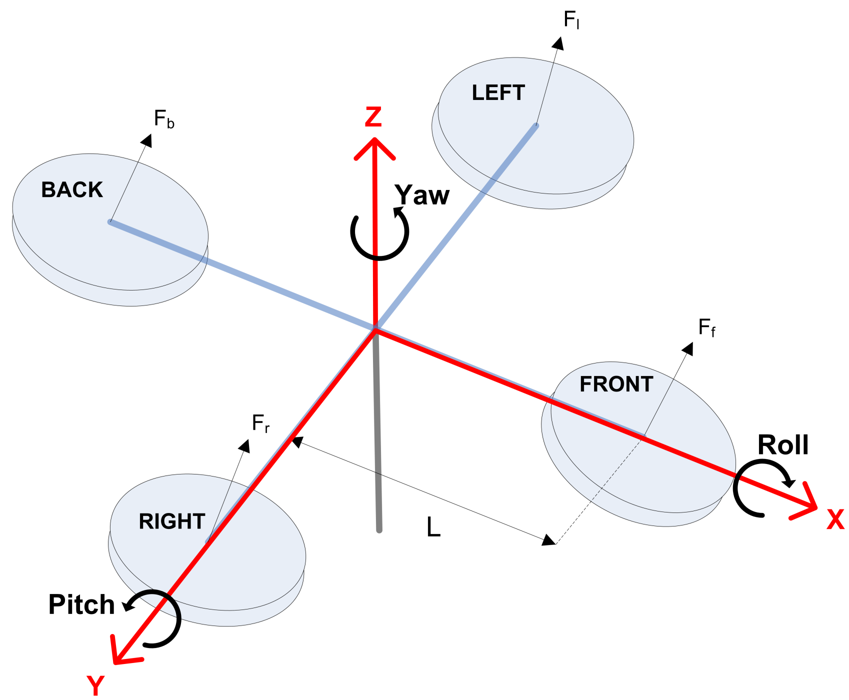


Figure 2.1: Simple free-body diagram of 3 DOF Hover

The 3 DOF Hover modeling conventions used are:

1. The 3 DOF Hover is horizontal (i.e., parallel with the ground) when the pitch and roll angles are zero, $\theta_p = 0$ and $\theta_r = 0$.
2. Yaw angle increases positively,$\dot{\theta}_y(t) > 0$ when the body rotates in the **counter-clockwise (CCW)** direction.
3. Pitch angle increases positively, $\dot{\theta}_p > 0$ when rotated CCW.
4. Roll angle increases positively, $\dot{\theta}_r > 0$, when rotate CCW.

When a positive voltage is applied to any motor a positive thrust force is generated and this causes the corresponding propeller assembly to rise. The thrust force generated by the front, back, right, and left motors are denoted by F_f, F_b, F_r, and F_l, respectively. The thrust forces generated by the front and back motors primarily control the motions about the pitch axis while the right and left motors primarily move the hover about its roll axis. Notice that the pitch angle increases when the thrust force from the front motor is larger than back motor $F_f > F_b$. The roll angle increases when the thrust force from the right motor is larger than the left motor, $F_r > F_l$.

2.1.2 Pitch and Roll Axis Model

The dynamics for each axis can be described by the general equation

$$J\ddot{\theta} = \Delta F L$$

where θ is the angle of the pivot, L is the distance between the propeller motor and the pivot on the axis, J is the moment of inertia about the axis, and ΔF is the differential thrust-force. With the force diagram in Figure 2.2, we can model the pitch axis using the equation

$$J_p\ddot{\theta}_p = K_f(V_f - V_b) \tag{2.1}$$

where K_f is the thrust-force constant, V_f is the front motor voltage, V_b is the back motor voltage, θ_p is the pitch angle, and J_p is the moment of inertia about the pitch axis. Notice that this follows the conventions in Figure 2.1, where the pitch angle increases when the front motor voltage is larger than the back motor.

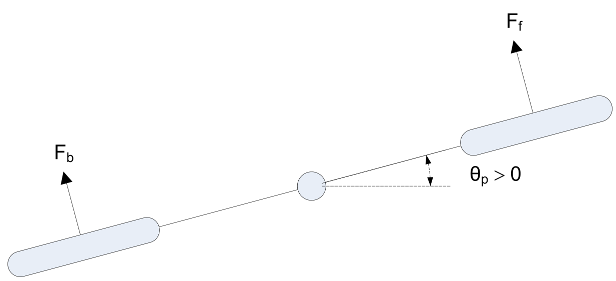


Figure 2.2: Free-body diagram of pitch axis

Similarly, for the roll axis we have

$$J_r\ddot{\theta}_r = K_f(V_r - V_l) \tag{2.2}$$

where K_f is the thrust-force constant, V_r is the right motor voltage, V_l is the left motor voltage, θ_r is the roll angle, and J_r is the moment of inertia about the roll axis. The roll angle increases when the right motor voltage is larger than the left motor.

2.1.3 Yaw Axis Model

The motion about the yaw axis, illustrated in Figure 2.3, is caused by the difference in torques exerted by the two clockwise and two counter-clockwise rotating propellers

$$J_y\ddot{\theta}_y = \Delta\tau = \tau_l + \tau_r - \tau_f - \tau_b$$

where τ_l and τ_r are the torques generated by the left and right *clockwise* propellers and τ_f and τ_b are the torques exerted by the front and back counter-clockwise rotors. Note that the counter-clockwise torques are negative. The torque generated by all the propellers is assumed to be $\tau = K_t V_m$, where K_t is the thurst-torque constant and V_m is the motor voltage. Thus in terms of applied voltage, the yaw axis equation of motion is

$$J_y\ddot{\theta}_y = K_t(V_r + V_l) - K_t(V_f + V_b). \tag{2.3}$$

2.1.4 State-Space Model

The state-space representation is given by

$$\dot{x} = Ax + Bu$$

and

$$y = Cx + Du.$$

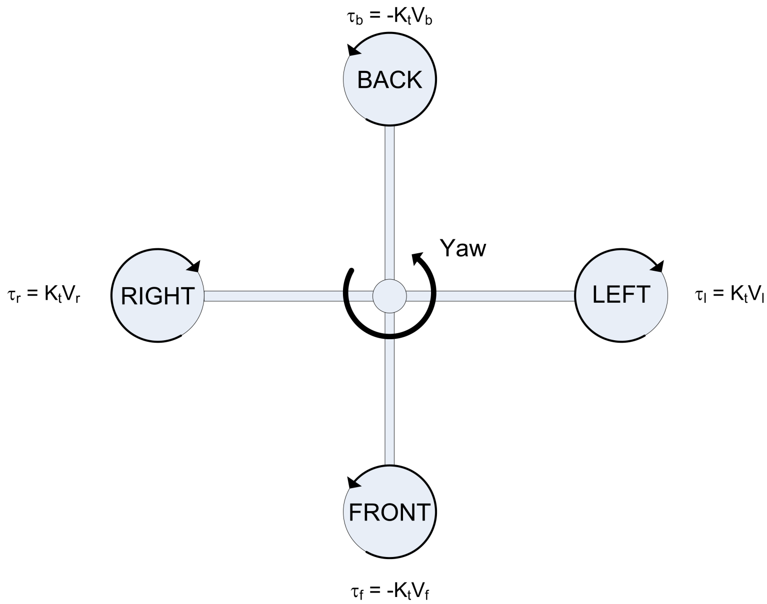


Figure 2.3: Free-body diagram of yaw axis

For the 3 DOF Hover, we define the state vector

$$x^\top = \left[\begin{array}{cccccc} \theta_y & \theta_p & \theta_r & \dot{\theta}_y & \dot{\theta}_p & \dot{\theta}_r \end{array}\right], \tag{2.4}$$

the output vector

$$y^\top = \left[\begin{array}{ccc} \theta_y & \theta_p & \theta_r \end{array}\right]$$

and the control vector

$$u^\top = \left[V_f \quad V_b \quad V_r \quad V_l\right].$$

Using the equations of motion given in equations 2.1, 2.2, and 2.3, the corresponding 3 DOF Hover state-space matrices (as derived in the Maple worksheet) are

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{K_t}{J_y} & -\frac{K_t}{J_y} & \frac{K_t}{J_y} & \frac{K_t}{J_y} \\ \frac{LK_f}{J_p} & -\frac{LK_f}{J_p} & 0 & 0 \\ 0 & 0 & \frac{LK_f}{J_r} & -\frac{LK_f}{J_r} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The model parameters used in the (A, B) matrices are defined in the Quanser 3 DOF Hover User Manual.

2.2 Control Design

2.2.1 State-Feedback

In this section a state-feedback controller is designed to regulate the pitch, roll, and yaw angles of the Quanser 3 DOF Hover to desired positions. The control gains are computed using the Linear-Quadratic Regulator algorithm in Section 2.2.2.

The state-feedback controller entering the motors is defined

$$u = \begin{bmatrix} V_f & V_b & V_r & V_l \end{bmatrix}^\top = \begin{cases} K(x_d - x) + u_{\text{bias}} & \text{if } u \geq 0 \\ 0 & \text{if } u < 0 \end{cases} \tag{2.5}$$

where x is defined in Equation 2.4, $K \in \Re^{4\times6}$ is the control gain,

$$x_d = \begin{bmatrix} \theta_{d,y} & \theta_{d,p} & \theta_{d,r} & 0 & 0 & 0 \end{bmatrix}$$

is the setpoint vector (i.e., desired or reference angles) and

$$u_{\text{bias}}^\top = \begin{bmatrix} V_{\text{bias}} & V_{\text{bias}} & V_{\text{bias}} & V_{\text{bias}} \end{bmatrix}$$

is the bias voltage, i.e., a fixed constant voltage applied to each motor. Adding a bias voltage to each propeller helps prevent the voltage from going below zero and being cutoff. By keeping the rotors in motion, this can also help make the system more responsive.

Caution: Due to the low resistance of the motor, switching between positive and negative voltage can cause permanent damage to the power amplifier. The controller given in Equation 2.5 only applies positive voltage to the motors.

Allowing only positive thrust also makes it resemble more closely to how actual VTOL and helicopter devices operate, i.e., their propellers cannot reverse direction.

2.2.2 Linear Quadratic Regulator

The control gains are computed using the Linear-Quadratic Regulator scheme. Using the feedback law

$$u = -Kx$$

the weighting matrices

$$Q = \begin{bmatrix} 500 & 0 & 0 & 0 & 0 & 0 \\ 0 & 350 & 0 & 0 & 0 & 0 \\ 0 & 0 & 350 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 20 \end{bmatrix}$$

and

$$R = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix}$$

and the state-space matrices (A, B) found previously, the control gain

$$K = \begin{bmatrix} -111.8 & 132.3 & 0 & -41.41 & 36.23 & 0 \\ -111.8 & -132.3 & 0 & -41.41 & -36.23 & 0 \\ 111.8 & 0 & 132.3 & 41.41 & 0 & 36.23 \\ 111.8 & 0 & -132.3 & 41.41 & 0 & -36.23 \end{bmatrix}$$

is calculated by minimizing the cost function

$$J = \int_0^\infty x^\top Q x + u^\top R u \, dt.$$

3 IN-LAB PROCEDURE

3.1 Controller Simulation

In this section we will use the Simulink diagram shown in Figure 3.1 to simulate the closed-loop control of the Quanser 3 DOF Hover system. The system is simulated using the linear model summarized in Section 2.1. The Simulink model uses the state-feedback control described in Section 2.2. The feedback gain K is found using the Matlab LQR command (LQR is described briefly in Section 2.2.2). The goal is to make sure the generated gain can stabilizes the system while not saturating the dc motors.

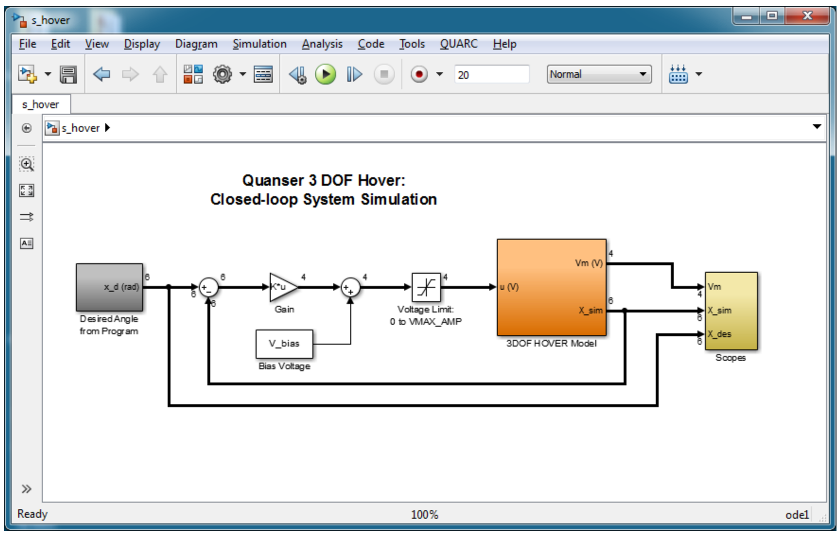


Figure 3.1: Simulink diagram used to simulate 3 DOF Hover system

The *3 DOF HOVER Model* block shown in Figure 3.1 uses the Simulink State-Space block to load the dynamics model from the workspace. High-pass filters are used to obtain the velocity states.

IMPORTANT: Before you can conduct these experiments, you need to make sure that the lab files are configured according to your setup. If they have not been configured already, then you need to go to Section 4 to configure the lab files first.

3.1.1 Objectives

- Investigate the closed-loop position control performance of the LQR using a linear model of the 3 DOF Hover system.
- Ensure the controller does not saturate the actuator.

3.1.2 Procedure

Follow these steps to simulate the closed-loop response of the 3 DOF Hover:

1. For details on how to set up the simulation, see Section 4.
2. Load the Matlab software.
3. Open the Simulink model called *s_hover.mdl*, shown in Figure 3.1.

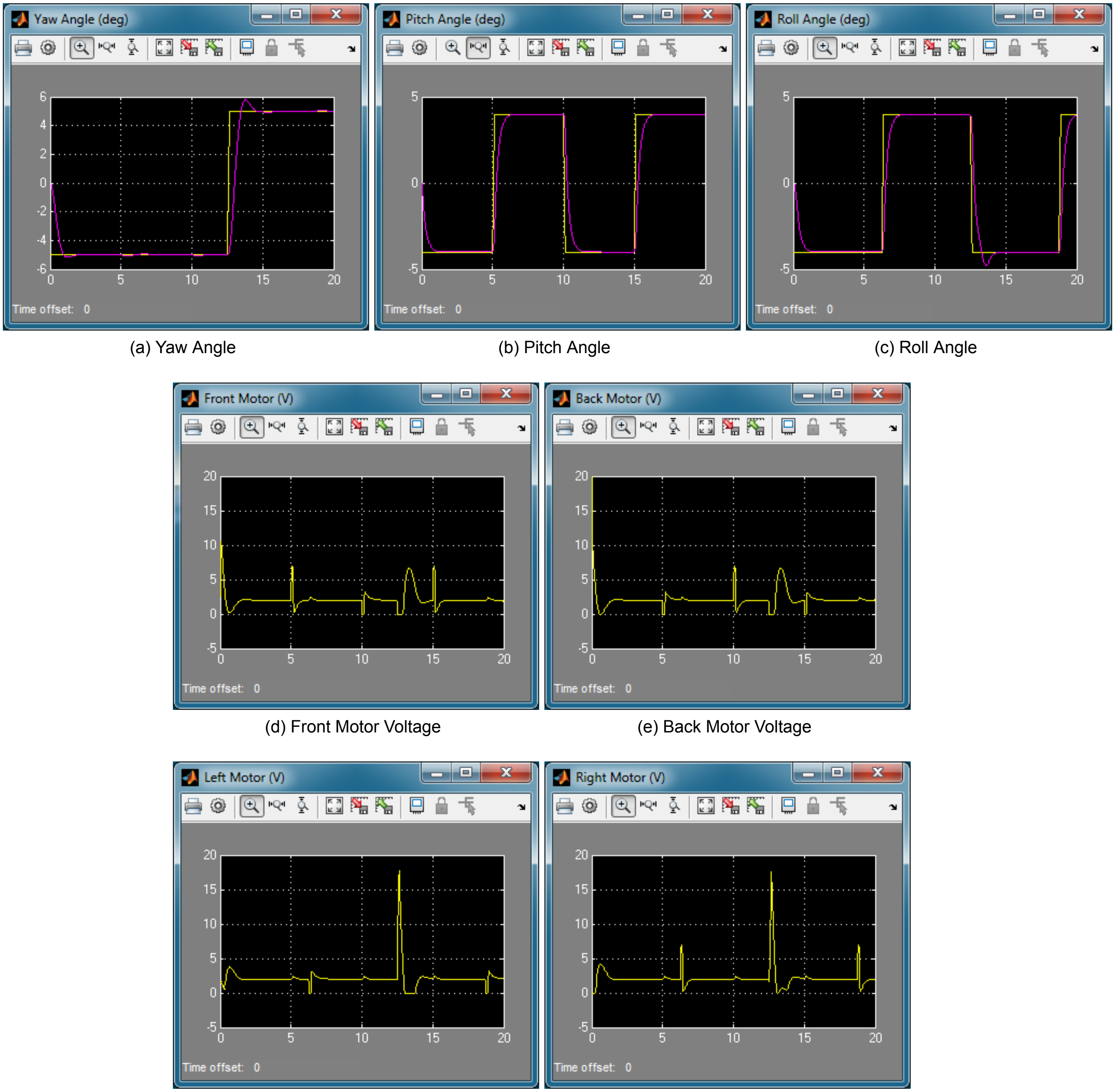
4. Run the setup_hover.m script.

5. To command a desired yaw step of ± 5 degrees at 0.04 Hz, pitch step of $\pm$2 degrees at 0.1 Hz frequency, and a roll angle step of $\pm$2 degrees at 0.08 Hz, open the *Desired Angle from Program* block and set the following:

   - *Amplitude: Yaw (deg)* gain block to 5 degrees.
   - In *Signal Generator: Roll* block, set the *Frequency* input box to 0.04 Hz.
   - *Amplitude: Pitch (deg)* gain block to 4 degrees.
   - In *Signal Generator: Pitch* block, set the *Frequency* input box to 0.1 Hz.
   - *Amplitude: Roll (deg)* gain block to 4 degrees.
   - In *Signal Generator: Roll* block, set the *Frequency* input box to 0.08 Hz.

6. Ensure the 3 DOF Hover pitch, yaw, and roll axis angle scopes as well as the all the motor voltage scopes are open. If not, go into the *Scopes* subsystem and double-click on those sinks.

7. Start the simulation. The scopes should be displaying responses similar to Figure 3.2. The yaw, pitch, and roll angles should track the commanded square position signal scope. The Saturation block only allows positive control voltages and limits them to a maximum of 24V (which can be changed). At low command angles, the voltages are changing about the bias voltage 2 V, i.e., $V_{bias} = 2$V.



(a) Yaw Angle (b) Pitch Angle (c) Roll Angle

(d) Front Motor Voltage (e) Back Motor Voltage

(f) Left Motor Voltage (g) Right Motor Voltage

Figure 3.2: Simulated closed-loop response

3.2 Implementation

The q_hover Simulink diagram shown in Figure 3.3 is used to perform the balance control on the 3 DOF Hover. The *3 DOF HOVER* subsystem contains QUARC® blocks that interface with the DC motor and encoders of the 3 DOF Hover system.

IMPORTANT: Before you can conduct these experiments, you need to make sure that the lab files are configured according to your setup. If they have not been configured already, then you need to go to Section 4 to configure the

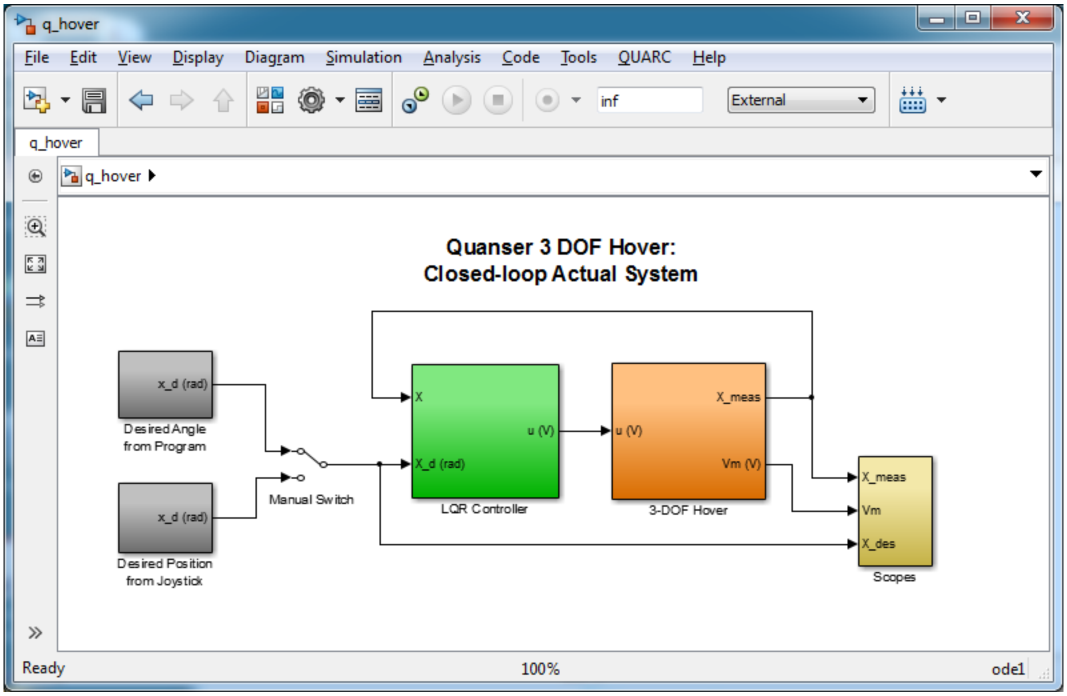


Figure 3.3: Simulink model used with QUARC®to run controller on the 3 DOF Hover.

lab files first.

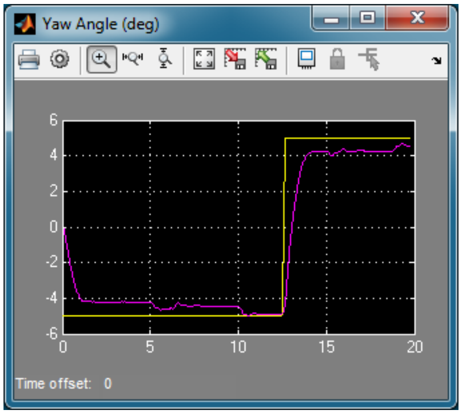
3.2.1 Objectives

Implement with QUARC the previously designed LQR in order to control the position of the 3 DOF Hover.

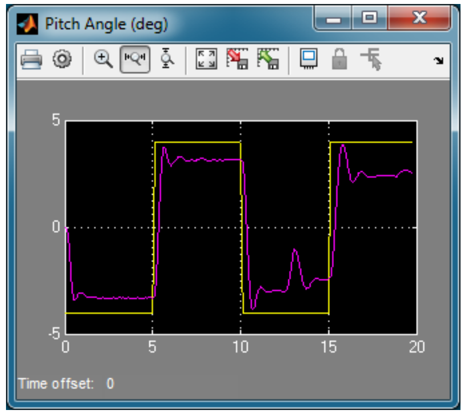
3.2.2 Procedure

Follow this procedure:

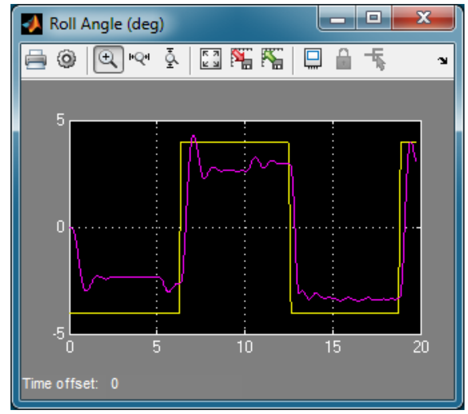
1. Run the setup_hover.m script using the LQR weighting matrices that you used in the simulation in Section 3.1.
2. Ensure the 3 DOF Hover pitch, yaw, and roll axis angle scopes as well as the all the motor voltage scopes are open. If not, go into the *Scopes* subsystem and double-click on those sinks.
3. In the Simulink diagram, go to QUARC | Build.
4. Set the manual switch is to the upward position to generate commands from the *Desired Angle from Program* block (i.e., not the joystick).
5. Click on QUARC | Start to run the controller. See the sample response shown in Figure 3.4 using a yaw step of ± 5 degrees at 0.04 Hz, a pitch of ± 4 degrees at 0.1 Hz, and a roll of ± 4 degrees at 0.08 Hz.



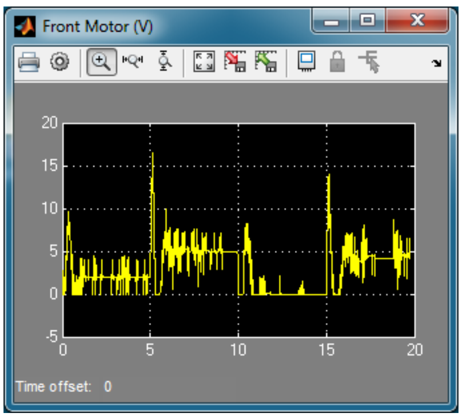
(a) Yaw Angle



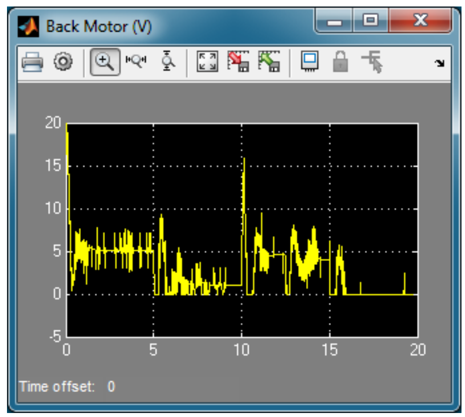
(b) Pitch Angle



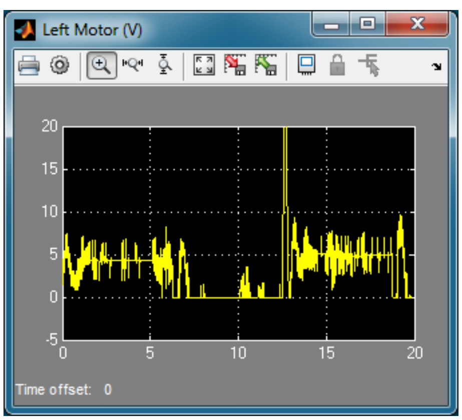
(c) Roll Angle



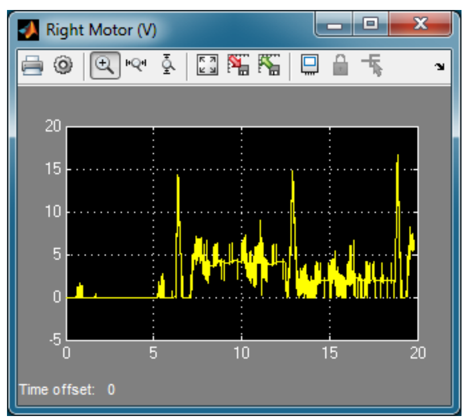
(d) Front Motor Voltage



(e) Back Motor Voltage



(f) Left Motor Voltage



(g) Right Motor Voltage

Figure 3.4: Sample closed-loop response of 3 DOF Hover

6. If your specifications have not been met, you can finely tune the LQR weighting matrices on-the-fly using the following command:

```
>> K = lqr(A,B, diag([Q(1,1),Q(1,1),Q(3,3),Q(4,4),Q(5,5)]),R)
```

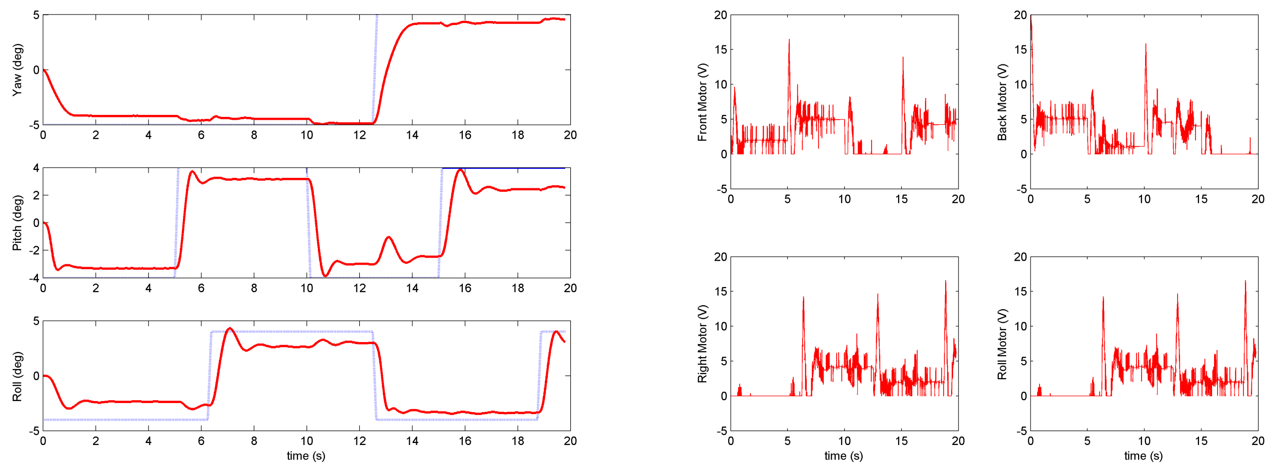
7. The desired angle can be also be generated using a USB joystick described in 3 DOF Hover User Manual ([4]). To use the joystick, set the Manual Switch in q_hover, shown in Figure 3.3, to the downward position.

⚠ **Caution:** Do not switch between Program and Joystick command when the controller is running. Change switch setting when the controller is OFF.

8. To stop the experiment, click on the QUARC | Stop button.
9. Power off the amplifier(s).

3.2.3 Analysis

An example of the closed-loop balance response is shown in Figure 3.5. You can generate this using the plot command after running the q_hover QUARC controller.



(a) Axis Positions (b) Motor Voltage

Figure 3.5: 3 DOF Hover closed-loop control response

The tracking performance of the pitch and roll are affected by the yaw control. When the yaw is stabilized at -5 degrees, the pitch step has a better peak time and steady-state error then when the yaw is stablized at 5 degrees. This may be because when the yaw is at -5 degrees, the front and back motor have a higher average voltage which creates more overall thrust force and hence better control performance. Similarly, the roll angle tracks its desired angle better when the yaw is at 5 degrees, i.e., when the left and right motors have higher average voltage.

This is the drawback of the decoupled linear controller. It is based on a simple model and does not compensate for the coupling and other unmodeled effects in the system. Increasing V_{bias} does improve performance, but can draw too much current from the motors. The steady-state error cab easily be improved using an integrator (an anti-windup strategy would be recommended) or a more advanced algorithm that compensates for disturbances and uncertainties (e.g., adaptive or robust type control).

4 SYSTEM REQUIREMENTS

Required Software

- Microsoft Visual Studio (MS VS) or Microsoft Windows SDK compiler tool
- Matlab® with Simulink®, Real-Time Workshop, and the Control System Toolbox
- QUARC®

See the QUARC® software compatibility chart in [3] to see what versions of the compiler tool and Matlab are compatible with your version of QUARC and for what OS.

Required Hardware

- Data acquisition (DAQ) device **with 3x encoder inputs and 4x analog output channels** that is compatible with QUARC®. This includes Quanser DAQ boards such as Q8-USB, QPID, and QPIDe and some National Instruments DAQ devices. For a full listing of compliant DAQ cards, see Reference [1].
- Quanser 3 DOF Hover
- Quanser VoltPAQ-X4 power amplifier, or an equivalent amplifier solution for four motors.

Before Starting Lab

Before you begin this laboratory make sure:

- QUARC® is installed on your PC, as described in [2].
- DAQ device has been successfully tested (e.g., using the test software in the Quick Start Guide or the *QUARC Analog Loopback Demo*).
- 3 DOF Hover and amplifier are connected to your DAQ board as described its User Manual [4].

4.1 Overview of Files

File Name	Description
Quanser 3 DOF Hover User Manual.pdf	This manual describes the hardware of the 3 DOF Hover system and explains how to setup and wire the system for the experiments.
Quanser 3 DOF Hover Laboratory Guide.pdf	This document demonstrates how to obtain the linear state-space model of the system, simulate the closed-loop system, and implement controllers on the 3 DOF Hover plant using QUARC®.
setup_hover.m	The main Matlab script that sets the motor and sensor parameters as well as the control gain. Run this file only to setup the laboratory.
config_hover.m	Returns the 3 DOF Hover system specifications and sensor calibration constant.
s_hover.mdl	Simulink file that simulates the closed-loop control of a Quanser 3 DOF Hover system using state-feedback control.
q_hover.mdl	Simulink file that implements the state-feedback control on the Quanser 3 DOF Hover system using QUARC®.
3 DOF Hover Equations.mws	Maple worksheet used to develop the model for the 3 DOF Hover experiment. Waterloo Maple 9, or a later release, is required to open, modify, and execute this file.
3 DOF Hover Equations.html	HTML presentation of the Maple Worksheet. It allows users to view the content of the Maple file without having Maple installed. No modifications to the equations can be performed when in this format.

Table 4.1: Files supplied with the 3 DOF Hover

4.2 Setup for Simulation

Before beginning the in-lab procedure outlined in Section 3.1, the s_hover Simulink diagram and the setup_hover.m script must be configured.

Follow these steps:

1. Load the Matlab software.
2. Browse through the *Current Directory* window in Matlab and find the folder that contains the file *setup_hover.m*.
3. Open the *setup_hover.m* script.
4. **Configure setup_hover script**: Make sure the script is setup to match this setup:
   - K_AMP to 3 (make sure you amplifier gain setting is also 3)
   - VMAX_AMP to 24V. You could also set this lower
   - CMD_RATE_LIMIT to 60 deg/s. This limits the rate of all setpoint step commands (which results in a smoother control signal).
   - You can also tune other parameters, e.g., those found in the *Filter* and *Joystick Parameters* sections.
5. Run the Matlab script *setup_hover.m* to load the state-space model matrices, control gains, and various other parameters into the Matlab workspace. The LQR control gain should be displayed in the Matlab Command Window.
6. Open the *s_hover.mdl* Simulink diagram, shown in Figure 3.1.

4.3 Setup for Running on 3 DOF Hover

Before performing the in-lab exercises in Section 3.2, the q_hover Simulink diagram and the setup_hover.m script must be configured.

Follow these steps to get the system ready for this lab:

1. Setup the 3 DOF Hover system as detailed in the 3 DOF Hover User Manual ([4]).
2. **Make sure the 3 DOF Hover is balanced** before starting any controller. For more information, go to the 3 DOF Hover User Manual ([4]).
3. Configure and run setup_hover.m as explained in Section 4.2.
4. Open the *q_hover.mdl* Simulink diagram, shown in Figure 3.3.
5. **Configure DAQ:** Ensure the HIL Initialize block in the *3 DOF Hover* subsystem is configured for the DAQ device that is installed in your system. See QUARC User Manual ([1]) for more information on configuring the HIL Initialize block.
6. Turn ON the power amplifier(s).
7. The desired angle can be also be generated using a USB joystick described in 3 DOF Hover User Manual ([4]). To use the joystick, set the Manual Switch shown in Figure 3.3 to the downward positions. The rate at which the desired angle increases or decreases given a joystick position can be changed using the K_JOYSTICK_X and K_JOYSTICK_Y variables that are set in the *setup_hover.m* script file. Note that the same rate limiter parameter CMD_RATE_LIMIT is used for the joystick as well.

REFERENCES

[1] Quanser Inc. *QUARC User Manual*.

[2] Quanser Inc. *QUARC Installation Guide*, 2009.

[3] Quanser Inc. *QUARC Compatibility Table*, 2010.

[4] Quanser Inc. *3 DOF Hover User Manual*, 2013.