Experiments on 2-DOF Helicopter Using Approximate Dynamic Programming

Anthony Birge   Andrew Fandel   Advisor: Dr. Suruz Miah

November 20, 2017

Nomenclature

DOF	Degrees of Freedom
V-REP	Virtual Robot Experimentation Platform
UAV	Unmanned Aerial Vehicle
ϕ	Roll angle (longitudinal axis rotation)
θ	Pitch angle (lateral axis rotation)
ψ	Yaw angle (vertical axis rotation)

1 Project Description

Unmanned Aerial Vehicles (UAVs) offer a small-form solution to many aerial tasks, and provide many benefits, such as cost effectiveness and the lack of a pilot. Consequently, they are rapidly performing many more tasks in the civilian and military domains. UAVs can be controlled automatically by algorithms that serve to perform a given task with minimal control input to the UAV from its control system. Control algorithms that minimize the effort needed to control the UAV by actuation have been advanced in recent literature. 2-Degree of Freedom (2-DOF) helicopters exist as a low-budget solution to implement control algorithms in a testing lab setting. This project aims to advance motion control of a 2-DOF helicopter under various operating conditions using machine learning algorithms. Motion control strategies will be employed for a possible minimization of energy consumed, navigation time, or even the risk for an UAV to complete its pre-defined tasks in an environment.

One of the most challenging tasks in control applications is to model a system, such as the helicopter considered in this project, using a set of ordinary nonlinear differential equations. Under certain operating conditions, the complex model of a helicopter can be simplified to provide appropriate actuator commands to a helicopter. This work leverages the mechanical aspect of a 2-DOF helicopter, the Quanser AERO, manufactured by Quanser Inc. Figure 1 depicts the Quanser AERO configured as a 2-DOF helicopter. The AERO can be configured as a half-quadcopter by rotating one of the rotors a quarter of a turn. Note that the Quanser AERO can be modeled as a linear differential equations, therefore, a set of well-defined motion control strategies can be applied to control its pitch and yaw motion when its base

Figure 1: Quanser AERO configured as 2-DOF helicopter.

is fixed. Here, we focus on implementing model-based reinforcement learning strategy to determine the sub-optimal actuator commands to the helicopter's actuators for it to track its pre-defined motion trajectories.

To demonstrate the effectiveness of the proposed control technique, we will implement the proposed control technique to both configurations of the AERO in various operating conditions. The proposed control technique will be simulated in both configurations in MATLAB, and the half-quadcopter configuration will be simulated in Virtual Robot Experimentation Platform (V-REP) [1]. Both configurations will then be implemented to the Quanser AERO. A user is expected to control the helicopter using a desktop or laptop computer in conjunction with a single-board microcomputer, such as a Raspberry Pi or BeagleBone. The ultimate goal of implementation will have a personal computer/laptop communicating to the single-board computer via Wi-Fi from which the single-board computer will control the Quanser AERO. This implementation should result in a cost-effective and convenient control technique.

[1]See http://www.coppeliarobotics.com/ for details.

Figure 2: High-level system block diagram.

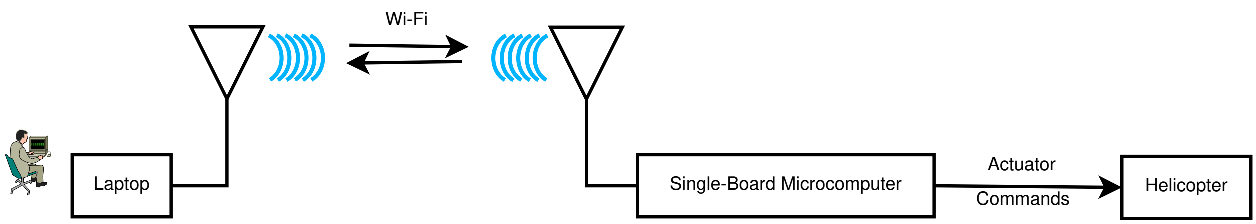

Figure 3: Helicopter subsystem block diagram.

2 System Architecture

2.1 High-Level System

A high-level system block diagram of the targeted implementation is shown in Figure 2.

2.1.1 Inputs

- *Desired Configurations* - The pitch and yaw of the 2-DOF helicopter are the two desired configurations to be input into the helicopter setup. The pitch is the vertical adjustment of the helicopter, while the yaw is the horizontal adjustment.

2.1.2 System

- *Helicopter* - The helicopter is the physical Quanser AERO. The approximate dynamic programming controller will be implemented to the helicopter through the computer and single-board computer as seen in Figure 2.

2.1.3 Outputs

- *Helicopter's Response/Outputs* - The measured pitch and measured yaw of the helicopter constitute the response of the system. If the controller functions properly and is implemented correctly, the response of the helicopter will closely track that of the desired configurations.

2.2 Helicopter Subsystem

The helicopter configuration to be used in the project is the Quanser AERO 2-DOF helicopter. The Quanser AERO can be configured both as a half-quadcopter and helicopter.

2.2.1 System/Signals

- *Laptop* - The laptop controls all operations of the system. All commands are initialized by the computer to be processed by the other blocks of the system.
- *Wi-Fi* - The computer will send a Wi-Fi signal to either a Raspberry Pi or Beagle Bone single-board computer. This signal will contain the necessary instructions for operation of the single-board computer.
- *Single-Board Microcomputer* - The Raspberry Pi or Beagle Bone single-board computer will receive instructions from the computer through a Wi-Fi signal. These instructions will in turn be used to operate the helicopter block.
- *Helicopter* - The helicopter is the Quanser AERO 2-DOF helicopter. The subsystem block diagram for the helicopter can be seen in Figure 3.

3 Modes of Operations

- ***Stage 1*** - *Pitch Control*
  - Yaw will be forced to be constant mechanically, and the controller will make the system track a given pitch, which may vary over time.
- ***Stage 2*** - *Yaw Control*
  - Pitch will be forced to be constant mechanically, and the controller will make the system track a given yaw, which may vary over time.
- ***Stage 3*** - *Pitch and Yaw Control*
  - The controller will track time varying functions of both pitch and yaw.

4 Mathematical Problem Formulation

The error for our implementation is the difference between the desired pitch, θ, and yaw, ψ, and the actual(measured) pitch and actual yaw as seen in Equation 1. The approximate dynamic programming controller should minimize the root mean squared error of the system. A visual representation of this can be seen in Figure 4. By minimizing the root mean squared error, we will approach the optimal response.

$$\mathbf{e}(t) = \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \\ e_4(t) \end{bmatrix} = \begin{bmatrix} \theta^d(t) - \theta(t) \\ \psi^d(t) - \psi(t) \\ \dot{\theta}^d(t) - \dot{\theta}(t) \\ \dot{\psi}^d(t) - \dot{\psi}(t) \end{bmatrix} \tag{1}$$

Figure 4: Visual representation of the error between the desired configurations and the actual helicopter response.

The approximate dynamic programming controller will take time to optimize the control parameters of the system. The end goal will be that of Equation 2.

$$\mathbf{e}(t) \rightarrow \mathbf{0} \ as \ \mathbf{t} \rightarrow \infty \tag{2}$$

The approximate dynamic programming controller will make use of the state-space model a desired system. This can be seen in Equation 3 and Equation 4.

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{3}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{4}$$

For the implementation of Quanser AERO 2-DOF helicopter, $\mathbf{x}(t) = \begin{bmatrix}\theta & \psi & \dot{\theta} & \dot{\psi}\end{bmatrix}'$ and $\mathbf{u}(t) = \begin{bmatrix}\mathrm{V}_p & \mathrm{V}_y\end{bmatrix}'$ where V_p is the applied voltage of the main rotor and V_y is the applied voltage of the tail rotor.