Title
Autonomous Ornithopter Flight with Sensor-Based Seeking Behavior

Author
Baek, Stanley Seunghoon

Publication Date
2011

Autonomous Ornithopter Flight with Sensor-Based Seeking Behavior

by

Stanley Seunghoon Baek

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Ronald S. Fearing, Chair
Professor J. Karl Hedrick
Assistant Professor Pieter Abbeel

Spring 2011

Autonomous Ornithopter Flight with Sensor-Based Seeking Behavior

Abstract

Autonomous Ornithopter Flight with Sensor-Based Seeking Behavior

by

Stanley Seunghoon Baek

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Ronald S. Fearing, Chair

This thesis presents the design of autonomous flight control algorithms for a flapping-wing aerial robot with onboard sensing and computational resources. We use a 13 gram ornithopter with biologically-inspired clap-and-fling mechanism. For autonomous flight control, we have developed 1.0 gram control electronics integrated with a microcontroller, inertial and visual sensors, communication electronics, and motor drivers. We have also developed a simplified aerodynamic model of ornithopter flight to reduce the order of the control system. With the aerodynamic model and the orientation estimation from on-board inertial sensors, we present flight control of an ornithopter capable of flying toward a target using onboard sensing and computational resources only. To this end, we have developed a dead-reckoning algorithm to recover from the temporary loss of the target which can occur with a visual sensor with a narrow field of view. With closed-loop height regulation of the ornithopter, we propose a method for identifying the discrepancy between the tethered flight force measurement and the free flight aerodynamic force. Lastly, we present a nondimensionalized analysis of a motor-driven flapping-wing system and experimentally demonstrate a resonant drive to reduce average battery power consumption for DC motor-driven flapping-wing robots.

To my wife, Jiwon

Contents

List of Figures

List of Tables

Acknowledgments

First and foremost I would like to express my deepest appreciation to my advisor, Professor Ron Fearing, for his constant support and advice since my undergraduate study at Berkeley. Without his guidance, I would have never made this work possible. He has been really supportive of my work not only with his remarkable knowledge and keen insight in the field of robotics, but also with his kind consideration and encouragement. I feel very privileged and thankful to have him as an advisor. Moreover, as a senior in life, he has always been available to give me invaluable advice whenever I needed one.

I would like to thank Professor J. Karl Hedrick, for his invariant consideration and advice, for serving on my qualifying examination committee as well as my dissertation committee. His nonlinear dynamic systems class has also expanded my understanding of control systems. I am very grateful to Professor Pieter Abbeel for serving on my dissertation committee, and especially for his feedback for the thesis. His insightful comments and suggestions have made the thesis much stronger. I would also like to express my appreciation for the efforts of Professor Robert Dudley and Seth Sanders, who served on my qualifying examination committee.

I am thankful to many former and current members of the Biomimetic Millisystems Lab at U.C. Berkeley, Rick Groff, Rob Wood, Ranjana Sahai, Srinath Avadhanula, Carmel Majidi, Erik Stelz, Aaron Hoover, Yohei Maeno, Bryan Schubert, Fernando Garcia Bermudez, Jongho Lee, Paul Birkmeyer, Kevin Peterson, Kevin Ma, Andrew Gillies, Nick Kohut, Andrew Pullin, Jaakko Karras, and Xiao-Yu Fu for their true friendship as well as insightful support and comments for my research.

Most of all, I would like to thank my wife, Jiwon, for all her care and unconditional support in the course of my graduate studies. I am also thankful to my lovely two sons, Jason and Andrew, who have incredibly boosted my morale with their adorable doings and sayings.

Part I

Introduction and Background

Chapter 1

Introduction

Flapping-wing flight has long been a perpetual source of deep mystery and enchantment. The exceptional flight performance of birds and insects has been sufficient to intrigue scientists and engineers. Recent achievements of understanding unsteady aerodynamic forces generated by birds and insects have encouraged engineers to develop biologically inspired flapping-wing micro aerial vehicles. Such small-scale aerial vehicles have great potential for search and rescue, reconnaissance, and surveillance among other fields as they can traverse the highly unstructured environment which larger robots (or terrestrial robots) cannot bypass.

Flapping-wing aerial vehicles are more attractive than fixed-wing aircraft or rotary-wing aircraft at small scales. Fixed-wing aircraft lack hovering capability and have lower agility to avoid indoor obstacles. Rotary-wing aircraft such as helicopters and quadrotors are capable of hovering, and some small-scale quadrotors are very maneuverable [28, 29]. Rotary-wing aircraft, however, suffer from wall-proximity effects and are usually inefficient in the low Reynolds number regime. It could also be argued that flapping-wing aircraft are lower bandwidth systems than rotary-wings aircraft. The bandwidth of the control inputs to maintain lift and stability can be much lower for comparably sized flapping-wing implementations [30]. In terms of efficiency, studies in 2001 and 2002 showed that flight with a propeller, even at small scales, could be more efficient than with flapping wings [11, 44]. However, a more recent study in 2009 shows that optimized flapping-wing motion can save up to 27% of the aerodynamic power required by the optimal steady flight [34].

Though engineers have demonstrated a number of small-scale flapping-wing robots capable of sustained flight, the autonomous flight control of such robots is still nascent. It is mainly due to the inherent high-nonlinearity of such robots and lack of system identification. Also, small-scale flapping-wing robots usually have a minimum number of actuators due to the size constraints, and therefore the systems are generally under-actuated. The goal of this thesis is to design and implement control algorithms for autonomous flight of a flapping-wing robot. We also propose closed-loop flight control as a tool for aerodynamic force analysis near a stable equilibrium point. Lastly, we discuss a method to improve power efficiency by running flapping-wing robots at their resonance to reduce inertial costs of accelerating and decelerating their wings.

1.1 Control Challenges for Flapping-Wing Robots

Autonomous flight of small-scale flapping-wing robots is still under development. Difficulties in the development of autonomous flapping-wing flight at small-scale are outlined below:

1. Most small-scale aerial vehicles developed so far cannot generate enough lift to carry large payloads, such as for rich onboard sensing and computational resources.
2. These robots are usually highly under-actuated because the size constraint limits the number of actuators. They are also highly-coupled multi-input and multi-output (MIMO) systems.
3. Smaller scale flapping-wing air vehicles usually require higher flapping frequencies, which produce high-frequency vibrations. The inertial force generated by flapping motion of the wings is transmitted to the body and makes the body vibrate at the flapping frequency. Slightly asymmetric flapping motion of the wings also produce undesirable lateral acceleration and yaw moment.
4. System identification of such systems is difficult due to the inherent unsteady air flow around the wings. Without good system identification, designing state space feedback control, such as linear quadratic regulator (LQR) control, is very challenging.

Our approaches to autonomous flight control of a small-scale flapping-wing robot include a reduced-order control system using a simplified flight model, passive pendulum-like stability, and a failure recovery algorithm. We have developed a simple aerodynamic model of a flapping-wing robot using time average forces. From a controls standpoint, instantaneous force control during each flapping stroke is generally not possible since the bandwidth of the actuator is usually not large enough to change the wing motions during each stroke. To achieve pendulum-like stability, we have created a CAD (SolidWorks 2010) model to locate the center of mass of the system at the desired location. The location of the center of mass is where we can passively stabilize the robot. A carefully chosen location of the center of mass also minimizes the pitch oscillation. A dead reckoning algorithm has also been developed to recover from a temporary loss of the sensed target which can occur with a visual sensor with a narrow field of view.

1.2 Aerodynamic Forces for Free Flight

An accurate measurement of free flight aerodynamic forces in flapping-wing robots is a recurring issue. This is because enough lift force must be generated while simultaneously achieving a stable equilibrum for user defined flight. Indeed, control engineers are usually interested in the free flight aerodynamic forces of air-vehicles near the stable equilibria. Then, we can measure accelerations of the body in air to estimate the controlled free flight aerodynamic forces for given input signals, such as wing velocity, rudder position, etc.

The aerodynamic forces have been measured with a force/torque sensor, such as a load cell [32] or double cantilevers with strain gauges [50]. Wind tunnel measurements have also been conducted to measure lift and thrust forces simultaneously [16]. Although tethered measurement provides a good approximation of aerodynamic forces before free flight, the difference between the measured forces and the real forces aloft could be unacceptably large. There are several reasons for the difference between free flight and tethered measurements. On the one hand, the fixed body of the ornithopter and structural compliance can give rise to vibrations which either absorb propulsive power or change wing stroke kinematics. On the other hand, estimating flight force directions in free flight is inherently noisy.

1.3 Efficient Energetics Using Resonant Excitation

Efficient power usage is fundamental to the development of small-scale flapping-wing robots for the following reasons:

1. Small-scale robots use portable power sources with limited energy capacity.
2. Aerial vehicles require substantial amounts of power to generate lift force for the robot itself as well as the payload.
3. Autonomous robots require additional power for onboard electronics to process sensor data as well as closed-loop controllers.

One approach to efficient power usage is to use integrated compliant mechanisms driven at their resonant frequency. At the smallest scales, compliant mechanisms, resonant excitation, and piezoelectric actuation work well, due to high speeds, low transmission losses, and high power density. At larger scales, motor actuation is more suitable. Interestingly, most flapping wing robots with motor driven mechanisms do not explicitly use a resonant drive system.

Previous work in resonant drive of flapping wing MAVs has considered minimizing input power assuming constant velocity [41]. However, for heavily loaded structures, motor velocity will not be constant without a speed controller, which takes active power for deceleration. Another related result considers minimizing peak torque [21, 26], which improves motor sizing, yet does not consider the effects of the battery resistance. We show that under certain loading conditions, the battery and motor resistance will influence the time average power required.

Flapping-wing air vehicles can improve efficiency by running at resonance to reduce inertial costs of accelerating and decelerating the wings. For battery-powered, DC motor-driven systems with gears and cranks, the drive torque and velocity is a complicated function of battery voltage. Hence, resonant behavior is not as well defined as for flapping-wing systems with elastic actuators.

Motor-driven, flapping wing MAVs have been built, including the 'Microbat' [17, 35, 36], Kawamura's 2.3 gram MAV [16], the 12 gram Interactive Toys Vamp, and the 'DelFly' [8]. Interestingly, these mechanisms do not explicitly use a resonant drive system. Potentially, resonant excitation can offer increased lift output, with less battery power. However, while analysis of resonant excitation is intrinsically simple for systems driven by piezoelectric actuators (e.g. [12]), it is not straightforward for motor actuated systems. The generation of oscillatory kinematics necessary for flapping-wing flight requires the integration of a crank-arm, which has a fundamental geometric nonlinearity. The integration of compliant mechanisms adds additional nonlinear complexity.

1.4 Contributions

The contributions of this work are as follows:

- **Control algorithm for autonomous ornithopter flight** - We have demonstrated autonomous flight control of a 13 gram ornithopter capable of flying toward a target without any remote assistance. For this demonstration, we have developed a closed-loop attitude regulator for the ornithopter using onboard sensing and computational resources. We have also developed an algorithm for the ornithopter to predict the target location when the target is temporarily not visible.

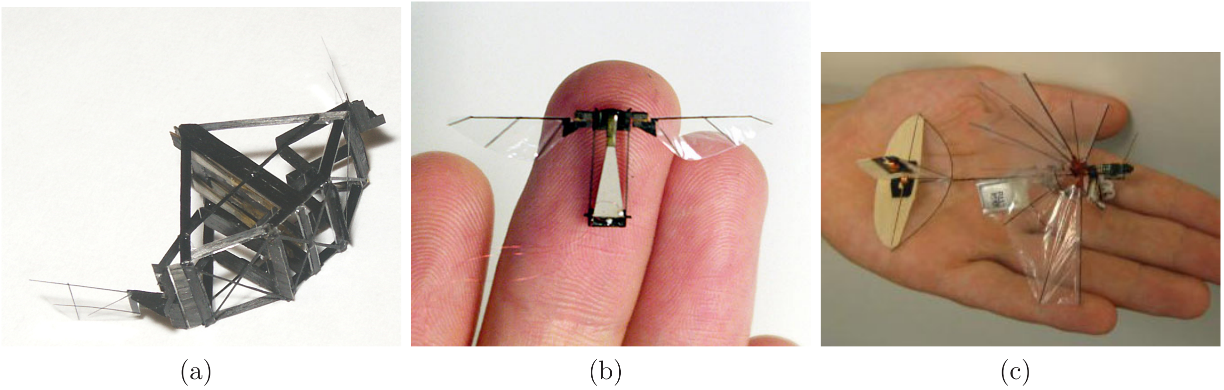
- **Identification of free flight forces of an ornithopter** - We have demonstrated the identification of free flight aerodynamic forces at a stable equilibrium point of an ornithopter and compare them with the tethered flight aerodynamic forces. For the aerodynamic analysis at a stable equilibrium, we have developed a closed-loop altitude regulation for the ornithopter using an external camera. The results show that the tethered aerodynamic force measurement of a 12 gram ornithopter with zero induced velocity underestimates the total flight force by 24.8 mN or 20%.

- **Efficient resonant drive for DC motor-driven ornithopter** - The analysis of a resonant drive has been studied to reduce average battery power consumption for DC motor-driven flapping-wing robots consisting of a compliant element. For this analysis, we have performed a nondimensionalized analysis of a motor-driven slider crank, considering motor and battery resistance. This analysis is used to demonstrate the benefits of an efficient resonant drive on a a flapping-wing robot integrated with a tuned compliant element. The nondimensional analysis can be applied to the generic class of a motor-driven slider crank mechanisms.

- **Lightweight electronics for small-scale robots** - We have developed a 1.0 gram control electronics board consisting of a 40 MHz microprocessor, a 4 MB flash memory, a 6-axis inertial measurement unit, a wireless transceiver, two motor drivers, and two visual sensors. The board can also feature surface mount connectors to accommodate daughter cards to expand the range of sensing and actuating capabilities. The control board is designed for completely autonomous flight of a 10-gram scale ornithopter.

Chapter 2

Previous Work



(a) (b) (c)

Figure 2.1: Previous Work in small-scale flapping-wing robots. (a) The Micromechanical Flying Insect (MFI) at Berkeley [40]. (b) The Harvard Microrobotic Fly [51]. (c) The DelFly Micro [6].

Engineers have made remarkable progress toward the development of flapping-wing micro aerial vehicles. An early attempt at a flapping-wing micro aerial vehicle came from Shimoyama *et al.* [38]. The authors proposed an implementation of a robot using rigid plates and elastic joints to fabricate an external skeleton similar to the structure of an insect. More recently, work from the Biomimetic Millisystems Lab at Berkeley explored using the Smart Composite Microstructures (SCM) process in fabrication of small-scale flapping-wing robots [49]. Fig. 2.1a shows the Micromechanical Flying Insect (MFI), a 100 mg flapping-wing robot fabricated using SCM process with four piezoelectric actuators to drive passively rotating wings [2, 40]. The Harvard Microrobotics Laboratory demonstrated the first take-off of an at-scale robotic insect [48, 51]. Fig. 2.1b shows the 60 mg biologically inspired robot that flaps its wings at 110 beats per second using one piezoelectric actuator.

Since the selection of lightweight batteries at this scale was limited, the development of such milligram-scale flying robots was still focused on tethered flight.

Whereas the milligram-scale flying robots were still tethered, a few successful developments in the flapping-wing robots at larger scales were demonstrated. Van Breugel *et al.* developed a passively stable 24 gram flapping-wing robot capable of hovering [45]. The robot took advantage of the clap-and-fling[1] mechanism, in addition to passive wing bending to simplify the design and enhance performance. Delft University developed a series of flapping-wing robots using clap-and-fling mechanism; DelFly I, DelFly II, and DelFly Micro [6, 7, 23, 24]. Shown in Fig. 2.1c is the latest version of DelFly, the DelFly Micro, which is 3 grams and 10 cm from wing tip to wing tip.

These remarkable developments in small-scale flapping-wing robots have extended the capabilities of flight control for a number of higher level behaviors, such as surveillance and reconnaissance. In the rest of this chapter, we first provide a brief review of the development of autonomous flight control of flapping-wing robots. We will also review a number of approaches to efficient resonant excitation for motor-driven flapping-wing robots integrated with a compliant element.

2.1 Flight Control for Flapping-Wing Robots

Autonomous flight control of flapping-wing MAVs have been studied with diverse approaches. Deng *et al.* [9, 10] designed LQR-based feedback laws for hovering control using an approximate time-invariant model of the Micromechanical Flying Insect [12] with time averaged aerodynamic forces. Khan *et al.* developed a differential flatness-based nonlinear controller for a time-averaged system that could be applied to flapping-wing MAVs [20]. Other development of closed-loop control for a flapping-wing MAVs was demonstrated by Shigeoka [37]. Acquiring acceleration and position data using an accelerometer and a Vicon motion tracking system, Shigeoka developed a simple dynamic model for a flapping-wing MAV and demonstrated simulation results using frequency analysis. Their approaches performed well in computer simulations, yet they have not been implemented in a real flapping-wing robot.

Some of the significant research on the development of autonomous flight in a real flapping-wing robot is summarized here. Tedrake *et al.* [43] demonstrated autonomous flight of a 2-meter wingspan ornithopter with enough payload to carry a 400 gram sensor and computer package. For small-scale ornithopters, engineers developed flight controllers using a ground station for computationally expensive processes such as video processing. De Croon *et al.* implemented an obstacle avoidance algorithm for a 16 gram flapping wing DelFly II with two onboard cameras [6, 7]. In their work, a ground station personal computer was used to process sensor data and compute control signals. For fixed-wing small-scale aerial vehicles,

---

[1] In the clap-and-fling mechanism, the wings clap together and then fling apart. A study shows that the clap-and-fling mechanism can significantly increase lift coefficients, especially for small insects [31].

Zufferey *et al.* demonstrated fully autonomous flight of a 10 gram aerial vehicle by mapping optical flow signals into control commands in an indoor environment [53]. The developments of autonomous flight discussed so far, along with the autonomous ornithopter flight in this thesis, are summarized in Table 2.1.

Table 2.1: Developments of autonomous flight

Developers	Specifications	Wings	Control	Design Objectives
Tedrake *et al.*	2.0 m wing span	flapping	onboard	stable flight
De Croon *et al.*	16 grams	flapping	remote	obstacle avoidance
Zufferey *et al.*	10 grams	fixed	onboard	visual navigation
Baek (Chapter 5)	13 grams	flapping	onboard	target seeking

2.2 Resonant Drive for Flapping-Wing Robots

One approach to efficient power usage in flapping-wing MAVs is to use integrated compliant mechanisms driven at their resonant frequency. Goldfarb developed piezoelectric-actuated flapping-wing MAVs excited at the resonant frequency [4,13,25]. Agrawal demonstrated a novel, motor-driven four-bar crank-rocker mechanism that integrated a spring to store energy during a flapping cycle [19,20,26,27]. Resonant drives and compliant mechanisms have been implemented in 100 mg insect-scale wing flapping mechanisms [1,2,12,40,52] as well as a 5.8 gram hummingbird-inspired flapping mechanism [3]. Recently, Wood [47,48,51] demonstrated lift-off of an insect-scale 60 mg flapping-wing MAV, employing a 110 Hz wing beat excited at the resonant frequency with a piezoelectric actuator.

Some of the significant research on the integration of resonant drive mechanisms in DC motor driven flapping-wing MAVs is summarized here. Khatait *et al.* [21] demonstrated a four-bar crank-rocker mechanism that minimized the peak torque requirement of the DC motor by optimizing the compliant elements of the structure consisting of two flexural hinges. The authors achieved a reduction of the order of 2.85 in the peak driving torque in a specific mechanism considered. Tantanawat and Kota [41] proposed integrating distributed compliance to reduce the peak input power requirement of the DC motor on a flapping mechanism. In their study, the peak power needed to drive the proposed compliant flapping mechanism is found to be 50% less than a rigid-link four-bar flapping mechanism without a spring, and 15% less than the one with a spring. Madangopal *et al.* [27], using nonlinear optimization tools, designed a flapping mechanism with springs attached to the load to reduce the variation in motor torque. The authors demonstrated 56.32% reduction in the variation of the peak torque.

Previous work [21,27,41], discussed above, uses peak input power as a design metric. But for mobile battery-powered devices, average input power takes precedence over peak power since average power is directly related to battery capacity. The previous work also assumes the motor is operating at constant velocity, which is difficult to maintain with high cyclic loads. A battery as a voltage source with an internal battery resistance is more representative of a practical system, rather than assuming constant velocity operation. This is due to the fact that a closed loop speed control requires additional power and components, such as speed sensors and H-bridges, and may require excess peak torques to maintain constant velocity with large load variations.

Part II

Flight Control

Chapter 3

Ornithopter Platform and Aerodynamic Model

In this chapter, we present the ornithopter platforms, the light-weight control electronics, and the PID controller used for the rest of this thesis.

3.1 Ornithopter Platform

The ornithopter used in this thesis has been obtained by modifying from a Silverlit Wing-Master I-Bird as shown in Fig 3.1. From the original 4 winged ornithopter, the styrofoam outer body has been removed to reduce the weight. The removal of the styrofoam, however, results in undesirable bending and torsional compliance of the structure that absorbs some of the power to be transferred to the environment. As a result, the aerodynamic forces generated by the wings are significantly reduced. To compensate for this, we have replaced the main backbone frame with stiff carbon fiber T-beams so that the aerodynamic forces remain the same.

Another modification was to replace the rudder at the end of I-Bird's tail with a propeller. The rudder was driven by a magnetic actuator to the left or right direction for yaw moment. One of the problems of the magnetic actuator was that it required too much current ($\sim$300 mA), and the ornithopter could not generate enough lift force when the magnetic actuator was activated. Another problem was that it had only tristate modes, i.e., full left, full right, or neutral, which limited the control policy to be bang-bang control. For these reasons, we replaced the rudder with a bidirectional propeller driven by a 4 mm diameter pager motor. After this modification, the ornithopter consumed less than 100 mA current at the maximum speed of the propeller while the propeller generated much higher yaw moment than the rudder. It also enabled us to implement a PID controller for yaw regulation instead of a bang-bang control. The specifications of the modified ornithopter are summarized in Table 3.1.

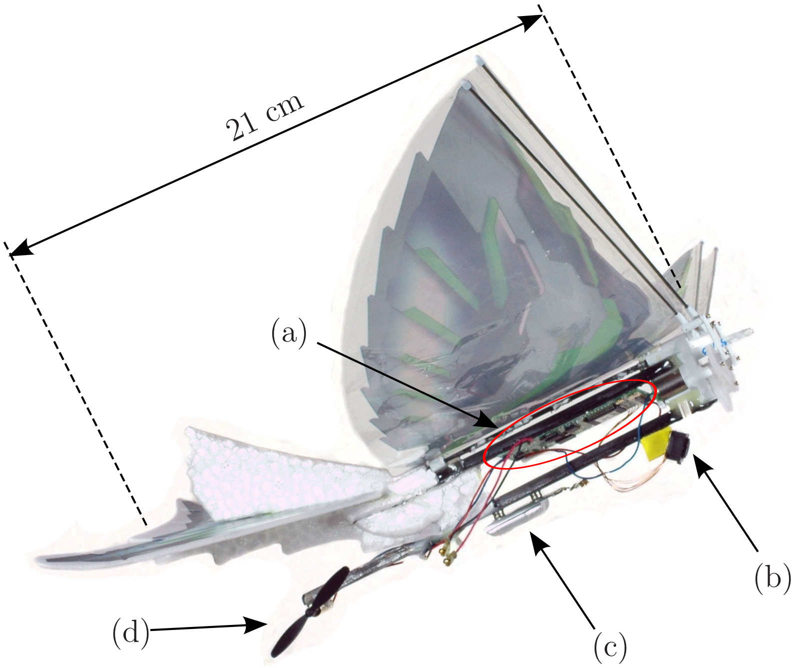
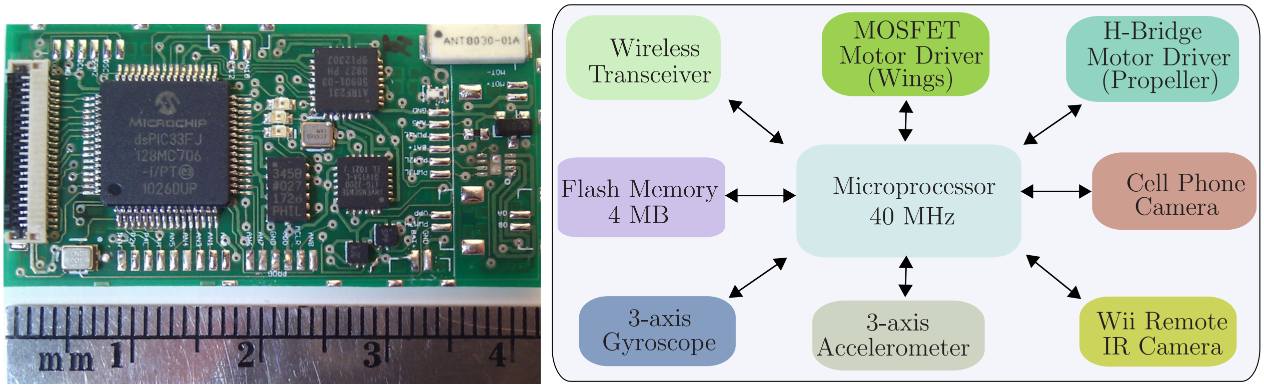


Figure 3.1: The modified ornithopter equipped with (a) control electronics, (b) a visual sensor, (c) a battery, and (d) a propeller.

Table 3.1: Specifications of the Modified Ornithopter

Wing span	28 cm
Body length	21 cm
60 mAh Li-Poly Battery	1.8 grams
Motor & gear box for wings	2.8 grams
Steering motor & propeller	0.9 grams
Wings	1.8 grams
Frames & tail	2.6 grams
Control board	1.0 grams
Infrared camera	0.33 grams
Connectors & wires	2.1 grams
Total Weight	13.4 grams



(a) (b)

Figure 3.2: The 1.0 gram control board consisting of a 40 MHz dsPIC microprocessor, a 4MB flash memory, a 3-axis accelerometer, a 3-axis gyroscope, an 802.15.4 wireless transceiver, a 2-channel motor driver, an IR camera connector, and a cell phone camera connector. (a) the control board with a ruler (b) functional block diagram for the control board

3.2 Design of 1.0 gram Control Electronics

For small-scale flapping-wing robots, lightweight control electronics is a crucial design consideration since such small-scale robots generally cannot carry a large payload. One of the solutions for lightweight electronics would be custom integrated circuits (ICs), but the cost of IC developments is generally very high. A more realistic solution is to build control electronics on a custom lightweight printed circuit board (PCB). With careful design of PCB and selection of electric components, we have been able to significantly reduce the overall weight of the robot.

Shown in Fig. 3.2a is the 1.0 gram[1] control board consisting of a Microchip 16 bit dsPIC microprocessor running at 40 MHz, a 4MB flash memory, a 3-axis accelerometer, a 3-axis gyroscope, an 802.15.4 wireless transceiver, a 2-channel detachable motor driver consisting of a MOSFET and an H-Bridge, an infrared camera connector, and a cell phone camera connector. The blank printed circuit board was manufactured using 25 μm thick core FR4 fiberglass for light-weight robotic applications. The board can also feature surface mount connectors to accommodate daughter cards to expand a range of sensing and actuating capabilities. The control board is designed for completely autonomous flight of an ornithopter, i.e., there are no remote control inputs or external control signals during flight. The wireless transceiver is only used for telemetry. Figure 3.2b shows a functional block diagram for the control board.

---

[1] The weight is measured without visual sensors. The infrared camera is 0.33 grams, and the cell phone camera is 0.30 grams.

The control board contains a 40 MIPS dsPIC33FJ128MC706 microprocessor running at 3.3 V through a MIC5335 low dropout linear regulator. The dsPIC microprocessor has been chosen for its low power consumption and for a range of robotic applications that require a number of peripherals - we are using 2 I^2C channels, 2 SPI channels, 1 UART channel, and 4 PWM channels. The 16 kilobytes internal SRAM is mainly used for buffers to temporarily store packets received or to be transmitted through the wireless communication link. The real-time data such as sensor outputs, state estimates, and control input signals for later review are saved in the Atmel 32-megabit DataFlash memory (AT45DB321D) interfaced via an SPI channel at the speed of 10 MHz. Using this flash memory, we can prevent important data from being lost when the microcontroller is reset or the real-time communication via wireless link is broken.

The board contains a complete implementation of the IEEE 802.15.4 standard for low-rate wireless personal area networks. The Atmel low power 2.4 GHz transceiver (AT86RF231) is utilized for the wireless communication with a host computer. The transceiver is interfaced with the dsPIC processor via an SPI channel at the speed of 6 MHz and communicates with the host computer at the effective speed of 115kbps. We have also developed a USB dongle for the host computer using the same transceiver and dsPIC processor so that the computer can communicate with the control board through the dongle. The interface to the robot that enables initiating flight and retrieving the data saved in the flash memory is written in Python. The software also communicates with the robot to configure control parameters, filter coefficients, etc.

For inertial measurements, a 3-axis gyroscope and a 3-axis accelerometer are interfaced with the control board via the I^2C channels at the speed of 400 KHz. The sampling rate of the inertial sensors can go up to 1 KHz and the sensors have configurable internal low pass filters for each axis. The sensors are calibrated for every power recycle and reset operation.

An OmniVision OV7660FSL VGA camera can be connected to the board for visual navigation such as optical flow, collision avoidance, target following, etc. [14]. The camera is initialized via an I^2C channel, and the pixel data is transmitted via parallel IO ports. An infrared camera can be connected to the control board through the other I^2C channel. The infrared camera, originated from a Nintendo Wii remote controller, can detect up to 4 infrared emitters at a time and transmits the positions of the emitters at 400 Hz in an image frame of 1024×768 along with the emitters' intensity. The sampling rate of the infrared camera can go up to 1 KHz. More details for this infrared camera can be found in [46].

The motor drivers have a MOSFET and an H-Bridge making it capable of driving an unidirectional motor as well as a bidirectional motor at the same time. The MOSFET is driven by a PWM signal to drive the main motor to generate flapping motion of wings, and the H-Bridge is driven by a PWM signal and a general I/O port to drive the bidirectional propeller (or the rudder) to generate yaw moment. The back-EMF voltage for the motor connected to the MOSFET can be measured using an ADC channel. An RC low pass filter is used to remove the noise for the measurement. The back-EMF measurement is used to not only regulate the motor speed but also estimate the pitch oscillation of the ornithopter.

The motor driver is designed to be removed and replaced with a surface-mount connector so that daughter boards with a range of motor drivers can take the place of the existing motor drivers.

A 60 mAh Lithium-Polymer battery provides power for the electronics, the DC motors to drive the flapping wings and the propeller (or the magnetic actuator to steer the rudder). The DC motor can drive the wings at 17-20 Hz depending on battery charge. The battery has been carefully mounted on the body of the ornithopter so that we can achieve pendulum-like passive stability.

Three LEDs are used to indicate various status modes of the board, and a MN1380 battery monitor is used to indicate the battery voltage status. A number of connectors for ADC channels, general input/output ports, and a UART channel make the board expandable for other micro-robotic applications. In order to reduce the weight of the board, we have also implemented a bootloader so that the microcontroller can be programmed via the wireless communication link without a connector for a hardware programmer.

3.3 PID Controller for Orientation and Height Regulation

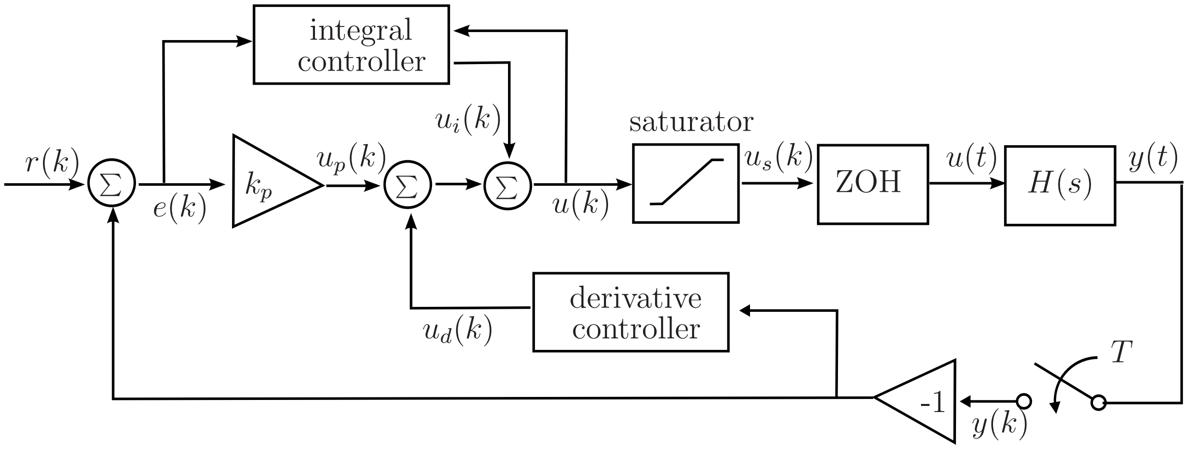


Figure 3.3: Block diagram for the PID controllers. The integral controller consists of the integral gain and a discrete-time integral anti-windup routine, and the derivative controller consists of a difference equation and a low pass filter.

The closed-loop height regulation and orientation regulation used in this thesis are implemented by a PID controller, given by

$$u = k_p(r - y) + k_i \int_0^t (r(\tau) - y(\tau))d\tau + k_d\left(\gamma \dot{r} - \dot{y}\right), \tag{3.1}$$

where u is the duty cycle for a motor, k_p, k_i, and k_d are, respectively, the proportional, integral, and derivative gains for the PID controller, r is the desired input (desired height or orientation angles), y is the measured (or estimated) output, and γ is the reference weight for the derivative controller. We set $\gamma = 0$ to avoid large transients in the control signal when the reference is changed.

We have implemented a discrete-time PID controller for (3.1) running in the control board discussed in Sec. 3.2. We have implemented a timer that generates one interrupt every T seconds. The discrete-time PID controller written in the C programming language is executed by the interrupt service routine (ISR) for each timer interrupt triggered. The block diagram shown in Fig. 3.3 represents the discrete time realization of the PID controller.

The derivative controller in Fig. 3.3 consists of a difference equation and a low pass filter, which is given by

$$u_d(k) = k_d \cdot f_{lpf}(\frac{y(k-1) - y(k)}{T}), \tag{3.2}$$

where $f_{lpf}(\cdot)$ is a discrete-time Butterworth[2] low pass filter written in the C programming language. The low pass filter has been implemented to reduce the high frequency noise caused by the difference equation. The filter order and cutoff frequency have been determined based on examinations of experimental data. The coefficients of the filter have been obtained using the Matlab *butter* function.

The integral controller has been implemented with a discrete-time anti-windup algorithm by testing actuator saturation. If the input duty cycle, denoted by $u(k)$, is not saturated, we want to accumulate the error for the integral controller. Otherwise, the accumulated error should remain unchanged. The integral controller is summarized by

$$u_i(k+1) \quad = \quad \begin{cases} u_i(k) & \text{if } u(k) > u_{max} \text{ or } u(k) < u_{min}, \\ u_i(k) + k_i T(r(k) - y(k)) & \text{otherwise,} \end{cases}$$

where u_{max} is the maximum input and u_{min} is the minimum input.

The saturator block in Fig. 3.3 is realized in the C programming language to keep the duty cycle to the system, $u_s(k)$, in the user defined ranges, (u_{min}, u_{max}). We want $u_s(k)$ $\in$ (-100%, 100%) for the propeller control and $u_s(k) \in$ (60%, 100%) for the flapping-wing motor. Note that u_{min} for the flapping-wing motor is 60% not 0%. This constraint prevents the ornithopter from descending too quickly.

---

[2] We have chosen the Butterworth filter among various signal processing filters because it has as flat a frequency response as possible in the passband.

Chapter 4

Flight Forces and Altitude Regulation

In this chapter, we will demonstrate closed-loop altitude regulation for an ornithopter using an external camera and an onboard microprocessor. With its rudder fixed in one direction, the ornithopter always orbits around in front of the camera with a regulated altitude. With this stable height regulation, we will discuss a method of identification of free-flight aerodynamic forces near the stable equilibrium. Then, we will compare these forces with the tethered flight forces measured with a load cell.

4.1 Aerodynamic Model

4.1.1 Aerodynamic Forces

Figure 4.1 depicts the coordinate systems and the forces acting on the body of the ornithopter. The origin of the body coordinates ($\mathbf{e}_t$, $\mathbf{e}_n$) is attached to the center of mass of the ornithopter. The tangential axis of the body coordinate, $\mathbf{e}_t$, is defined as the line from the center of mass, $\mathbf{c}$, passing through the point, $\mathbf{r}$, on which the resultant aerodynamic forces are acting, and the normal axis of the coordinate, $\mathbf{e}_n$, is perpendicular to the tangential axis. The equations of motion in the body coordinate are given by

$$\begin{aligned} m\ddot{x}_b - m\dot{\theta}\dot{y}_b &= -mg\sin\theta + F_T, && (4.1)\\ m\ddot{y}_b + m\dot{\theta}\dot{x}_b &= -mg\cos\theta + F_N, && (4.2)\\ I\ddot{\theta} &= dF_N, && (4.3) \end{aligned}$$

where F_T is the tangential force in the $\mathbf{e}_t$ direction, F_N is the normal force in the $\mathbf{e}_n$ direction, x_b is the displacement in the $\mathbf{e}_t$ direction, y_b is the displacement in the $\mathbf{e}_n$ direction, θ is the angle between $\mathbf{e}_t$ and $\mathbf{E}_1$, d is the distance between the center of mass and the origin of the forces, m is the mass of the ornithopter, I is the moment of inertia, and g is the gravitational acceleration. The equations of motion can also be written in the inertial frame ($\mathbf{E}_1$, $\mathbf{E}_2$) fixed

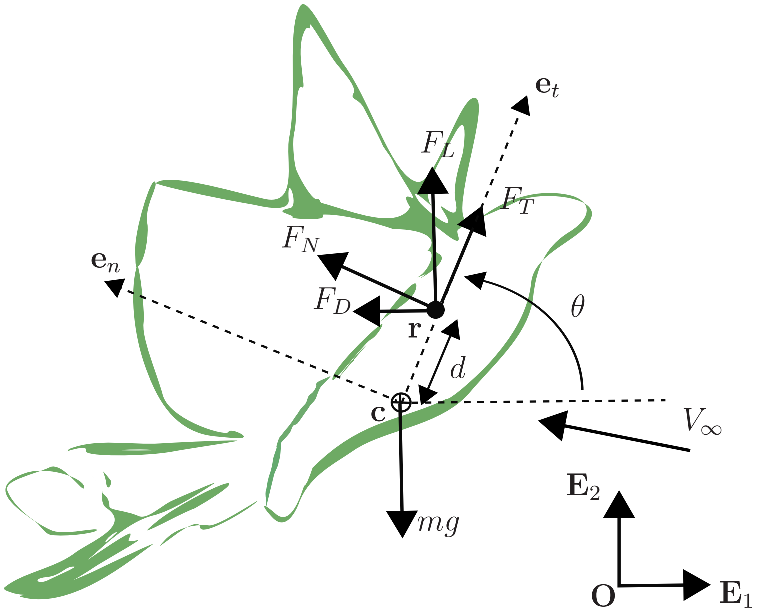


Figure 4.1: The coordinate systems and the forces acting on the body of ornithopter.

to the ground as follows:

$$m\ddot{x} = -F_D, \tag{4.4}$$
$$m\ddot{y} = -mg + F_L, \tag{4.5}$$
$$I\ddot{\theta} = dF_L \cos\theta + dF_D \sin\theta, \tag{4.6}$$

where x is the displacement in the $\mathbf{E}_1$ direction, y is the displacement in the $\mathbf{E}_2$ direction, F_L is the lift force in the $\mathbf{E}_2$ direction, and F_D is the drag force in the $-\mathbf{E}_1$ direction. The relationship between $(F_L,\, F_D)$ and $(F_T,\, F_N)$ is given by

$$F_L = F_T \sin\theta + F_N \cos\theta \tag{4.7}$$
$$F_D = -F_T \cos\theta + F_N \sin\theta \tag{4.8}$$

It is important to note that the aerodynamic forces, $\mathbf{F} = (F_T,\, F_N,\, F_L,\, F_D)$, are functions of the wing velocity ω, the induced velocity V_∞, and the angle of attack α (the angle between the tangential axis and the direction of the induced velocity). Furthermore, $\mathbf{F}$ can be decomposed into the following two functions

$$\mathbf{F} = \mathbf{F}(V_\infty, \omega, \alpha) = \mathbf{F}^w(\omega) + \mathbf{F}^i(V_\infty, \alpha)$$

where $\mathbf{F}^w(\omega)$ is the force solely generated by the flapping motion of the wings while the body of the ornithopter is stationary, i.e., $\dot{x}_b = 0$ and $\dot{y}_b = 0$, and $\mathbf{F}^i(V_\infty, \alpha)$ is the aerodynamic force *induced* by the motion of the body. Whereas $\mathbf{F}(V_\infty, \omega, \alpha)$ is the total resultant force acting on the body for free flight, $\mathbf{F}^w(\omega)$ is the tethered flight force that can be measured using conventional force/torque sensors, such as a load cell. Hence, $\mathbf{F}^i(V_\infty, \alpha)$ is the discrepancy between the real flight force and the tethered flight force. This is the main difference between fixed wing aerodynamic forces and flapping wing aerodynamic forces where fixed wing lift and drag forces are the functions of only the induced velocity and the angle of attack[1]. While both lift force and drag force become zero with $V_\infty = 0$ for fixed wing flight, the resultant force acting on a flapping wing body is still nonzero with $V_\infty = 0$. In this case, $\mathbf{F}^w(\omega)$ still produces the lift force for hovering while $\mathbf{F}^i(V_\infty, \alpha) = 0$.

4.1.2 Passive Stability

The equilibria of the system can be found with $(\dot{\phi}, \ddot{x}, \ddot{y}) = (0, 0, 0)$ or $(\dot{\phi}, \ddot{x}_b, \ddot{y}_b) = (0, 0, 0)$. From (4.1), (4.2), and (4.3), we can find only one equilibrium at $F_N = 0$, $\phi = 90°$, and $F_T = mg$ (or $F_L = F_T = mg$ and $F_D = F_N = 0$). It is important to note that the system is pendulum-like passive stable at this equilibrium. Basically, for any small disturbance in the $\mathbf{e}_n$ direction, the body will start oscillating about the pitch axis and gradually approach $\phi = 0$ with a slight fluctuation of $\dot{x}$ and $\dot{y}$. Therefore, F_N should be kept as small as possible to minimize pitch oscillation. An additional approach to minimize pitch oscillation is carefully choosing the location of the center of mass so that the resultant force on the body directs toward $\mathbf{e}_t$ and the mean value of F_N vanishes. A stabilizer (or elevator) could be combined at the tail to actively damp out oscillations, but it requires more weight and more power.

Assuming that the system stays near the stable equilibrium point, we are less interested in the identification of I and d since $\ddot{\phi} \simeq 0$. On the other hand, the identification of free flight aerodynamic forces is still important. With closed-loop velocity control, we can easily estimate the force discrepancy, or aerodynamic induced force, $\mathbf{F}^i$, with constant $\dot{y}$ and $\dot{x}$. For example, with $\ddot{y} = 0$ and $\ddot{x} = 0$ from (4.4) and (4.5), we have $F_D = 0$ and $F_L = mg$. Consequently, we have

$$F_L^i(V_\infty, \alpha) = mg - F_L^a(\omega) \tag{4.9}$$

$$F_D^i(V_\infty, \alpha) = -F_D^a(\omega) \tag{4.10}$$

4.2 Force Measurements

The propulsive force, $\mathbf{F}^a(\omega)$, has been measured using a force sensor (ATI Nano 43). Fig. 4.2 shows the force measurement setup for the ornithopter with the 6-axis load cell. Using this setup, we have measured the propulsive flight forces, F_T^a and F_N^a.

---

[1] The lift force is $L = \frac{1}{2}C_L(\alpha)\rho u^2 S$ and the drag force is $D = \frac{1}{2}C_D(\alpha)\rho u^2$ S, where α is the angle of attack, u is the induced velocity, S is the reference area, and ρ is the air density which is constant for a small range altitude flight.

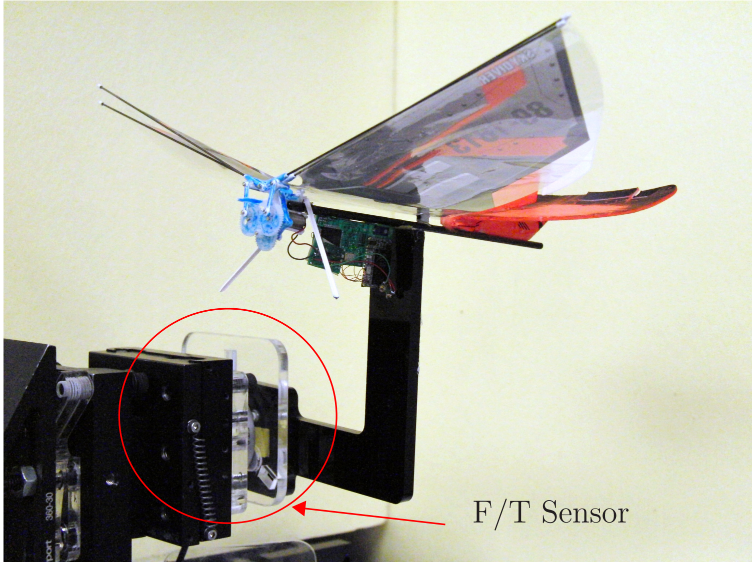


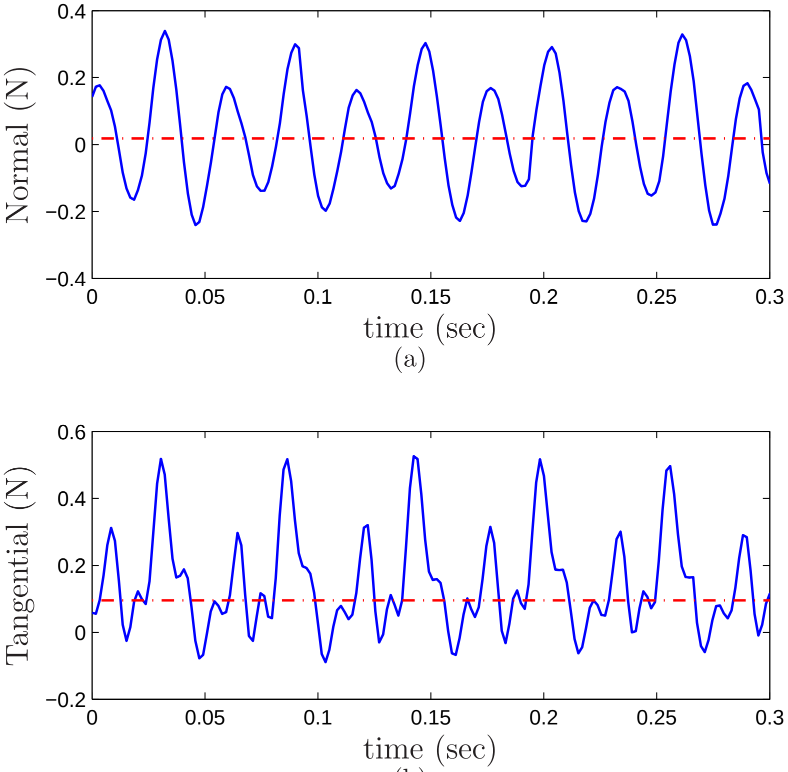
Figure 4.2: Force measurement setup for the ornithopter with a load cell. The forces of the L bracket have been subtracted.

An example of the measured thrust forces with 80% PWM duty cycle is shown in Fig. 4.3. The fundamental frequency of the forces is indeed the flapping frequency of the wings. Although the peak to peak values are almost 0.6 N, the time average normal force and tangential force are only $F_N^a = 18.15$ mN and $F_T^a = 94.95$ mN, respectively. The total average force of 96.7 mN predicts the 12.4 gram ornithopter cannot lift itself at 80% duty cycle if $V_\infty = 0$.

Shown in Fig. 4.4 are the time average normal force, tangential force, and flapping speed of wings with various duty cycles. For these average values, we took steady state force measurement for 10 seconds with a 600 Hz sample rate. The circles in the plots are the measured data points and the solid lines are the first order polynomial fitted lines. The wing speed as well as forces are linear functions of the duty cycles in the range between 50% and 100% as shown in the plots.

4.3 Tracking and Control Methods

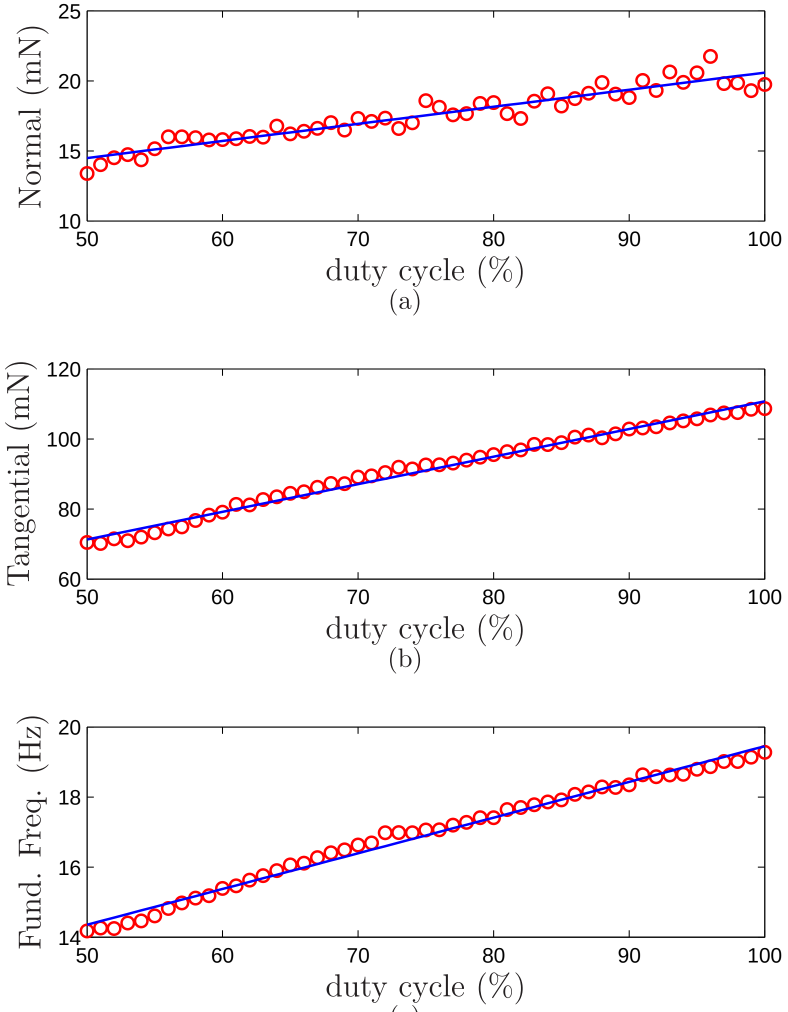
An external camera with the frame rate of 15 Hz and the resolution of 320×240 is connected to a PC to track the ornithopter. The resolution of 320×240 pixels with average distance of 3.8 meters from the camera gives pixel resolution of approximately 1.2 cm at the ornithopter. The tail of the ornithopter is painted in orange, and the Hue, Saturation, and Value (HSV) color model [39] is used to extract the orange blob from the image captured from



(a)

(b)

Figure 4.3: Measurement of normal force (a) and tangential force (b) with 80% duty cycle. The average forces are 18.15 mN for normal and 94.95 mN for tangential, as represented by the dashed-dotted lines.



(a)

(b)

(c)

Figure 4.4: The average normal force (a), average tangential force (b), and flapping speed of wings (c) with various duty cycles. The circles are the measured data points and the solid lines are the first order polynomial fitted lines.

(a) (b)

Figure 4.5: A camera captured image of a free flying ornithopter (a) and the orange blob of the ornithopter found by the blob detection algorithm (b).

the camera. The appropriate values of HSV depend on lighting conditions in an environment, and so they need to be chosen at the beginning of each experiment. Unlike the RGB color model, however, the Value component is the only variable in HSV that needs to be adjusted in most trials.

To minimize the time spent for image processing, a Kalman filter has been implemented to predict the next blob position as follows:

$$\begin{aligned}\mathbf{x}(k+1 \mid k) &= \mathbf{x}(k) + T\mathbf{u}(k) + \mathbf{w}(k),\\ \mathbf{y}(k) &= \mathbf{x}(k) + \mathbf{v}(k),\end{aligned}$$

where $\mathbf{x}$ is the position vector, $\mathbf{u} = (\mathbf{x}(k) - \mathbf{x}(k-1))/T$ is the velocity vector, $T = 67$ ms is the time interval, $\mathbf{y}$ is the measured position vector, and $\mathbf{w}$ and $\mathbf{v}$ are independent white Gaussian noise with $\sigma_w = 34.9$ cm/s and $\sigma_v = 1.72$ cm, respectively. Here, $\mathbf{x}(k+1 \mid k)$ is the prediction of the next blob position. With this prediction, the blob detection algorithm based on OpenCV (Open Source Computer Vision) searches for an orange blob only in a small rectangular region (50×50 pixels or approximately 30×30 cm) centered at the predicted point by the Kalman filter. Fig. 4.5a shows an image of the ornithopter taken by the external camera, and Fig. 4.5b shows the blob detected by the algorithm with $0 \leq \text{Hue} \leq 13$, $95 \leq \text{Saturation} \leq 255$, and $133 \leq \text{Value} \leq 255$.

Once the ground station estimates the center of the location of the ornithopter, the reference error, $e(k)$, between the estimated position and the desired position is computed. Then, the error value is transmitted to the ornithopter through the wireless communication link. The onboard microprocessor takes this data and computes the duty cycle of PWM signal using the PID controller discussed in Section 3.3. The PID controller gains with $k_p =$

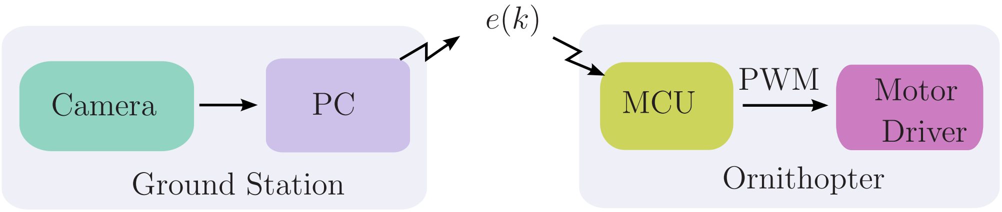


Figure 4.6: Signal flow for closed-loop altitude regulation

0.35, $k_i = 0.2$, and $k_d = 0$ have been determined by a trial and error method. The low pass filter for derivative controller discussed in Section 3.3 has not been used for this work. The microprocessor, then, generates the PWM signal to run the main flapping DC motor. The signal flow of this control law is summarized in Fig. 4.6.

4.4 Experimental Results and Discussion

With an external camera as an altitude sensor, we have successfully achieved closed-loop height regulation of the ornithopter. The rudder is fixed in one direction so that the ornithopter always orbits around in the same direction in front of the camera. Fig. 4.7 shows a step response of the closed-loop system. The lateral motion (in the $\mathbf{E}_1$ direction) shows a pure sinusoidal curve with a slowly varying DC term, which indicates a lateral drift due to external disturbances, such as an indoor breeze. The sinusoidal curve with a nearly constant frequency and constant peak-to-peak amplitude indicates a constant speed in lateral direction. The longitudinal displacement and the desired altitude are also shown. The maximum error in the plot is only about 10 cm, which is located at $t \approx 18$s. The error after $t = 20$s. is less than 3 cm. The average duty cycle for the DC motor is 80% for this experiment. A sequence of video frames in the range between $t = 32.51$s and $t = 33.53$s is shown in Fig. 4.8. The solid cyan circles are the locations of orange tails detected by the blob tracking algorithm and the white horizontal lines in the middle of frames indicate the desired altitude.

The orbital motion of the ornithopter was the result of the force on the rudder fixed in one direction, not the result of the wing motions or the induced velocity. With a nearly constant lateral speed, therefore, it is still safe to assume that $\ddot{x}$ due to F_L and F_D is close to zero. In other words, the centripetal acceleration is provided solely by the force on the rudder, and the other component of the acceleration in the tangential direction is still zero. Also the average longitudinal acceleration in the range between $t = 25$s and $t = 33$s is close to zero[2]. From this result and the inspection of images taken by the camera, we know that

---

[2] Due to the vibratory nature of flapping mechanisms, there are always nonzero body accelerations at a high frequency, but the accelerations at a low frequency (< 1 Hz) at hovering could be zero.

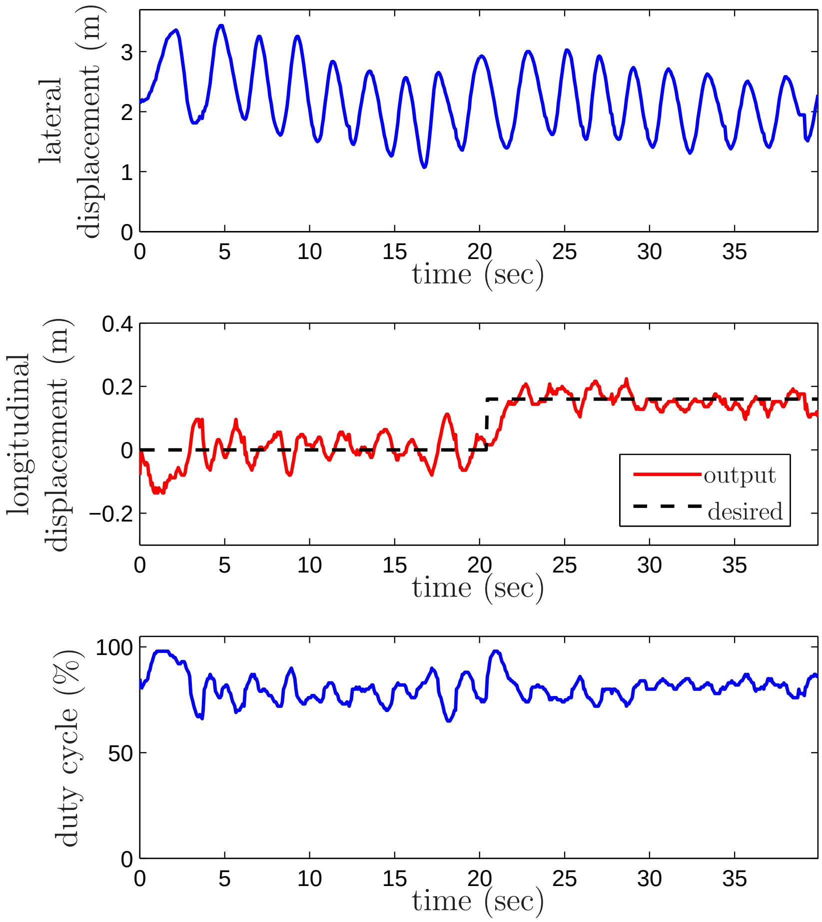


Figure 4.7: Closed-loop step response of height regulation. (top) lateral displacement, (middle) longitudinal displacement and desired altitude, and (bottom) duty cycle for the driving DC motor.

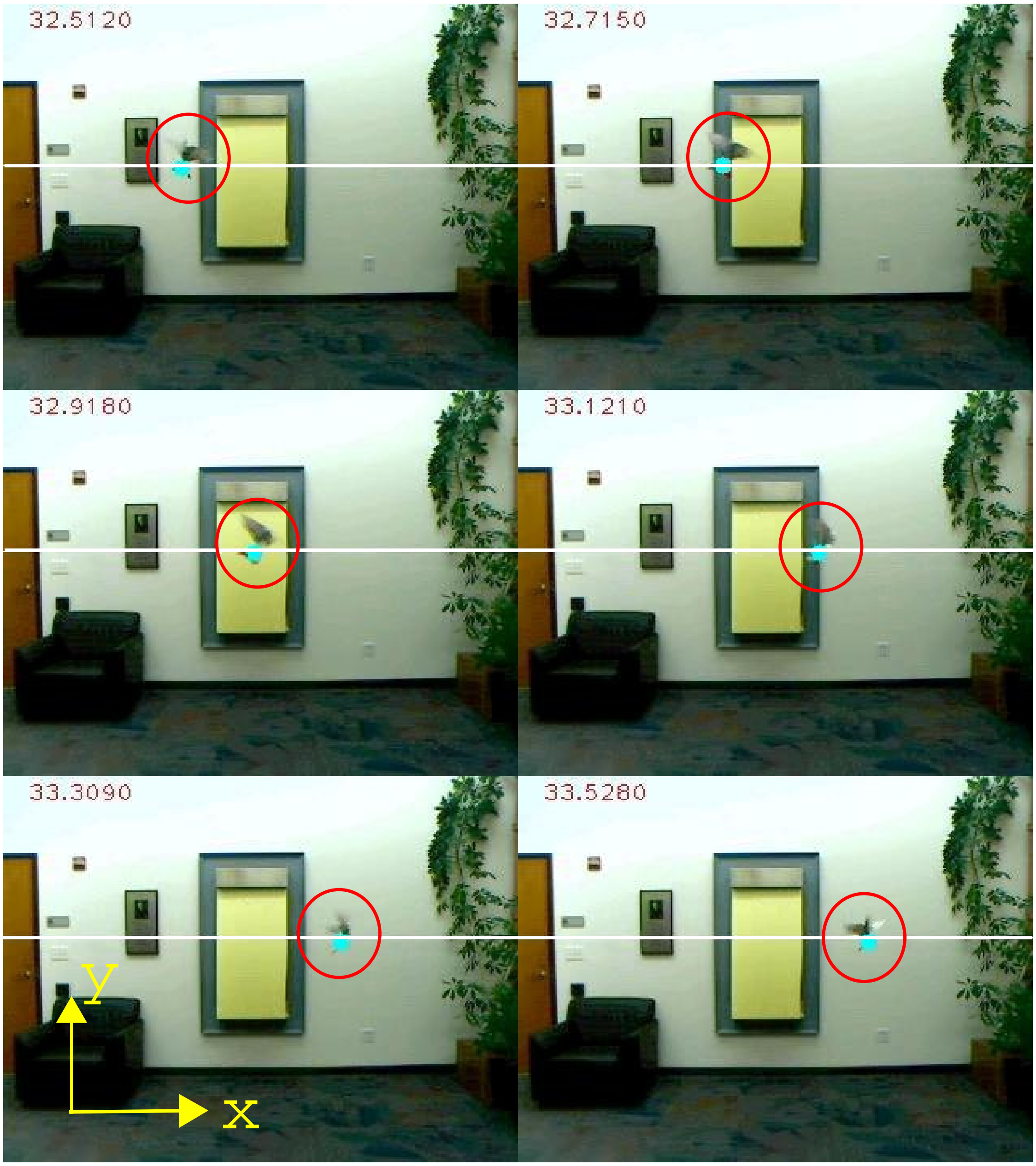


Figure 4.8: A sequence of video frames shows successful altitude control. The solid cyan circles are the locations of orange tails detected by the blob tracking algorithm. The white horizontal lines in the middle of frames indicate the desired altitude.

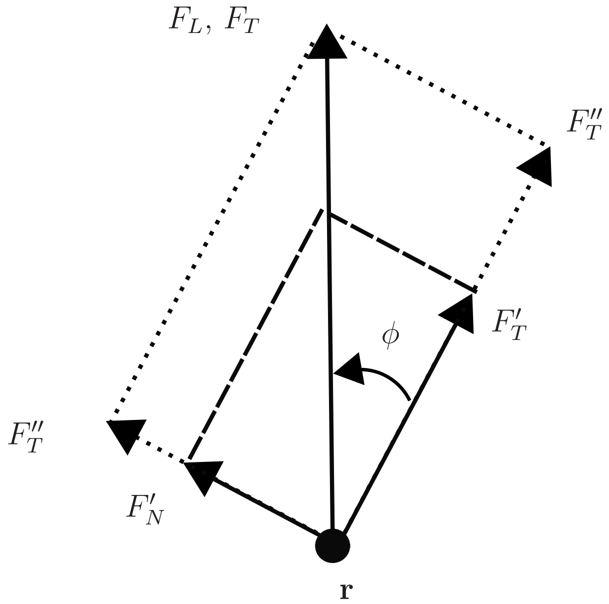


Figure 4.9: Diagram of actual forces and measured forces

$\dot{\theta} = 0$ in average and the free flight aerodynamics forces are $F_D = 0$ and $F_L = mg = 121.5$ mN.

As discussed in Section 4.1, we have $\theta = 90°$, $F_L = F_T = mg$ and $F_D = F_N = 0$ at the equilibrium point. From the load cell measurement, however, we have $F_N^w = 18.15$ mN and $F_T^w = 94.95$ mN. The difference here indicates that (1) the force vectors for measurement did not agree with the actual tangential and normal force vectors (2) the tethered measurement underestimated the flight forces. Fig. 4.9 shows the actual force vectors and the measured force vectors (F'_T, F'_N). In order to fix the measurement, we must first find the angle ϕ between the actual flight forces and measured flight forces. We have $\phi = \tan^{-1}(F'_N/F'_T) = 10.8°$ in this case. The magnitude of the sum of two vectors (F'_N, F'_T) is 96.7 mN. From this result, we know that tethered measurement with zero induced velocity underestimates the lift force by F_L^i = 121.5 - 96.7 = 24.8 mN at 80% duty cycle. F''_T -F'_T and F''_N-F'_N are the underestimated forces in the measurement coordinates.

Shown in Fig. 4.10 are the response of height regulation with an added mass of 0.34 grams. It shows an oscillatory longitudinal response due to the changes in pitch angle. Because of the added mass, the system needs to generate a larger lift force, which is coupled with the drag force. This larger drag force results in larger oscillatory behavior of pitch angle.

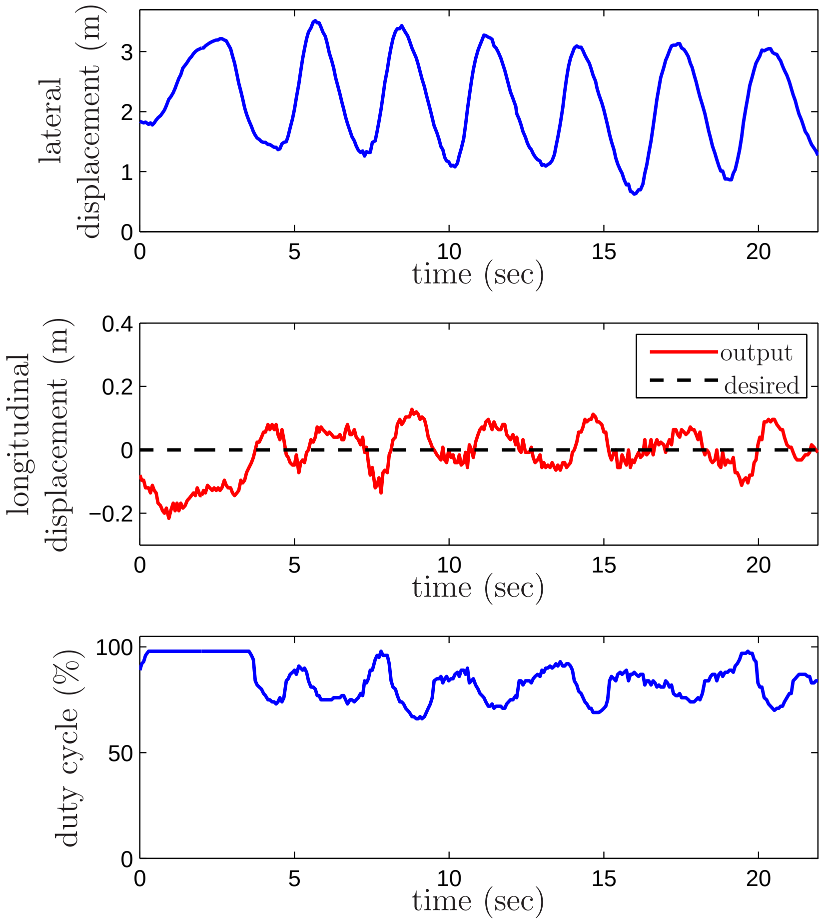


Figure 4.10: Closed-loop step response of height regulation with an added mass. (top) lateral displacement, (middle) longitudinal displacement and desired altitude, and (bottom) duty cycle for the driving DC motor.

Chapter 5

Flight Control for Target Seeking

In this chapter, we present closed-loop onboard attitude regulation for the ornithopter, equipped with inertial sensors, visual sensors, and a microprocessor for onboard processing of sensor signals as well as control input signals. We also discuss dead reckoning to the target location when the target is temporarily not exposed on the visual sensor. Based on the development of onboard attitude regulation and the dead reckoning algorithm, we demonstrate autonomous flight of the ornithopter flying toward a target without any external resources.

5.1 Attitude Estimation

The coordinate systems[1] and the forces generated by the wings and the propeller are depicted in Fig. 5.1. The body coordinates ($\mathbf{e}_x$, $\mathbf{e}_y$, $\mathbf{e}_z$), attached to the center of mass, are chosen to be the principal axes of inertia of the ornithopter. $\mathbf{T}$ represents the propulsive force on the average center of pressure generated by the flapping motion of the wings. The center of pressure is denote by c.p. in Fig. 5.1. The force, $\mathbf{R}$, produced by the propeller generates the moment about the $\mathbf{e}_z$ axis.

The kinematic equations of angular motion of the ornithopter can be written in 3-2-1 (or Z-Y-X) Euler angles [33],

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \\ q\cos\phi - r\sin\phi \\ q\sin\phi\sec\theta + r\cos\phi\sec\theta \end{pmatrix}, \tag{5.1}$$

where p, q, and r are angular rates in the $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$ axis, respectively, and ϕ, θ, and ψ are the Euler angle representations for roll, pitch, and yaw, respectively.

---

[1] The coordinate systems used in this section are 3-dimensional unlike the coordinate systems in Section 4.1. Note that the axes defined here are not the same as those in Section 4.1.

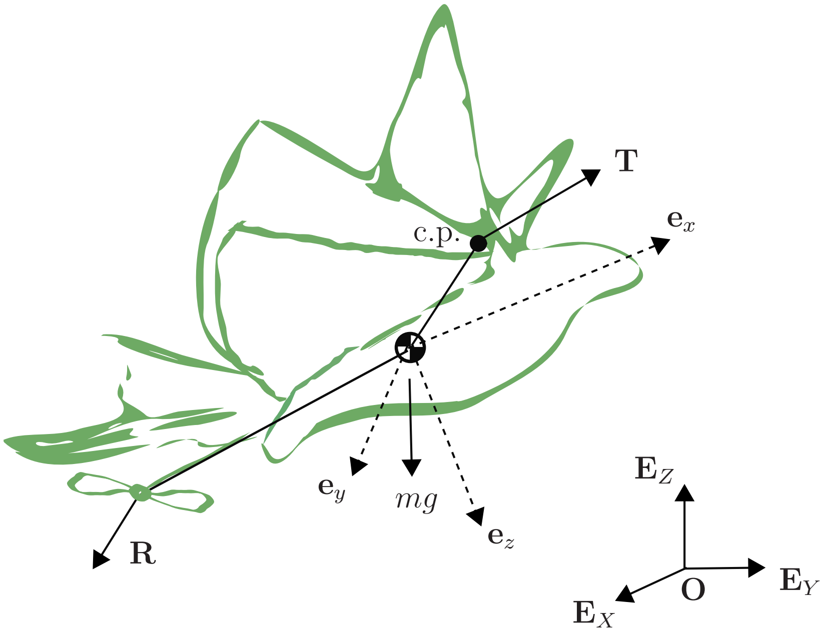


Figure 5.1: The coordinate systems and the forces generated by the wings and the propeller.

To estimate the orientation of the ornithopter, we calibrate the gyroscope before each flight in order to remove constant biased measurement errors. During flight we obtain the angular velocities of the ornithopter in the body axes (p, q, r) using the 3-axis gyroscope and compute ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) for every 2.5 milliseconds using (5.1). Then, we integrate ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) to estimate the real time orientation of the ornithopter in Euler angles, (ϕ, θ, ψ).

The integration-based attitude estimation used in this work requires much less computational power than other estimators based on Bayesian statistics such as [5, 15, 22, 42]. However, it has a serious drawback; integration drift is gradually accumulated and generates a large amount of error as time elapses. The integration drift for the gyroscopes used in this work is approximately 0.1 degrees per second. Although the ground truth orientation estimates are erroneous over time, the deviation angles between the ornithopter's orientation angles and the target location is not affected by the drift when we determine the desired orientation angles, which will be discussed in following section.

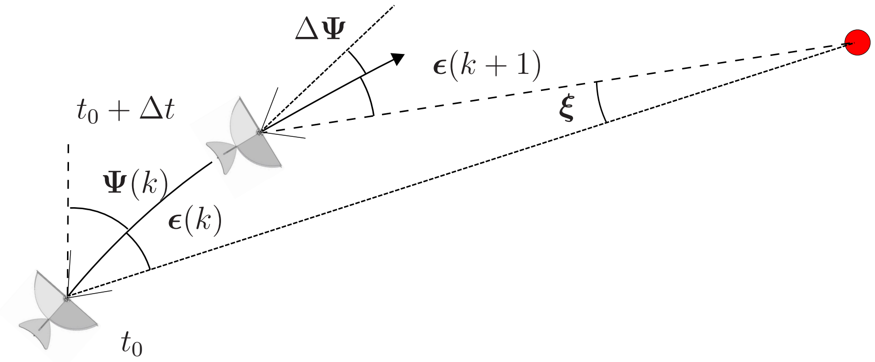


Figure 5.2: Prediction of target location at time $k+1$ based on the current orientation and its rate at time k.

5.2 Prediction of Target Location

5.2.1 Target Tracking Algorithm

The desired yaw angle, $r_\psi(k)$, and desired pitch angle, $r_\theta(k)$, at time k, are computed by

$$\begin{aligned} r_\psi(k) &= \psi(k)+\epsilon_\psi(k), && (5.2)\\ r_\theta(k) &= \theta(k)+\epsilon_\theta(k), && (5.3) \end{aligned}$$

where ϵ_ψ and ϵ_θ are, respectively, the yaw deviation angle and the pitch deviation angle between the ornithopter's heading and the target. We can rewrite (5.2) and (5.3) in vector form,

$$\mathbf{r}(k)=\mathbf{\Psi}(k)+\boldsymbol{\epsilon}(k), \qquad (5.4)$$

where $\mathbf{r}=(r_\psi,r_\theta)^\mathsf{T}$, $\mathbf{\Psi}=(\psi,\theta)^\mathsf{T}$, and $\boldsymbol{\epsilon}=(\epsilon_\psi,\epsilon_\theta)^\mathsf{T}$.

The deviation angle, $\boldsymbol{\epsilon}$, is computed by

$$\boldsymbol{\epsilon}(k)=\mathbf{\Omega}(k)\mathbf{p}(k),$$

where $\mathbf{p}$ is a vector of pixel values of the target location directly obtained from the infrared camera, defined by $(p_x,p_y)^T$, and $\mathbf{\Omega}$ is the combination of scaling matrix and rotational transformation defined by

$$\mathbf{\Omega}(k)=\begin{pmatrix} s_x & 0\\ 0 & s_y \end{pmatrix}\begin{pmatrix} \cos(\phi(k)) & \sin(\phi(k))\\ -\sin(\phi(k)) & \cos(\phi(k)) \end{pmatrix}.$$

Here, $s_x=20°/1024$ and $s_y=15°/768$ are the scale factors to convert pixel locations to angles, and the roll angle, ϕ, is the rotation of the infrared camera about the $\mathbf{e}_x$ axis.

As the ornithopter travels, the orientation angle, $\mathbf{\Psi}$, as well as the deviation angle, $\boldsymbol{\epsilon}$, change in time as shown in Fig. 5.2. Therefore, the attitude regulation using the desired orientation angle computed by (5.4) can be improved with the prediction of the orientation angle and the deviation angle at the subsequent state. This prediction can also be used in dead reckoning of the target location when the target is temporarily lost.

At $t = t_0$, the deviation angle is $\boldsymbol{\epsilon}(k)$. For Δt, the orientation of the ornithopter is changed by $\Delta\mathbf{\Psi}(k+1) \triangleq \mathbf{\Psi}(k+1)$ - $\mathbf{\Psi}(k)$, and the deviation angle at $k+1$ becomes $\boldsymbol{\epsilon}(k+1)$. From Fig. 5.2, we can find

$$\Delta\Psi(k+1) + \boldsymbol{\epsilon}(k+1) \approx \boldsymbol{\epsilon}(k) + \boldsymbol{\xi}, \tag{5.5}$$

where $\boldsymbol{\xi}$ is the angle between the position of the ornithopter at t_0 and the position at $t_0 + \Delta t$ with respect to the target. Since the distance between the ornithopter and the target is unknown, $\boldsymbol{\xi}$ is also unknown. Then, using (5.4) and (5.5), the prediction of the desired orientation can be written by

$$\begin{aligned} \mathbf{r}(k+1|k) &= \mathbf{\Psi}(k) + \boldsymbol{\epsilon}(k+1) \\ &\approx \mathbf{\Psi}(k) + \boldsymbol{\epsilon}(k) + \boldsymbol{\xi}(k) - \Delta\mathbf{\Psi}(k+1) \\ &\approx \mathbf{r}(k) + \boldsymbol{\xi}(k) - \Delta\mathbf{\Psi}(k). \end{aligned}$$

Now, we can formulate a Kalman filter to predict the desired orientation angle as follows

$$\begin{aligned} \mathbf{r}(k+1|k) &= \mathbf{r}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{w}(k), \\ \boldsymbol{\epsilon}(k) &= \mathbf{r}(k) + \mathbf{D}\mathbf{u}(k) + \mathbf{v}(k). \end{aligned}$$

Here, $\mathbf{u}(k) = (\mathbf{\Psi}(k), \mathbf{\Psi}(k-1))^T$ is the input vector to the Kalman filter, the process noise $\mathbf{w}(k)$ and the measurement noise $\mathbf{v}(k)$ are independent zero mean Gaussian noises, $\mathbf{B}$ is the process weighting factor to compensate the unknown value of $\boldsymbol{\xi}$[2] and defined by

$$\mathbf{B} = \begin{pmatrix} -\alpha_\psi & 0 & \alpha_\psi & 0 \\ 0 & -\alpha_\theta & 0 & \alpha_\theta \end{pmatrix},$$

where α_ψ is the process weighting factor for the yaw angle and α_θ is the process weighting factor for the pitch angle, and D is defined by

$$\mathbf{D} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}.$$

The value of α is determined by the state of target tracking. We have experimentally found that $(\alpha_\psi, \alpha_\theta) = (1.2, 1.2)$ is sufficient if the target is exposed on the infrared camera. The target, however, is occasionally not exposed on the camera after large external disturbances or large overshoots from the desired trajectory.[3] For this case, we need to have a dead reckoning algorithm to estimate the target location without any sensor input.

---

[2] If the target is located infinitely far away from the ornithopter, the practical value of ξ would be zero. In this case, the process weighting factor, α, will be 1.

[3] The infrared camera used in this work has a narrow field of view, which is ±20° horizontally and ±15° vertically.

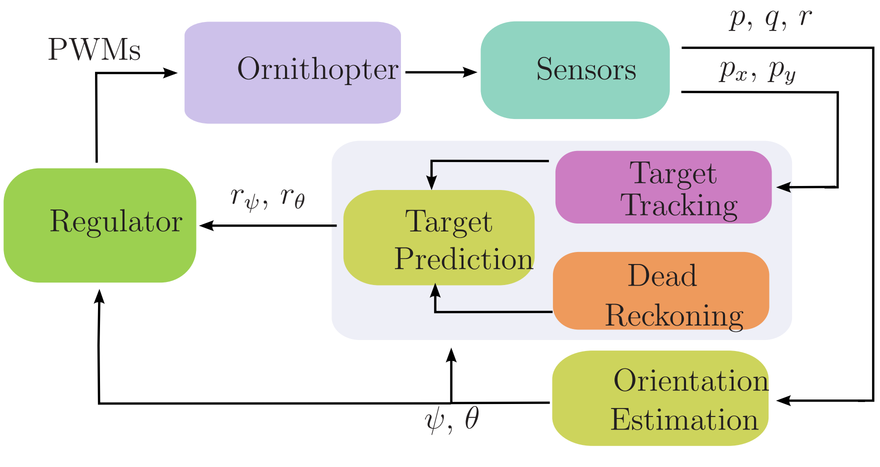


Figure 5.3: Signal flow for closed-loop onboard controller

5.2.2 Dead Reckoning to Target Location

When the target is not exposed on the camera, the innovation term in the Kalman filter is ignored since the measurement from the visual sensor is meaningless. Then, an algorithm for dead reckoning takes the place of target tracking as depicted in Fig. 5.3 to predict the target location.

Figure 5.4 illustrates an example of the ornithopter's trajectory when the ornithopter momentarily loses the target. At $t = t_1$, the target is lost, and for $t_1 \leq t \leq t_2$, the ornithopter flies away from the target, i.e., $\text{sgn}(\epsilon_\psi) \neq \text{sgn}(\Delta\psi)$. For $t_2 \leq t \leq t_3$, the target is still lost, but the ornithopter flies toward the target, i.e., $\text{sgn}(\epsilon_\psi) = \text{sgn}(\Delta\psi)$. The target is rediscovered at $t = t_3$.

The dead reckoning algorithm keeps the desired pitch angle unchanged while the target is lost, i.e., $r_\theta(k+1) = r_\theta(k)$. For the desired yaw angle, we have experimentally found that $\alpha_\psi = 2$ is sufficient most of time when the ornithopter is flying away from the target. When the ornithopter is turning toward the target while the target is still lost, we keep the desired yaw angle unchanged, i.e., $r_\psi(k+1) = r_\psi(k)$. The dead reckoning discussed so far is summarized as follows[4]:

$$\begin{aligned} r_\theta(k+1) &= r_\theta(k), \\ r_\psi(k+1) &= \begin{cases} r_\psi(k) & \text{if } \text{sgn}(\epsilon_\psi) = \text{sgn}(\Delta\psi), \\ r_\psi(k) - 2\Delta\Psi & \text{otherwise.} \end{cases} \end{aligned}$$

---

[4] This algorithm would not work if the target is unobservable for a long period of time, possibly due to large disturbance

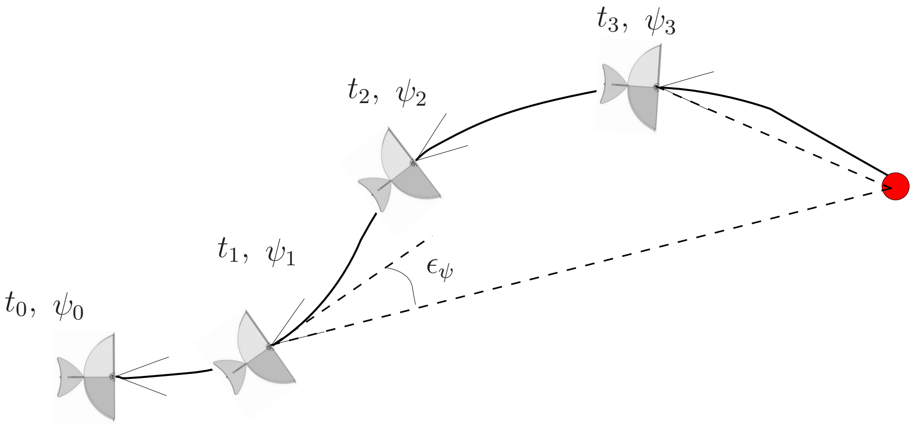


Figure 5.4: Illustration of dead reckoning of target location. The two lines at the head of the ornithopter represent the field of view of the infrared camera, and the dot represents the location of the target.

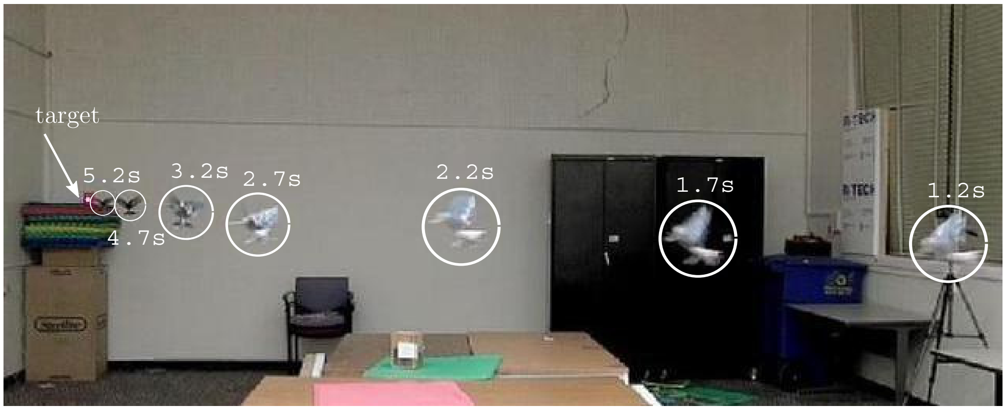
5.2.3 Control Method

Autonomous flight control of the ornithopter is implemented with the onboard sensors and microprocessor as described in Section 3.1. We have used the PID controller discussed in Section 3.3. There are two PID controllers running simultaneously at 400 Hz for yaw and pitch regulation. The desired inputs to the PID controllers (denoted as $r(k)$ in Fig. 3.3) for this chapter are the desired orientation angles, r_ψ and r_θ, and the output signals from the plant (denoted as $y(k)$ in Fig. 3.3) are the estimates of the orientation angles, ψ and θ. For the PID controllers, we assume that the propeller mainly contributes to the yaw moment and the wings mainly contribute to the pitch moment and the thrust.

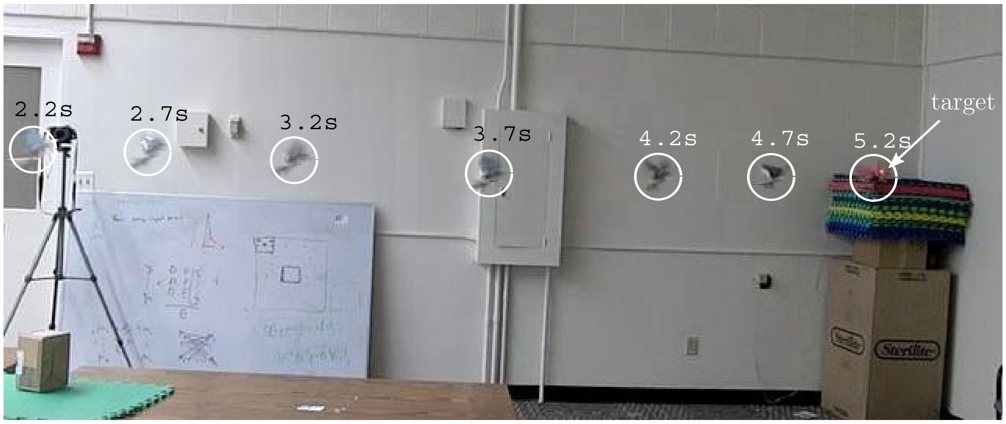
5.3 Experimental Results and Discussion

We conducted a number of experiments to verify the control method described in Section 5.2.3. First, we placed an infrared emitting target approximately 10 meters away from the initial location of the ornithopter. To test loss of the target, the ornithopter was released with an initial nonzero roll angle. Subsequently, the ornithopter could turn away from the target at the beginning of its flights and lost the target in one or two seconds. To inspect the trajectories of the ornithopter flights, we took movies of the ornithopter while it was flying toward the target. The mosaics of a flight toward the target in the longitudinal view and the lateral view are shown in Fig. 5.5.

The orientation angles (ψ and θ), the deviation angles (ϵ_ψ and ϵ_θ), the desired angles (r_ψ and r_θ), and the trajectory of the flight in Fig. 5.5 are plotted in Fig. 5.6. The ornithopter flew away from the desired yaw angle at the beginning of its flight as shown in Fig. 5.6a

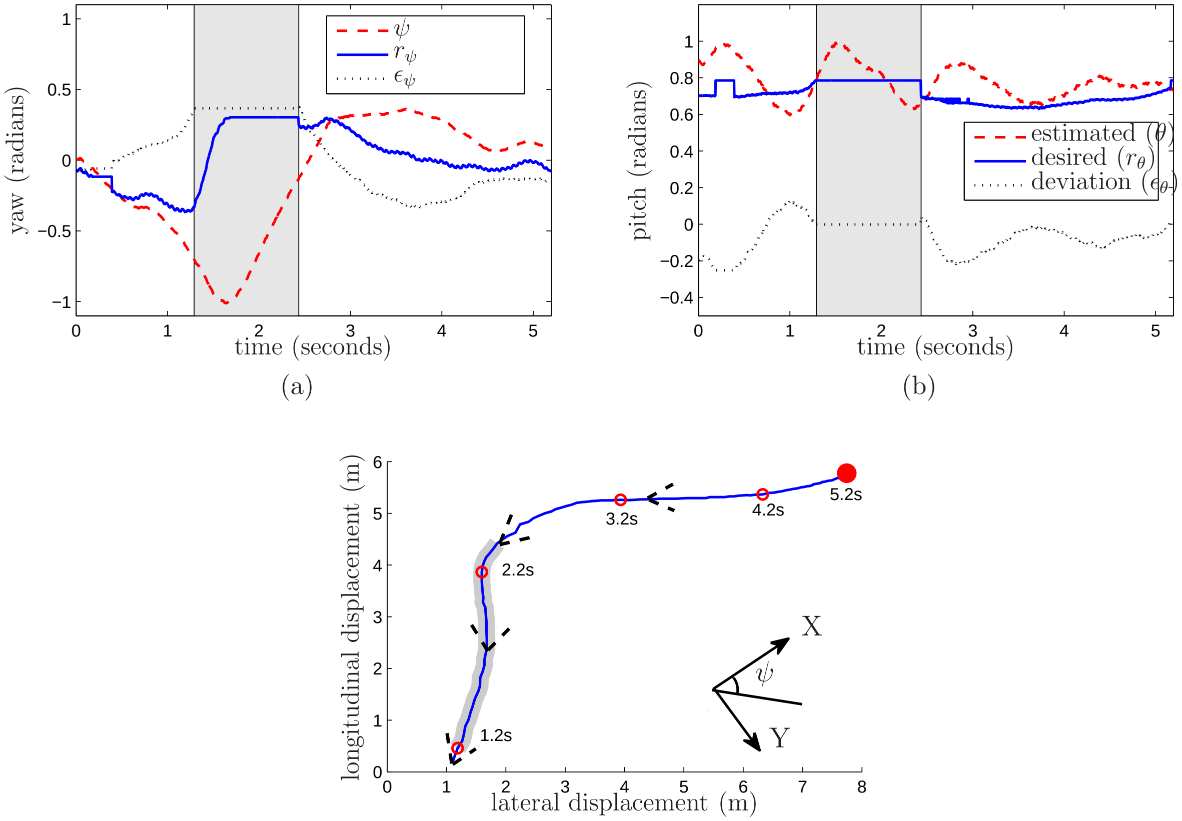


(a)



(b)

Figure 5.5: Mosaics of the ornithopter flying toward the target. The numbers denote the time stamp. (a) longitudinal view and (b) lateral view.



(a) (b)

(c)

Figure 5.6: An experimental result of flight control of target seeking. The regions covered by gray shadows indicate that the target was lost. (a) yaw angle regulation (b) pitch angle regulation (c) the trajectory of the ornithopter flying toward the target. The dashed lines in (c) represents the field of view of the infrared camera, the numbers next to circles denote the time stamp, and the dot on the upper right corner represents the target.

because of the initial nonzero roll angle. Subsequently, it lost the target as the target was no longer exposed on the camera. The loss of target is represented by regions covered by gray shadows in Fig. 5.6. The constant values of the deviation angles in the gray area shown in Fig. 5.6a and Fig. 5.6b indicate that the deviation angles were not updated due to the loss of the target. As the ornithopter flew away from the target while the target was lost, the desired angle rapidly increased for 1.2 s $\leq t \leq$ 1.7 s because we set $\alpha = 2$. When t was approximately 1.7 s, the ornithopter started turning back toward the target and the desired yaw angle stayed constant for $r_\psi(k+1) = r_\psi(k)$. As the yaw angle approached close to the desired yaw angle, the target was discovered by the camera at $t \approx$ 2.4 s. After a small overshoot, the yaw angle stayed close to the desired value as shown in Fig. 5.6a.

The desired pitch angle remained unchanged due to $r_\theta(k+1) = r_\theta(k)$ while the target was lost. After the target was discovered at $t \approx$ 2.4 s, the desired pitch angle was updated with the deviation angle. Although the pitch angle was oscillating with approximately 0.7 Hz because of the inherent dynamic property of the ornithopter, it stayed close to the desired pitch angle as time elapsed as shown in Fig. 5.6b.

From the movies, we were able to extract the last 4 seconds of the path data of the ornithopter as shown in Fig. 5.6c.[5] The dot on the upper right corner in Fig. 5.6c represents the target location. The dashed lines represent the field of view of the infrared camera. It is evident from the trajectory that the target was not exposed on the camera momentarily, but the ornithopter still managed to turn toward the target and reached it at last.

We performed more experiments to investigate the accuracy of the flight control for target seeking as shown in Fig. 5.7. The set of ornithopter trajectories show that the ornithopter successfully rediscovered the target after the target was lost, and the ornithopter finally reached the target for most of the trials. With 20 trials with different initial roll angles, the ornithopter reached 17 times within a radius of 0.5 m from the target.

---

[5] Due to the narrow field of view of the external cameras, we were not able to take movies for the beginning of the flight.

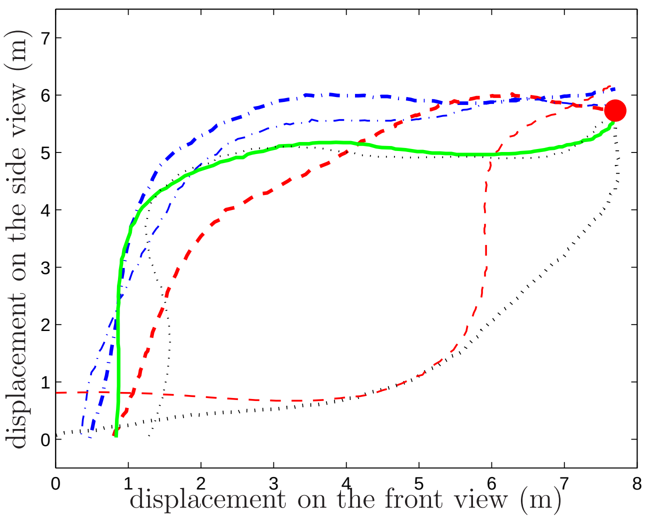


Figure 5.7: A set of trajectories of ornithopter flights toward the target. The dot on the upper right corner represents the target.

Part III

Power Efficient Design

Chapter 6

Efficient Resonant Drive of Flapping-Wing Robots

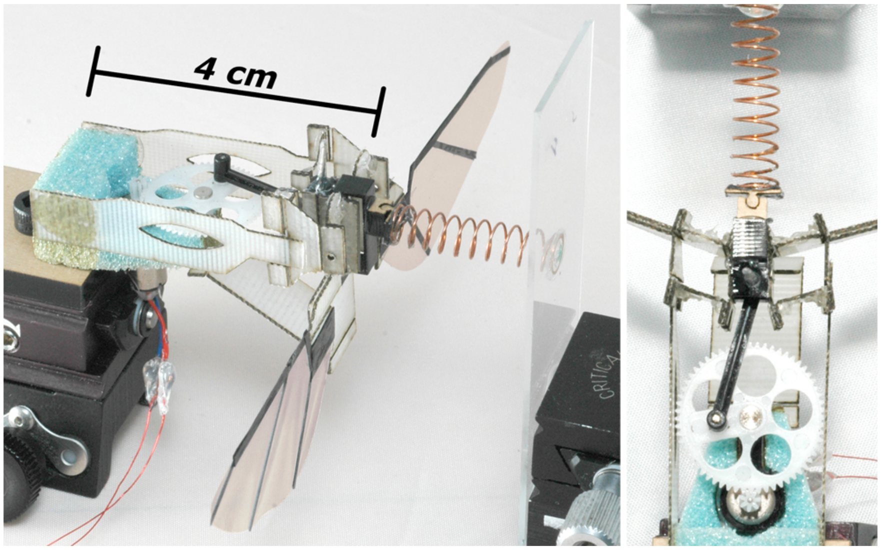


Figure 6.1: A 5.8 gram hummingbird-inspired flapping mechanism developed at UC Berkeley [3].

In this chapter, we will present a nondimensional analysis of a nonlinear motor-driven crank-arm mechanism coupled to a mass-spring system as a solution to general motor driven oscillators, such as used in flapping-wing robots. Using this analysis, we will demonstrate a link between the resonant frequency of a compliant mechanism and the ideal motor input voltage to achieve maximum power efficiency. We will also demonstrate the effect of the geometric nonlinearity associated with the crank arm and the effect of damping, both motor and aerodynamic. Finally, we will apply this analysis to a 5.8g flapping mechanism shown in Fig. 6.1 and empirically identify the improvement to the system's power plant efficiency.

6.1 Analysis

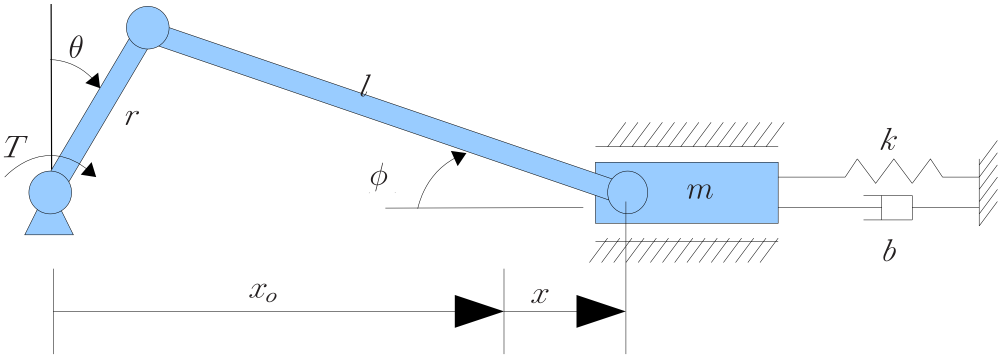


Figure 6.2: A load connected to a spring and a damper is driven by a DC motor through a crank arm.

Compliant mechanisms driven by a DC motor such as the flapping robot pictured in Fig. 6.1 can be modelled as depicted in Fig. 6.2. The crank arm driven by a motor/gear is connected to a load (slider) with a spring and a damper. The equation of motion is expressed as:

$$\eta^2 J_m\ddot{\theta} + \eta^2 b_m\dot{\theta} + \frac{r\cos(\phi-\theta)}{\cos(\phi)}(m\ddot{x} + b|\dot{x}|\dot{x} + kx) = T, \tag{6.1}$$

where

m = the mass of the load,
k = the spring constant,
b = the aerodynamic damping coefficient,
ϕ = the angular position of the connecting rod,
θ = the angular position of the crank arm,
l = the lengths of the connecting rod,
r = the lengths of the crank arm,
J_m = the moment of inertia of the motor,
b_m = the damping coefficient of the motor,
η = the gear ratio,
T = the torque applied to the crank arm.

In (6.1), the $b|\dot{x}|\dot{x}$ term is the nonlinear aerodynamic damping force.

The displacement of the slider, x, can be expressed as

$$x = r \sin\theta$$

Substituting x and its derivatives, $\dot{x}$ and $\ddot{x}$, into (6.1), we obtain

$$\ddot{\theta}(\eta^2 J_m + \beta m r^2 \cos\theta) + \dot{\theta}\eta^2 b_m + \beta k r^2 \sin\theta + \dot{\theta}^2(\beta b r^3 \operatorname{sgn}(\dot{\theta}\cos\theta)\cos^2\theta - \beta m r^2 \sin\theta) = T \quad (6.2)$$

Here, β is defined as

$$\beta = \frac{\cos(\phi - \theta)}{\cos\phi} = \cos\theta + \frac{\lambda\cos\theta\sin\theta}{\sqrt{1 - \lambda^2\cos^2\theta}},$$

where $\lambda \triangleq r/l$ is the ratio of the crank arm length to the connecting rod length. For the second equality, the law of sines is applied to remove ϕ;

$$r\cos\theta = l\sin\phi. \quad (6.3)$$

The $\operatorname{sgn}(\dot{\theta}\cos\theta)$ term in (6.2) can also be simplified to $\operatorname{sgn}(\cos\theta)$ for the direction of motor revolution does not change.

The defining equations of the DC motor are

$$T_m = K_t i, \quad (6.4)$$

$$L\frac{di}{dt} + iR_m + K_e\omega = V_s, \quad (6.5)$$

where V_s is the input voltage, T_m is the torque generated by the DC motor and $T = \eta T_m$, L is the inductance of the motor, i is the induced current, R_m is the motor and battery resistance, K_e is the back EMF coefficient, and K_t is the torque coefficient of the motor. Since L is generally very small, we can safely set $L = 0$ for the rest of the analysis in this chapter. Substituting (6.4) and (6.5) into (6.2), we can obtain the equation of the overall dynamic system:

$$\ddot{\theta}(\eta^2 J_m + \beta m r^2 \cos\theta) + \dot{\theta}\eta^2\left(b_m + \frac{K_t K_e}{R_m}\right) + \beta k r^2 \sin\theta$$
$$+ \dot{\theta}^2(\beta b r^3 \operatorname{sgn}(\cos\theta)\cos^2\theta - \beta m r^2 \sin\theta) = \frac{\eta K_t}{R_m} V_s \quad (6.6)$$

To obtain a nondimensional equation of motion, define the following variables:

$$\begin{aligned}
\omega_n &\triangleq \sqrt{\frac{k}{m}} \text{ : the undamped natural frequency of the slider,}\\
\tau &\triangleq \omega_n t \text{ : the nondimensional time,}\\
\Omega &\triangleq \frac{d\theta}{d\tau} = \frac{\dot{\theta}}{\omega_n} \text{ : the nondimensional angular velocity of the motor,}\\
\dot{\Omega} &\triangleq \frac{d^2\theta}{d\tau^2} = \frac{\ddot{\theta}}{\omega_n^2} \text{ : the nondimensional angular acceleration of the motor,}\\
J &\triangleq \eta^2 \frac{J_m}{mr^2} \text{ : the nondimensional moment of inertia of the motor,}\\
B_m &\triangleq b_m + \frac{K_t K_e}{R_m} \text{ : the effective motor damping,}\\
\zeta &\triangleq \frac{br}{2m} \text{ : the damping ratio of the slider,}\\
\zeta_m &\triangleq \eta^2 \frac{B_m}{2mr^2\omega_n} \text{ : damping ratio of the motor,}\\
\tilde{T} &\triangleq \frac{T}{mr^2\omega_n^2} \text{ : the nondimensional motor torque.}
\end{aligned}$$

We can now obtain a nondimensional equation of motion for (6.6) as follows:

$$(J + \beta\cos\theta)\dot{\Omega} + 2\zeta_m\Omega + \beta\sin\theta + (2\zeta\beta\,\mathrm{sgn}(\cos\theta)\cos^2\theta - \beta\sin\theta)\Omega^2 = u, \tag{6.7}$$

where u is the nondimensional input voltage defined as

$$u = \frac{\eta K_t}{mr^2 R_m \omega_n^2} V_s.$$

6.2 Case Studies

6.2.1 Case I: Constant Speed

It is rather easy to understand the behavior of the resonant drive with a constant speed excitation due to the similarity to linear mass-spring systems. For the case of constant motor speed, we can set $\dot{\Omega} = 0$ in (6.7) and we have

$$2\zeta_m\Omega + \beta\sin\theta + (2\zeta\beta\,\mathrm{sgn}(\cos\theta)\cos^2\theta - \beta\sin\theta)\Omega^2 = u.$$

The variation of torque required to drive the system in one cycle is shown in Fig. 6.3. The peak value of the output torque becomes minimal close to $\Omega = 1.0$, where the system is excited

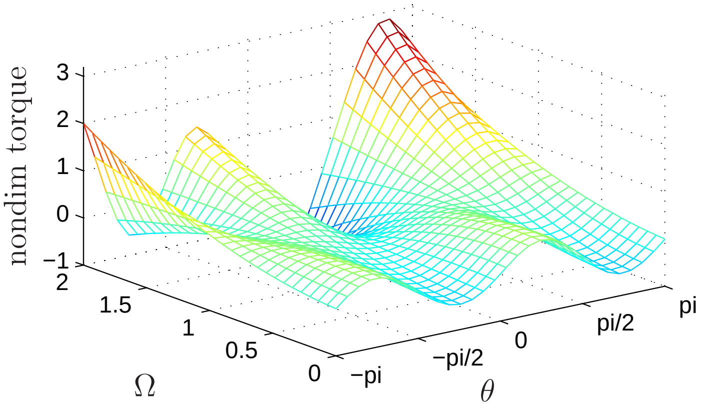
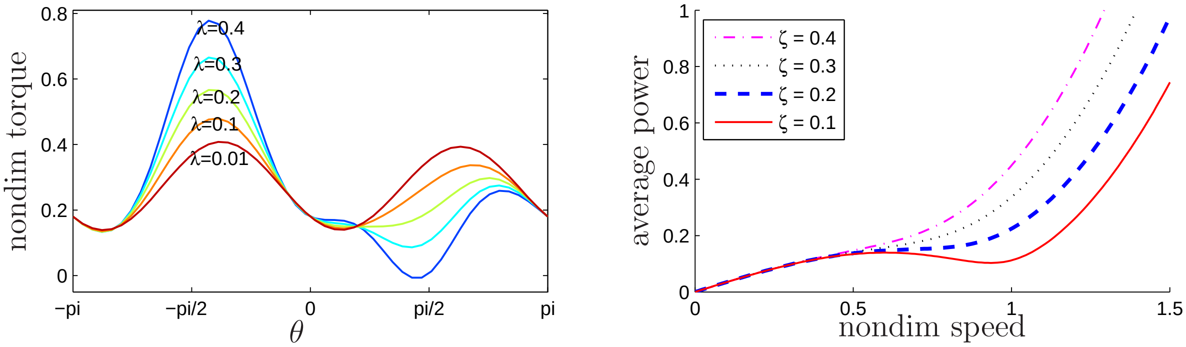


Figure 6.3: Variation of nondimensional torque in one cycle for various constant angular speed with $\lambda = 0.1$, $\zeta_m = 0.1$, and $\zeta = 0.1$.

at its resonant frequency. This plot agrees with the results of the minimal torque approach studied by Khatait *et al.* [21]. In their study, they have demonstrated that there is a certain value of torsional stiffness of flexural joints corresponding to the driving frequency that minimizes the peak input torque. Indeed, the corresponding driving frequency in their study is fundamentally identical to the resonant frequency of the compliant system discussed here. Additionally, the choice of spring constant in the study of Tantanawat and Kota [41] indeed provides their compliant system with the resonant frequency matching with the desired flapping frequency so that the peak input power is minimized.

It should also be noted that for the values of Ω away from 1, negative torques are required to drive the system at a constant speed, i.e., excessive kinetic energy of the inertial load is transmitted back to the motor instead of converted to strain energy in the spring. Tantanawat and Kota [41] have assumed in their study that this negative input power, which is the energy absorbed by the motor from the load, is fully recovered. However, this is not true when taking into account motor and battery resistance - power is still dissipated with negative current in the motor and battery resistances. From a controls standpoint, absorbing the energy for later use or supplying negative current to decelerate the plant is undesirable.

In fact, the variation of input torque can also be reduced tremendously by reducing the ratio of the crank arm length to the connecting rod length, λ. As shown in Fig. 6.4a, the peak input torque becomes smaller as λ decreases. The torque variation also becomes more symmetrical, and negative input torque is no longer required. Indeed, with $\lambda \sim 0$, ϕ becomes 0 and the force applied to the mass through the connecting rod becomes a pure sinusoidal input to the mass-spring system. However, an excessively small value of λ (very long connecting rod) results in undesirable bending modes of the connecting rod. $\lambda \geq 0.2$ is also undesirable because it requires not only high peak torques but also negative torques to



(a) (b)

Figure 6.4: Nondimensional torque and average power for constant motor speed. (a) Variation of nondimensional torque in one cycle for various values of λ with $\Omega = 0.9$, $\zeta_m = 0.1$, and $\zeta = 0.1$. (b) Nondimensional average power required to drive the system at a constant angular speed for various damping ratios with $\lambda = 0.1$, $\zeta_m = 0$

keep the speed constant.

Nondimensionalized average power required to drive the system in one cycle with respect to damping ratio is shown in Fig 6.4b. The average power is reduced at $\Omega{\sim}1$ when $\zeta \leq 0.2$, i.e., at $\Omega{\sim}1$ we can achieve a local minimum of the power to drive the crank.[1] As the damping ratio increases, an addition of a spring to the system makes a negative contribution because the input power supplies not only the energy dissipated in the damper but also the energy to be stored in the spring.

6.2.2 Case II: Constant Input Voltage

When a constant voltage is applied to the DC motor, the resonant drive behavior becomes more complicated since the frequency of the applied force (or the motor speed) to a load no longer remains constant. The motor speed and torque, however, stay almost constant if the system is driven at its resonant frequency and the damping ratios are small. Shown in Fig. 6.5a and Fig. 6.5b are, respectively, the motor speed and the torque for small damping ratios of $\zeta_m = 0.002$ and $\zeta = 0.01$ ($\lambda = 0.2535$). When the nondimensional input voltage u is 0.25, the motor speed and torque indeed remain almost constant. Furthermore, the motor speed is almost one, i.e., the system is driven at its resonant frequency.

The peak input power, calculated by $\Omega \cdot \tilde{T}$, becomes minimal when the nondimensional input voltage u is 0.25 as shown in Fig. 6.5c. It should be noted that the induced motor current is directly proportional to the motor torque as presented in (6.4). Hence, the electrical

---

[1] The global minimum occurs at Ω=0, but $\Omega{\sim}0$ is not an interesting region.

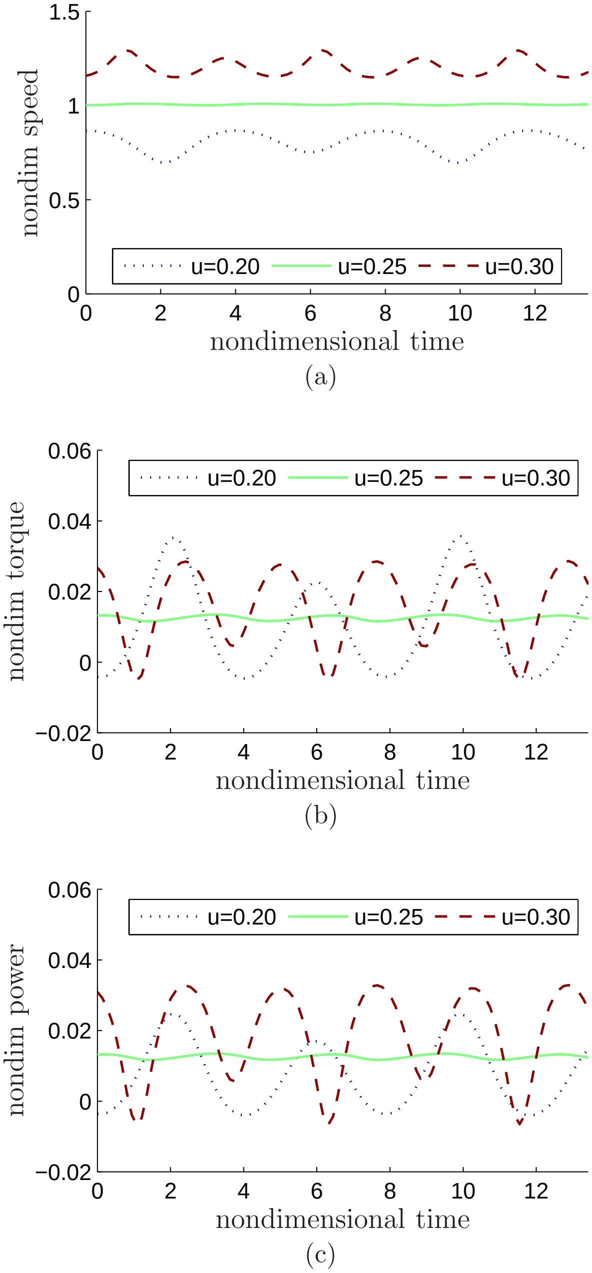


Figure 6.5: Effect of input voltages with $\zeta_m = 0.002$, $\zeta = 0.01$, and $\lambda = 0.2535$. (a) the variation of nondimensional speed. (b) the variation of nondimensional torque. (c) the variation of nondimensional power.

power for the motor as well as the mechanical power from the motor have the minimal peak-to-peak values. As the input voltage increases or decreases, the torque becomes more oscillatory and eventually becomes negative, i.e., some of the kinetic energy of the load is transmitted back to the battery instead of stored in the spring.

The effect of the aerodynamic damping ratio, ζ, is shown in Fig. 6.6. As the damping ratio increases, the required torque to drive the load clearly increases while the motor speed decreases. Both motor speed (Fig. 6.6a) and torque (Fig. 6.6b) become more oscillatory, which can be expected from the $\zeta\beta\cos^2\theta$ term in (6.7). The effect of ζ_m is, however, independent of the oscillating behaviors as expected from the $2\zeta_m\Omega$ term, and the DC value of Ω monotonically decreases as ζ_m increases. The effect of the parameter ζ_m is basically the same as a damping ratio of a linear mass-spring-damper system. Figure. 6.6c shows the nondimensional power increase as the damping ratio increases, which is not surprising. As a side note, this model also provides an estimate of motor velocity for a given battery voltage, for which no closed-form solution is available.

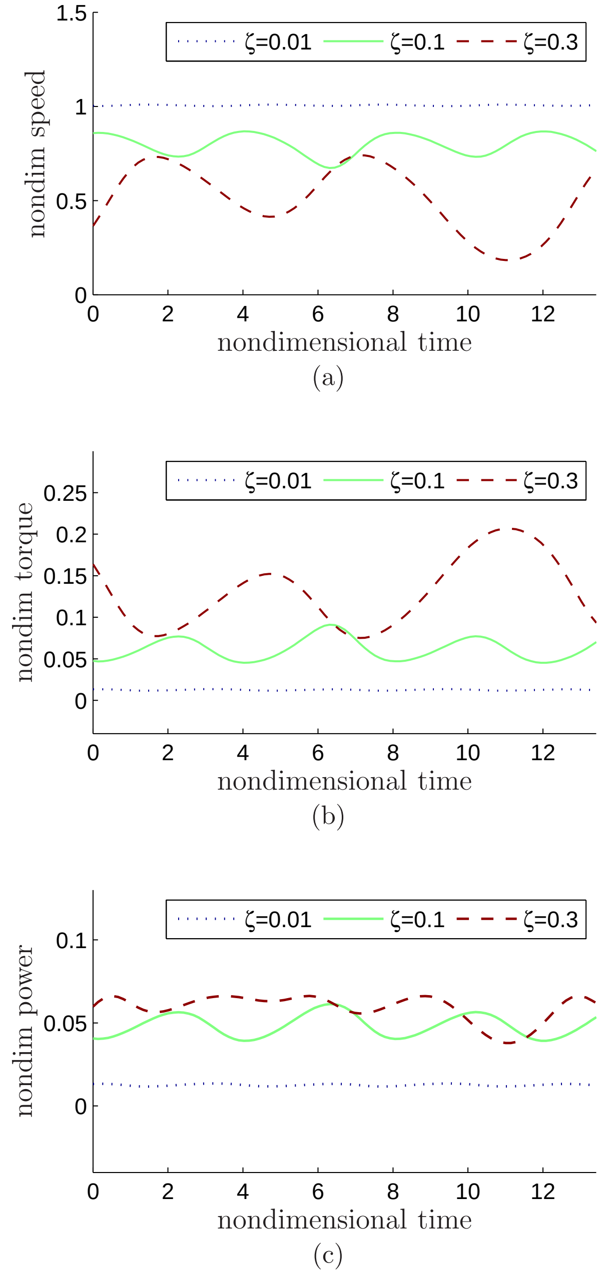
The required average input power with respect to the mean of nondimensional motor speed, $\mathrm{E}[\Omega]$, is shown in Fig. 6.7.[2] While the average power with no spring monotonically increases with the speed, the average power with a spring has a minimum. The best improvement of efficiency occurs when $\mathrm{E}[\Omega]$ is approximately 1. It is well known that the magnitude response for a linear mass-spring system becomes less than unity with an out-of-phase drive and therefore no power benefits can be achieved when compared to a spring-less system. However, power reduction can be achieved for the motor driven compliant system studied here even though the operating speed is significantly higher than the undamped natural frequency of the load. The significant prediction is that low damped mechanical systems with a spring driven by a DC motor can be excited at a higher speed than the system without a spring for the same amount of input power. Hence, greater power can be obtained or a smaller motor can be used.

6.3 Experiments

The flapping mechanism shown in Fig. 6.1 consists of two mirrored levers actuated by a single DC motor [3]. The fulcrum of each lever is cantilevered by a beam and hinge from a base structure that also houses the DC motor coupled to a gearbox and crank. Due to the kinematic constraints set by the equal lengths of both fulcrum beams, the system effectively becomes a slider-crank mechanism. Motion normal to the lever plane is restricted by the geometry of the fulcrum beams, and an additional Sarrus linkage, positioned between the driving block and the base structure and operating out of plane, effectively constrains the motion of the driving block along the intended linear path. The specifications of the flapping mechanism are given in Table 6.1.

---

[2] Data is not available for $\mathrm{E}[\Omega] < 0.5$ because small values of constant input voltage cannot overcome the initial spring force to complete a revolution.



(a)

(b)

(c)

Figure 6.6: Effect of damping ratio with $\zeta_m = 0.002$, $u = 0.25$, and $\lambda = 0.2535$. (a) the variation of nondimensional speed. (b) the variation of nondimensional torque. (c) the variation of nondimensional power.

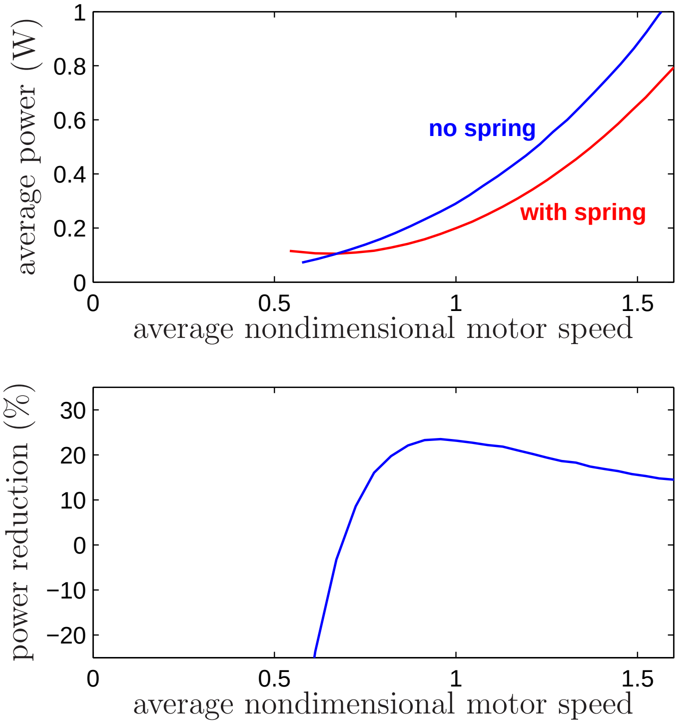


Figure 6.7: average power (top) and power reduction using a spring (bottom) with respect to the mean of nondimensional motor speed E[Ω]. ($\zeta_m = 0.002$ and $\zeta = 0.05$)

Table 6.1: Specifications of the flapping mechanism

motor inertia	1.41gm-mm^2
motor damping	8.26gm-mm^2/s
gear ratio	1:8.57
wing inertia	142.9gm-mm^2
air damping	0.05-0.11kg/m for $\dot{\theta}$=120-160s^{-1}
spring constant	140N/m
crank arm	4 mm
connecting rod	17 mm
total mass	5.8 gm

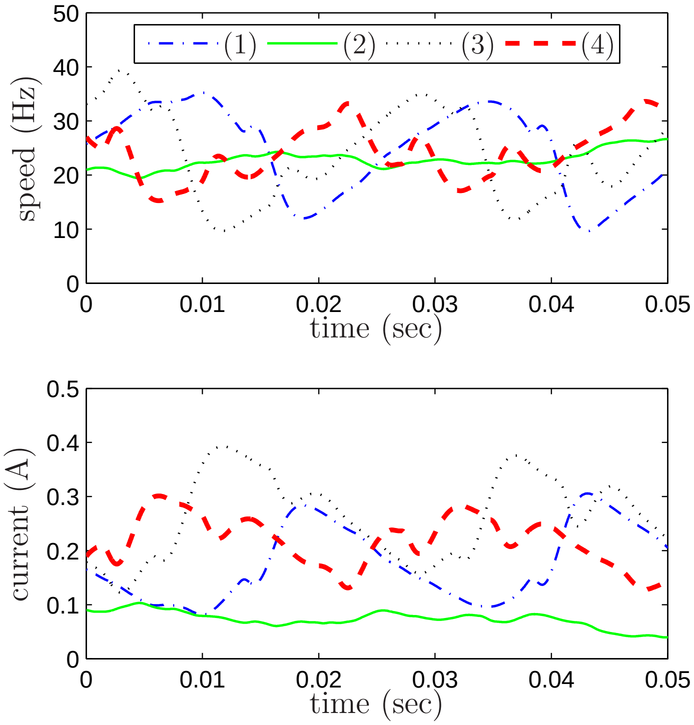


Figure 6.8: Measurement of motor speed (top) and induced current (bottom) with constant input voltages. (1) no spring in vacuum at V = 1.45V (2) with spring in vacuum at V = 1.05V (3) no spring in air at V = 1.70V (4) with spring in air at V = 1.55V.

The motor velocity and the measured current of the flapping mechanism with and without a linear coiled spring were measured as shown in Fig. 6.8. The additional 140N/m spring was added between the slider crank and the ground. The undamped natural frequency of the load was calculated as $\omega_n = 155$rad/s (24.5Hz) with the effective wing mass of 5.83×10^{-3}kg.[3] Measurements were also taken in vacuum in order to verify the analysis for the crank-arm model with extremely low aerodynamic damping. With the measurements in vacuum, the aerodynamic damping versus internal damping of the mechanism could be examined. The motor damping ratio and aerodynamic damping ratio are calculated to be $\zeta_m = 0.13$ and $\zeta = 0.03$, respectively.[4] For the same mean motor speed of 24Hz, the speed variation as well as current variation for the mechanism with a spring are smaller than those without a spring both in air and in vacuum. The RMS current (and torque) is also lower with a spring than without a spring, which implies that lower mechanical power is required with a spring to generate the same flapping frequency.

The average power and power reduction for various flapping frequencies are shown in Fig. 6.9. At high input voltages where the average motor speeds were greater than 15Hz, the average power for the system with spring became less than the system without spring. With extremely low aerodynamic damping in vacuum, we could achieve up to 60% power reduction. Up to 30% power reduction was also achieved when flapping at 15-30Hz with a constant input voltage of 1-3V DC. This can increase continuous flight time and decrease battery mass. The percent power reduction remained almost constant as the flapping frequency increases.

The effect of an integrated spring on the lift force generated by our flapping mechanism was briefly examined. The results showed a 20-30% increase in average lift force for the spring-integrated system, relative to the spring-less system operating at the same flapping frequency. The electrical input power was also reduced, as predicted. Hi-speed video showed a more symmetric upstroke and downstroke in the spring-integrated system. The experiment indicates that the spring, in addition to reducing input power at a given operating frequency, is also beneficially altering the flapping wing trajectory. This result essentially agrees with the study by Khan *et al* [18] in that appropriate spring constant and inertia can increase the lift force with less input power. In their study, the spring constant and inertia were chosen by numerical optimization.

To briefly verify our crank-arm model analysis on another system, an Interactive Toy's VAMP RC ornithopter was modified by adding a 980N/m spring as shown in Fig. 6.10 . The ornithopter flaps at 14-16Hz with a fully charged 40mAh lithium polymer battery. On-board electronics, battery, and styrofoam body were removed from the ornithopter, and the motor was directly connected to a power supply. The undamped natural frequency of the load was calculated as $\omega_n = 118.4$rad/s (18.8Hz) with the effective wing mass of 69.85×10^{-3}kg.

---

[3] The effective mass of the wings were calculated as $m_w = 2J_w/d^2$, where J_w is the moment inertia of a single wing and d is the distance between the fulcrum and the pin joint coupling the connecting rod and the lever arm.

[4] To calculate the aerodynamic damping, motor velocity and current for the flapping mechanism in vacuum as well as in air was measured. Using these measurements, the damping coefficient, b, was estimated.

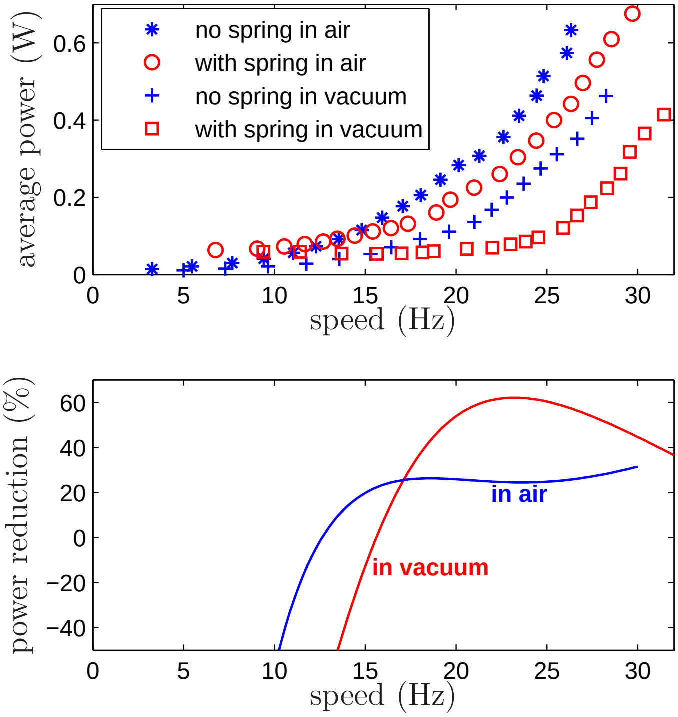


Figure 6.9: Measurement of average power (top) and power reduction using a spring (bottom) for a flapping wing robot driven by constant input voltages

Measurements were also taken without a spring to compare the power consumption.

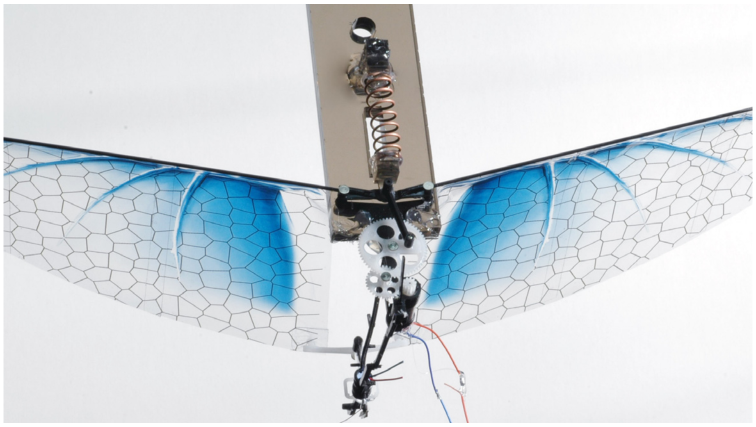


Figure 6.10: An Interactive Toy's VAMP RC ornithopter with an added spring.

Shown in Fig. 6.11 are the average power and power reduction for various flapping speeds. When the average motor speed was greater than 12Hz, the average power for the system with spring became less than for the system without spring. At 16Hz, up to 19% power reduction has been achieved. It would be possible to obtain a similar amount of power reduction even at higher flapping frequencies if the motor could operate at higher input voltages.

The experiments discussed in this section manifest that resonant excitation by a constant voltage can reduce the average power consumption for DC motor-driven flapping-wing robots. We have made use of the nondimensional analysis to carefully choose the spring constant for the compliant elements. As a result, a nontrivial amount of power reduction has been achieved for both flapping-wing robots.

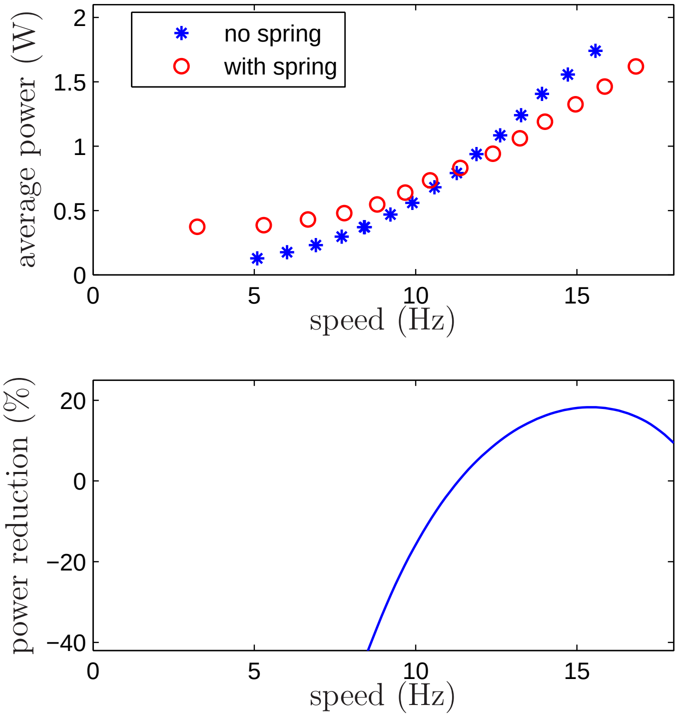


Figure 6.11: Measurement of average power (top) and power reduction using a spring (bottom) for an Interactive Toy's VAMP RC ornithopter.

Part IV

Conclusions

Chapter 7

Conclusions

This work describes the development of the components necessary for the realization of small-scale autonomous ornithopter flight. The height regulation described in Chapter 4 and the target-seeking algorithm described in Chapter 5 are the core development for autonomous ornithopter flight. The identification of the flight forces at a stable equilibrium discussed in Chapter 4 shows the direction toward the full system identification using closed-loop control. The resonant excitation for motor-driven robots with the nondimensional analysis described in Chapter 6 provides a methodology to guide the mechanical design and predict optimal operating points based on power efficiency.

7.1 Flight Control of Flapping-Wing Robots

As a first step toward autonomous flight, closed-loop altitude control using an external camera has been implemented on a flapping-wing micro aerial vehicle. An external camera running at 15 Hz has been used to measure the ornithopter's altitude. Also, a blob tracking algorithm with a Kalman filter has been developed to track the location of the ornithopter. Using these developments, we have demonstrated the ornithopter flying at regulated altitudes with the maximum error of 10 cm for a step input.

An analysis of flight forces acting on flapping wing robots explains the deviation between the measured forces and the free flight forces. Using this analysis and the height regulation, we have identified that tethered aerodynamic force measurement of a 12.4 gram ornithopter with zero induced velocity underestimates the total force by 24.8 mN, likely due to the absence of induced velocity and the tether restraining the body.

For autonomous flight for a 13 gram ornithopter without any external assistance, we have described implementation of closed-loop attitude control using onboard sensors and a microcontroller. For this implementation, we have developed a 1.0 gram microcontroller board integrated with inertial and visual sensors, a wireless transceiver, and motor drivers. An algorithm using a Kalman filter has been developed to predict the desired orientation angle when the target is temporarily not observable. With these developments, we have presented the ornithopter flying toward an infrared emitting target using an onboard infrared camera. The 28 cm wing-span ornithopter was able to land within a radius of 0.5 m from the target with more than 85% success (N = 20).

7.2 Resonant Excitation of DC Motor-Driven Flapping-Wing Robots

A nondimensional analysis of a motor driven compliant system has been developed to estimate the efficiency of power transmission to the environment for a resonant flapping-wing mechanism. With the nondimensional analysis of a motor driven compliant system, we have analytically developed a methodology to guide the mechanical design and predict optimal operating points based on power efficiency. We have experimentally demonstrated that the average power (as well as the peak torque) can be reduced by resonant excitation with a constant voltage for motor-driven oscillating systems with a spring. Moreover, simulations show that an average power of 0.43W with 0.13W peak-to-peak is required to drive a crank-arm system with a spring at the average speed of 150 rad/sec (s.d. = 7.7 rad/sec) with a constant voltage of 2.65V, whereas an average power of 0.48W with 0.22W peak-to-peak (not including the power for speed controller) is required for the constant speed of 150 rad/sec.[1] This 10% reduction in average power directly translates to a 10% increase in range or payload. In addition to this specific example, the generalized crank-arm model presented in this thesis can be easily applied to most motor-driven oscillating systems by constructing an equivalent crank-arm model.

7.3 Future Work

System identification of the ornithopter is an essential part of the control based on a state space model. The states of the ornithopter can be estimated by a sensor fusion algorithm using vision-based relative velocity sensing (optical flow) and inertial sensors. Also, direct measurement of acceleration using inertial sensors, which are already integrated in the control board, will enable us to measure free flight aerodynamic forces. With better understanding of the map between the input and the output of the ornithopter based on the system identification, we believe we can implement a better control policy for higher level behaviors, such as coordinated flight of multiple ornithopters or obstacle avoidance.

We can also take advantage of reinforcement learning for the control policies. However, real-time processing of reinforcement learning would be too expensive to run on the onboard

---

[1] The specifications of the system in the simulations are: $\eta = 8.6$, $\zeta_m = 0.1$, $\zeta = 0.01$, $\lambda = 0.2$, $K_t = 0.2$ Nmm/A, $K_e = 0.67$ mV-sec, and $J_m = 1.4$ gmm^2.

electronics. Instead, we can search the optimal policy on a ground station with a high computational power processor, and transmit the control signals via the wireless communication link.

Improvement of the ornithopter platform will be necessary to include additional actuators and sensors. First, we can modify the ornithopter's body to reduce the weight. The styrofoam tail can be replaced with carbon fiber frames and thin polymer sheets to reduce the weight. With this modification, we expect a weight reduction of at least 1.5 grams. We can also modify the wing transmission for higher propulsion forces. An increase in lift force can be achieved by increasing the wing stroke angle by modifying the crank-arm transmission system. Also, adding an additional actuator producing a pitch moment will improve the control of the pitch motion, which is coupled with the propulsive forces generated by the wings. For approximately 18 Hz pitch oscillations generated by the wings, a thin plate driven by a low bandwidth magnetic actuator would be feasible to generate a sufficient aerodynamic moment. Since the infrared camera has a narrow field of view, integrating another infrared camera or a small cellular phone camera with a wide field of view would be necessary for higher-level behavior applications such as multiple-target tracking and multiple coordinated flight.

Part V

Appendix

Appendix A

Dynamometer for DC Motors

In the design of motor-driven systems, the choice of a motor that can provide the required torque and speed, possibly with minimal power and weight, is one of the most significant process. For most of commercially-available DC motors, specifications of the motors, such as stalled torques, the unloaded motor speeds, and the maximum power outputs for various input voltages, are usually provided. For small pager motors, usually provided with limited specifications, stalled torques and unloaded speeds for various input voltages can be measured using simple apparatus such as a force/torque sensor. However, there is a significant drawback in the measurement of stalled torque. The motors are overheated if they are stalled, which results in inaccurate torque measurements and possibly serious damage to the motors.

Most of the time, motors are integrated with gears to generate higher torques in compensation for lower speeds. Theoretically, the motor properties with gears can be calculated with the gear ratio. For example, the torque coefficient with gears is given by

$$\text{torque coefficient with gears} = (\text{torque coefficient}) \times (\text{gear ratio})^2$$

under the assumption of no transmission loss in the gears. In reality, the friction in gears - especially, the friction in small-scale gears with the diameter of ∼1 cm - is nontrivial. Therefore, the direct measurements of torques and speeds with gears provide more accurate estimates of motor properties.

In this chapter, we introduce a method to determine motor properties - torque coefficient, speed constant, back EMF coefficient, and damping coefficient - using the bench-top dynamometer shown in Fig. A.1. The dynamometer, designed by Aaron Hoover at the Biomimetic Millisystems Lab at U.C. Berkeley, consists of a reflective photosensor (Omron EE-SY413 Reflective Photomicrosensor) and a speed encoder with large inertia. The speed encoder, printed on a sheet of paper, is attached to an acrylic plate, and it is driven by a motor (it can also be driven via gears). The motor is connected to a power supply for a constant voltage. The photosensor, consisting of a light detector and a light emitter, outputs 5 volts when the emitted light is reflected by the white surface on the speed encoder and 0

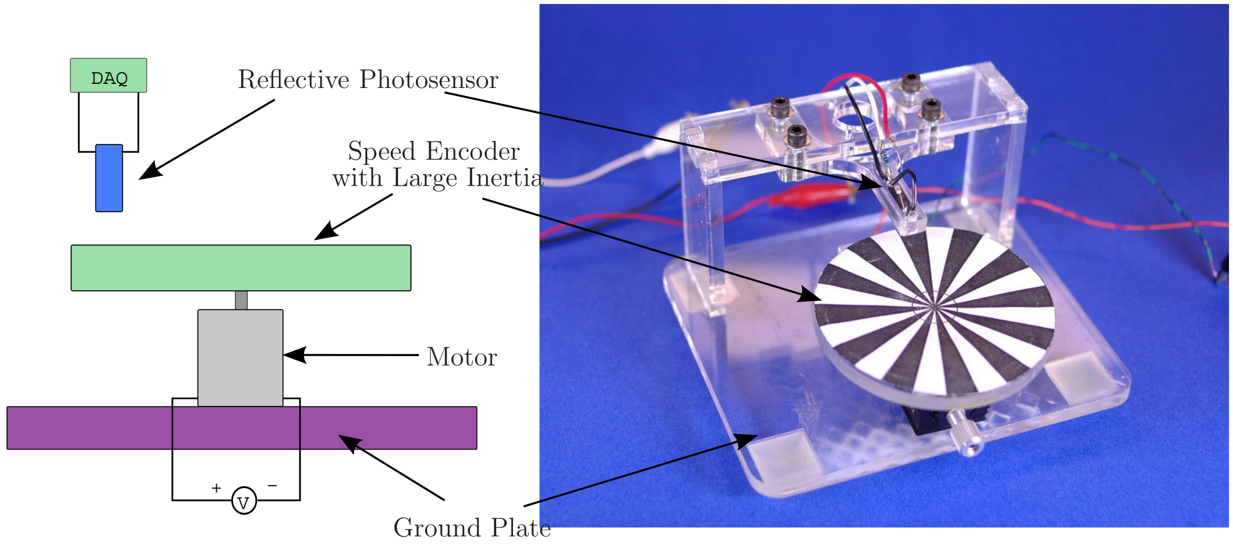


Figure A.1: Benchtop dynamometer for DC motors. The apparatus pictured on the right is designed by Aaron Hoover at the Biomimetic Millisystems Lab at U.C. Berkeley.

volts for the black surface. The output is captured at 25 KHz to estimate the angular speed of the motor.

A.1 Estimation of Motor Properties

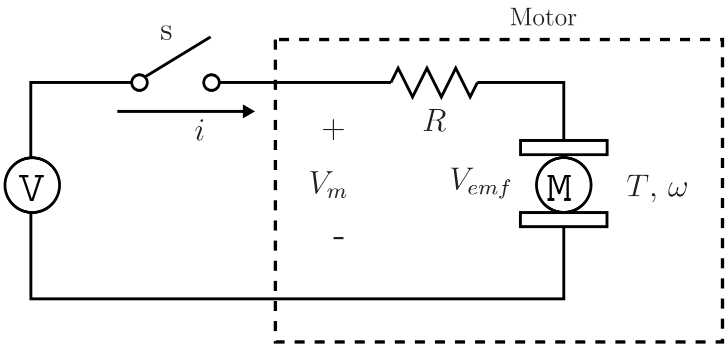


Figure A.2: A circuit model for a DC motor

The equation of motion for a DC motor is expressed as:

$$J_m\dot{\omega} + b_m\omega = T, \tag{A.1}$$

where ω is the speed of the motor, J_m is the moment of inertia of the speed encoder (the moment of inertia of the motor can be ignore since it is much smaller than the moment of inertia of the speed encoder), b_m is the damping coefficient of the motor, and T is the torque generated by the motor. The defining equations of a DC motor are

$$\begin{aligned} T &= k_t i, && \text{(A.2)} \\ L\frac{di}{dt} + iR + k_e\omega &= V, && \text{(A.3)} \end{aligned}$$

where V is the input voltage, L is the inductance of the motor, i is the induced current, R is the internal motor resistance, k_e is the back EMF coefficient, and k_t is the torque coefficient of the motor. Since L is generally very small, we can safely set $L = 0$. Figure A.2 shows the equivalent circuit model for a DC motor with $L = 0$.

Suppose that the switch, s, in Fig. A.2 is closed at $t = 0$. Then, the motor speed, ω, gradually increases until it reaches the steady state, where $\dot{\omega} = 0$. The transient and steady state period are denoted by **I** and **II**, respectively, in Fig. A.3. After the motor speed stays its steady state value for approximately 5 seconds (or at $t = 10$ s), the switch is reopened. This results in $i = 0$ with nonzero back EMF voltage, i.e., $V_{emf} \neq 0$ since $\omega \neq 0$. Fig. A.3a and Fig. A.3b show, respectively, the motor speed, ω, and the motor voltage, V_m.

A.1.1 Damping Coefficient

For $t > 10$ s, we have from (A.2) and (A.1)

$$J_m\dot{\omega} + b_m\omega = 0. \tag{A.4}$$

We know the solution of (A.4) is

$$\omega = A\exp(-\frac{b_m}{J_m}t), \tag{A.5}$$

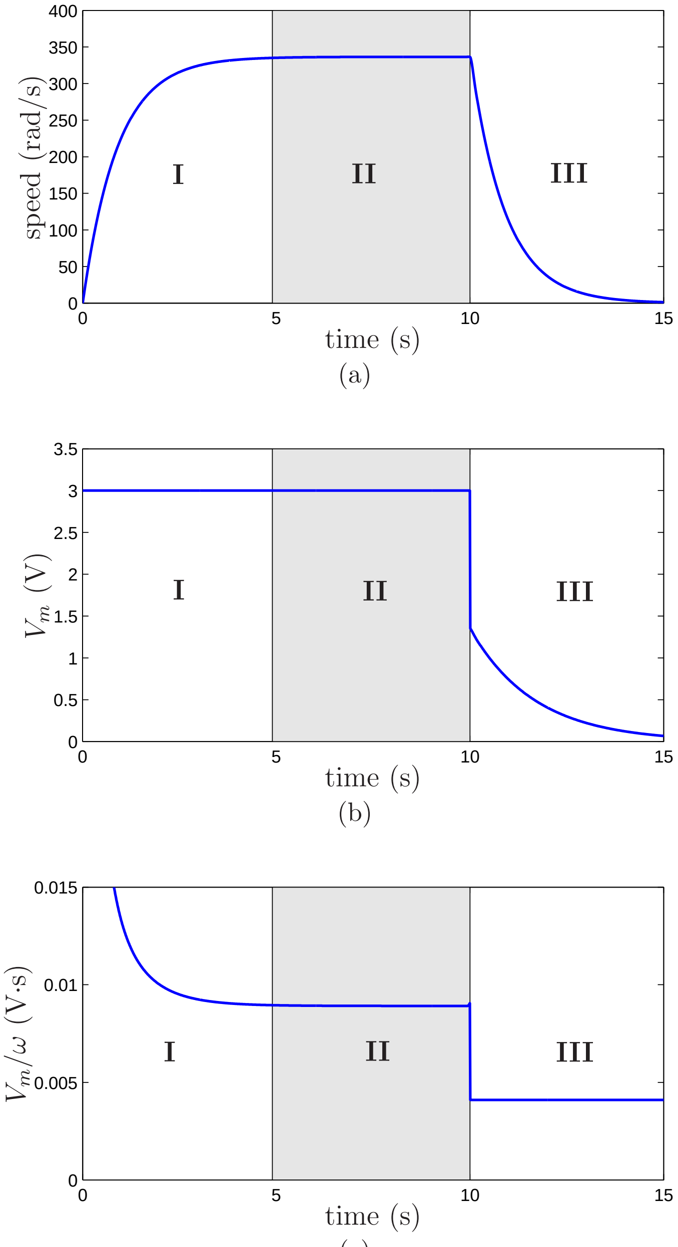
where the initial motor speed, A, is the steady state motor speed at $t = 10$ s. This exponential function is depicted in the region **III** in Fig. A.3a. The time constant, $\tau \triangleq J_m/b_m$, can be determined by a curve fitting function, such as the Matlab *polyfit* function, after taking the logarithm of the curve in the region **III** in Fig. A.3a. Then, the motor damping coefficient, b_m, can be estimated by J_m/τ with a known value of J_m.

A.1.2 Back EMF Coefficient

We can also determine k_e with $i = 0$ for $t > 10$ s from (A.3), which is given by

$$k_e = \frac{V_m}{\omega}.$$

The region **III** in Fig. A.3c shows the ratio of the motor voltage to the motor speed, equivalently the back EMF coefficient of the motor, k_e.



(a)

(b)

(c)

Figure A.3: An example of (a) motor speed, (b) motor voltage, and (b) their ratio. The region **I** is the transient period, the region **II** is the steady state period, and the region **III** is the zero input current period.

A.1.3 Torque Coefficient

In order to estimate the torque coefficient, k_t, we can combine (A.1), (A.2), and (A.3) with $L = 0$ to obtain

$$J_m \dot{\omega} + b_m \omega = k_t \frac{V - k_e \omega}{R}. \tag{A.6}$$

With $\dot{\omega} = 0$ at the steady state, we have

$$k_t = \frac{R b_m \omega}{V - k_e \omega} = \frac{R b_m \frac{\omega}{V}}{1 - k_e \frac{\omega}{V}} = \frac{R b_m k_s}{1 - k_e k_s},$$

where the speed constant, k_s, is the ratio of the steady state motor velocity to the input voltage, i.e., $k_s \triangleq \omega / V$ given that $\dot{\omega} = 0$. The region **II** in Fig. A.3c shows $1/k_s$.

A.2 Estimation of Maximum Power

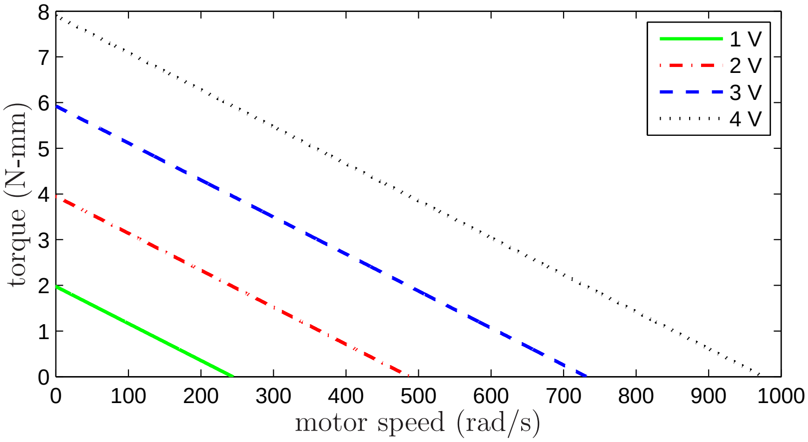


Figure A.4: Motor torque vs. motor speed for various input voltages.

Using (A.2) and (A.3) with $L = 0$, we can derive the torque-velocity equation,

$$T = \frac{k_t}{R} V - \frac{k_e k_t}{R} \omega.$$

With the measurements of ω and V_m and the estimates of k_t and k_e for various constant input voltages, we can draw torque-speed curves for a motor, such as shown in Fig. A.4. The mechanical power generated by the motor is given by

$$p = T \cdot \omega = \frac{k_t}{R} V \cdot \omega - \frac{k_e k_t}{R} \omega^2. \tag{A.7}$$

To obtain the argument of $\hat{\omega}$ that maximizes the mechanical power, we take a derivative of p with respect to ω and set it to zero;

$$\frac{dp}{d\omega} = \frac{k_t}{R}V - 2\frac{k_e k_t}{R}\omega = 0.$$

Then, the solution is

$$\hat{\omega} = \frac{V}{2k_e}. \tag{A.8}$$

It should be noted that $\hat{\omega}$ is just a half of the unloaded motor speed - the x-intercepts in Fig. A.4 are the unloaded motor speeds. The maximum power can be calculated by substituting (A.8) into (A.7), which is given by

$$p_{max} = \frac{1}{4R}\frac{k_t}{k_e}V^2.$$

A.3 Specifications of Small DC Motors

Using the dynamometer, we measured the motor speed and the motor voltage with various input voltages. Then, using the method discussed in this chapter, we estimated the properties of Didel[1] and Vamp[2] motors and summarized in Table A.1.

---

[1] www.didel.com

[2] The RC ornithopter pictured in Fig. 6.10.

Table A.1: Specifications of Didel and Vamp motors. The maximum power and the power density were estimated at V = 4 V.

Motors	b_m (gram·mm^2/s)	k_e (mV·s)	k_t (N·mm/A)	k_s (rad/V·s)	max power (mW)	power density (W/kg)
Didel 04S-10	8.61	0.78	0.71	810	97	142
Didel 04S-24	6.49	0.48	0.37	1034	111	236
Didel 04-10	9.01	0.55	0.37	1265	269	389
Didel 04-40	8.62	0.78	0.71	810	97	142
Didel 06-25	14.59	0.57	0.45	1003	240	198
Vamp motor	8.26	0.67	0.19	1239	321	208

Appendix B

Schematics of ImageProc2

The schematics and the bill of material (BOM) for the control board, aka ImageProc2, are attached in this chapter.

Table B.1: Bill of material for ImageProc2

Part	Value	Device
BALUN	HHM1710D1	BALUNA
C1	0.1uF	CAP0402-CAP
C2	1nF	CAP0402-CAP
C3	1uF	CAP0402-CAP
C4	1uF	CAP0402-CAP
C5	1uF	CAP0402-CAP
C6	1uF	CAP0402-CAP
C7	2.2nF	CAP0402-CAP
C8	0.1uF	CAP0402-CAP
C9	0.1uF	CAP0402-CAP
C10	10nF	CAP0402-CAP
C11	1nF	CAP0402-CAP
C12	10uF	CAP0603-CAP
C13	0.1uF	CAP0402-CAP
C14	0.1uF	CAP0402-CAP
C15	1uF	CAP0402-CAP
C16	1uF	CAP0402-CAP
C17	1uF	CAP0402-CAP
C18	0.1uF	CAP0402-CAP
C19	22pF	CAP0402-CAP
C20	22pF	CAP0402-CAP
C21	12pF	CAP0402-CAP

continued on next page

Table B.1: Bill of material for ImageProc2 *continued*

Part	Value	Device
C22	12pF	CAP0402-CAP
C23	0.1uF	CAP0402-CAP
C24	1uF	CAP0402-CAP
C25	1uF	CAP0402-CAP
C26	1uF	CAP0402-CAP
C27	2.2uF	CAP0402-CAP
C28	0.1uF	CAP0402-CAP
C29	10uF	CAP0603-CAP
D1	PMEG2015EJ	DSOD323
JP1	MPCS2PH	MPCS2PH
JP2	51441-2493-C	OV7660FSL_CON-24
JP3	ICD2	MPCS6PH
JP4	WII	MPCS6PH
JP5	MPCS6PH	MPCS6PH
JP6	MPCS2PH	MPCS2PH
JP7	MPCS2PH	MPCS2PH
JP8	MPCS2PH	MPCS2PH
JP9	MPCS2PH	MPCS2PH
JP10		SOLDER_PADS_9P
LED1	Red	LEDCHIP-LED0603
LED2	Green	LEDCHIP-LED0603
LED3	Yellow	LEDCHIP-LED0603
PROC	DSPIC33FJ128MC706A	DSPIC33FJ128MC706-I/PT
Q1	ZXMN2B01FCT	MOSFET-NCHANNELSMD
R1	1k	R-US_R0402
R2	100k	R-US_R0402
R3	1k	R-US_R0402
R4	1k	R-US_R0402
R5	4.7k	R-US_R0402
R6	4.7k	R-US_R0402
R7	4.7k	R-US_R0402
R8	4.7k	R-US_R0402
R9	10k	R-US_R0402
R10	27k	R-US_R0402
R12	470	R-US_R0402
R13	1k	R-US_R0402
R14	470	R-US_R0402

continued on next page

Table B.1: Bill of material for ImageProc2 *continued*

Part	Value	Device
R15	470	R-US_R0402
U1	MIC5335-SPYMT	MIC5320-JGYML
U2	MIC5320-JGYML	MIC5320-JGYML
U3	ITG-3200	ITG-3200-00
U4	AT86RF231-ZU	AT86RF231-ZU
U5	AT45DB321D	AT45DB321D
U6	ECS-2033	OSCILLATORS
U7	A3903S	A3903S
U8	ADXL345-LGA12	ADXL345
U9	MN13822-L	VOL_DETECTOR
U10	ECS-160-8	CRYSTALS
U11	ANTENNA	ANTENNA
U12	ANTENNA-CHIP	ANTENNA-CHIP

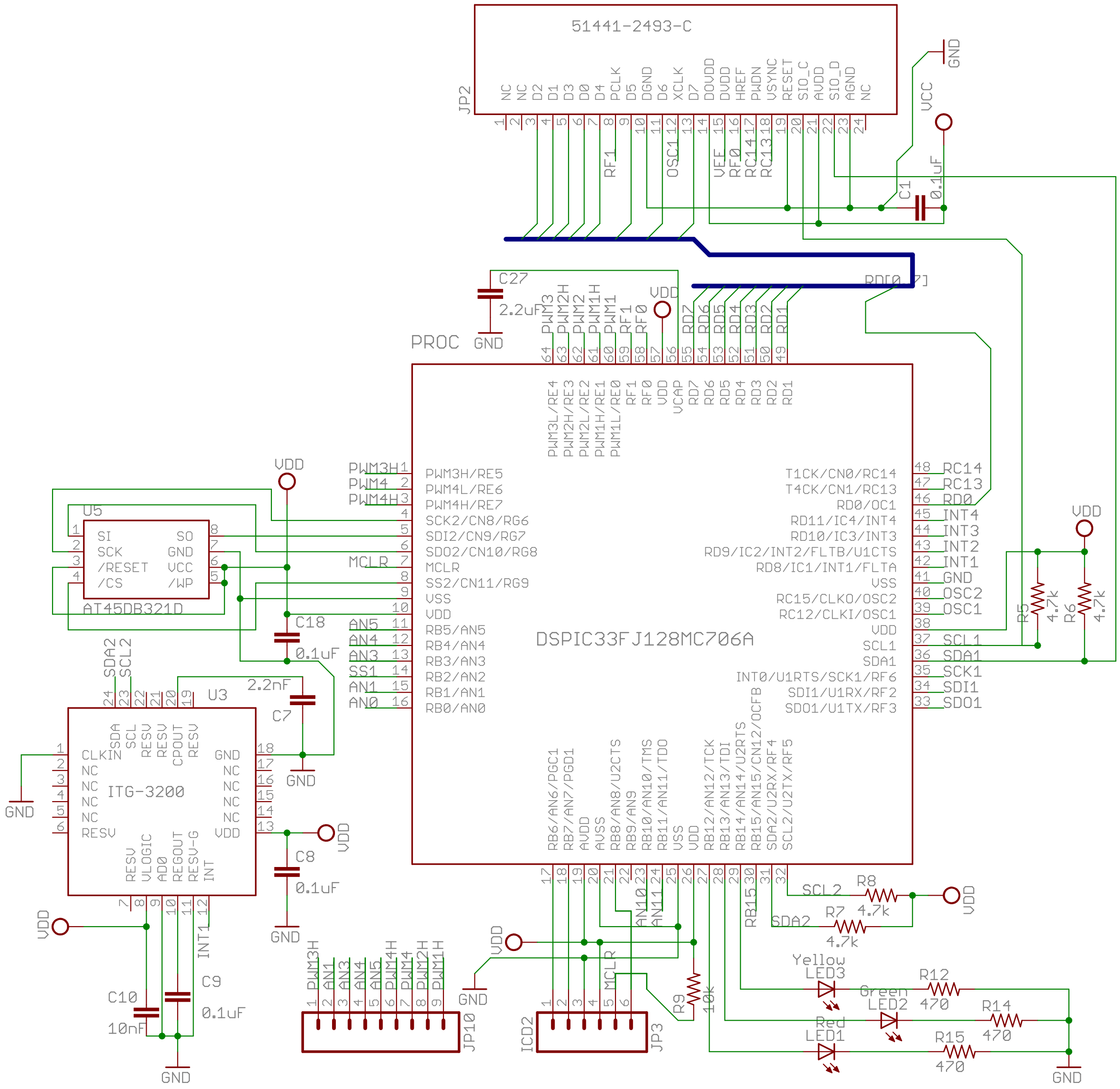


Figure B.1: Schematics of ImageProc2 - microcontroller, gyroscope, flash memory, and cell phone camera connector

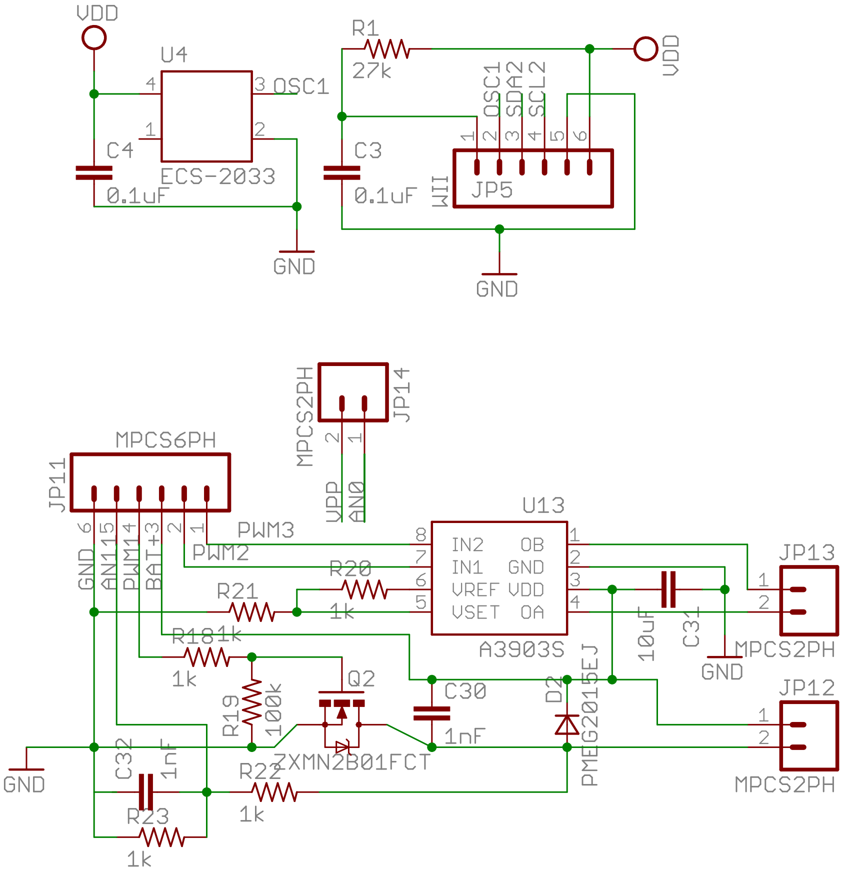


Figure B.2: Schematics of ImageProc2 - motor driver, oscillator, and infrared camera connector

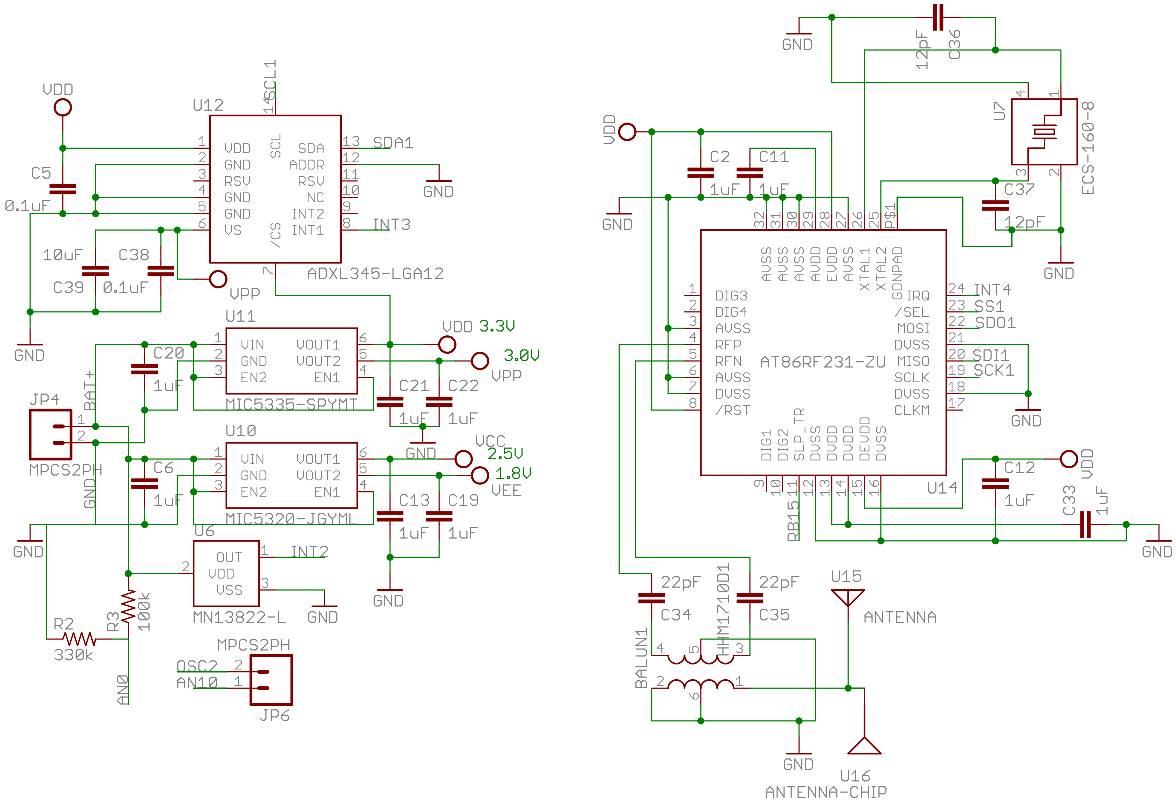


Figure B.3: Schematics of ImageProc2 - wireless transceiver, accelerometer, battery monitor, and voltage regulators

Appendix C

Matlab Simulation Code for Resonant Drive

The Matlab simulation codes used for the plots in Chapter 6 are attached in this chapter.

Listing C.1: This code generates Fig. 6.3 and Fig. 6.4a.

```
% Analysis of the dynamics of a crank arm connected to a
% mass-spring-damping system.
% Last modified on July 2, 2009
% Author: Stan Baek

lambda = .1; % ratio of crank arm length to connecting arm length
zeta_m = 0.1; % damping ratio of the motor
zeta = 0.1; % damping ration of the slider

theta = linspace(-pi,pi,30); % angular position of the crank arm
omega = linspace(0, 2, 30); % nondimensional angular velocity
[Theta Omega] = meshgrid(theta,omega);

beta = cos(Theta)...
    + lambda*cos(Theta).*sin(Theta)./sqrt(1-lambda^2.*cos(Theta).^2);
u = 2*zeta_m*Omega + beta.*sin(Theta)...
    + Omega.^2.*beta.*(2*zeta*sign(cos(Theta))*cos(Theta).^2 - sin(Theta));

mesh(Theta, Omega, u);
set(gcf,'Position', [474   399   426   250]);
set(gca, 'XTick', [-pi -pi/2 0 pi/2 pi]);
set(gca, 'XTickLabel', {'-pi' '-pi/2' '0' 'pi/2' 'pi'});
set(gca, 'YTick', [0 0.5 1 1.5 2]);
set(gca, 'YTickLabel', {'0' '0.5' '1' '1.5' '2'});
xlabel('theta');
ylabel('Omega');
```

```
zlabel({'T'});
axis([-pi pi 0 2 -1 3.2])

omega = [0.8 0.9 1 1.1 1.2];
num_plot = length(omega);
cmap = colormap('jet');
colorOrder = cmap(floor(length(cmap)/num_plot): ...
    floor(length(cmap)/num_plot):length(cmap),:);

% Effect of lambda
omega = 0.9;
zeta_m = 0.1;
zeta_s = 0.1;
Lambda = [0.4 0.3 .2 0.1 0.01];
Theta = linspace(-pi,pi,50);
figure
set(gcf,'Position', [474   399   426   250]);
for i = 1:length(Lambda)
    Omega=omega;
    lambda = Lambda(i);

    mu = lambda*cos(Theta)./sqrt(1-lambda^2*sin(Theta).^2);
    gamma = (1+mu).*sin(Theta);
    dmu = -lambda*sin(Theta)./sqrt(1-lambda^2*sin(Theta).^2)+...
      lambda^3*cos(Theta).^2.*sin(Theta)./(1-lambda^2*sin(Theta).^2).^(3/2);

    T = Omega.*(2*zeta_m + 2*zeta_s*gamma.^2)+...
       Omega.^2.*gamma.*(cos(Theta) + mu.*cos(Theta) + dmu.*sin(Theta))...
        -gamma.*(cos(Theta) + cos(Theta)./mu - sqrt(1/lambda^2 -1) );

    plot(Theta, T, 'Color',colorOrder(i,:));
    hold on;
end

legend( strcat('\lambda = ', num2str(Lambda')));
set(gca, 'XTick', [-pi -pi/2 0 pi/2 pi]);
set(gca, 'XTickLabel', {'-pi' '-pi/2' '0' 'pi/2' 'pi'});
xlabel('theta');
ylabel('T');
```

Listing C.2: This code generates Fig. 6.4b.

```
% Analysis of the dynamics of a crank arm connected to a
% mass-spring-damping system.
% This script shows the output power of a geared motor with
% respect to constant driving angular speed.
% When the system is excited at the resonant frequency,
% the required output power is going to be minimum.
```

```
Theta = linspace(-pi,pi,300);
zeta = [0.1, 0.2, 0.3, 0.4];
zeta_m = 0.0;
lambda = 1/10;
OMEGA = linspace(0.001, 2, 200);
beta = cos(Theta) + ...
    lambda*cos(Theta).*sin(Theta)./sqrt(1-lambda^2.*cos(Theta).^2);

Power = zeros(length(zeta), length(OMEGA));

for j = 1:length(zeta)
    for i = 1:length(OMEGA)
        Omega = OMEGA(i);

        T = Omega*2*zeta_m + ...
            Omega^2*beta.*(2*zeta(j)*sign(cos(Theta)).*cos(Theta).^2 - ...
            sin(Theta)) + beta.*sin(Theta);

        Power(j, i) = norm(T*Omega)/sqrt(length(T));
    end
end

figure;
plot(OMEGA, Power(1,:), 'b', OMEGA, Power(2,:), 'r');
hold on;
plot(OMEGA, Power(3,:), 'g', OMEGA, Power(4,:), 'k');
xlabel('Omega');
ylabel('MechPower');
axis([0 1.5 0 1]);
```

Listing C.3: This code generates Fig. 6.5.

```
function ode_analysis

global Jm lambda Kt Ke Rm bm eta m r b k

Jm = 1.4107e-9;
eta = 8.57;
r = 4e-3;
l = 17e-3;
lambda = r/l;

Kt = 0.1916e-3;
Ke = 0.6723e-3;
Rm = 3.2;

leverarm = 7e-3;
```

```
J = 142.91e-9;
m = 2*J/leverarm^2;

bm = 6.8196e-10; %zeta_m = 0.002;
b = 2.9165*0.01; %zeta = 0.01;
k = 105;
omega_n = sqrt(k/m);

u = [0.20 0.25 0.30];

Vin = u*m*r^2*Rm*omega_n^2/(eta*Kt);

numPlot = length(Vin);
cmap = colormap('jet');
colors = cmap(1:ceil((length(cmap))/(numPlot-1))-1:length(cmap),:);

figure(1);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('Omega');

figure(2);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('T');
hold on

figure(3);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('Power');
hold on

numSamples = 2000;

for i = 1:length(Vin)

    theta_init = 0;
    dtheta_init = 200;

    init = [theta_init, dtheta_init];
    tint = linspace(0,2,numSamples);    %time interval over which solution ...
       is desired.
```

```
    options = odeset('RelTol',1e-4,'AbsTol',[1e-4]);
    [t,Y]=ode45(@dynamics, tint, init, options, Vin(i));

    %%%%%%%%%%%%%%%%%%%%%    PLOTS    ***************
    theta = Y(:,1);
    omega = Y(:,2);

    Vemf = eta*Ke*omega;
    current = (Vin(i) - Vemf)/Rm;
    torque = eta*Kt*current;

    Omega = omega(numSamples/2:end)/omega_n;
    tau = omega_n*(t(numSamples/2:end)-t(numSamples/2));
    figure(1)
    plot(tau, Omega, 'color', colors(i,:));

    T = torque(numSamples/2:end)/(m*r^2*omega_n^2);
    figure(2);
    plot(tau, T, 'color', colors(i,:));

    P = T.*Omega;
    figure(3);
    plot(tau, P, 'color', colors(i,:));
    mean(P)
end

figure(1);
legend('u=0.20', 'u=0.25', 'u=0.30', 'orientation', 'horizontal', ...
   'location', 'south');
axis( [0 0.1*omega_n 0 1.5]  );
set(gca,'FontSize',11);

figure(2);
legend('u=0.20', 'u=0.25', 'u=0.30', 'orientation', 'horizontal');
axis( [0 0.1*omega_n -0.02 0.06]  );
set(gca,'FontSize',11);

figure(3);
legend('u=0.20', 'u=0.25', 'u=0.30', 'orientation', 'horizontal');
axis( [0 0.1*omega_n -0.02 0.06]  );
set(gca,'FontSize',11);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function dy=dynamics(t,y, u)
%dynamics.m returns derivative information to Matlab
global Jm lambda Kt Ke Rm bm eta m r b k

dy = zeros(length(y),1);
```

```
theta = y(1);
dtheta = y(2);

beta = cos(theta) + ...
   lambda*cos(theta)*sin(theta)/sqrt(1-lambda^2*cos(theta)^2);

J = m*r^2*beta*cos(theta) + eta^2*Jm;
B1 = eta^2*(bm + Kt*Ke/Rm);
B2 = beta*(sign(cos(theta))*b*r^3*cos(theta)^2 - m*r^2*sin(theta));
K = beta*k*r^2*sin(theta);
U = eta*Kt/Rm*u;

ddtheta = 1/J * (U - B1*dtheta - B2*dtheta^2 - K);
dy(1) = dtheta;
dy(2) = ddtheta;
```

Listing C.4: This code generates Fig. 6.6.

```
function ode_analysis

global Jm lambda Kt Ke Rm bm eta m r b k

Jm = 1.4107e-9;
eta = 8.57;
r = 4e-3;
l = 17e-3;
lambda = r/l;

Kt = 0.1916e-3;
Ke = 0.6723e-3;
Rm = 3.2;

leverarm = 7e-3;
J = 142.91e-9;
m = 2*J/leverarm^2;

bm = 6.8196e-10; %zeta_m = 0.002;
B = 2.9165*[0.01 0.1 0.3];   %zeta = b/2.9165
k = 105;
omega_n = sqrt(k/m);

u = 0.25;
Vin = u*m*r^2*Rm*omega_n^2/(eta*Kt);

numPlot = length(B);
cmap = colormap('jet');
colors = cmap(1:ceil((length(cmap))/(numPlot-1))-1:length(cmap),:);
```

```
figure(1);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('Omega');

figure(2);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('T');
hold on

figure(3);
set(gcf,'DefaultLineLineWidth',1.2);
set(gcf, 'position', [616   409   424   239]);
hold on;
xlabel('time');
ylabel('Power');
hold on

numSamples = 2000;

for i = 1:length(B)

    theta_init = 0;
    dtheta_init = 200;

    b = B(i);

    init = [theta_init, dtheta_init];
    tint = linspace(0,2,numSamples);    %time interval over which solution ...
       is desired.
    options = odeset('RelTol',1e-4,'AbsTol',[1e-4]);
    [t,Y]=ode45(@dynamics, tint, init, options, Vin);

    %%%%%%%%%%%%%%%%%%%%%    PLOTS    ***************
    theta = Y(:,1);
    omega = Y(:,2);

    Vemf = eta*Ke*omega;
    current = (Vin - Vemf)/Rm;
    torque = eta*Kt*current;

    Omega = omega(numSamples/2:end)/omega_n;
    tau = omega_n*(t(numSamples/2:end)-t(numSamples/2));
    figure(1);
```

```
    plot(tau, Omega, 'color', colors(i,:), 'LineWidth', 1);

    T = torque(numSamples/2:end)/(m*r^2*omega_n^2);
    figure(2);
    plot(tau, T, 'color', colors(i,:), 'LineWidth', 1);

    P = T.*Omega;
    figure(3);
    plot(tau, P, 'color', colors(i,:));

end

figure(1);
legend('\zeta=0.01', '\zeta=0.1', '\zeta=0.3', 'orientation', 'horizontal');
axis( [0 0.1*omega_n 0 1.5]  );
set(gca,'FontSize',11);

figure(2);
legend('\zeta=0.01', '\zeta=0.1', '\zeta=0.3', 'orientation', 'horizontal');
axis( [0 0.1*omega_n -0.04 0.3]  );
set(gca,'FontSize',11);

figure(3);
legend('\zeta=0.01', '\zeta=0.1', '\zeta=0.3', 'orientation', 'horizontal');
axis( [0 0.1*omega_n -0.04 0.13]  );
set(gca,'FontSize',11);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function dy=dynamics(t,y, u)
%dynamics.m returns derivative information
global Jm lambda Kt Ke Rm bm eta m r b k

dy = zeros(length(y),1);
theta = y(1);
dtheta = y(2);

beta = cos(theta) + ...
   lambda*cos(theta)*sin(theta)/sqrt(1-lambda^2*cos(theta)^2);

J = m*r^2*beta*cos(theta) + eta^2*Jm;
B1 = eta^2*(bm + Kt*Ke/Rm);
B2 = beta*(sign(cos(theta))*b*r^3*cos(theta)^2 - m*r^2*sin(theta));
K = beta*k*r^2*sin(theta);
U = eta*Kt/Rm*u;

ddtheta = 1/J * (U - B1*dtheta - B2*dtheta^2 - K);
dy(1) = dtheta;
dy(2) = ddtheta;
```

Bibliography

[1] S. Avadhanula, R. Wood, D. Campolo, and R. Fearing, “Dynamically tuned design of the MFI thorax,” in *IEEE International Conference on Robotics and Automation*, Washington, DC, May 2002, pp. 52–59.

[2] S. Avadhanula, R. Wood, E. Steltz, J. Yan, and R. Fearing, “Lift force improvements for the micromechanical flying insect,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, October 2003, pp. 1350–1356.

[3] S. Baek, K. Ma, and R. Fearing, “Efficient resonant drive of flapping-wing robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, October 2009, pp. 2854–2860.

[4] A. Cox, D. Monopoli, D. Cveticanin, M. Goldfarb, and E. Garcia, “The development of elastodynamic components for piezoelectrically actuated flapping micro-air vehicles,” *Journal of Intelligent Material Systems and Structures*, vol. 13, pp. 611–615, September 2002.

[5] J. L. Crassidis and F. L. Markely, “Unscented filtering for spacecraft attitude estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 26, pp. 536–542, August 2003.

[6] G. de Croon, K. de Clerq, R. Ruijsink, B. Remes, and C. de Wagter, “Design, aerodynamics, and vision-based control of the DelFly,” *International Journal of Micro Air Vehicles*, vol. 1, no. 2, pp. 71 – 97, 2009.

[7] G. de Croon, E. de Weerdt, C. de Wagter, and B. Remes, “The appearance variation cue for obstacle avoidance,” in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2010, pp. 1606–1611.

[8] (2009) Delfly. Delft University of Technology. The Netherlands. [Online]. Available: http://www.delfly.nl/

[9] X. Deng, L. Schenato, and S. S. Sastry, “Flapping flight for biomimetic robotic insects: part II - flight control design,” *IEEE Trans. on Robotics*, vol. 22, no. 4, pp. 789–803, 2006.

[10] X. Deng, L. Schenato, W. C. Wu, and S. S. Sastry, "Flapping flight for biomimetic robotic insects: part I - system modeling," *IEEE Trans. on Robotics*, vol. 22, no. 4, pp. 776–788, 2006.

[11] C. Ellington and J. Usherwood, *Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Applications*. AIAA, 2001, ch. 12: Lift and drag characteristics of rotary and flapping wings, pp. 231–248.

[12] R. Fearing, K. Chiang, M. Dickinson, D. Pick, M. Sitti, and J. Yan, "Wing transmission for a micromechanical flying insect," in *IEEE International Conference on Robotics and Automation*, vol. 2, 2000, pp. 1509–1516.

[13] G. Fischer, A. G. Cox, M. Gogola, M. K. Gordon, N. Lobontiu, D. Monopoli, E. Garcia, and M. Goldfarb, "Elastodynamic locomotion in mesoscale robotic insects," in *SPIE Conference on Electroactive Polymer Actuators and Devices*, vol. 3669, March 1999, pp. 362–368.

[14] F. Garcia Bermudez and R. Fearing, "Optical flow on a flapping wing robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, October 2009, pp. 5027–5032.

[15] C. W. Kang and C. G. Park, "Attitude estimation with accelerometers and gyros using fuzzy tuned Kalman filter," in *the European Control Conference*, Budapest, Hungary, August 2009, pp. 3713–3718.

[16] Y. Kawamura, S. Souda, S. Nishimoto, and C. Ellington, *Bio-mechanisms of Swimming and Flying: Fluid Dynamics, Biomimetic Robot, and Sports Science*. Springer, 2008, ch. 26: Clapping-wing Micro Air Vehicle of Insect Size, pp. 319–330.

[17] M. Keennon and J. Grasmeyer, "Development of the Black Widow and Microbat MAVs and a vision of the future of MAV design," in *AIAA International Air and Space Symposium and Exposition: The Next 100 Years*, Dayton, Ohio, July 2003.

[18] Z. Khan, K. Steelman, and S. Agrawal, "Development of insect thorax based flapping mechanism," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 3651–3656.

[19] Z. A. Khan and S. K. Agrawal, "Design of flapping mechanisms based on transverse bending phenomena in insects," in *IEEE International Conference on Robotics and Automation*, Orlando, FL, May 2006, pp. 2323–2328.

[20] ——, "Design and optimization of a biologically inspired flapping mechanism for flapping wing micro air vehicles," in *IEEE International Conference on Robotics and Automation*, Rome, Italy, April 2007, pp. 373–378.

[21] J. P. Khatait, S. Mukherjee, and B. Seth, “Compliant design for flapping mechanism: A minimum torque approach,” *Mechanism and Machine Theory*, vol. 41, pp. 3–16, 2006.

[22] D. B. Kingston and A. W. Beard, “Real-time attitude and position estimation for small UAVs using low-cost sensors,” in *AIAA 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*, 2004, pp. 2004–6488.

[23] D. Lentink, “Exploring the biofluiddynamics of swimming and flight,” Ph.D. dissertation, Wageningen University, 2008.

[24] D. Lentink, S. R. Jongerius, and N. L. Bradshaw, *Flying Insects and Robots*. Edited by Dario Floreano, Jean-Christophe Zuffery, Mandyam V. Srinavasan, and Charlie Ellington, Springer, 2009, ch. 14: The Scalable Design of Flapping Micro-Air Vehicles Inspired by Insect Flight, pp. 185–205.

[25] N. O. Lobontiu, M. K. Gordon, G. Fischer, E. Garcia, and M. Goldfarb, “Design and analysis of elastodynamic locomotion for robotic insects,” in *SPIE conference on Microrobotics and Micromanipulation*, vol. 3519, November 1998, pp. 118–127.

[26] R. Madangopal, Z. Khan, and S. K. Agrawal, “Energetics-based design of small flapping-wing micro air vehicles,” *IEEE/ASME Transactions on Mechatronics*, vol. 11, no. 4, p. 433, August 2006.

[27] R. Madangopal, Z. A. Khan, and S. K. Agrawal, “Biologically inspired design of small flapping wing air vehicles using four-bar mechanism and quasi-steady aerodynamics,” *ASME Journal of Mechanical Design*, vol. 127, pp. 809–816, July 2005.

[28] D. Mellinger, M. Shomin, and V. Kumar, “Control of quadrotors for robust perching and landing,” in *Proceedings of the International Powered Lift Conference*, Oct 2010.

[29] N. Michael, J. Fink, and V. Kumar, “Cooperative manipulation and transportation with aerial robots,” *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, Jan 2011.

[30] R. C. Michelson and S. Reece, “Update on flapping wing micro air vehicle research: Ongoing work to develop a flapping wing, crawling “Entomopter”,” in *13th Bristol International RPV Conference*, April 1998, pp. 19 – 24.

[31] L. A. Miller and C. S. Peskin, “A computational fluid dynamics of ‘clap and fling’ in the smallest insects,” *The Journal of Experimental Biology*, vol. 208, pp. 195–212, 2004.

[32] Q.-V. Nguyen, H. Park, N. Goo, and D. Byun, “A flying insect-like flapper actuated by a compressed LIPCA,” in *IEEE International Conference on Robotics and Biomimetics*, December 2007, pp. 19 – 24.

[33] O. M. O'Reilly, *Intermediate Dynamics for Engineers.* New York, NY: Cambridge University Press, 2008.

[34] U. Pesavento and Z. J. Wang, "Flapping wing flight can save aerodynamic power compared to steady flight," *Physical Review Letter*, vol. 103, no. 11, p. 118102, September 2009.

[35] T. N. Pornsin-Sirirak, S. W. Lee, H. Nassef, J. Grasmeyer, Y. Tai, C. Ho, and M. Keennon, "MEMS wing technology for a battery powered ornithopter," in *The 13th IEEE Annual International Conference on MEMS*, 2000, pp. 709–804.

[36] T. Pornsin-Siririak, Y. Tai, H. Nassef, and C. Ho, "Titanium-alloy MEMS wing technology for a micro aerial vehicle application," *Journal of Sensors and Actuators A: Physical*, vol. 89, pp. 95–103, March 2001.

[37] K. S. Shigeoka, "Velocity and altitude control of an ornithopter micro aerial vehicle," Master's thesis, The University of Utah, 2007.

[38] I. Shimoyama, H. Miura, K. Suzuki, and Y. Ezura, "Insect-like microrobots with external skeletons," *IEEE Control Systems Magazine*, vol. 13, no. 1, pp. 37–41, February 1993.

[39] A. R. Smith, "Color gamut transform pairs," in *SIGGRAPH 78*, August 1978.

[40] E. Steltz, S. Avadhanula, and R. Fearing, "High lift force with 275 Hz wing beat in MFI," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, 2007, pp. 3987–3992.

[41] T. Tantanawat and S. Kota, "Design of compliant mechanisms for minimizing input power in dynamic applications," *ASME Journal of Mechanical Design*, vol. 129, pp. 1064–1075, October 2007.

[42] A. Tayebi, S. McGilvray, A. Roberts, and M. Moallem, "Attitude estimation and stabilization of a rigid body using low-cost sensors," in *IEEE Conference on Decision and Control*, New Orleans, LA, December 2007, pp. 6424–6429.

[43] R. Tedrake, Z. Jackowski, R. Cory, J. W. Roberts, and W. Hoburg, "Learning to fly like a bird," Massachusetts Institute of Technology Computer Science and Artificial Intelligence Lab, Tech. Rep., 2006.

[44] J. Usherwood and C. Ellington, "The aerodynamics of revolving wings I - II," *Journal of Experimental Biology*, vol. 205, no. 11, pp. 1547–1576, 2002.

[45] F. van Breugel, Z. E. Teoh, and H. Lipson, *Flying Insects and Robots.* Edited by Dario Floreano, Jean-Christophe Zuffery, Mandyam V. Srinavasan, and Charlie Ellington, Springer, 2009, ch. 13: A Passively Stable Hovering Flapping Micro-Air Vehicle, pp. 171–184.

[46] WiiBrew, "Wiimote," 2011, (accessed on 20-March-2011). [Online]. Available: http://wiibrew.org/wiki/Wiimote

[47] R. J. Wood, "Design, fabrication, and analysis of a 3DOF, 3cm flapping-wing MAV," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, 2007, pp. 1576–1581.

[48] ——, "The first takeoff of a biologically inspired at-scale robotic insect," *IEEE Trans. Robotics*, vol. 24, no. 2, pp. 341–347, April 2008.

[49] R. J. Wood, S. Avadhanula, R. Sahai, E. Steltz, and R. S. Fearing, "Microrobot design using fiber reinforced composites," *J. Mech. Design*, vol. 130, no. 5, May 2008.

[50] R. J. Wood and R. S. Fearing, "Flight force measurements for a micromechanical flying insect," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, November 2001, pp. 355–362.

[51] R. Wood, "Liftoff of a 60mg flapping-wing MAV," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, November 2007, pp. 1889 –1894.

[52] J. Yan, R. Wood, S. Avadhanula, M. Sitti, and R. Fearing, "Towards flapping wing control for a micromechanical flying insect," in *IEEE International Conference on Robotics and Automation*, Seoul, Korea, 2001, pp. 253–262.

[53] J.-C. Zufferey, A. Beyeler, and D. Floreano, *Flying Insects and Robots.* Edited by Dario Floreano, Jean-Christophe Zuffery, Mandyam V. Srinavasan, and Charlie Ellington, Springer, 2009, ch. 6: Optic Flow to Steer and Avoid Collision in 3D, pp. 73–86.