Optimal Low-Power Control Strategies for an Autonomous Piezoelectric Micro-Robot

by

Biju Edamana

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
2012

Doctoral Committee:

Assistant Professor Kenn R. Oldham, Chair
Professor Dennis M. Sylvester
Associate Professor Bogdan Epureanu
Associate Professor Brent Gillespie

This thesis is dedicated to my mother Subhadra Edamana

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my adviser Prof. Kenn Oldham for his guidance, support and mentorship. He has been very patient and was available all the time for answering all my questions. His mentoring was very balanced. He gave me enough freedom to work on my own but made sure that I didn't lose track. I admire the fact that he is very clear in what he is expecting from each of his students and knowing those goals early and working on them at my own pace made my PhD years some of the best till now.

I am extremely thankful to my dissertation committee members, Prof. Dennis Sylvester, Prof. Bogdan Epureanu and Prof. Brent Gillespie for asking tough questions and providing guidance. Their suggestions were very crucial in presenting the ideas in the best possible way in this thesis.

I want to extend my gratitude to Prof. Dennis S. Bernstein who gave me the first chance to conduct research in summer 2007 at University of Michigan. I am also very thankful to Prof. Hosam Fathy who played a big role in introducing me to Prof. Oldham as well as recommending me.

I would like to extend my thanks to my labmates Bongsu Haan, Chris Rhee, Jeong-Hoon Ryou and Peter Park. Bongsu helped me with his design and fabrication of the switching circuit which was used in experiments discussed in the third chapter of this dissertation. Peter's strain gauge circuit design was used in macro-scale experiments. Also, I want to thank Chris for all those discussions we had during our coffee times from which I learned a lot.

I made a lot of good friends in last few years at Ann Arbor who made my time here really wonderful. I would like to thank Krithika, Kaustubh, Prasanna, Saurabh, Rich, Toni and Vishnu for all the fun times we had during my PhD years. Also, Kaustubh helped me to write the image processing MATLAB code used in the micro-scale experiments as well as I learnt a lot from our discussions.

At last, I want to thank my family who made the way for me to chase my dream starting from my childhood. My mother and sister Bindu had made lot of sacrifices in all these years and I don't think I can thank them enough. Also, I want to thank my brother Krishnan Potty for all his support.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

ABSTRACT

Optimal low-power control strategies for an autonomous piezoelectric micro-robot

by

Biju Edamana

Chair: Kenn Richard Oldham

This dissertation presents optimal control strategies for autonomous operation of a piezoelectric micro-robot with a very strict energy budget and payload constraints. Due to the capacitive nature of piezoelectric actuators, traditional analog amplifier based controllers and pulse width modulation based controllers use excessive amounts of energy compared to the actual physical work done by the actuators. Hence, an optimal on-off controller was developed which is the building block for the control strategies proposed in this dissertation. Strategies are presented for minimizing energy consumption during both the charge and discharge phases of actuator and when using feedback.

First, an optimal on-off controller and a partial charge recovery controller are presented. These controllers are developed for driving simple movements of actuators in the robot from a given initial position to a prescribed final position in a given time with minimum number charging of the actuator from the external power source. Charge recovery controller recovers drained energy at the actuator discharge phase as well. Conceptually these optimal movements of the actuators can be combined to form an efficient gate sequence for the robot. Compared to traditional controllers, the

proposed on-off controller consumes only a fraction of energy and the charge recovery controller uses even lesser; in certain cases charge recovery controller consumption was observed to be one third of on-off controller consumption for the same motion.

Second, a three step strategy is proposed for adding an energy consuming position sensor to the system in such a way that the sensor is turned on sparingly in order to achieve a certain level of error performance. Under certain assumptions, which may be verified relatively simply, the proposed strategy can be used to find optimal time points to turn on the sensor and update input sequences.

Finally, a hybrid energy system architecture containing a small solar cell and an off-the-shelf micro-battery which can be used on-board is proposed. A study on selecting optimization parameters based on gait sequences for choosing one of the power sources is presented. Together this is an attempt to form a benchmark for autonomous micro-robotics using piezoelectric actuators from an energy consumption point of view.

CHAPTER I

Introduction

Products of the study of autonomous mobile robotics have been very successful in terms of their mobility and autonomy in numerous applications in terrains ranging from the deep sea to outer space. There are successful macro-robots such as Mars rovers [1] which has the capability to travel on difficult terrains, maneuver around obstacles and carry its own energy source, sensing and control circuitry. At the terrestrial level, if the same ability can be achieved by millimeter-scale robots, they could be very useful in areas ranging from disaster response to inspection of otherwise inaccessible infrastructure. There are many challenges to realizing dramatically scaled-down versions of autonomous robots with the aforementioned capabilities. On the actuation side, recent developments in piezoelectric actuator design have shown load bearing and motion characteristics suitable for the high maneuverability and autonomy desired in autonomous micro-robots. However, in order to take full advantage of this actuation mechanism, development of an efficient control system and energy management system is necessary, which is the subject of this dissertation. Such a system would make use of a combination of existing and newly developed optimal control strategies.

1.1 Prior attempts in micro-robotics

There have been many attempts to create micro-robots at the millimeter scale with various degrees of success. One of the earliest attempts (in 1994) was by Yasuda et.al. [2]. It consisted of an untethered insect-like robot of size $1.5mm \times 0.7mm$ using resonant actuators. It operated in a vibration energy field generated by a piezoelectric vibrator. The robot consisted of resonant actuators responding to mutually exclusive resonance frequencies, operated based on the frequencies of the vibrator. Later, two thermally actuated robots were reported in 1999 by Kladitis et.al. [3] and Ebefors et.al. [4]. Kladitis's paper presented two robots, designed for micro-positioning in manufacturing processes, both mimicking six legged insects, with a size of $10mm \times 10mm \times 0.5mm$. Ebefors' robot's motion system was based on the ciliary motion principle and had a weight carrying capacity of 2500mg (more than 30 times its deadweight). Another interesting micro-robotic design was by Hollar et.al. [5] in 2003. This was a very small robot of size $8.5mm \times 4mm \times 0.5mm$ and powered by a solar cell. It was a two-legged walking robot which dragged its rear end. The first piezoelectric actuated micro-robot was reported in 2005 by Son et.al.[6]. It was a robot capable of bi-directional walking using a piezoelectric unimorph actuator. Its size and weight were $53mm \times 45mm \times 19mm$ and 23.25g respectively. A wheeled robot named Alice was also reported in 2005 [7] which is an all terrain version of the author's earlier robot presented in [8]. These robots are highly autonomous and capable of going over obstacles if necessary. Another very small design was introduced by Donald et.al in 2006 [9]. This type was of size $60\mu m \times 250\mu m \times 10\mu m$ and capable of rotation and translation in a structured environment utilizing scratch drive actuators. 2007 saw an interesting design of an autonomous jumping robot [10]. It was capable of making large motions but had yet to develop capabilities for multiple controlled jumps. In 2008 and 2009, there were two designs based on magnetic actuation by Vollmers et.al [11] and Pawashe et.al. [12]. Vollmers design was of size $300\mu m \times 300\mu m \times 70\mu m$

and was capable of moving forward, backward and turning with maximum speed of 12.5mm/s. The Pawashe robot was of size $250\mu m \times 130\mu m \times 100\mu m$ with translational speed of 8mm/s. The most recent walking micro-robot in the literature is a thermally actuated micro-robot by Erdem et. al. [13]. It was a 512-legged robot capable of carrying an energy source which can provide enough energy for 10 minutes of operation. Finally it should be mentioned that another on-going research by Sitti [14] utilizes piezoelectric actuation.

Out of all the aforementioned designs, only [5],[7] and [9] have reported autonomous operation capabilities. [5] has limited mobility since it had to drag its back and technically didn't carry its entire weight on its legs. Although [7] is highly autonomous, its operation is based on wheeled locomotion, which again limits its mobility compared to walking robots. The third autonomous micro-robot by [9] was reported to be operable only in a structured environment from which it received electrical power and control signals. Hence, despite their commendable autonomy none of these designs are in a good enough position to grow to the mobility level which their macro-level counter parts have achieved.

Although they were reported to be tethered, thermally actuated walking micro-robots presented in [4] and [13] had capabilities to achieve autonomy by carrying battery sized payloads. However, recent piezoelectric actuators can achieve longer stroke lengths and larger motion range compared to thermal actuators by using innovative actuator designs [15]. The micro-robot discussed in this thesis is an attempt to take advantage of these design capabilities of piezoelectric actuation in achieving better mobility compared to thermal actuators. The nominal robot to be considered in this dissertation is designed to have an area of $17mm \times 17mm$ and carries a payload of 20mg with a hexapod configuration. This is in the size range of the thermal actuators discussed but estimated to be capable of providing better mobility and eventually approaching biological ranges of motion through multi-degrees of freedom leg joints

and tens of degrees in the joint angles.

1.2 Motivation

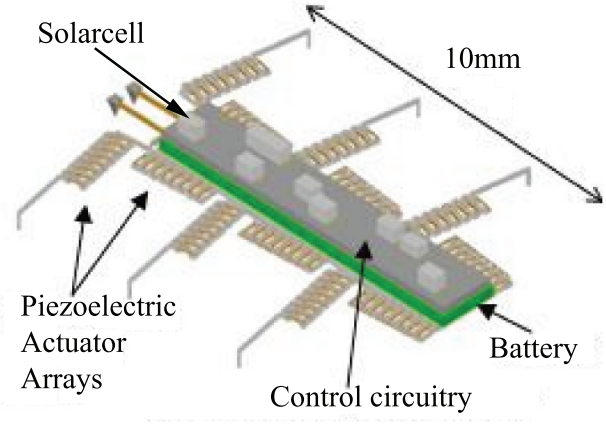

Figure 1.1: Concept-drawing of an autonomous micro-robot based on thin-film piezoelectric actuator joint arrays

For actuation in macro-scale robots there are many well developed technologies available, such as electric motors and pneumatic and hydraulic actuators. Miniaturizing these technologies has not been very successful to the millimeter scale. However, scaling-down electromechanical systems has enabled use of certain other phenomena for actuation, such as electrostatic attraction or the piezoelectric effect, which are not as prominent in the macro-scale. Out of these various actuation schemes, piezoelectric actuation has advantages that makes it suitable for use in an autonomous mobile micro-robot. Although thermal actuators have large force capacities, their power consumption is high [16, 17, 4]. Hence they require heavy power sources onboard that don't suit autonomous applications. Electrostatic actuators consume less energy, but the force generated per unit area is small [5]. Scratch drive electrostatic actuators can be operated only on certain substrates preventing complete autonomous operation [18],[9]. On the other hand, piezoelectric actuators are comparatively light weight, have high bandwidth, high force capacity and high force per unit area, and

typically consume less power [19][14][20]. They can produce a large force over a small stroke length or a small force over a large stroke length, depending on how they are integrated with other mechanisms in a micro-device. Importantly, they can be arranged in series to produce both large forces and large stroke lengths [15]. The type of actuators used in this study are capable of generating up to 3×10^{-9} N.m of work from a 500 μm $\times$ 100 μm area [15]. They can produce up to 3 degrees of rotation at 20V when coupled to a micro-robotic leg joint with sufficient weight-bearing capability. The actuators can also be combined in parallel or in series to produce large joint angles, up to the required rotation for a highly mobile micro-robot as shown conceptually in Fig. 1.1.

In order to take advantage of the potential of the MEMS actuator discussed above to propel micro-robots of millimeter scale, the actuator's associated control system also needs to be miniaturized. This raises general issues associated with miniaturization as well as issues unique to piezoelectric actuator control. One design objective of these micro-robots is to have a size of a few millimeters. At this scale, from the current actuation technology they can carry power sources that weigh a few milligrams and hence have a capacity of only a few milliwatts [21]. Therefore, there are strict energy constraints on size, weight and energy consumption of control circuitry that are unique to micro-scale systems. For example, adding an electronic component that consumes a few milliwatts to improve performance on a macro-scale robot is not as critical as compared to carefully choosing the basic set of components needed by the micro-robot.

There are challenges in micro-robotic control unique to piezoelectric actuator control as well. Piezoelectric actuators act as capacitive loads on driving electric circuitry. Hence, when voltage is applied across the actuator, depending on its electromechanical coupling coefficient, a fraction of the energy consumed is used to do mechanical work and the remainder is stored as mechanical and electrical energy. When the volt-

age is removed, the actuator returns to its original position and the stored energy is drained off. In most cases, traditional analog amplifier-based controllers are designed to operate on resistive loads. When these controllers are used with piezoelectric actuators, they can consume as much as or more than 99% of energy supplied [22]. As an alternative to these controllers, Main et al. [22] proposed PWM (Pulse Width Modulation) based switching controllers for piezoelectric actuators. PWM controllers operate at switching frequencies well above the natural frequency of the plant upon which its operating. On the other hand energy consumption is proportional to the switching frequency in piezoelectric actuators. Hence, reducing switching frequency from the PWM-level can be beneficial in saving energy, though this has its own drawbacks, as will be discussed later. Although the control strategies developed in this thesis are applied to piezoelectric actuators specifically, they can be adapted, in general, to use on other capacitive actuation schemes as well.

1.3 Research objectives

1.3.1 On-off control

In the second chapter of this dissertation an optimal on-off controller is proposed that minimizes the number of times an actuator is turned on, subject to certain performance constraints. Although on-off controllers reduce energy consumption drastically compared to analog or PWM inputs, there are drawbacks associated with them. For a general discrete linear system with an on-off controller, the reachable state space is limited over a finite time horizon. From a given set of initial states, the system can reach only a certain number of discrete points in the state space in a given time. However, by solving an integer programming problem it is possible to find input sequences that enable the system to reach most of the state space (or at least that region of the space that can be reachable using the given input level) with an acceptable level of

tolerance in a reasonable time horizon.

A switching control system is a simple case of the larger field of hybrid systems. Morse [23] gives an overview of hybrid systems consisting of switching between systems with different dynamics. These switchings are different from the well studied bang-bang optimal control solution where the optimal solution for certain control problems with bounded inputs is found to be switching between a maximum and minimum. Although in the optimal solution the input is switching between a minimum and a maximum, the switching is not a constraint. Previous optimization techniques designed specifically for on-off control schemes have minimized the time to reach a desired set of system states [24], or the amount of time spent with an 'on' input to reach a desired set of states at a specified time [25] [26]. However, these approaches do not account for switching costs, such as the energy usage to charge a capacitive actuator. The work done in this dissertation, is more closely related to [27], which did consider a general quadratic objective function in optimal control design. The switching costs discussed in this dissertation can be converted to a quadratic objective function as shown in the second chapter. In [27] Bemporad et.al. introduces a framework for modelling a broad class of systems described by logic rules, constraints and dynamics that are denoted as Mixed Logic Dynamic (MLD) systems. The switching control system discussed in this dissertation can be represented as an MLD system. Although the controllers developed in the second and third chapter are mostly implemented in open loop, feedback can be introduced as a model predictive controller.

From an application standpoint, the objective of the optimal on-off controllers developed in the second chapter is to rotate a micro-robotic leg connected to an array of piezoelectric actuators from a given initial angle to a desired final angle with a desired final angular velocity in a given time with a minimum of actuator charging. Since the on-off controller is incapable of driving the leg to any prescribed leg-angle

and angular velocity in a given time, there is a tolerance level associated with the final angle and angular velocities. This is achieved by solving an integer programming problem with a quadratic objective function and linear state constraints.

To accomplish this, a second order linear system is identified for the leg with its angle of rotation and angular velocity as states. Dynamics of the system and final states with the acceptable tolerance levels are used as constraints in the optimization problem. Two methods are used to solve for an optimal switching sequence. The first one is a penalty function based method, combining the objective function and constraints into a relaxed unconstrained continuous optimization problem. The second method uses a standard integer programming solver for solving the optimization problem with a quadratic objective function and linear constraints. The controller design can be extended to a closed-loop controller by model predictive control methods, which is expanded upon in chapter four with regards to sensor scheduling. It should be noted that the controller developed in the second chapter has most of the drawbacks of other open-loop controllers. First, it needs the identified model to match the actual system very closely. Also, the relatively low-switching frequency results in oscillatory output motions and this could excite high-frequency dynamics, although this was not a large effect in the prototype actuator used in the experimental setup. In practice, higher order dynamics could be incorporated into the optimization by identifying a higher order system; although this could not necessarily help in reducing oscillations over the entire course of motions, the effect of the oscillations on the final states can be reduced. However, since the objective is to reduce energy consumption drastically and these limitations of on-off control have minor effects on prototype devices, the controller is judged to meet initial controller design requirements.

1.3.2 Charge recovery controller

As mentioned earlier, a piezoelectric actuator acts primarily as a capacitor when connected to an electric source. Hence, under normal operation when the actuator is turned off the energy stored in the actuator is drained to the ground. Earlier researchers had identified the potential to recover almost all of this energy using charge recovery circuits [22, 28]. Campolo et.al. [28] introduced a circuit that consists of an inductor, storage capacitor and a couple of diodes which could recover as much as 92% charge (82% energy) with lossless components. This was achieved using resonance in the RLC circuit and suitably choosing the circuit components.

The inductor in the circuit is a critical component in micro-robot applications because of its weight. The energy that can be saved is a function of the inductance in the circuit and hence a function of the weight of the inductor used, given existing inductor fabrication technology. To recover nearly all the energy, as in [28], a heavy inductor is required, which is not affordable in micro-robotics. Hence, a novel partial charge recovery strategy is introduced using a lighter inductor. This has the additional benefit of having intermediate voltage levels available as inputs between the minimum and maximum voltage levels of on-off control. These additional voltage levels enable a controller to reach more points in the state space of the system and hence to operate at a tighter tolerance level or at comparable tolerance levels while charging actuators less often. In order to convert the optimization problem of minimizing the number of chargings of the actuator using an external energy source to an integer programming problem capable of handling charge recovery, additional constraints are required to model the dynamics of the charge recovery circuit. These additional constraints increase the computational complexity, but reduce actuator power consumption. This is discussed in detail in chapter three.

1.3.3 Sensor scheduling

The objective of the first two controllers introduced was to consume the minimum energy possible to perform certain micro-actuator motions, and thus to set a benchmark for the feasibility of using piezoelectric actuation under control in autonomous micro-robotics. However, some level of feedback is necessary to reduce the effect of disturbances or discrepancies in the model used for optimization. Unfortunately, with existing sensing technology, position sensors are highly energy consuming themselves and using them at the same sampling frequency as the on-off controllers may be prohibitive for autonomous operation of micro-robots. Hence, an optimal intermittent feedback controller was developed that updates the optimal on-off sequences at certain time instants based on measurements taken at selected measurement times. The measurement times and input update times are decided based on intermittent Kalman filter equations that minimize the uncertainty of the states at the desired time.

Several previous studies on sensor scheduling have some relevance to this work. Recently, Mo and Sinopoli [29] proposed the existence of a critical measurement rate for an unstable system which bounds the error on state estimates. They have developed these results for an unreliable wireless sensor network where the system was modeled as an unstable system. Similar studies can be found in [30],[31], [32]. In [33], Li et. al. discussed a convex optimization strategy that minimizes the maximum error of certain states by having a sensor visit those states frequently. Another group of researchers has been investigating efficient use of multiple sensors by turning on a smaller set of sensors at a time [34],[35], including for model-predictive applications [36]. This study differs from previous works by restricting controlled system inputs to on-off switching. As the final set of states becomes a function of binary variables, the problem can become a complex one to optimize even with certain simplifying assumptions: first, there is only a single sensor which is turned on sparingly (a small

finite number of times over the duration of motion) and second, the objective function is expressed only in terms of the final states. Following a sensor measurement, adjustment of future on-off inputs to the system is similar to that of certain hybrid or model-predictive controllers. For example, for the controller in this section, a strategy derived from the receding horizon approach for switching controller optimization of [27] is adopted. Similar model-predictive control algorithms with final costs have been studied in [37]. However, these previous works generally assume fixed sampling rates with frequent measurements, concentrating their studies on the optimal controller design, while in this study controller design is combined with the sensor scheduling problem, for the on-off input, single-sensor conditions noted above.

1.3.4 Hybrid energy system for switching control of micro-robotic actuation

Some of the earlier attempts in designing power systems for piezoelectric micro-robots were presented in [38],[19]. Karpelson et.al. in [38] proposed power electronic circuits suitable for driving piezoelectric actuators for milligram-scale micro-robots, while Steltz et.al. considered heavier robots weighing a few grams. They agree upon the fact that only battery technology is developed enough to act as primary energy source in micro-robotics. Also, conversion of battery outputs to very high voltages, using various boost converter circuits that have been useful at macro-scale, is neither easily miniaturizable nor efficient at low output power levels. These factors constrain the availability of components that can be used in the micro-scale circuit design.

With the switching control strategy proposed in this dissertation, the range of motion of piezoelectric actuators, i.e. voltage-mode actuators [20], depends on the maximum switching voltage of the controller. Hence in order to fully utilize the range of the actuator, a 20-30V voltage supply is required by the switching controller. Meanwhile, the instantaneous power requirement while charging the actuator is more

than that which micro-batteries can provide. Hence, they can't be connected directly to batteries through the switching controllers.

The fifth chapter of this dissertation will propose a power system for micro-robotics in which the power requirement is met using a storage capacitor supplied alternatively by batteries or solar cells as environmental conditions dictate. Storage capacitors are of sufficient size as to provide the current required to charge the actuator instantaneously. As an example, from analysis and simulation it is observed that a storage capacitor of capacitance $1\mu F$ can charge the actuator of capacitance $1nF$ approximately 20 times whiles its voltage changes from 19V to 18V. The second piece of a battery-operated power system is a voltage converter. This has been implemented in a simulated power system model by carefully designing a standard boost converter circuit with components optimized for the given capacitive load charged to 20V-30V range.

Inspired by the idea of adding an energy harvesting system from [5], a solar cell is then added that can potentially increase the interval between the times when the battery needs recharging.There are several energy harvesting technologies available at the MEMS scale, such as photovoltaic cells and piezoelectric, electrostatic, or electromagnetic devices harvesting kinetic energy from vibrations. Photovoltaic cells maintain a steady flow of power under constant lighting conditions. Vibration harvesting is too low-power for active walking consideration but it is an area of potential future work. Other harvesting technologies produce oscillating voltages and currents requiring more sophisticated power conditioning units. Higher power density per unit area and simpler power conditioning units make the choice of using solar cells straightforward. Hence, an array of solar cells is considered with parameters taken from a sample commercial miniature solar cell, the SANYO AM-1815 having an area of $17mm \times 17mm$ and consisting of 32 cells. This arrangement can produce a voltage of 19V to 20V under normal indoor lighting conditions and can be used to charge the

storage capacitors directly. From simulations it was observed that the solar cell can power the robot for very low speeds, without use of battery power.

1.4 Contributions

Although familiar to control researchers, the idea of optimal switching control is novel to piezoelectric actuation. Reduction in switching frequencies from tens of kilohertz to less than 1 kHz sets a benchmark in energy consumption for small piezoelectric actuators. Although the scheme has certain drawbacks in performance, it creates a way to make autonomous motion control in micro-robotics feasible. Secondly, the idea of partial charge recovery is introduced. Compared to previous charge recovery techniques, this is more suitable to micro-robotics because of the weight advantage of components as well as its complementary nature to on-off control in providing additional intermediate voltage levels. The third contribution is a three step sensor scheduling strategy to make use of an energy consuming sensor sparingly for an on-off controller. Under certain assumptions, this method provides a way to determine when to take measurements and when to update the input sequences (solving N order less number of integer programming problems). Finally, a hybrid energy system is proposed within the stringent constraints of size and weight of micro-robotics.

The remainder of the dissertation is organized based on the above topics. In the second chapter, the open-loop on-off controller and methods of its optimization is described. In the third chapter, the on-off controller is extended to a partial charge recovery controller and also the tradeoffs of having heavy charge recovery circuitry are discussed. In the fourth chapter the idea of optimal intermittent feedback with an on-off controller is discussed. The fifth chapter describes in detail a proposed hybrid energy system and the sixth chapter concludes the thesis with a summary and discussion of the contributions.

CHAPTER II

Optimal switching controller

As mentioned in the introduction, the simplest alternative strategy to counter higher power consumption of traditional analog amplifiers is the use of switching controllers, such as shown in Fig. 2.1. For switching control of capacitive loads, power consumption is proportional to switching frequency. Hence to minimize power consumption, it is necessary to minimize the number of times the actuators are turned on while meeting performance requirements. In this chapter, we discuss an integer programming based optimization approach to find an optimal switching sequence which minimizes the number of switchings to achieve certain servo constraints of a micro-robotic leg joint.

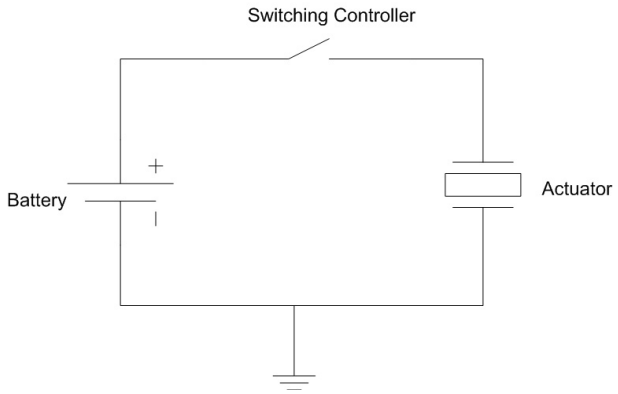

Figure 2.1: The basic switching controller for a piezoelectric actuator

This chapter is organized as follows. In the section 2.1 the modeling of the actua-

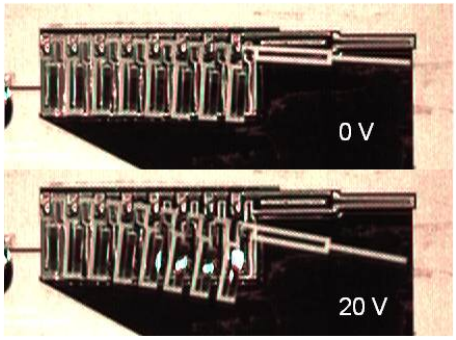

Figure 2.2: Sample image of leg joint at 0 and 20 V

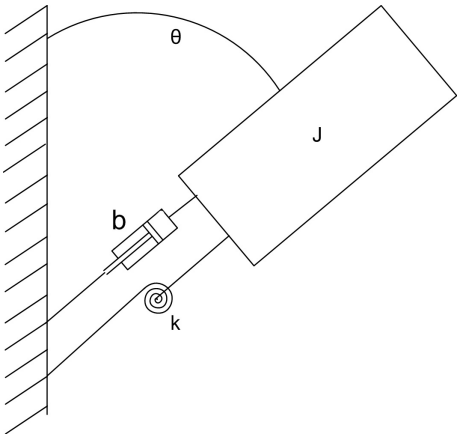

Figure 2.3: The dynamic model of the leg joint

tor is described. In section 2.2 structure of the optimization problem is explained. In section 2.3 two different approaches are given to solve the integer programming problem. In section 2.4 the optimization method is verified, comparing its results with the results obtained from complete enumeration as well as compared with a traditional LQR controller. In section. 2.5 a method to improve robustness is proposed as well as simulational results showing the effect of feedback. Experimental and simulational results are compared in the section 2.6. Section 4.6 discusses the advantages and limitations of the controller strategy and concludes the chapter.

2.1 System Description

A sample image of a prototype micro-robot leg joint, fabricated and tested at the Army Research Laboratory and Michigan Lurie Nanofabrication Facility is given in Fig 2.2 [39]. The dynamics of the joint can be lumped into a mass-spring-damper system as shown in Fig 2.3. Often, the dynamics of individual joints can be represented by a second order differential equation of the form,

$$J\ddot{\theta} + b\dot{\theta} + k\theta = Gu \tag{2.1}$$

where J, b,and k denotes the inertia, damping and stiffness respectively and G represents the actuator gain, and u the input voltage to the actuator. The values of these parameters used are given in table 2.1. This can be converted to state space with the states the angle of rotation (θ) and the angular velocity ($\dot{\theta}$) of the actuator,

$$\begin{aligned} \dot{x} &= A_c x + B_c u \\ y &= C_c x \end{aligned} \tag{2.2}$$

where

$$x = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}$$

$$A_c = \begin{bmatrix} 0 & 1 \\ -k/J & -b/J \end{bmatrix}$$

$$B_c = \begin{bmatrix} 0 \\ G/J \end{bmatrix}$$

and

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

In general these continuous equations can be replaced by a set of linear equations by discretizing them at sampling times T_s,

$$x\left((k+1)T_s\right) = A_d x(kT_s) + B_d U_{max} u(k) \tag{2.3}$$

$$y(kT_s) = C_d x(kT_s) \tag{2.4}$$

where, U_{max} is the battery voltage. There are two important constraints on the system when using pure switching controller. The first is that the inputs $u(k), k = 1, 2, \ldots, n$ can take only two values, namely $\{0, 1\}$. Second, the transitions between input values can take place only at the sampling instants, meaning u_k can change its value only at these times. These constraints limit the reachable subspace of the system in a given time duration.

2.2 Problem Description

2.2.1 Constraints

There are two types of constraints in the optimization problem, namely final state constraints and dynamics of the system. The final state constraints are derived from the quasi-static walking gate of the robot. In the robotic-leg level the waking gait constraints convert to rotation of the actuators to a prescribed angle and velocity (typically zero) in a given time starting from a stationary position. Since both the time of input transition and the magnitude of input are constrained, the states can not be driven to any arbitrary point in the state space in a given time. Hence, the constraints are set to drive the final states to a satisfactory ϵ neighborhood of the desired final states in given time. The final state constraints can be written

mathematically as $x_d - \epsilon \leq x(t_f) \leq x_d + \epsilon$. This forms $2p$ linear inequality constraints, where p is the order of the system and equal to 2 in this case. To follow dynamics of the system for a given time $T_f = n * T_s$, the states of the system should satisfy the $p \times n$ equations of the form (4.1). So overall there are $(p+2) \times n$ linear constraints in this optimization problem, as well as the binary constraints on the input variables mentioned earlier.

2.2.2 Objective

The objective function is to minimize energy consumption while satisfying the constraints. Energy consumption of a generic piezoelectric actuator consists of two parts, J_C and J_R, corresponding to capacitive and resistive energy losses in the system, respectively.

When a piezoelectric actuator with zero initial voltage is connected to a higher voltage (In this problem, this happens when $u(k)$ changes from 0 to 1), the voltage across the actuator increases and current decreases exponentially as in a typical RC circuit. Irrespective of the value of the resistance in the circuit, the energy consumed from the energy source during this process depends only on the capacitance of the actuator and is equal to CU_{max}^2. This energy loss is termed the capacitative loss in the system J_C and in general can be replaced by any arbitrary "cost-to-switch". Mathematically, this is defined for a discrete time process by

$$J_C = \sum_{k=1}^{n} CU_{max}^2((u(k) - u(k-1))^2 + u(0)^2) \tag{2.5}$$

where C is the capacitance of the piezoelectric actuator and U_{max} the on voltage applied to the actuators.

The second part of the objective function, J_R, includes energy lost to resistive dissipation due to leakage current in the on-off drive circuit or through the piezoelectric

actuator while in the charged state, and is given by

$$J_R = \sum_{k=0}^{n} \frac{U_{max}^2}{R} T_s u(k) \tag{2.6}$$

where R is the resistance of the system. In other words this is the energy required for keeping $u(k)$ at 1. Again, here the quantity U_{max}^2/R can be generalized in this case to an arbitrary "cost-to-hold".

Hence, the optimization problem is a quadratic binary programming problem where the objective is to minimize the total energy cost, $J = J_C + J_R$ subjected to the state dynamics ($m \times n$ constraints),the binary inputs (n constraints), and the final state constraints ($2m$ constraints) given.

2.3 Solution Methods

Presence of binary variables makes the optimization problem NP-hard (non-deterministic polynomial-time hard) and conventional optimization methods fail to solve it. For n binary input variables 2^n combinations are possible, hence the numerical complexity increases exponentially with the number of variables. There have been attempts to use dynamic programming to solve such problems [40], but when the number of binary variables are large, these methods are numerically prohibitive. There are two types of algorithms used in general to solve these problems. Firstly, relaxation based algorithms in which the integer constraints are dropped, but implemented implicitly. The success of this method is problem dependent. In second method, the entire solution space is enumerated by dividing the problem into smaller problems. There are prepackaged solvers available for this method.

2.3.1 Penalty function approach and non-linear optimization

In this approach the optimization problem is converted to an unconstrained continuous optimization problem using penalty functions as proposed by [41].

2.3.1.1 Step-I

The first step is to convert the dynamics constraints and final state constraints to a penalty function form and add to the objective function. Since only the state at the desired final states are constrained, the dynamics equations can be combined with the final constraints to form m constraint equations.

$$x(T_f) = x(n * T_s) = A_d^n x(0) + \sum_{k=0}^{n-1} A_d^{(n-k-1)} B_d U_{max} u(k) \tag{2.7}$$

Assuming initial states to be zero the constraints can be re-written as,

$$\sum_{k=0}^{n-1} A_d^{(n-k-1)} B_d U_{max} u(k) \in x_d \pm \epsilon \tag{2.8}$$

In order to convert the problem to an unconstraint optimization problem the objective function is modified as,

$$J_{constraint} = \min[(\sum_{k=0}^{n-1} A_d^{(n-k-1)} B_d U_{max} u(k) - x_d)^T \sum_{k=0}^{n-1} A_d^{(n-k-1)} B_d U_{max} u(k) - x_d] \tag{2.9}$$

Now by using a suitable weighing factor w_1, the objective functions can be combined to $J + w_1 J_{constraint}$.

2.3.1.2 Step-II

A coordinate transformation is done on binary variables $u(k)$ in this step. The new variables are $v(k) = 2u(k) - 1$. This implies $v(k) = -1$ when $u(k) = 0$ and $v(k) = 1$ when $u(k) = 1$. The constraint $v(k)^2 = 1$ will now make sure that the variables are

binary. Again this constraint is added to the objective function as a penalty function. The new objective function for the unconstraint continuous optimization will be,

$$J_{new} = \min[J + w_1 J_{constraint} + w_{2k}(v(k)^2 - 1)] \tag{2.10}$$

The objective function is a convex function (convexity is determined by initializing the optimization problem at many random combinations of variables) and a gradient based optimization solver such as fmincon in MATLAB optimization toolbox can be used to minimize the objective function.

2.3.2 Branch and Bound Method

Branch and Bound methods are alternative standard methods for solving integer programming problems. The drawback of the penalty function-based method is that its success depends on the weighing factors used in adding penalty functions to the objective function, which are not required in branch and bound method. In a direct numerical comparison the penalty function method is efficient, however the availability of prepackaged solvers makes the branch and bound method more convenient in many situations.

In a branch and bound algorithm complete enumeration of the entire binary tree is done in a systematic manner. The entire solution space is explored, divided into feasible sub-domains, and valid upper and lower bounds are derived for each sub-domains. The infeasible and non-optimal regions are eliminated (pruned) during the process, and this is done using the minimum numerical operations possible.

There are many commercial solvers available to solve binary programming problems using the branch and bound method. For this problem, a combination of AMPL (A Modeling Language for Mathematical Programming) [42] and CPLEX [43] was used. AMPL was used to model the optimization problem. In AMPL both the system dynamics constraints and final state constraints can be given directly and

variables can be fixed as binary. After modeling in AMPL, the CPLEX solver was used to solve it using branch and bound method.

2.4 Verification of optimization and comparison with LQ optimal control

To ensure that the optimization method is working properly, the results of the both optimization methods were compared with that of brute force optimization for a short time period. In the brute force method all possible combinations of $u(i)'s$ were checked for feasibility and those with lowest cost, measured by the respective cost functions, were selected. This approach is of course numerically inefficient, as one must check the cost for 2^n combinations of input, making it impractical for a larger time period. The results were matched for both the penalty function method and branch and bound method with their corresponding objective functions.

Similar to Linear Quadratic (LQ) optimal control methods, the penalty function based integer programming method can be used to design controllers that provide a trade-off between the state cost (accuracy) and controller cost (energy consumption). To implement a comparable Linear Quadratic Regulator (LQR), the system is first converted to discrete domain. Then error dynamics of the system is used for the LQR tracking and an integrator is added which has the following advantages,

i)It makes the steady state error go to zero.

ii)The integrator state (derivative of input) can be used to calculate the capacitative loss.

The cost function for the competitive LQR controller has three terms, the first one penalizes the deviation of the final state from the desired state, the second one is the resistive cost and the final one is the capacitative cost. The cost function is given

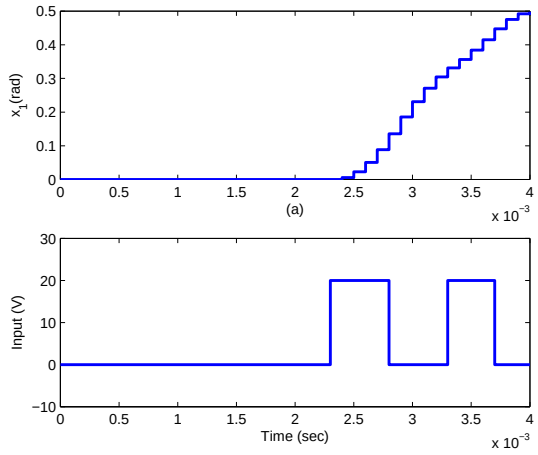

Figure 2.4: Sample simulated system output using optimal On-Off controller with loose positioning constraint, showing (a) output angle and (b) system input u

below,

$$\int_0^{t_f} E[(\theta - \theta_{ref})^2 + r(u^2/R + 1/2Cu\dot{u})]dt \tag{2.11}$$

It can be descritized with a sampling time Ts as given below:

$$\sum_{i=0}^{n} E[((\theta(i) - \theta_{ref})^2 + r(u(i)^2/R))Ts + r(1/2Cu(i)\Delta u(i))] \tag{2.12}$$

Where,

$$\Delta u(i) = u(i+1) - u(i), i = 0 \ldots n-1,$$

and $u(i)$ is the input applied at the i^{th} time step and 'r' represents the relative weighting on the control cost over the state cost.

A set of sample system responses using the minimal energy open-loop on-off optimal controller is given in Fig. 2.4 and Fig. 2.5. In these examples, a single leg link is driven to a desired final angle. When only a single leg link is to be controlled,

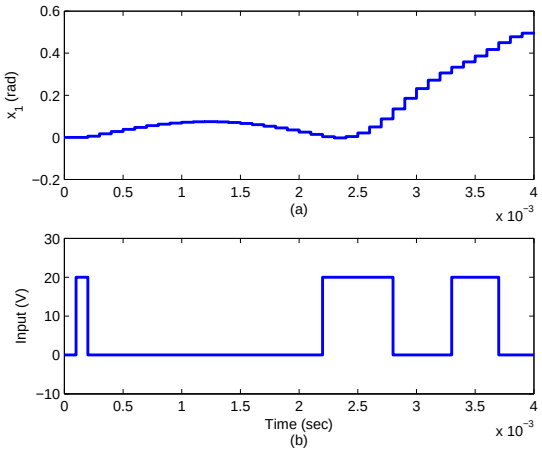

Figure 2.5: Sample system output using optimal On-Off controller with strict positioning constraint, showing (a) output angle and (b) system input u

Table 2.1: The nominal values for the parameters used for the design of optimal sequences for the systems

Parameters	Simulation system	Macro system	Micro system
R (Ω)	$3 * 10^9$	$3 * 10^9$	$3 * 10^9$
C (F)	$1 * 10^{-9}$	$1 * 10^{-9}$	$1 * 10^{-9}$
$U_{max}(V)$	20	20	20
J ($kg.m2$)	$1.4 * 10^{-12}$	1	1
b ($N.m.s/rad$)	$3.4 * 10-11$	14	193.2283
k ($N.m/rad$)	$3.2 * 10-6$	40350	$3.1200 * 10^6$
G ($N.m/V$)	$8 * 10^{-8}$	64151	$1.9656 * 10^4$
Sampling time (T_s)(sec)	0.0001	0.001	0.0001

the control input can be quite simple, as in the example shown. In Fig. 2.4 the input switches twice and when the constraints on states are stringent the controller needed three switches as shown in Fig. 2.5 which corresponds to a cheaper controller or higher state cost as in linear-quadratic controller terminology. To explore controller behavior, this on-off controller was compared to the LQR controller for the system, had feedback and analog rather than on-off inputs been available.Two LQR responses corresponding to cheap and expensive controllers are given in Fig. 2.6 and 2.7 respectively. Both the on-off and LQR controllers produce qualitatively similar trajectories from the initial to the final value. In addition, the capacitative portion of

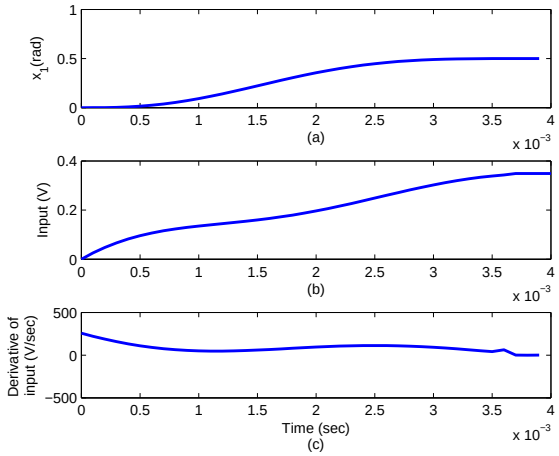

Figure 2.6: Simulated LQR response with a cheap controller, analogous to strict positioning constraint, showing (a) output angle (b) system input u and (c) derivative of input

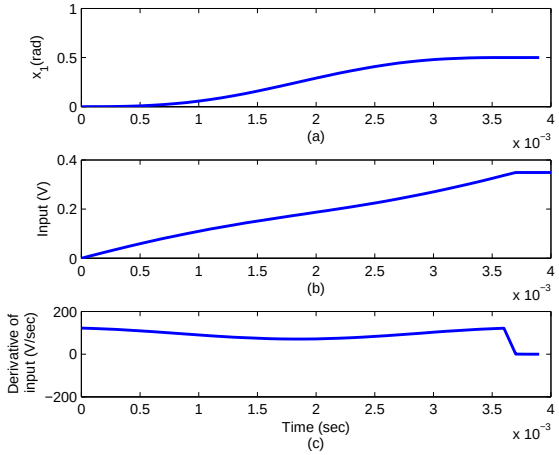

Figure 2.7: Simulated LQR response with an expensive controller, analogous to a loose positioning constraint, showing (a) output angle (b) system input u and (c)derivative of input

the cost functions from the respective controllers is found to be less for the optimal on-off controller, thanks to the minimal number of transitions that it dictates.

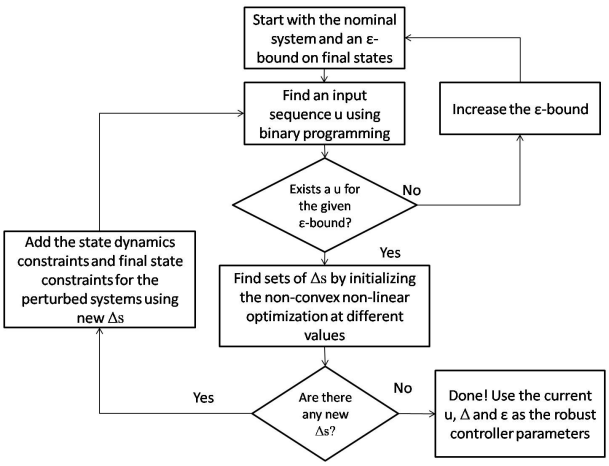

Figure 2.8: The robust sequence algorithm

2.5 Simulational analysis

2.5.1 Modification for robustness

Uncertainty in the inertia, damping and stiffness estimates may also affect the performance of a dynamic system. It is useful to minimize the error for the worst case system with a bounded uncertainty, Δ_{bound}. This is a minimax problem as given,

$$\min_{u}\{\max_{\Delta} \|x(n) - x_{nominal}(n)\|\},\ \ \|\Delta\| \leq \Delta_{bound} \tag{2.13}$$

Since the direct solution of (2.13) is numerically infeasible because of the binary inputs and min-max objective function, the algorithm given in Fig. 2.8 and described below was developed to find an input sequence that gives a satisfactory performance. Two optimization techniques are employed here. Continuous non-linear optimization for determining the worst case systems (Δ values) and binary programming (this is explained in section 2.3) for finding the best robust input sequence. In this particular case, as applied to a micro-robotic leg joint problem three parameter uncertainties are considered, namely Δ_m, Δ_b, and Δ_k corresponding to uncertainties in inertia, damping, and stiffness. It is assumed that these uncertainties are bounded on either sides. But, the maximization of error with respect to the Δ's is not a convex opti-

mization. Hence, the optimization is initialized at several random values with in the given Δ range and the corresponding worst case system for each was found. For the procedure, let (A_1, B_1),(A_2, B_2),... (A_m, B_m) be the dynamics of each of the worst case systems obtained by the maximization. Then the final state constraints for all m systems are added to the binary programming to find the robust u as:

$$
\begin{aligned}
x_d - \epsilon \leq \sum_{i=0}^{n-1} A_1^{n-i-1} B_1 U_{max} u(i) \leq x_d + \epsilon \\
x_d - \epsilon \leq \sum_{i=0}^{n-1} A_2^{n-i-1} B_2 U_{max} u(i) \leq x_d + \epsilon \\
\vdots \\
x_d - \epsilon \leq \sum_{i=0}^{n-1} A_m^{n-i-1} B_m U_{max} u(i) \leq x_d + \epsilon
\end{aligned}
\tag{2.14}
$$

These constraint equations help the designer to identify a specific input sequence u such that if it exists, will keep the final states of all the above systems within some ϵ neighborhood of the desired final state.

2.5.2 Robustness analysis results

From the simulations, it was found that if the parameter variation (in all parameters) is within 10% of the nominal value for the sample system, then it was possible to find an on-off sequence which keeps the final state within 10% of the desired final state. Examples of perturbed systems are shown in Fig. 2.9. When the parameter variations are kept within $\pm$10% of nominal values, the final states of the randomly perturbed system are within $\pm$10% of the nominal value, 0.3$\pm$0.03, and this was consistent over the situations we examined.

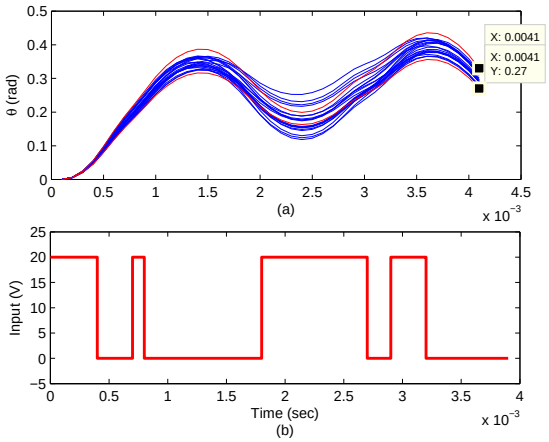

Figure 2.9: (a) Response of the perturbed systems when a robust sequence is applied; the red lines show the worst case systems (b) the corresponding robust on-off sequence

2.5.3 Closing the loop using a model predictive controller

Although it is not practical to implement on a real system, a simulation study was conducted to see the improvements feedback would made on the system performance. The optimal switching sequence was found in a receding horizon manner. At the beginning the optimal sequence was found for the entire range, but only the first input value was applied. After collecting the sensor information from the first time instant, the optimization is again done for the new horizon to obtain a modified sequence. This process continued for the entire duration applying only the first input value from the each optimization. A constant disturbance was assumed to be entering the system throughout and results from two cases are shown here. In the first case (Fig. 2.10) it was assumed that the disturbance in the input voltage is 2V and in the second case it was assumed to be -4V (Fig. 2.11). In both cases the objective were to reach the neighborhood of 0.1rad with zero final velocity.

In the first case the improvement is clearly visible. The final state without feedback settles at 0.117 radians where as with the feedback its around 0.99 radians which is close to the desired value. In the second case under larger disturbance the open

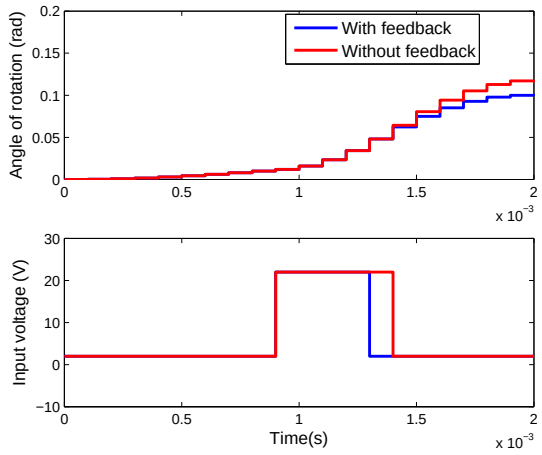

Figure 2.10: Comparison between open loop and closed loop optimal controllers with a 2V constant disturbance

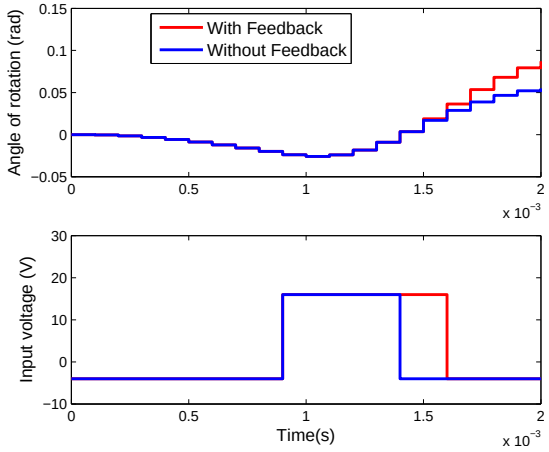

Figure 2.11: Comparison between open loop and closed loop optimal controllers with -4V constant disturbance

loop response settles well below at 0.55 radians and the closed loop at 0.87 radians. The main inference from this study was that, closing the loop doesn't guarantee satisfactory results always because of the constraints on the input, but in general it improves the performance compared to the open loop controller. More sophisticated feedback strategies are discussed later in chapter 4.

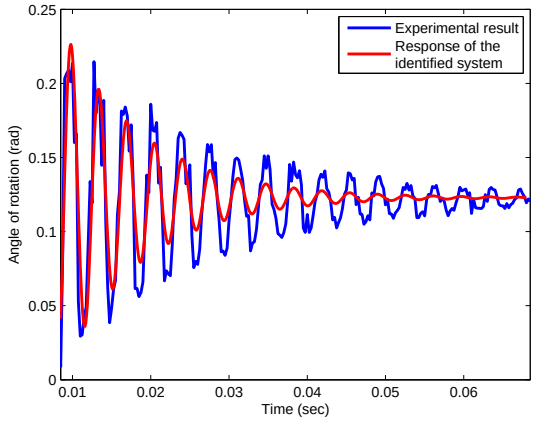

Figure 2.12: Step response of the MEMS actuator which was used for system identification

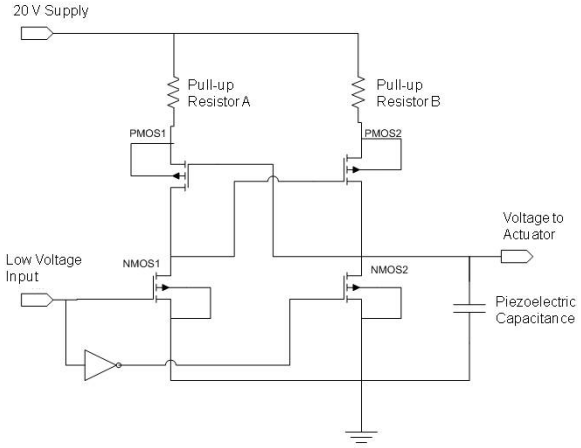

Figure 2.13: Switching drive circuit with pull-up resistors to limit leakage current

2.6 Experimental results

2.6.1 Experimental system description

The implementation of on-off switching on micro-systems requires a fast switching circuits. The on-off switching circuit designed for our application consists of a CMOS inverter with a level shifter. CMOS inverters are a commonly used switching circuit configuration in the integrated circuit design area for reducing power consumption because in the ideal case there is no static current and power is consumed only at

the 'on' or 'off' transition time. However, a CMOS inverter alone cannot be used directly for driving piezoelectric actuators. While a piezoelectric actuator should be driven at 20 to 30 V or more, most IC circuits, as are typically used to implement a control law, operate at 5 V, 3.3 V or less. Therefore, a level shifter based on the CMOS Inverter was designed to interface between a high voltage and a conventional IC process [44]. In order to reduce the power consumption of the circuit, two resistors are added to the basic level shifter. The conceptual circuit is shown in Fig. 2.13.The novel addition of these resistors to the inverter helps reduce peak leakage current during the switching transitions. While this loss is typically of little consequence when using larger actuators, it can be a substantial portion of energy consumption when working with micro-scale piezoelectric actuators having comparatively small capacitance. However, the additional cost can be accommodated in the optimization problem by adding it to cost-to-switch as mentioned in section 2.2.

Before conducting the experiments on micro-actuator the control scheme was tested on a macro scale piezoelectric actuator. It is a 40 mm long, 10 mm wide Ceratec, Inc. bimorph actuator with a strain gauge attached to it for measuring the deflection in terms of the voltage through its sensing circuitry. The on-off switching sequence was loaded into a TMS320F28335 microprocessor which was interfaced to the bimorph actuator through a fast switching circuit. The output voltage was measured using a Tektronix TDS2024B oscilloscope at 2Gsamples/s and the data was captured using National Instruments SignalExpress Tektronix Edition Software. From the step response of the system, the following second order system given was identified between input voltage and strain gauge voltage which is treated as the system output:

$$\frac{y(s)}{u(s)} = \frac{64151}{s^2 + 14s + 40350} \tag{2.15}$$

The micro-robotic leg joint used in the experiment is shown in Fig. 2.2. It was also operated using the microprocessor and the switching circuit. The motion of the

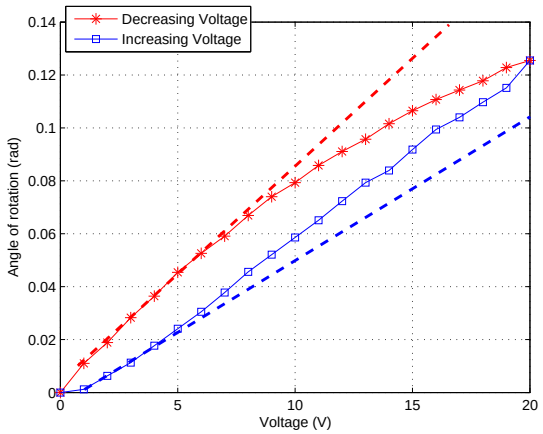

Figure 2.14: Hysteresis curve for the MEMS actuator with the dotted lines showing the variation in slope

leg was captured using a high speed camera at 4000 frames per second and the angle of rotation was measured using the MATLAB Image Processing Toolbox. Using the step response given in Fig. 2.12, a second order system was identified between the input voltage and the angle of rotation (radians):

$$\frac{y(s)}{u(s)} = \frac{1.9656 * 10^4}{s^2 + 2 * 0.0547 * 1766.4s + 1766.4^2} \tag{2.16}$$

To measure hysteresis, a static input voltage was varied from 0 V to 20 V and back, giving the hysteresis plot in Fig. 2.14. This hysteresis effect is included as an uncertainty in stiffness for designing a robust sequence, which is used in the experimental result discussed in the next section.

2.6.2 Comparison of experimental and simulation results

Fig. 2.15 shows a comparison of simulated and experimental responses obtained from the macro system when the an optimal input sequence is applied. The optimization constraint on the final output in this example was to reach 0.5 ± 0.1 V in strain gage sensing circuit output at 20 milliseconds. A binary optimization was done

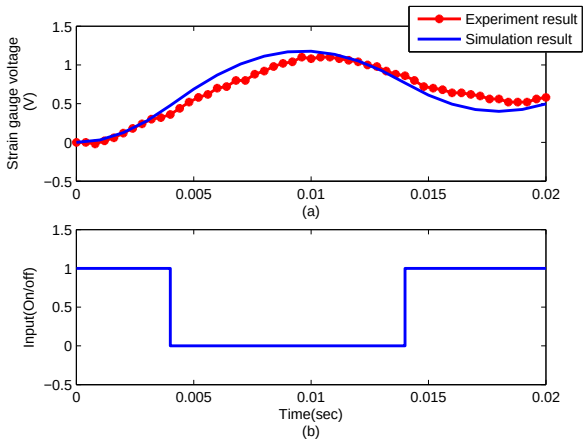

Figure 2.15: Comparison of results obtained by experiment and simulation for the Macro-scale system and the corresponding switching sequence

on the identified macro system given in (2.15) to obtain the input sequence shown in Fig. 2.15b and the constraints are verified using the MATLAB simulation shown. The same input sequence was applied to the macro system and is shown in Fig. 2.15a. The experimental response follows very closely with the simulation and reaches $0.58V$ at the desired time.

A similar approach with additional robustness constraints to account for hysteresis effects was applied to the MEMS actuator given in (2.16). In the optimization, the constraints were applied on both angle of rotation as well as angular velocity at 4 milliseconds for all the perturbed systems with stiffness varying between $k_{min} =$ 127.906 V/rad to $k_{max} = 190.476$ V/rad which is about $\pm 20\%$ of the nominal value, $k_{nominal} = 158.73$ V/rad. The requirement was to make the angle of rotation reach 0.15 ± 0.03 rad and angular velocity reach 0 ± 1 rad/s. From the optimization it was found out that the minimum possible tolerance on angle is ± 0.03 rad because of the 20% uncertainty in the stiffness value. The input sequence and simulation responses shown in Fig. 2.16 were obtained as a result, which satisfies all the constraints in the simulation. When the same sequence was applied on the physical system, the response shown with a red line was obtained. The experimental result follows the

simulation result closely and reaches about 0.11 rad at the stipulated time, which is very close to the expectation from the simulation. Also, the direction of rotation reverses in the same video frame, indicating that the angular velocity goes through zero verifying the final velocity constraint.

During this experiment, the cumulative energy consumption of the micro-robotic leg together with the switching circuitry was also measured. The current profiles while the switch was turned 'on' and 'off' were measured using a current probe and oscilloscope and are shown in Fig. 2.17. The power supply was kept constant at 20V during the entire experiment and each turning 'on' cost $4.6 * 10^{-7}$ joules and each turning 'off' consumed $1.4*10^{-7}$ joules. Of this quantity, $2.2*10^{-7}$ joules is attributed to charging at the microactuator, resulting in total energy loss in the circuit alone of just $3.6*10^{-7}$ joules per cycle. This is smaller than the predicted energy usage, which appears to be due to additional resistance within the leg between electrodes for the leg and the piezoelectric actuators; the extra resistance further increases the effect of the pull-up resistors included in the circuit design. Total power consumption of a robot using this controller would depend on step frequency, but for walking gates of 20 Hz or lower, power consumption would be in the tens of microwatts or smaller. This is within the power availability we predict for a micro-robot based on piezoelectric actuators, and much smaller than power consumption of an analog controller or PWM controller with much higher switching frequencies.

2.7 Conclusion

Although the optimal control method can be extended to include state feedback by using a model predictive control approach as shown earlier, the use of open-loop on-off control to regulate the motion of a piezoelectric actuator is driven entirely by the need to ensure extremely low power consumption from the entirety of a servo control system, and results in significant trade-offs in performance for the sake of

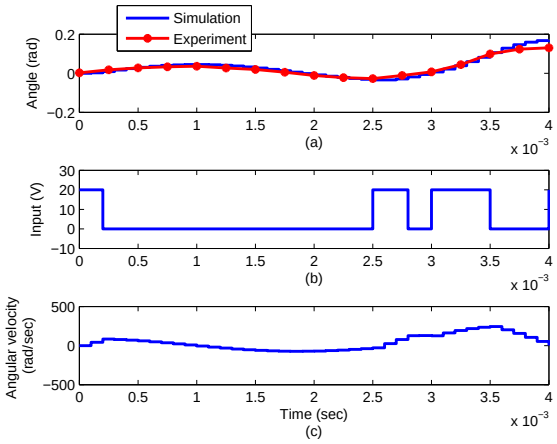

Figure 2.16: (a) Comparison of displacement results obtained by experiment and simulation for the MEMS system when under an optimal on-off sequence for 0.15 rad final displacement (b) On-off voltage input applied (c) Angular velocity observed in the simulation, showing successful return to 0 rad/s at final time

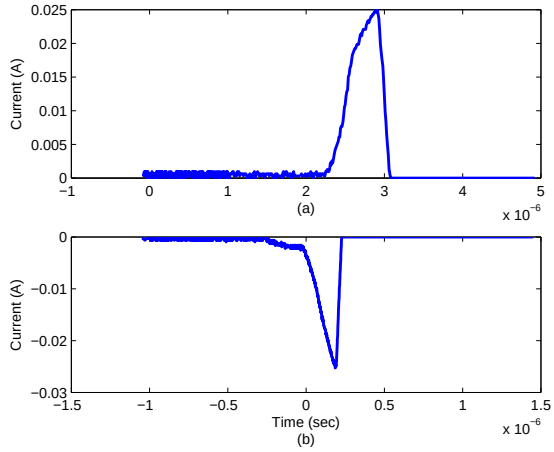

Figure 2.17: Current consumption of switching circuitry and the MEMS actuator while the switch is (a) turned on and (b) turned off

power reduction. A near optimal feedback strategy using a position sensor sparingly is discussed later in chapter 4. The simple switching interface between controller and actuators, and small number of transitions utilized, allows for control of micro-actuators that act primarily as a capacitive load with very little energy consumption. However, such a controller explicitly forgoes the use of feedback to improve response time, robustness, or other closed-loop controller benefits due to power limitations,

and the relatively low switching frequency results in oscillatory output motions with specific desired output states being achieved only at a specific time. In addition, switching controllers may often excite high-frequency dynamics, though this is not a large effect in the experimental test actuators examined here. If present, higher-order dynamics may be incorporated into the optimization procedure described here by expanding the system order, but the controller will only act to ensure that those dynamics do not influence behavior at the final time, not to avoid vibration or oscillation at intermediate transitions.

Nonetheless, this controller could provide useful baseline control for autonomous micro-robotics. On-off control with a desired final state at a given time is especially well-suited to quasi-static walking, with multiple legs driven through coordinated motions over a specified time, at which legs in contact with the ground are raised and raised legs lowered for the next step. In such an application, total inertia of the system would be much larger than that of the leg alone, while damping in the system will depend on the geometry of the body and feet as well as legs. In practice, the environment and loads on the robot may change dramatically, requiring some level of adaptation or feedback. A variety of optimized on-off sequences under different load conditions may be stored for use, or the optimal on-off sequence may serve to initiate adaptive control using only occasional or very low-frequency sensor feedback to make modest adjustments to the input sequence while keeping power consumption low.

The chief conclusions of this work, then, are based on the idea that while thin-film piezoelectric actuation may one day enable unique mobility capabilities from micro-scale autonomous robots, achieving directed appendage movements will require aggressive reduction in power consumption throughout the servo control system. In this chapter, we focus reducing power consumption by the actuators and in the drive circuit interfacing low-voltage control electronics with a high-voltage actuator supply. In particular, we introduce a simple optimization method for achieving minimum

energy on-off control when switching costs are substantial. We have applied the control algorithm to both a macro-scale piezoelectric test bed and to prototype micro-robotic leg joints, successfully directing the states of the systems to desired target values.

CHAPTER III

Optimal charge recovery

The advantage of a pure on-off controller was its simplicity and the energy savings it made while charging piezoelectric actuator, but more energy can be saved while the actuator discharges. This can be done by using a circuit such as the one given in Fig. 3.1. In the circuit shown a storage capacitor, an inductor, two diodes and two switches are added to the switching controller discussed in the chapter 2. This arrangement is used to save the energy which would have drained when the actuator is discharged. Detailed operation of the circuit is discussed in section 3.1.2. This strategy was first proposed by [22] and [28] which recovered almost all the energy drained from the actuator while it turned off using a complete charge recovery circuit. One disadvantage of such a circuit is the presence of a heavy inductor. Hence, the circuit was modified for micro-robotics to use a lighter inductor recovering a fraction of the discharged energy, also referred to as partial charge recovery. Moreover, partial charge recovery provides intermediate voltages between the minimum and maximum voltage levels in a pure on-off controller. Hence the controller is more flexible and capable of spanning more points in the state space. However, the optimization of this problem is more challenging compared to a pure on-off controller, as will be discussed in this chapter.

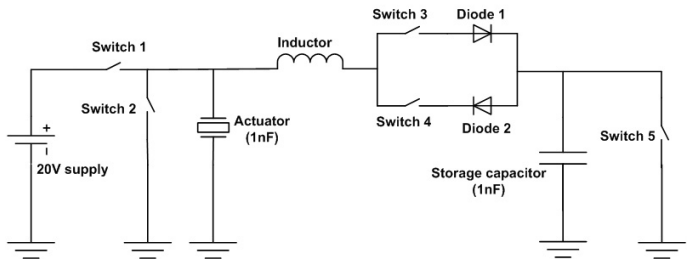

Figure 3.1: The charge recovery circuit

3.1 Modeling of the system

3.1.1 Dynamics

The dynamics of the system are modeled using a mass-spring-damper system similar to the one given in the first chapter. The differential equation is then converted to state space format using the angle of rotation and angular velocity as states. Then it is descritized with a sampling time T_S and the resulting equations are given below,

$$x\left((k+1)T_s\right) = A_d x(kT_s) + B_d u(k) \tag{3.1}$$

$$y(kT_s) = C_d x(kT_s) \tag{3.2}$$

The only difference in these equations with those in the last chapter is the values that are taken by the input variables $u(k)$. They were restricted to take only binary values in the first chapter. But by using the partial charge recovery the input voltages at time k, $u(k)$ can take more values depending on its previous value $u(k-1)$. This constraint is explained in later sections.

3.1.2 Electrical Model

As mentioned earlier a typical charge recovery circuit is given in Fig. 3.1, which is derived from that in [28]. It consists of one storage capacitor, an inductor, a pair

of diodes and a few switches. The storage capacitor is designed to be of the same capacitance as the actuator. The 20 V supply is connected to the actuator through switch 1. The actuator can discharge itself through switch 2, or it can charge the storage capacitor when switch 3 closes. Then, the stored energy on storage capacitor can be returned to the actuator through the switch 4 or discharged to ground through the switch 5. Only one of the switches 1 to 4 can be turned on at a time and switch 5 may be turned on only when switch 3 and switch 4 are in the off position.

Further explanation is required for the modes when either switch 3 or 4 is closed. Consider the situation when the actuator voltage is greater than the storage capacitor voltage. In order to transfer charge from the actuator to storage capacitor, switch 3 is closed. During the time that the current flows from actuator to storage capacitor the circuit can be modeled as a second order system in terms of charges stored in actuator q_1 and storage capacitor q_2 (3.3). For effective charge recovery, circuit parameters are chosen to make the system underdamped. The diode present in the circuit will prevent any reverse current and will keep the capacitor voltages constant at the overshoot points. Circuit dynamics prior to the diode action are given by

$$L(\ddot{q_1} - \ddot{q_2}) + R(\dot{q_1} - \dot{q_2}) + \frac{q_1 - q_2}{C/2} = V_D \tag{3.3}$$

where V_D is the voltage drop across the diode. Letting $q_1(0) = V_{10}C$, $q_2(0) = V_{20}C$ and the differential voltage $V = (q_1 - q_2)C$, where V_{10} and V_{20} are the initial voltages on the actuator and storage capacitor respectively. The above equation can be written in terms of differential voltage V as given below,

$$\begin{aligned} L\ddot{V} + R\dot{V} + \frac{V}{C/2} &= \frac{V_D}{C} \\ V(0) &= V_{10} - V_{20} \end{aligned} \tag{3.4}$$

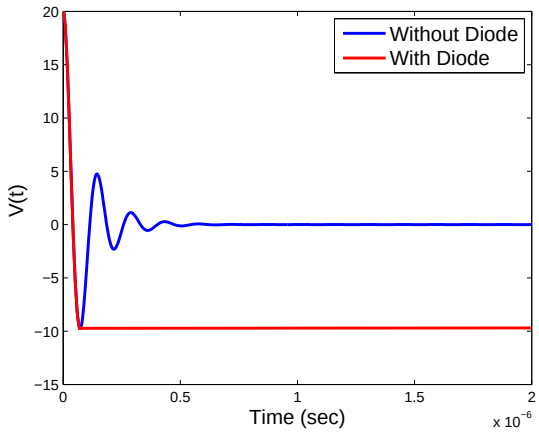

Figure 3.2: A typical variation in the differential voltage for circuit parameters $L = 1\mu H, R = 20\Omega, C = 1nF$ and ideal voltage drop in the diode

Assuming that $V_{10} > V_{20}$ before switching, then $V(t)$ will follow a step response until it reaches the maximum value and will stay there because of the diode. A typical response is shown in the Fig. 3.2. The response of the system up to the overshoot point then can be written as,

$$V(t) = V_D + (V(0) - V_D)e^{-\alpha t}(cos\beta t + \frac{\alpha}{\beta}sin\beta t) \tag{3.5}$$

where $\alpha = \frac{R}{2L}$ and $\beta = \frac{\sqrt{\frac{8L}{C} - R^2}}{2L}$.

$V(t)$ reaches its maximum value when $t = \frac{\pi}{\beta}$ and the corresponding differential voltage will be the difference between the actuator voltage, V_{11}, and the storage capacitor voltage, V_{21}, after switching.

$$V_{max} = V_{11} - V_{21} = V_D - (V(0) - V_D)e^{-\frac{\alpha\pi}{\beta}} \tag{3.6}$$

The above equation together with the charge conservation equation

$$C(V_{10} + V_{20}) = C(V_{11} + V_{21}) \tag{3.7}$$

can be used to evaluate the actuator voltage and the storage capacitor voltage after switching:

$$V_{11} = V_D(\frac{1+\mu}{2}) + V_{10}(\frac{1-\mu}{2}) + V_{20}(\frac{1+\mu}{2}) \tag{3.8}$$

$$V_{21} = -V_D(\frac{1+\mu}{2}) + V_{10}(\frac{1+\mu}{2}) + V_{20}(\frac{1-\mu}{2}) \tag{3.9}$$

where $\mu = e^{-\frac{\alpha\pi}{\beta}}$. Using a diode with negligible voltage drop compared to the maximum voltage, the above equations can be approximated to form symmetrical equations.

$$V_{11} = V_{10}(\frac{1-\mu}{2}) + V_{20}(\frac{1+\mu}{2}) \tag{3.10}$$

$$V_{21} = V_{10}(\frac{1+\mu}{2}) + V_{20}(\frac{1-\mu}{2}) \tag{3.11}$$

These equations are derived for the case when the actuator voltage is greater than the storage capacitor voltage. If the storage capacitor voltage is higher, the charge can be returned to the actuator by closing the switch 4 (and opening switch 3) and the equations are still valid.

3.2 Optimization method

3.2.1 Objective function

The objective of the optimization is to drive the states of the system to a desired set of values in a prescribed time using minimum energy. Hence, the objective function

is the energy used by the actuator directly from the power source. As in the previous chapter, this energy consumption can be divided into two parts. The first part is termed capacitative loss J_C, which is consumed whenever the actuator is charged using external power. Or in other words, when a transition in voltage states on the actuator occurs that is powered externally, it is quantified as

$$J_C = \sum_{k=1}^{n} C(u_{switch_1}(k)(u(k)^2 - u(k-1)^2) + u_{switch_1}(0)u(0)^2) \tag{3.12}$$

where C is the capacitance of the piezoelectric actuator and $u_{switch_1}(k)$ is a binary variable which takes value 1 if switch 1 is on at k^{th} time instant and 0 otherwise and $u(k)$ represents the voltage on the actuator at the k^{th} instant, which can take any allowed voltage given in the automata described in the following section.

The second portion of possible energy use is a resistive loss denoted by J_R which occurs during the time when the actuator is externally powered,

$$J_R = \sum_{k=0}^{n} \frac{u(k)^2}{R} T_s u_{switch_1}(k) \tag{3.13}$$

where R is the resistance of the system and T_s the sampling time used for discretization. In piezoelectric films with high leakage resistance, the resistive loss is negligible over short movement duration compared to the capacitative loss, and hence it is not considered in this optimization.

3.2.2 Constraints

There are three types of constraints in the optimization problem: the dynamics of the system, the final desired states and the constraints on the inputs which are explained in the following subsections. The dynamics constraints and final state

constraints are the same as discussed in the last chapter.

3.2.2.1 Input transition constraints

From the optimization point of view, these constraints on input transition is the difference between the charge recovery optimization problem and the on-off controller optimization problem discussed in chapter 2. In a simple on-off controller the input $u(k)$ can take either voltage levels of u_{min} or u_{max} only with transitions happening at the sampling instant. In charge recovery problem intermediate voltages can be achieved as well depending on the size of the components (primarily the inductor) in the charge recovery circuit as given in (3.6). Several scenarios for use of the charge recovery circuit are described below.

Case 1: A simple case is considered first. The storage capacitor discharges all its energy whenever the actuator gets charged or discharged, effectively resetting the states of the automaton. By this assumption the intermediate voltages can be predetermined. The scenario is explained in the following paragraph.

The actuator is charged to V_{max} when switch 1 turns on and it will either drain the charge to ground through switch 2 or it will charge the storage capacitor through switch 3. If it drains the energy the actuator voltage will go back to V_{min} and if it charges the storage capacitor the actuator voltage will reach $V_{Intermediate_1}$. From $V_{Intermediate_1}$ the actuator can return to V_{min} or V_{max}, but the storage capacitor will drain all its energy. If the actuator recovers the charge through the switch 4, it will attain $V_{Intermediate_2}$. From this state, actuator voltage may be maintained, fully charged or discharged. This results in just 4 voltage levels as shown in the automaton given in Fig. 3.3. The states of the automaton represent the actuator voltage at each instant of time.

In order to convert these constraints to equations, a new set of binary variables v_1, v_2, v_3, v_4 are introduced, each corresponds to an input voltage in the set

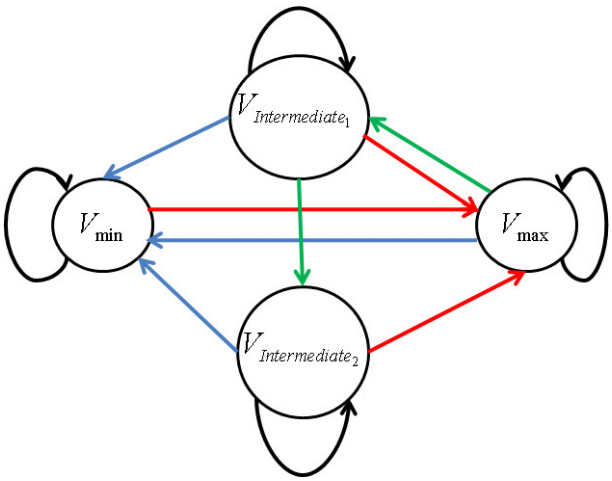

Figure 3.3: The automaton showing the constraints on actuator voltage transitions for the simplest case

$V_{min}, V_{max}, V_{Intermediate_1}, V_{Intermediate_2}$. So the input voltage at time $t = k * T_s$ can be written as,

$$u(k) = v_1(k)V_{min} + v_2(k)V_{max} + v_3(k)V_{Intermediate_1} + v_4(k)V_{Intermediate_2} \tag{3.14}$$

and the following constraint ensures that the system is in only one state at a time (note that $v's$ are binary):

$$v_1(k) + v_2(k) + v_3(k) + v_4(k) = 1 \forall k \in 0..n \tag{3.15}$$

Additional constraints ensure illegal transitions such as a transition from V_{max} to $V_{intermediate_1}$ do not happen. This can be done by making sure that $v_2(k)$ and $v_3(k+1)$ are not simultaneously one, as by the inequality constraint $v_2(k) + v_3(k+1) \leq 1$.

Case 2: In the second case we remove constraints on the transitions of storage capacitor voltages. This is mathematically more challenging to optimize, but it has

greater potential to improve the efficiency of the system. The intermediate actuator and storage capacitor voltages are not constants in this case. The values the voltages can take at any point of time $t = (k+1)T_s$ depend on the charge stored in the actuator and storage capacitor at the previous instant $k*T_s$. At time $t = kT_s$ let the voltages on actuator and storage capacitor be V_{10} and V_{20} respectively. Then at the next instant the actuator voltage can take any of the following values $\{0, 20, V_{10}[\frac{1-\mu}{2}] + V_{20}[\frac{1+\mu}{2}]\}$ and storage capacitor voltage can take any value from the set $\{0, V_{20}[\frac{1-\mu}{2}] + V_{10}[\frac{1+\mu}{2}]\}$. The derivation of these voltages is shown in (3.6).

This problem has a new non-linear constraint to make sure that the actuator voltage and the storage capacitor voltage are one of those allowed at the current time instant, as given below. The binary variables $w_1(k)..w_7(k)$ ensure that the state in the automaton, represented in the form "(actuator voltage, storage capacitor voltage)," is one of the allowed automaton states, as shown in Fig. 3.4, which represents a system of allowed binary switching sequence:

$$u(k) = w_1(k)V_{max} + w_2(k)V_{min} + w_3(k)(u(k-1)[\frac{1-\mu}{2}] + u_s(k-1)[\frac{1+\mu}{2}]) + w_4(k)u(k-1) + w_5(k)u(k-1) + w_6(k)V_{max} + w_7V_{min} \quad (3.16)$$

$$u_s(k) = w_1(k)V_{min} + w_2(k)V_{min} + w_3(k)(u_s(k-1)[\frac{1-\mu}{2}] + u(k-1)[\frac{1+\mu}{2}]) + w_4(k)V_{min} + w_5(k)u_s(k-1) + w_6(k)u_s(k-1) + w_7u_s(k-1) \quad (3.17)$$

$$w_1(k) + w_2(k) + w_3(k) + w_4(k) + w_5(k) + w_6(k) + w_7(k) = 1 \quad (3.18)$$

Here $u(k)$ and $u_s(k)$ are actuator voltage and storage capacitor voltage at time $t = k*T_S$ respectively. Some of these constraints are non-linear due to the multiplication of variables with binary variables. These type of constraints can be converted to a set of linear constraints by introducing a set of new variables by the following procedure [45]. Consider a new variable $z(k)$, to replace the term $w_4(k)u(k-1)$ in (3.16). By adding the following four additional linear constraints the above equation can be

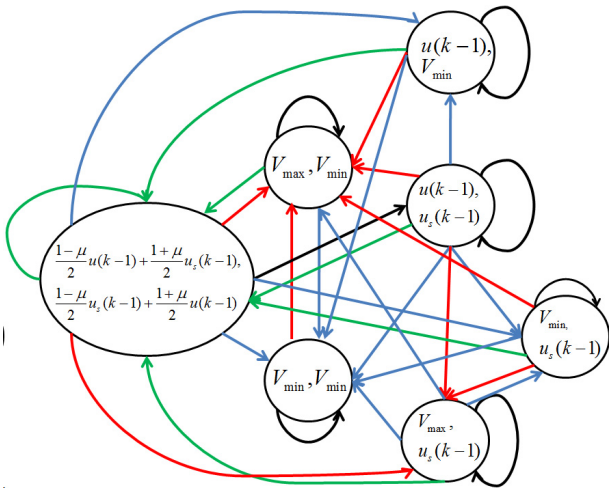

Figure 3.4: Automaton showing the constraints on the input voltage for the case 2

converted to a linear equation on z.

$$\begin{aligned} z(k) &\leq M w_4(k) \\ z(k) &\geq m w_4(k) \\ z(k) &\leq u(k-1) - m(1-w_4(k)) \\ z(k) &\geq u(k-1) - M(1-w_4(k)) \end{aligned} \tag{3.19}$$

where $M = max(u(k-1)) = V_{max}$ and $m = min(u(k-1)) = V_{min}$. Similarly each of the product terms are replaced by new variables and additional linear constraints are added.

The constraints on transitions of actuator voltage and storage capacitor voltages are shown in the Fig. 3.4. The transitions marked red involve external power usage, green employ the charge recovery circuit, black represents staying at the same state and blue transitions are discharge of either actuator or storage capacitor. From the initial optimization it was observed that the full range of possible leg motions (i.e. target final angles) can be achieved with one external powered charging of the actuator at the beginning of the motion. Thus, the problem of motion optimization may be extended to maximizing the energy stored in the storage capacitor at the end of the optimization horizon for future use.

3.2.2.2 Modified optimization with final storage capacitor voltage maximization

Since the external energy used for a given required rotation is the minimum possible under the basic optimization problem (a single charging from the external power source) for the range of possible actuator final angles, the optimization objectives were modified to better suit repeated motions. The new objective function is set to maximize the storage capacitor voltage at the end of the optimization horizon. By maximizing the final storage voltage more than one step of actuation can be achieved by one externally powered voltage switching for smaller leg motions or the additional energy required to repeat the current motion can be minimized for larger leg motions.

$$J_{modified} = u_s(n) \tag{3.20}$$

where n is the number of allowed time steps to reach the desired final state as given the earlier section. The minimum energy requirement of one external powered charging of actuator applied as an additional constraint. Mathematically this is done by limiting the number of transition from V_{min} to V_{max} to one,

$$\sum_{k=0}^{n} w_2(k)w_1(k+1) \leq 1 \tag{3.21}$$

This nonlinear constraint was converted to a linear one using the method shown in (3.19). The constraints on the modified system are shown in the new automaton given in Fig. 3.5.

The structure of the optimization problem for the on-off controller and for different scenarios in charge recovery are compared in Table 3.1.

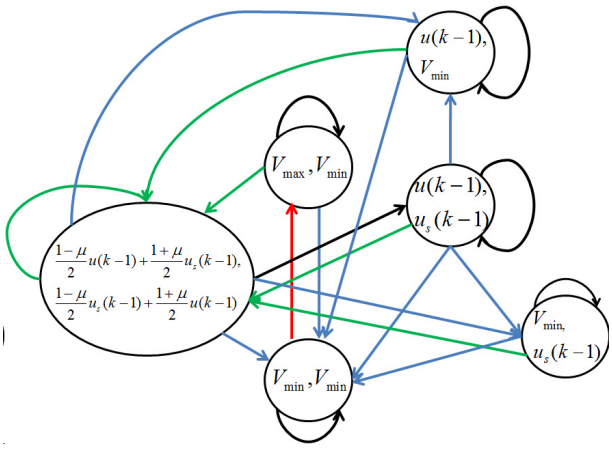

Figure 3.5: Automaton showing the constraints in the modified optimization

Table 3.1: Comparison between on-off and charge recovery optimization

	On-off controller	Charge recovery		
		case1	case2	Modified objective function
Objective function	Quadratic	Quadratic	Quadratic	Linear
Constraints	Linear	Linear	Modified to linear	Modified to linear
Number of variables	2n	4n	14n	12n

3.2.3 Optimization solver

All three problems discussed above had linear constraints after the conversion given in (3.19). The first two cases had quadratic objective functions and the third one was linear. All the constraint equations and objective function were written in AMPL and solved using CPLEX.

3.3 Experimental and simulational results

3.3.1 Experimental setup

Off-the-shelf fast switches (which turn on and off in less than 200 ns) from analog devices, ADG5412BRUZ, were used in the implementation of the controllers. The experiments were done on both a macro-scale system as well as on a micro-scale

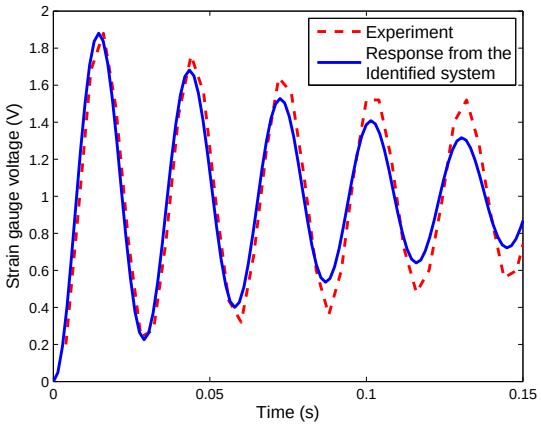

Figure 3.6: Step response of the macro system for a 10V input

actuator similar to the case with the on-off controller discussed in last chapter. The first set of experiments were done on the macro-actuator which is a 40 mm long, 10 mm wide Ceratec, Inc. bimorph actuator. A step input of 10V was applied and response is given in Fig. 3.6. A second order system was fitted between the applied voltage and the voltage from the strain gauge used to measure the deflection of the beam. A sampling 0.01s was used to discretize the system to use optimization methods. The static intermediate voltages were found to be 7V and 15.4V for the circuit with 1nF capacitor and 1mH inductor.

$$\frac{y(s)}{u(s)} = \frac{1}{2.1268 \times 10^{-4} * s^2 + 0.0037s + 10} \tag{3.22}$$

The MEMS actuator was also operated using the microprocessor and Analog devices switches. Motion of the rotational joint was captured using a high speed camera at 8000 frames per second and the angle of rotation was measured using the MATLAB Image Processing Toolbox. A step input of 20V was applied on this actuator and the corresponding response is shown in Fig. 3.7. The following second order system was identified between the input voltage and the angle of rotation of the leg in radians.

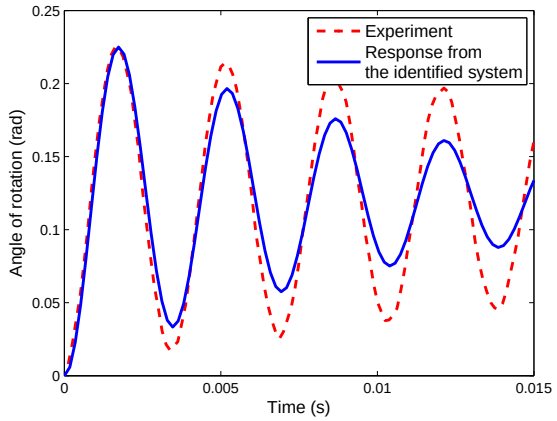

Figure 3.7: Step response of the micro system for a 20V input

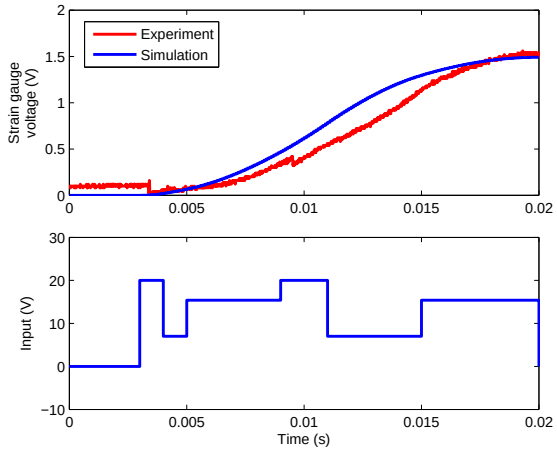

Figure 3.8: Macro system-charge recovery controller response for an optimal sequence to reach 1.5V at 20ms

Although the identified response doesn't match with the experiment result for the entire step response, it matches well for the initial period where the system is supposed to be during the operational period. In this case the charge recovery intermediate voltages were 9.6V and 12.4V for a 3.1nF storage capacitor and 4mH inductor.

$$\frac{y(s)}{u(s)} = \frac{1}{5.0039 \times 10^{-5} * s^2 + 0.0093s + 164.6091} \tag{3.23}$$

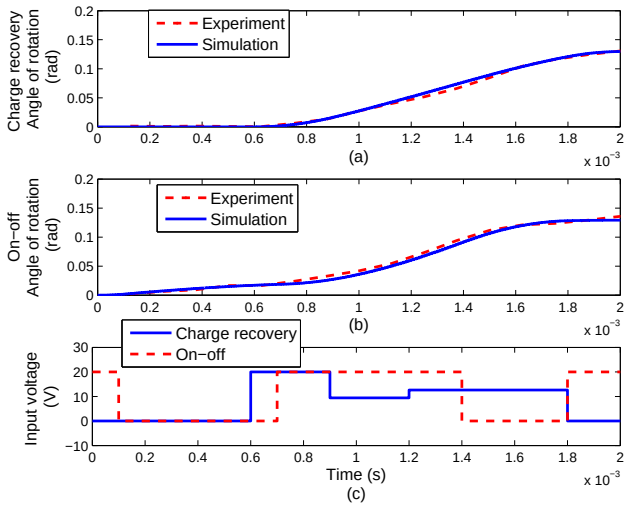

Figure 3.9: (a) Micro system-charge recovery controller response for an optimal sequence to reach 0.13rad at 2ms, (b) Micro system on-off controller response for an optimal sequence for the same state constraints (c) Optimal charge recovery controller and on-off sequences

3.3.2 Comparison between simulational and experimental result

The macro-system was discretized using a sampling time of 1ms and fixed intermediate voltage charge recovery was applied using the second order system identified in (3.22). The objective in the example shown was to reach (1.5±0.01V,±10V/s) at the strain gauge output at a final time of 20ms using minimum energy. In both experimental tests the simplest (Case I) version of charge recovery operation was used. In the simulation the final voltage settles to 1.4904V at 20ms as expected and the experimental response closely follows as shown in Fig. 3.8. The actuator is externally charged twice; once from 0 to 20V and a second time from 15.4 to 20V, consuming 0.2814μJ in its first step. For a second step onwards, depending on any leakage current in the storage capacitor, the storage capacitor can be used to help the first 0 to 20V charging as well, reducing the energy consumption.

Since the natural frequency of the micro-system was approximately 10 times more than that of macro-scale actuator, it was discretized with a smaller sampling time of 0.1ms. In this case the objective was to reach (0.130 ± 0.001 radians, $\pm$1rad/s)

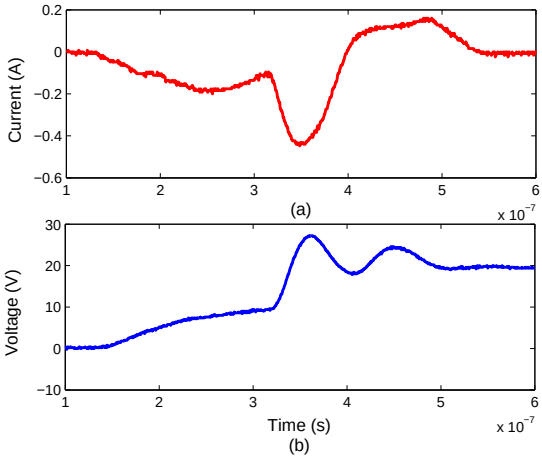

Figure 3.10: Current consumption of switching circuitry and the MEMS actuator while the switch is turned on from 0V to 20V

at 2ms. In the simulation the angle reaches 0.1297 radians at 2ms, which is in the satisfactory range, and the experiment also follows very closely and settles around the same value as shown in Fig. 3.9(a). The actuator is externally charged only once and the energy consumption is 0.2μJ for the first step and would be less from the second step onwards as explained in the last paragraph.

3.3.3 Comparison with on-off controller for power consumption

The advantage of charge recovery strategy over the on-off controller in terms of energy consumption is evaluated here. With the same state constraints used to find optimal charge recovery switching sequence, a pure on-off switching sequence which minimizes energy consumption was found. Both the simulation and experimental response of this sequence are shown in Fig. 3.9(b). The pure on-off optimization of this can be done by removing the charge recovery constraints. The simulation and experiment results match, with the on-off controller switches three times to achieve the desired performance. Hence it consumes $0.6\mu J$, three times more than the first step in charge recovery strategy consumed.

In order to measure the energy consumption in the switching circuitry and the

Table 3.2: Micro-actuator parameters used in the dynamic intermediate charge recovery simulation study

Parameters	Values
R (Ω)	$3 * 10^9$
C (F)	$1 * 10^{-9}$
$V_{max}(V)$	20
J ($kg.m2$)	$1.4 * 10^{-12}$
b ($N.m.s/rad$)	$2.7 * 10-10$
k ($N.m/rad$)	$4.4 * 10-6$
G ($N.m/V$)	$2.8 * 10^{-8}$
$R_{Diode}(\Omega)$	2
Sampling time (T_s)(sec)	0.0001

MEMS actuator, current and voltages were measured while the switch is turned on charging the actuator from 0 to 20V. The switching happens in less than a microsecond as shown in Fig. 3.10. The energy consumption was evaluated during this period from the current and voltage and it turns out to be $0.275\mu J$ which is in the similar order of the expected value given in the last section.

3.3.4 Multi-stage charge recovery with dynamic intermediate voltages

Dynamic intermediate voltage charge recovery could not be implemented in the experimental system because of discrepancies in the electrical model with the experimental setup. First, while the leakage current through the thin piezoelectric layers was not large enough to be a significant source of power draw for the small durations where the actuator is connected to the power source, it is enough for voltage and charge to decrease significantly over the entire course of a motion. Second, the effective dielectric constant of a thin-film piezoelectric material is dependent on applied voltage, and would need to be measured across possible operating voltages to accurately construct the automaton representation of charge recovery behavior. Thus, the more advanced charge recovery models (Case 2 and multi-step optimization) are more appropriate to electrostatic actuation (a more constant capacitive load) or bulk piezo-

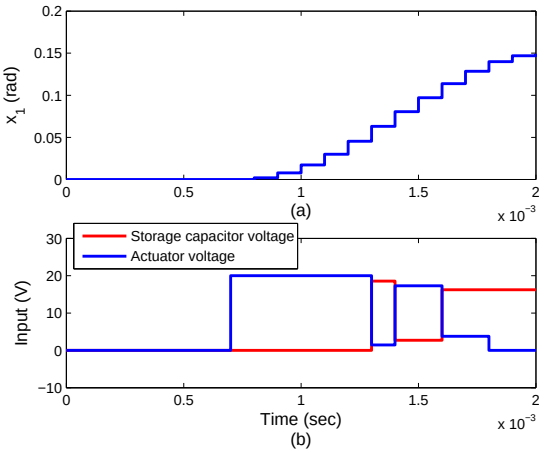

Figure 3.11: Charge recovery with dynamic intermediate voltages simulation result

electric actuators (where leakage and dielectric effects are smaller, although hysteresis effects would need to be included). Nonetheless, a simulation study was conducted to explore potential advantages of multi-stage dynamic intermediate voltage charge recovery on an ideal model. From this study it was found that for possible range of angles (0.2 radians with 20V supply) with satisfactory accuracy (accuracy of the order of 0.001 radians for angles and a few radians per second for angular velocities) could be reached by a single externally powered switching in dynamic intermediate voltage charge recovery. Hence, the objective was changed to extract more from the single power consuming switching. This is done by maximizing the final storage capacitor voltage and limiting the externally powered switching to one. Average power consumption in the on-off controller and the charge recovery scenarios are 600μW and 250μW respectively over a time period of 2ms per step in this case.

The state constraints in this example are set to be $x_d \in$ (0.14 rad, 0rad/s) $\pm$ (0.001rad, 1rad/s) at 2ms and single externally powered switching is added as a constraint, leaving the objective here to maximize the end storage capacitor voltage. In Fig. 3.11 a typical response for this scenario is given. The final storage voltage is 16.24V. So in the remaining steps this voltage together with external source can be utilized to charge the actuator. Hence the average power consumption for each

remaining step is only 86.8μW over 2ms.

3.3.5 Closed loop extension using model predictive control

At present, sensors have not been integrated effectively with the micro-robotic legs. However, the optimization method can be extended to take advantage of feedback if sensors are available. In this simulational study we compared the system performance in the presence of a constant disturbance with and without feedback. The feedback optimal sequences were found using the receding horizon philosophy [45]. In the receding horizon method, at the beginning of the motion the optimal sequence was found for the entire range, but only the first input value was applied. After collecting the sensor information from the first time instant, the optimization is again done for the new horizon starting from the second time step to the end to obtain a new sequence.This process continued for the entire duration applying only the first input value from the each optimization.

The objective in this study is to reach in the neighborhood of 0.1 radians at 2ms. In the first case where the disturbance is a constant 2V, the feedback successfully steers the system to a satisfactory range where open loop system fails which is shown in Fig. 3.12. In the second case where the disturbance levels were increased to -4V level, both methods fail, as shown in Fig. 3.13. However the closed loop system takes the final state to a closer value than the open loop. Hence, in general, the closed loop showed better performance compared to the open loop controller, and charge recovery may be used in both open- and closed-loop scenarios.

3.3.6 Simulation with a second actuator used as a storage capacitor

A simulation study was also conducted to investigate the possibility of coordinating motion of two actuators in which one acts as a storage capacitor. This is intended to mimic the use of charge recovery in a robot utilizing two sets of legs, one set to drive

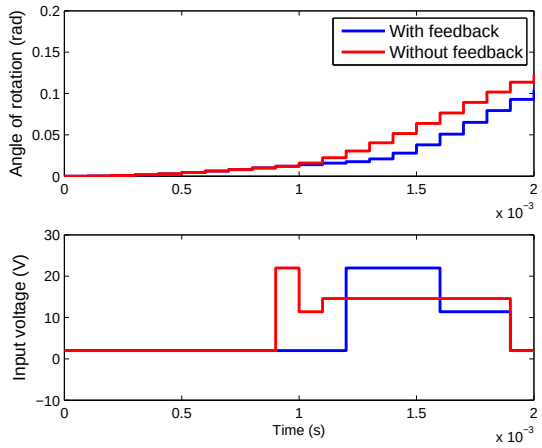

Figure 3.12: Comparison between open loop and closed loop optimal controllers with 2V constant disturbance

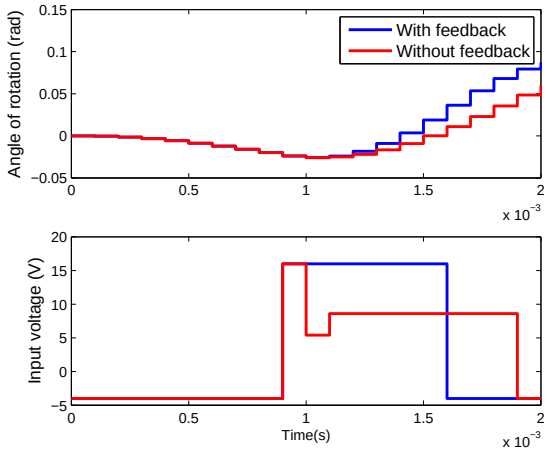

Figure 3.13: Comparison between open loop and closed loop optimal controllers with -4V constant disturbance

the robot forward during a step and the other set returning to its nominal position over the same time period. The objective in this simulation was (0.15 ± 0.001 radians, 0 rad/s$\pm$1rad/s) at 3ms for actuator 1, which starts from a stationary position at 0 radians. Actuator 2, which acts as a storage capacitor starts from (0.15rad,0 rad/s), as though it has completed a step, and is desired to stop at 0 radians at 3ms. As can be seen in Fig. 3.14(a) the actuators achieved the desired goal. We observe from the study that this puts additional constraints on the optimization and the feasible region is smaller. Hence, an extended time period was found to be necessary to coordinate

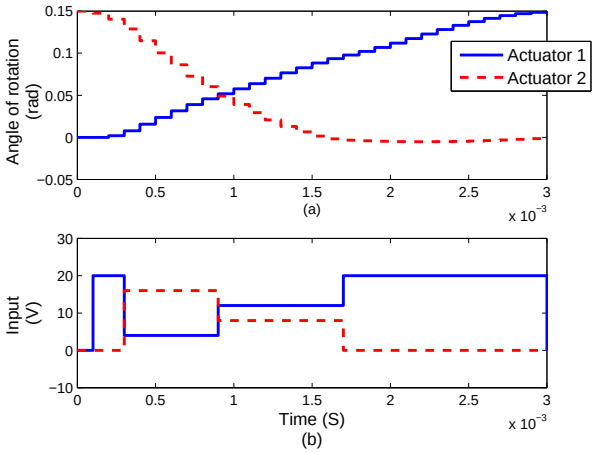

Figure 3.14: Responses of two actuators while actuator 2 acts as a storage capacitor and returns to zero displacement state

both legs effectively, and certain goals which were obtainable in the charge recovery of a single actuator alone may not be feasible when two actuators are combined.

3.4 Inductor sizing

A major objective of this study is to minimize the weight on board of the robot due to the power storage and related circuitry. The inductor is the heaviest component in the charge recovery circuit (power electronic circuit in general) and the inductance directly influences the intermediate voltages and hence the performance. Weight increases with the inductance, as does charge recovery circuit efficiency. Hence, it is necessary to chose the correct inductor for maximizing the usefulness of the charge recovery circuit.

Each externally powered switching from 0 V will cost CV_{max}^2 joules. So if n number of switches are made in a time period of T, the power required will be $CV_{max}^2 \times n/T$. In the results discussed earlier there are three switches in 2milliseconds and hence the power consumption is 600μW. On the other hand the charge recovery with dynamic intermediate voltage strategy uses 86.8μW. So the power savings is 513.2μW. Considering a sample battery power density of 200 W/Kg [21], this can

Table 3.3: Accuracy vs. Weight savings for different inductors

ϵ_1	ϵ_2	on-off power used (μW)	L=56nF		L=550nF	
			power used (μW)	Weight saved (mg)	power used (μW)	Weight saved (mg)
0.01	10	400	42	0.92	33	0.73
0.01	1	400	125	0.63	43	0.68
0.001	1	600	125	1.67	52	1.64
0.001	0.1	800	143	2.53	57	2.61

save a battery weight of 2.57mg. But a chip inductor with inductance 200nF weighs 0.97mg [46]. Hence the net weight saving is 1.60mg.

A simulation study was conducted to understand the trade off in weight savings and accuracy for various inductor sizes, with a comparison between using a small inductor and a large inductor shown in the table 3.3. The state constraints for this study were $x_d \in$(0.15 rad, 0rad/s)$\pm(\epsilon_1, \epsilon_2)$ at 2 ms and the permissible error bounds ϵ_1 and ϵ_2 are given in the first two columns. The pure on-off controller power consumption is given in the third column. In the remaining columns the power used and weight savings from the two inductors are also given. The weights of the inductors are 0.75mg and 1.1mg respectively [46].

The average power consumption increased for better accuracy for all the controllers. But, because of the availability of intermediate voltages in charge recovery controllers, the increase in the energy consumption with accuracy was much smaller with partial charge recovery compared to pure on-off controllers. When the tolerance levels in the states are higher, the use of lighter inductors are beneficiary because of their higher weight savings. But for the tighter tolerance levels, the weight savings were very similar. Overall, using a lighter inductor generally offered better weight savings, given a range of potential error tolerances.

3.5 Conclusion

After implementing the pure on-off controller, the next step of micro-robot control studied was to incorporate a charge recovery circuit. This saves a portion of the energy drained from actuators when voltage is discharged. Compared to the two possible input values of a pure on-off controller, a partial charge recovery controller has four or more different input levels which makes the optimization problem more complex. However, the additional input levels make the controller more flexible and allow it to span more reachable points in the state space over a given time period, and the incorporated charge recovery can decrease actuator power consumption. Integer programming-based optimization was again employed here to minimize the energy consumed by the MEMS piezoelectric actuator-charge recovery system. The optimization incorporated both the dynamics of the actuator as well as circuitry constraints. The benefit of this optimization is shown as the savings in the combined weight of battery and the related circuitry, as well as improved accuracy in motion.

To minimize the weight on board a micro-robot for a given motion, the partial charge recovery strategy is used which is also advantageous for reducing the necessary inductor size needed to recover some energy from the actuator. Although this only saves a substantial fraction of the energy drained (compared to a complete charge recovery), it enables the flexible controller and eliminates the need to use heavier inductors. A study was also conducted to select the correct inductor for a given motion. Since the actuator is designed for an autonomous micro robot which will walk in a quasi-static manner, the final angle is going to be the optimum angle for a particular gait. Hence the inductor can be chosen for a particular standard gait sequence of the robot. In practice, the net effect of the charge recovery controller implementation is to reduce overall mass of a micro-robotic power system significantly, while enabling smoother desired motions from piezoelectric actuators.

Since the control strategies implemented here are open-loop, their performance

depends on the success of the model identification. Presence of disturbance and other model discrepancies can not be dealt without feedback. We have showed the benefits of this in the simulational study. However, the satisfactory results are obtained under the laboratory conditions with careful system identification. The primary objective of designing a low-power benchmark controller which can be carried by the micro-robot is satisfied.

CHAPTER IV

Sensor scheduling

Since micro-actuator motions are often subject to disturbances, some level of feedback is generally desirable if specified displacements are to be reliably achieved. On the other hand, existing MEMS position sensors, such as piezoresistive or capacitive deflection sensors consume significant amounts of energy if they are used at the same frequency as the input on-off controller. Hence, a method for using an energy consuming sensor sparingly together with the on-off controller was developed. Intermittent measurement times along with input update times and values for on-off control of linear systems are selected to minimize state error under sensing and control constraints imposed by strict energy limits.

As a frame of reference, capacitances of proposed thin-film lead-zirconate-titanate micro-actuators for micro-robotics are often on the order of 0.1 to 1 nF [47] [15], requiring nominally 40 to 400 nJ per charge/discharge cycle at 20 V. In comparison, state-of-the-art commercial low-power capacitive sensing circuits with A/D conversion require the equivalent of about 500 nJ per sample acquisition (50 μW for 100 samples/s [48]), and piezoresistors at this scale would need to be operated at less than 2% duty cycle to achieve comparable energy consumption. Experimental circuits based on semi-analog, duty-cycle-output processing appear capable of dramatically reducing energy consumption [49], and sensors based on direct voltage sensing from

piezoelectric elements have likewise been demonstrated with much smaller energy consumption per sample [50]. However, energy consumption per sample taken by these techniques still ranges from about 2 to 85% of that required for each piezoelectric actuator use, and these latter techniques are less directly compatible with low-power micro-controllers and displacement sensing, respectively. Meanwhile, adjusting future on-off control inputs at only a limited number of control input update times, as will be done in this chapter, reduces the amount of computation and/or memory required to perform nonlinear programming or reference a lookup-table to optimize control inputs.

Thus, this chapter examines how to best perform on-off control of systems when only a small number of feedback measurements and updates to control inputs are permitted over a finite time period. Several previous studies on sensor scheduling have some relevance to this study. Recently, Mo and Sinopoli in [29] proposed the existence of a critical measurement rate for an unstable system which bounds the error on state estimates. They have developed these results for an unreliable wireless sensor network where the system was modeled as an unstable system. Similar studies can be found in [30],[31], [32]. In [33] Li et. al. discussed a convex optimization strategy that minimizes the maximum error of certain states by having a sensor visit those states frequently. Another group of researchers has been investigating efficient use of multiple sensors by turning on a smaller set of sensors at a time [34],[35], including for model-predictive applications [36]. This study differs from previous work by restricting controlled system inputs to on-off switching. As the final set of states becomes a function of binary variables, the problem can become a complex one to optimize even with certain simplifying assumptions: first, there is only a single sensor which is turned on sparingly (a small finite number of times over the duration of motion) and second, the objective function is expressed only in terms of the final states. Following a sensor measurement, adjustment of future

on-off inputs to the system is similar to that of certain hybrid or model-predictive controllers. For example, for the controller in this paper, a strategy derived from the receding horizon approach for switching controller optimization of [27] is adopted, while similar model-predictive control algorithms with final costs have been studied in [37]. However, these previous works generally assume fixed sampling rates with frequent measurements, concentrating their studies on the optimal controller design, while in this paper controller design is combined with the sensor scheduling problem, for the on-off input, single-sensor conditions noted above.

This chapter is organized as follows: section 4.1 introduces the stochastic LTI system used for this study and the objective function and constraints on the problem; in section 4.2, the general optimization problem is reduced to a set of simpler optimizations requiring less computational effort under certain conditions; in section 4.3 a three step strategy is described for finding the best measurement times, input update times and input sequence for achieving specified performance; section 4.4 discusses four case studies showing the effectiveness of the new solution strategy in various scenarios; in section 4.5 an optimal on-off controller with energy costs included is combined with sensor scheduling for a near optimal solution with regards to actuator energy usage; section 4.6 concludes the chapter with advantages and disadvantages of the sensor scheduling strategy.

4.1 System description

The discretized state space representation of the system to be studied, having n states and a single input and output, is:

$$\begin{aligned} x\left(k+1\right) &= Ax(k) + Bu(k) + B_w w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned} \tag{4.1}$$

where A, B, B_w and C are the state matrix, input matrix, disturbance input matrix, and output matrix, respectively same as given in the last chapters, and $w(k) \sim N(0, Q)$ and $v(k) \sim N(0, R)$ represent gaussian disturbance and measurement noise. The initial state is also assumed to be a normal random variable $x(0) \sim N(0, P_0)$. The system is assumed to be stabilizeable and detectable if there are no input and measurement constraints, but for the controller to be implemented the input variable $u(k)$ is assumed to be binary and the transition of input variables happens only at the sampling time instants. Because of this constraint, only discrete points in state space are reachable over finite time, and the objective of the intermittent feedback controller is to minimize the expected value of the quadratic error function from desired final states, x_d, at time N, as in

$$\min_{u(i)=\{0,1\}} E[(x(N) - x_d)'(x(N) - x_d)] \tag{4.2}$$

while making measurements sparingly (i.e., taking measurements at $M << N$ time steps). Since the energy consuming sensor is turned on only at a few instances in a given optimization horizon, it is important to choose those times carefully. The question studied here is given a specified number of measurements (or in other words, given an allowed sensor energy consumption), what is the best performance that can be achieved and what are the sampling and update times associated with that performance. Although constant unknown disturbances are expected to be the predominant disturbance in micro-robotic applications, the more general problem of randomness entering at each time step is considered. If desired, error due to constant disturbances may be included with appropriate addition of integrators to the system model. Likewise, integration terms related to output can be added if error over the duration of motion is of interest, rather than just the states of the system at the final time.

As a preliminary, consider the deterministic open-loop problem, in which optimal input sequences (values for $u(0)$ to $u(N-1)$) can be obtained by solving an integer pro-

gramming problem minimizing the quadratic error function $(x(N) - x_d)'(x(N) - x_d)$. Constraints on this problem are the deterministic part of the dynamic constraints given in (4.1), or the solution when $w(k) = 0$ and $v(k) = 0$. Solving for the nominal open-loop inputs without disturbance is the first step in the solution strategy discussed later. However, to improve the expected value of the quadratic error function in the presence of disturbances and noise with a fixed, small number of measurements, future on-off inputs may be re-optimized to update the input on-off sequence based on measurement information. Note that updating input sequences more often than the number of measurements doesn't provide any additional benefit, since no improved knowledge of actual system states is obtained. Also, it is shown later that in certain situations, updating inputs after every measurement has little benefit for system performance and desirable results can be achieved by updating the input sequence fewer times than the number of measurements, thus also avoiding unnecessary on-line computations or lookup table memory storage.

To introduce the sensing terminology, consider the simplest case where measurements are taken only once over the optimization horizon. Let measurement time be T_l. States of the system based on this measurement can be estimated using an intermittent Kalman filter for any time instant at or after the measurement time. Based on these estimates the input sequence can be updated at any time $T_u \in [T_l, N]$ by solving a second integer programming for $N - T_u$. The estimates of the states at the update time using Kalman filter equations can be written as,

$$\hat{x}(T_u) = A^{(T_u - T_l)}\hat{x}_{T_l/T_l} + \sum_{i=T_l}^{T_u-1} A^{T_u-1-i}Bu(i) \tag{4.3}$$

$$\hat{x}_{T_l/T_l} = \hat{x}(T_l) + P_{T_l}C'[CP_{T_l}C' + R]^{-1}$$

$$(Cx(T_l) - C\hat{x}(T_l) + v(T_l)) \tag{4.4}$$

$$\hat{x}(T_l) = A^{T_l}\hat{x}(0) + \sum_{i=0}^{T_l-1} A^{T_l-1-i}Bu(i) \tag{4.5}$$

$$P_{T_l} = A^{T_l}P_0A'^{T_l} + \sum_{i=0}^{T_l-1} A^iBQB'A'^i \tag{4.6}$$

where $\hat{x}_{T_l/T_l}$ is the estimation of states after measurement and $\hat{x}(T_l)$ is the estimation of states before measurement. As mentioned, a new input sequence for the remaining $N - T_u$ time steps is found by solving an integer programming problem that minimizes $(\hat{x}(N) - x_d)'(\hat{x}(N) - x_d)$ where $\hat{x}(N)$ is given as,

$$\hat{x}(N) = A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) \tag{4.7}$$

4.2 Problem reduction

Consider a general problem with a fixed number of measurements. Let the number of measurements be a. Also, let the number of updates be $b \leq a$. The objective function in this case can be written as,

$$\min_{T_{l1},T_{l2}..T_{la},T_{u1}..T_{ub},u(0)..u(N-1)} E\{[x(N) - x_d]'[x(N) - x_d]\} \tag{4.8}$$

The overall optimization involves finding optimal measurement times, update times and input sequences that minimize the expected value of the quadratic error function of final states. If there are no input constraints, this problem can be split into

two simpler optimization problems involving lesser computational effort. The first optimization problem is to determine the optimal measurement times and the second problem is to evaluate the inputs after the last measurement time based on the measurements (assuming that enough time steps remain to steer all controllable states towards their desired final values). In this section it is shown that, even under the on-off input constraints the overall optimization can be separated into simpler optimization problems, provided some assumptions taken to do so are verified afterward. Key results are derived for a single measurement case, then expanded to multiple measurements.

4.2.1 Single measurement

In this case, the optimization problem is to determine the minimum of $E[(x(N) - x_d)'(x(N) - x_d)]$ for a single measurement and corresponding measurement time T_l and update time T_u.

The final states can be written as,

$$x(N) = A^{N-T_u}x(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}(Bu(i) + B_w w(i)) \tag{4.9}$$

This can be modified by introducing the estimates at T_u to

$$\begin{aligned} x(N) = {} & A^{N-T_u}\hat{x}(T_u) + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) \\ & + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i) \end{aligned} \tag{4.10}$$

Assuming that there exists $u(i), i = T_u..N-1$ such that

$$A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) \approx x_d \tag{4.11}$$

or in other words,

$$E(||A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$$
$$<< E[(x(N) - x_d)'(x(N) - x_d)] \tag{4.12}$$

then (4.10) can be rewritten as,

$$x(N) = x_d + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i) \tag{4.13}$$

It is critical to note that under this solution strategy, (4.12) determines whether this simplifying approach is valid; this assumption will be revisited in the overall solution procedure in section 4.3. It is also worth noting that under on-off control constraints, the actuation error term in (4.12) is nondecreasing, as each additional time step provides an additional potential adjustment to the input that may cancel out additional deviation from the final desired states based on knowledge at T_u. Thus, given a selected measurement time T_l, T_u would ideally be set equal to T_l. However, in low-power control conditions, computation speed is generally limited, so it is often convenient to update inputs after some delay, T_d, from the measurement, such that $T_u = T_l + T_d$, and thus the following analysis allows T_u to vary from T_l.

So long as (4.12) holds true, utilizing intermittent Kalman filter equations, the objective function can be written as a function of measurement time T_l alone, separate from the update time T_u, as shown in Appendix A. Hence, for single measurement case the measurement time T_l can be found by solving the following simpler optimization

problem,

$$\min_{T_l}[trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i} - A^{N-T_l} P_{T_l} C'[C P_{T_l} C' + R]^{-1} C P_{T_l} A'^{N-T_l}\}] \tag{4.14}$$

where P_{T_l} is an intermittent Kalman filter error covariance. The quantity in (4.14) represents the expected value of the minimum error if input constraints have negligible effect on final error, or in other words if the desired state is reachable within some error bound as of the update time. Furthermore, once the measurement time is found, the input sequence can be updated by solving an integer programming problem at any time between T_l and the latest time that the assumption (4.12) is satisfied within a desired level of accuracy. This step is explained in detail in section 4.3.

4.2.2 Multiple measurements

Consider the case with two measurements taken at T_{l1} and $T_{l2} > T_{l1}$. To utilize the information gathered from the second measurement, input must be updated at or after T_{l2}, letting this time be $T_u \geq T_{l2}$.

Using intermittent Kalman filter equations, as derived in Appendix B, the optimization can be reduced to

$$\min_{T_{l1},T_{l2}}[trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i} - A^{N-T_{l1}} P_{T_{l1}} C'[C P_{T_{l1}} C' + R]^{-1} C P_{T_{l1}} A'^{N-T_{l1}}\} - A^{N-T_{l2}} P_{T_{l2}} C'[C P_{T_{l2}} C' + R]^{-1} C P_{T_{l2}} A'^{N-T_{l2}}\}] \tag{4.15}$$

Extending this result to a general multiple measurements case where measurements are done at $T_{l1}, T_{l2}..T_{la}$, the times can be found by solving the optimization

problem,

$$\min_{T_{l1},T_{l2}..T_{la}} [trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i} - \sum_{i=1}^{a} A^{N-T_{li}} P_{T_{li}} C'[C P_{T_{li}} C' + R]^{-1} C P_{T_{li}} A'^{N-T_{li}}\}] \quad (4.16)$$

When the assumption made earlier is not satisfied, additional updates should be done. Consider the case where b number of updates are made at $T_{u1}, T_{u2}..T_{ub}$, $T_{ub} \geq T_{la}$, with the modified assumption that $E[||A^{N-T_{ub}}\hat{x}(T_{ub}) + \sum_{i=T_{ub}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2] << E[(x(N) - x_d)'(x(N) - x_d)]$. If the modified assumption holds true, again the optimization problem reduces to finding the set of measurement times without regard to update time as expressed in (4.16).

4.3 Solution strategy

Consider a controllable and observable system with no input constraints. Note that this assumption is different from the stabilizability and detectability assumption mentioned earlier for systems discussed in other sections. In order to achieve a desired set of states at the end of an optimization horizon N, measurements need to be taken at least n steps before the end of the optimization horizon, where n is the order of the system. This will give enough time to choose the inputs to drive the states to the desired set. The above scenario occurs when there is complete measurement of the states. However when measurements are partial or if there is measurement noise, states must be estimated. Hence, the measurement time could be different from $N-n$.

Similarly, if the system is completely controllable and if there are no input constraints, theoretically it is possible to select a combination of inputs starting at $N-n$ to drive the states to the desired states. However, if there are input constraints, the decision to update the inputs generally needs to be made at an earlier time instant.

In the current problem, since the inputs are on-off, their selection is made by solving an integer programming problem. This makes the problem even harder due to the NP hard nature of the integer programming problems.

Intuitively, in a single measurement case, one approach is to find the latest time for updating the input sequence with minimal cost and determine the measurement time which minimizes the state covariance at the given update time. However, this proved to be significantly suboptimal. As can be seen from (A.3) the measurement time needs to be selected to minimize the trace of state covariance matrix at the end of the optimization horizon, not at the update time. This outcome will also be illustrated in the next section by case study 2, using a brute force solution.

Step 1:Open loop optimal controller

As noted in section 4.1, the first step in the solution is to find an optimal switching sequence which minimizes the error in the final states with respect to a set of desired states when disturbances are absent. The objective function in this case can be written as,

$$[\sum_{i=0}^{N-1} A^{N-1-i}Bu(i) - x_d]'[\sum_{i=0}^{N-1} A^{N-1-i}Bu(i) - x_d] \tag{4.17}$$

where $u(i) \in 0, 1$ are binary. Techniques for solving this problem were described in [47].

Step 2: Selection of measurement times

Using the result from the previous section, measurements times are optimized under the assumptions of sufficient time for updating being available. The measurement times $T_{l1}, T_{l2}..T_{la}$ can be obtained by solving the optimization problem

$$\min_{T_{l1},T_{l2}..T_{la}} [trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i}$$
$$- \sum_{i=1}^{a} A^{N-T_{li}} P_{T_{li}} C'[C P_{T_{li}} C' + R]^{-1} C P_{T_{li}} A'^{N-T_{li}}\}] \tag{4.18}$$

Step 3: Assumption verication

Case 1: Single update

However, the assumption must be verified for the last measurement time to complete the procedure, so that an update time ($T_u = T_l$ or $T_u = T_l + T_d$, depending on use of delay) can be obtained. If the assumption is satisfied for the last measurement time, only one input update is required. Since the mean and covariance of state at the final measurement time can be estimated easily, the assumption for the single update case can be checked easily using the following procedure.

The mean of the state estimate at T_{la} can be written as,

$$E\{\hat{x}(T_{la})\} = \sum_{i=0}^{T_{la}-1} A^{T_{la}-1-i} B u(i) \tag{4.19}$$

where $u(i)$ in the above equation is the input sequence obtained from the initial open loop optimization. The covariance of the states at T_{la} based on measurements at $T_{l1}, T_{l2}...T_{la}$ can be found using the recurring intermittent Kalman filter equations; covariance of states at T_{la} after measurement ($P_{T_{la}/T_{la}}$) can be written in terms of covariance before measurement ($P_{T_{la}}$) as

$$P_{T_{la}/T_{la}} = P_{T_{la}} - P_{T_{la}} C'[C P_{T_{la}} C' + R]^{-1} C P_{T_{la}} \tag{4.20}$$

and $P_{T_{la}}$ can be written in terms of covariance of states after previous measurement

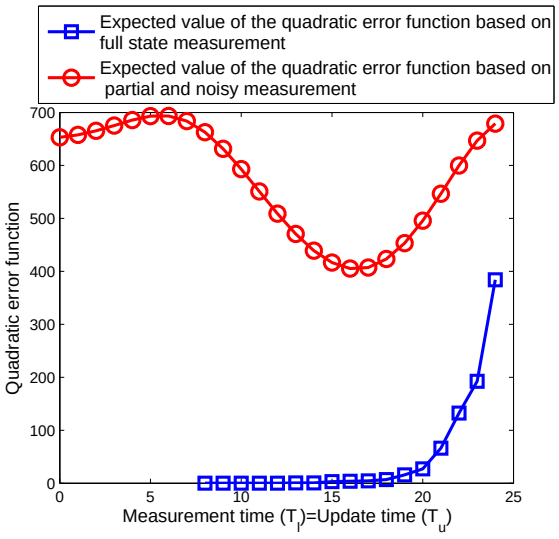

Figure 4.1: Comparison of error functions with full and partial measurements

at PT_{la-1}/T_{la-1} as

$$P_{T_{la}} = A^{T_{la}-T_{la-1}} P_{T_{la-1}} A'^{T_{la}-T_{la-1}} + \sum_{i=0}^{T_{la}-T_{la-1}-1} A^i B Q B' A'^i \tag{4.21}$$

This can be extended to the initial state covariance P_0. A sample set of gaussian pseudo-random variables with the given mean and covariance are generated for $\hat{x}(T_{la})$. For each set of these random variables, corresponding input sequences minimizing $||A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2$ can be determined by solving an integer programming problem. In order to check the assumption, $E(||A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$ is evaluated at the end to compare with (4.18).

Sometimes, it is desirable to determine the latest update time satisfying the assumption since it gives more time to compute the updated sequence, hence allowing the use of a processor with a slower frequency. This is achieved by separating the restrictions posed by on-off constraints from the estimation errors. For that, the error function was evaluated when full state information is available (output matrix C=I

and no noise) and compared with the estimation errors obtained from (4.18). The update can be done at any time that the input constraints don't hinder the ability to reduce the error function to a negligible level, compared to the estimation errors. A result obtained from such a comparison for the parameters given in the case studies is shown in Fig 4.1. In this example, the best measurement time is 16 and update can be done as late as 20 without incurring significant disadvantages to final controller cost.

Case 2: Multiple updates

When the assumption is not satisfied at the final update time as discussed in the last section, updating more than once can reduce the value of $E(||A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$ and hence multiple updates can be useful. In the multiple update case, since integer programming is used to obtain the input sequence after the first update, the probability distribution of $\hat{x}(T_{la})$ cannot be determined analytically. Instead, a Monte Carlo method is used to obtain probability distributions.

The alternate update time selection method in this case begins by updating inputs at all the measurement times in the Monte Carlo simulation and checking the assumption for the final update time as was done in case 1. If the assumption is satisfied, the optimal measurement times obtained from the reduced optimization remains optimal. In this case, there is a possibility of reducing the number of updates which is beneficial in reducing the number of on-line computations. This can be done by using the following algorithm:

To reduce the number of updates, if possible, starting from the final measurement time find the latest time instant T_{u1} for which $E(||A^{N-T_{u1}}\hat{x}(T_{u1}) + \sum_{i=T_{u1}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2) \ll E[(x(N) - x_d)'(x(N) - x_d)]$ using Monte Carlo methods. Let $T_{lm} \leq T_{u1} \leq T_{ln}$. Now evaluate the assumption incorporating the new update again on the final measurement time. The new update reduces the value of $E(||A^{N-T_{la}}\hat{x}(T_{la}) +$

$\sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$; if the reduction is sufficient to satisfy the assumption, one of the new optimal time updates is T_{u1} associated with T_{la}. If the assumption is not satisfied, again the Monte Carlo method needs to be applied to find the latest update time between T_{ln} and T_{la}. Repeating this method until the assumption is satisfied at the final measurement time gives a reduced number of updates if possible.

The separation of optimizing controller and sensor scheduling is not possible if the assumption is not satisfied at the final measurement time even after updating at all the measuring time instants. In this case in order to obtain a set of optimal measurement and update times combined optimization needs to be done using a sufficiently large sample set of random variables via Monte Carlo methods.

Computational advantage

In a single measurement and update case, in order to determine T_l and T_u using brute force optimization, ideally all feasible combinations would be considered, where $T_l \in [0, N-1]$ and $T_u \in [T_l, N-1]$, though if a fixed desirable delay between T_l and T_u is selected beforehand, the number of combinations is reduced. For each of these combinations an integer programming problem of size $N - T_u$ needs to be solved. Moreover, each of these integer programming problems needs to be done over a number of sets (say S total sets) of random variables to determine the combination of T_l and T_u that minimizes the mean of $(\hat{x}(N) - x_d)'(\hat{x}(N) - x_d)$, via Monte Carlo methods. Although this can be done off line, the NP-hard nature of integer programming makes it very expensive for a larger problem size. In the proposed strategy, T_l can be evaluated algebraically without solving any integer programming problem. In order to find T_u, integer programming problems of size $N - T_u$ may only need to be solved for a single T_u, to verify that assumptions on final state reachability are met, though simulations at additional time steps may be performed if delay between T_u and T_l is flexible, and necessary to perform through the entire optimization horizon

if the assumptions are not met. To summarize, by the proposed method, as few as S number of integer programming problems often have to be solved compared to $N \times S$ in the brute force method. In the multiple measurement case, the computational advantage is twofold. By the strategy proposed, it is possible to reduce the number of updates without effecting the overall performance. This will reduce the memory storage and/or computational capacity required on-board. Moreover, the proposed strategy gives a systematic way of finding each update time one-by-one compared to solving a large optimization problem involving all the update times, measurement times and binary inputs.

4.4 Case studies

The following second order system was identified for the micro-robotic leg shown in Fig. 2.3.

$$A = \begin{bmatrix} 0.984 & 9.795 \times 10^{-5} \\ -270.343 & 0.955 \end{bmatrix}, B = \begin{bmatrix} 1.273 \times 10^{-4} \\ 2.527 \end{bmatrix}$$

The sampling time used for discretization was 0.1 mS and input voltage at the on state was set to be 20V. The range of motion for this voltage level was 0.2 radians. For illustrative examples, the objective is set to reach in the neighborhood of (0.1 rad, 0 rad/s) at N=25$^{\text{th}}$ time step. Since the variation in angular velocity is large compared to that of angular displacement, in order to normalize the error, the first state is weighted by the natural frequency (5.3141×10^4 rad/s) of the system. Disturbances are assumed to be normally distributed with standard deviation of $\sqrt{10}$V and initial conditions are also set to be normally distributed with standard deviations 10^{-1} and 10 respectively.

Throughout the simulational study the integer programming problems were solved

using CPLEX.

4.4.1 Full measurement with no noise

First a simple case is considered. The output matrix C is assumed to be the identity matrix and it is assumed that there is no noise present in the measurements. Also, the input sequence is updated at the measurement time itself (no delay). Substituting $C = I$ and $R = 0$ in (4.14) the objective function reduces it to the following. Note that the objective function is the weighted sum of variances in the final states represented in $[\frac{rad}{S}]^2$.

$$\begin{aligned}\min_{T_l}[trace\{&A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i} \\ &- A^{N-T_l} P_{T_l} A'^{N-T_l}\}] \\ &= \min_{T_l}[trace\{\sum_{i=T_l}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i}\}] \end{aligned} \tag{4.22}$$

This equation suggests that the best option is to measure at $N-1$ to capture the effect of disturbances on the last time period. However, since there are no choices left to update inputs at this instant, the assumption that $A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i} B u(i) \approx x_d$ becomes invalid. Hence, measurement needs to be done at an earlier time step when there are enough input choices left to drive the final states close to x_d. This is verified by the simulation result given in Fig. 4.2. The variation in the objective function for a measurement time range of steps 8 to 24 is shown in the figure. In this particular case variation in the objective function for T_l=14 to 21 is small and any time in this range is acceptable. After 21st time step, although the estimation error of the final states is smaller, the input choices available are not sufficient to reduce the final error to the desired range.

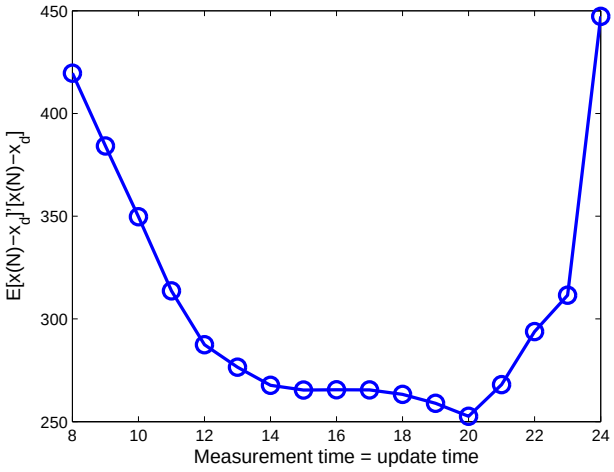

Figure 4.2: Variation in the objective function with respect to measurement time for the full state measurements

4.4.2 Partial measurement with noise

In this case only the first state (angular displacement) is measured and hence the output matrix $C = [1 \quad 0]$. The objective, disturbance levels and initial conditions are set to be the same as the previous case. The measurement noise levels assumed to have a standard deviation of 0.01 radians.

4.4.2.1 Single measurement and single update

First, to illustrate the selection of the best measurement time, (A.4) is plotted for $T_l = [0, N-1]$ in Fig. 4.3. The reduced objective function achieves a minimum for $T_l = 16$. In order to verify the assumption from (4.12), $E(||A^{N-T_l}\hat{x}(T_l) + \sum_{i=T_l}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$ is evaluated at $T_l = 16$ using a Monte Carlo simulation with a sample size of 1000. Error from the update is on the order of 2 (2.346 for the used sample set), very small compared to cost from (A.4) given in Fig. 4.3, thus confirming the validity of the assumption in this case.

In order to further verify this result, a comparison with the results obtained from the brute force method is also done. A complete enumeration of all possible com-

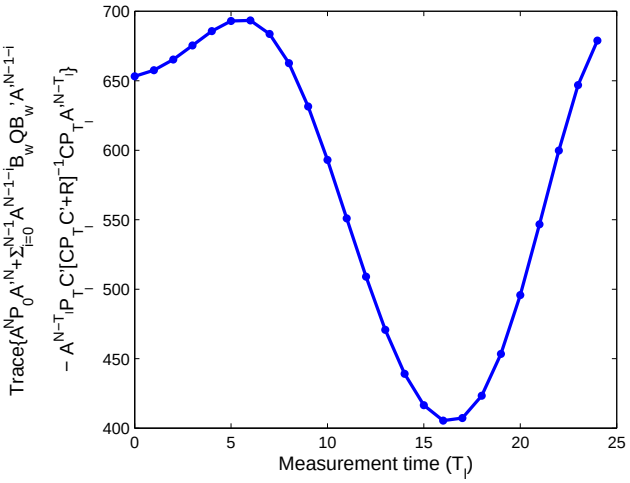

Figure 4.3: Variation of reduced objective function with respect to measurement time for the partial state measurement case

binations of measurement and update times were done for a sample set of size 1000 pseudo-random variables. Results are shown in Fig. 4.4, which confirms that the best measurement time is 16, as well as the fact that update can be done anytime between the 16 and 19th time steps with $< 1\%$ error introduced to the final cost. In other words, up to three step delays for computation are tolerable, and evaluation of this fact using the reduced computation sequence would require Monte Carlo simulations only for time steps 16 to 20.

4.4.2.2 Multiple measurement and single update

When the latest update time is not too late, the quadratic error function can be reduced if more energy is spent on sensors, in other words by taking more measurements. For example, in this case two measurements are taken resulting in a better performance for the same disturbance and noise levels. In order to find the best measurement times T_{l1} and T_{l2}, (4.15) is plotted in Fig. 4.6. It was found that the best measurement times are at 16 and 17 and the corresponding error function is 349.5. As the next step described in the solution strategy the assumption was verified for

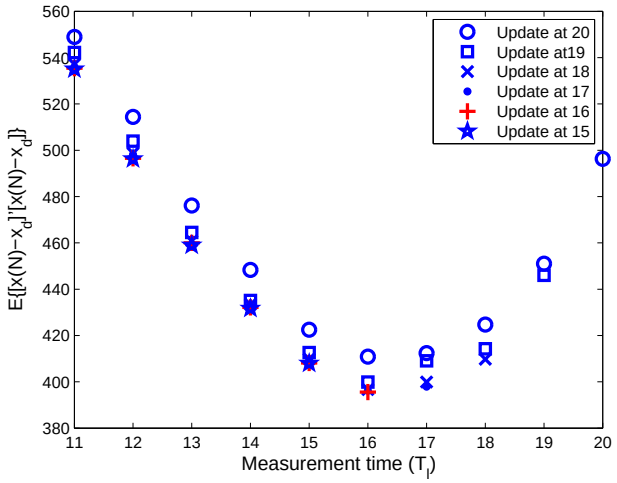

Figure 4.4: Variation in the objective function with respect to measurement time and update using complete enumeration time for the partial state measurement case

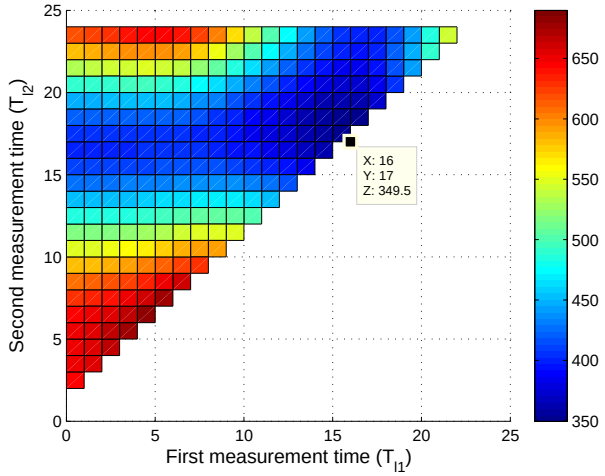

Figure 4.5: Variation in objective function with two measurement times

an update time $T_u = 17$. Also, from the Monte Carlo method the objective function was found to be 350.3 verifying the result.

4.4.2.3 Multiple measurement and multiple update

When disturbance levels are higher, $E(||A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2)$ becomes larger for a relatively late last measurement time and it becomes

beneficial to update more than once. In this scenario the disturbance standard deviation was chosen to be $Q = 50$ and all the other parameters kept the same. Variation in (4.15) is plotted in Fig. 4.5. Measurements are taken twice over the entire optimization horizon. The best measurement times from the reduced optimization given in (4.15) are $T_{l1} = 18$ and $T_{l2} = 24$, however because these times leave only a single time step for the second update to take effect. Measurement times were indexed forward one time step to $T_{l1} = 17$ and $T_{l2} = 23$. The expected value of the objective function from (4.15) is 1400 as shown in the figure. The assumption was checked using the Monte Carlo method updating at $T_{l2} = 23$ only and it was found that $E(||A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2) = 325.8$ resulting in an objective function value of 1802.3. In order to get closer to the expected analytical solution two updates were then done at $T_{u1} = 17$ and $T_{u2} = 23$, resulting in a better value of the objective function at 1565.5. The reason for the difference in this value from the analytical solution can be accounted for by the large error (for this scenario, though small compared to single update scenario) in $E(||A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d||^2) = 138.1$.

The point of this example was to verify that in multiple measurement cases where the assumption that the state is reachable from the final update is not satisfied at the last measurement time, it is beneficial to update more than once. Also, when the assumption is not satisfied the difference between the value calculated by (4.15) and $E(trace\{[x(N) - x_d]'[x(N) - x_d]\})$ is due to the variance of $A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d$ and cross covariance between $A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d$ and (4.15). Since the calculations of these terms involves integer programming, evaluation of them analytically is not generally possible.

The results of all these case studies are compiled in Table 4.1. The number of integer programming problems used to solve each case is also tabulated. Sample time responses are shown in Fig. 4.7 and Fig. 4.8, for the trajectories of the first

and second states, respectively. This set of responses corresponds to one of the worst open-loop performances in Monte Carlo simulations. The solid line represents the nominal performance in the absence of disturbances and initial state uncertainties. As expected, after 25 time steps (at 2.5mS) nominal states reach very close to (0.1 rad, 0 rad/s). The dashed lines show the perturbed open-loop performance, with the deviation in the final states due to random disturbances and initial conditions, using the nominal input sequence. The dotted line represents closed loop performance with a single measurement and the dashed-dot line shows closed loop performance with two measurements and a single update. Both of the closed loop performances show improvement over the open-loop performance. In the open-loop response, the first state reaches only 0.85 rad at the desired time showing an error of 15%. On the other hand, the single measurement case settles at 0.91 rad and two measurements case reaches at 1.04 rad, only 9% and 4% error. For second state, the open-loop response reaches 62.85 rad/s, single measurement arrives at 23.47rad/s and the two measurement case settles at 16.22 rad/s. Closed loop performance did show improvement in nearly all Monte Carlo simulations, though the difference between single and multiple measurements is generally not as significant as for this specific simulation.

4.4.2.4 Implementation of feedback

In order to implement intermittent feedback, a library of on-off updates will be stored on-board. For the case discussed above where measurements and updates are made at 16th time step, up to 2^9 on-off sequences of length 9 needs to be stored in the worst case. However with the chosen parameters given above, when on-off updates are analysed from the monte-carlo simulation, it was found that there are 56 candidate on-off sequences required for the entire measurement space. Hence, the measurement space can be divided into 56 subsets based on the update sequences. When a measurement falls in one of these subsets, corresponding on-off sequence is

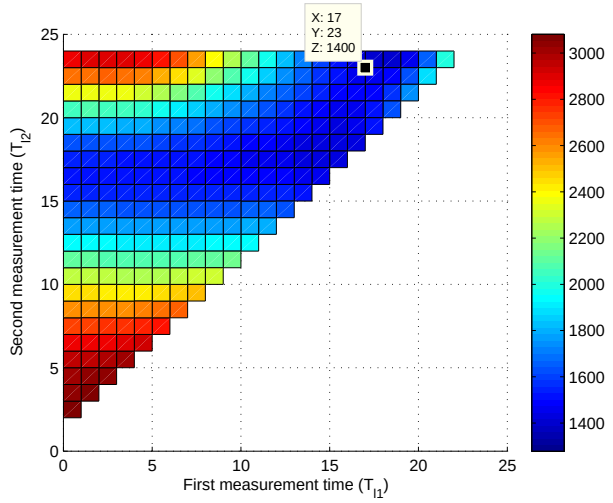

Figure 4.6: Variation in objective function with two measurement times with higher disturbance level

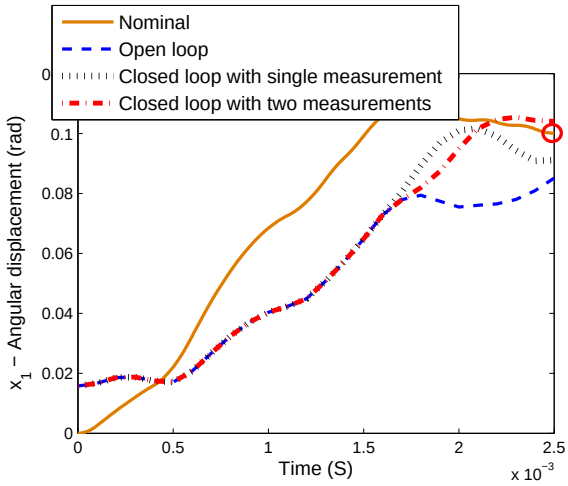

Figure 4.7: Time response comparison of first state showing the effect of feedback

used to update the nominal sequence. The probability of a measurement to be in one of the subranges of the measurement and the corresponding probability is shown in Fig. 4.9.

Similarly, when the measurement is done at 16th time step and update is done at 19th time step, it was expected that the number of sequence updates required for the entire measurement space would be 2^6. However, it was observed to be only 31 from monte-carlo simulations. The probability distribution of the measurement

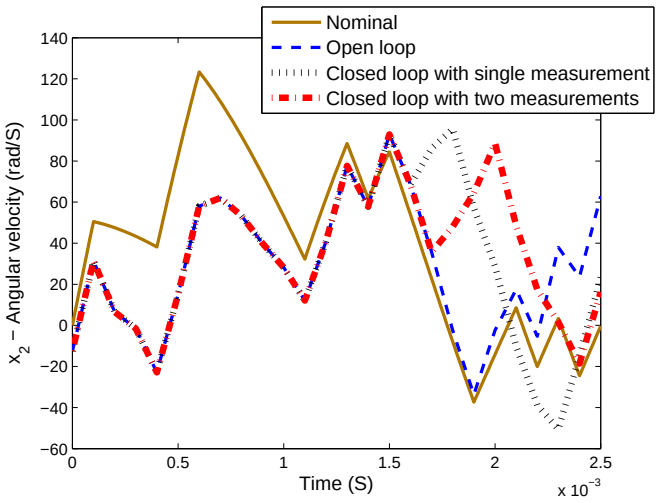

Figure 4.8: Time response comparison of second state showing the effect of feedback

Table 4.1: Compilation of results

	Expected error from reduced optimization	Error shown in Monte Carlo method	Number of integer programming problems solved	
			brute force method	Reduced optimization
Open-loop	695.4	697.9	1	1
Single measurement	405.4	404.1	25001	1001
Two measurements and single update	349.5	350.3	300001	1001
Two measurements and single update (higher disturbance level)	1400	1802.3	300001	1001
Two measurements and two updates (higher disturbance level)	1400	1565.5	324001	2001

space for this case is shown in Fig. 4.10. In cases when on-board memory is limited, only a subset of the update sequences can be stored. In Fig. 4.11, probability of an optimal sequence to be present in the stored set when a certain number of sequences

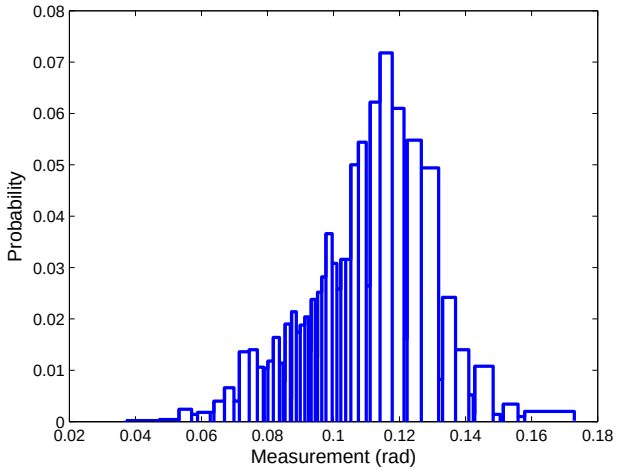

Figure 4.9: Partition of measurement space based on on-off updates made at T=16

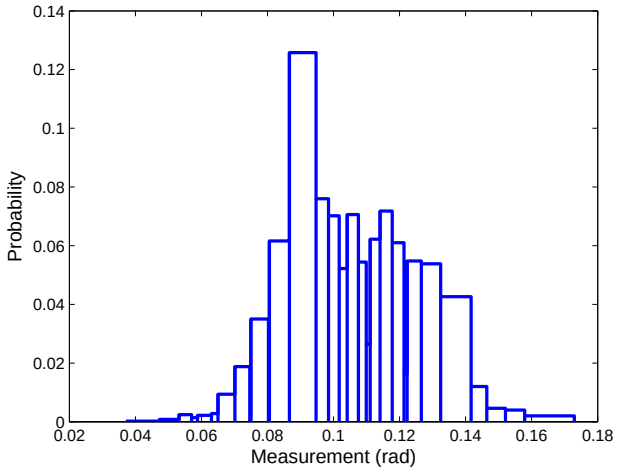

Figure 4.10: Partition of measurement space based on on-off updates made at T=19

are stored is given.

4.5 Combining sensor scheduling and on-off controller optimization

In the previous sections the objective function used for optimization was the error in the final states from a desired target. However, the secondary objective in this study is to minimize the energy consumption of the piezoelectric actuator itself, given

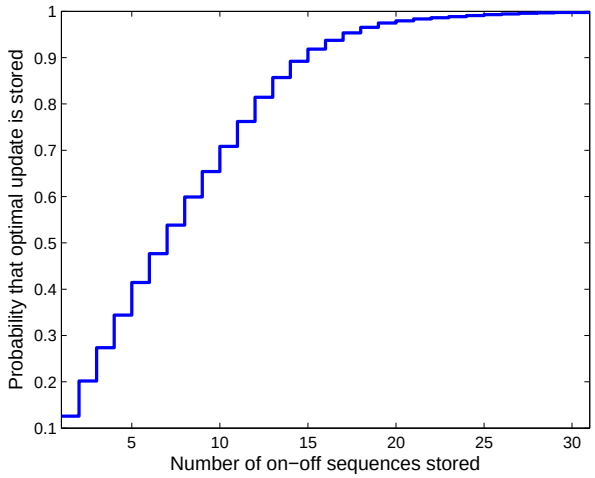

Figure 4.11: Cumulative probability that an optimal update is available for a given measurement vs number of update sequences stored

a small fixed sensor energy usage. In order to achieve this, the solution strategy can be modified. In the modified strategy, when open loop input sequences are calculated in step 1 the following objective function is used instead of the previously used quadratic error function; the new objective function below is the switching cost associated with charging the piezoelectric actuator each time,

$$J_C = \sum_{k=0}^{n} C(u(k)^2 - u(k-1)^2) \tag{4.23}$$

where C is the capacitance of the piezoelectric actuator array (Fig.2.2.b). The constraints for the optimization are the deterministic part of the dynamics equations given in (4.1). A Monte Carlo study was done to document the advantages of this modification on the energy consumption and results are plotted. The variation in the mean error of the final state with respect to the measurement time is plotted in Fig. 4.12. The circled values correspond to the combined optimization and the starred values correspond to the sensor scheduling alone where the sole objective is to minimize the mean error function. All the results correspond to an update time

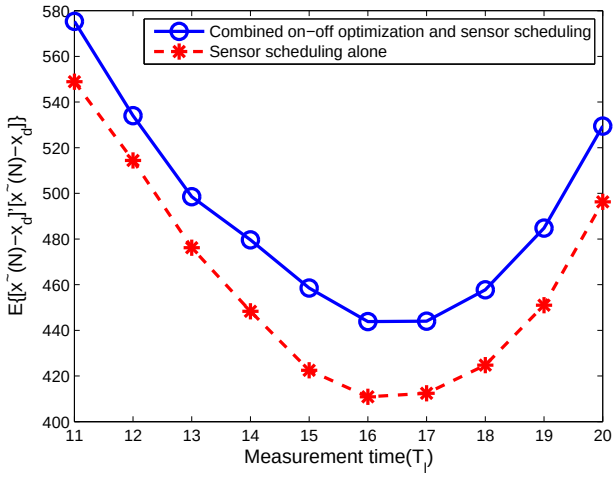

Figure 4.12: Variation in the objective function with respect to measurement time for combined sensor scheduling and energy optimal on-off controller

$T_u = 20$ keeping all the other parameters similar to the second case study. The average number of switches corresponding to the energy consumption with respect to measurement time is calculated. From the energy consumption calculation it was seen that, on an average it takes 2 switches for the combined optimization, where as it takes 6 switches for the sensor scheduling alone. Hence, although there is a small increase in the error values, the reduction in the energy consumption is substantial with the modified cost function.

4.6 Conclusion

Use of an energy consuming sensor sparingly with an on-off controller is studied here, where the sensor is used only for a limited number of sampling instants over a finite optimization horizon. Using state estimates from these measurements, the optimal open-loop on-off input sequence driving states of the system to desired final values is updated at one or more update times following measurements. A three-step, near-optimal strategy for selecting measurement and update times is proposed. First, the open-loop input sequence without any measurements is obtained. Second,

measurement times are selected independent of update times under the assumption that updating inputs after the measurement time or some desired delay later will approximately cancel error in states projected from the measurements. Third, a Monte Carlo simulation of response to stochastic variation for the proposed update time is performed, to verify that assumption's validity. Procedures for adjusting measurement and update times if the assumption fails, for eliminating unnecessary updates having negligible impact on performance, and for incorporating actuator power consumption are proposed. Several case studies for a micro-robotic leg joint are examined in simulation to explore proposed controller design effectiveness.

The procedure described has potential usefulness for very low power control systems, where analog inputs and sensor measurements are expensive from an energy standpoint, and rapid computation or data lookup are not necessarily available. However, it is worth noting that the proposed results, even accounting for actuator energy consumption, are not guaranteed to be optimal for overall energy consumption. Rather, limits are effectively placed on energy use for sensing and on computation time, and near optimal energy usage just for actuation is obtained under these constraints. A second limitation is that the current approach focuses on gaussian disturbances at each time step, which is a common approach to controller design and can be adapted to some other situations, but may not always best describe disturbances experienced by micro-systems for which this control approach is primarily intended. Meanwhile, the approach provides decent computational benefits over baseline optimization procedures, such as full evaluation by Monte Carlo methods. On-line computation or data lookup in the low-power system is also kept to a low level. Possible future enhancements to the control approach include accounting for system model variation and evaluating minimal total energy requirements, including sensor and computation costs, for various performance levels in terms of final state error.

CHAPTER V

Hybrid energy system for switching control of micro-robotic actuation

As described in the Introduction, at present levels of development, battery technology remains the most suitable primary power source for autonomous micro-robotics applications. Most current micro-battery technologies produce energy below 5V level and stacking them in series to produce a higher voltage causes a significant reduction in energy density [38]. Thus, since piezoelectric actuators are best operated around 20 - 30V, a voltage converter is required to step-up the battery voltage. The addition of an energy harvesting module to the power system has the potential to make a robot completely autonomous (with indefinite lifetime, at least theoretically) or to extend life time or the time interval between the recharging of a battery in a suitable environment. Likewise as mentioned in the introduction, the most suitable current choice for harvesting is a photovoltaic cell. Because of the size, weight and fabrication constraints imposed by MEMS, a simple power system architecture is proposed in Fig. 5.1. The system is explained in detail in section 5.1.

A comparison of several MEMS energy storage devices are given in Fig 5.2 [21]. Almost universally, power-dense sources show less energy density and vice versa. Nuclear batteries and fuel cells have very high energy density but can be used to drive very light loads only because of the low power-density. On the other hand,

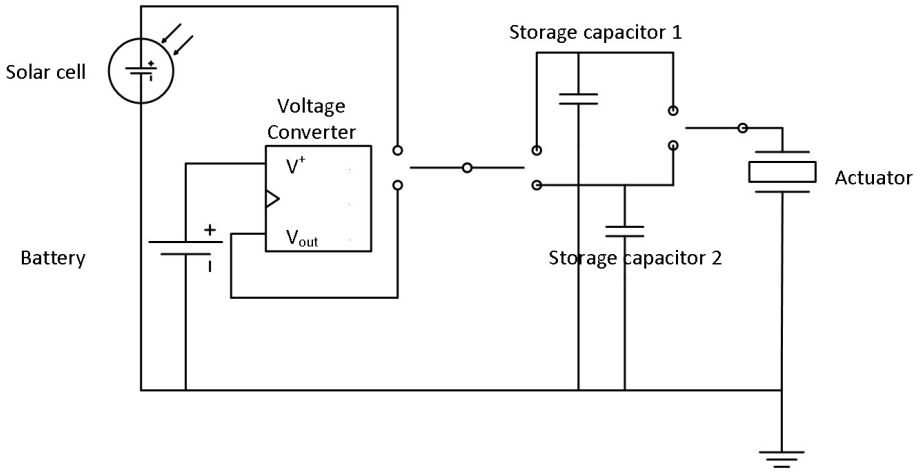

Figure 5.1: Proposed hybrid power system for micro-robot

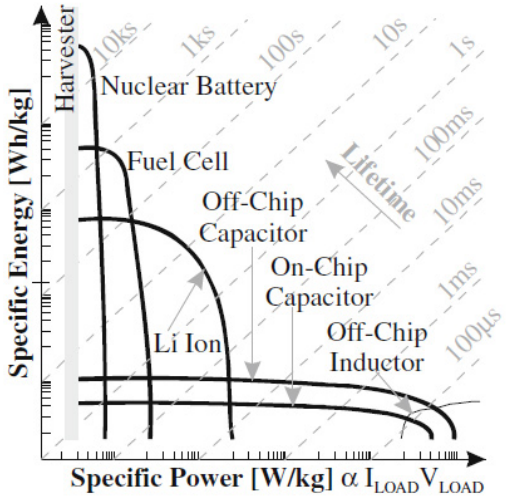

Figure 5.2: Power-Energy performance of various energy sources

capacitors can produce instantaneous power, but only momentarily because of low energy density. Li-ion batteries show a balanced performance between both the sides and hence they are projected as the primary storage device for the micro-robot and together with a storage capacitor, the configuration is expected to meet the power and energy requirement.

This chapter is organized as follows: section 5.1 discusses the basic architecture of the circuits used for the power system; section 5.2 describes a simulational study conducted to convert gait parameters to the number of actuator charging events

required; and section 5.3 examines which power source to use based on the area of solar cell, light intensity and actuator requirements.

5.1 Power system architecture

The proposed configuration of power sources is given in Fig. 5.1. It consists of a solar cell, micro-battery, voltage converter circuit, a pair of storage capacitors, an array of piezoelectric actuators and pair of switches. The first switch determines which power source charges which storage capacitor and the second switch determines which storage capacitor connects to the actuator. The switches are turned on such a way that,when one storage capacitor is connected to power sources the other storage capacitor charges the actuator.

The on-off sequence of the switch connecting a storage capacitor to actuators is optimized for minimum energy and state constraints as discussed in chapter 2 and it involves integer programming. Since processing power on-board is not sufficient to do integer programming, these sequences are predetermined for a certain switching voltage level. Hence, the variation in the voltage level on the storage capacitor connecting to the actuator need to be small and should be in a known range for accurate actuator movements. Storage capacitors are charged when they reach $V_{s_{min}}$ to $V_{s_{max}}$ either by connecting to the solar cell or by connecting to the voltage converter attached to the battery.

Typical variation in current and power generated by a small solar cell with voltage for a given light intensity is shown in Fig. 5.3. As can be seen, for a given light intensity there is an optimum voltage and current level at which solar cell generates maximum power. Hence in the design phase, a set of solar cells has to be chosen in such a way that the maximum power point (MPP) of their combination at the given nominal light intensity matches with the desired storage capacitor voltage level. This will result in the solar cell operating at its MPP most of the time. A voltage converter can be used

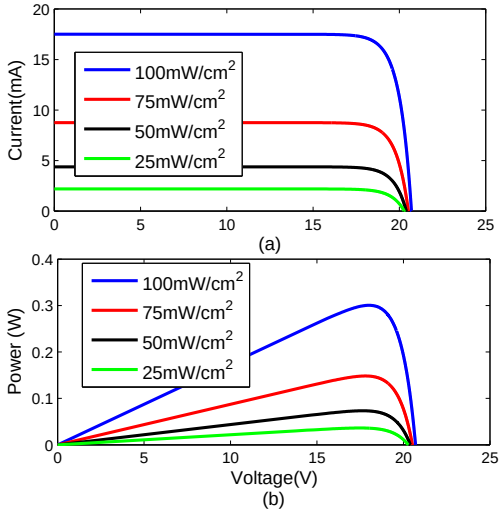

Figure 5.3: Typical solar cell characteristics

in between the solar cells and capacitors but the associated efficiency is typically very poor with small capacitive loads. Also, in order to maintain solar cell at MPP and do the voltage conversion, impedance matching circuit is necessary. Hence, in order to simplify the circuit the solar cell is connected directly to the storage capacitor whose voltage is maintained very close to MPP. The voltage converter on the battery side is a standard boost converter whose components are optimized for a capacitive load by simulational and experimental study.

Performance of the power source is evaluated by the number of times it can charge the actuator per second, which is referred to as the available switching frequency. Gait performance required by the robot can be characterized by the desired final state of the actuator at the desired time, tolerance of the final states, sampling time of the controller and desired step frequency. The above mentioned gait requirements can be converted again to a required switching frequency. Strategies for meeting these requirements by using a solar cell array or battery are discussed in the following sections.

5.2 Actuator requirements

Actuator requirement are evaluated under the assumption that the optimal gait sequence of the robot can be split in to simple movements of the actuator, which can be modeled as state space constraints as discussed in earlier chapters. The parameters involved in these constraints are the final desired state at the desired time (x_d and T_f), the tolerance on the final states (ϵ) and sampling time (T) of the on-off input. Variation in switching frequency required with each of these parameters was studied.

For a set of values of $x_d = (\theta_d, \dot{\theta_d})$, different desired time (T_f), tolerance levels (ϵ) and sampling time (T) values were used to find the minimum number of switchings of an actuator required to satisfy the state constraints. T_f gives a measure of speed of operation and the minimum number of switches corresponds to the required switching frequency mentioned earlier. It was observed that for a given tolerance, required switching frequency is inversely proportional to T_f (or directly proportional to speed of operation). In general, having a longer sampling time results in lower switching frequency (hence lower power consumption). To achieve tighter tolerance levels, it is necessary to have more time steps ($N = \frac{T_f}{T}$) in the optimization horizon. Hence, for fast and accurate operation, it is necessary to have a shorter sampling time which in turn requires higher processing capacity on-board, which is associated with a higher energy consumption for computation.

One of the interests of the current case study is to find reachable final angles (θ_d) with very small final velocity for a combination of large sampling time, small optimization horizon and minimal actuator energy consumption. A comparison of two cases is shown in Fig. 5.4. The first case is that of a higher sample rate with a sampling time of $T = 1 \times 10^{-4}s$ over a number of time steps $N = 40$. Secondly, keeping the same final time $T_f = 4 \times 10^{-3}$ a lower sample rate case with a sampling time of $T = 4 \times 10^{-4}s$ is considered. In both cases final angles that can be reached with a single switching are considered. The tolerance level in the figure is normalized

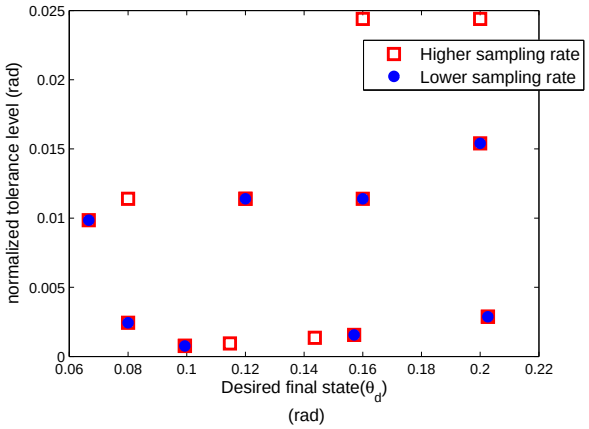

Figure 5.4: Variation in tolerance level with final angle for two sample rates

for both angular displacement and velocity states by dividing tolerance in the final angular velocity by a factor of natural frequency of the system. As can be observed, a controller acting at a higher sampling rate can reach all the points that are reachable by the lower sampling rate. In addition, there are new reachable points in the state space because of the increased number of switching options available with the higher sampling rate. The downside of a higher sampling rate is the higher processing capacity required on-board and associated power consumption.

Another study was conducted to understand the variation in energy consumption (measured in terms of number of actuator switches required) with different sampling rates for different final leg angles. Since the maximum angle that can be reached with a 20V supply is approximately 0.2 radians, a set of four angles $(0.2rad, 0.16rad, 0.12rad, 0.08rad)$ were chosen as the desired final angles (θ_d). Two different tolerance levels were chosen for a final time of $T_f = 80 \times 10^{-4}s$. Five different sampling rates were chosen with number of samples $N = 20, 40, 80, 160$ and 320. Results are shown in Fig. 5.5. The top plot shows the variations for a looser tolerance level compared to a tighter tolerance in the bottom plot. Certain states with tighter tolerance are not reachable with lower sampling rates (for example $x_1(T_f) = 0.16rad$) and certain states $x_1(T_f) = 0.2rad$ are not reachable with lower sample rates in both

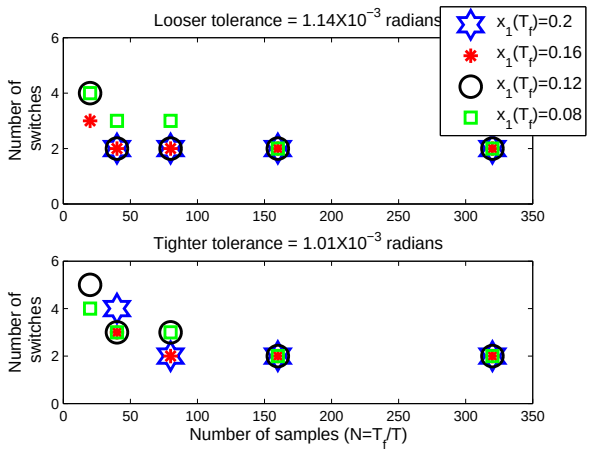

Figure 5.5: Energy consumption vs sample rate for fixed tolerance levels

the cases. More states are reachable at higher sampling rate and they require less energy from the actuator point of view as well. But, again this will require a faster processor on-board similar to the last case.

5.3 Power source characteristics

In the architecture used, selection of a power source means whether to use the solar cell or battery. From the preliminary studies it was found that the solar cell can charge storage capacitor more efficiently than the battery when the storage capacitor voltage is maintained around MPP. Hence, whenever the solar cell can generate enough power to meet actuator requirements it is used directly. Other times the battery is used through the voltage converter to replenish the storage capacitor.

For a slow walking gate under indoor operation with sufficient light (over 200LUX), the solar cell can power the robot. Parameters of the solar cell array are chosen from an off-the-shelf solar cell and its total area is assumed to be 17mm $\times$ 17mm. In order to maintain an output voltage around 20V under indoor operation, there should be a total of 32 cells arranged in series with a total area mentioned above. Switching losses upon charging and discharging of the actuator are assumed to be 600nJ, based on the measurements given in chapter 2. With the solar cells the strategy was to

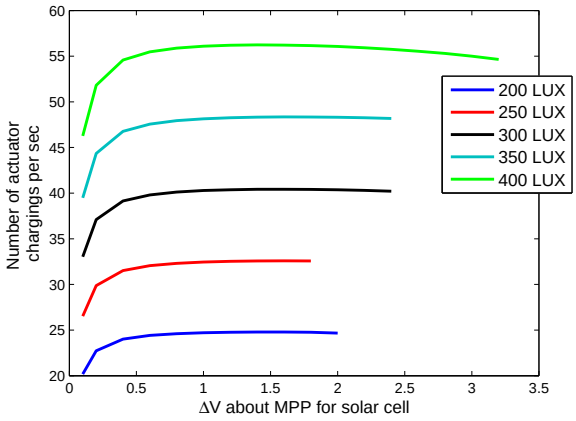

Figure 5.6: Variation in maximum switching frequency with the changes in operating voltage about MPP of solar cell

use them directly to charge the storage capacitor from $V_{s_{min}}$ to $V_{s_{max}}$. For indoor operation with a change in light intensity from 200LUX to 400LUX the maximum power point changed from 18.88V at $0.804\mu A$ to 19.64V at $1.75\mu A$. For each of these light intensities $V_{s_{min}}$ and $V_{s_{max}}$ are chosen in such a way that the solar cell operates around the MPP at all times. For example, for a light intensity of 200LUX, MPP is 18.88V at $0.804\mu A$ and $V_{s_{min}}$ and $V_{s_{max}}$ are chosen to be 18.2V and 19.6V to provide a maximum switching frequency of 24.78switches per second, according to Virtual Test Bed simulations [51]. The variation in the maximum number of switches that the solar cell can provide for a given deviation about MPP at various light intensities is given in Fig. 5.6. In the initial part of the plot the solar cell operates very closely around MPP but because of high switching frequency (hence high switching loses), the number of switches it can provide to the actuator is small. On the other hand a higher ΔV about MPP results in less frequent switching to charge the storage capacitor but solar cell operation is inefficient. Also, it was found that the energy gain from using a feedback controller to change operating point of the solar cell with a change in light intensity is moderate.

Fig. 5.7 shows the maximum number of times an actuator can be turned on in a

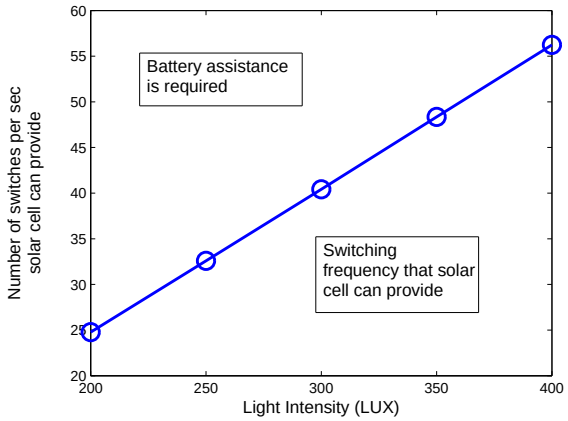

Figure 5.7: Variation in maximum switching frequency with Light intensity

second using a solar cell alone at various light intensities. If the operation frequency of the actuator requires switching frequency of more than this, the battery has to used for actuation. Hence, this plot shows when only the solar cell can be used and when the battery needs to be turned on to charge the actuator.

5.4 Discussion

As mentioned earlier, a battery is needed to assist faster walking gates of the micro-robot. In order to determine the switching frequencies a battery can provide, a battery with charge characteristics based on an off-the-shelf Cymbet micro-battery (EnerChip^{TM} CBC050) is modeled in Virtual Test Bed software. To charge a storage capacitor of capacitance $1\mu F$ from 18V to 19V through a voltage converter, a fully charged battery takes approximately 0.24 milliseconds. With the given storage capacitors, an actuator can be charged 17 times while the storage capacitor voltage drops from 19V to 18V hence providing a switching frequency of over 70800Hz. The voltage converter with optimized components has an efficiency of approximately 20%; unfortunately, with the existing boost converter topologies in MEMS for low-power applications, conversion efficiency is very poor. One of the tasks of future work is to explore newer voltage converter architectures that can be miniaturized.

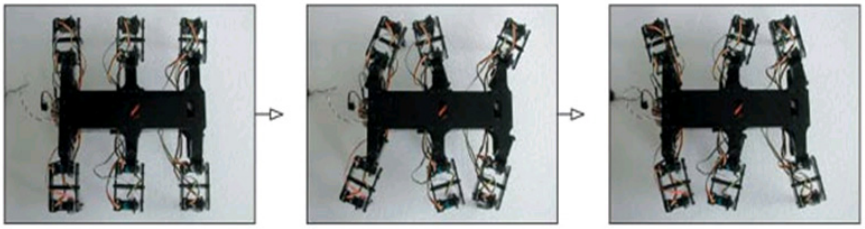

Figure 5.8: A typical tripod walking gait sequence for straight line walking of a hexapod robot

The objective of this thesis was to devise efficient control strategies for simple rotations of micro-robotic legs driven by piezoelectric actuators. These simple motions can be combined to form efficient walking gaits. For example, for a hexapod design, a tripod gait can be achieved by simple and synchronized rotations of the legs as shown in Fig. 5.8[52]. In the figure, the robot is shown walking towards right side. At any time, the robot moves three of its legs (two outer legs on one side and the middle one on the other side) and the other three supports the weight and keep the robot balanced. By alternating the moving and supporting legs the hexapod robot can move forward.

As explained earlier in this chapter, the key parameters deciding the controller for walking gait are the final state (x_d), the tolerances on final states (ϵ) and the final time (T_f) and optimal on-off sequence for the particular gait also depends on the sampling time (T). The first step in expanding the strategies described in this dissertation for the entire robot is to develop lumped models for the dynamics of the robot with the selected legs moving at the same time (legs are chosen such that the combination forms the gait sequence). Using this model, an optimization study can be done to decide an optimal combination of gait parameters for achieving certain walking gait. Using these parameters, an optimal on-off control sequence can be generated for each leg separately.

For open-loop operation, a certain number of optimal on-off/charge recovery control sequences for each leg can be stored in the on-board memory for different walking sequences such as fast or slow straight line walking, turning etc. Also, provided the robot power budget allows, a piezoelectric or capacitive position sensor can be attached to each leg. This sensor can be used to measure angular rotations of each leg intermittently as discussed in the 4th chapter. As mentioned in chapter 4, the intermittent feedback strategy can be implemented by storing a certain number of additional on-off update sequences on-board. From the stored sequences, suitable updates can be chosen based on the intermittent measurements.

According to previous estimates for thin-film piezoelectric walking robots [39], a hexapod design with the current actuator design can carry a payload of 20mg. Assuming half of the payload is allowed for battery, a thin-film Li-ion battery of weight 10mg with an energy density of 200W.hr/kg [21] can power over 10^6 chargings of the actuator. By choosing optimal final angles mentioned earlier, this can result in over 300,000 steps. The hexapod designs can move 1mm per step and hence the robot is estimated to be capable of walking over 300m from the battery payload using the simple on-off control strategy.

CHAPTER VI

Conclusion

6.1 Summary of dissertation

This dissertation presents a systematic development of optimal control strategies, from an optimal open-loop controller to a closed-loop controller with an intermittent sensor, for a piezoelectric micro-robot. The core issue to be addressed was excessive energy consumption of traditional control schemes while operating piezoelectric actuators that act as capacitative loads.

Energy consumption is proportional to switching frequency in piezoelectric actuators and traditional PWM-based controllers operate at very high switching frequency. Hence, in the second chapter a more energy efficient on-off controller as well as methods to optimize it was proposed. The optimization algorithm minimizes the number of times the piezoelectric actuator is turned on, subject to certain state constraints. A penalty-function-based method and a branch and bound based algorithm implemented in CPLEX were discussed. The integer programming methods were verified against brute force methods. In addition, on-off controllers were compared to a traditional LQ-controllers to interpret behavior and energy consumption. A method to make these on-off controllers more robust was also proposed. On-off control sequences were experimentally verified on a prototype micro-actuator as well as on a macro-scale system.

In the third chapter, energy savings obtained using a partial charge recovery circuit were discussed. While full charge recovery circuits, requiring heavy inductors, had been proposed by earlier researchers, in this chapter multiple benefits of using a lighter inductor were discussed along with trade-offs associated with inductor weights. The on-off control optimization method was modified for additional state constraints associated with the charge recovery circuit. It was found that in certain cases energy usage can be reduced up to two thirds by using moderately heavy inductors. Partial charge recovery has the additional benefits of having intermediate voltage levels between the minimum and maximum voltage levels of the on-off controller, which allows the target states to be achieved with tighter tolerance.

As an extension to the open-loop controllers, in the fourth chapter a position sensor was added to the control system. Strategies for efficient use of this energy consuming sensor were discussed. A three step strategy for selecting measurement times, selecting input update times and calculating modified input sequence was proposed. Several case studies illustrating the method and showing its advantages are also discussed. Computational advantage of the proposed strategy was tabulated along with a comparison from a full simulation result and proposed reduced optimization.

In the fifth chapter, a hybrid power system consisting of a photovoltaic cell, a commercially available micro-battery and voltage converter was proposed for the use on the micro-robot. The power system architecture was designed based on limited resources allowed on-board based on weight and size constraints, and case study results providing insight into application micro-robotics are given.

6.2 Summary of contributions

1. An alternative control strategy for piezoelectric micro-actuators in place of traditional op-amp based controllers and PWM-based controllers is proposed. An integer programming based approach was shown to optimize the controllers min-

imizing the energy usage and satisfying the given state constraints. A penalty function based heuristic optimization method with a convex objective function was also developed to solve the integer programming problem.

2. A partial charge recovery strategy is proposed for saving energy when the actuator is discharged. Compared to a full charge recovery where almost all the energy is recovered using heavy inductors, in the proposed method, a lighter inductor is used saving some of the drained energy. In addition to advantage in weight saving, this method opens opportunities for additional intermediate voltage levels between the maximum and minimum levels allowed in the on-off controller. Also, an optimization procedure was developed for minimizing the energy intake from external energy source.

3. A strategy for using an energy consuming sensor sparingly for an on-off controller is developed. A three step strategy was proposed which can identify measurement times and input update times, by solving N (where N is the number of time steps in the optimization horizon) order fewer integer programming problems compared to a brute force method under certain assumptions.

4. A simple hybrid power system architecture consisting of a photovoltaic cell, an off-the-shelf micro-battery and voltage converter was proposed for the autonomous operation of piezoelectric micro-robot.

6.3 Future works

6.3.1 Extension to higher dimensional model

For optimization purposes, the actuator was modeled as second order linear system in the thesis. This can be extended to a higher dimensional model to incorporate more accurate dynamic model. Also, the effect of inclusion of non-linearities could

be studied. Optimization is done for simple leg rotations which can be combined for complex gait sequences. Its also possible to do optimization from a supervisory control point of view for an entire gait sequence without specifying the simpler rotational constraints incorporating complex dynamics models developed in the lab. A study can be conducted to find a gain in accuracy with numerical effort required to achieve these goals.

6.3.2 Voltage converters for capacitive loads

Traditional voltage converter architectures are designed for resistive loads and components are optimized for load characteristics. However, piezoelectric actuators are capacitive in nature. Voltage converters used in this study were traditional boost converters with components sized to suit certain capacitive loads. Hence, a study can be conducted to explore for a voltage converter architecture which works better with capacitive load. This can be beneficial in other areas such as renewable energy sources where super capacitors are used as energy storage devices.

6.3.3 Use of reconfigurable circuit elements

Use of reconfigurable components in the power circuitry can be used to extract maximum out of components already on-board without adding additional weight. For example, in the current design there are inductors in the charge recovery circuit as well as in the voltage converter circuit. If both circuitry as well as other circuit components can be re-designed reducing number of these heavy components it can be beneficial. This can be done using switches and the optimization methods developed in this research can be utilized to optimize the resulting switching circuits.

6.3.4 List of publications

1. Biju Edamana and Kenn Oldham, A Near-Optimal Sensor Scheduling Strategy for an On-Off Controller with an Expensive Sensor, under review with IEEE/ASME Transactions on mechatronics.

2. Biju Edamana and Kenn Oldham, Optimal Low-Power Piezoelectric Actuator Control with Charge Recovery for a Micro-Robotic Leg , accepted for publication by IEEE/ASME Transactions on mechatronics.

3. Biju Edamana and Kenn Oldham, Optimal Charge Recovery of Thin-Film Piezoelectric Actuators for an Autonomous Micro-Robot, 2011 American Control Conference, San Francisco, California, USA June 29 - July 1, 2011.

4. Kenn Oldham, Biju Edamana, Bongsu Hahn, Coordinated Voltage Conversion and Low-Power Micro-actuator Switching 4th Annual Dynamic Systems and Control Conference, October 31 - November 2, 2011, Arlington, VA, USA.

5. Biju Edamana, Bongsu Hahn, Jeffrey S. Pulskamp, Ronald G. Polcawich, and Kenn Oldham Modeling and Optimal Low-Power On-Off Control of Thin-Film Piezoelectric Rotational Actuators IEEE/ASME Transactions on mechatronics, volume: 16, Issue: 5, year: 2011.

6. Biju Edamana and Kenn Oldham, An Optimal On-Off Controller with Switching Costs using Non-linear Binary Programming, Proc. 2009 American Control Conference, Hyatt Regency Riverfront, St. Louis, MO, USA, June 10-12, 2009.

APPENDICES

APPENDIX A

Single measurement case for intermittent sampling

Derivation of (4.14) under the assumption (4.12) is described in this section. Tracking system states at the final time, $T(N)$, back to the measurement time, T_l, (4.13) can be described using Kalman filter gain K_l as in (4.7),

$$
\begin{aligned}
x(N) &= x_d + A^{N-T_l}(x(T_l) - \hat{x}_{T_l/T_l}) + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i) \\
&= x_d + A^{N-T_l}[x(T_l) - (\hat{x}(T_l) + K_l(Cx(T_l) - \\
&\qquad C\hat{x}(T_l) + v(T_l)))] + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i) \\
&= x_d + A^{N-T_l}[(I - K_l C)(x(T_l) - \hat{x}(T_l))] \\
&\qquad - A^{N-T_l} K_l v(T_l) + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i)
\end{aligned}
\tag{A.1}
$$

The error in estimation of states $x(T_l) - \hat{x}(T_l)$ is due to the randomness of the initial states and the normal disturbances until $T_l - 1$. Thus (A.1) can be rewritten as,

$$\begin{aligned} x(N) = x_d + A^{N-T_l}[(I - K_l C)(A^{T_l}x(0) + \\ \sum_{i=0}^{T_l-1} A^{T_l-1-i}B_w w(i))] - A^{N-T_l}K_l v(T_l) \\ + \sum_{i=T_l}^{N-1} A^{N-1-i}B_w w(i) \end{aligned} \tag{A.2}$$

Hence $E\{[x(N) - x_d]'[x(N) - x_d]\}$ can be written as,

$$\begin{aligned} & E\{[x(N) - x_d]'[x(N) - x_d]\} = \\ & trace\{A^{N-T_l}(I - K_l C)P_{T_l}(I - K_l C)'A'^{N-T_l} \\ & + A^{N-T_l}K_l R K_l' A'^{N-T_l} + \sum_{i=T_l}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i}\} \end{aligned} \tag{A.3}$$

$$\begin{aligned} & = trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i} \\ & - A^{N-T_l}P_{T_l}C'[CP_{T_l}C' + R]^{-1}CP_{T_l}A'^{N-T_l}\} \end{aligned} \tag{A.4}$$

APPENDIX B

Multiple measurements case for intermittent sampling

This section describes the derivation of (4.15) for two or more measurements. The final state $x(N)$ can be written based on the states and the expected values of states at update time $x(T_u)$ as,

$$\begin{aligned} x(N) &= A^{N-T_u}\hat{x}(T_u) + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) \\ &\quad + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i) \end{aligned} \tag{B.1}$$

Under the assumption (4.12), tracking the states back to the measuring time T_{l2}, the equation can be modified using Kalman filter gain K_{l2} as,

$$\begin{aligned} x(N) &= x_d + A^{N-T_{l2}}[x(T_{l2}) - (\hat{x}(T_{l2}) + K_{l2}(Cx(T_{l2}) - \\ &\qquad C\hat{x}(T_{l2}) + v(T_{l2})))] + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i) \\ &= x_d + A^{N-T_{l2}}[(I - K_{l2}C)(x(T_{l2}) - \hat{x}(T_{l2}))] \\ &\quad - A^{N-T_{l2}}K_{l2}v(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i) \end{aligned} \tag{B.2}$$

This can be traced back to the states at first measurement time T_{l1} and written as a sum of independent random variables,

$$
\begin{aligned}
x(N) = x_d + & \\
& [A^{N-Tl2} - A^{N-Tl2}K_{l2}C][A^{T_{l2}-T_{l1}} - A^{T_{l2}-T_{l1}}K_{l1}C] \\
& (A^{T_{l1}}x(0) + \sum_{i=0}^{T_{l1}-1} A^{T_{l1}-1-i}B_w w(i)) \\
& - [A^{N-T_{l2}} - A^{N-T_{l2}}K_{l2}C]A^{T_{l2}-T_{l1}}K_{l1}v(T_{l1}) \\
& + [A^{N-Tl2} - A^{N-Tl2}K_{l2}C] \sum_{i=T_{l1}}^{T_{l2}-1} A^{T_{l2}-1-i}B_w w(i) \\
& - A^{N-T_{l2}}K_{l2}v(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i)
\end{aligned}
\tag{B.3}
$$

Hence $E\{[x(N) - x_d]'[x(N) - x_d]\}$ can be written as,

$$
\begin{aligned}
& E\{[x(N) - x_d]'[x(N) - x_d]\} = \\
& trace\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i} \\
& - A^{N-T_{l1}}P_{T_{l1}}C'[CP_{T_{l1}}C' + R]^{-1}CP_{T_{l1}}A'^{N-T_{l1}}\} \\
& - A^{N-T_{l2}}P_{T_{l2}}C'[CP_{T_{l2}}C' + R]^{-1}CP_{T_{l2}}A'^{N-T_{l2}}\}
\end{aligned}
\tag{B.4}
$$

BIBLIOGRAPHY

[1] http://marsrovers.jpl.nasa.gov/gallery/press/opportunity/20120117a.html.

[2] T. Yasuda, I. Shimoyama, and H. Miura. Microrobot actuated by a vibration energy field. *Sensors and Actuators A: Physical*, 43(1-3):366370, May 1994.

[3] P. E. Kladitis, V. M. Bright, K. F. Harsh, and Y. C. Lee. Prototype microrobots for micropositioning in a manufacturing process and micro unmanned vehicles. *Proc. 12th IEEE Int. Conf. MEMS*, page 570575, 1999.

[4] T. M. Ebefors, U. Johan, E. Klvesten, and G. Stemme. A walking silicon microrobot. *proceedings of 10th International Conference on Solid-State Sensors and Actuators, Sendai, Japan*, pages 1202–1205, 1999.

[5] S. Hollar, A. Flynn, C. Bellow, and K.S.J. Pister. Solar powered 10 mg silicon robot. *proceedings of MEMS 2003, Kyoto, Japan*, pages 706–711, 2003.

[6] K. J. Son, V. Kartik, J. A. Wickert, and M. Sitti. A piezoelectric unimorph actuator based precision positioning miniature walking robot,. *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, pages 176–182, 2005.

[7] G. Caprari and R. Siegwart. Mobile micro-robots ready to use: Alice. *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3295–3300, 2005.

[8] G. Caprari, P. Balmer, R. Piguet, and R. Siegwart. The autonomous micro robot alice: a platform for scientific and commercial applications. *Micromechatronics and Human Science, 1998. MHS '98. Proceedings of the 1998 International Symposium on*, pages 231 – 235, 1998.

[9] B.R. Donald, G.G. Levy, C.D. McGray, I. Papotry, and D. Rus. An untethered electrostatic, globally controllable mems micro-robot. *Journal of Microelectromechanical Systems*, 15(1):1–15, February 2006.

[10] S. Bergbreiter and K.S.J. Pister. Design of an autonomous jumping microrobot. *Robotics and Automation, 2007 IEEE International Conference on*, pages 447 – 453, 2007.

[11] K. Vollmers, D. R. Frutiger, B. E. Kratochvil, and B. J. Nelson. Wireless resonant magnetic microactuator for untethered mobile microrobots. *Applied Physics Letter*, 92(14):144103, April 2008.

[12] C. Pawashe, S. Floyd, and M. Sitti. Multiple magnetic microrobot control using electrostatic anchoring. *Applied Physics Letter*, 94(16):164108, April 2009.

[13] E.Y. Erdem, Yu-Ming Chen, M. Mohebbi, J.W. Suh, G. Kovacs, R.B. Darling, and K.F. Bohringer. Thermally actuated omnidirectional walking microrobot. *Microelectromechanical Systems, Journal of*, 19(3):433 – 442, 2010.

[14] M. Sitti. Piezoelectrically actuated four-bar mechanism with two flexible links for micromechanical insect thorax. *IEEE/ASME Transactions on Mechatronics*, 8(1):26–36, March 2003.

[15] K. Oldham, J. Pulskamp, R. Polcawich, and M. Dubey. Thin-film pzt actuators with extended stroke. *Journal of Microelectromechanical Systems*, 17(4):890–899, August 2008.

[16] M.H. Mohhebi, M.L. Terry, K.F. Bhringer, G.T.A Kovacs, and J.W. Suh. Omnidirectional walking microrobot realized by thermal microactuator arrays. *proceedings of 2001 ASME International Mechanical Engineering Congress, New York, NY, USA*, pages 1–7, 2001.

[17] A. Bonvilain and N. Chaillet. Microfabricated thermally actuated microrobots. *proceedings of 2003 IEEE International Conference on Robotics and Automation, Taipei, Taiwan*, pages 2960–2965, 2003.

[18] R.J. Linderman and V.M. Bright. Nanometer precision positioning robots utilizing optimized scratch drive actuators. *Sensors and Actuators A: Physical*, A91(3):292–300, July 2001.

[19] E. Steltz, M. Seeman, S. Avadhanula, and R. S. Fearing. Power electronics design choice for piezoelectric microrobots. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1322 – 1328, 2006.

[20] Michael Karpelson, Gu-Yeon Wei, and Robert J. Wood. A review of actuation and power electronics options for flapping-wing robotic insects. *2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008*, pages 779–786, 2008.

[21] M. Chen, J.P. Vogt, and G.A. Rincon-Mora. Design methodology of a hybrid micro-scale fuel cell-thin-film lithium ion source. *IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), Montreal, Canada*, pages 5–8, August 2007.

[22] J.A. Main, D.V. Newton, L. Massengill, and E. Garcia. Efficient power amplifiers for piezoelectric applications. *Smart Materials and Structures*, 5(3):766–775, August 1996.

[23] A. S. (Ed.) Morse. Control using logic based switching, fall 1995 block island, ri, usa (pp. 79-91). *London UK: Springer-Verlag.*, 2010.

[24] C.Y. Kaya and J.L. Noakes. Computational method for time-optimal switching control. *Journal of Optimization Theory and Applications*, 117(1):69–92, APRIL 2003.

[25] W. Singhose, T. Singh, and W. Seering. On-off control of flexible spacecraft with specified fuel usage. *Proceedings of 1997 American Control Conference*, pages 2308–2312, 1997.

[26] B.J. Driessen. On-off minimum-time control with limited fuel usage: Near global optima via linear programming. *Proceedings of 2000 American Control Conference*, pages 3875–3877, 2000.

[27] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–27, 03 1999.

[28] Domenico Campolo, Metin Sitti, and Ronald S. Fearing. Efficient charge recovery method for driving piezoelectric actuators with quasi-square waves. *IEEE Transaction on Ultrasonics, Ferroelectrics, and Frequecy Control*, 50(3):237–244, MARCH 2003.

[29] Y. Mo and B. Sinopoli. Kalman filtering with intermittent observations: Tail distribution and critical value. *IEEE Transactions on Automatic Control*, PP:54–65, 2011.

[30] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S.S. Sastry. Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, 95:163 – 187, 2007.

[31] M. Epstein, L. Shi, A. Tiwari, and R. M. Murray. Probabilistic performance of state estimation across a lossy network. *Automatica*, 2008.

[32] O. C. Imer, S. Yuksel, and T. Basar. Optimal control of dynamical systems over unreliable communication links. *Automatica*, page 14291439, September 2006.

[33] C. Li and N. Elia. A sub-optimal sensor scheduling strategy using convex optimization. *2011 American Control Conference on O'Farrell Street, San Francisco, CA, USA June 29 - July 01, 2011.*

[34] Ying. He and Edwin K.P. Chong. Sensor scheduling for target tracking: A monte carlo sampling approach. *Digital Signal Processing*, 16:533–545, September 2006.

[35] M.P. Vitus, Zhang. Wei, A. Abate, Jianghai Hu, and C.J.; Tomlin. On efficient sensor scheduling for linear dynamical systems. *2010 American Control Conference Marriott Waterfront, Baltimore, MD, USA June 30-July 02, 2010*, pages 4833 – 4838, 2010.

[36] E. Iwasa and K. Uchida. Model predictive sensor scheduling. *in proceedings of International Conference on Control, Automation, and Systems*, pages 2260–2265, 2008.

[37] D. Chu, T. Chen, and H.L. Marquez. Finite horizon robust model predictive control with terminal cost constraints. *IEEE Proceedings on Control Theory Applications*, 153(2):156–166, 2006.

[38] M. Karpelson, G.-Y. Wei, and R.J. Wood. Milligram-scale high-voltage power electronics for piezoelectric microrobots. *In Proceedings of the International Conference on Robotics and Automation, Kobe, Japan*, pages 2217–2224, May 2009.

[39] K. Oldham, C.-H. Rhee, J.-H. Ryou, R.G. Polcawich, and J.S. Pulskamp. Lateral thin-film piezoelectric actuators for bio-inspired micro-robotic locomotion. *In Proceedings of the 3rd International Conference on Micro and Nano Systems, San Diego, CA, USA*, pages 759–768, August 2009.

[40] Francesco Borrelli, Mato Baotic, Alberto Bemporad, and Manfred Morari. Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica*, 41(10):1709–1721, October 2005.

[41] Xiaoling Sun Duan Li. *Nonlinear integer programming*. Springer, 2006.

[42] `http://www.ampl.com`.

[43] `http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/`.

[44] P.H. Saul, K.M. Brunson, and R.J.T. Brunyan. Versatile high voltage level shift and driver for mems applications. *Electronics Letters*, 39(2):185–186, 2003.

[45] A. Bemporad and N. Girogetti. A sat-based hybrid solver for optimal control of hybrid systems. *Lecture Notes in Computer Science*, 2993:126–141, 2004.

[46] `http://www.coilcraft.com`.

[47] Biju. Edamana, Bongsu. Hahn, Jeffrey S. Pulskamp, Ronald G. Polcawich, and Kenn Oldham. Modeling and optimal low-power on-off control of thin-film piezoelectric rotational actuators. *IEEE/ASME Transactions on mechatronics*, 16(5):884 – 896, OCTOBER 2011.

[48] Analog Devices Inc. Ultra-low power, 2-channel, capacitance converter for proximity sensing. *www.analog.com*, pages 1–28, 2007.

[49] Julia H.-L. Lu, Mac Inerowicz, Sanghoon Joo, Jong-Kee Kwon, and Byunghoo Jung. A low-power, wide-dynamic-range semi-digital universal sensor readout circuit using pulsewidth modulation. *IEEE Sensors Journal*, 11(5):1134–44, 2011.

[50] R.R. Harrison, P.T. Watkins, R.J. Kier, R.O. Lovejoy, D.J. Black, B. Greger, and F. Solzbacher. A low-power integrated circuit for wireless 100-electrode neural recording system. *IEEE Journal of Solid-State Circuits*, 42(1):123–133, 2007.

[51] `http://vtb.engr.sc.edu/vtbwebsite/#/Overview`.

[52] `http://www.ratstar.com/?v=variant&img=3`.