Modelling, Control and Estimation Techniques For Micromachined Electrostatic Actuators Using Macro Magnetic Actuators

by

Chong Li

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 6, 2016

Keywords: MEMS, Solenoids, Disturbance Rejection Control, Variable Capacitor/Inductor Modeling, Self-Sensing

Approved by

George T. Flowers, Chair, Professor of Mechanical Engineering
Robert N. Dean, Co-Chair, McWane Professor of Electrical and Computer Engineering
John Hung, Professor of Electrical and Computer Engineering
David M. Bevly, Professor of Mechanical Engineering

Abstract

Parallel plate actuators (PPAs) are fundamental devices in micro-electro-mechanical systems. PPAs' main drawback is their limited open loop stable traveling range which is caused by their nonlinear electrostatic force. Thus, feedback control techniques are required in applications which need large and precision motions. However, controlling and analyzing PPAs' behavior could be limited by several factors. The fabrication process of PPAs is expensive; the miniatured dimension of PPAs makes motion detection and experimental setup difficult.

This dissertation proposes an alternative approach to prototype analysis, control and estimation techniques for PPAs by investigating PPAs' dual systems, which are macro magnetic type solenoids. Solenoids have the similar kinematics and their stable traveling range is affected by the nonlinear magnetic force. As a test component, the advantages of solenoids include the low cost and macro size which is convenient to package and detection.

An iterative solution method is developed to study the behaviors of the actuators in circuity environment. First, the expressions of the time-variant capacitor in a PPA, AC source and their derivatives with respect to time are determined. An approximated solution combining the initial solution and its iteratively derived higher-order terms is reached. Then, the time-variant inductor in a solenoid with a restrained condition that the circuit is powered by DC sources is modeled. The iterative solution using a small signal theorem is also employed to obtain an approximate closed form solution for the time variant inductor. The simulation and experimental study further demonstrated that: (i) this iterative solution can effectively analyze the dynamics of the square law devices with a time-variant capacitor or inductor; and (ii) computing additional higher-order terms derived from the initial solution can further improve the solutions accuracy.

A practical disturbance controller is developed to extend stable range of solenoids, which could be also extended to PPAs. The inputoutput linearization control method is an effective technique to extend the stable range. But in practice, however, the time-delay effect from both measurement and actuation can make the system less damped and therefore more sensitive to disturbances. This effect was analyzed and a digital proportional and integrator controller plus extended state observer (ESO) is proposed to enhance the performance of the electromagnetic actuator. Simulation and experimental tests show that this combined proportional and integral and ESO technique can extend the stable range of motion to 77.6 % of full stroke with less sensitivity to external disturbances.

The feasibility of transplanting self sensing from solenoids to PPAs is investigated. The observability improvement of PPA design is investigated. This study proposes an alternative approach of designing an estimator using the measured voltage across the PPA without additional sensing structures and distortions. This novel method can improve the performance and reduce the device's footprint with full state (displacement and velocity) feedback information, using a series resistor. A system model using this configuration is investigated. The observability of this self-sensing technique is analyzed using a small signal model. Then, a singular value decomposition (SVD) is applied to examine how to further improve the observability by choosing appropriate parameters. Simulation and numerical studies were performed which validate this method.

Acknowledgments

First of all, I'd like to thank to my advisors, Prof. George T. Flowers and Prof. Robert N. Dean. They brought me to Auburn University and advised me for many years. I really appreciate their kind help. It was a great pleasure to work with them. I'd like to thank Prof. John Hung. Not only did his courses gained my knowledge, but he also gave me lot of help on my research projects. Thanks also to Prof. David Bevly who both advised me on my course studies and on my research works.

I also want to thank my entire family. My wife, Xinning Wang, has been always accompanied me in the past years and dedicated her life to me. My father, Guangping Li, my mother, Ailing Kong, and my brother, Xin Li, have also supported me. I deeply appreciate their dedication to me.

Thanks also to my good friends, Wenjing Yan, Haoyue Yang, Luke L. Jenkins and Xianglin Wang, for their help in my research works and for their companionship.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Electrostatic parallel-plate actuators (PPA) are commonly used in MEMS devices, with applications in RF MEMS devices [35][68], resonators [39], accelerometers [37], variable capacitors [6], micro mirrors [80] and gyroscopes [57]. PPAs operate by reducing the distance between the two electrodes in response to an applied voltage. A PPA has an open loop, voltage controlled stable displacement range, x, of $0 \leq x < x_0/3$, where x_0 is the rest gap distance between the two electrodes. Open loop attempts to further increase displacement result in unstable motion with the two electrodes snapping into contact.

Verifying proposed control techniques has several practical issues. Fabricating MEMS devices in a micro-fabrication facility is expensive and time consuming. The motion of the PPA, principally displacement and velocity, is also difficult to acquire, which is required in most feedback control strategies. Typically, there are three approaches to state measurement: optical detection [66], capacitance measurement [37] and piezoresistive detection [61]. Optical detection requires that the plate area of the PPA be large enough to reflect a laser beam, and it must possess a suitable motion direction for use with an interferometer. Capacitance measurement requires that a stationary electrode be added to the MEMS device in a suitable orientation where capacitance will vary with structure motion. This technique can suffer from parasitic capacitance affecting the external circuitry. Time varying capacitance measurement requires suitable interface circuitry [21]. Piezoresistive detection requires that piezoresistive structures be added to flextures, where they will experience mechanical strain due to structural motion. Although the interface electronics is much simpler than in capacitive detection, adding piezoresistors can significantly complicate the MEMS fabrication process. In addition to motion detection, another issue is that PPAs often require high

voltages to operate them. The typical PPA bandwidth is often between 100 Hz and 10 KHz [71], which requires high sampling rates for accurate motion detection.

The nonlinear plunger type solenoids are used as actuators in many industrial applications [11][18]. This kind of actuator also has a limited open loop stable range, but most applications utilize the solenoids as on-off components, which can ignore this characteristic. Position control of solenoids has also been proposed, including: nonlinear magnetic field mapping linearization [13], PWM control [70] and a dual solenoid configuration [78]. Solenoids are alternative actuators for evaluating control techniques for PPAs. Many commercial models are available in the market at relatively low cost. Due to its physically larger size compared to MEMS PPAs, motion detection is also reliable with optical sensors or strain gauges.

The main contribution of this dissertation is solving key problems of PPAs by studying behaviors of solenoids, which are dual systems. The solved problems including:

(1) Developing a practical and reliable position controller to improve the stable traveling range. This control technique also improves the actuators' performance under disturbances.

(2) Developing a approximated solution to solving the dynamics of solenoids/PPAs when they are considered as components in electronics circuits.

(3) Implanting the self sensing technique using state estimation from solenoids to PPAs. The observability under different system parameters and states is also analyzed.

The remainder of this dissertation is organized as follows. Chapter 2 presents the background, including modeling of the PPAs and solenoids, and challenges about feedback control in these systems. Chapter 3 presents an iterative solution to analysis when the systems are coupled with circuits elements. Chapter 4 presents a high performance control technique with uncertain time delays which result from variable inductance/capacitance. Chapter 5 discusses the feasibility of applying state estimator type self sensing technique to PPAs/solenoids and how to improve the estimators' performance.

Chapter 2

Background

2.1 Kinematics of Actuators Possessing a Square Characteristic

2.1.1 Kinematics of PPAs

A illustrative drawing of a PPA is presented in Fig. 2.1.1. The planar electrodes are arranged in a parallel configuration with an overlapping surface area, A, and a rest distance between them of x_0. The bottom electrode is fixed in space and the top electrode is allowed to move toward or away from the bottom electrode. The system's suspension system constrains the motion of the top electrode and is modeled by the system spring constant, k. C_{act} is the capacitance between the two electrodes and is modeled by

$$C_{act} = \frac{\varepsilon_0 \varepsilon_r A}{x_0 - x}; \tag{2.1}$$

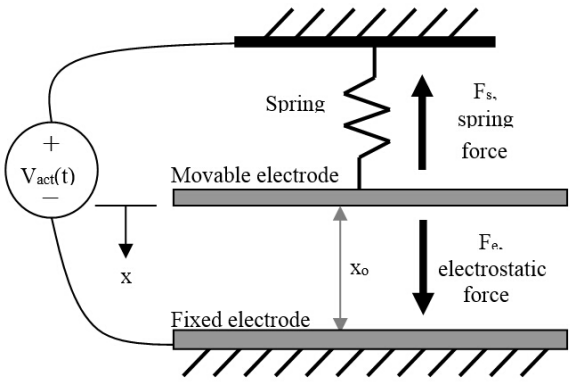


Figure 2.1: An illustration of a parallel plate actuator.

when the displacement is zero, the capacitance reaches a minimum value C_m:

$$C_m = \frac{\varepsilon_0 \varepsilon_r A}{x_0} \tag{2.2}$$

where x is the motion of the top electrode, ε_0 is the permittivity of free space and ε_r is the relative permittivity of the gas in between the electrodes. If a voltage, V_{act}, is applied across the two electrodes, an electrostatic force,

$$F_e(x, V_{act}) = \frac{\varepsilon_0 \varepsilon_r A V_{act}^2}{2(x_0 - x)^2}, \tag{2.3}$$

results that pulls the top electrode closer to the bottom electrode until its motion is balanced by the spring force

$$F_s = kx \tag{2.4}$$

from the suspension system. In this simplified PPA model, the suspension system is assumed to be satisfactorily modeled by a linear spring. Additionally, capacitive fringing field effects have been ignored. This is a reasonable assumption as long as the electrode overlap area is much greater than the electrode separation distance squared [32]. Using (2.3), a nonlinear differential equation that models the PPA system dynamics can be developed:

$$m\ddot{x} = -c\dot{x} - kx + \frac{\varepsilon_0 \varepsilon_r A V_{act}^2}{2(x_0 - x)^2}. \tag{2.5}$$

Here, m is the proof mass, typically the movable electrode, and c is the system coefficient of damping. The system's natural frequency, ω_0, is:

$$\omega_0 = \sqrt{\frac{k}{m}}, \tag{2.6}$$

while Q, the mechanical quality factor, is

$$Q = \frac{\omega_0 m}{c}. \tag{2.7}$$

MEMS devices are generally designed to be highly underdamped [32], with the exception of MEMS accelerometers, which are usually designed to be critically damped [9].

When the PPA is operated in its open loop stable range of motion, the electrostatic force and the spring force are in equilibrium. But, since the spring force is linearly proportional to displacement and the electrostatic force is a nonlinear function of both displacement and applied voltage, the actuator has a limited open loop stable range of displacement of $0 \leq x < x_0/3$. If the applied voltage is further increased in attempt to reach a displacement of $x_0/3$ or further, the system will no longer be in equilibrium since the electrostatic force will always be greater than the spring force. This results in the movable electrode accelerating toward and snapping into contact with the fixed electrode. This event is called snap-in. The minimum applied voltage that will cause snap-in is referred to as the pull-in voltage [9], V_{pi}, where

$$V_{pi} = \sqrt{\frac{8(x_0)^3}{27\varepsilon_0\varepsilon_r A}}. \tag{2.8}$$

2.1.2 Kinematics of Solenoids

A typical configuration of a solenoid in a spring-mass-damper system is illustrated in Fig. 2.1.2. Its mathematical description is:

$$m\ddot{x} + c\dot{x} + kx = \frac{\mu_0\mu_r N^2 A I^2}{2(d + x_0 - x)^2}, \tag{2.9}$$

where x is the displacement of the armature, μ_0 is the permeability of free space, A is the cross-sectional area of the core, N is the number of the turns of the coil, μ_r is the relative

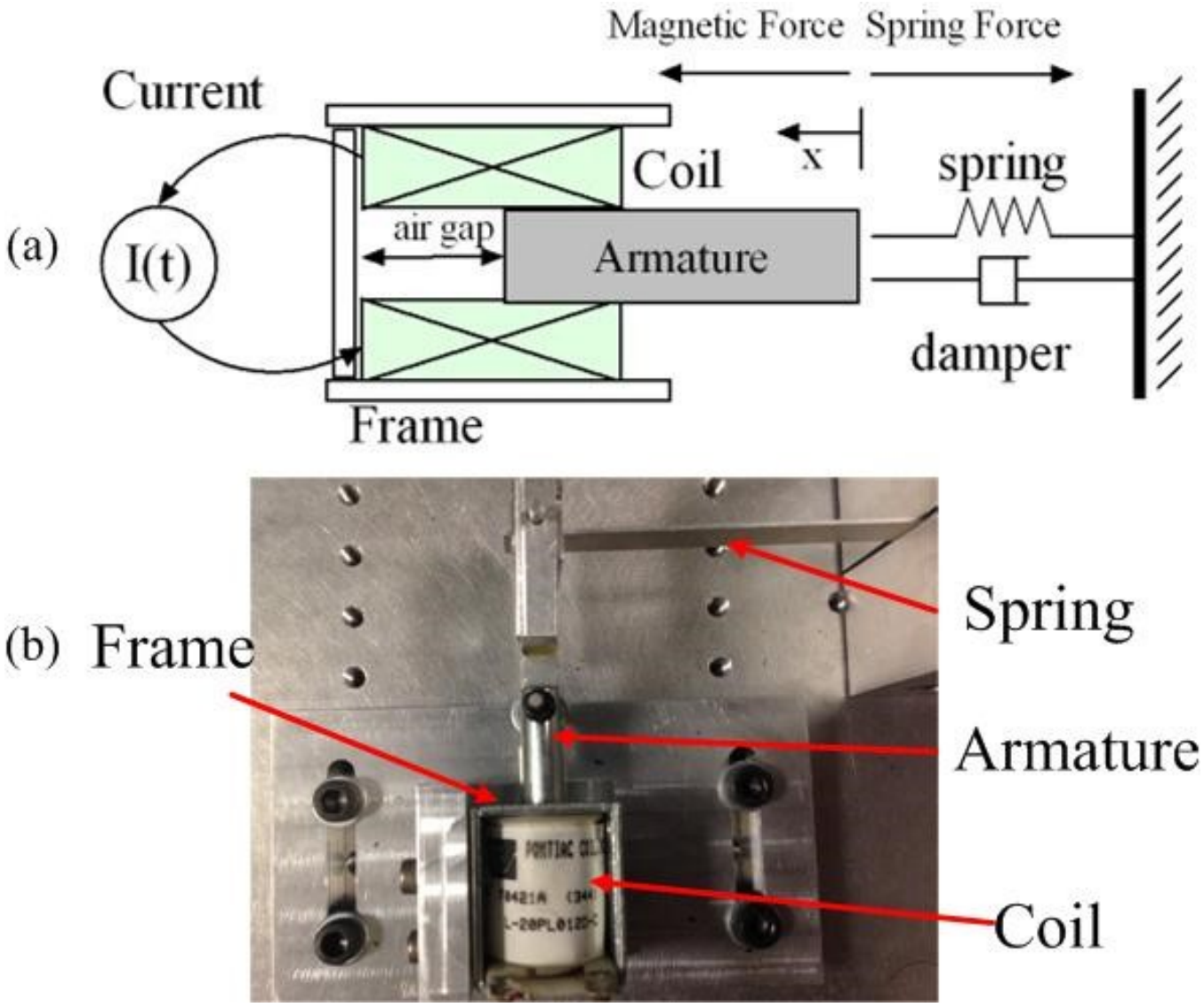


Figure 2.2: (a) is an Illustration of a solenoid with a spring and a damper. (b) is an example of a commercial solenoid fixed to a metal cantilever as the spring

permeability of the dielectric material between the coil and the armature, x_o is the initial air gap between the armature and the back side of the frame and d is the additional initial air gap related to the solenoid's geometry which is much smaller than x_0 in general. Also, m is the proof mass of the armature, k is the stiffness of the spring and c is the system damping coefficient. $I(t)$ is the current through the coil [78][70]. $L(x)$ is the inductance of the coil that depends on the air gap x [78][70], which is:

$$L(x) = \frac{\mu_0 \mu_r A}{d + x_0 - x(t)}. \tag{2.10}$$

The plunger is driven by a magnetic force which could be described by:

$$F(x, I(t)) = \frac{\mu_0 \mu_r N^2 A I^2}{2(d + x_0 - x)^2}. \tag{2.11}$$

At the equilibrium:

$$kx = \frac{\mu_0 \mu_r N^2 A I^2}{2(d + x_0 - x)^2}. \tag{2.12}$$

Because the magnetic force is also nonlinear, this type of actuator also has the pull-in effect. The pull-in current is:

$$I_{pi} = \sqrt{\frac{8(x_0)^3}{27 \mu_0 \mu_r N^2 A}}. \tag{2.13}$$

2.1.3 The Pull-in Effect and Stable Range of PPAs

The stable traveling range of PPAs is 1/3 of x_0. A linear approximation by (2.5) can be formed by Taylor series expansion of $F_e(x, V)$ [60]. From the Taylor series:

$$\frac{\partial F_e(x, V)}{\partial x} = \frac{\varepsilon_0 \varepsilon_r A V_{act}^2}{(x_0 - x)^3}, \tag{2.14}$$

Let x_a be a desired displacement over the stable range and V_a be the voltage from (2.3). Then:

$$F_e(x,V) = \frac{\varepsilon_0 \varepsilon_r A V_a^2}{2(x_0-x)^2}(x-x_a)^0 + \frac{\varepsilon_0 \varepsilon_r A V_a^2}{2(x_0-x)^3}(x-x_a)^1, \tag{2.15}$$

Define N and K_{EL} as:

$$\begin{aligned} N &= \frac{\varepsilon_0 \varepsilon_r A V_a^2}{2(x_0-x)^2} - x_a \frac{\varepsilon_0 \varepsilon_r A V_a^2}{2(x_0-x)^3} \\ K_{EL} &= \frac{\varepsilon_0 \varepsilon_r A V_a^2}{2(x_0-x)^3}. \end{aligned} \tag{2.16}$$

Then substituting (2.21) for $F_e(x,V)$ in (2.5) yields the linear approximation:

$$m\ddot{x} + c\dot{x} + (k - K_{EL})x = N. \tag{2.17}$$

The characteristic equation has root at:

$$s_{1,2} = \frac{1}{2}\left(\frac{-c}{m} + \sqrt{(\frac{-c}{m})^2 - 4(\frac{k-K_{EL}}{m})}\right). \tag{2.18}$$

This system is only stable by open loop voltage control for $0 \leq x < x_0/3$. If the PPA voltage is increased in attempt to increase the displacement beyond $x_0/3$, the two electrodes will snap into contact.

2.1.4 The Pull-in Effect and Stable Range of Solenoids

Because solenoids has the same type of driven force, the PPAs' pull-in principle can also applied to solenoids and generate the same result. The stable traveling range of solenoids is also 1/3 of x_0. A linear approximation around an operational point of the solenoid can be

formed by Taylor series expansion of $F_m(x, I)$. From the Taylor series:

$$\frac{\partial F_m(x,I)}{\partial x} = \frac{\mu_0 \mu_r N^2 A I_a^2}{(d + x_0 - x)^3}, \tag{2.19}$$

Let x_a be a desired displacement over the stable range and I_a be the voltage from (4.9). Then:

$$F_m(x,I) = \frac{\mu_0 \mu_r N^2 A I^2}{2(x_0 - x)^2}(x - x_a)^0 + \frac{\mu_0 \mu_r N^2 A I_a^2}{2(x_0 - x)^3}(x - x_a)^1, \tag{2.20}$$

Define N and K_M as:

$$\begin{aligned} N &= \frac{\mu_0 \mu_r N^2 A I^2}{2(x_0 - x)^2} - x_a \frac{\mu_0 \mu_r N^2 A I_a^2}{2(x_0 - x)^3} \\ K_M &= \frac{\mu_0 \mu_r N^2 A I_a^2}{2(x_0 - x)^3}. \end{aligned} \tag{2.21}$$

Then substituting (2.21) for $F_m(x, I)$ in (2.5) yields the linear approximation:

$$m\ddot{x} + c\dot{x} + (k - K_M)x = N. \tag{2.22}$$

The characteristic equation has root at:

$$s_{1,2} = \frac{1}{2}\left(\frac{-c}{m} + \sqrt{\left(\frac{-c}{m}\right)^2 - 4\left(\frac{k - K_M}{m}\right)}\right). \tag{2.23}$$

This system is only stable by open loop voltage control for $0 \leq x < x_0/3$. If the current is increased in attempt to increase the displacement beyond $x_0/3$, the armature will contact the bottom of the frame.

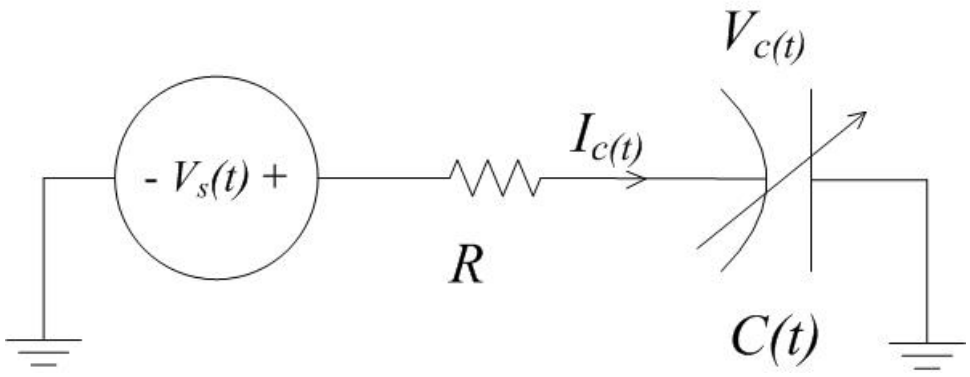


Figure 2.3: A Thevenin equivalent schematic diagram of a time-variant capacitor with a series resistor and an AC voltage source.

2.2 Characteristics of Square Law Actuator Systems in Circuits

2.2.1 PPAs with a Series Resistor

PPAs contain variable capacitance which depends on the displacement of the movable electrode. Consider a time-variant capacitor with a series resistor, R, which is used to protect the capacitive element and prevent the power source from shorting to ground in case the MEMS device's electrodes physically contact each other. The circuit's schematic is given in Fig. 2.3. In addition, different configurations can be transformed into this model using the Thevenin equivalent circuit method.

The circuit's behavior is described by:

$$I_c(t) = (V_s(t) - V_c(t))/R = \dot{V}_c C(t) + V_c \dot{C}(t), \tag{2.24}$$

where V_s is the power source, V_c is the voltage across the variable capacitor and I_c is the current through it. Though (3.5) fully characterizes the circuit's behavior, it is difficult to solve this nonlinear differential equation in practice and obtain a closed form solution.

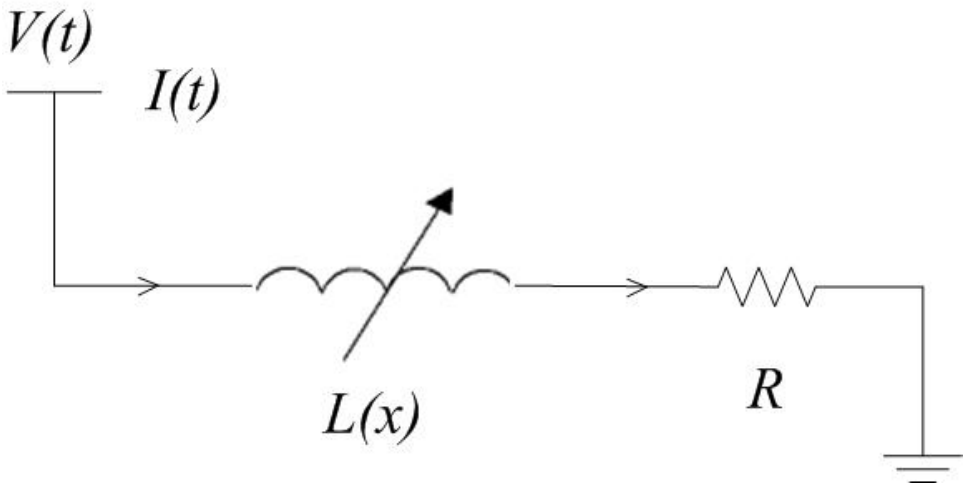


Figure 2.4: Solenoid equivalent circuit

2.2.2 Solenoids with a Series Resistor

A solenoid can be treated as variable inductors when it is in an electronics circuit. Based on (2.10), $L(x)$ is a variable inductor; the dynamics of the circuit in Fig. 2.4 are modeled as:

$$L(x)\frac{dI(t)}{dt} + \frac{dL(x)}{dt}I(t) + I(t)R = V(t). \tag{2.25}$$

According to (4.9) and (4.10), the electrical and mechanical dynamics are coupled with each other, which makes solving (4.11) difficult.

2.3 Stabilization methods of PPAs

A number of techniques have been investigated in order to increase the stable displacement range of PPAs using various controller architectures. Examples include the series capacitor method (SCM) [65], synthetic voltage division [20], charge control [14], negative capacitance control [64], various electrode configurations [9], voltage driven linear feedback control [10], nonlinear output feedback stabilization [11], sliding mode control [12], small signal model design [13] and input-output linearization [14]. Each of these techniques offers certain benefits and detriments.

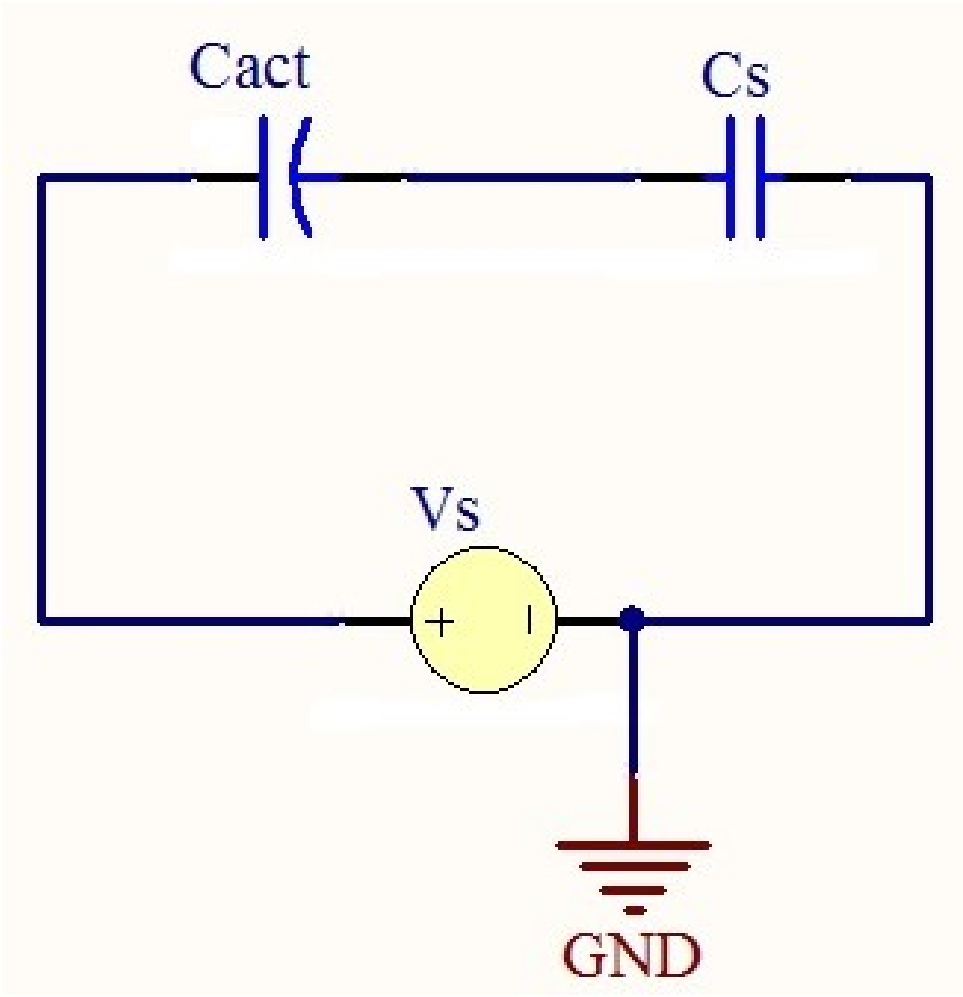


Figure 2.5: The schematic of a PPA with a series capacitor.

2.3.1 Ideal Series Capacitor Method

The SCM is the most classical way to extend the stable range of PPAs. The SCM uses an additional capacitor in series with the PPA to increase the stable range of electrode motion. A schematic diagram for the SCM is presented in Fig. 2.5, where C_{act} is the variable capacitance of the PPA, and C_s is the series capacitance. V_s is the source voltage which is provided by an external power source, and V_{act} is the resulting voltage across the PPA and equals:

$$V_{act} = \frac{1}{\frac{C_{act}}{C_s} + 1} V_s. \tag{2.26}$$

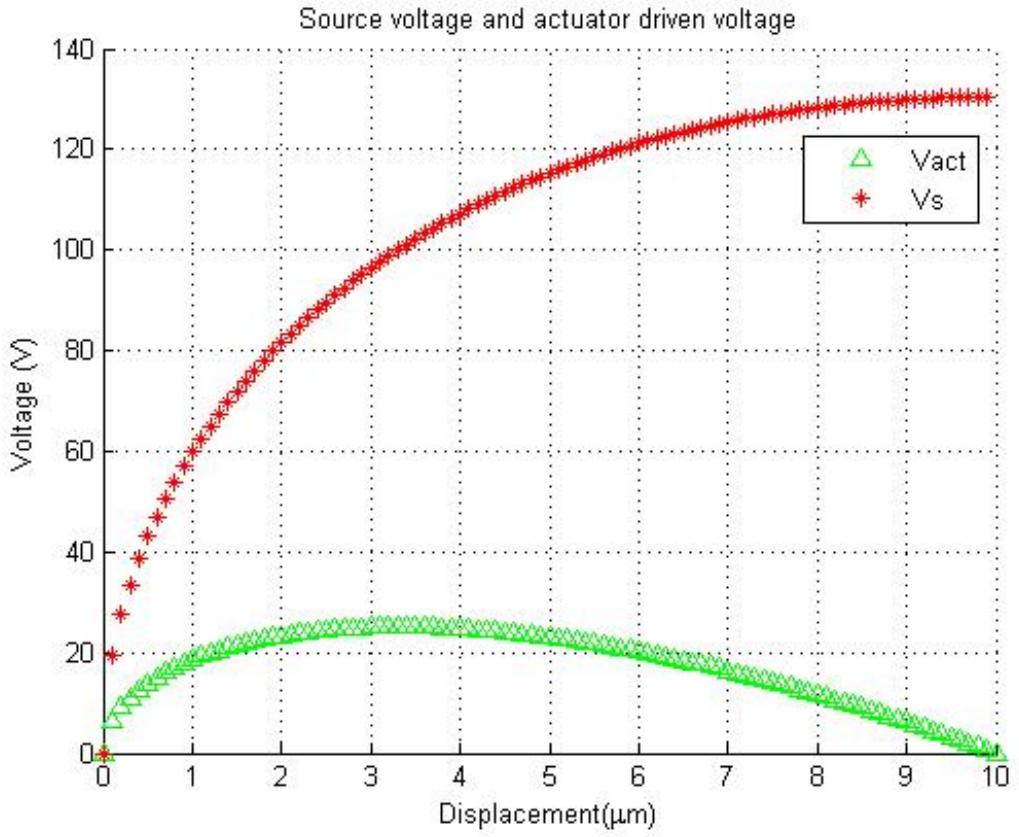


Figure 2.6: V_{act} and V_s plots of the example PPA.

By substituting (2.26) into (2.5), the system model could be described as:

$$m\ddot{x} = -c\dot{x} - kx + \frac{\varepsilon_0 \varepsilon_r A V_s^2}{2(x_0 - x + \frac{\varepsilon_0 \varepsilon_r A}{C_s})^2}, \tag{2.27}$$

If the series capacitor, C_s, is no greater than half of the minimum capacitance of the PPA, C_m, the stable range of motion for the PPA can be extended to the entire rest gap. This method, in effect, emulates an increase in the rest gap by a factor of three. When this resulting virtual rest gap is greater than or equal to $3x_0$, the actuator will have a stable range of x_0. However, the SCM requires a much higher input voltage to drive the PPA due to the voltage drop across the series capacitor.

Consider a typical example of a PPA in air having a surface area, A, of 2.91mm by 2.91mm, and a rest gap, x_0, of 10μm, a system spring constant, k, of 158.2 N/m, a mechanical

quality factor, Q, of 50, and a proof mass, m, of 7.5 $\times 10^{-6}$ kg. Its resulting source voltage and PPA drive voltage are shown in Fig. 2.6., for a series capacitor value equal to one half of C_m, where C_m is 7.5pF. It is clear that the input voltage, V_s, is much higher than the voltage actually across the PPA. The maximum voltage for V_s is approximately 130V, which is problematic for modern analog electronic devices. Additionally, a value for C_s of 3.75pF is also problematic to obtain in discrete form.

2.3.2 Series Capacitor Method with Parasitic Capacitance

In practice, the SCM is also limited by parasitic capacitance. Consider an example of the PPA used in section 2.2 with a capacitance at rest of 7.5pF. Considering that the PPA must be electrically connected to the rest of the electrical system, the PPA will be electrically in parallel with stray or parasitic capacitance. To analyze the effects of this parasitic capacitance, C_p, it is necessary to include it in the system model. The resulting schematic diagram is presented in Fig. 2.7. Therefore V_{act} becomes:

$$V_{act} = \frac{1}{\frac{C_{act}}{C_s} + \frac{C_p}{C_s} + 1} V_s. \tag{2.28}$$

According to (2.5) and (2.28), the systems dynamics equation is:

$$m\ddot{x} = -c\dot{x} - kx + \frac{\varepsilon_0 \varepsilon_r A V_s^2}{2((x_0 - x)(1 + \frac{C_p}{C_s}) + \frac{\varepsilon_0 \varepsilon_r A}{C_s})^2}. \tag{2.29}$$

At equilibrium:

$$kx = \frac{\varepsilon_0 \varepsilon_r A V_s^2}{2((x_0 - x)(1 + \frac{C_p}{C_s}) + \frac{\varepsilon_0 \varepsilon_r A}{C_s})^2}. \tag{2.30}$$

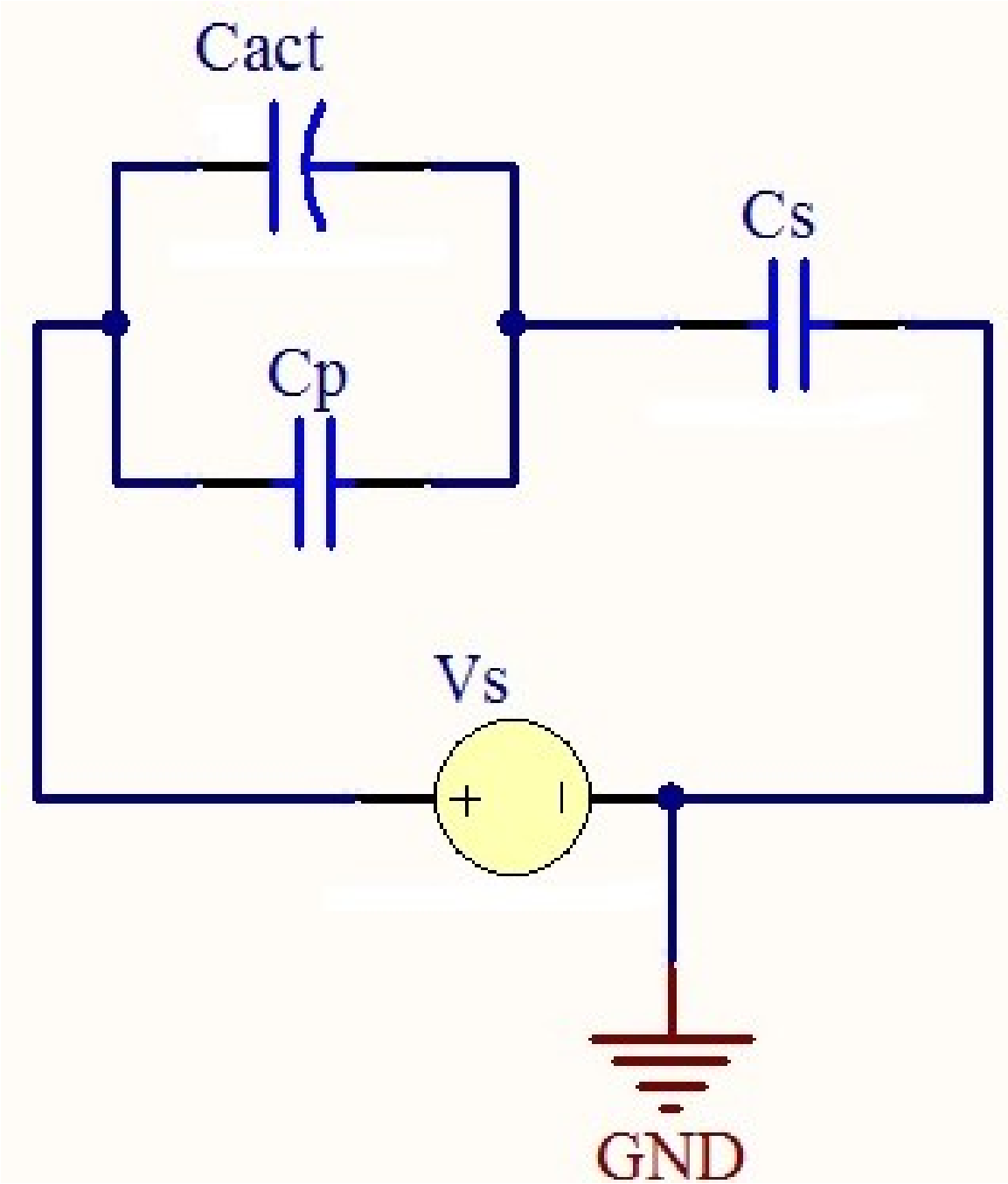


Figure 2.7: The schematic of a PPA with a series capacitor and a parasitic capacitor.

The condition of stability is that the electrostatic force grows slower than the spring force, which means:

$$\frac{\partial F_e(x,V)}{\partial x} \leq \frac{\partial kx}{\partial x}. \tag{2.31}$$

Combining (2.29), (2.30) and (2.31):

$$x \leq \frac{x_0}{3} + \frac{\varepsilon_0 \varepsilon_r A}{3(C_s + C_p)}. \tag{2.32}$$

which indicates that the parasitic capacitance could decrease the stable range. Its effect is equivalent to increasing the value of C_s, while the SCM is valid only for C_s less than one half of C_m. Since C_s+C_p is larger than one half of C_m, the system will not have a full range of stable displacement. The minimum stable range is still 1/3 of x_0, if C_s+C_p goes to infinite, while the term in the right of (2.32) will be canceled.

From a control systems' point of view, (2.28) can be expanded into a feedback control diagram having two feedback loops, as shown in Fig. 2.8.

2.4 Self Sensing Techniques

The self sensing technique is defined by estimating the mechanical motions (displacement and velocity) by measuring the electrical values (voltage and current) without external sensors. This technique does have certain advantages include reducing the costs and footprints for the actuator system.

2.4.1 Self Sensing for Induction Actuators

Self sensing techniques are commonly used in induction actuators [63][41]. In general, there are two ways to estimate the mechanical motions by measuring the current inside solenoids:

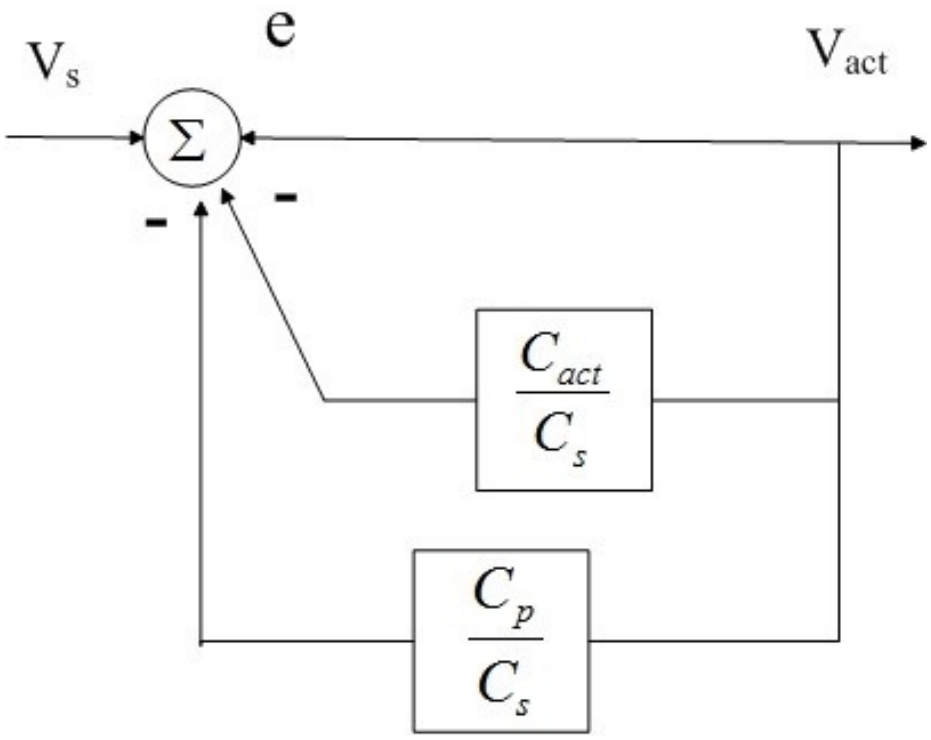


Figure 2.8: Control diagram of the series capacitor method with a parasitic capacitor.

(1) Introducing external excitation sources and measuring the corresponding excitation responses [63].

(2) Using state estimators/observers to trace the motions without external excitation sources [41]. Fig. 2.9 shows the principle of the first approach. An external excitation source is added to generate high frequency signals to the actuator, which is much higher than its natural frequency. Thus, the extra signals will generate small mechanical motions. The current inside the solenoid can be divided into control part and sensing part in the frequency domain. The phase delay of the current that is introduced by the high frequency excitation source in the circuit is dependent on the value of the inductance. Also, according to (2.10), the value of the inductor is depending on the position of the plunger. So, The mechanical states can be obtained by processing the sensing part after a high pass filter. This approach's resolution is high, but it requires additional hardware and the excitation source that may introduce ripples into the actuator.

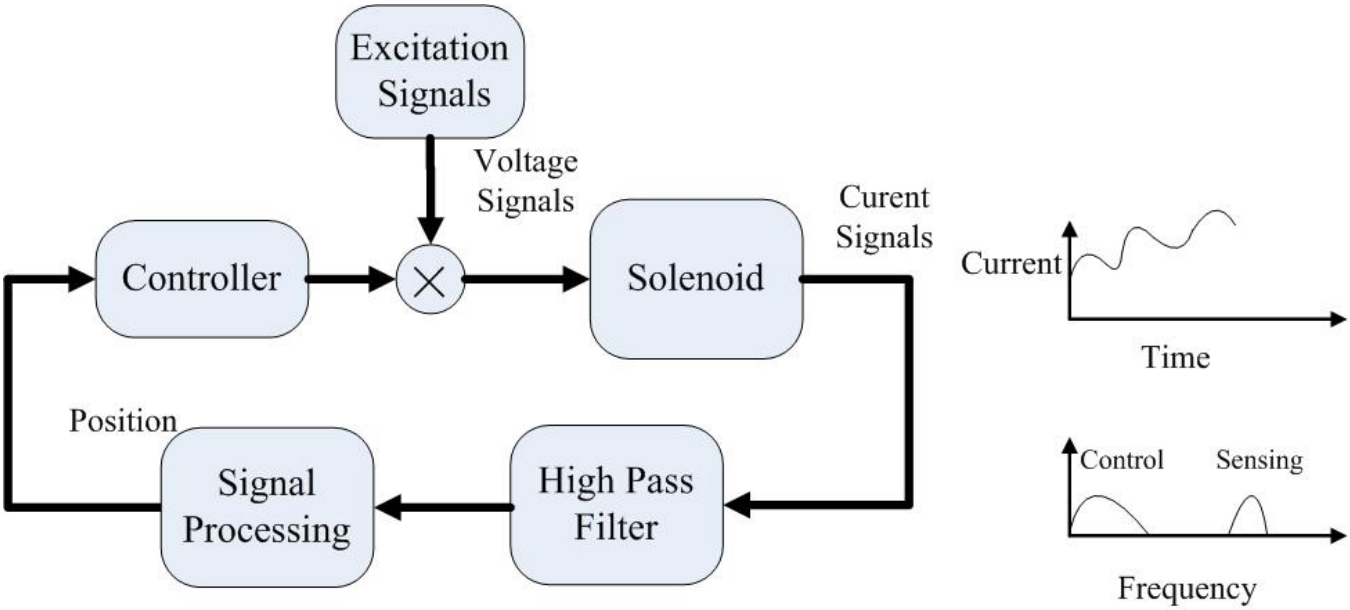


Figure 2.9: Self sensing diagram for a solenoid using high frequency excitations.

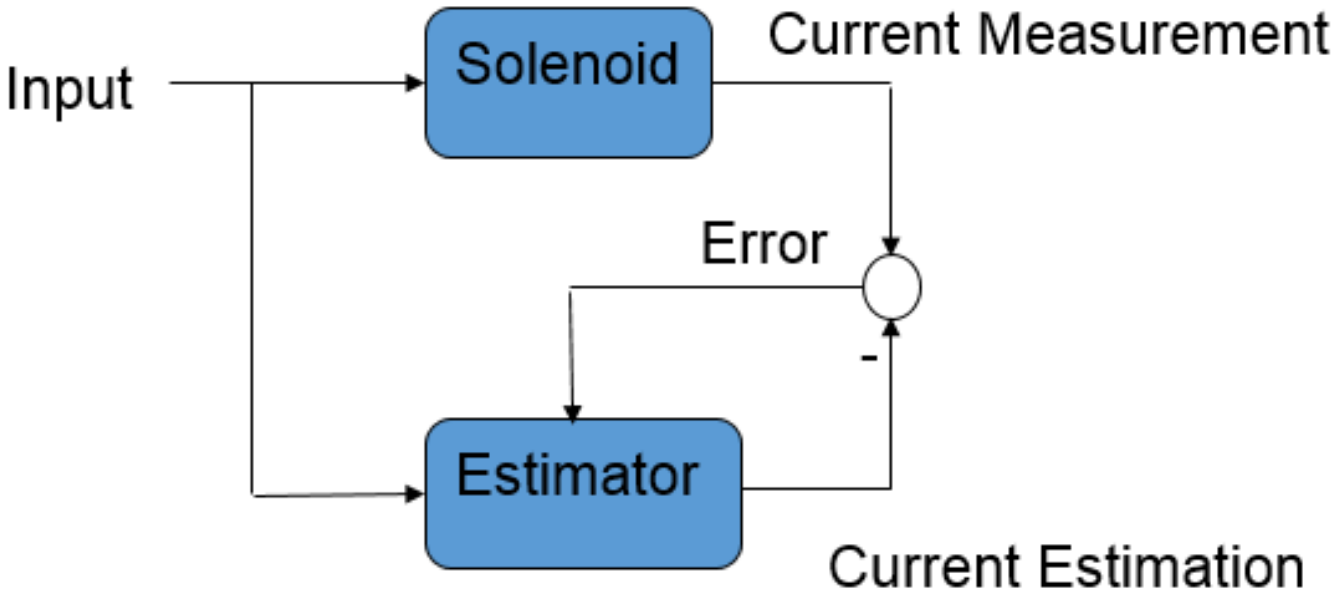


Figure 2.10: Self sensing diagram for a solenoid using an state estimator.

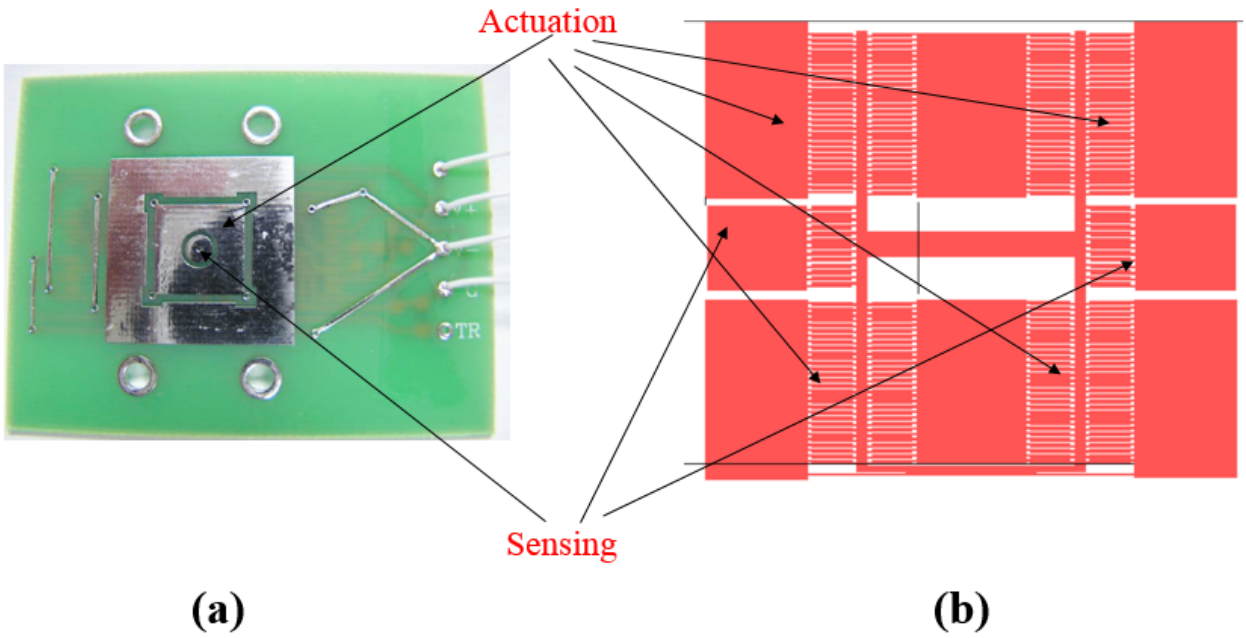


Figure 2.11: Examples of additional strucutres for a PPA using high frequency excitations self sensing.

Fig. 2.10 shows the principle of estimating states using a state estimator. The state estimator does not require additional signal sources. It can estimate the position and velocity of the solenoid with a known input voltage and the current inside the actuator. This method will require less hardware than the previous method and will not affect the system's normal operation.

2.4.2 Self Sensing for PPAs

Self sensing techniques are also applied in the MEMS area. Using self sensing techniques in PPAs has certain advantages, including reducing the size of the device without requiring specific sensing structures. Fig. 2.11 gives examples of PPAs with sensing structures where (a) is a bottom fixture for a PPA with a central sensing electrode and (b) is a demonstration of a electrostatic actuator with separate sensing structures.

Fig. 2.12 demonstrates the self sensing method using an external excitation source, which is similar to solenoids. An external periodical excitation source which frequency is

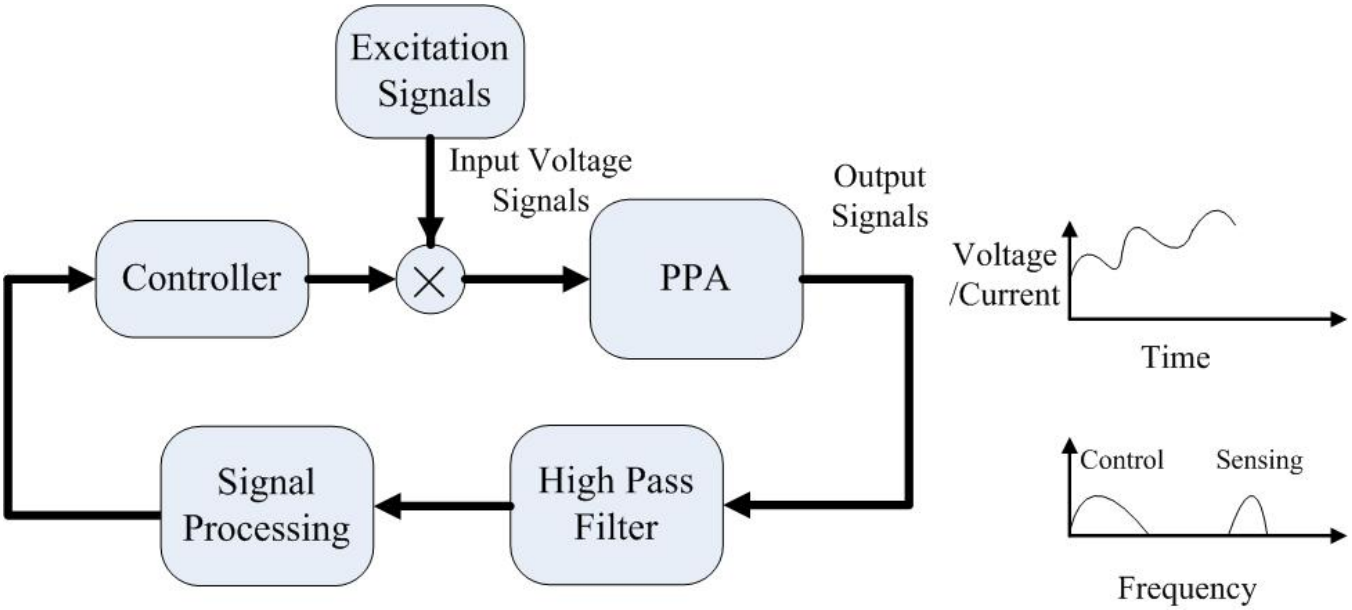


Figure 2.12: Self sensing diagram for a PPA using high frequency excitations.

much higher than the natural frequency of a PPA will introduce small amplitude motion compared to the entire traveling range [42]. The phase delay of the voltage across the device or the current in the circuit is dependent on the value of the capacitance. Also, according to (2.1), the value of the capacitor is dependent on the position of the movable electrode. Then, the information about the position can be determined.

Though this approach is reliable, it requires additional sensing structures and electronic circuits. Besides, it cannot provide information about the velocity. Fig. 2.13 demonstrates the idea of self sensing technique for PPAs using a state estimator. It requires a series resistor with the PPA. If the system parameters are identified, the input voltage is known and the voltage across the PPA can be measured, the mechanical states (position and velocity) can be obtained by using state estimators. Chapter 5 will give the details.

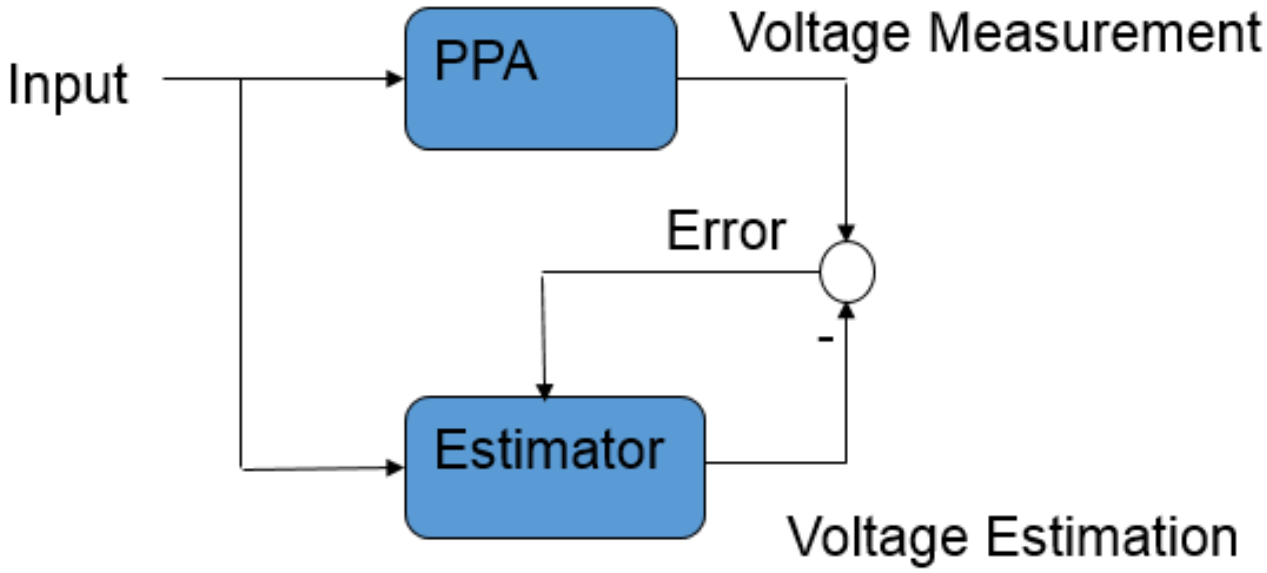


Figure 2.13: Self sensing diagram for a PPA using high frequency excitation.

Chapter 3

Approximated Iterative Solutions for Solenoids and PPAs

3.1 Problem Statement

3.1.1 Time-variant Capacitors in PPAs

Time-variant capacitors exist in many microelectromechanical systems (MEMS) devices. They could be designed as sensing structures where the variation in capacitance represents different external physical parameters; examples include humidity sensors [43], vibration sensors [10], strain sensors [69], pressure sensors [55], gyroscopes [26] and accelerometers [47]. On the other hand, electrostatic actuators also include time-variant capacitors, which are key components in many applications such as resonators [79][25], micro mirrors [25], series switches [29], compliant structures [7] and RF devices [59][46].

Thus, it is important to accurately analyze systems possessing time-variant capacitors. For a time-variant capacitor, $C(t)$, the current, $I_c(t)$, through it is:

$$I_c(t) = \dot{V}(t)C(t) + V(t)\dot{C}(t). \tag{3.1}$$

Because (3.4) results in a nonlinear circuit model, it is difficult to obtain a closed form solution. In some circumstances, this problem can be simplified by considering the variable capacitance as a constant, especially if it is powered by a DC voltage source or the AC source's frequency is much higher than the MEMS device's bandwidth. Then, linear circuit analysis can be applied to obtain a reasonably accurate solution. However, the time-variation cannot be ignored in many applications. For example, consider a MEMS resonator vibrating at a 2 KHz resonant frequency. In order to accomplish the feedback control, a 100 KHz voltage signal is applied to the device to detect the capacitance to measure the proof mass motion,

which is a common configuration [36][17]. Typically, the capacitance is treated as time-invariant in this situation. Since the detection signal and the mechanical vibration should not affect each other, because the detection signal's frequency is much greater than the device's bandwidth, it is commonly considered that no additional components from the mechanical motion will be introduced [24]. In reality, however, the time-variant capacitor caused by the mechanical motion will introduce more harmonics into the detection signal, with a frequency spacing equal to the frequency of the fundamental component. The spectrum of this example is given in Fig.3.1 from a MATLAB Simulink simulation of the system. This phenomenon, though, has been experimentally documented in MEMS devices [19][22]. In this figure, it is clear that harmonics with 2 KHz intervals are introduced, where each interval is equal to the vibration frequency. These harmonics can corrupt the measurement readings and introduce additional noise into the electrical system. Linear circuit analysis does not account for this very real and observable effect.

Notice that the result contains both harmonics and intermodulation products of the AC source and the resonant frequency. Linear circuit analysis considers neither of these components. R.N. Dean et al [21] proposed a nonlinear circuit analysis method for a time-variant MEMS capacitor system driven with a DC source, but the case with an AC voltage source was not investigated. To solve this problem, a standard circuit model needs to be considered.

Consider a time-variant capacitor with a series resistor, R, which is used to protect the capacitive element and prevent the power source from shorting to ground in case the MEMS device's electrodes physically contact each other. The circuit's schematic is given in Fig. 3.2. In addition, different configurations can be transformed into this model using the Thevenin equivalent circuit method.

The circuit's behavior is described by:

$$I_c(t) = (V_s(t) - V_c(t))/R = \dot{V}_c C(t) + V_c \dot{C}(t), \tag{3.2}$$

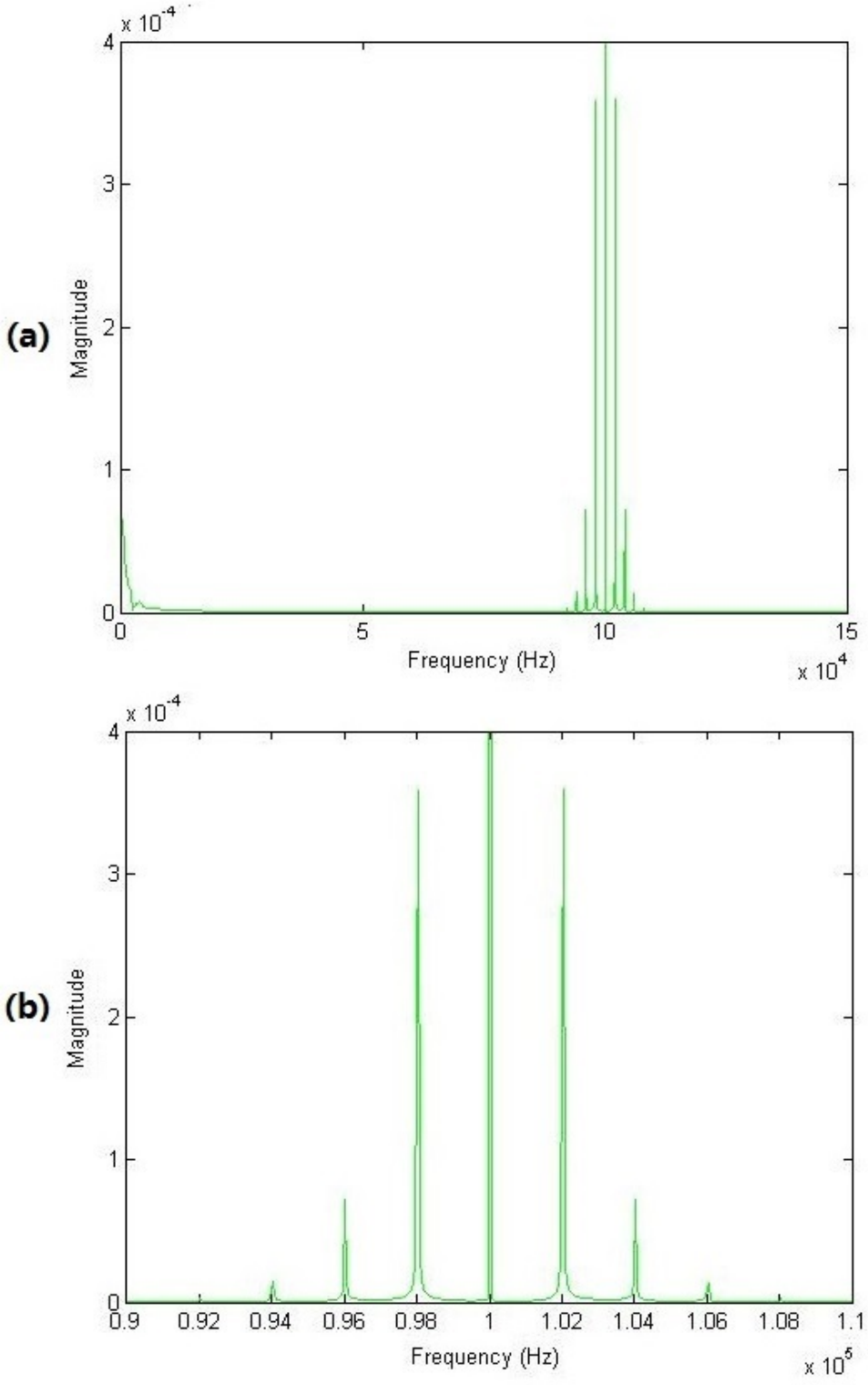


Figure 3.1: Spectral analysis of a MEMS resonator with detection signals. (a) is the overall spectrum. (b) is the spectrum around the detection signal's fundamental frequency.

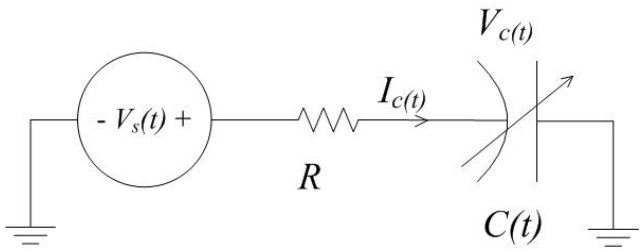


Figure 3.2: A Thevenin equivalent schematic diagram of a time-variant capacitor with a series resistor and an AC voltage source.

where V_s is the AC power source, V_c is the voltage across the variable capacitor and I_c is the current through it. Though (3.2) fully characterizes the circuit's behavior, it is difficult to solve this nonlinear differential equation in practice and obtain a closed form solution.

3.1.2 Time-variant Inductors in Solenoids

Time-variant inductors exist in many industrial applications, such as electric motors [38], magnetic bearings [33], Linear Variable Differential Transformers (LVDT) [74], piezoelectric actuators [8] and fluidic valves [76]. The changing inductance will make the flux inside a motor nonuniform, which causes eddy currents to occur. The eddy current loss (induction loss) is a major loss in addition to Ohmic losses in the copper, hysteresis loss and mechanical loss [1][52], especially if a motor is running at high speed [34][27]. Many research efforts have investigated this kind of induction loss issue in order to not only improve the power efficiency, but to also cancel the steady state ripples resulting from the eddy currents.

On the other hand, many techniques utilize the eddy currents for sensing, such as metal dection sensors [77] and non-destructive sensors [28]. Some automatic electrical braking systems use eddy currents for magnetic breaking [5]. Solving a nonlinear circuit with a time-variant inductor is difficult. The typical way to solve this problem is considering the time-variant inductor as time-invariant, which will yield a linear circuit analysis. However, when the eddy currents are not neglectable, the solutions are incorrect. With modern computer processing systems, numerical methods such as the Runge-Kutta algorithm are used

to numercially solve these ordinary differential equations. Nevertheless, in many applications such as model predictive control with high speed motors, it is still not fast enough and economical [3]. In some circumstances, a system with a variable inductor is driven by a DC source and the differentiation could be calculated. Here, an alternative fast method can be applied to obtain a sufficiently precise approximate solution.

This investigation proposes a nonlinear technique for analyzing systems containing time-variant inductors, such as motors and solenoids, that are driven by or can be approximated as being driven by a DC source. The method is given in the following steps: (a) Identifying the circuit model of a motor and transforming it into its Norton equivalent circuit model. (b) Extracting a time-varying formula for the inductance. (c) Calculating the current through the motor using a set of iterative equations. The rest of the paper provides the details. Considering the inductor inside a DC motor to be time-invariant, the voltage,V_l, across it is:

$$V_l = \dot{I}_l L, \tag{3.3}$$

where L is the time-invariant inductor and I is the current through the inductor.

The prediction models do not have to perfectly match the actual system, but a better model can improve the performance. For a time-variant inductor in real practice, $L(t)$, the voltage, V_l, that across the inductor is:

$$V_l = \dot{I}_l L(t) + I_l \dot{L}(t). \tag{3.4}$$

The time varying inductor makes the circuit a nonlinear system and linear circuit analysis techniques can therefore not be used to obtain a closed form solution. Linear circuit analysis can be applied if considering the inductor as a constant, where its value can be taken as the averaged value of $L(t)$. It could be an issue if the application requires high precision when using the approximated linear analysis. Using linear circuit analysis has some disadvantages such as superpositioning cannot be used and the output frequency content differs

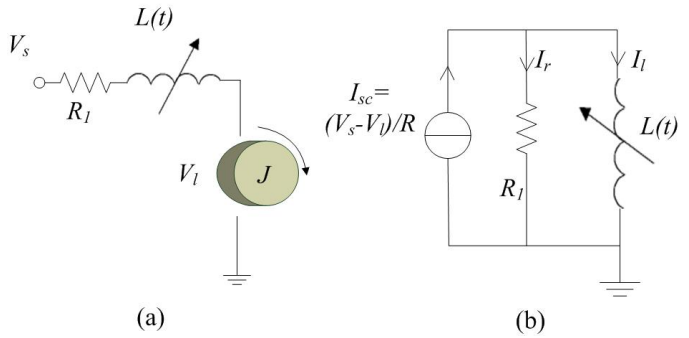


Figure 3.3: A variable inductor's schematic diagram and its equivalent Norton circuit

from the input frequency content. Even though nonlinear differential equations can usually be solved using numerical techniques, they are time consuming, which require high quality hardware and are difficult to implement, which can be key problems in predictive control. Furthermore, applications like predictive control usually require sampling and predicting the process as fast as it can especially when dealing with a rapid response system, which is a conflict with the limited calculation speed for this complicated nonlinear system.

So, a simplified analysis can be used considering the case where the inductor circuit contains only resistors and DC sources, which is a typical configuration in inductive position sensors [62][74] and the steady state operation of a DC motor. An example circuit is shown in Fig. 1. Fig. 1(a) is the equivalent circuit of a DC motor, where V_s is the steady state voltage, R_1 is the series resistance inside the motor, V_l is the voltage generated by the rotor and load, and $L(t)$ is the variable inductor. It could be transformed to a Norton equivalent circuit for the sake of analysis as shown in Fig. 1(b). In this equivalent model, I_{sc} and R_n are the short circuit current source and the Norton equivalent resistor related to the original voltage input V_s and resistor, I_r is the current through the parallel resistor, R_n, and I_l is the current through the inductor $L(t)$. The rest of the discussion is based on the standard Norton equivalent model.

The characterization of the inductors inside a motor can be done using either analytical methods [49][67] or finite element analysis (FEA) [2]. The circuit operation is described by:

$$V_l = (I_{sc} - I_l)R_n = \dot{I}_l L(t) + I_l \dot{L}(t), \tag{3.5}$$

where V_l is the voltage across the variable inductor. Though (3.5) fully characterizes the circuit's behavior, it is difficult to solve this nonlinear differential equation in practice and obtain a closed form solution.

3.2 Iterative Approxiamted Solutions for Time-variant Capacitors/Inductors

Because (**??**) and (3.5) are difficult to solve, an alternative iterative approach can be applied to obtain an approximate solution [21].

3.2.1 Iterative Approxiamted Solutions for Time-Variant Capacitor with an AC Power Source

The first step is deriving an initial approximate solution $V_{c0}(t)$, which is sufficiently close to the $V_c(t)$ [44]. To achieve this initial solution, the variable capacitor in a MEMS device can be modeled as:

$$C(t) = C_0 + C_1(t), \tag{3.6}$$

where C_0 is time-invariant, and $C_1(t)$ is the time-variant part that must be less than C_0 to ensure that $C(t)$ is always positive. The derivative of (3.6) is:

$$\dot{C}(t) = \dot{C}_1(t). \tag{3.7}$$

Ignoring the time-variant part and considering this circuit as a linear circuit, linear circuit analysis can be applied to obtain the initial solution $V_{c0}(t)$. The approximated circuit's

transfer function is:

$$\frac{V_c(S)}{V_s(S)} = \frac{1}{RC_0 s + 1}. \tag{3.8}$$

Then the first step is the analysis of this circuit ignoring the effect of $C_1(t)$ and considering it as a linear circuit. The purpose of this step is to obtain the steady state of $V_c(t)$, denoted by $V_{c0}(t)$. Then, $V_{c1}(t)$ can be calculated by (**??**) and (3.6). Correspondingly, the $I_{c1}(t)$ term is:

$$I_{c1}(t) = \dot{V}_{c0}C(t) + V_{c0}\dot{C}(t). \tag{3.9}$$

Thus, using a small signal analysis to obtain V_{c1}:

$$V_{c1}(t) = -I_{c1}(t)R. \tag{3.10}$$

Additional terms can be calculated recursively:

$$I_{ck+1}(t) = \dot{V}_{ck}C(t) + V_{ck}\dot{C}(t). \tag{3.11}$$

The $I_{lk}(t)$ is then solved using as many terms as required to obtain sufficient accuracy. The overall equation for $V_c(t)$ is:

$$V_c(t) = V_{c0}(t) + \sum_{k=0}^{\infty}(\dot{V}_{ck}(t)C(t) + V_{ck}(t)\dot{C}(t))R^{k+1}. \tag{3.12}$$

To verify the proposed technique, the following case study is considered. Consider a MEMS resonator oscillating at ω Hz and a ϕ Hz frequency signal is used to detect its

capacitance. The capacitance's expression is:

$$C(x) = \frac{\varepsilon_0 \varepsilon_r A}{x_0 - x}, \tag{3.13}$$

where x is the displacement of the movable electrode, ε_0 is the permittivity of free space, ε_r is the relative permittivity of the dielectric material between the two electrodes, A is the overlapping surface area of the electrodes and x_o is the initial gap between the two electrodes.

Assuming that the MEMS device is stimulated by an external mechanical sinusoidal displacement input:

$$x = y_1 sin(\omega t), \tag{3.14}$$

where y_1 is of a small magnitude and ω is the stimulating frequency. The capacitance, $C(t)$ is:

$$C(t) = \frac{\varepsilon_0 \varepsilon_r A}{x_0 - y_1 sin(\omega t)}, \tag{3.15}$$

If $x \ll x_0$, the capacitance's expression can be approximated as [22]:

$$C(t) = y_0 + y_1 sin(\omega t), \tag{3.16}$$

The derivative of $C(t)$ is:

$$\dot{C}(t) = y_1 \omega cos(\omega t), \tag{3.17}$$

where y_0 is the time-invariant part when $x = 0$:

$$y_0 = \frac{\varepsilon_0 \varepsilon_r A}{x_0}. \tag{3.18}$$

Let the AC power source, V_s, be equal to:

$$V_s(t) = A_s sin(\phi t). \tag{3.19}$$

The derivative of $V_s(t)$ with respect to time is:

$$\dot{V_s}(t) = A_s \phi cos(\phi t). \tag{3.20}$$

To obtain the initial solution, V_{c0}, the linear circuit's steady state solution is used:

$$V_{c0}(t) = A_s sin(\omega t). \tag{3.21}$$

Based on (3.9) and (3.17), I_{c1} can be obtained:

$$\begin{aligned} I_{c1}(t) &= \phi A_s cos(\phi t)(y_0 + y_1 sin(\omega t)) \\ &+ A_s sin(\phi t) y_1 \omega cos(\omega t). \end{aligned} \tag{3.22}$$

Then, V_{c1} can be found using (3.10):

$$\begin{aligned} V_{c1}(t) &= -I_{c1}(t)R \\ &= -R(\phi A_s cos(\phi t) y_1 sin(\omega t) + A_s sin(\phi t) y_1 \omega cos(\omega t)). \end{aligned} \tag{3.23}$$

Similarly, V_{c2} can be calculated recursively

$$\begin{aligned} V_{c2}(t) &= R^2 A_s(-y_0 \phi^2 sin(\phi t) - y_1 \phi^2 sin(\phi t) sin(\omega t) \\ &+ y_1 \phi \omega cos(\phi t) cos(\omega t) + y_1 \phi \omega cos(\phi t) cos(\omega t) \\ &- y_1 \omega^2 sin(\phi t) sin(\omega t))(y_0 + y_1 sin(\omega t)) \\ &- R^2 A_s(\phi cos(\phi t) y_1 sin(\omega t) + sin(\phi t) y_1 \omega cos(\omega t)) \omega y_1 cos(\omega t) \end{aligned} \tag{3.24}$$

3.2.2 Iterative Approxiamted Solutions for Time-variant Inductor with a DC Source

To obtain the approximated solution for time-variant inductor with a DC Source, the first step is to model the variable inductor as:

$$L(t) = L_0 + L_1(t), \tag{3.25}$$

where L_0 is time-invariant, and $L_1(t)$ is the time-variant part that must be less than L_0 to ensure that $L(t)$ is always positive. The derivative of (3.25) is:

$$\dot{L}(t) = \dot{L}_1(t). \tag{3.26}$$

The first step is analysis of this circuit ignoring the effect of $L_1(t)$ and considering it as a linear circuit. The purpose of this step is to obtain the steady state of $I_l(t)$, denoted by $I_{l0}(t)$. Then $V_{l1}(t)$ can be calculated by (3.5) and (3.25). Correspondingly, the $I_{l1}(t)$ term is:

$$I_{l1}(t) = -V_{l1}(t)/R_n. \tag{3.27}$$

Additional terms can be caculated recursively:

$$V_{lk+1}(t) = \dot{I}_{lk}L(t) + I_{lk}\dot{L}(t). \tag{3.28}$$

The $I_{lk}(t)$ shall be solved using as many terms as required to obtain sufficient precision. The overall equation of $I_l(t)$ is:

$$I_l(t) = I_{l0}(t) + \sum_{k=0}^{\infty}(\dot{I}_{lk}(t)L(t) + I_{lk}(t)\dot{L}(t))/R_n^{k+1}. \tag{3.29}$$

Solenoids are electro-magnetic actuators, which have a similar formula to inductance motors. A solenoid has a variable inductor and can be modelled as:

$$L(x) = \frac{\mu_0 \mu_1 N^2 A}{d + x_0 - x}, \tag{3.30}$$

where x is the displacement of the armature, μ_0 is the permeability of free space, μ_1 is the relative permeability of the dielectric material between the coil and armature, A is the cross-sectional area of the core, N is the number of coils, x_o is the initial air gap between the armature and the backside of the frame, and d is the additional initial air gap related to the solenoid's geometry [72].

In the applications of magnetic field measurement [4] and motion control [15], the solenoid is stimulated by a external mechanical sinusoidal displacement input:

$$x = y_1 sin(\omega t), \tag{3.31}$$

where y_1 is of a small magnitude and ω is the stimulating frequency. If $x \ll x_0$, the inductor's expression can be approximated as:

$$L(x) = y_0 + y_1 sin(\omega t), \tag{3.32}$$

where y_0 is the time-invariant part when $x = 0$:

$$y_0 = \frac{\mu_0 \mu_1 N^2 A}{d + x_0}. \tag{3.33}$$

The derivative of $L(t)$ is:

$$\dot{L}(t) = y_1 \omega cos(\omega t). \tag{3.34}$$

Considering the Norton equivalent circuit model for the solenoid circuit with a constant current source I_{sc}, then

$$I_{l0}(t) = I_{sc}. \tag{3.35}$$

Based on (3.34) and (3.35), apply (3.5) to obtain v_{l1}:

$$v_{l1} = I_{sc} y_1 \omega cos(\omega t). \tag{3.36}$$

Then I_{l1} can be found using (3.28)

$$I_{l1}(t) = -v_{l1}/R = -I_{sc} y_1 \omega cos(\omega t)/R_n. \tag{3.37}$$

Then I_{l2}, I_{l3} and I_{l4} can be calculated recursively

$$I_{l2}(t) = -I_{sc} y_0 y_1 \omega^2 sin(\omega t)/R_n^2 + I_{sc} y_1^2 \omega^2 cos(2\omega t)/R_n^2, \tag{3.38}$$

$$\begin{aligned} I_{l3}(t) = (&I_{sc} y_0^2 y_1 \omega^3 cos(\omega t) + 3 I_{sc} y_0 y_1^2 \omega^3 sin(2\omega t) \\ &+ I_{sc} y_1^3 \omega^3 (0.5 cos(\omega t) - 1.5 cos(3\omega t))/R_n^3, \end{aligned} \tag{3.39}$$

$$\begin{aligned} I_{l4}(t) = -((&y_0^2 y_1 cos(\omega t) + 3 y_0 y_1^2 sin(2\omega t) \\ &- y_1^3 (0.5 cos(\omega t) - 1.5 cos(3\omega t))(y_1 cos(\omega t)) \\ &- (-y_0^2 y_1 sin(\omega t) + 6 y_0 y_1^2 cos(2\omega t) \\ &- y_1^3 (-0.5 sin(\omega t) + 4.5 sin(3\omega t)) \\ &(y_0 + y_1 sin(\omega t))) I_{sc} \omega^4 / R_n^4. \end{aligned} \tag{3.40}$$

3.3 Validation

3.3.1 Simulation Study for Time-variant Capacitor with an AC Power Source

To verify the feasibility of the proposed technique, a MATLAB Simulink model was built and analyzed. The values of the series resistance and the capacitance is typically less than 1M Ω and 20 pF, respectively [40][53]. The excitation signal applied to the MEMS devices tends to about ten times greater than the device's resonant frequency, where typical values could be up to 100-200 KHz. Thus, the time-invariant y_0 was 20 pF, while y_1 was 5 pF, the series resistor, R, was 10 kΩ, the mechanical vibration frequency was 2 KHz, the AC voltage source's amplitude was 1 V and its frequency, ϕ, was 100 KHz in this simulation study.

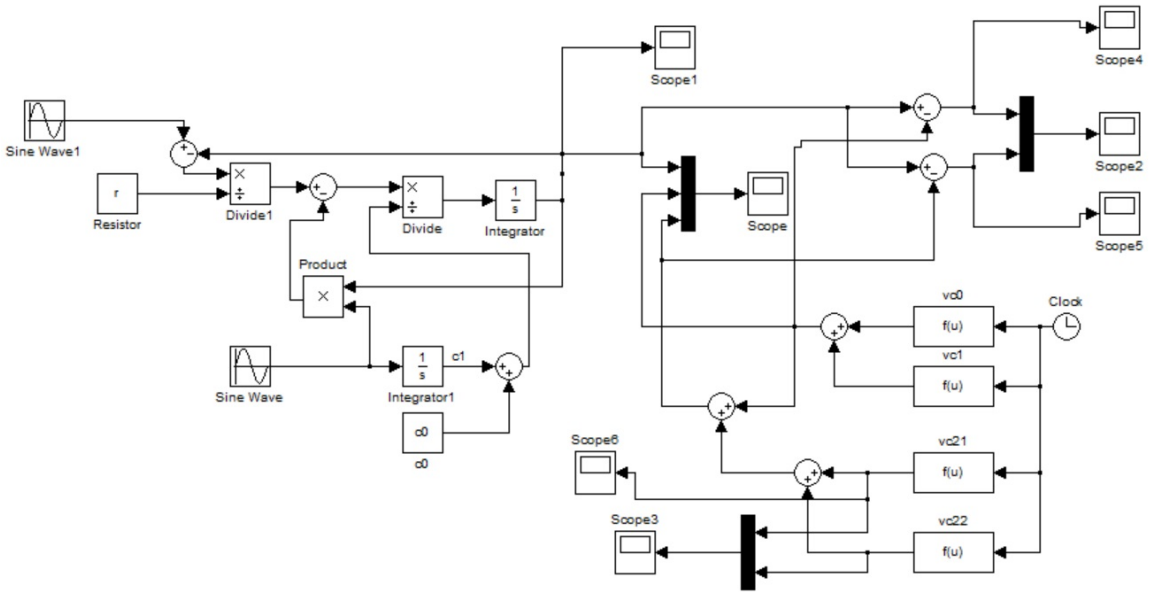


Figure 3.4: Simulink model.

The system (3.32) was solved using a numerical method (BOgacki-Shampine) with a fixed time step of 1×10^{-8} s. The iterative solution with up to two high-order terms was simultaneously computed for comparison. Fig. 3.4 demonstrates the Simulink model. The left side is the system (3.32) and the right side calculates the iterative solutions with different orders.

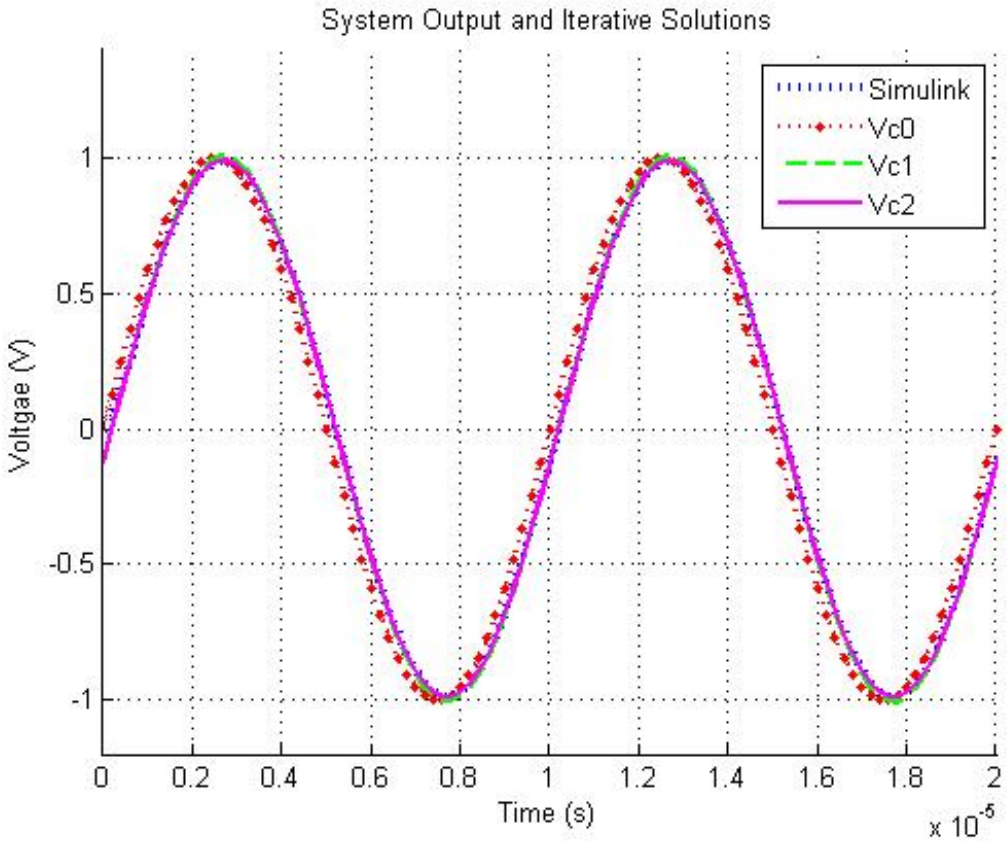


Figure 3.5: Simulation results of the Simulink solution and the iterative solution zoomed in on higher-order terms.

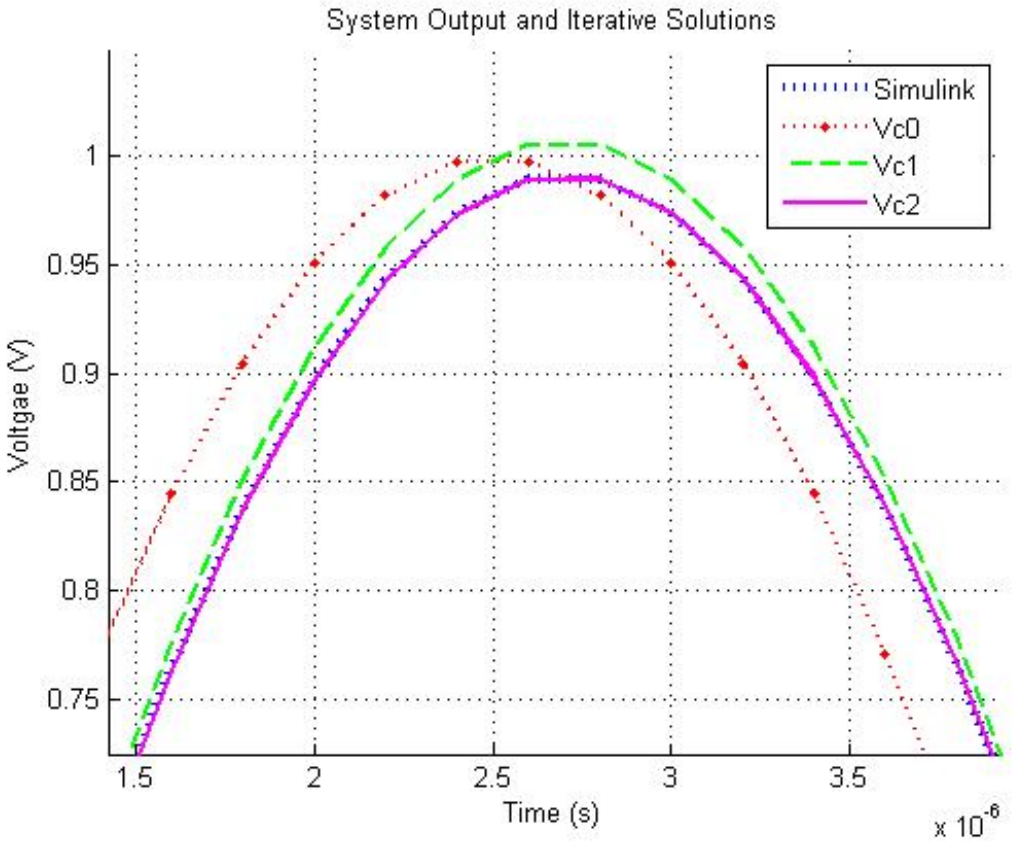


Figure 3.6: Zoomed in simulation results of the Simulink solution and the iterative solution.

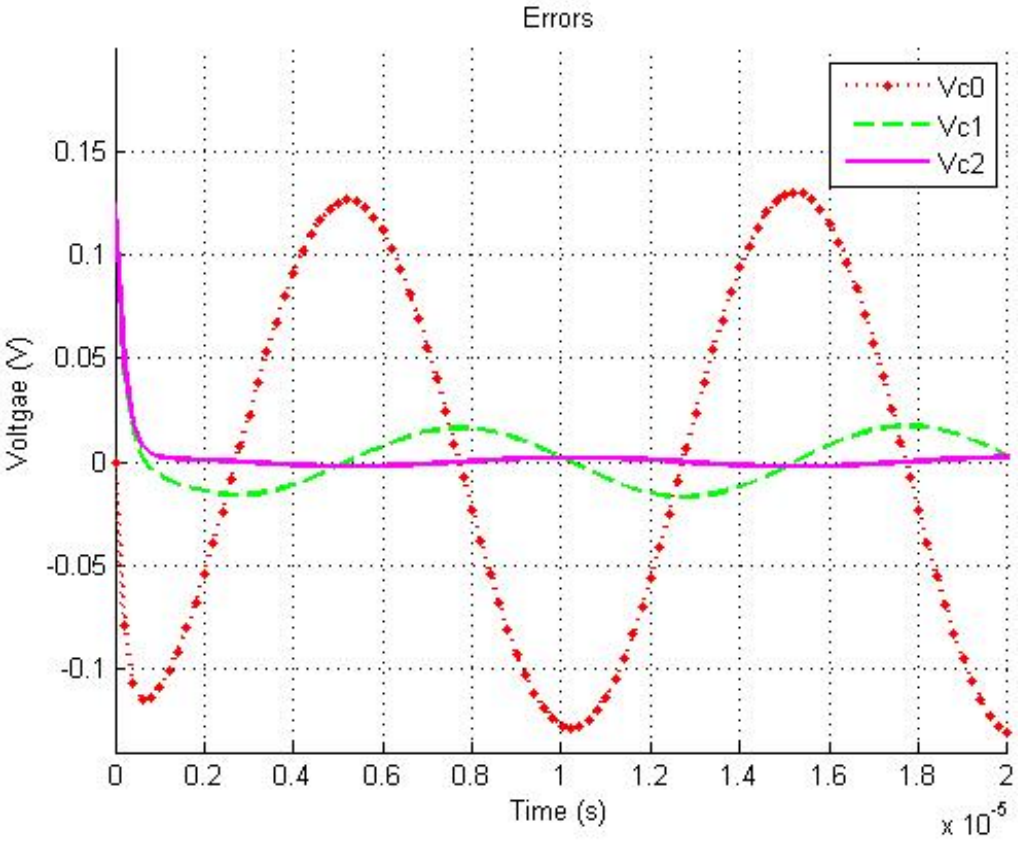


Figure 3.7: Simulation errors between the Simulink solution and the iterative solution zoomed in on higher-order terms.

In Fig. 3.5, Fig. 3.6 and Fig. 3.7 the waveforms with the caption "Simulink" present the output of the Simulink model based on (3.5) and (3.32), the waveforms with caption V_{c0}, V_{c1} and V_{c2} indicate the iterative solutions with different higher-order terms. The waveform V_{c0} is the linear circuit solution.

Fig. 3.5 shows the time response of the system and the analytical solutions with up to two higher-order terms. The results generated by the proposed method match the theoretical result well in the steady state. For the linear circuit solution, it appears that its amplitude was more precise than the solution with V_{c1}, but it has an obvious phase shift which generated more errors. It is clear that the iterative solution with just one term has more error than the result with two terms. Fig. 3.6 presents the comparison zoomed in on different higher-order terms. This figure further demonstrates that adding additional high-order terms can increase the solution's accuracy. Fig. 3.7 shows the errors using different high-order terms. It is shown that the linear circuit analysis yielded the greatest errors with an amplitude that was more than 0.1 V. The solution with one higher-order term produced less error, wtih an amplitude that was 0.03 V. In contrast, the error using two higher-order terms was the smallest, which was less than 0.001 V.

3.3.2 Simulation Study for Time-variant Inductor with an DC Power Source

To verify the feasibility of the proposed technique, a series of simulations was performed. A commercial solenoid was chosen as the target device. Its resistance was 18.7 Ω and its inductance varied between 18.6 mH and 64.8 mH depending on the position of its stroke. A DC power supply was connected with a tuned voltage of 6 V, which indicates that its Norton equivalent current source I_{sc} was set to 0.32 A. The time-invariant y_0 was 43.2 mH while y_1 was 1.05 mH and the stimulating frequency was 20 Hz.

At the same time, the differential equation describing the system (3.32) with the above parameters was built in MATLAB Simulink, and the iterative solution (3.29) with four different high order terms was obtained for comparison. In Fig. 3.8, the left side is the

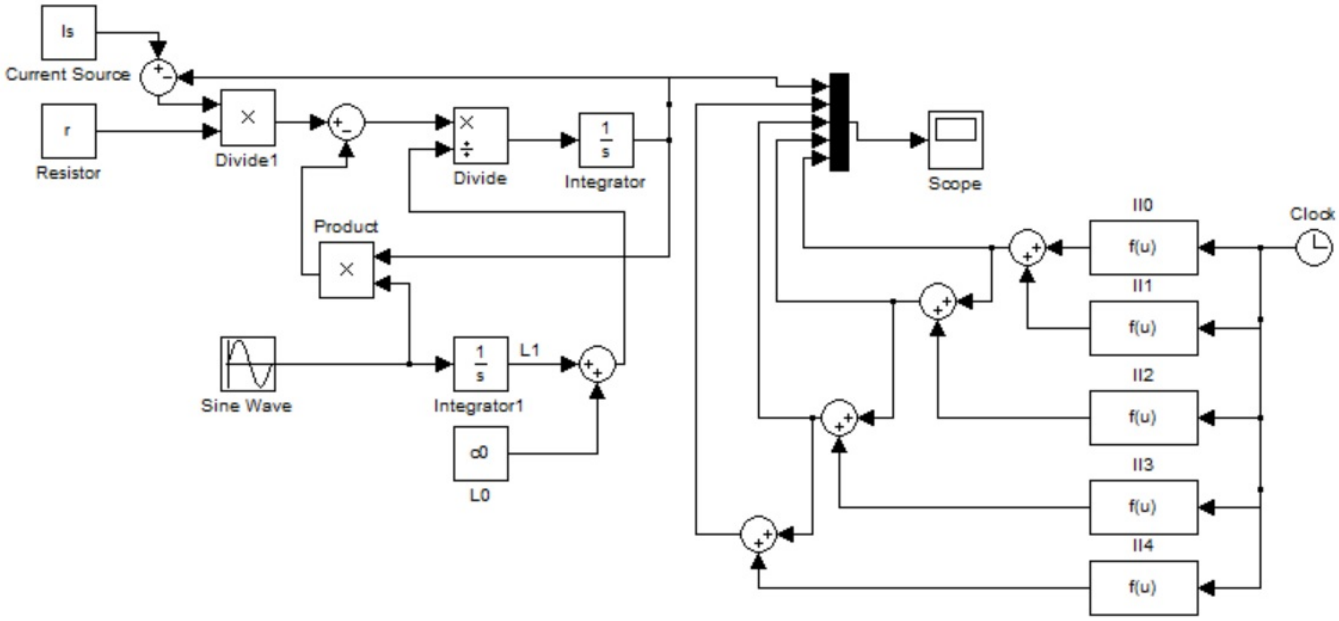


Figure 3.8: The Simulink block diagram for the test.

system (3.32) and the right side is the iterative solutions with different orders. In Fig. 3.9, Fig. 3.10 and Fig. 3.11 the waveform with the caption "Simulink" presents the output of the Simulink model based on (3.5) and (3.32), the waveforms with caption l_1, l_1-l_2, l_1-l_3 and l_1-l_4 indicate the iterative solutions with different higher order terms. Fig. 3.9 shows the time response of the system and the analytical solutions with up to 4th order terms. The results generated by the proposed method match the theoretical result well in the steady state. Fig. 3.10 presents the errors between the different orders configuration's analytical solutions and the simulated system. In the worst case, the steady states errors are less than $\pm$ 5 mA with one term and error decreases rapidly with more terms. Fig. 3.11 shows the zoomed in errors in percentage scale. It is shown that the final steady errors are less than $\pm$ 3% with one term, and less than $\pm$ 0.1% with four terms. All three figures demonstrate the increase in precision with additional higher order terms.

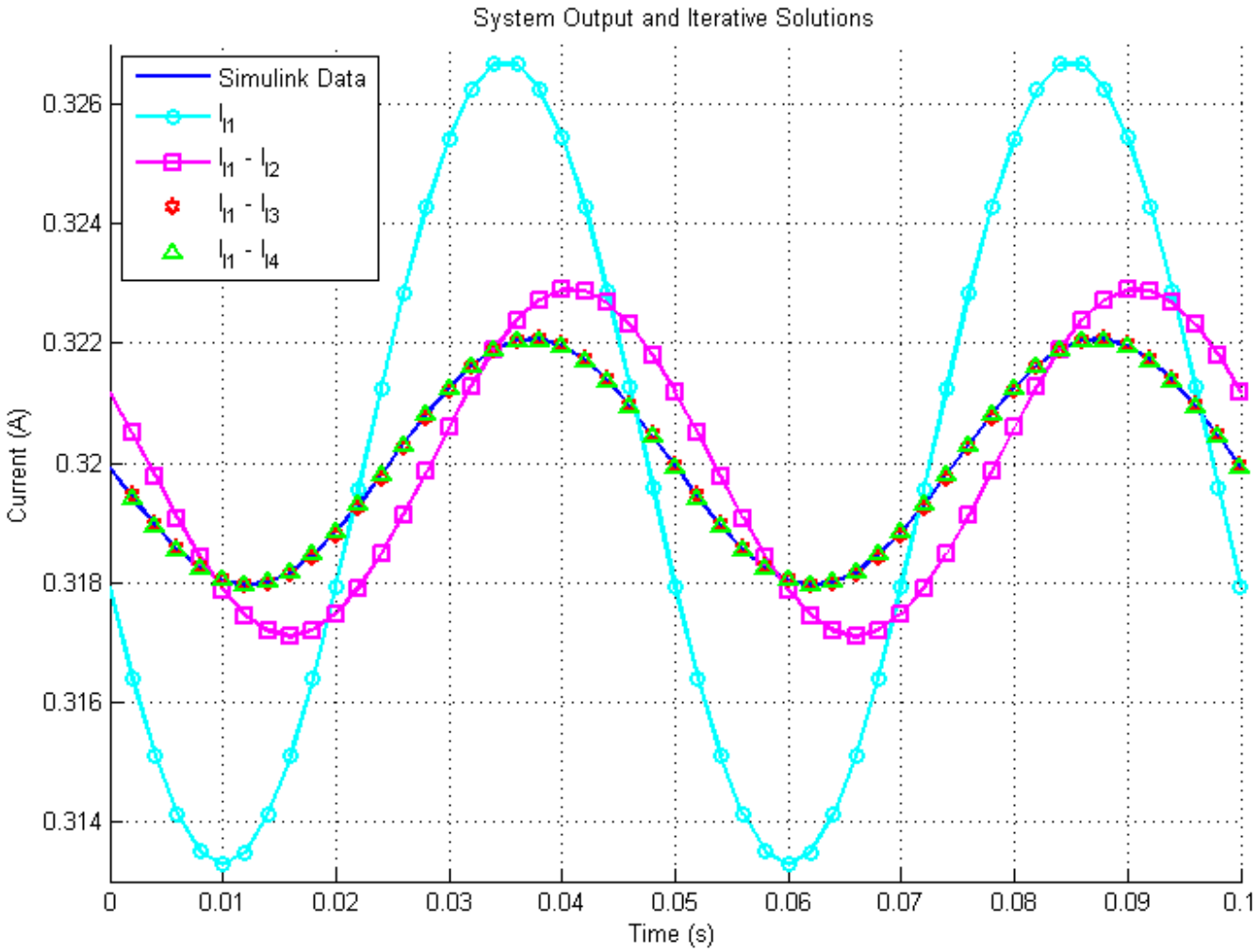


Figure 3.9: Simulation output and iterative solutions.

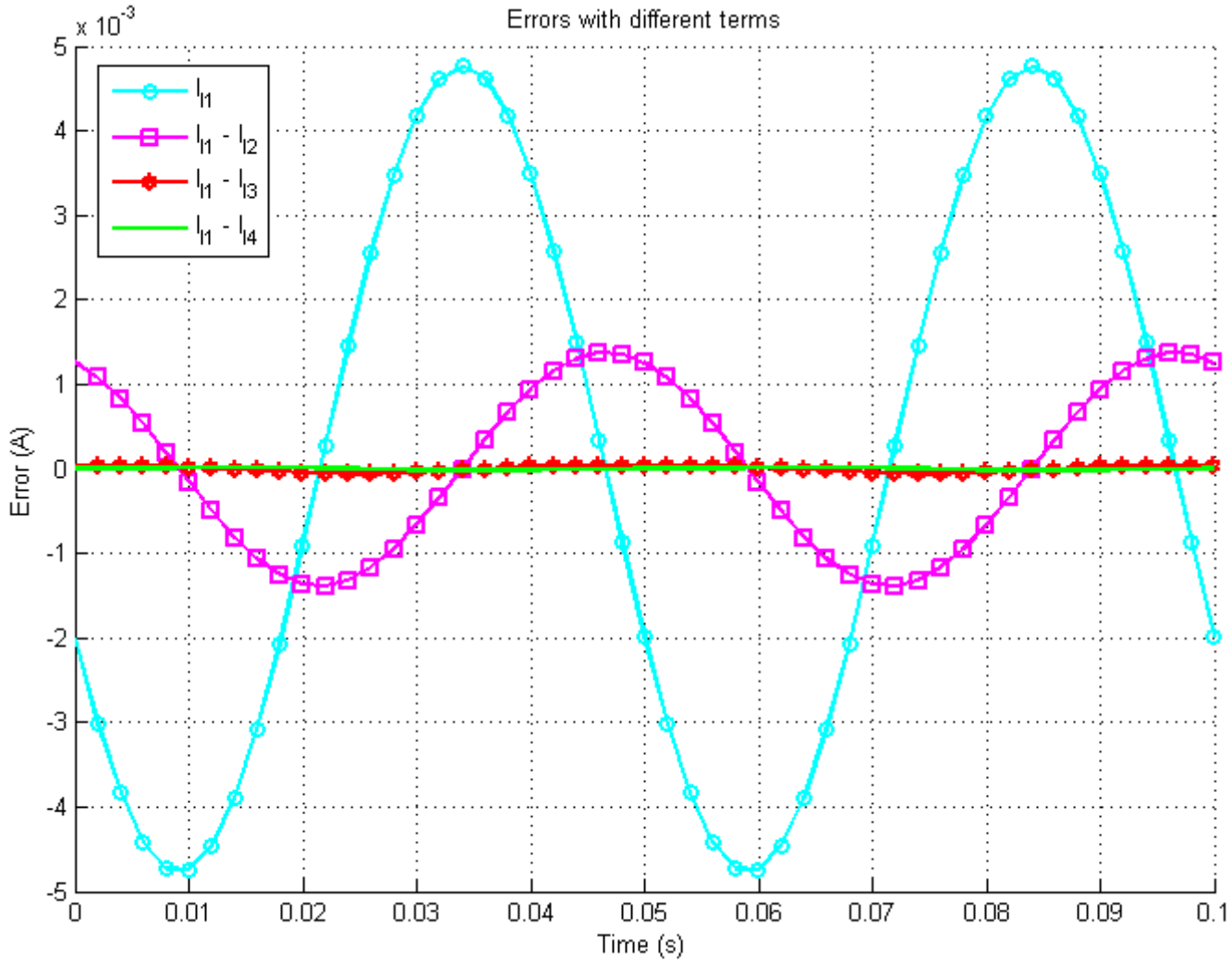


Figure 3.10: Simulation errors between the Simulink solution and the iterative solution.

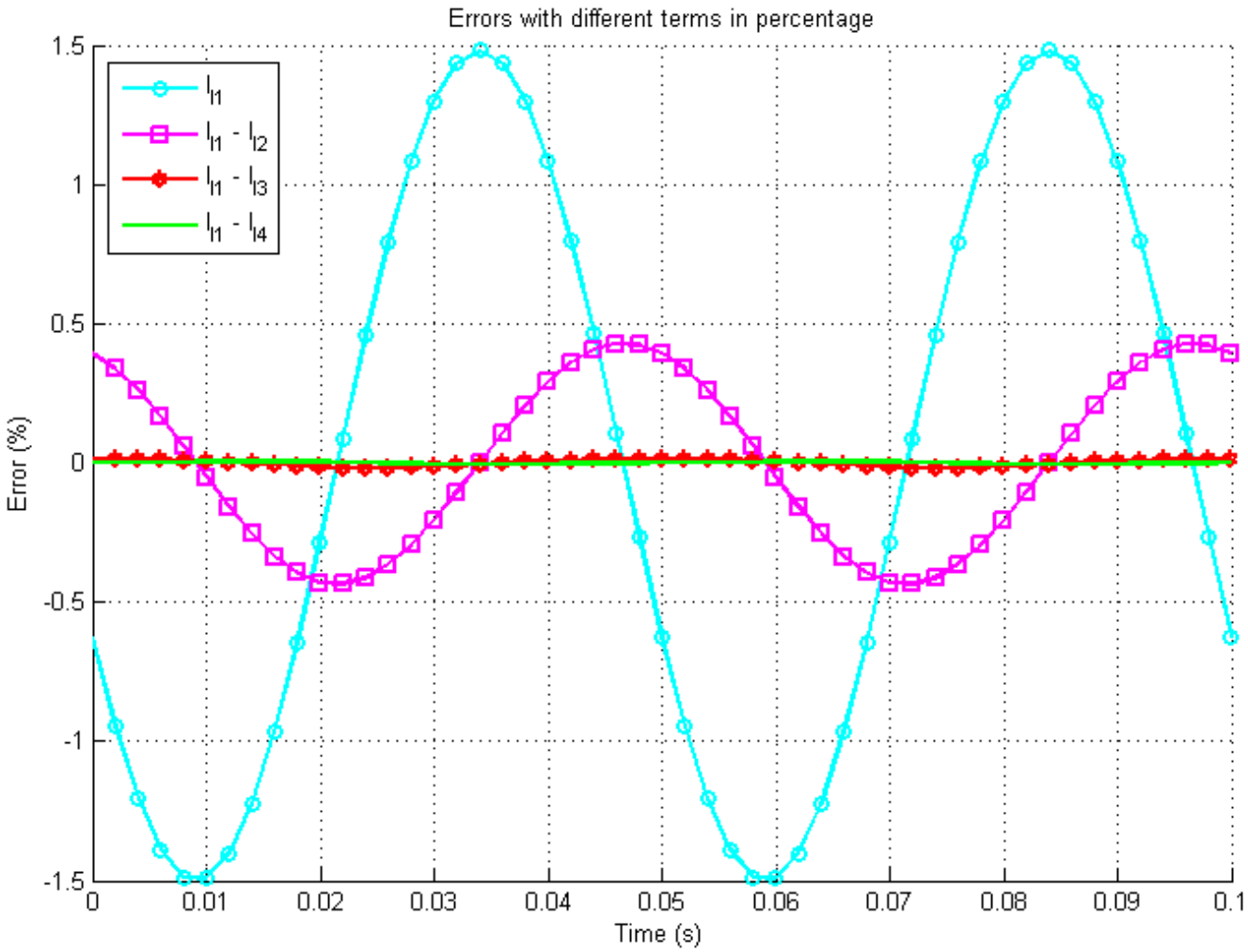


Figure 3.11: Simulation errors between the Simulink solution and the iterative solution by percentage.

3.3.3 Experimental Study for Time-variant Inductor with an DC Power Source

The solenoid was fixed on a LDS800 - 440 large shaker, which could generate the sinusoidal input stimulation as shown in Fig. 3.12. The current actually through the solenoid was measured using a LEM LTS 6 -NP current transducer and sampled using National Instrument's A/D borad numbered 9223. A low pass filter was applied to decrease the noise level, because the switching power supply was noisy and the current transducer was imperfect.

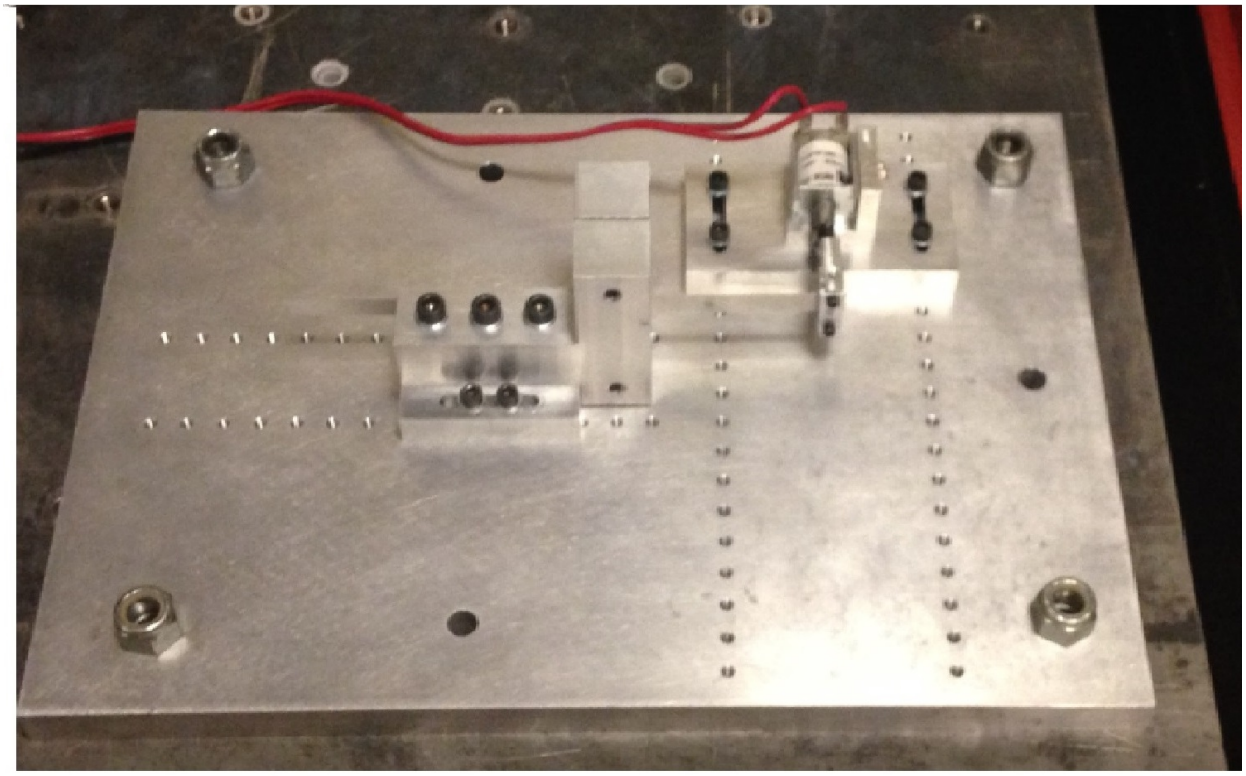
Fig. 3.13 shows the time response of the system under test and the analytical solutions with up to 4th order terms. The experimental results match the theory well. Fig. 3.14 presents the errors between the different orders configuration's analytical solutions and the simulated system. In the worst case, the steady states errors are less than $\pm$ 4.5 mA with one term. Fig. 3.15 shows the zoomed in errors in percentage scale. It is shown that the final steady errors are less than $\pm$ 1.5% with one term and less than $\pm$ 0.5% with four terms. Compared to simulation studies, due to measurement uncertainty and noise, the experimental data is less precises than the results predicted by computer modeling, but they still match the theory well and prove again that more terms can enhance the precision effectively. These results demonstrate that the proposed method can be used to predict the states of a circuit with a time-variant inductor and has the potential to improve the prediction model in predictive controllers.

3.4 Summary

An iterative analysis technique was proposed to solve a MEMS device's nonlinear circuit consisting of a time-variant capacitor and an AC power source connected to the MEMS device through a resistive network. A simulation study demonstrated that this method provides a more accurate solution compared to regular linear circuit analysis. The solution's accuracy can be increased by adding additional higher-order terms.

A nonlinear analysis technique was proposed for obtaining an approximate closed form model for time-variant inductor based systems consisting of the inductor, DC sources and

(a)



(b)

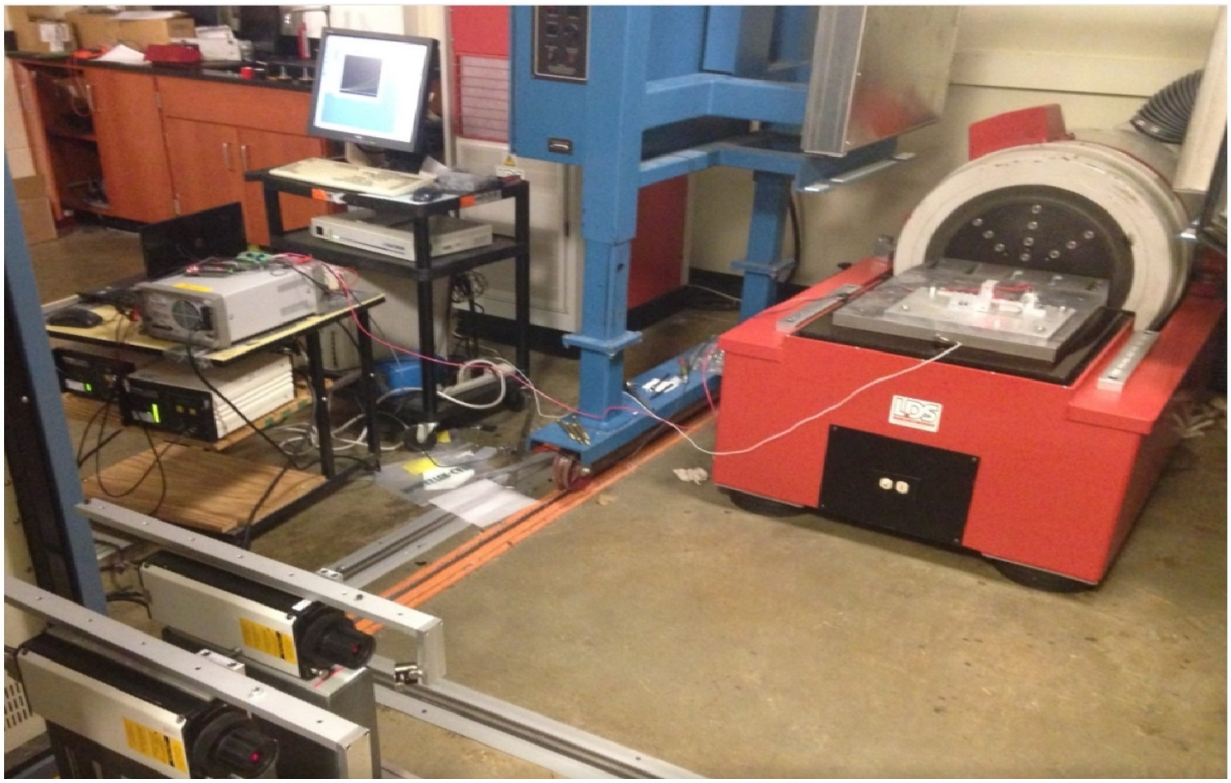


Figure 3.12: The test setup: (a) A solenoid to be tested mounted on a fixture. (b) The solenoid and fixture mounted on the vibration test system.

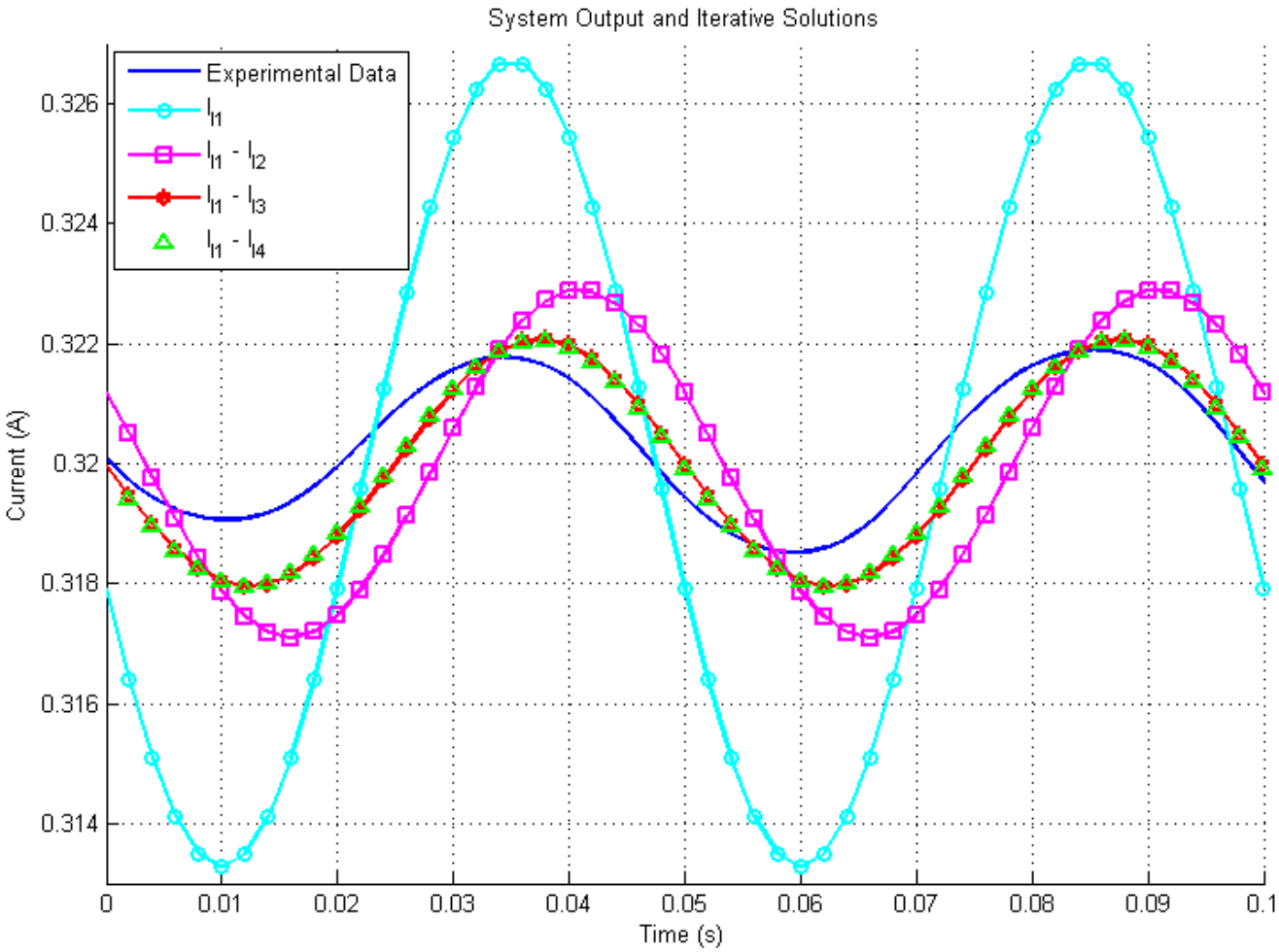


Figure 3.13: Experimental output and iterative solutions.

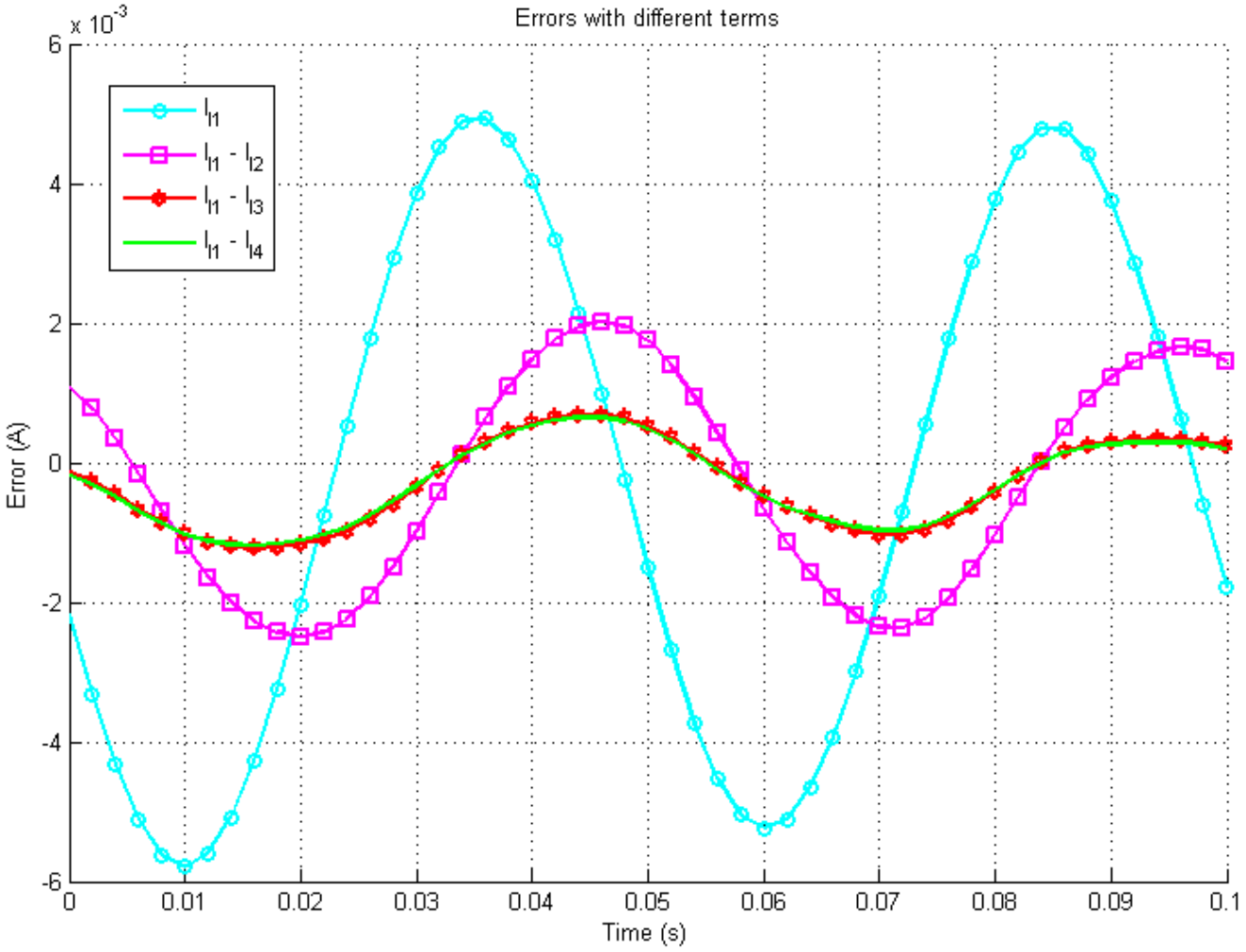


Figure 3.14: Errors between the experimental data and the iterative solutions.

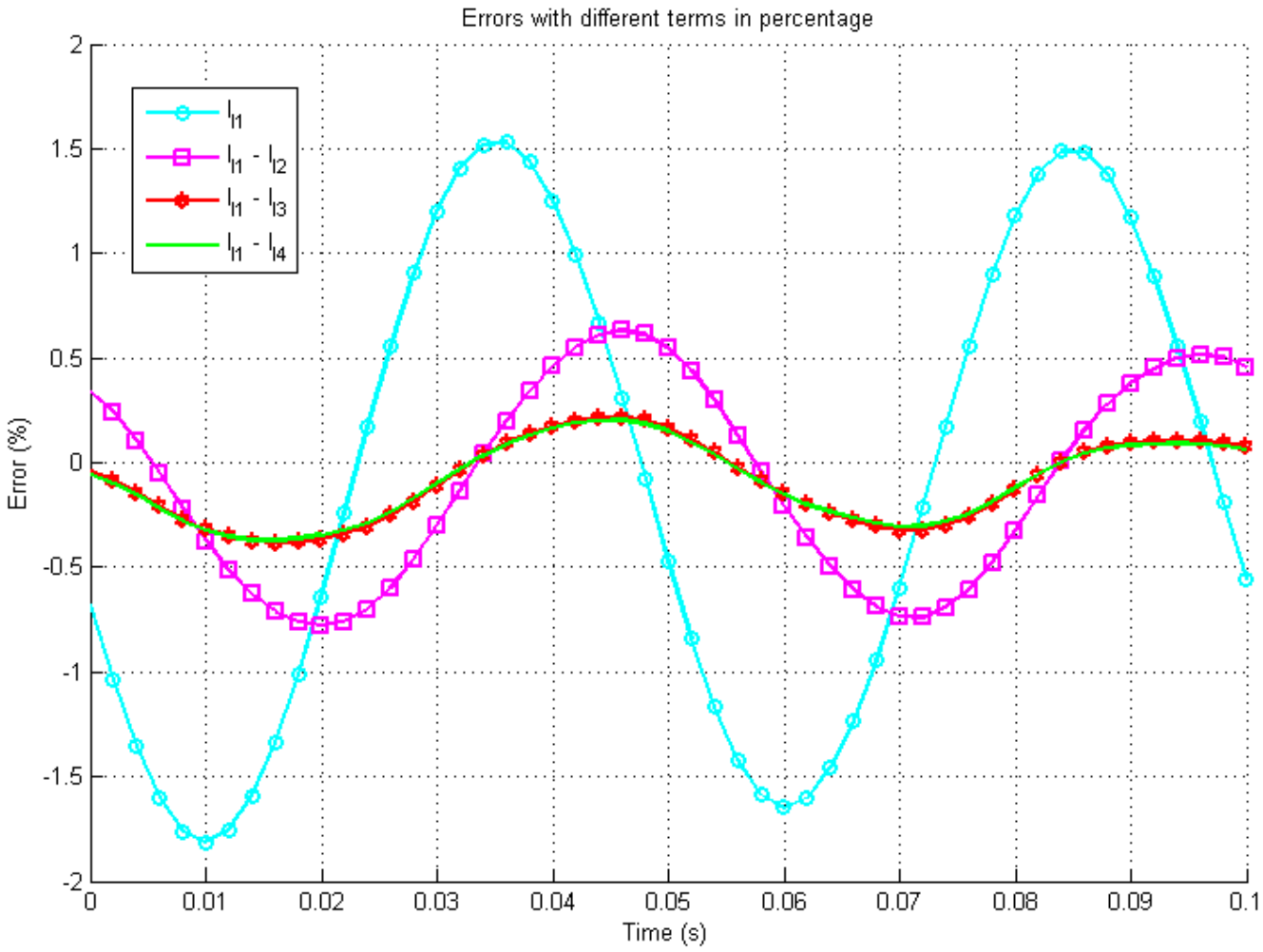


Figure 3.15: Errors between the experimental data and the iterative solutions by percentage.

resistors. First, the Norton equivalent circuit model is obtained. Then the approximate closed form solution for the current through the inductor is obtained using an iterative analysis approach. A case study with simulation and experimental verification demonstrated the technique‘s effectiveness in modelling the system. The results using the proposed technique compared to the numerical differential equation solution resulted in an error of less than 0.1% in simulation and 0.5% in experiments with a four term solution. Including additional higher order terms will further enhance the accuracy. This method is practical and straight forward to implement, and can improve the applications which contain time-variant inductors.

Chapter 4

Disturbance Rejection Control Techniques

4.1 Problem Statement

The open loop stable range of motion for a PPA is x, where $0 \leq x < x_0/3$ and x_0 is the distance between the two electrodes when the applied voltage is 0V. Attempting to further increase the electrode displacement by increasing the applied voltage will result in an unstable condition where the two electrodes snap into contact. Thus, feedback control techniques are desired to extend the PPAs' stable range.

Solenoids are also affected by the same snap-in problem, which was discussed in chapter 2. Though applications like valves [23] that use solenoids as on-off switch components are focused on whether they have enough power to drive the system, other applications like positioners [12] are concerned with position control. One potential drawback to this type of actuator is its limited stable travel range, which is less than 1/3 of the full range of motion. In many harsh environments, external disturbances, such as mechanical vibration, also adversely affect the performance of these actuator systems.

Many techniques have been investigated to extend the stable range of motion of these types of actuators, including: nonlinear magnetic field mapping [45], a dual solenoid configuration [78], pulse-width modulation (PWM) control [70], sliding mode control [58] and adaptive control [56]. However, these proposed solutions have certain limitations: (a) they require prior electrical potential field or magnetic field distribution knowledge, (b) they suffer from time delay in feedback signals, (c) they result in additional hardware costs, (d) they require complex implementation, and (e) they are sensitive to external disturbances.

In this chapter, an input-output linearization control technique with a velocity estimator and a PI controller are proposed to extend the stable range and to reject external disturbances. The proposed method requires knowledge of the initial air gap of the PPA/solenoid, which is easier to measure compared to the electrical potential field or magnetic field. The effect of time delay is analyzed and the performance is improved under the time delay and external disturbances. The controller is implemented using an inexpensive microcontroller. The velocity estimator's structure is straightforward to implement in a microcontroller.

4.2 System Modeling

4.2.1 Uncertain Time Delay in PPAs

Consider a PPA with a series resistor, which is a typical configuration. The system's dynamics can be described by:

$$\begin{aligned} m\dot{x_2} &= -cx_2 - kx_1 + \frac{\varepsilon_0 \varepsilon_r A V_c^2}{2(x_0 - x_1)^2} \\ \dot{V_c} &= \frac{1}{c(x_1)}(-\frac{V_c}{R} - \frac{V_c \varepsilon_0 \varepsilon_r A}{2(x_0 - x_1)^2} x_2 + \frac{V_s}{R}). \end{aligned} \tag{4.1}$$

The planar electrodes are arranged in a parallel configuration with an overlapping surface area, A, and a rest distance between them which is defined as x_0. The bottom electrode is fixed in space and the top electrode is allowed to move toward or away from the bottom electrode. x_1 is the displacement of the device; x_2 is the velocity; m is the mass of the movable electrode. The system's suspension system constrains the motion of the top electrode and is modeled by the system spring constant, k; c is the damping ratio. ε_0 is the permittivity of vacuum and ε_r is the permittivity of the dielectric material between the two electrodes; R is a series resistor to prevent shorting the power supply if the two electrodes come into contact, which is a typical configuration for PPAs; V_s is the power source and V_c is the voltage across

the PPA; $c(x)$ is the actuator's variable capacitance which is modeled as:

$$c(x_1) = \varepsilon_0 \varepsilon_r A/(x_0 - x_1). \tag{4.2}$$

Since $c(x_1)$ is a variable capacitor, the dynamics of the circuit in Fig. 2 are modeled as:

$$V_S(t) - V_c(t) = (\dot{V}_c(t)C(x_1) + V(t)_c\dot{C}(x_1))R. \tag{4.3}$$

According to (4.1) and (4.2), the electrical and mechanical dynamics are coupled with each other, which makes solving (4.3) difficult. To evaluate the behavior between the input voltage and actual current in the circuit, a linear circuit analysis is applied first. It assumes that the variable inductor $C(x_1)$ has a constant value

$$C_1 = C(x^*) \tag{4.4}$$

around an equilibrium point x^*. Thus, (4.3) can be simplified as:

$$V_S(t) - V_c(t) = \dot{V}_c(t)C(x_1)R. \tag{4.5}$$

Its transfer function can be described by:

$$\frac{V_c(s)}{V_s(s)} = \frac{1}{RCs + 1}. \tag{4.6}$$

The transfer function indicates that there is always a time delay between the voltage source V_s and the voltage V_C that is actually applied on the PPA. According to linear system theory, the settling time t_s of this first order system (4.6) is:

$$t_s = 4RC_1. \tag{4.7}$$

However, this settling time approximation is only valid when the mechanical motion is sufficiently small around the equilibrium point x^*. Otherwise, the settling time is uncertain. Substituting the control voltage into (4.1), considering the uncertain time delay effect, the system model can be simplified as:

$$m\frac{d}{dx}(\frac{dx}{dt}) + c\frac{dx}{dt} + kx(t) = \frac{\varepsilon_0\varepsilon_r A V_s(t-\tau)^2}{2(x_0 - x_1)^2} \tag{4.8}$$

4.2.2 Uncertain Time Delay in Solenoids

Consider a solenoid in a spring-mass-damper system, which mathematical description is:

$$m\frac{d}{dt}(\frac{dx}{dt}) + c\frac{dx}{dt} + kx(t) = \frac{\mu_0\mu_r N^2 A I(t)^2}{2(d + x_0 - x(t))^2}, \tag{4.9}$$

where $x(t)$ is the displacement of the armature, μ_0 is the permeability of free space, A is the cross-sectional area of the core, N is the number of the turns of the coil, μ_r is the relative permeability of the magnetic material between the coil and the armature, x_o is the initial air gap between the armature and the back side of the frame and d is the additional initial air gap related to the solenoid's geometry which is much smaller than x_0 in general. Also, m is the proof mass of the armature, k is the stiffness of the spring and c is the system damping coefficient. $I(t)$ is the current through the coil [78][70]. In practice, the input to a solenoid is typically a voltage rather than a current. Given a voltage source, $V(t)$, driving the solenoid, the equivalent circuit model for the solenoid is shown in Fig. 4.1. R is the series resistance of the solenoid coil, $I(t)$ is the current through the solenoid, $L(x)$ is the inductance of the coil that depends on the air gap x [78][70], which is:

$$L(x) = \frac{\mu_0\mu_r A}{d + x_0 - x(t)}. \tag{4.10}$$

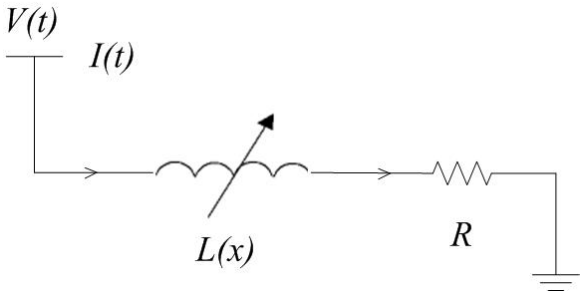


Figure 4.1: Solenoid equivalent circuit

The current determines the magnetic force. Cheung et al. have studied the current dynamics of solenoids driven by a voltage source [13]. Since $L(x)$ is a variable inductor, the dynamics of the circuit in Fig. 4.1 are modeled as:

$$L(x)\frac{dI(t)}{dt} + \frac{dL(x)}{dt}I(t) + I(t)R = V(t). \tag{4.11}$$

According to (4.9) and (4.10), the electrical and mechanical dynamics are coupled with each other, which makes solving (4.11) difficult. To evaluate the behavior between the input voltage and actual current in the circuit, a linear circuit analysis is applied first. It assumes that the variable inductor $L(x)$ has a constant value

$$L_1 = L(x^*) \tag{4.12}$$

around an equilibrium point x^*. Thus, (4.11) can be simplified as:

$$L_1\frac{dI}{dt} + I(t)R = V(t). \tag{4.13}$$

Its transfer function can be described by:

$$\begin{aligned}\frac{I(s)}{V(s)} &= \frac{1}{L_1 s + R} \\ &= \frac{1/R}{(L_1/R)s + 1}.\end{aligned} \tag{4.14}$$

The transfer function indicates that there is always a time delay between the voltage source and the current. According to linear system theory, the settling time t_s of this first order system (4.14) is:

$$t_s = 4((L_1/R). \tag{4.15}$$

However, this settling time approximation is only valid when the mechanical motion is sufficiently small around the equilibrium point x^*. Otherwise, the settling time is uncertain. Substituting the control voltage into (1), considering the uncertain time delay effect and ignoring the parameter d, the system model can be considered as:

$$m\frac{d}{dx}(\frac{dx}{dt}) + c\frac{dx}{dt} + kx(t) = \frac{\beta V(t-\tau)^2}{2(x_0 - x(t))^2}, \tag{4.16}$$

where β is the time-invariant coefficient of the solenoid:

$$\beta = \frac{\mu_0 \mu_r N^2 A}{2R^2}, \tag{4.17}$$

and τ is the uncertainty time delay coefficient which depends on the solenoid's parameters R and $L(x)$.

4.3 Controller Design and Analysis for Solenoids

4.3.1 Ideal input-output linearization

To extend the stable range and performance of the solenoid system, closed-loop control techniques are necessary. An input-output linearization control method and a low voltage controller were demonstrated in [16][20] for a Micro-ElectroMechanical System (MEMS) electrostatic actuator, which possesses dynamics very similar to that of solenoids. Both types of actuators are approximately square law devices. Adopting this method based on the model (1) and using a current source, an input-output linearization law can be introduced

to approximately cancel the nonlinearity:

$$I(t) = (x_0 - x(t))\sqrt{k_c I_c(t)}, \tag{4.18}$$

where k_c is a positive gain on the input and $I_c(t)$ is the new control input, where $I_c(t) > 0$. Therefore the system becomes:

$$m\frac{d}{dt}(\frac{dx}{dt}) = -c\frac{dx}{dt} - kx(t) + \frac{k_c\mu_0\mu_r AN^2}{2}I_c(t), \tag{4.19}$$

which mathematically cancels out the square law dependence on current.

4.3.2 Input-output Linearization Technique with Time delay

However, the system could easily be adversely affected by measurement time delay or controller time delay. Therefore the control law based on (4.18) using a voltage source becomes:

$$V(t) = (x_0 - x(t))\sqrt{k_c V_c(t)}, \tag{4.20}$$

where $V_c(t)$ is the new control input and $V_c(t) > 0$. In this situation, the system dynamics become:

$$m\frac{d}{dt}(\frac{dx}{dt}) = -c\frac{dx}{dt} - kx(t) + \frac{k_c\beta(x_0 - x(t-\tau))^2 V_c(t)}{(x_0 - x(t))^2}. \tag{4.21}$$

Solenoids are macro actuators, which usually have response times slower than electronic circuits. Thus, the time delay is relatively small compared to the mechanical system. Then, using Taylor Series to expand the time delay term, where

$$x(t-\tau) \approx x(t) - \tau\frac{dx}{dt}, \tag{4.22}$$

and neglecting the higher order terms, the system model becomes:

$$\begin{aligned} m\frac{d}{dt}(\frac{dx}{dt}) &= -c\frac{dx}{dt} - kx(t) + k_c\beta V_c(t) \\ &+ \frac{2k_c\beta\tau}{x_0 - x(t)}\frac{dx}{dt}V_c(t) + \frac{k_c\beta\tau}{(x_0 - x(t))^2}(\frac{dx}{dt})^2 V_c(t). \end{aligned} \tag{4.23}$$

Notice that there is a nonlinear damping effect:

$$m\frac{d}{dt}(\frac{dx}{dt}) = -g(x(t), \frac{dx}{dt}, V_c(t)) - kx(t) + k_c\beta V_c(t). \tag{4.24}$$

It's necessary to evaluate the influence of the nonlinear function, g, to the ideal system (4.19). A state variable model and its small signal analysis will be applied. At first, select a group of state variables:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ \frac{dx}{dt} \end{bmatrix}. \tag{4.25}$$

In practice, only the displacement variable $x_1(t)$ is available. Considering this constraint and adding an observation equation that detects the displacement, (4.24) can be rewritten as:

$$\begin{aligned} \frac{dx_1}{dt} &= x_2(t) \\ m\frac{dx_2}{dt} &= -g(x_1(t), x_2(t), V_c(t)) - kx_1(t) + k_c\beta V_c(t) \\ y(t) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \end{aligned} \tag{4.26}$$

At an equilibrium point, the displacement x_{1e}, x_{2e} and the control input V_{ce} satisfy:

$$\begin{cases} kx_{1e} = k_c\beta V_{ce} \\ x_{2e} = 0 \end{cases}. \tag{4.27}$$

According to (4.27), a group of new state variables around a certain equilibrium point is selected as:

$$\begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \\ \delta V_e(t) \end{bmatrix} = \begin{bmatrix} x_1(t) - x_{1e} \\ x_2(t) - x_{2e} \\ V_c(t) - V_{ce}(t) \end{bmatrix}. \tag{4.28}$$

The linearized small signal model is:

$$\begin{aligned} \begin{bmatrix} \delta \frac{dx_1}{dt} \\ \delta \frac{dx_2}{dt} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} + \frac{2k_c\beta\tau}{m(x_0 - x_{1e})} \end{bmatrix} \begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ k_c\beta \end{bmatrix} \delta V_{ce}(t) \\ \delta y(t) &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \end{bmatrix}. \end{aligned} \tag{4.29}$$

For evaluating its influence, it is convenient to examine the transfer function of (4.29):

$$G_1(s) = \frac{X_1(s)}{V(s)} = \frac{k_c\beta/k}{\frac{m}{k}s^2 + \frac{m}{k}(c - \frac{2\beta\tau}{x_0 - x_{1e}})s + 1}, \tag{4.30}$$

which can be compared to the ideal model:

$$G_2(s) = \frac{X_1(s)}{V(s)} = \frac{k_c\beta/k}{\frac{m}{k}s^2 + \frac{m}{k}cs + 1}. \tag{4.31}$$

The actual damping is less than that predicted by the ideal model. As a result, higher inputs could lead the system to oscillate or become unstable. So extra damping needs to be introduced to the system to ensure acceptable performance.

4.3.3 Linearization with Extended State Observer

Damping can be increased by measuring or estimating proof mass velocity and feeding back this term. Directly differentiating the displacement is inaccurate due to measurement noise. Therefore, velocity observers are necessary to obtain a better estimate for feedback. However, considering the system uncertainty, regular observers are difficult to construct. The active disturbance rejection control (ADRC) technique is an alternative approach that can be applied to this kind of problem. It has been demonstrated that extended state observers (ESO) can estimate system states without certain system models [30][75][48]. The key idea of this technique is to consider the nonlinear parts of the system as the external state. Considering that a nonlinear second order system has the form of:

$$\begin{aligned} \frac{dx_1^*}{dt} &= x_2^*(t) \\ \frac{dx_2^*}{dt} &= f^*(x_1^*(t), x_2^*(t), w(t)) + bu(t), \end{aligned} \tag{4.32}$$

where $x_1^*(t)$, $x_2^*(t)$ are the state variables, $w(t)$ are the external disturbances, $f^*(x_1^*(t), x_2^*(t), w(t))$ is a nonlinear function with the variables and the disturbance, b is the input coefficient and $u(t)$ is the control input. J. Han [30] proposed a solution that can eliminate this issue using the measurement combined with a series of nonlinear functions. However, this method's parameters lack guidelines for adjustment. A linear approach was illustrated with tuning methods provided for ADRC and ESO [75][48]. Based on the linear approach, the system is

augmented as:

$$\begin{aligned}
\begin{bmatrix} \frac{dz_1}{dt} \\ \frac{dz_2}{dt} \\ \frac{dz_3}{dt} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \\
y(t) &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix}.
\end{aligned} \tag{4.33}$$

Therefore the estimator's structure becomes:

$$\begin{aligned}
\frac{d\hat{z}_1}{dt} &= \hat{z}_2(t) + L_1(y(t) - \hat{z}_1(t)) \\
\frac{d\hat{z}_2}{dt} &= \hat{z}_3(t) + L_2(y(t) - \hat{z}_1(t)) + bu(t) \\
\frac{d\hat{z}_3}{dt} &= L_3(y(t) - \hat{z}_1(t)).
\end{aligned} \tag{4.34}$$

where $\hat{z}_1(t)$, $\hat{z}_2(t)$ and $\hat{z}_3(t)$ are the estimated value of $z_1(t)$, $z_2(t)$ and $z_3(t)$. In this case, let the displacement, $x_1(t)$, be $z_1(t)$, the velocity, $x_2(t)$, be $z_2(t)$, the external disturbances be $z_3(t)$ and $y(t)$ be the input $u(t)$. L_1, L_2 and L_3 are observer parameters to be chosen which will decide the bandwidth of the estimator. Using this estimated velocity, $\hat{z}_2(t)$ and the external disturbance, $\hat{z}_3(t)$, extra damping and disturbance rejection are introduced to the control law:

$$V(t) = (x_0 - x(t))\sqrt{k_c V_c(t) - \eta_1 \hat{z}_2(t) - \eta_2 \hat{z}_3(t)}, \tag{4.35}$$

where η_1 and η_2 are positive numbers. Hence, the damping is enhanced and external disturbances can be reduced:

$$\begin{aligned} m\frac{d}{dt}(\frac{dx}{dt}) &= -g(x(t), \frac{dx}{dt}, V_c(t)) - kx(t) + k_c\beta V_c(t) \\ &- \eta_1\beta\hat{z}_2(t) - \eta_2\beta\hat{z}_3(t). \end{aligned} \tag{4.36}$$

Because there is a time delay between the controller's command and settling the actual current signal, the control effort with the ESO will also have the time delay effect. η_1 and η_2 can be tuned by starting from small values, then increasing their values until the performance becomes satisfactory.

Based on (4.29), $\hat{z}_2(t)$ with time delay will mathematically decrease the mass of the armature, which will not affect dynamic performance.

4.3.4 Linearization with PI and ESO

For the sake of the system's response time and position tracking, a proportional and integral (PI) controller can be placed ahead of the control law given in (4.35). The overall system diagram with proportional and integral and extended state observer (PIESO) is shown in Fig. 4.2. In this control diagram, r is the reference input, $e_1(t)$ is the error between the displacement and input, $e_2(t)$ is the estimation error between the estimator and measured displacement and $u(t)$ is the actual voltage across the solenoid. The PI controller with respect to $V_c(t)$ is:

$$V_c(t) = K_p e_1(t) + K_i \int e_1(t)dt. \tag{4.37}$$

Thus, (4.35) and (4.37) can be combined together to enchance the system's performance.

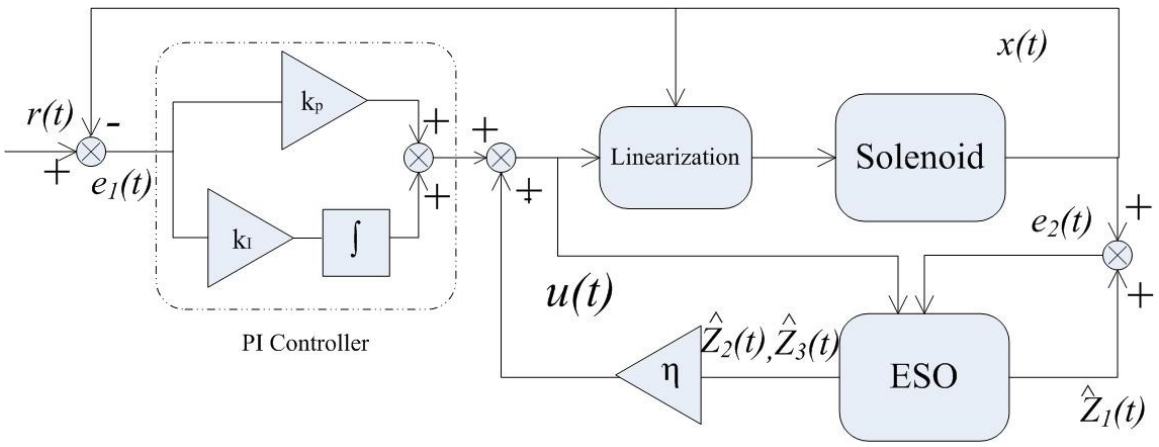


Figure 4.2: Block diagram of the controller with Input-Output Linearization and PIESO

4.3.5 Digital Implementation

Digital controllers have certain advantages: (a) complicated control algorithms can be implemented, (b) state observers are available, (c) they are less sensitive to the harsh environment in which the system must operate and (d) they can be easily updated. The implementation of the control law though a microcontroller requires discrete form. Given the sample rate T, (4.37) becomes:

$$V_c(k) = V_c(k-1) + K_p(e_1(k) - e_1(k-1)) + \frac{K_I}{T}e_1(k), \tag{4.38}$$

and (4.35) can be approximated as:

$$V(k) = (x_0 - x(k))\sqrt{k_c V_c(k) - \eta_1 \hat{z}_2(k) - \eta_2 \hat{z}_3(k)}. \tag{4.39}$$

Since (4.39) requires the estimated value of velocity, $\hat{z}_2(k)$, and external disturbance, $\hat{z}_3(k)$, the discrete form of the ESO is:

$$\begin{aligned}
&\hat{z}_1(k) = \hat{z}_1(k-1) + T(\hat{z}_2(k-1) + L_{d1}(y(k-1) \\
&- \hat{z}_1(k-1))) \\
&\hat{z}_2(k) = \hat{z}_2(k-1) + T(\hat{z}_3(k-1) + L_{d2}(y(k-1) \\
&- \hat{z}_1(k-1)) + bu(k)) \\
&\hat{z}_3(k) = \hat{z}_3(k-1) + T(L_{d3}(y(k-1) \\
&- \hat{z}_1(k-1))).
\end{aligned} \tag{4.40}$$

where L_{d1}, L_{d2} and L_{d3} are parameters to be chosen to adjust the bandwidth of the estimator. The tuning method is introduced by [75]. Equations (4.38), (4.35) and (4.40) are difference equations that are easy to program in microcontrollers.

4.4 System Configuration

The experimental validation system included a solenoid system, a laser vibrometer, a microcontroller, a power supply and an amplifier, which is illustrated as Fig. 4.3. A commercial solenoid, Pontic F421, was configured for the test. Its coil resistance was 22.5 Ω with a maximum voltage input of 15 V. The solenoid was attached to a metal fixture to hold it, and a thin metal cantilever was used as the spring for the actuator system. Mechanical testing of the spring indicated that the spring stiffness was 302.98 N/m, and the damping coefficient was 42.03 kg/s. Considering the detection range of the displacement sensor and the initial force of the solenoid, the total displacement was configured to be 6.96 mm. The natural frequency of the solenoid system was 16.4 Hz, which indicated its bandwidth. At the initial position, the inductance was 45.5 mH, so based on (4.15), in the worst case the time delay was approximately 8 ms.

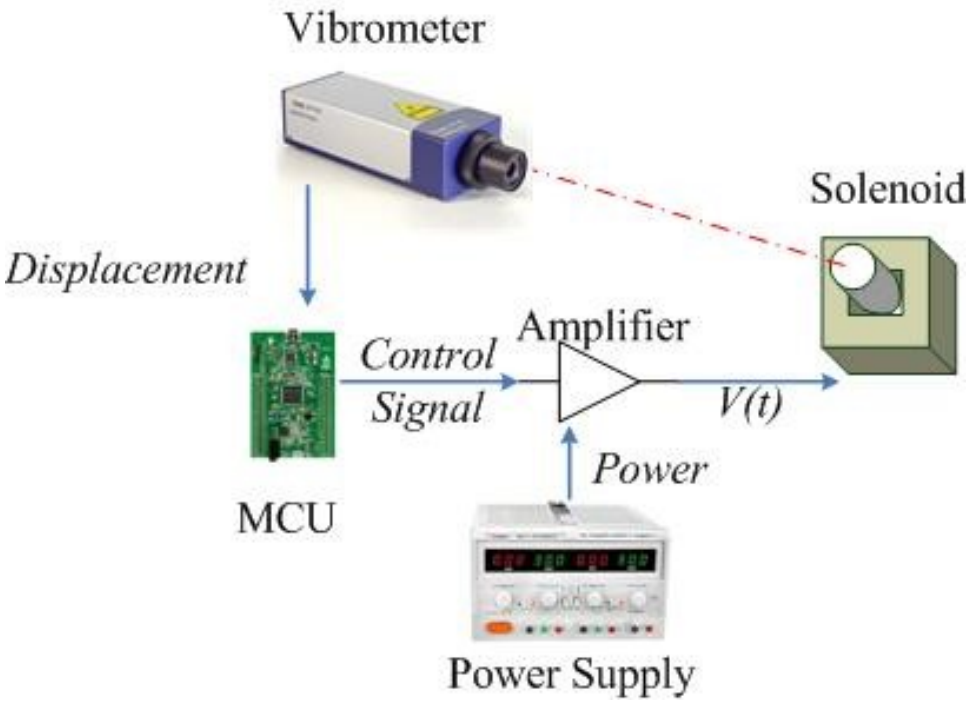


Figure 4.3: System Configuration.

A STM32F407 microcontroller was used for this experiment, which had a 168 MHz core and was assembled with a 12 bit analog-to-digital / digital-to-analog converter. Thus, the solenoid required a 12 V at 0.5 A power supply. A support electronics circuit board with a push-pull amplifier was used to support the microcontroller board and the solenoid system. Proof mass displacement detection was accomplished using a Poly-tech OFV353 laser vibrometer. The vibrometer output was an analog signal with a data rate of 40 kHz and a resolution of 0.1 mm. The control interval T was chosen as 5 ms. First of all, the spring stiffness was examined as shown in Fig. 4.4. Using least squares estimation, the spring constant was determined to be 302.89 N/m. The mass of the plunger was determined to be 22.7 g using a weighing scale. As shown in Fig. 4.5, the transmissibility between the plunger and the fixture was also measured. The natural frequency was 16.7 Hz and the mechanical

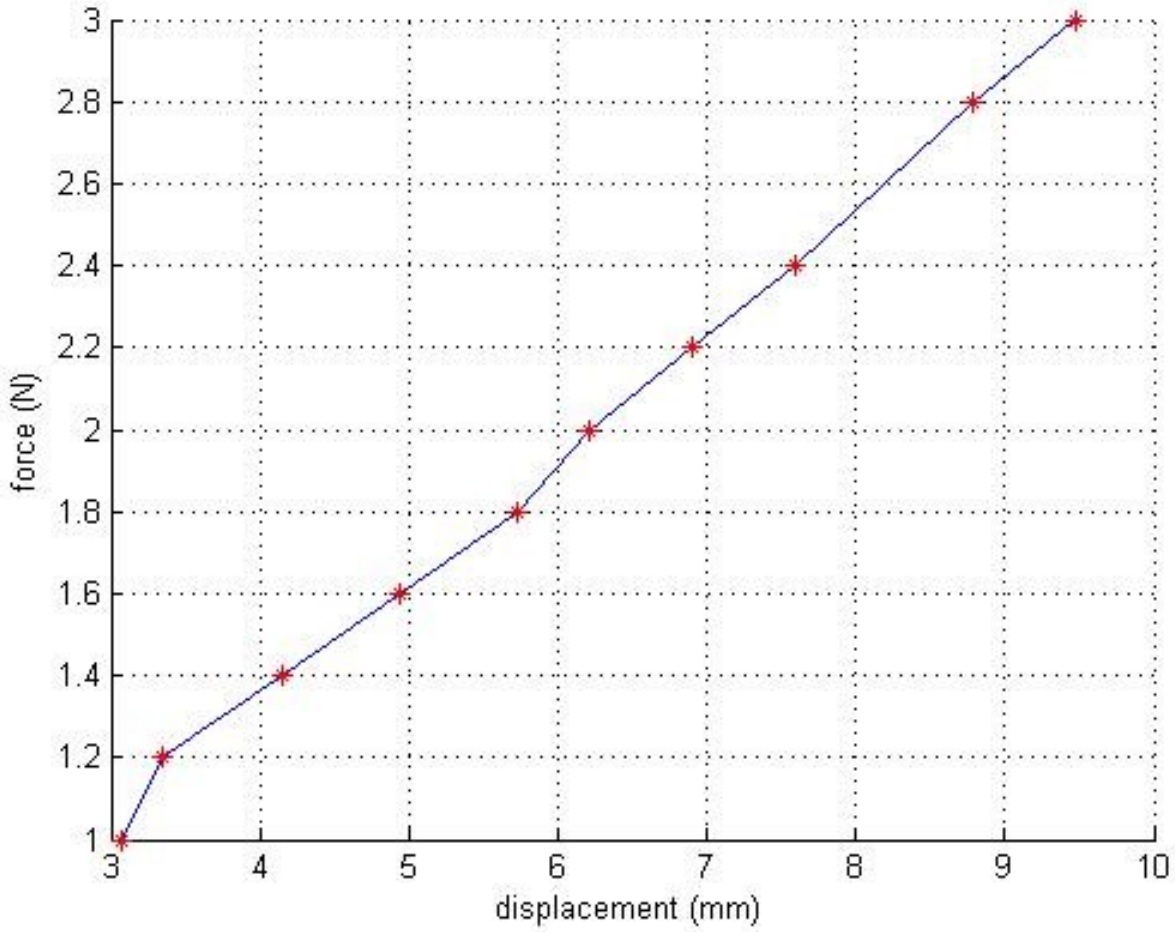


Figure 4.4: The stiffness of the cantilever.

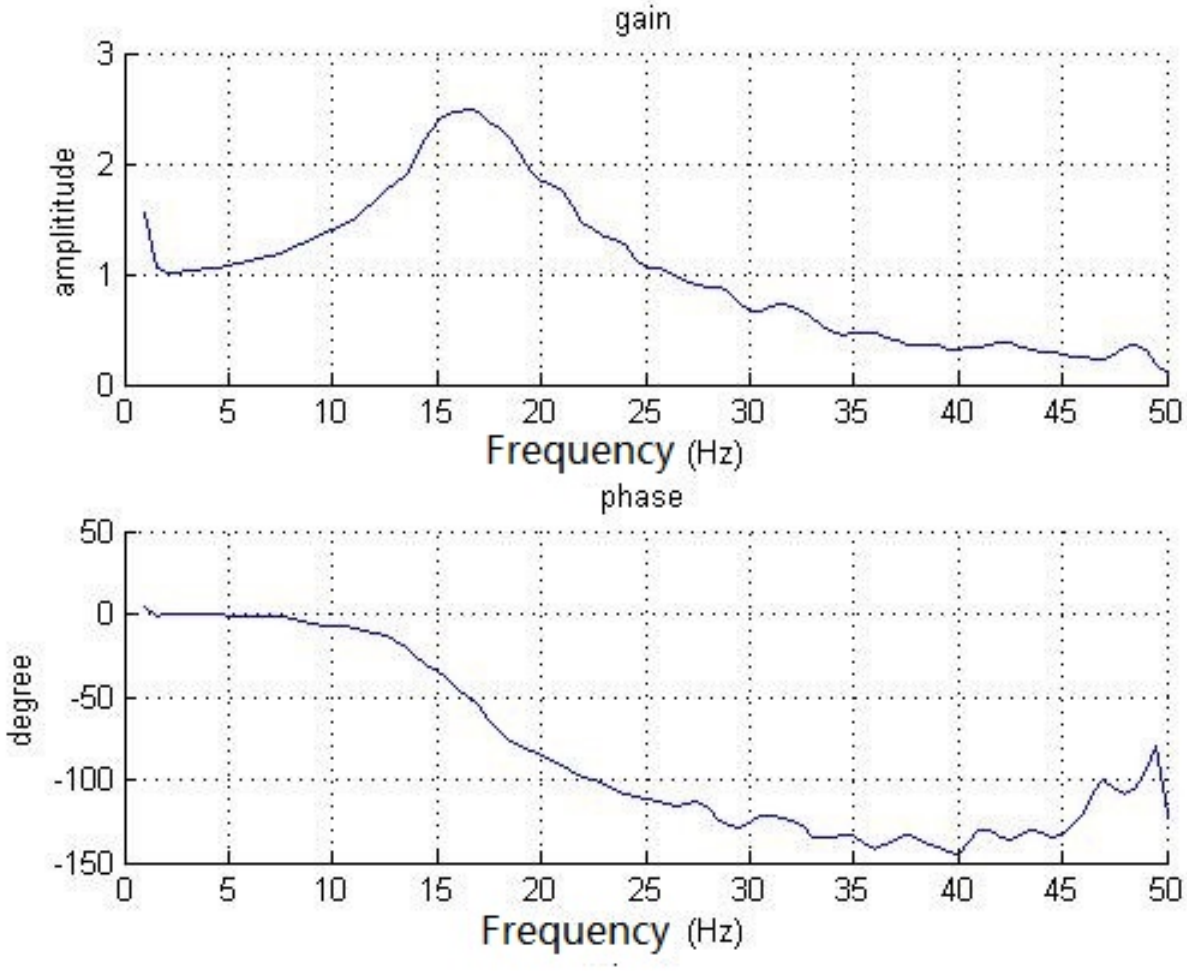


Figure 4.5: The stiffness of the cantilever.

quality factor was 2.45. After system identification, the final model was:

$$22.7 \times 10^{-3}\frac{dx}{d}(\frac{dx}{dt}) + 42.03\frac{dx}{dt} + 302.89x(t) = \frac{2.107 \times 10^{-4} I(t)^2}{(6.65 \times 10^{-3} - x(t))^2}. \tag{4.41}$$

Considering the natural frequency, the desired bandwidth was chosen as $F_c = 16$ Hz. Based on the tuning method given in [48][54], the gains of the ESO were:

$$L_{d1} = 0.3758, L_{d2} = 12.0641, L_{d3} = 122.8657. \tag{4.42}$$

4.5 System Simulation

Prior to experimental validation, a system simulation was performed using MATLAB SIMULINK. The target actuator is the configured one above. First, the time delay effect on the input-output linearization was verified. The original system had a maximum stable range of 2.3 mm in theory, so a set point of 2 mm was chosen to evaluate the system. Then the PIESO controller was simulated to extend its stable range and to follow the settling point. After that, the PIESO controller's performance under external disturbances was evaluated using sinusoidal signals to model the disturbances. The results are given and discussed in the experimental validation section.

4.6 Experimental Validation

A series of experiments were performed using the hardware in section 4. Considering the bandwidth and numerical resolution, the control rate was chosen to be 200 Hz. Test plans were the same as in the simulation, which included the time delay effect and the PIESO controller with and without disturbances. The sinusoidal disturbances were vibrational and were generated by a LDS-850 large shaker, which is shown in Fig. 4.6.

4.6.1 Time Delay Effect

Fig. 4.7 demonstrates the time delay effect of the damping. In this figure, the curves are the step responses which occur at time 0 s. Simulation results indicate that the original system is overdamped, there is no overshoot and it reaches steady state in less than 0.1 s. By contrast, the IOL applied with time delay reduces the damping coefficient so that its overshoot can be larger than 75% of the steady state. When ESO introduces the extra damping, the system converges rapidly within 0.1 s and only one overshoot peak is smaller than 40% of the set point. Results from simulation and experiments support the point of view that time delay can decrease the damping with IOL. The original damping led to a fast convergence time and small overshoots and IOL decreased it, which took more than 0.6 s to reach the final state. The PIESO increased the damping so that the number of overshoot peaks was much less than IOL and it converges within 0.15 s. Besides the time delay effect, the measured initial gap is not perfectly known, which cannot totally cancel the driven force's nonlinearity that also affects the feedback controller's performances.

4.6.2 PIESO Without Disturbances

This test verified the effectiveness of PIESO to extend the solenoid's travel range and to follow the reference input. In practice, a maximum stable position of 5 mm was achieved, which reached 77.6% of the total range. Also considering that the maximum stable range of the open loop system was 2.2 mm, the set points were 2.2 mm, 4 mm and 5 mm, respectively. The step responses occurring at 0.5 s are illustrated in Fig. 4.8. Results from both simulations and experiments show that the system can achieve the desired point without error. In the simulation, responses had overshoots and took more time (around 0.3 to 0.4 s) to converge. The experimental results ran faster (around 0.2 to 0.3 s), but suffered from inaccurate measurement, which led to chattering around the desired position. The trace of the reference input equaling 2 mm in the experiment is an example. A further set point failed, which was primarily due to the solenoid having a permanent magnet in the bottom, which

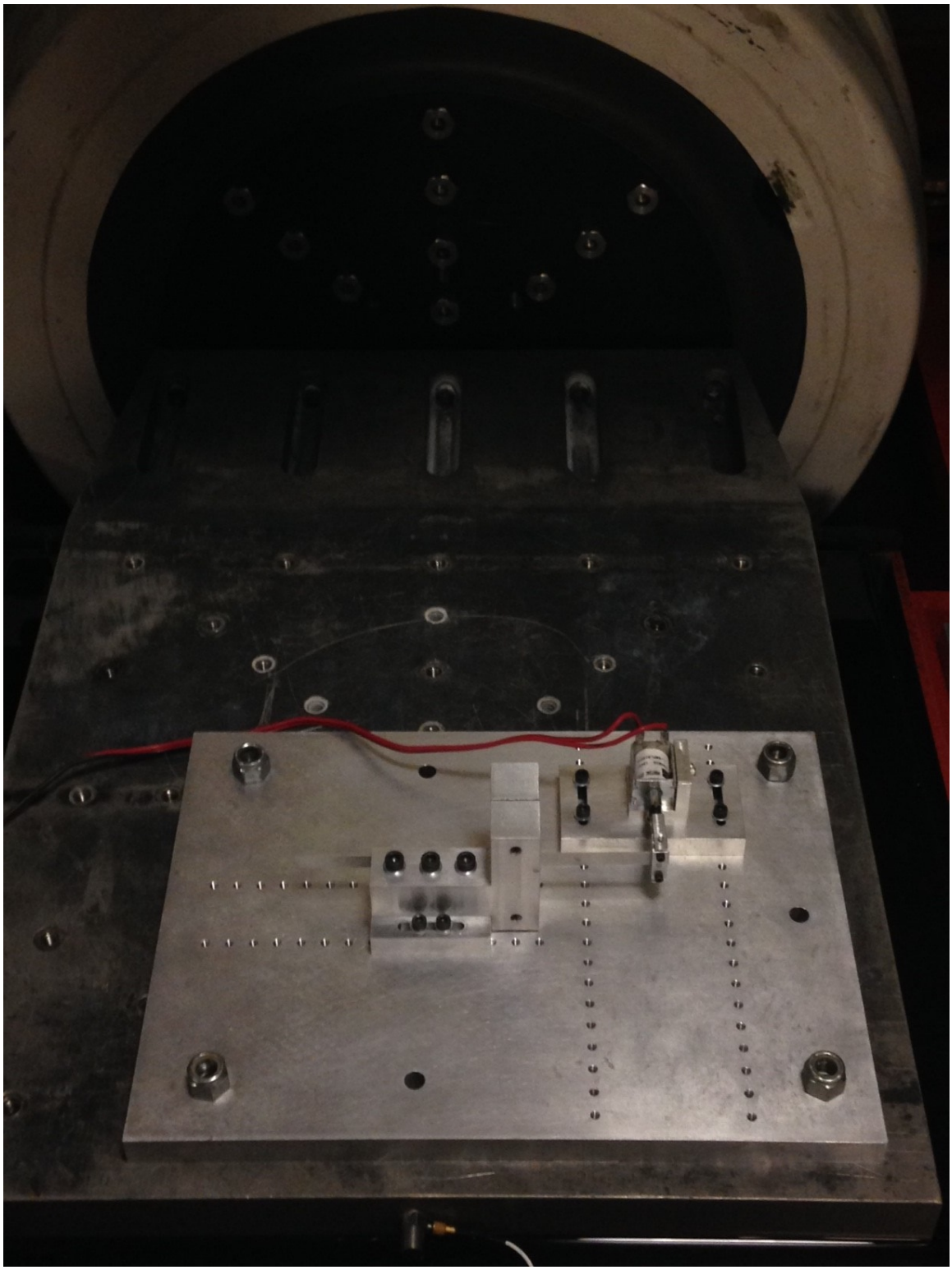


Figure 4.6: System mounted on the LDS-850 large shaker

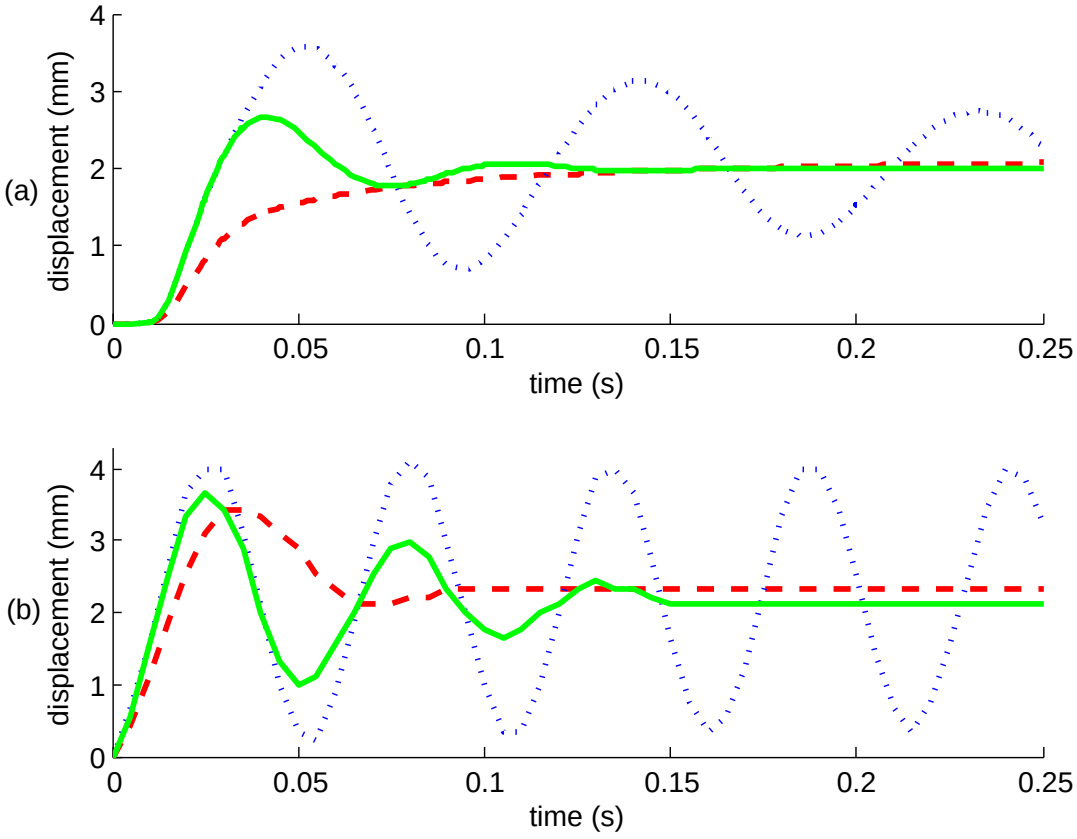


Figure 4.7: Comparison of step responses of open loop (dotted red), IOL (dashed blue) and IOL with ESO (solid green). (a) is the simulation part and (b) is the experimental part.

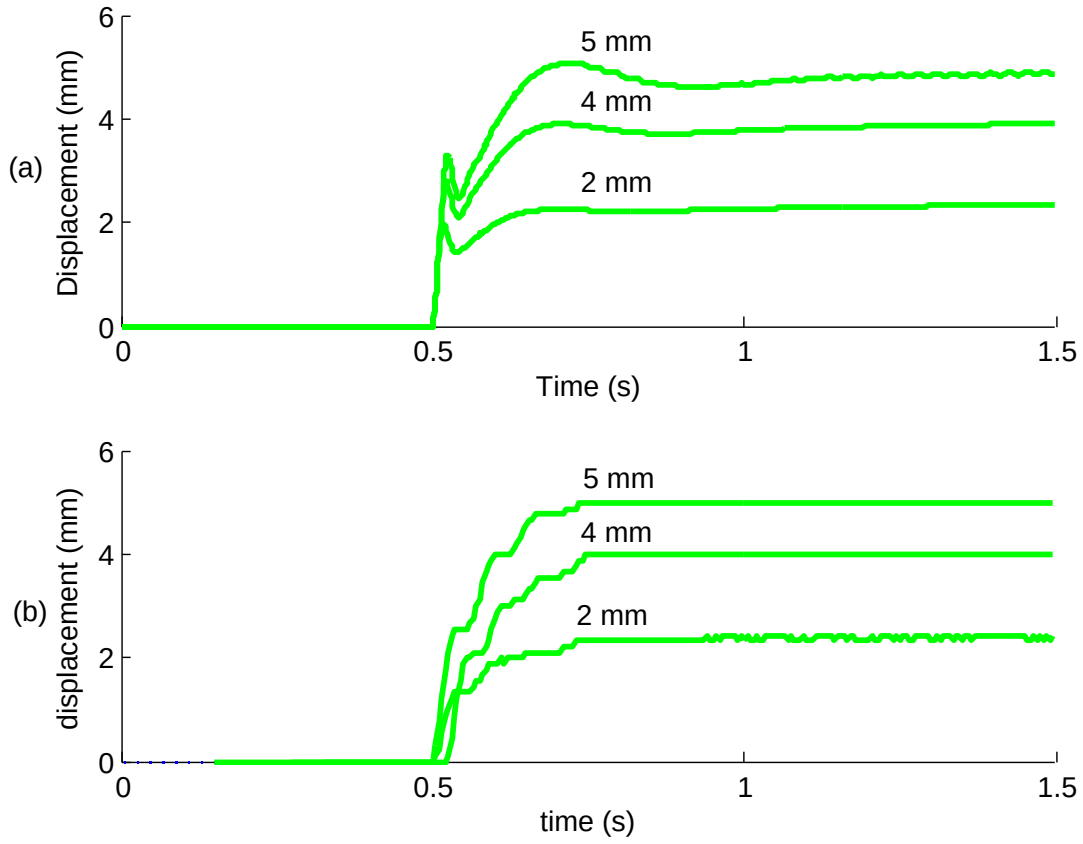


Figure 4.8: Comparison of PIESO controller's step responses with set points of 2mm, 4mm and 5mm. (a) is the simulation, (b) is the experiment.

is designed to provide more sealed force, which significantly changes the system's behavior from the developed model as displacement approaches this point. The results prove that PIESO can extend the solenoid's stable travel range and eliminate the displacement error.

4.6.3 PIESO With Disturbances

Finally, Fig. 4.9 to Fig. 4.11 illustrate the step responses of IOL and PIESO under disturbances. Based on the displacement sensor's capability and the system's natural frequency, disturbances were chosen as sinusoidal disturbances with an amplitude of 1 mm and frequencies of 10 Hz, 16 Hz and 20 Hz, which is below, at and above its natural frequency, respectively. The amplitude of the disturbances are all 1 mm. For clarity, the set points are 2.4 mm, 4 mm, 5 mm, respectively. When the disturbance frequency is 10 Hz, the output is

a sinusoidal signal that has the same frequency and amplitude as the input, in theory. The IOL response matches this point in simulation. Though the average final value is 2 mm, it ripples around it. The higher set point failed both in simulation and experiments. The system will snap in together or oscillate with a very large amplitude. The PIESO can extend its stable range to 5 mm (77.6% of total range) at this situation. However, it also oscillates around the desired position. At this frequency, the oscillation has an amplitude of about 0.1 mm. In practice, the external disturbance does not appear until the reference input command is given. It is caused by the complicated solenoid fixture. The static friction force holds the stroke when the external disturbances have insufficient energy. The IOL oscillates with a large amplitude (larger than 4 mm) and does not converge within 1 s. The PIESO can also obtain a stable range of 77.6% with a rise time around 0.2 s with little oscillation.

The responses with a disturbance of 16 Hz in simulation are similar to the results with 10 Hz, but the PIESO's performances are worse. The settling times are longer than 0.5 s and oscillation amplitudes are around 0.4 mm. The IOL response in the experimental test shows a higher excited oscillation and the PIESO's results show that it can reach the desired point without oscillation within 0.2 s.

The results with the disturbances even above the natural frequency demonstrate that the external disturbances can influence the system's performance. The simulation and experimental IOL responses demonstrate that the output amplitude is amplified at this frequency (3 mm in the simulation and 3.5 mm in the experiment). The displacements after a step input using IOL also vibrated with large amplitudes and will not converge. In simulation, the PIESO's responses also oscillated at all three set points, but the amplitudes are 70% less than the IOL. In contrast, the PIESO performs well at set points of 4 mm and 5 mm in practice, but oscillates at 2.4 mm. At the set point of 2.4 mm, the PIESO vibrates with an amplitude about 0.5 mm, while there are no oscillations at the set points of 4 mm and 5 mm. The settling times of PIESO are all less than 0.3 s in the experiments. The PIESO can cancel the disturbances, maintain the set position and extend the stable range. Typically, a

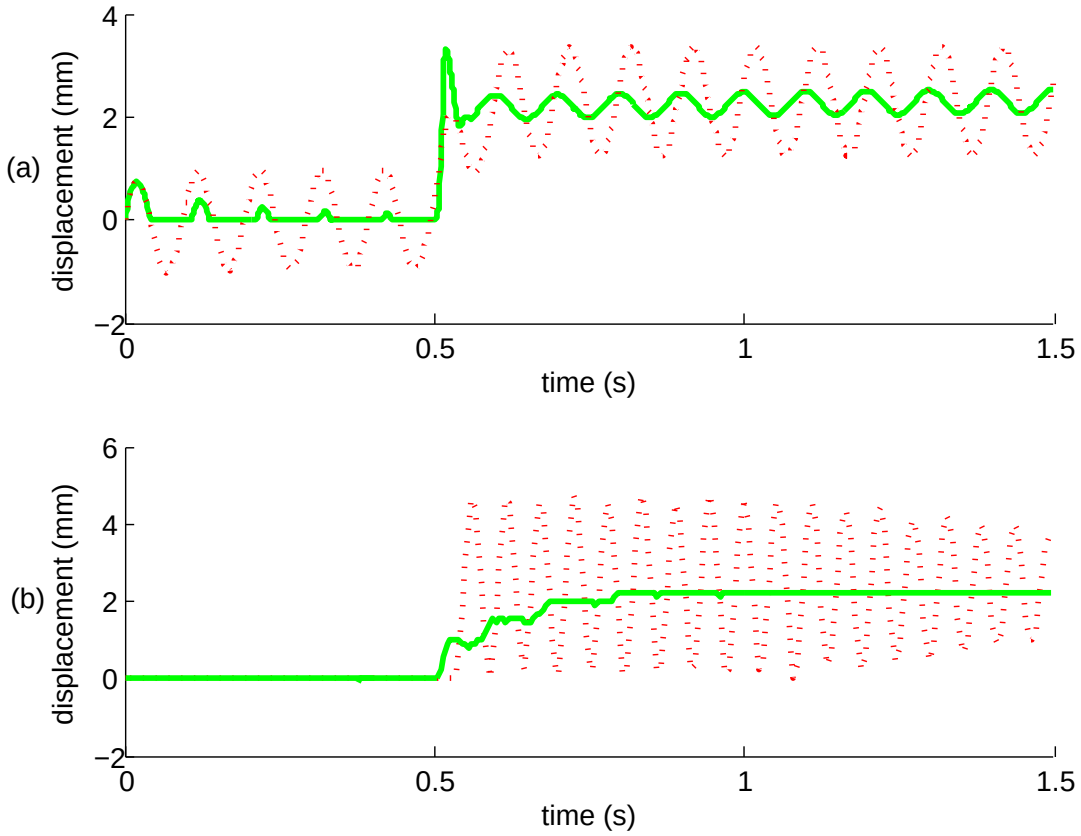


Figure 4.9: Comparison of IOL and PIESO controller's responses via a set point of 2.4mm with a 10Hz, 1mm amplitude sinusoidal disturbance: IOL (dotted red), PIESO (solid green). IOL with a set point higher than 2.4mm failed. (a) is the simulation, (b) is the experiment.

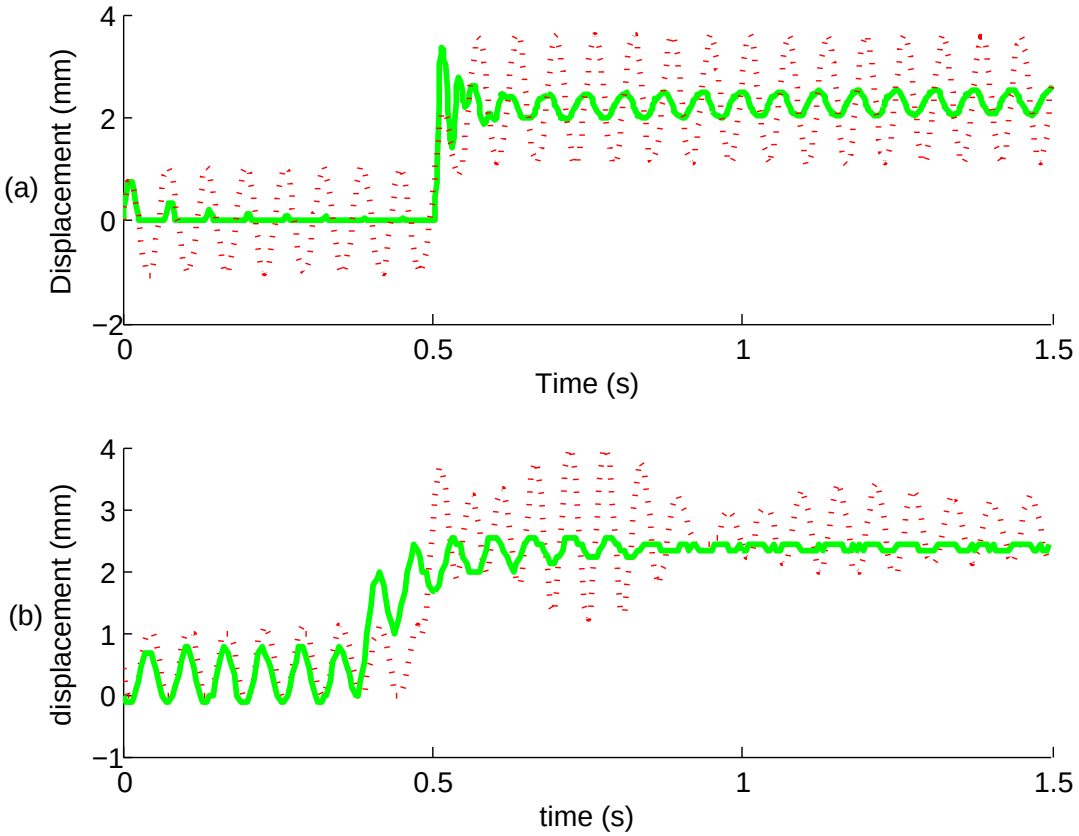


Figure 4.10: Comparison of IOL and PIESO controller's responses via a set point of 2.4mm with a 16Hz, 1mm amplitude sinusoidal disturbance: IOL (dotted red), PIESO (solid green). IOL with a set point higher than 2.4mm failed. (a) is the simulation, (b) is the experiment.

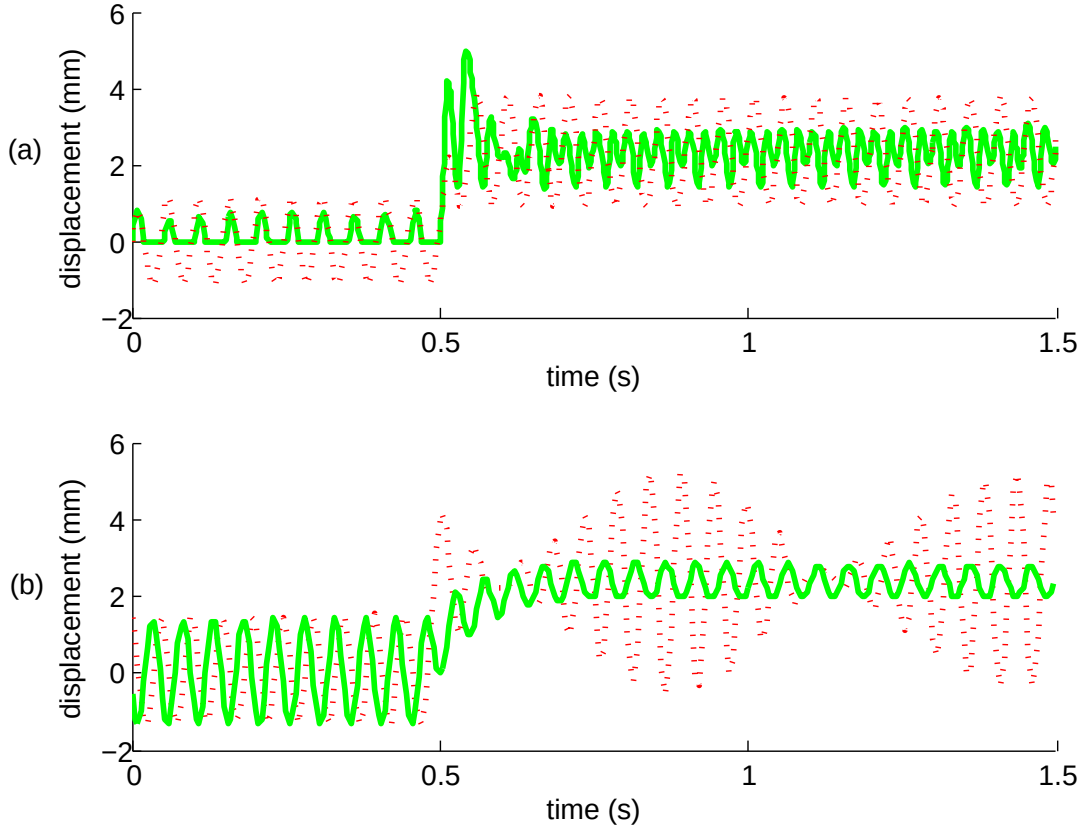


Figure 4.11: Comparison of IOL and PIESO controller's responses via a set point of 2.4mm with a 20Hz, 1mm amplitude sinusoidal disturbance: IOL (dotted red), PIESO (solid green). IOL with a set point higher than 2.4mm failed. (a) is the simulation, (b) is the experiment.

higher set point can cancel the disturbances better because the spring force is stronger, while the magnetic force can only be applied in one direction. The output response's amplitude at a frequency that is above its resonant frequency is higher than a amplitude which is below the resonant frequency, because the control method changes the damping coefficient and equivalent mass. The experimental results were better than the simulation results for two reasons: (1) the simulations did not consider the effect of the static and dynamic frictions, but instead used a constant damping coefficient, and (2) to avoid the unstable numerical solutions caused by an algebrical loop, there was a memory unit in the simulation feedback loop, which could introduce more time delay to the system.

4.7 Summary

Reviewing the simulation and experimental results, it is clear that the time delay has the effect of decreasing the damping. Even if the original mechanical system is not slightly damped, time delay caused by measurement and actuation can influence the input-output linearization technique's performance. In reality, the existence of friction may enhance the performance under the time delay problem, but it is not reliable. The PIESO can effectively extend the stable range and maintain the position without perfect knowledge of the system characteristics. This requires accurate feedback and adjusting the ESO's parameters depending on the designer's experiences.

Chapter 5

Self Sensing Analysis and Observability Improvement

5.1 Problem Statement

The operational mechanism of PPAs is the generation of an electrostatic force between a fixed electrode and a movable electrode when a voltage is applied across them. The detection of the actuators' mechanical motion is desired when they are combined with feedback control techniques, especially when the application requires high performance or is affected by disturbances. The required signals can be captured by a variety of sensing techniques, including capacitive, piezoresistive and optical.

Additionally, electrostatic actuators can be modeled as a type of variable capacitor, which depends on the gap between two electrodes. Thus, the displacement of the actuator can be obtained by measuring the capacitance [51]. This capacitive measurement approach has certain advantages: (1) its interface circuits can be packaged with the MEMS unit and (2) it is minimally sensitive to changes in temperature.

However, this practical method often requires high frequency excitation signal sources or additional sensing structures [73]. The excitation power source not only affects the performance in the actuator's steady state, but it may also generate harmonics that distort the measurement signals [19]. In addition, information about velocity may not be obtained without specific sensing structures. The additional structures occupy more space in each die, which could increase the cost and size, or decrease the performance.

In this chapter, a self-sensing technique is proposed to avoid these problems. It utilizes an estimator with a series resistor configuration. The estimator can estimate the displacement and velocity by measuring the voltage of the power supply and the voltage across the actuator itself. The remainder of this chapter presents the details.

5.2 Modeling and Observability Analysis

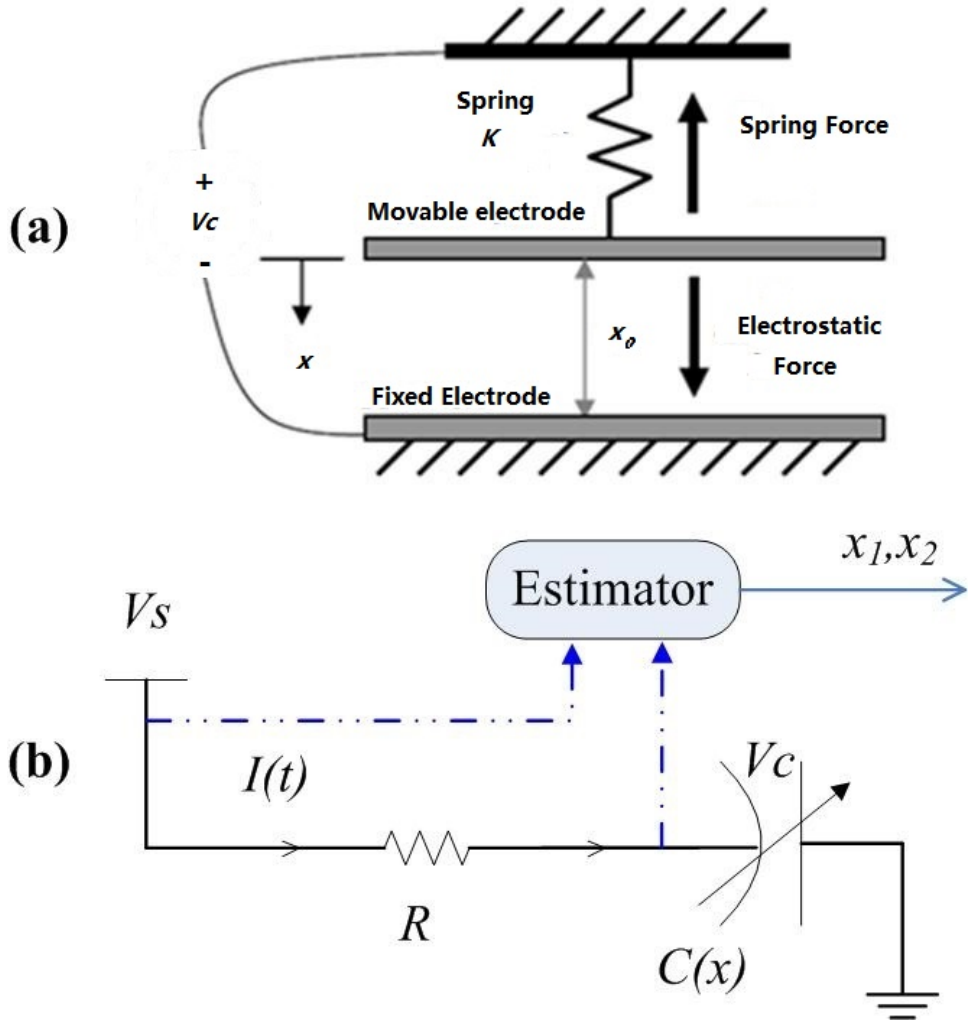


Figure 5.1: An illustration of a parallel plate actuator. (a) is the mechanical structure and (b) is the equivalent circuit model.

To verify the feasibility of the proposed method, it is necessary to prove that the displacement and velocity of the movable electrode can be derived by measuring the voltage, or in other words, the system must be observable under this configuration. The first step is discussing the mathematical model of PPAs. An illustrative drawing of a PPA is presented in Fig. 1, where (a) describes the mechanical part and (b) shows the equivalent circuit. The

overall system's dynamics can be described by:

$$\begin{aligned} m\dot{x_2} &= -cx_2 - kx_1 + \frac{\varepsilon_0 \varepsilon_r A V_c^2}{2(x_0 - x_1)^2} \\ \dot{V_c} &= \frac{1}{c(x_1)}(-\frac{V_c}{R} - \frac{V_c \varepsilon_0 \varepsilon_r A}{2(x_0 - x_1)^2} x_2 + \frac{V_s}{R}). \end{aligned} \tag{5.1}$$

The planar electrodes are arranged in a parallel configuration with an overlapping surface area, A, and a rest distance between them of x_0. The bottom electrode is fixed in space and the top electrode is allowed to move toward or away from the bottom electrode. x_1 is the displacement of the device; x_2 is the velocity; m is the mass of the movable electrode. The system's suspension system constrains the motion of the top electrode and is modeled by the system spring constant, k; c is the damping coefficient. ε is permittivity of the dielectric material between the two electrodes; R is the series resistor to prevent shorting the power supply if the two electrodes come into contact, which is a typical configuration for PPAs; V_s is the power source and V_c is the voltage across the PPA; $c(x)$ is the actuator's variable capacitance which could be modeled as:

$$c(x_1) = \frac{\varepsilon_0 \varepsilon_r D}{x_0 - x_1}. \tag{5.2}$$

Assuming that the voltage across the device can be measured, the observation function is:

$$y = Cx = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ V_c \end{bmatrix}, \tag{5.3}$$

where y is the measured signal and C is the observation matrix.

To explore the observability by measuring V_c, a linear system analysis is performed where the system is linearized around an equilibrium point $(x_{1e}, 0, V_{ce})$ and $V_{ce} = V_s$:

$$\delta\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ -k/m + \frac{\varepsilon_0 \varepsilon_r D V_{ce}^2}{m(x_0 - x_{1e})^3} & -c/m & \frac{\varepsilon_0 \varepsilon_r D V_{ce}}{m(x_0 - x_{1e})^2} \\ 0 & -\frac{1}{2(x_0 - x_{1e})} V_{ce} & -\frac{x_0 - x_{1e}}{R \varepsilon_0 \varepsilon_r D} \end{bmatrix} \delta x. \tag{5.4}$$

For the sake of clarity, (5.4) can be rewritten using the followed notation:

$$\delta\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ a_{21} & a_{22} & a_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix} \delta x = A\delta x, \tag{5.5}$$

where A is the system matrix of this linearized system.

The dynamics of the linear estimator can be described by:

$$\dot{\hat{x}} = A\hat{x} + L(y - C\hat{x}), \tag{5.6}$$

where $\hat{x}$ is the estimated state and L is the gain vector of the predicted errors. Using the observability criterion, the system is observable if and only if the observability matrix:

$$Q_o = \begin{bmatrix} C & CA & CA^2 \end{bmatrix}^T \tag{5.7}$$

has full rank. Based on (5.3), (5.4) and (5.5), Q_o can be calculated as:

$$Q_o = \begin{bmatrix} 0 & 0 & 1 \\ 0 & a_{32} & a_{33} \\ a_{21}a_{32} & a_{22}a_{32} + a_{32}a_{33} & a_{33}^2 + a_{23}a_{32} \end{bmatrix}. \tag{5.8}$$

To determine the rank of Q_o, it is convenient to compute its determinant, which equals:

$$|G| = a_{21}a_{32}^2. \tag{5.9}$$

The determinant indicates that a_{21} and a_{32} must be nonzero to ensure that the system is observable. Thus, if the system is supplied with a power source and the overall spring force is larger than the electrostatic force, its mechanical states x_1 and x_2 can be derived.

5.3 Device Design Optimization

Although the observability has been proved, it is important to further explore the device's characteristics in order to design a PPA with good observability. Thus the estimator is optimized if the measurement has balanced information about all three states. To address this, a singular value decomposition (SVD) is applied to study the singularity of (5.8):

$$Q_o = U\Sigma V^T, \tag{5.10}$$

where $\Sigma = diag(\sigma_1, \sigma_2, \sigma_3)$ is a diagonal matrix, $\sigma_1 \geq \sigma_2 \geq \sigma_3$ are the singular values of Q_o; U and V are orthogonal matrices. Based on principal component analysis theory, the singular values represent the magnitude of the states [50]. To weight the singularity of Q_o, using the O_2 observability index in order to compare the difference between the biggest and smallest amplitude:

$$O_2 = \frac{\sigma_1}{\sigma_3}. \tag{5.11}$$

In actual practice, it is useful to decrease the value of O_2. Since solving for the singular values using numerical algorithms directly is difficult, a simpler criterion is introduced based

on the properties of SVD. The uppper bound of O_2 is:

$$\frac{\|G\|_F^3}{|G|} \geq \frac{\sigma_1}{\sigma_3}, \tag{5.12}$$

where $\|G\|_F$ is the Frobenius norm of G:

$$\|G\|_F = (\sum_{i=1}^{3}\sum_{j=1}^{3}|a_{ij}|^2)^{1/2}. \tag{5.13}$$

Because the determinant of Q_o is already given in (5.9), it would be useful to find its linkage with the singular values:

$$|G| = \sigma_1\sigma_2\sigma_3. \tag{5.14}$$

Thus, if $|G| = a_{21}a_{32}^2$ is increasing and $\|G\|_F$ is decreasing, the observability O_2 will decrease, which means the actuator's observability is improved (less sensitive to noise and finite word length).

The observability can be improved if an element in G changes which satisfies:

$$\frac{\partial \|G\|_F^3}{\partial a_{ij}} \leq \frac{\partial |G|}{\partial a_{ij}}. \tag{5.15}$$

which means the changing of this parameter contributes more to the determinant compared to the cubic of its norm. According to (5.15), the selection of the parameters during the design process can be optimized. Though the impacts of some parameters on observability are not monotonic, a few immediate observations can be made: larger series resistance, smaller initial gap and smaller spring constant will improve the observability. A lower damping ratio may also increase the observability, but MEMS devices tend to be lightly damped, so there is usually little margin to decrease it.

5.4 Validation

Using MATLAB Simulink, simulation studies were performed to verify the concept. An estimator using voltage feedback was designed for a PPA. The parameters of the actuator were shown in Table 5.1. The step responses of the system dynamic model and estimator are

Table 5.1: Paramters of the device

Symbol	Description	value	unit
k	spring constant	100	N/m
c	damping coefficient	7.5 $\times 10^{-4}$	N $\cdot$ s/m
m	proof mass	7.5	μg
A	overlap area of the electrodes	2.9 $\times$ 2.9	mm
R	series resistance	300 M	Ω
x_0	initial gap distance	50	μm
ε_0	permittivity of free space	8.854×10^{-12}	F/m
ε_r	permittivity of air	1.0006	N/A

shown in Fig. 5.2. Both the estimated displacement and velocity achieve steady state within approximately 3 ms. The steady state errors are less than 1 μm and 0.1 mm/s, respectively. The results demonstrated the feasibility of this type of estimator.

Results regarding the relationship between parameters and the observability index are shown in Fig. 5.3. This data supports the arguments which were presented in the previous section. Increasing the spring constant can adversely affect the observability while a larger initial gap will enhance it.

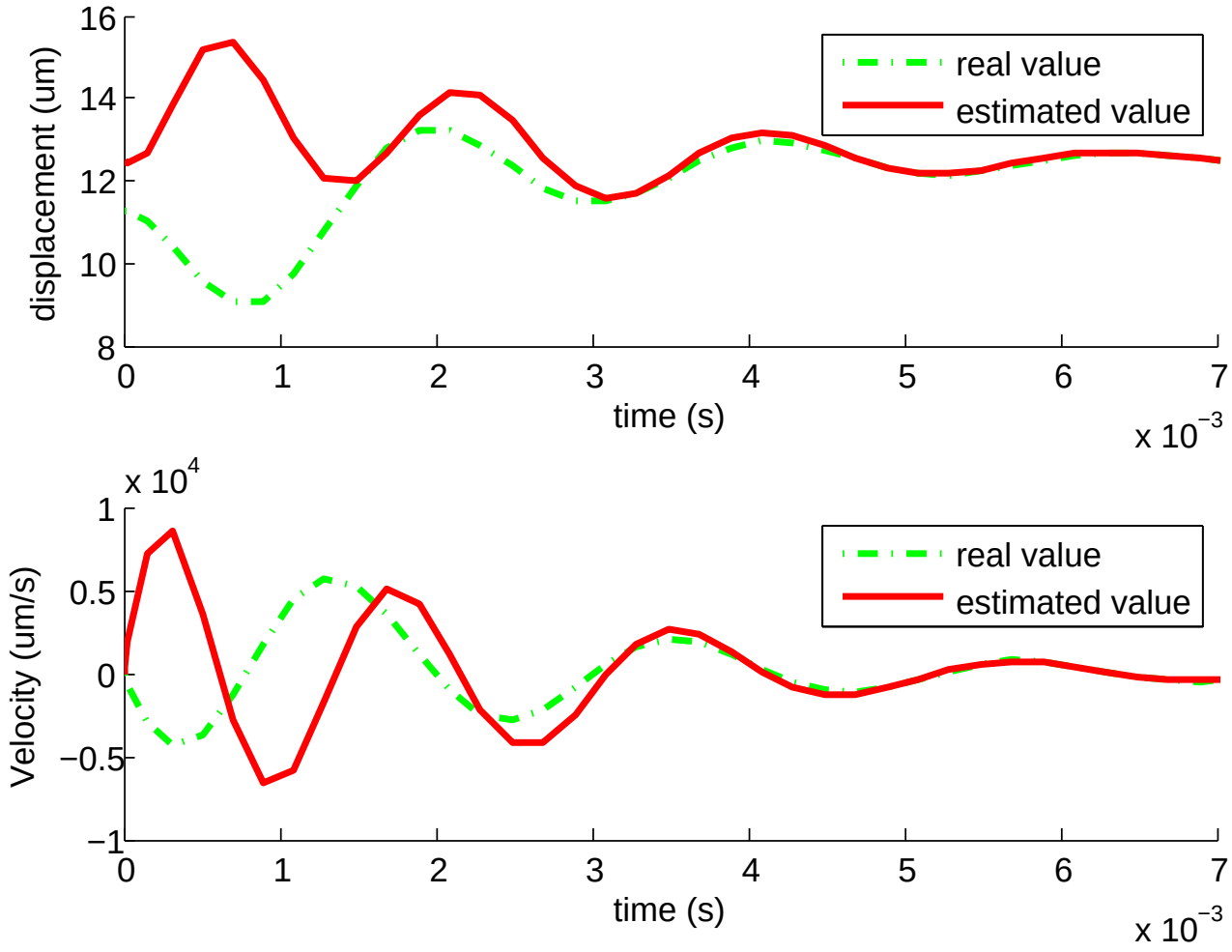


Figure 5.2: Simulation results: (a) is the displacement and (b) is the velocity.

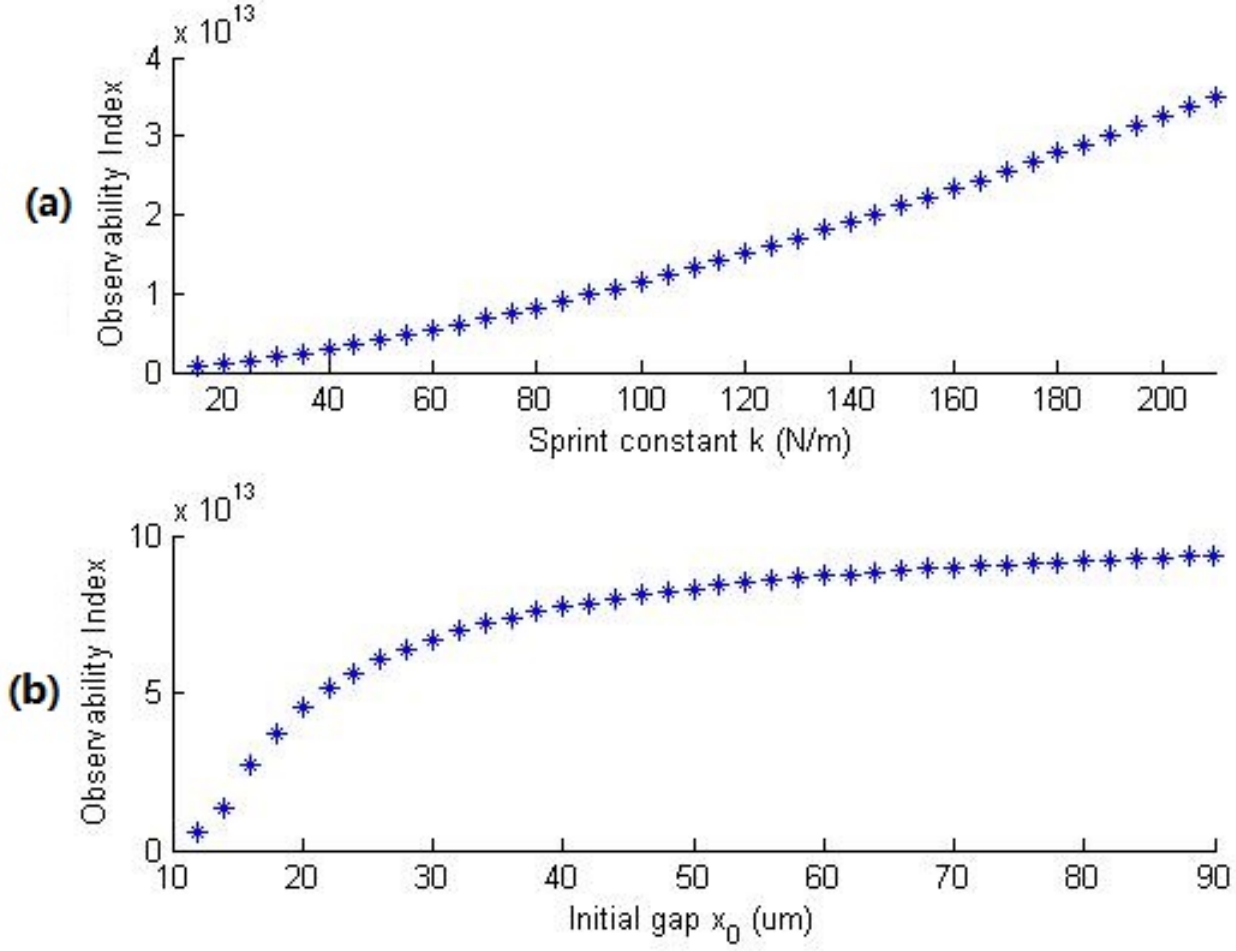


Figure 5.3: The observability indices with changing of (a) the spring contant and (b) initial gap.

5.5 An Exploration of Nonlinear Observability Analysis for PPAs

Though the observability analysis for the PPA around an equilibrium point was obtained, it was limited to a small range of states (around equilibrium points). Thus, nonlinear observability analysis which can cover more states is explored in this chapter. The nonlinear observability is based on differential geometry [31]. Consider a nonlinear system:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x). \end{aligned} \tag{5.16}$$

The lie derivative is define by [31]:

$$\mathcal{L}_f h = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i. \tag{5.17}$$

(5.17) can be rewritten as:

$$\mathcal{L}_f^1 h = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i \tag{5.18}$$

and the zero order of lie derivative is:

$$\mathcal{L}_f^0 h = h. \tag{5.19}$$

The second order lie derivative can also be obtained:

$$\mathcal{L}_f^2 h = \mathcal{L}_f(\mathcal{L}_f^1 h) \tag{5.20}$$

Then, the higher order lie derivative can be defined recursively:

$$\mathcal{L}_f^n h = \mathcal{L}_f(\mathcal{L}_f^{n-1} h) \tag{5.21}$$

To identify the local observability of a state of an n by n system, a new matrix $l(x)$ is defined by:

$$l(x) = \begin{bmatrix} \mathcal{L}_f^0 h \\ \mathcal{L}_f^1 h \\ \mathcal{L}_f^2 h \\ ... \\ \mathcal{L}_f^{n-1} h. \end{bmatrix} \tag{5.22}$$

The Jacobian around a state x_a of (5.22) is:

$$O_n = \left. \frac{\partial l(x)}{\partial x} \right|_{x=x_a}. \tag{5.23}$$

The system is locally observable if the matrix O_n has full rank. Recall the PPAs' dynamic equation:

$$\begin{aligned} \dot{x_1} &= x_2 \\ m\dot{x_2} &= -cx_2 - kx_1 + \frac{\varepsilon_0 \varepsilon_r A V_c^2}{2(x_0 - x_1)^2} \\ \dot{V_c} &= \frac{1}{c(x_1)}(-\frac{V_c}{R} - \frac{V_c \varepsilon_0 \varepsilon_r A}{2(x_0 - x_1)^2} x_2 + \frac{V_s}{R}). \end{aligned} \tag{5.24}$$

For simplification, the equation of V_c can be rewritten as:

$$\dot{V_c} = -\frac{V_c(x_0 - x_1)}{R\varepsilon_0 \varepsilon_r A} - \frac{V_c}{2(x_0 - x_1)} x_2 + \frac{V_s(x_0 - x_1)}{R\varepsilon_0 \varepsilon_r A}. \tag{5.25}$$

To explore the observability of the PPA, $l(x)$ should be obtained at first. According to the observation function:

$$\mathcal{L}_f^1 h = y = h = V_c. \tag{5.26}$$

The higher order terms can be calculated accordingly:

$$\begin{aligned}\mathcal{L}_f^1 h &= \dot{y} = \dot{h} = \mathcal{L}_f^1 h \\ &= -\frac{V_c(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A} - \frac{V_c}{2(x_0 - x_1)}x_2 + \frac{V_s(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A}\end{aligned} \tag{5.27}$$

$$\begin{aligned}&\mathcal{L}_f^2 h = \ddot{y} = \ddot{h} = \mathcal{L}_f^2 h \\ &= (\frac{V_c - V_s}{R\varepsilon_0\varepsilon_r A} - \frac{V_c}{2(x_0 - x_1)^2}x_2)x_2 \\ &- \frac{V_c}{2(x_0 - x_1)}(-\frac{c}{m}x_2 - \frac{k}{m}x_1 + \frac{\varepsilon_0\varepsilon_r A V_c^2}{2m(x_0 - x_1)^2}) \\ &+ (-\frac{(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A} - \frac{1}{2(x_0 - x_1)}x_2)(-\frac{V_c(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A} - \frac{V_c}{2(x_0 - x_1)}x_2 + \frac{V_s(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A}).\end{aligned} \tag{5.28}$$

Then, the Jacobian can be determined:

$$O_n = \left.\frac{\partial l(x)}{\partial x}\right|_{x=x_a} = \begin{bmatrix} 0 & 0 & 1 \\ O_{n21} & O_{n22} & O_{n23} \\ O_{n31} & O_{n32} & O_{n33} \end{bmatrix}. \tag{5.29}$$

The elements in (5.29) are:

$$O_{n21} = \frac{V_c - V_s}{R\varepsilon_0\varepsilon_r A} - \frac{V_c}{2(x_0 - x_1)^2}x_2 \tag{5.30}$$

$$O_{n22} = -\frac{V_c}{2(x_0 - x_1)} \tag{5.31}$$

$$O_{n23} = -\frac{(x_0 - x_1)}{R\varepsilon_0\varepsilon_r A} - \frac{1}{2(x_0 - x_1)}x_2 \tag{5.32}$$

$$\begin{aligned} O_{n31} = & \frac{-3A^3(\varepsilon_0\varepsilon_r)^3R^2V_c^3 - 8m(V_c - V_s)(x_0 - x_1)^5}{4A^2(\varepsilon_0\varepsilon_r)^2mR^2(x_0 - x_1)^4} \\ & + \frac{(2A^2(\varepsilon_0\varepsilon_r)^2R^2V_cx_0 - x_1)(kx_0(x_0 - x_1) - x_2(c(-x_0 + x_1) + mx_2)}{4A^2(\varepsilon_0\varepsilon_r)^2mR^2(x_0 - x_1)^4} \end{aligned} \tag{5.33}$$

$$\begin{aligned} O_{n32} = & \frac{m(4V_c - 3V_s)(x_0 - x_1)^2}{2A\varepsilon_0\varepsilon_r mR(x_0 - x_1)^2} \\ & + \frac{A\varepsilon_0\varepsilon_r RV_c(c(x_0 - x_1) - mx_2)}{2A\varepsilon_0\varepsilon_r mR(x_0 - x_1)^2} \end{aligned} \tag{5.34}$$

$$\begin{aligned} O_{n33} = & \frac{1}{4A^2(\varepsilon_0\varepsilon_r)^2mR^2(x_0 - x_1)^3}(-3A^3(\varepsilon_0\varepsilon_r)^3R^2V_c^2 \\ & + 4m(x_0 - x_1)^5 + 8A\varepsilon_0\varepsilon_r mR(x_0 - x_1)^3x_2 \\ & + A^2(\varepsilon_0\varepsilon_r)^2R^2(x_0 - x_1)(2k(x_0 - x_1)x_1 - x_2(-2cx_0 + 2cx_1 + mx_2))) \end{aligned} \tag{5.35}$$

The PPA is observable around a state x_a if (5.29) has full rank. The rank of (5.29) can be determined by calculating its determinant which is:

$$|O_n| = O_{21}O_{n32} - O_{n22}O_{n31}, \tag{5.36}$$

which could be determined numerically.

5.6 Summary

The self-sensing technique to estimate the mechanical states of a class of MEMS electrostatic actuators is feasible when a reasonably soft spring exists and the displacement is nonzero. The variation of the actuators' parameters will also affect the observability. This technique is optimized with a lower natural frequency and a smaller initial gap. The nonlinear observability analysis was also discussed.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this work, three key problems for PPAs were investigated by studying their dual systems (solenoids), which include modeling, disturbance control techniques and estimation using self sensing techniques. In the modeling part, the dynamics circuit model which contains a variable capacitor/inductor was analyzed, where variable capacitors/inductors are key components that exist in PPAs/solenoids. An approximated solution technique for variable inductor with a DC power source was derived and validated by simulation and experiments. An approximated solution for a variable capacitor with an AC power source was derived and validated by experiments.

A practical control technique for PPAs/solenoids contains input-output linearization and a disturbance rejection controller were designed. The difficulties of the original input-output linearization method were analyzed, which could be improved by using the cascade disturbance controller using the extended state observer. Simulation and experimental validation were explored for solenoids.

The feasibility of the self-sensing technique for PPAs was analyzed. Their observability using linear system theory around an equilibrium point was proved. The observability can be improved using a SVD analysis. The nonlinear observability was also discussed.

6.2 Future Works

PPAs are high bandwidth systems with low damping ratios, which is challenging to extend their stable operating range with acceptable costs. It would be worthwhile to implement

the proposed input-output linearization with disturbance rejection controller for PPAs. The feedback posi-cast controller could be another competitive approach to solve the PPAs' low damping ratio problem. Model based multi-rate digital controllers would also be interesting.

To further address the potential of self-sensing techniques for PPAs, a control technique combined with a self-sensing estimator is also desired.

Bibliography

[1] Lassi Aarniovuori, Lasse IE Laurila, Markku Niemela, and Juha J Pyrhonen. Measurements and simulations of dtc voltage source converter and induction motor losses. *Industrial Electronics, IEEE Transactions on*, 59(5):2277–2287, 2012.

[2] A. Adnan and D. Ishak. Finite element modeling and analysis of external rotor brushless dc motor for electric bicycle. In *Research and Development (SCOReD), 2009 IEEE Student Conference on*, pages 376–379, Nov 2009.

[3] Sohel Anwar. Predictive yaw stability control of a brake-by-wire equipped vehicle via eddy current braking. In *American Control Conference, 2007. ACC'07*, pages 2308–2313. IEEE, 2007.

[4] P Arpaia, C Petrone, and L Walckiers. Experimental validation of solenoid magnetic centre measurement by vibrating wire system. In *XX IMEKO World Congress Metrology for Green Growth*, pages 9–14, 2012.

[5] Jae-Nam Bae, Yong-Eun Kim, Young-Wook Son, Hee-Seok Moon, Chang-Hee Yoo, and Ju Lee. Self-excited induction generator as an auxiliary brake for heavy vehicles and its analog controller. *Industrial Electronics, IEEE Transactions on*, 62(5):3091–3100, May 2015.

[6] M Bakri-Kassem and RR Mansour. High tuning range parallel plate mems variable capacitors with arrays of supporting beams. In *Micro Electro Mechanical Systems, 2006. MEMS 2006 Istanbul. 19th IEEE International Conference on*, pages 666–669. IEEE, 2006.

[7] S. Bashmal, W.A. Oke, and Y. Khulief. Vibration analysis of an elastically restrained microcantilever beam under electrostatic loading using wavelet-based finite element method. *Micro Nano Letters, IET*, 10(3):147–152, 2015.

[8] SS Bedair, JS Pulskamp, CD Meyer, RG Polcawich, and IM Kierzewski. Modeling, fabrication and testing of mems tunable inductors varied with piezoelectric actuators. *Journal of Micromechanics and Microengineering*, 24(9):095017, 2014.

[9] Stephen Beeby. *MEMS mechanical sensors*. Artech House, 2004.

[10] Jonathan Bernstein, Raanan Miller, William Kelley, and Paul Ward. Low-noise mems vibration sensor for geophysical applications. *Microelectromechanical Systems, Journal of*, 8(4):433–438, 1999.

[11] Zhenning Cao, Fangyuan Chen, Ning Bao, Huacheng He, Peisheng Xu, Saikat Jana, Sunghwan Jung, Hongzhen Lian, and Chang Lu. Droplet sorting based on the number of encapsulated particles using a solenoid valve. *Lab on a chip*, 13(1):171–178, 2013.

[12] Mei-Yung Chen, Hsuan-Han Huang, and Shao-Kang Hung. A new design of a sub-micropositioner utilizing electromagnetic actuators and flexure mechanism. *Industrial Electronics, IEEE Transactions on*, 57(1):96–106, 2010.

[13] NC Cheung, KW Lim, and MF Rahman. Modelling a linear and limited travel solenoid. In *Industrial Electronics, Control, and Instrumentation, 1993. Proceedings of the IECON'93., International Conference on*, pages 1567–1572. IEEE, 1993.

[14] JC Chiou and YC Lin. A novel capacitance control design of tunable capacitor using multiple electrostatic driving electrodes. In *Nanotechnology, 2001. IEEE-NANO 2001. Proceedings of the 2001 1st IEEE Conference on*, pages 319–324. IEEE, 2001.

[15] Kang-Min Choi, Hyung-Jo Jung, Heon-Jae Lee, and Sang-Won Cho. Feasibility study of an mr damper-based smart passive control system employing an electromagnetic induction device. *Smart Materials and Structures*, 16(6):2323, 2007.

[16] Robert N. Dean Chong Li and John Y. Hung. Nonlinear feedback control to enhance stable performance of micromachined electrostatic parallel plate actuators. *International Journal of Automation and Power Engineering*, 3(1):49–52, 2014.

[17] Thomas Christen. A 15-bit 140-w scalable-bandwidth inverter-based modulator for a mems microphone with digital output. *Solid-State Circuits, IEEE Journal of*, 48(7):1605–1614, 2013.

[18] Liang Chu, Yanli Hou, Minghui Liu, Jun Li, Yimin Gao, and Mehrdad Ehsani. Study on the dynamic characteristics of pneumatic abs solenoid valve for commercial vehicle. In *Vehicle Power and Propulsion Conference, 2007. VPPC 2007. IEEE*, pages 641–644. IEEE, 2007.

[19] Robert N Dean, Abby Anderson, Stanley J Reeves, George T Flowers, et al. Electrical noise in mems capacitive elements resulting from environmental mechanical vibrations in harsh environments. *Industrial Electronics, IEEE Transactions on*, 58(7):2697–2705, 2011.

[20] Robert N Dean, Christopher Wilson, James P Brunsch, and John Y Hung. A synthetic voltage division controller to extend the stable operating range of parallel plate actuators. In *Control Applications (CCA), 2011 IEEE International Conference on*, pages 1068–1074. IEEE, 2011.

[21] Robert N Dean and Christopher G Wilson. Nonlinear circuit analysis for time-variant microelectromechanical system capacitor systems. *Micro & Nano Letters*, 8(9):515–518, 2013.

[22] Robert Neal Dean Jr, George T Flowers, Roland Horvath, Nicole Sanders, John Y Hung, Thaddeus Roppel, et al. Characterization and experimental verification of the nonlinear distortion in a technique for measuring the relative velocity between micromachined structures in normal translational motion. *Sensors Journal, IEEE*, 7(4):496–501, 2007.

[23] Dai Dong and Xiaoning Li. Simulation and experimental research on the response of a novel high-pressure pneumatic pilot-operated solenoid valve. In *Mechatronics and Machine Vision in Practice (M2VIP), 2012 19th International Conference*, pages 480–484. IEEE, 2012.

[24] Jingyan Dong and Placid M Ferreira. Electrostatically actuated cantilever with soi-mems parallel kinematic stage. *Microelectromechanical Systems, Journal of*, 18(3):641–651, 2009.

[25] Amro M Elshurafa, Kareem Khirallah, Hani H Tawfik, Ahmed Emira, Ahmed KS Aziz, and Sherif M Sedky. Nonlinear dynamics of spring softening and hardening in folded-mems comb drive resonators. *Microelectromechanical Systems, Journal of*, 20(4):943–958, 2011.

[26] Ran Fang, Wengao Lu, Guannan Wang, Tingting Tao, Yacong Zhang, Zhongjian Chen, and Dunshan Yu. A low-noise high-voltage interface circuit for capacitive mems gyroscope. *Journal of Circuits, Systems, and Computers*, 22(09):1340019, 2013.

[27] F. Gabriel. High-frequency effects in modeling ac permanent-magnet machines. *Industrial Electronics, IEEE Transactions on*, 62(1):62–69, Jan 2015.

[28] Javier García-Martín, Jaime Gómez-Gil, and Ernesto Vázquez-Sánchez. Non-destructive techniques based on eddy current testing. *Sensors*, 11(3):2525–2565, 2011.

[29] B. Ghodsian, P. Bogdanoff, and D. Hyman. Wideband dc-contact mems series switch. *Micro Nano Letters, IET*, 3(3):66–69, September 2008.

[30] Jingqing Han. From pid to active disturbance rejection control. *Industrial Electronics, IEEE transactions on*, 56(3):900–906, 2009.

[31] Robert Hermann and Arthur J Krener. Nonlinear controllability and observability. *IEEE Transactions on automatic control*, 22(5):728–740, 1977.

[32] Mehran Hosseini, Guchuan Zhu, and Yves-Alain Peter. A new formulation of fringing capacitance and its application to the control of parallel-plate electrostatic micro actuators. *Analog Integrated Circuits and Signal Processing*, 53(2-3):119–128, 2007.

[33] John Y Hung, Nathaniel G Albritton, and Fan Xia. Nonlinear control of a magnetic bearing system. *Mechatronics*, 13(6):621–637, 2003.

[34] Fang Jiancheng, Liu Xiquan, Bangcheng Han, and Kun Wang. Analysis of circulating current loss for high-speed permanent magnet motor. *Magnetics, IEEE Transactions on*, 51(1):1–13, Jan 2015.

[35] Che-Heung Kim. Mechanically coupled low-voltage electrostatic resistive rf multithrow switch. *Industrial Electronics, IEEE Transactions on*, 59(2):1114–1122, 2012.

[36] Masatoshi Kitamura, Yasutaka Kuzumoto, Shigeru Aomori, and Yasuhiko Arakawa. High-frequency organic complementary ring oscillator operating up to 200 khz. *Applied physics express*, 4(5):051601, 2011.

[37] Michael Kraft, Christopher Lewis, Thomas Hesketh, and Stefan Szymkowiak. A novel micromachined accelerometer capacitive interface. *Sensors and Actuators A: Physical*, 68(1):466–473, 1998.

[38] C. Lai, L.V. Iyer, K. Mukherjee, and N.C. Kar. Analysis of electromagnetic torque and effective winding inductance in a surface-mounted pmsm during integrated battery charging operation. *Magnetics, IEEE Transactions on*, PP(99):1–1, 2015.

[39] Giacomo Langfelder, Alessandro Caspani, and Alessandro Tocchio. Design criteria of low-power oscillators for consumer-grade mems resonant sensors. *Industrial Electronics, IEEE Transactions on*, 61(1):567–574, 2014.

[40] Jang-Sub Lee, Eun-Shil Yoo, Chi-Hyun Park, Jun-Eon An, Chan Gook Park, and Jin Woo Song. Development of a piezoresistive mems pressure sensor for a precision air data module. In *Control, Automation and Systems (ICCAS), 2014 14th International Conference on*, pages 874–878. IEEE, 2014.

[41] Jean Levine, Jacques Lottin, and Jean-Christophe Ponsart. A nonlinear approach to the control of magnetic bearings. *Control Systems Technology, IEEE Transactions on*, 4(5):524–544, 1996.

[42] Chong Li, Robert Neal Dean, and George T Flowers. Nonlinear circuit analysis technique for microelectromechanical systems with a time-variant capacitor and an ac power source. *Micro & Nano Letters, IET*, 10(11):637–640, 2015.

[43] Jiawei Li, Ying Liu, Min Tang, Jie Li, and Xinhua Lin. Capacitive humidity sensor with a coplanar electrode structure based on anodised porous alumina film. *Micro Nano Letters, IET*, 7(11):1097–1100, November 2012.

[44] Ying Li. Monotone iterative method for numerical solution of nonlinear odes in mosfet rf circuit simulation. *Mathematical and Computer Modelling*, 51(3):320–328, 2010.

[45] KW Lim, NC Cheung, and MF Rahman. Proportional control of a solenoid actuator. In *Industrial Electronics, Control and Instrumentation, 1994. IECON'94., 20th International Conference on*, volume 3, pages 2045–2050. IEEE, 1994.

[46] Yasser Mafinejad, Abbas Kouzani, Khalil Mafinezhad, and Iran Mashad. Review of low actuation voltage rf mems electrostatic switches based on metallic and carbon alloys. *Journal of Microelectronics, Electronic Components and Materials*, 43(2):85–96, 2013.

[47] Francis B Malit and MC Ramos. Characterization of ral bipedal robot capacitive mems accelerometer using electrical impedance measurements. In *TENCON 2012-2012 IEEE Region 10 Conference*, pages 1–5. IEEE, 2012.

[48] Robert Miklosovic, Aaron Radke, and Zhiqiang Gao. Discrete implementation and generalization of the extended state observer. In *American Control Conference, 2006*, pages 6–pp. IEEE, 2006.

[49] T.J.E. Miller, M.I. McGilp, D.A. Staton, and J.J. Bremner. Calculation of inductance in permanent-magnet dc motors. *Electric Power Applications, IEE Proceedings* -, 146(2):129–137, Mar 1999.

[50] Bruce C Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *Automatic Control, IEEE Transactions on*, 26(1):17–32, 1981.

[51] Steven Ian Moore and SOR Moheimani. Displacement measurement with a self-sensing mems electrostatic drive. *Microelectromechanical Systems, Journal of*, 23(3):511–513, 2014.

[52] Shigeo Morimoto, Yi Tong, Yoji Takeda, and Takao Hirasa. Loss minimization control of permanent magnet synchronous motor drives. *Industrial Electronics, IEEE Transactions on*, 41(5):511–517, 1994.

[53] Biswanath Mukherjee, KBM Swamy, TV Krishnan, and Satyaki Sen. A simple low cost scheme for closed loop operation of mems capacitive accelerometer. In *Students' Technology Symposium (TechSym), 2014 IEEE*, pages 111–115. IEEE, 2014.

[54] Klaus Mutschler, Shivam Dwivedi, Sabrina Kartmann, Stefan Bammesberger, Peter Koltay, Roland Zengerle, and Laurent Tanguy. Multi physics network simulation of a solenoid dispensing valve. *Mechatronics*, 24(3):209–221, 2014.

[55] M Narducci, L Yu-Chia, W Fang, and J Tsai. Cmos mems capacitive absolute pressure sensor. *Journal of Micromechanics and Microengineering*, 23(5):055007, 2013.

[56] C NatNataraj and DongBin Lee. Model-based adaptive control for a solenoid-valve system. In *ASME 2010 International Mechanical Engineering Congress and Exposition*, pages 1009–1015. American Society of Mechanical Engineers, 2010.

[57] Tamara G Nesterenko, Aleksei N Koleda, Evgenii S Barbin, and Sergey V Uchaikin. Temperature error compensation in two-component microelectromechanical gyroscope. *Components, Packaging and Manufacturing Technology, IEEE Transactions on*, 4(10):1598–1605, 2014.

[58] T Nguyen, J Leavitt, F Jabbari, and JE Bobrow. Accurate sliding-mode control of pneumatic systems using low-cost solenoid valves. *Mechatronics, IEEE/ASME Transactions on*, 12(2):216–219, 2007.

[59] Anna Persano, Fabio Quaranta, Maria Concetta Martucci, Pietro Siciliano, and Adriano Cola. On the electrostatic actuation of capacitive rf mems switches on gaas substrate. *Sensors and Actuators A: Physical*, 232:202–207, 2015.

[60] M Prudenziati. Handbook of sensors and actuators. *Thick Film Sensors*, 1, 1994.

[61] Muthapillai Rajavelu, Dhakshnamoorthy Sivakumar, Joseph Daniel Rathnam, and Koilmani Sumangala. Enhanced sensitivity with extended linearity in mems piezoresistive pressure sensor. *Micro & Nano Letters, IET*, 8(10):753–756, 2013.

[62] Falk Roewer, Ulrich Kleine, K-E Salzwedel, Felix Mednikov, Ch Pfaffinger, and Martin Sellen. A programmable inductive position sensor interface circuit. *INTEGRATION, the VLSI journal*, 38(2):227–243, 2004.

[63] Alexandre Schammass, Raoul Herzog, Philipp Bühler, and Hannes Bleuler. New results for self-sensing active magnetic bearings using modulation approach. *Control Systems Technology, IEEE Transactions on*, 13(4):509–516, 2005.

[64] Joseph Seeger, Bernhard E Boser, et al. Charge control of parallel-plate, electrostatic actuators and the tip-in instability. *Microelectromechanical Systems, Journal of*, 12(5):656–671, 2003.

[65] Joseph I Seeger, Selden B Crary, et al. Stabilization of electrostatically actuated mechanical devices. In *Proceedings of the 1997 International Conference on Solid-State Sensors and Actuators*, pages 1133–1136, 1997.

[66] Yingfeng Shan and Kam K Leang. Design and control for high-speed nanopositioning: serial-kinematic nanopositioners and repetitive control for nanofabrication. *Control Systems, IEEE*, 33(6):86–105, 2013.

[67] Wang Shishan, Liu Zeyuan, and Deng Zhiquan. Solution of inductance for bearingless switched reluctance motor by using enhanced incremental energy method. In *Electric Utility Deregulation and Restructuring and Power Technologies, 2008. DRPT 2008. Third International Conference on*, pages 2850–2855, April 2008.

[68] Ananth Sundaram, Madhurima Maddela, Ramesh Ramadoss, and Lucas M Feldner. Mems-based electronically steerable antenna array fabricated using pcb technology. *Microelectromechanical Systems, Journal of*, 17(2):356–362, 2008.

[69] Michael Suster, Jun Guo, Nattapon Chaimanonart, Wen H Ko, and Darrin J Young. A high-performance mems capacitive strain sensing system. *Microelectromechanical Systems, Journal of*, 15(5):1069–1077, 2006.

[70] M Taghizadeh, A Ghaffari, and F Najafi. Modeling and identification of a solenoid valve for pwm control applications. *Comptes Rendus Mecanique*, 337(3):131–140, 2009.

[71] Shahrzad Towfighian, A Seleim, EM Abdel-Rahman, and GR Heppler. A large-stroke electrostatic micro-actuator. *Journal of Micromechanics and Microengineering*, 21(7):075023, 2011.

[72] ND Vaughan and JB Gamble. The modeling and simulation of a proportional solenoid valve. *TRANSACTIONS-AMERICAN SOCIETY OF MECHANICAL ENGINEERS JOURNAL OF DYNAMIC SYSTEMS MEASUREMENT AND CONTROL*, 118:120–125, 1996.

[73] L. Welter, P. Dreux, H. Aziza, and J.-M. Portal. Embedded high-precision capacitor measurement system based on ring-oscillator. *Electronics Letters*, 51(6):521–523, 2015.

[74] Shang-Teh Wu, Szu-Chieh Mo, and Bo-Siou Wu. An lvdt-based self-actuating displacement transducer. *Sensors and actuators A: Physical*, 141(2):558–564, 2008.

[75] Chen Xing, Li Donghai, Gao Zhiqiang, and Wang Chuanfeng. Tuning method for second-order active disturbance rejection control. In *Control Conference (CCC), 2011 30th Chinese*, pages 6322–6327. IEEE, 2011.

[76] Yi Xiong, Jianhua Wei, and Ruilin Feng. Adaptive robust control of a high-response dual proportional solenoid valve with flow force compensation. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, page 0959651814549647, 2014.

[77] Wuliang Yin and AJ Peyton. Thickness measurement of non-magnetic plates using multi-frequency eddy current sensors. *NDT & E International*, 40(1):43–48, 2007.

[78] Lan Yu and Timothy N Chang. Zero vibration on–off position control of dual solenoid actuator. *Industrial Electronics, IEEE Transactions on*, 57(7):2519–2526, 2010.

[79] Quan Yuan, Wei Luo, Hui Zhao, Bohua Peng, Jinling Yang, and Fuhua Yang. Frequency stability of rf-mems disk resonators. *Electron Devices, IEEE Transactions on*, 62(5):1603–1608, May 2015.

[80] Jianglong Zhang, Zhongxia Zhang, YC Lee, Victor M Bright, and John Neff. Design and investigation of multi-level digitally positioned micromirror for open-loop controlled applications. *Sensors and Actuators A: Physical*, 103(1):271–283, 2003.

Appendices

Appendix A

Publications

1. Chong Li, Haoyue Yang, Luke. L. Jenkins, R.N. Dean, G.T. Flowers, John. Y. Hung, Enhanced Performance Control of an Electromagnetic Solenoid System Using a Digital Controller, IEEE Trans. On Control Systems Technology, in press.

2. Chong Li, R.N. Dean, G.T. Flowers Analysis and Dynamic Simulation of the Synthetic Voltage Division Controller for Extending the Parallel Plate Actuator Stable Range of Motion, Microsystem Technologies, Accepted.

3. Chong Li, R.N. Dean, G.T. Flowers A nonlinear circuit analysis technique for microelectromechanical systems with a time-variant capacitor and an AC power source, IET Micro and Nano Letters, Volume 10, Issue11, 2015.

4. Chong Li, C. Lavinia Elana Robert N. Dean, George T. Flowers, A MEMS PPA Based Active Vibration Isolator, 2016 IMAPS Device Packaging Conference, accepted.

5. Chong Li, Robert N. Dean, George T. Flowers, Nonlinear Observability Analysis of Micro-machined Electrostatic Actuators Using Self-Sensing, 2016 IMAPS Device Packaging Conference, accepted.

6. Chong Li, Yixuan Wu, Haoyue Yang, L. Jenkins, R.N. Dean, G.T. Flowers A Microcontroller Approach to Measuring the Transmissibility of MEMS Devices, 2015 IMAPS Device Packaging Conference,Scottsdale/Fountain Hills, AZ, March 11-14, 2015.

7. Chong Li, Robert N. Dean, George T. Flowers, John Y. Hung, Nonlinear Feedback Control to Enhance Stable Performance of Micromachined Electrostatic Parallel Plate Actuators, International Journal of Automation and Power Engineering, Volume 3, Issue1, 2014.

Nonlinear circuit analysis technique for microelectromechanical systems with a time-variant capacitor and an AC power source

Chong Li, Robert Neal Dean, George T. Flowers

Department of Mechanical Engineering, Auburn University, Auburn, AL 36830, USA
E-mail: czl0047@auburn.edu

Published in Micro & Nano Letters; Received on 8th July 2015; Revised on 1st September 2015; Accepted on 2nd September 2015

Microelectromechanical systems (MEMS) utilise time-variant capacitors as transducers in many applications. However, this kind of component can introduce harmonics and disturbances into the circuit with an AC power source, which is difficult to evaluate through closed-form solutions. This Letter proposes an iterative solution to analyse the dynamics of MEMS devices which contain a time-variant capacitor and an AC source. First, the expressions of the time-variant capacitor, AC source and their derivatives with respect to time are determined. Then, an initial solution that is sufficiently close to the actual solution is determined using linear circuit analysis. On the basis of the previous steps and the principles of the iterative method, an approximated solution combining the initial solution and its iteratively-derived higher-order terms is reached. Adding additional higher-order terms can improve the accuracy of the solution. A case study considering a MEMS device which has an AC power source and sinusoidal motion was performed using MATLAB Simulink. The simulation study further demonstrated that: (i) this iterative solution can effectively analyse the dynamics of MEMS devices with a time-variant capacitor and an AC power source; and (ii) computing additional higher-order terms derived from the initial solution can further improve the solution's accuracy.

1. Introduction: Time-variant capacitors exist in many microelectromechanical systems (MEMS) devices. They could be designed as sensing structures where the variation in capacitance represents different external physical parameters; examples include humidity sensors [1], vibration sensors [2], strain sensors [3], pressure sensors [4], gyroscopes [5] and accelerometers [6]. On the other hand, electrostatic actuators also include time-variant capacitors, which are the key components in many applications such as resonators [7, 8], micro mirrors [8], series switches [9], compliant structures [10] and RF devices [11, 12].

Thus, it is important to accurately analyse systems possessing time-variant capacitors. For a time-variant capacitor, $C(t)$, the current, $I_c(t)$, through it is

$$I_c(t) = \dot{V}(t)C(t) + V(t)\dot{C}(t). \tag{1}$$

Since (1) results in a nonlinear circuit model, it is difficult to obtain a closed-form solution. In some circumstances, this problem can be simplified by considering the variable capacitance as a constant, especially if it is powered by a DC voltage source or the AC source's frequency is much higher than the MEMS device's bandwidth. Then, linear circuit analysis can be applied to obtain a reasonably accurate solution. However, the time variation cannot be ignored in many applications. For example, consider an MEMS resonator vibrating at 2 kHz resonant frequency. In order to accomplish the feedback control, a 100 kHz voltage signal is applied to the device to detect the capacitance to measure the proof mass motion, which is a common configuration [13, 14]. Typically, the capacitance is treated as time invariant in this situation. Since the detection signal and the mechanical vibration should not affect each other, because the detection signal's frequency is much greater than the device's bandwidth, it is commonly considered that no additional components from the mechanical motion will be introduced [15]. In reality, however, the time-variant capacitor caused by the mechanical motion will introduce more harmonics into the detection signal, with frequency spacing equal to the frequency of the fundamental component. The spectrum of this example is given in Fig. 1 from a MATLAB SIMULINK simulation of the system. This phenomenon, though, has been experimentally documented in MEMS devices [16, 17]. In this figure, it is clear that harmonics with 2 kHz intervals are introduced, where each interval is equal to the vibration frequency. These harmonics can corrupt the measurement readings and introduce additional noise into the electrical system. Linear circuit analysis does not account for this very real and observable effect.

Note that the result contains both harmonics and intermodulation products of the AC source and the resonant frequency. Linear circuit analysis considers neither of these components. Dean and Wilson [18] proposed a nonlinear circuit analysis method for a time-variant MEMS capacitor system driven with a DC source, but the case with an AC voltage source was not investigated. To solve this problem, a standard circuit model needs to be considered.

Consider a time-variant capacitor with a series resistor, R, which is used to protect the capacitive element and prevent the power source from shorting to ground in case the MEMS device's electrodes physically contact each other. The circuit's schematic is given in Fig. 2. In addition, different configurations can be transformed into this model using the Thevenin equivalent circuit method.

The circuit's behaviour is described by

$$I_c(t) = (V_s(t) - V_c(t))/R = \dot{V}_c C(t) + V_c\dot{C}(t), \tag{2}$$

where V_s is the AC power source, V_c is the voltage across the variable capacitor and I_c is the current through it. Although (2) fully characterises the circuit's behaviour, it is difficult to solve this nonlinear differential equation in practice and obtain a closed-form solution.

2. Analysis approach: Since (2) is difficult to solve, an alternative iterative approach can be applied to obtain an approximate solution [18]. The first step is deriving an initial approximate solution $V_{c0}(t)$, which is sufficiently close to the $V_c(t)$ [19]. To achieve this initial solution, the variable capacitor in an MEMS device can be modelled as

$$C(t) = C_0 + C_1(t), \tag{3}$$

where C_0 is time invariant, and $C_1(t)$ is the time-variant part that

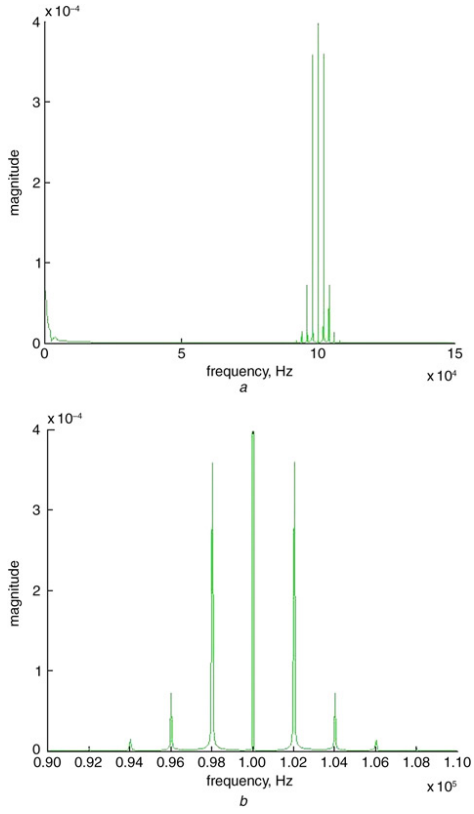


Fig. 1 *Spectral analysis of a MEMS resonator with detection signals is the*
a Overall spectrum
b Spectrum around the detection signal's fundamental frequency

must be less than C_0 to ensure that $C(t)$ is always positive. The derivative of (3) is

$$\dot{C}(t) = \dot{C}_1(t). \tag{4}$$

Ignoring the time-variant part and considering this circuit as a linear circuit, linear circuit analysis can be applied to obtain the initial solution $V_{c0}(t)$. The approximated circuit's transfer function is

$$\frac{V_c(S)}{V_s(S)} = \frac{1}{RC_0 s + 1}. \tag{5}$$

Then the first step is the analysis of this circuit ignoring the effect of $C_1(t)$ and considering it as a linear circuit. The purpose of this step is to obtain the steady state of $V_c(t)$, denoted by $V_{c0}(t)$. Then, $V_{c1}(t)$ can be calculated by (2) and (3). Correspondingly, the $I_{c1}(t)$ term is

$$I_{c1}(t) = \dot{V}_{c0}C(t) + V_{c0}\dot{C}(t). \tag{6}$$

Thus, using a small signal analysis to obtain V_{c1}

$$V_{c1}(t) = -I_{c1}(t)R. \tag{7}$$

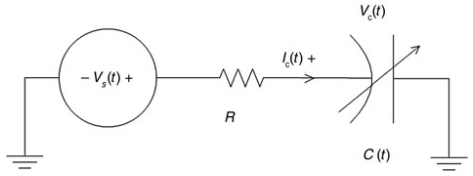


Fig. 2 *Thevenin equivalent schematic diagram of a time-variant capacitor with a series resistor and an AC voltage source*

Additional terms can be calculated recursively

$$I_{ck+1}(t) = \dot{V}_{ck}C(t) + V_{ck}\dot{C}(t). \tag{8}$$

The $I_{lk}(t)$ is then solved using as many terms as required to obtain sufficient accuracy. The overall equation for $V_c(t)$ is

$$V_c(t) = V_{c0}(t) + \sum_{k=0}^{\infty} (\dot{V}_{ck}(t)C(t) + V_{ck}(t)\dot{C}(t))R^{k+1}. \tag{9}$$

3. Case study

3.1. System modelling and analysis: To verify the proposed technique, the following case study is considered. Consider an MEMS resonator oscillating at ω Hz and a ϕ Hz frequency signal is used to detect its capacitance. The capacitance's expression is

$$C(x) = \frac{\varepsilon_0 \varepsilon_r A}{x_0 - x}, \tag{10}$$

where x is the displacement of the movable electrode, ε_0 is the permittivity of free space, ε_r is the relative permittivity of the dielectric material between the two electrodes, A is the overlapping surface area of the electrodes and x_o is the initial gap between the two electrodes.

Assuming that the MEMS device is stimulated by an external mechanical sinusoidal displacement input

$$x = y_1 \sin(\omega t), \tag{11}$$

where y_1 is a small magnitude and ω is the stimulating frequency. The capacitance $C(t)$ is

$$C(t) = \frac{\varepsilon_0 \varepsilon_r A}{x_0 - y_1 \sin(\omega t)}, \tag{12}$$

If $x \ll x_0$, the capacitance's expression can be approximated as [17]

$$C(t) = y_0 + y_1 \sin(\omega t). \tag{13}$$

The derivative of $C(t)$ is

$$\dot{C}(t) = y_1 \omega \cos(\omega t), \tag{14}$$

where y_0 is the time-invariant part when $x = 0$

$$y_0 = \frac{\varepsilon_0 \varepsilon_r A}{x_0}. \tag{15}$$

Let the AC power source, V_s, be equal to

$$V_s(t) = A_s \sin(\phi t). \tag{16}$$

doi: 10.1049/mnl.2015.0276

The derivative of $V_s(t)$ with respect to time is

$$\dot{V}_s(t) = A_s \phi \cos(\phi t). \tag{17}$$

To obtain the initial solution, V_{c0}, the linear circuit's steady-state solution is used

$$V_{c0}(t) = A_s \sin(\omega t). \tag{18}$$

On the basis of (6) and (14), I_{c1} can be obtained

$$\begin{aligned} I_{c1}(t) = {} & \phi A_s \cos(\phi t)(y_0 + y_1 \sin(\omega t)) \\ & + A_s \sin(\phi t) y_1 \omega \cos(\omega t). \end{aligned} \tag{19}$$

Then, V_{c1} can be found using (7)

$$\begin{aligned} V_{c1}(t) &= -I_{c1}(t)R \\ &= -R(\phi A_s \cos(\phi t) y_1 \sin(\omega t) \\ &\quad + A_s \sin(\phi t) y_1 \omega \cos(\omega t)). \end{aligned} \tag{20}$$

Similiarly, V_{c2} can be calculated recursively

$$\begin{aligned} V_{c2}(t) = {} & R^2 A_s(-y_0 \phi^2 \sin(\phi t) - y_1 \phi^2 \sin(\phi t) \sin(\omega t) \\ & + y_1 \phi \omega \cos(\phi t) \cos(\omega t) + y_1 \phi \omega \cos(\phi t) \cos(\omega t) \\ & - y_1 \omega^2 \sin(\phi t) \sin(\omega t))(y_0 + y_1 \sin(\omega t)) \\ & - R^2 A_s(\phi \cos(\phi t) y_1 \sin(\omega t) \\ & + \sin(\phi t) y_1 \omega \cos(\omega t)) \omega y_1 \cos(\omega t) \end{aligned} \tag{21}$$

3.2. Simulation study: To verify the feasibility of the proposed technique, a MATLAB SIMULINK model was built and analysed. The values of the series resistance and the capacitance are typically <1 MΩ and 20 pF, respectively [20, 21]. The excitation signal applied to the MEMS devices tends to be about ten times greater than the device's resonant frequency, where typical values could be up to 100–200 kHz. Thus, the time invariant y_0 was 20 pF, while y_1 was 5 pF, the series resistor, R, was 10 kΩ, the mechanical vibration frequency was 2 kHz, the AC voltage source's amplitude was 1 V and its frequency, ϕ, was 100 kHz in this simulation study.

The system (13) was solved using a numerical method (Bogacki–Shampine) with a fixed time step of 1×10^{-8} s. The iterative solution with up to two high-order terms was simultaneously computed for comparison. Fig. 3 demonstrates the SIMULINK model. The left side is the system (13) and the right side calculates the iterative solutions with different orders.

In Figs. 4–6 the waveforms with the caption 'Simulink' present the output of the Simulink model based on (2) and (13), the waveforms with caption V_{c0}, V_{c1} and V_{c2} indicate the iterative solutions

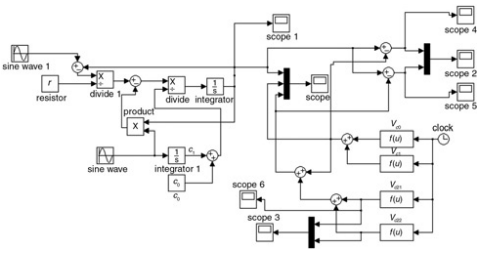


Fig. 3 *Simulink model*

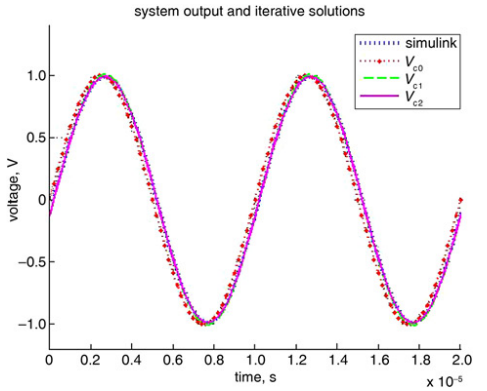


Fig. 4 *Simulation results of the Simulink solution and the iterative solution zoomed in on higher-order terms*

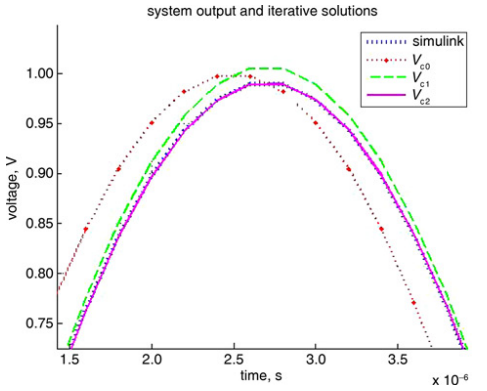


Fig. 5 *Zoomed in simulation results of the Simulink solution and the iterative solution*

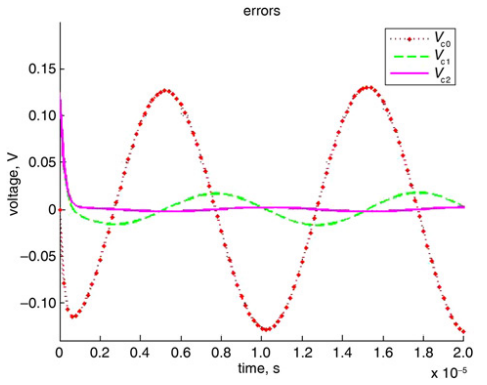


Fig. 6 *Simulation errors between the Simulink solution and the iterative solution zoomed in on higher-order terms*

with different higher-order terms. The waveform V_{c0} is the linear circuit solution.

Fig. 4 shows the time response of the system and the analytical solutions with up to two higher-order terms. The results generated by the proposed method match the theoretical result well in the steady state. For the linear circuit solution, it appears that its amplitude was more precise than the solution with V_{c1}, but it has an obvious phase shift which generated more errors. It is clear that the iterative solution with just one term has more error than the result with two terms. Fig. 5 presents the comparison zoomed in on different higher-order terms. This figure further demonstrates that adding additional high-order terms can increase the solution's accuracy. Fig. 6 shows the errors using different high-order terms. It is shown that the linear circuit analysis yielded the greatest errors, with an amplitude that was more than 0.1 V. The solution with one higher-order term produced less error, with an amplitude that was 0.03 V. In contrast, the error using two higher-order terms was the smallest, which was <0.001 V.

4. Conclusions: An iterative analysis technique was proposed to solve an MEMS device's nonlinear circuit consisting of a time-variant capacitor and an AC power source connected to the MEMS device through a resistive network. A simulation study demonstrated that this method provides a more accurate solution compared with regular linear circuit analysis. The solution's accuracy can be increased by adding additional higher-order terms.

5 References

[1] Li J., Liu Y., Tang M., *ET AL.*: 'Capacitive humidity sensor with a coplanar electrode structure based on anodised porous alumina film', *Micro Nano Lett.*, 2012, **7**, (11), pp. 1097–1100

[2] Bernstein J., Miller R., Kelley W., *ET AL.*: 'Low noise MEMS vibration sensor for geophysical applications', *J. Microelectromech. Syst.*, 1999, **8**, (4), pp. 433–438

[3] Suster M., Guo J., Chaimanonart N., *ET AL.*: 'A high-performance MEMS capacitive strain sensing system', *J. Microelectromech. Syst.*, 2006, **15**, (5), pp. 1069–1077

[4] Narducci M., Yu-Chia L., Fang W., *ET AL.*: 'CMOS MEMS capacitive absolute pressure sensor', *J. Micromech. Microeng.*, 2013, **23**, (5), p. 055007

[5] Fang R., Lu W., Wang G., *ET AL.*: 'A low-noise high-voltage interface circuit for capacitive MEMS gyroscope', *J. Circuits, Syst. Comput.*, 2013, **22**, (9), p. 1340019

[6] Malit F.B., Ramos M.: 'Characterization of RAL bipedal robot capacitive MEMS accelerometer using electrical impedance measurements'. TENCON 2012-2012 IEEE Region Ten Conf., 2012, pp. 1–5

[7] Yuan Q., Luo W., Zhao H., *ET AL.*: 'Frequency stability of RF-MEMS disk resonators', *IEEE Trans. Electron Devices*, 2015, **62**, (5), pp. 1603–1608

[8] Elshurafa A.M., Khirallah K., Tawfik H.H., *ET AL.*: 'Nonlinear dynamics of spring softening and hardening in folded-MEMS comb drive resonators', *J. Microelectromech. Syst.*, 2011, **20**, (4), pp. 943–958

[9] Ghodsian B., Bogdanoff P., Hyman D.: 'Wideband DC-contact MEMS series switch', *Micro Nano Lett.*, 2008, **3**, (3), pp. 66–69

[10] Bashmal S., Oke W., Khulief Y.: 'Vibration analysis of an elastically restrained microcantilever beam under electrostatic loading using wavelet-based finite element method', *Micro Nano Lett.*, 2015, **10**, (3), pp. 147–152

[11] Persano A., Quaranta F., Martucci M.C., Siciliano P., Cola A.: 'On the electrostatic actuation of capacitive rf mems switches on gaas substrate', *Sensors and Actuators A: Physical*, 2015, **232**, pp. 202–207

[12] Mafinejad Y., Kouzani A., Mafinezhad K., *ET AL.*: 'Review of low actuation voltage RF MEMS electrostatic switches based on metallic and carbon alloys', *J. Microelectron. Electro. Compon. Mater.*, 2013, **43**, (2), pp. 85–96

[13] Kitamura M., Kuzumoto Y., Aomori S., *ET AL.*: 'High-frequency organic complementary ring oscillator operating up to 200 kHz', *Appl. Phys. Express*, 2011, **4**, (5), p. 051601

[14] Christen T.: 'A 15-bit 140-W scalable-bandwidth inverter based modulator for a MEMS microphone with digital output', *IEEE J. Solid-State Circuits*, 2013, **48**, (7), pp. 1605–1614

[15] Dong J., Ferreira P.M.: 'Electrostatically actuated cantilever with SOI-MEMS parallel kinematic stage', *J. Microelectromech. Syst.*, 2009, **18**, (3), pp. 641–651

[16] Dean R.N., Anderson A., Reeves S.J., *ET AL.*: 'Electrical noise in MEMS capacitive elements resulting from environmental mechanical vibrations in harsh environments', *IEEE Trans. Ind. Electron.*, 2011, **58**, (7), pp. 2697–2705

[17] Dean R.N. Jr, Flowers G.T., Horvath R., *ET AL.*: 'Characterization and experimental verification of the nonlinear distortion in a technique for measuring the relative velocity between micromachined structures in normal translational motion', *IEEE Sens. J.*, 2007, **7**, (4), pp. 496–501

[18] Dean R.N., Wilson C.G.: 'Nonlinear circuit analysis for time-variant microelectromechanical system capacitor systems', *Micro Nano Lett.*, 2013, **8**, (9), pp. 515–518

[19] Li Y.: 'Monotone iterative method for numerical solution of nonlinear odes in MOSFET RF circuit simulation', *Math. Comput. Model.*, 2010, **51**, (3), pp. 320–328

[20] Lee J.-S., Yoo E.-S., Park C.-H., *ET AL.*: 'Development of a piezoresistive MEMS pressure sensor for a precision air data module'. IEEE 2014 14th Int. Conf. on Control, Automation and Systems (ICCAS), 2014, pp. 874–878

[21] Mukherjee B., Swamy K., Krishnan T., *ET AL.*: 'A simple low cost scheme for closed loop operation of MEMS capacitive accelerometer'. 2014 IEEE Students' Technology Symp. (TechSym), 2014, pp. 111–115

doi: 10.1049/mnl.2015.0276

Enhanced-Performance Control of an Electromagnetic Solenoid System Using a Digital Controller

Chong Li, *Student Member, IEEE*, Haoyue Yang, *Student Member, IEEE*, Luke L. Jenkins, *Student Member, IEEE*, Robert N. Dean, *Senior Member, IEEE*, George T. Flowers, and John Y. Hung, *Fellow, IEEE*

***Abstract*— Solenoids can be used as linear or incremental motion actuators, most often by coupling the solenoid armature to a linear spring. In this configuration, its limited open-loop stable range (less than one-third of the full range) affects performance and limits applications. The input–output linearization control method is an effective technique to extend the stable range. But in practice, however, the time-delay effect from both measurement and actuation can make the system less damped and therefore more sensitive to disturbances. This effect was analyzed and a digital proportional and integrator controller plus extended state observer (ESO) is proposed to enhance the performance of the electromagnetic actuator. Simulation and experimental tests show that this combined proportional and integral and ESO technique can extend the stable range of motion to 77.6% of full stroke with less sensitivity to external disturbances.**

***Index Terms*— Active Disturbance Rejection Control, Extended State Observer, Magnetic Actuators, Nonlinear Control, Solenoids.**

I. Introduction

SOLENOIDS, which are driven by nonlinear magnetic forces, are widely used in many applications, such as positioners [1], digital actuator arrays [2], valves [3]–[5], antilock braking systems [6], aircraft actuation systems [7], [8], vehicle vibration control systems [9], and robotic manipulators [10]. Applications like valves [4] that use solenoids as ON–OFF switch components are focused on whether they have enough power to drive the system, but other applications like positioners [1] are concerned with position control. One potential drawback to this type of actuator is its limited stable travel range, which is less than one-third of the full range of motion. In many harsh environments, external disturbances, such as mechanical vibration, also adversely affect the performance of these actuator systems. Many techniques have been investigated to extend the stable range of motion of these types of actuators, including nonlinear magnetic field mapping [11], a dual solenoid configuration [12], pulsewidth modulation control [13], sliding mode control [14], and adaptive control [15]. However, these proposed solutions have certain limitations.

Manuscript received September 17, 2015; accepted November 19, 2015. Manuscript received in final form November 30, 2015. Recommended by Associate Editor A. Loria.

C. Li, H. Yang, and G. T. Flowers are with the Department of Mechanical Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: czl0047@auburn.edu; hzy0011@auburn.edu; flowegt@auburn.edu).

L. L. Jenkins, R. N. Dean, and J. Y. Hung are with the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: llj0005@auburn.edu; deanron@auburn.edu; hungjoh@auburn.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2015.2507060

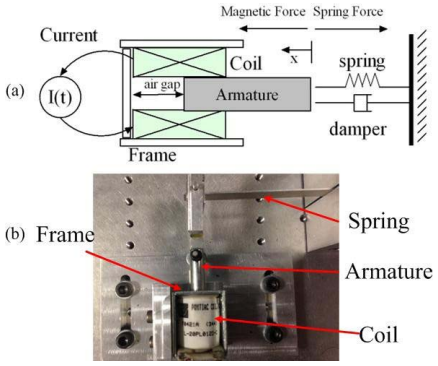


Fig. 1. (a) Illustration of a solenoid with a spring and a damper. (b) Example of a commercial solenoid fixed to a metal cantilever as the spring.

1) They require prior magnetic field distribution knowledge.
2) They suffer from time delay in feedback signals.
3) They result in additional hardware costs.
4) They require complex implementation.
5) They are sensitive to external disturbances.

In this investigation, an input–output linearization (IOL) control technique with a velocity estimator and a Proportional and integral (PI) controller is proposed to extend the stable range and to reject external disturbances. The proposed method requires knowledge of the initial air gap of the solenoid, which is easier to measure compared with the magnetic field. The effect of time delay is analyzed and the performance is improved under the time delay and external disturbances. The controller is implemented using an inexpensive microcontroller. The velocity estimator's structure is straightforward to implement in the microcontroller.

II. System Modeling

A typical configuration of a solenoid in a spring-mass-damper system is illustrated in Fig. 1. Its mathematical description is

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right)+c\frac{dx}{dt}+kx(t)=\frac{\mu_0\mu_r N^2 A I(t)^2}{2(d+x_0-x(t))^2} \quad (1)$$

where $x(t)$ is the displacement of the armature, μ_0 is the permeability of free space, A is the cross-sectional area of the core, N is the number of the turns of the coil, μ_r is the relative permeability of the dielectric material between

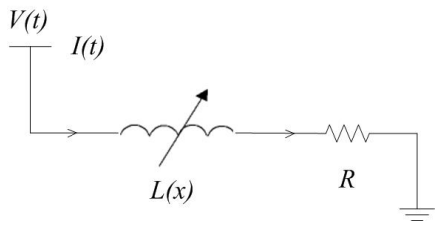


Fig. 2. Solenoid equivalent circuit.

the coil and the armature, x_o is the initial air gap between the armature and the back side of the frame, and d is the additional initial air gap related to the solenoid's geometry, which is much smaller than x_0 in general. Also, m is the proof mass of the armature, k is the stiffness of the spring, and c is the system damping coefficient. $I(t)$ is the current through the coil [12], [13]. In practice, the input to a solenoid is typically a voltage rather than a current. Given a voltage source $V(t)$ driving the solenoid, the equivalent circuit model for the solenoid is shown in Fig. 2. R is the series resistance of the solenoid coil, $I(t)$ is the current through the solenoid, and $L(x)$ is the inductance of the coil that depends on the air gap x [12], [13], which is

$$L(x) = \frac{\mu_0 \mu_r A}{d + x_0 - x(t)}. \tag{2}$$

The current determines the magnetic force. Cheung *et al.* [16] have studied the current dynamics of solenoids driven by a voltage source. Since $L(x)$ is a variable inductor, the dynamics of the circuit in Fig. 2 are modeled as

$$L(x)\frac{dI(t)}{dt} + \frac{dL(x)}{dt}I(t) + I(t)R = V(t). \tag{3}$$

According to (1) and (2), the electrical and mechanical dynamics are coupled with each other, which makes solving (3) difficult. To evaluate the behavior between the input voltage and the actual current in the circuit, a linear circuit analysis is applied first. It assumes that the variable inductor $L(x)$ has a constant value

$$L_1 = L(x^*) \tag{4}$$

around an equilibrium point x^*. Thus, (3) can be simplified as

$$L_1\frac{dI}{dt} + I(t)R = V(t). \tag{5}$$

Its transfer function can be described by

$$\begin{aligned}\frac{I(s)}{V(s)} &= \frac{1}{L_1 s + R} \\ &= \frac{1/R}{(L_1/R)s + 1}.\end{aligned} \tag{6}$$

The transfer function indicates that there is always a time delay between the voltage source and the current source. According to linear system theory, the settling time t_s of this first order system (6) is

$$t_s = 4(L_1/R). \tag{7}$$

However, this settling time approximation is valid only when the mechanical motion is sufficiently small around the equilibrium point x^*. Otherwise, the settling time is uncertain. Substituting the control voltage into (1), considering the uncertain time-delay effect and ignoring the parameter d, the system model can be considered as

$$m\frac{d}{dx}\left(\frac{dx}{dt}\right) + c\frac{dx}{dt} + kx(t) = \frac{\beta V(t-\tau)^2}{2(x_0 - x(t))^2} \tag{8}$$

where β is the time-invariant coefficient of the solenoid

$$\beta = \frac{\mu_0 \mu_r N^2 A}{2R^2} \tag{9}$$

where τ is the uncertainty time-delay coefficient, which depends on the solenoid's parameters R and $L(x)$.

III. Controller Design and Analysis

A. Ideal Input–Output Linearization

To extend the stable range and performance of the solenoid system, closed-loop control techniques are necessary. An IOL control method and a low-voltage controller were demonstrated in [17] and [18] for a microelectromechanical system (MEMS) electrostatic actuator, which possesses dynamics very similar to that of solenoids. Both types of actuators are approximately square law devices. Adopting this method based on model (1) and using a current source, an IOL law can be introduced to approximately cancel the nonlinearity

$$I(t) = (x_0 - x(t))\sqrt{k_c I_c(t)} \tag{10}$$

where k_c is a positive gain on the input and $I_c(t)$ is the new control input, where $I_c(t) > 0$. Therefore, the system becomes

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right) = -c\frac{dx}{dt} - kx(t) + \frac{k_c \mu_0 \mu_r A N^2}{2} I_c(t) \tag{11}$$

which mathematically cancels out the square law dependence on current.

B. Input–Output Linearization Technique With Time Delay

However, the system could easily be adversely affected by measurement time delay or controller time delay. Therefore, the control law based on (10) using a voltage source becomes

$$V(t) = (x_0 - x(t))\sqrt{k_c V_c(t)} \tag{12}$$

where $V_c(t)$ is the new control input and $V_c(t) > 0$. In this situation, the system dynamics become

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right) = -c\frac{dx}{dt} - kx(t) + \frac{k_c \beta (x_0 - x(t-\tau))^2 V_c(t)}{(x_0 - x(t))^2}. \tag{13}$$

Solenoids are macroactuators, which have response times slower than electronic circuits. Thus, the time delay is relatively small compared with the mechanical system. Then, using Taylor series to expand the time-delay term, where

$$x(t-\tau) \approx x(t) - \tau\frac{dx}{dt} \tag{14}$$

and neglecting the higher order terms, the system model becomes

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right) = -c\frac{dx}{dt} - kx(t) + k_c\beta V_c(t) + \frac{2k_c\beta\tau}{x_0 - x(t)}\frac{dx}{dt}V_c(t) + \frac{k_c\beta\tau}{(x_0 - x(t))^2}\left(\frac{dx}{dt}\right)^2 V_c(t). \tag{15}$$

Note that there is a nonlinear damping effect

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right) = -g\left(x(t), \frac{dx}{dt}, V_c(t)\right) - kx(t) + k_c\beta V_c(t). \tag{16}$$

It is necessary to evaluate the influence of the nonlinear function g to the ideal system (11). A state variable model and its small signal analysis will be applied. At first, select a group of state variables

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ \dfrac{dx}{dt} \end{bmatrix}. \tag{17}$$

In practice, only the displacement variable $x_1(t)$ is available. Considering this constraint and adding an observation equation that detects the displacement, (16) can be rewritten as

$$\begin{aligned} \frac{dx_1}{dt} &= x_2(t) \\ m\frac{dx_2}{dt} &= -g(x_1(t), x_2(t), V_c(t)) - kx_1(t) + k_c\beta V_c(t) \\ y(t) &= [1 \quad 0]\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \end{aligned} \tag{18}$$

At an equilibrium point, the displacements x_{1e}, x_{2e} and the control input V_{ce} satisfy

$$\begin{cases} kx_{1e} = k_c\beta V_{\text{ce}} \\ x_{2e} = 0. \end{cases} \tag{19}$$

According to (16), a group of new state variables around a certain equilibrium point is selected as

$$\begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \\ \delta V_e(t) \end{bmatrix} = \begin{bmatrix} x_1(t) - x_{1e} \\ x_2(t) - x_{2e} \\ V_c(t) - V_{\text{ce}}(t) \end{bmatrix}. \tag{20}$$

The linearized small signal model is

$$\begin{aligned} \begin{bmatrix} \delta\dfrac{dx_1}{dt} \\ \delta\dfrac{dx_2}{dt} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{c}{m} + \dfrac{2k_c\beta\tau}{m(x_0 - x_{1e})} \end{bmatrix}\begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ k_c\beta \end{bmatrix}\delta V_{\text{ce}}(t) \\ \delta y(t) &= [1 \quad 0]\begin{bmatrix} \delta x_1(t) \\ \delta x_2(t) \end{bmatrix}. \end{aligned} \tag{21}$$

For evaluating its influence, it is convenient to examine the transfer function of (21)

$$G_1(s) = \frac{X_1(s)}{V(s)} = \frac{k_c\beta/k}{\frac{m}{k}s^2 + \frac{m}{k}\left(c - \frac{2\beta\tau}{x_0 - x_{1e}}\right)s + 1} \tag{22}$$

which can be compared with the ideal model

$$G_2(s) = \frac{X_1(s)}{V(s)} = \frac{k_c\beta/k}{\frac{m}{k}s^2 + \frac{m}{k}cs + 1}. \tag{23}$$

The actual damping is less than that predicted by the ideal model. As a result, higher inputs could lead the system to oscillate or become unstable. Therefore, extra damping needs to be introduced to the system to ensure acceptable performance.

C. Linearization With Extended State Observer

Damping can be increased by measuring or estimating proof mass velocity and feeding back this term. Directly differentiating the displacement is inaccurate due to measurement noise. Therefore, velocity observers are necessary to obtain a better estimate for feedback. However, considering the system uncertainty, regular observers are difficult to construct. Active disturbance rejection control (ADRC) technique is an alternative approach that can solve this kind of problem. It has been demonstrated that extended state observers (ESOs) can estimate system states without certain system models [19]–[21]. The key idea of this technique is to consider the nonlinear parts of the system as the external state. Considering that a nonlinear second-order system has the form of

$$\begin{aligned} \frac{dx_1^*}{dt} &= x_2^*(t) \\ \frac{dx_2^*}{dt} &= f^*\left(x_1^*(t), x_2^*(t), w(t)\right) + bu(t) \end{aligned} \tag{24}$$

where $x_1^*(t)$ and $x_2^*(t)$ are the state variables, $w(t)$ are the external disturbances, $f^*(x_1^*(t), x_2^*(t), w(t))$ is a nonlinear function with the variables and the disturbance, b is the input coefficient, and $u(t)$ is the control input. Han [19] proposed a solution that can eliminate this issue using the measurement combined with a series of nonlinear functions. However, this method's parameters lack guidelines for adjustment. A linear approach was illustrated with tuning methods provided for ADRC and ESO [20], [21]. Based on the linear approach, the system is augmented as

$$\begin{aligned} \begin{bmatrix} \dfrac{dz_1}{dt} \\ \dfrac{dz_2}{dt} \\ \dfrac{dz_3}{dt} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}w(t) \\ y(t) &= [1 \quad 0 \quad 0]\begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix}. \end{aligned} \tag{25}$$

Therefore, the estimator's structure becomes

$$\begin{aligned} \frac{d\hat{z}_1}{dt} &= \hat{z}_2(t) + L_1(y(t) - \hat{z}_1(t)) \\ \frac{d\hat{z}_2}{dt} &= \hat{z}_3(t) + L_2(y(t) - \hat{z}_1(t)) + bu(t) \\ \frac{d\hat{z}_3}{dt} &= L_3(y(t) - \hat{z}_1(t)) \end{aligned} \tag{26}$$

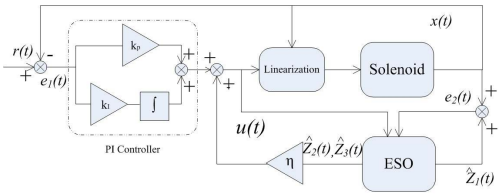


Fig. 3. Block diagram of the controller with IOL and PIESO.

where $\hat{z}_1(t)$, $\hat{z}_2(t)$, and $\hat{z}_3(t)$ are the estimated value of $z_1(t)$, $z_2(t)$, and $z_3(t)$, respectively. In this case, let the displacement $x_1(t)$ be $z_1(t)$, the velocity $x_2(t)$ be $z_2(t)$, the external disturbances be $z_3(t)$, and $y(t)$ be the input $u(t)$. L_1, L_2, and L_3 are observer parameters to be chosen, which will decide the bandwidth of the estimator. Using this estimated velocity $\hat{z}_2(t)$ and the external disturbance $\hat{z}_3(t)$, extra damping and disturbance rejection are introduced to the control law

$$V(t) = (x_0 - x(t))\sqrt{k_c V_c(t) - \eta_1 \hat{z}_2(t) - \eta_2 \hat{z}_3(t)} \quad (27)$$

where η_1 and η_2 are positive numbers. Hence, the damping is enhanced and external disturbances can be reduced

$$m\frac{d}{dt}\left(\frac{dx}{dt}\right) = -g\left(x(t), \frac{dx}{dt}, V_c(t)\right) - kx(t) + k_c\beta V_c(t) - \eta_1\beta\hat{z}_2(t) - \eta_2\beta\hat{z}_3(t). \quad (28)$$

Because there is a time delay between the controller's command and settling the actual current signal, the control effort with the ESO will also have the time-delay effect. η_1 and η_2 can be tuned by starting from small values, and then increasing their values until the performance becomes satisfactory.

Based on (8), $\hat{z}_2(t)$ with time delay will mathematically decrease the mass of the armature, which will not affect dynamic performances.

D. Linearization With PIESO

For the sake of the system's response time and position tracking, a proportional and integral (PI) controller can be placed ahead of the control law given in (27). The overall system diagram with proportional and integral and ESO (PIESO) is shown in Fig. 3. In this control diagram, r is the reference input, $e_1(t)$ is the error between the displacement and the input, $e_2(t)$ is the estimation error between the estimator and the measured displacement, and $u(t)$ is the actual voltage across the solenoid. The PI controller with respect to $V_c(t)$ is

$$V_c(t) = K_p e_1(t) + K_i \int e_1(t)dt. \quad (29)$$

Thus, (27) and (29) can be combined together to enhance the system's performance.

E. Digital Implementation

Digital controllers have certain advantages.

1) Complicated control algorithms can be implemented.
2) State observers are available.
3) They are less sensitive to the harsh environment in which the system must operate.

The implementation of the control law though a microcontroller requires discrete form. Given the sample rate T, (29) becomes

$$V_c(k) = V_c(k-1) + K_p(e_1(k) - e_1(k-1)) + \frac{K_I}{T}e_1(k) \quad (30)$$

and (27) can be approximated as

$$V(k) = (x_0 - x(k))\sqrt{k_c V_c(k) - \eta_1 \hat{z}_2(k) - \eta_2 \hat{z}_3(k)}. \quad (31)$$

Since (31) requires the estimated value of velocity $\hat{z}_2(k)$ and external disturbance $\hat{z}_3(k)$, the discrete form of the ESO is

$$\begin{aligned}\hat{z}_1(k) &= \hat{z}_1(k-1) + T(\hat{z}_2(k-1) + L_{d1}(y(k-1) - \hat{z}_1(k-1)))\\ \hat{z}_2(k) &= \hat{z}_2(k-1) + T(\hat{z}_3(k-1) + L_{d2}(y(k-1) - \hat{z}_1(k-1)) + bu(k))\\ \hat{z}_3(k) &= \hat{z}_3(k-1) + T(L_{d3}(y(k-1) - \hat{z}_1(k-1)))\end{aligned} \quad (32)$$

where L_{d1}, L_{d2}, and L_{d3} are parameters to be chosen to adjust the bandwidth of the estimator. The tuning method is introduced by [20]. Equations (27), (30), and (32) are difference equations that are easy to program in microcontrollers.

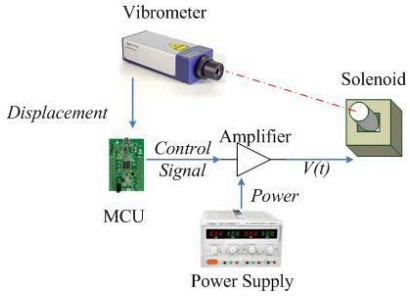


Fig. 4. System configuration.

IV. System Configuration

The experimental validation system included a solenoid system, a laser vibrometer, a microcontroller, a power supply, and an amplifier, which is illustrated in Fig. 4. A commercial solenoid, Pontic F421, was configured for the test. Its coil resistance was 22.5 Ω with a maximum voltage input of 15 V. The solenoid was attached to a metal fixture to hold it, and a thin metal cantilever was used as the spring for the actuator system. Mechanical testing of the spring indicated that the spring stiffness was 302.98 N/m, and the damping coefficient was 42.03 kg/s. Considering the detection range of the displacement sensor and the initial force of the solenoid, the total displacement was configured to be 6.96 mm. The natural frequency of the solenoid system was 16.4 Hz, which indicated its bandwidth. At the initial position, the inductance was 45.5 mH, so based on (7), in the worst case, the time delay was approximately 8 ms.

An STM32F407 microcontroller was used for this experiment, which had a 168-MHz core and was assembled with a 12-b analog-to-digital/digital-to-analog converter. Thus, the solenoid required a 12 V at 0.5-A power supply. A support electronics circuit board with a push–pull amplifier was used to support the microcontroller board and the solenoid system. Proof mass displacement detection was accomplished using a Poly-tech OFV353 laser vibrometer. The vibrometer output was an analog signal with a data rate of 40 kHz and a resolution of 0.1 mm. The control interval T was chosen as 5 ms.

After system identification, the final model was

$$22.7 \times 10^{-3}\frac{dx}{d}\left(\frac{dx}{dt}\right) + 42.03\frac{dx}{dt} + 302.89x(t) = \frac{2.107 \times 10^{-4} I(t)^2}{(6.65 \times 10^{-3} - x(t))^2}. \quad (33)$$

Considering the natural frequency, the desired bandwidth was chosen as $F_c = 16$ Hz. Based on the tuning method given in [5] and [21], the gains of the ESO were

$$L_{d1} = 0.3758, \quad L_{d2} = 12.0641, \quad L_{d3} = 122.8657. \quad (34)$$

V. System Simulation

Prior to experimental validation, a system simulation was performed using MATLAB SIMULINK. The target actuator is the configured one above. First, the time-delay effect on the IOL was verified. The original system had a maximum stable range of 2.3 mm in theory, so a set point of 2 mm was chosen to evaluate the problem. Then the PIESO controller was simulated to extend its stable range and to follow the settling point. After that, the PIESO controller's performance under external disturbances was evaluated using sinusoidal signals to model the disturbances. The results are given and discussed in Section VI.

VI. Experimental Validation

A series of experiments was performed using the hardware in Section IV. Considering the bandwidth and numerical resolution, the control rate was chosen to be 200 Hz. Test plans were the same as in the simulation, which included the time-delay effect and the PIESO controller with and without disturbances. The sinusoidal disturbances were vibrational and were generated by an ling dynamic systems (LDS)-850 large shaker, which is shown in Fig. 5.

A. Time-Delay Effect

Fig. 6 demonstrates the time-delay effect of the damping. In Fig. 6, the curves are the step responses which occur at time 0 s. The simulation results indicate that the original system is overdamped, there is no overshoot and it reaches steady state in less than 0.1 s. By contrast, the IOL applied with time delay reduces the damping coefficient so that its overshoot can be larger than 75% of the steady state. When ESO introduces the extra damping, the system converges rapidly within 0.1 s and only one overshoot peak is smaller than 40% of the set point. The results from simulation and

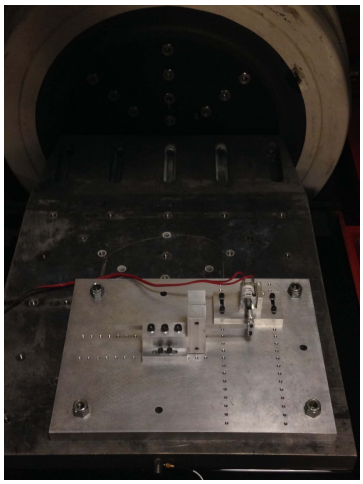


Fig. 5. System mounted on the LDS-850 large shaker.

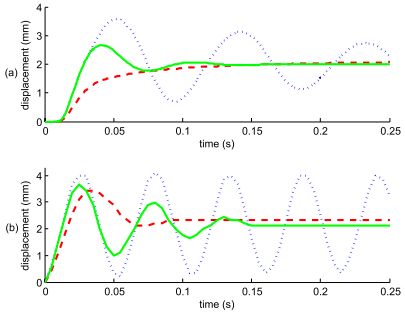


Fig. 6. Comparison of step responses of open loop (red dotted line), IOL (blue dashed line), and IOL with ESO (green solid line). (a) Simulation part. (b) Experimental part.

experiments support the point of view that time delay can decrease the damping with IOL. The original damping led to a fast convergence time and small overshoots and IOL decreased it, which took more than 0.6 s to reach the final state. The PIESO increased the damping so that the number of overshoot peaks was much less than IOL and it converges within 0.15 s. Besides the time-delay effect, the measured initial gap is not perfectly known, which cannot totally cancel the driven force's nonlinearity that also affects the feedback controller's performances.

B. PIESO Without Disturbances

This test verified the effectiveness of PIESO to extend the solenoid's travel range and to follow the reference input. In practice, a maximum stable position of 5 mm was achieved, which reached 77.6% of the total range. Also considering that the maximum stable range of the open-loop system was 2.2 mm, the set points were 2.2, 4, and 5 mm, respectively. The step responses occurring at 0.5 s are illustrated in Fig. 7.

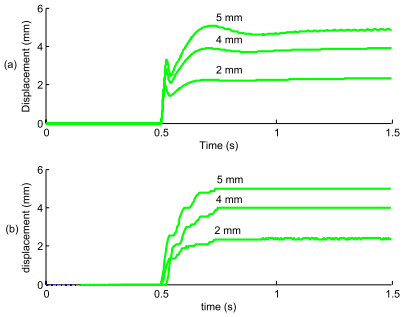


Fig. 7. Comparison of PIESO controller's step responses with set points of 2, 4, and 5 mm. (a) Simulation. (b) Experiment.

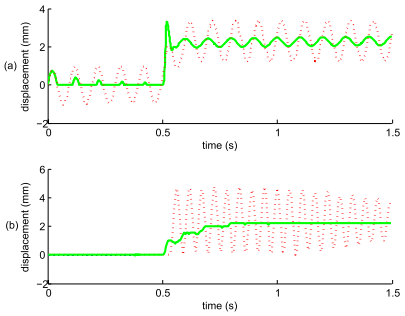


Fig. 8. Comparison of IOL and PIESO controller's responses via a set point of 2.4 mm with a 10-Hz 1-mm amplitude sinusoidal disturbance: IOL (red dotted line) and PIESO (green solid line). IOL with a set point higher than 2.4 mm failed. (a) Simulation. (b) Experiment.

The results from both simulations and experiments show that the system can achieve the desired point without error. In the simulation, responses had overshoots and took more time (around 0.3–0.4 s) to converge. The experimental results ran faster (around 0.2–0.3 s), but suffered from inaccurate measurement, which led to chattering around the desired position. The trace of the reference input equaling 2 mm in the experiment is an example. A further set point failed, which was primarily due to the solenoid having a permanent magnet in the bottom, which is designed to provide more sealed force, which significantly changes the system's behavior from the developed model as displacement approaches this point. The results prove that PIESO can extend the solenoid's stable travel range and eliminate the displacement error.

C. PIESO With Disturbances

Finally, Figs. 8–10 illustrate the step responses of IOL and PIESO under disturbances. Based on the displacement sensor's capability and the system's natural frequency, disturbances were chosen as sinusoidal disturbances with an amplitude of 1 mm and frequencies of 10, 16, and 20 Hz, which is below, at, and above its natural frequency, respectively. The amplitudes of the disturbances are all 1 mm. For clarity, the set points

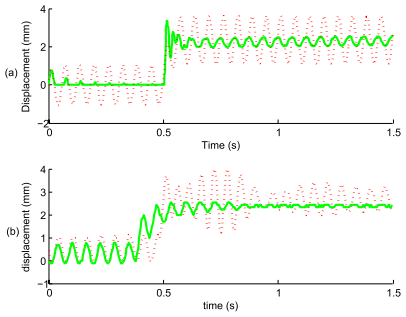


Fig. 9. Comparison of IOL and PIESO controller's responses via a set point of 2.4 mm with a 16-Hz 1-mm amplitude sinusoidal disturbance: IOL (red dotted line) and PIESO (green solid line). IOL with a set point higher than 2.4 mm failed. (a) Simulation. (b) Experiment.

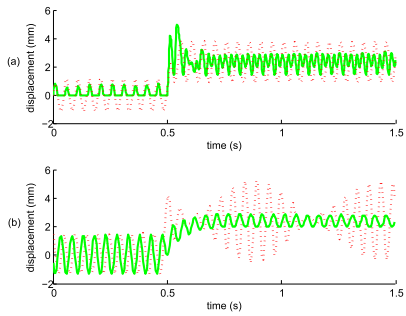


Fig. 10. Comparison of IOL and PIESO controller's responses via a set point of 2.4 mm with a 20-Hz 1-mm amplitude sinusoidal disturbance: IOL (red dotted line) and PIESO (green solid line). IOL with a set point higher than 2.4 mm failed. (a) Simulation. (b) Experiment.

are 2.4, 4, and 5 mm, respectively. When the disturbance frequency is 10 Hz, the output is a sinusoidal signal that has the same frequency and amplitude as the input, in theory. The IOL response matches this point in the simulation. Though the average final value is 2 mm, it ripples around it. The higher set point failed both in simulation and in experiments. The system will snap in together or oscillate with a very large amplitude. The PIESO can extend its stable range to 5 mm (77.6% of total range) at this situation. However, it also oscillates around the desired position. At this frequency, the oscillation has an amplitude of about 0.1 mm. In practice, the external disturbance does not appear until the reference input command is given. It is caused by the complicated solenoid fixture. The static friction force holds the stroke when the external disturbances have insufficient energy. The IOL oscillates with a large amplitude (larger than 4 mm) and does not converge within 1 s. The PIESO can also obtain a stable range of 77.6% with a rise time around 0.2 s with little oscillation.

The responses with a disturbance of 16 Hz in simulation are similar to the results with 10 Hz, but the PIESO's performances are worse. The settling times are longer than 0.5 s and

oscillation amplitudes are around 0.4 mm. The IOL response in the experimental test shows a higher excited oscillation and the PIESO's results show that it can reach the desired point without oscillation within 0.2 s.

The results with the disturbances even above the natural frequency demonstrate that the external disturbances can influence the system's performance. The simulation and experimental IOL responses demonstrate that the output amplitude is amplified at this frequency (3 mm in the simulation and 3.5 mm in the experiment). The displacements after a step input using IOL also vibrated with large amplitudes and will not converge. In simulation, the PIESO's responses also oscillated at all three set points, but the amplitudes are 70% less than the IOL. In contrast, the PIESO performs well at set points of 4 and 5 mm in practice, but oscillates at 2.4 mm. At the set point of 2.4 mm, the PIESO vibrates with an amplitude of about 0.5 mm, while there are no oscillations at the set points of 4 and 5 mm. The settling times of PIESO are all less than 0.3 s in the experiments. The PIESO can cancel the disturbances, maintain the set position and extend the stable range. Typically, a higher set point can cancel the disturbances better because the spring force is stronger, while the magnetic force can only be applied in one direction. The output response's amplitude at a frequency that is above its resonant frequency is higher than an amplitude that is below the resonant frequency, because the control method changes the damping coefficient and equivalent mass. The experimental results were better than the simulation results for two reasons.

1) The simulations did not consider the effect of the static and dynamic frictions, but instead used a constant damping coefficient.
2) To avoid the unstable numerical solutions caused by an algebrical loop, there was a memory unit in the simulation feedback loop, which could introduce more time delay to the system.

VII. Conclusion

Reviewing the simulation and experimental results, it is clear that the time delay has the effect of decreasing the damping. Even if the original mechanical system is not slightly damped, time delay caused by measurement and actuation can influence the IOL technique's performance. In reality, the existence of friction may enhance the performance under the time-delay problem, but it is not reliable. The PIESO can effectively extend the stable range and maintain the position without perfect knowledge of the system characteristics. This requires accurate feedback and adjusting the ESO's parameters depending on the designer's experiences.

References

[1] M.-Y. Chen, H.-H. Huang, and S.-K. Hung, "A new design of a submicropositioner utilizing electromagnetic actuators and flexure mechanism," *IEEE Trans. Ind. Electron.*, vol. 57, no. 1, pp. 96–106, Jan. 2010.

[2] L. Petit, A. Hassine, J. Terrien, F. Lamarque, and C. Prelle, "Development of a control module for a digital electromagnetic actuators array," *IEEE Trans. Ind. Electron.*, vol. 61, no. 9, pp. 4788–4796, Sep. 2014.

[3] T. Kajima and Y. Kawamura, "Development of a high-speed solenoid valve: Investigation of solenoids," *IEEE Trans. Ind. Electron.*, vol. 42, no. 1, pp. 1–8, Feb. 1995.

[4] D. Dong and X. Li, "Simulation and experimental research on the response of a novel high-pressure pneumatic pilot-operated solenoid valve," in *Proc. IEEE 19th Int. Conf. Mechatron. Mach. Vis. Pract. (M2VIP)*, Nov. 2012, pp. 480–484.

[5] K. Mutschler *et al.*, "Multi physics network simulation of a solenoid dispensing valve," *Mechatronics*, vol. 24, no. 3, pp. 209–221, 2014.

[6] L. Chu, Y. Hou, M. Liu, J. Li, Y. Gao, and M. Ehsani, "Study on the dynamic characteristics of pneumatic ABS solenoid valve for commercial vehicle," in *Proc. IEEE Vehicle Power Propuls. Conf. (VPPC)*, Sep. 2007, pp. 641–644.

[7] C. C. A. Eguti and L. G. Trabasso, "Design of a robotic orbital driller for assembling aircraft structures," *Mechatronics*, vol. 24, no. 5, pp. 533–545, 2014.

[8] Y. Li, H. Lu, S. Tian, Z. Jiao, and J.-T. Chen, "Posture control of electromechanical-actuator-based thrust vector system for aircraft engine," *IEEE Trans. Ind. Electron.*, vol. 59, no. 9, pp. 3561–3571, Sep. 2012.

[9] U. Koch, D. Wiedemann, and H. Ulbrich, "Model-based MIMO state-space control of a car vibration test rig with four electromagnetic actuators for the tracking of road measurements," *IEEE Trans. Ind. Electron.*, vol. 58, no. 12, pp. 5319–5323, Dec. 2011.

[10] A. Takai, N. Alanizi, K. Kiguchi, and T. Nanayakkara, "Prototyping the flexible solenoid-coil artificial muscle, for exoskeletal robots," in *Proc. IEEE 13th Int. Conf. Control, Autom., Syst. (ICCAS)*, Oct. 2013, pp. 1046–1051.

[11] K. W. Lim, N. C. Cheung, and M. F. Rahman, "Proportional control of a solenoid actuator," in *Proc. IEEE 20th Int. Conf. Ind. Electron., Control, Instrum. (IECON)*, vol. 3. Sep. 1994, pp. 2045–2050.

[12] L. Yu and T. N. Chang, "Zero vibration on–off position control of dual solenoid actuator," *IEEE Trans. Ind. Electron.*, vol. 57, no. 7, pp. 2519–2526, Jul. 2010.

[13] M. Taghizadeh, A. Ghaffari, and F. Najafi, "Modeling and identification of a solenoid valve for PWM control applications," *Compt. Rendus Mécanique*, vol. 337, no. 3, pp. 131–140, 2009.

[14] T. Nguyen, J. Leavitt, F. Jabbari, and J. E. Bobrow, "Accurate sliding-mode control of pneumatic systems using low-cost solenoid valves," *IEEE/ASME Trans. Mechatronics*, vol. 12, no. 2, pp. 216–219, Apr. 2007.

[15] C. Nataraj and D. Lee, "Model-based adaptive control for a solenoid-valve system," in *Proc. Int. Mech. Eng. Congr. Expo.*, 2010, pp. 1009–1015.

[16] N. C. Cheung, K. W. Lim, and M. F. Rahman, "Modelling a linear and limited travel solenoid," in *Proc. IEEE Int. Conf. Ind. Electron., Control, Instrum. (IECON)*, Nov. 1993, pp. 1567–1572.

[17] C. Li, R. N. Dean, G. T. Flowers, and J. Y. Hung, "Nonlinear feedback control to enhance stable performance of micromachined electrostatic parallel plate actuators," *Int. J. Autom. Power Eng.*, vol. 3, no. 1, pp. 49–52, 2014.

[18] R. N. Dean, C. G. Wilson, J. P. Brunsch, and J. Y. Hung, "A synthetic voltage division controller to extend the stable operating range of parallel plate actuators," in *Proc. IEEE Int. Conf. Control Appl. (CCA)*, Sep. 2011, pp. 1068–1074.

[19] J. Han, "From PID to active disturbance rejection control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 900–906, Mar. 2009.

[20] C. Xing, L. Donghai, G. Zhiqiang, and W. Chuanfeng, "Tuning method for second-order active disturbance rejection control," in *Proc. 30th Chin. Control Conf. (CCC)*, Jul. 2011, pp. 6322–6327.

[21] R. Miklosovic, A. Radke, and Z. Gao, "Discrete implementation and generalization of the extended state observer," in *Proc. IEEE Amer. Control Conf.*, Jun. 2006, pp. 2209–2214.

International Journal of Automation and Power Engineering (IJAPE) Volume 3 Issue 1, January 2014 www.ijape.org
DOI: 10.14355/ijape.2014.0301.10

Nonlinear Feedback Control to Enhance Stable Performance of Micromachined Electrostatic Parallel Plate Actuators

Chong Li*[1], Robert N. Dean[2], George T. Flowers[3], John Y. Hung[4]

[1,3]Dept. of Mechanical Engineering,
[2,4]Dept. of Electrical & Computer Engineering
Auburn University, Auburn, AL USA

*[1]czl0047@auburn.edu; [2]deanron@auburn.edu; [3]flowegt@auburn.edu; [4]hungjoh@auburn.edu

Abstract

MEMS parallel plate actuators on elastic suspensions are open loop stable over one third of their rest gap distance. To increase the range of stable motion, a nonlinear feedback controller is proposed in this paper. Verification and error analysis are performed using MATLAB Simulink.

Keywords

MEMS; Parallel Plate Actuator; Nonlinear Control

Introduction

Electrostatic parallel-plate actuators (PPA) are used in many types of MEMS devices, such as accelerometers, variable capacitors, RF devices, and micromirrors. PPAs operate by reducing the distance between the electrodes in response to an applied voltage. A PPA has an open loop voltage controlled stable displacement range, x, of $0 \le x < x_o/3$, where x_o is the rest gap distance between the two electrodes. Open loop attempts to further increase displacement result in unstable motion with the two electrodes snapping into contact. A number of techniques have been investigated for increasing the stable displacement range using various controller architectures. Examples include the series capacitor method, synthetic voltage division, charge control, negative capacitance control, various electrode configurations, voltage driven linear feedback control, nonlinear output feedback stabilization, and nonlinear input transformation. Each of these techniques offers certain benefits and detriments.

A PPA is a nonlinear square law device, where the force is proportional to drive voltage squared and inversely proportional to electrode separation distance squared. If the characteristics of the actuator are well known and the movable electrode displacement can be accurately and quickly measured, then a nonlinear gain term can be utilized to linearize and stabilize the operation of the PPA system. The remainder of this manuscript introduces this technique.

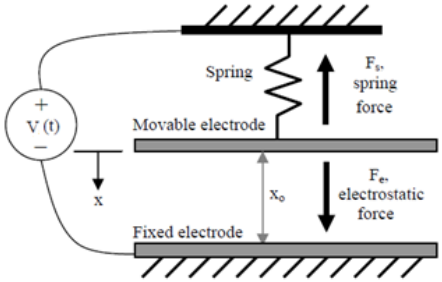


FIG. 1 AN ILLUSTRATION OF A PPA

Background

Actuator Model

Consider the PPA in Fig. 1 when it is integrated into a second order spring-mass-damper mechanical system:

$$m\ddot{x} + c\dot{x} + kx = f(x,V) = \frac{\varepsilon A V^2}{2(x_o - x)^2} \quad (1)$$

where x is the displacement of the movable electrode, ε is the permittivity of free space multiplied by the relative permittivity of the dielectric material between the two electrodes, A is the overlapping surface area of the electrodes and x_0 is the rest gap distance between the electrodes. V is the drive voltage between the two electrodes.

At equilibrium:

$$V = \sqrt{\frac{2x(x_o - x)^2}{\varepsilon A}} \quad (2)$$

Stability Analysis

A linear approximation by (1) can be formed by Taylor

series expansion of f(x, V). From the Taylor series:

$$\frac{\partial f(x,V)}{\partial x} = \frac{\varepsilon A V^2}{(x_o - x)^3} \tag{3}$$

Let x_a be a desired displacement over the stable range and V_a be the voltage from (2). Then:

$$f(x,V) \approx \frac{\frac{\varepsilon A V_a^{\,2}}{2(x_o - x_a)^2}}{0!}(x - x_a)^0 + \frac{\frac{\varepsilon A V_a^{\,2}}{(x_o - x_a)^3}}{1!}(x - x_a)^1 \tag{4}$$

N and K_{EL} are defined as:

$$N = \frac{\varepsilon A V_a^{\,2}}{2(x_o - x_a)^2} - \frac{x_a \varepsilon A V_a^{\,2}}{(x_o - x_a)^3}$$
$$k_{EL} = \frac{\varepsilon A V_a^{\,2}}{(x_o - x_a)^3} \tag{5}$$

Then substituting (5) for f(x,V) in (1) yields the linear approximation:

$$m\ddot{x} + c\dot{x} + (k - k_{EL})x = N \tag{6}$$

The characteristic equation has root at:

$$s_{1,2} = \frac{\frac{-c}{m} + \sqrt{\left(\frac{c}{m}\right)^2 - 4\left(\frac{k - k_{EL}}{m}\right)}}{2} \tag{7}$$

This system is only stable by open loop voltage control for $0 \le x \le x_o/3$. If the PPA voltage is increased in attempt to increase the displacement beyond $x_o/3$, the two electrodes will snap into contact.

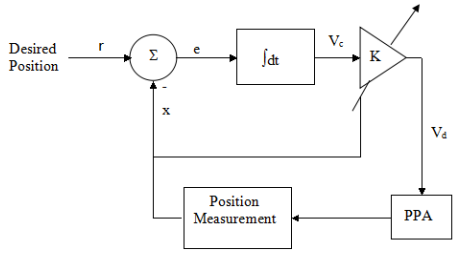


FIG. 2 IMPLEMENTATION OF A PPA NONLINEAR FEEDBACK CONTROL LAW WITH A INTEGRATOR CONTROLLER.

Nonlinear Feedback Control

Ideal Feedback Linearization Control

Consider the following nonlinear mapping function K:

$$K(V_c, x) = \sqrt{V_c}(x_0 - x) = V_d \tag{8}$$

V_d is the actual voltage across the PPA, and V_c is the control voltage before the nonlinear mapping function.

Substituting (8) for V in model (1), the resulting electrostatic force equation becomes:

$$f(x,v) = \frac{\varepsilon A V_c}{2} \tag{9}$$

which is a linear function of the control voltage, V_c. The PPA with nonlinear input map (9) becomes a linear system. A similar idea was proposed in,but no implementation was discussed. In this paper, the authors suggested a integrator closed loop controller, as illustrated in Fig. 2, that can then be used to adjust the displacement over the entire range of actuator with the nonlinear gain, K.

However, if the system is not perfectly implemented, the controller does not perform as designed. The primary sources of error are measurement error in displacement and error in the model for the rest gap displacement. Evaluating each error source independently allows for analysis of each error source.

Stable Range Analysis with Displacement Error

Error in measurement x can be modeled as a sensor scaling. The measured displacement x′ is a positive constant. If there is no scaling error, α=1.

$$x' = \alpha x \tag{10}$$

The PPA with the mapping can be modeled as:

$$f(x,V) = \frac{\varepsilon A(x_o - \alpha x)^2 V_c}{2(x_o - x)^2} \tag{11}$$

The new equilibrium condition is:

$$kx = \frac{\varepsilon A(x_o - \alpha x)^2 V_c}{2(x_o - x)^2} \tag{12}$$

Stability analysis about the condition for all poles to be in the left half s-plane is:

$$\frac{2x}{(x_o - x)} - \frac{2\alpha x}{(x_o - \alpha x)} < 1 \tag{13}$$

Therefore, the measurement error could decrease the stable traveling range of a PPA.

Stable Range Analysis with Initial Gap Error

Consider the effect of error in the model for the initial gap distance. The initial gap is scaled by β. If there is no manufacturing imperfection, then β=1.

$$x_0' = \beta x_o \tag{14}$$

The PPA with the mapping can be modeled as:

$$f(x,v) = \frac{\varepsilon A(\beta x_o - x)^2 V_c}{2(x_o - x)^2} \tag{15}$$

In this case, the equilibrium condition is:

$$kx = \frac{\varepsilon A(\beta x_o - x)^2 V_c}{2(x_o - x)^2} \tag{16}$$

The condition of stability satisfies:

$$\frac{2x}{(x_o - x)} - \frac{2x}{(\beta x_o - x)} < 1 \tag{17}$$

This also indicates that the error of initial gap distance

will decrease the stable traveling range.

Simulation Verification

A simulation via MATLAB Simulink was performed to verify the analysis. The PPA in this simulation has a spring constant k=158.2 N/m, a damping coefficient $c=7\times10^{-4}$ kg/s, a mass $m=7.5\times10^{-6}$ kg, an initial gap $x_0=10\mu m$, and a surface area $A = 8.41\times10^{-6}\ m^2$.

FIG. 3 shows that the result with the nonlinear term is perfectly canceled. The relationship between control voltage and displacement is linear. The simulation results agree with the mathematical analysis.

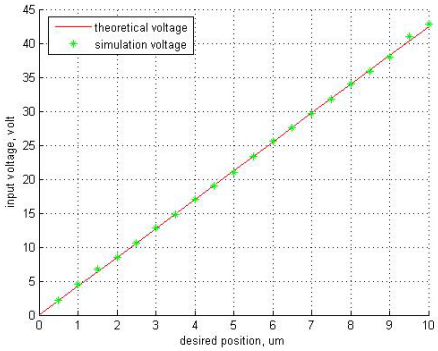


FIG. 3 DESIRED POSITION AND CONTROL VOLTAGE

FIG. 4 and FIG. 5 show that both measurement error and initial gap error could influence the stable range; as well, it is inferred that if the error coefficients equal 1, the PPA would have a full stable range, which matches the ideally situation. The differences between analysis and simulation results are due to the limited number of terms used in the Taylor's series linearization.

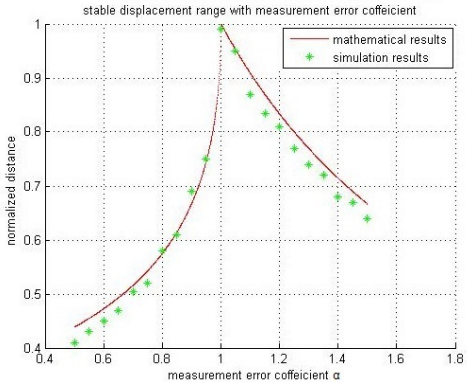


FIG. 4 STABLE RANGE WITH DISPLACEMENT MEASURE ERRORS

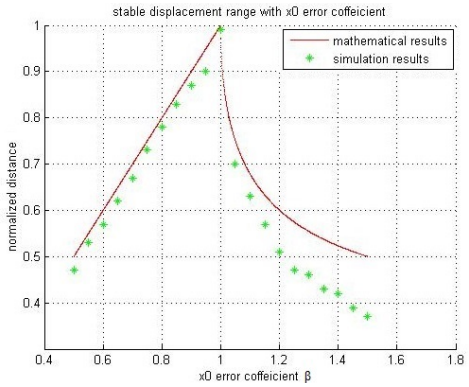


FIG. 5 STABLE RANGE WITH INITIAL GAP ERRORS

Conclusions

Although PPAs are useful in many MEMS applications, they suffer from a limited stable range of motion. This range can be extended using a nonlinear feedback controller to linearize the system. This technique was analysed and verified using MATLAB Simulink.

REFERENCES

M. Kraft, C. Lewis, T. Hesketh and S. Szymkowiah, "A novel micromachined accelerometer capacitive interface," Sensors and Actuators A: Physical, 68 (1998): 466-473.

M. Bakri-Kassem and R. R. Mansour, "high tuning range parallel plate MEMS variable capacitors with arrays of supporting beams," Proc. of the 19th IEEE Int. Conf. on Micro Electro Mechanical Systems, Istanbul, Turkey, 2006: 666-669.

A. Sundaram, M. Maddela, R. Ramadoss and L. M. Feldner, "MEMS-base electronically steerable antenna array fabricated using PCB technology, J. of Microelectro-mechanical Systems, 17(2008), 356-362.

J. Zhang, Z. Zhang, Y. C. Lee, V. M. Bright and J. Neff, "Design and invention of multi-level digitally positioned micromirror for open-loop controlled applications," Sensors and Actuators A: Physical, 103(2003), 271-283.

M. -H. Bao, Handbook of Sensors and Actuators, Vol. 8, Elsevier Science B. V., New York, 2000.

R.N. Dean, C. Wilson, J.P Brunsch and J.Y. Hung, "A synthetic voltage division controller to extend the stable operating range of parallel plate actuators," Proc. of the 2011 IEEE Multi-conference on Systems and Control,

Denver, CO, Sept. 28-30, 2011, 1068-1074.

J. I. Seeger, B. E. Boser, "Charge control of parallel-plate, electrostatic actuators and tip-in instability," J. Microelectromechanical Systems, 12(2003): 656-671.

J. I. Seeger and B. E. Boser, "Negative capacitance for control of gap-closing electrostatic actuators," in Digest Tech. Papers Transducers'03 Conference, Boston, MA, June 8-12, 2003: 484-487.

J. C. Chiou and Y. C. Lin, "A novel capacitance control design of tunable capacitor using multiple electrostatic driving electrodes," Proc. of IEEE-NANO 2001, M3.1 Nanoelectronics and Giga-Scale Systems, Maui, HI, Oct. 28-30, 2001: 319-324.

M. S. –C. Lu and G. K. Fedder, "Position control of parallel-plate microactuators for probe based data storage," J. Microelectromechanical Systems, 13(2004): 759-769.

D. H. S. Maithripala, J. M. Berg, and W. P. Dayawansa, "Nonlinear dynamic output feedback stabilization of electrostatically actuated MEMS," Proc. of 42nd IEEE Conference on Decision and Control, Vol. 1, Maui, HI, Dec. 9-12, 2003: 61-66.

Chu, P.B., Pister, K.S.J., "Analysis of closed-loop control of parallel-plate electrostatic microgrippers," Proceedings of 1994 IEEE International Conference on Robotics and Automation, pp.820-825 vol.1, 8-13 May 1994.

Chong Li received his MS (EE) and BS (EE) from Ocean University of China in 2012 and 2009, respectively. He is currently a PhD student in the ME Dept. at Auburn University, AL, USA. His research interests include MEMS and control.

Robert N. Dean received his PhD (ECE), MS (EE) and BE (EE) from Auburn University in 2006, 1991 and 1998, respectively. He is currently an associate professor in the ECE Dept. at Auburn University, Auburn, AL, USA. He conducts research in MEMS and electronics.

George T. Flowers received his PhD (ME) and MS (ME) from the Georgia Institute of Technology in 1988 and 1985, respectively, and the BME from Auburn University in 1984. He is currently a professor in the Department of Mechanical Engineering at Auburn University. He conducts research in dynamics, vibration, and control.

John Y. Hung is a professor of ECE Dept. at Auburn University, USA, where he has been on the faculty since 1989. His interests are in nonlinear systems and control.

Appendix B

Mathemtica Scripts and Outputs for PPAs' Nonlinear Observability

```
In[76]:= f1 = x2
f2 = (-c * x2 / m) - (k * x1 / m) + e * A * vc * vc / (2 * m * (x0 - x1) ^ 2)
f3 = (-vc (x0 - x1) / (R * e * A) - vc * x2 / (2 * (x0 - x1)) + vs * (x0 - x1) / (R * e * A))
"Lfh1"
t11 = D[f3, x1] * f1
t21 = D[f3, x2] * f2
t31 = D[f3, vc] * f3
"Auto PD of x1"
lx311 = D[t11 + t21 + t31, x1]
"Auto PD of x2"
lx321 = D[t11 + t21 + t31, x2]
"Auto PD of vc"
lx331 = D[t11 + t21 + t31, vc]



"manual"
t1 = ((vc - vs) / (R * e * A) - vc * x2 / (2 * (x0 - x1) ^ 2)) * x2
t2 = - (vc / (2 * (x0 - x1))) *
   ((-c * x2 / m) - (k * x1 / m) + e * A * vc * vc / (2 * m * (x0 - x1) ^ 2))
t3 = (- (x0 - x1) / (R * e * A) - x2 / (2 * (x0 - x1))) *
   (-vc (x0 - x1) / (R * e * A) - vc * x2 / (2 * (x0 - x1)) + vs * (x0 - x1) / (R * e * A))
(*D[t1,x1]
 D[t1+t2,x1]*)
"PD of x1"
lx31 = D[t1 + t2 + t3, x1]
"PD of x2"
lx32 = D[t1 + t2 + t3, x2]
"PD of vc"
lx33 = D[t1 + t2 + t3, vc]
"Auto Simplified PD of x1"
Simplify[lx311]
"Auto Simplified PD of x2"
Simplify[lx321]
"Auto Simplified PD of vc"
Simplify[lx331]

"Simplified PD of x1"
Simplify[lx31]
"Simplified PD of x2"
Simplify[lx32]
"Simplified PD of vc"
Simplify[lx33]

"determinant"
t11 * lx32 - t21 * lx31
"Simplified determinant"
Simplify[t11 * lx32 - t21 * lx31]
x2 = 0
```

t11 * lx32 - t21 * lx31

Out[76]= x2

Out[77]= $\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2} - \frac{k\,x1}{m} - \frac{c\,x2}{m}$

Out[78]= $-\frac{vc\,(x0-x1)}{A\,e\,R} + \frac{vs\,(x0-x1)}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)}$

Out[79]= Lfh1

Out[80]= $x2\left(\frac{vc}{A\,e\,R} - \frac{vs}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)^2}\right)$

Out[81]= $-\frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2} - \frac{k\,x1}{m} - \frac{c\,x2}{m}\right)}{2\,(x0-x1)}$

Out[82]= $\left(-\frac{x0-x1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)}\right)\left(-\frac{vc\,(x0-x1)}{A\,e\,R} + \frac{vs\,(x0-x1)}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)}\right)$

Out[83]= Auto PD of x1

Out[84]= $-\frac{vc\left(-\frac{k}{m} + \frac{A\,e\,vc^2}{m\,(x0-x1)^3}\right)}{2\,(x0-x1)} - \frac{vc\,x2^2}{(x0-x1)^3} - \frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2} - \frac{k\,x1}{m} - \frac{c\,x2}{m}\right)}{2\,(x0-x1)^2} +$
$\left(\frac{vc}{A\,e\,R} - \frac{vs}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)^2}\right)\left(-\frac{x0-x1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)}\right) +$
$\left(\frac{1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)^2}\right)\left(-\frac{vc\,(x0-x1)}{A\,e\,R} + \frac{vs\,(x0-x1)}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)}\right)$

Out[85]= Auto PD of x2

Out[86]= $\frac{vc}{A\,e\,R} - \frac{vs}{A\,e\,R} + \frac{c\,vc}{2\,m\,(x0-x1)} - \frac{vc\,x2}{(x0-x1)^2} -$
$\frac{vc\left(-\frac{x0-x1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)}\right)}{2\,(x0-x1)} - \frac{-\frac{vc\,(x0-x1)}{A\,e\,R} + \frac{vs\,(x0-x1)}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)}}{2\,(x0-x1)}$

Out[87]= Auto PD of vc

Out[88]= $-\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^3} - \frac{\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2} - \frac{k\,x1}{m} - \frac{c\,x2}{m}}{2\,(x0-x1)} + x2\left(\frac{1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)^2}\right) + \left(-\frac{x0-x1}{A\,e\,R} - \frac{x2}{2\,(x0-x1)}\right)^2$

Out[89]= manual

Out[90]= $x2\left(\frac{vc-vs}{A\,e\,R} - \frac{vc\,x2}{2\,(x0-x1)^2}\right)$

Out[91]= $$-\frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}-\frac{c\,x2}{m}\right)}{2\,(x0-x1)}$$

Out[92]= $$\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)\left(-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)}\right)$$

Out[93]= PD of x1

Out[94]= $$-\frac{vc\left(-\frac{k}{m}+\frac{A\,e\,vc^2}{m\,(x0-x1)^3}\right)}{2\,(x0-x1)}-\frac{vc\,x2^2}{(x0-x1)^3}-\frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}-\frac{c\,x2}{m}\right)}{2\,(x0-x1)^2}+$$
$$\left(\frac{vc}{A\,e\,R}-\frac{vs}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)^2}\right)\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)+$$
$$\left(\frac{1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)^2}\right)\left(-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)}\right)$$

Out[95]= PD of x2

Out[96]= $$\frac{vc-vs}{A\,e\,R}+\frac{c\,vc}{2\,m\,(x0-x1)}-\frac{vc\,x2}{(x0-x1)^2}-\frac{vc\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)}{2\,(x0-x1)}-\frac{-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)}}{2\,(x0-x1)}$$

Out[97]= PD of vc

Out[98]= $$-\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^3}-\frac{\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}-\frac{c\,x2}{m}}{2\,(x0-x1)}+$$
$$x2\left(\frac{1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)^2}\right)+\left(-\frac{x0-x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)$$

Out[99]= Auto Simplified PD of x1

Out[100]= $$\left(-3\,A^3\,e^3\,R^2\,vc^3-8\,m\,(vc-vs)\,(x0-x1)^5+\right.$$
$$\left.2\,A^2\,e^2\,R^2\,vc\,(x0-x1)\,(k\,x0\,(x0-x1)-x2\,(c\,(-x0+x1)+m\,x2))\right)\Big/\left(4\,A^2\,e^2\,m\,R^2\,(x0-x1)^4\right)$$

Out[101]= Auto Simplified PD of x2

Out[102]= $$\frac{m\,(4\,vc-3\,vs)\,(x0-x1)^2+A\,e\,R\,vc\,(c\,(x0-x1)-m\,x2)}{2\,A\,e\,m\,R\,(x0-x1)^2}$$

Out[103]= Auto Simplified PD of vc

Out[104]= $$\left(-3\,A^3\,e^3\,R^2\,vc^2+4\,m\,(x0-x1)^5+8\,A\,e\,m\,R\,(x0-x1)^3\,x2+\right.$$
$$\left.A^2\,e^2\,R^2\,(x0-x1)\,(2\,k\,(x0-x1)\,x1-x2\,(-2\,c\,x0+2\,c\,x1+m\,x2))\right)\Big/\left(4\,A^2\,e^2\,m\,R^2\,(x0-x1)^3\right)$$

Out[105]= Simplified PD of x1

Out[106]= $$\left(-3\,A^3\,e^3\,R^2\,vc^3-8\,m\,(vc-vs)\,(x0-x1)^5+\right.$$
$$\left.2\,A^2\,e^2\,R^2\,vc\,(x0-x1)\,(k\,x0\,(x0-x1)-x2\,(c\,(-x0+x1)+m\,x2))\right)\Big/\left(4\,A^2\,e^2\,m\,R^2\,(x0-x1)^4\right)$$

Out[107]= Simplified PD of x2

Out[108]= $$\frac{m\,(4\,vc-3\,vs)\,(x0-x1)^2+A\,e\,R\,vc\,(c\,(x0-x1)-m\,x2)}{2\,A\,e\,m\,R\,(x0-x1)^2}$$

Out[109]= Simplified PD of vc

Out[110]= $$\left(-3\,A^3\,e^3\,R^2\,vc^2+4\,m\,(x0-x1)^5+8\,A\,e\,m\,R\,(x0-x1)^3\,x2+A^2\,e^2\,R^2\,(x0-x1)\,(2\,k\,(x0-x1)\,x1-x2\,(-2\,c\,x0+2\,c\,x1+m\,x2))\right)\Big/\left(4\,A^2\,e^2\,m\,R^2\,(x0-x1)^3\right)$$

Out[111]= determinant

Out[112]= $$x2\left(\frac{vc}{A\,e\,R}-\frac{vs}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)^2}\right)\left(\frac{vc-vs}{A\,e\,R}+\frac{c\,vc}{2\,m\,(x0-x1)}-\frac{vc\,x2}{(x0-x1)^2}-\frac{vc\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)}{2\,(x0-x1)}-\frac{-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)}}{2\,(x0-x1)}\right)+\frac{1}{2\,(x0-x1)}vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}-\frac{c\,x2}{m}\right)\left(-\frac{vc\left(-\frac{k}{m}+\frac{A\,e\,vc^2}{m\,(x0-x1)^3}\right)}{2\,(x0-x1)}-\frac{vc\,x2^2}{(x0-x1)^3}-\frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}-\frac{c\,x2}{m}\right)}{2\,(x0-x1)^2}+\left(\frac{vc}{A\,e\,R}-\frac{vs}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)^2}\right)\left(\frac{-x0+x1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)}\right)+\left(\frac{1}{A\,e\,R}-\frac{x2}{2\,(x0-x1)^2}\right)\left(-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}-\frac{vc\,x2}{2\,(x0-x1)}\right)\right)$$

Out[113]= Simplified determinant

Out[114]= $$\frac{1}{16\,A^2\,e^2\,m^2\,R^2\,(x0-x1)^7}\left(4\,m\,(x0-x1)^3\,x2\left(-m\,(4\,vc-3\,vs)\,(x0-x1)^2+A\,e\,R\,vc\,(c\,(-x0+x1)+m\,x2)\right)\left(2\,vs\,(x0-x1)^2+vc\left(-2\,x0^2+4\,x0\,x1-2\,x1^2+A\,e\,R\,x2\right)\right)-vc\left(A\,e\,vc^2-2\,(x0-x1)^2\,(k\,x1+c\,x2)\right)\left(3\,A^3\,e^3\,R^2\,vc^3+8\,m\,(vc-vs)\,(x0-x1)^5-2\,A^2\,e^2\,R^2\,vc\,(x0-x1)\,(k\,x0\,(x0-x1)-x2\,(c\,(-x0+x1)+m\,x2))\right)\right)$$

Out[115]= 0

Out[116]= $$\frac{1}{2\,(x0-x1)}vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}\right)\left(\frac{-\frac{vc\,(x0-x1)}{A\,e\,R}+\frac{vs\,(x0-x1)}{A\,e\,R}}{A\,e\,R}-\frac{vc\left(-\frac{k}{m}+\frac{A\,e\,vc^2}{m\,(x0-x1)^3}\right)}{2\,(x0-x1)}+\frac{\left(\frac{vc}{A\,e\,R}-\frac{vs}{A\,e\,R}\right)\,(-x0+x1)}{A\,e\,R}-\frac{vc\left(\frac{A\,e\,vc^2}{2\,m\,(x0-x1)^2}-\frac{k\,x1}{m}\right)}{2\,(x0-x1)^2}\right)$$

Appendix C

Source Codes in The Micro Controller for Solenoid Control

main.c

```
#include <stm32f4xx.h>
#include "stdio.h"
#include "stm32f4_discovery.h"
#include "StateMachine.h"


int AD1_AVERG=0;
int AD1_REF=2106;
int AD1_RANGE=941;

int USB_REF_INPUT=0;

int CONTROLLER_NUMBER=-1;
void RCC_Config(void);

int main(void)
{
        RCC_Config();


  while (1)
  {
    MainStateMachine();

    /*
    if(DAC1Buffer!=0)
    {
      DAC1Buffer--;
```

```
                }
    else
    {
      DAC1Buffer=0xfff;
    }

    if(DAC2Buffer!=0)
    {
      DAC2Buffer--;
    }
    else
    {
      DAC2Buffer=0xfff;
    }
                */
 }
}



void RCC_Config(void)
{
        RCC_DeInit();           //
        RCC_HSEConfig(RCC_HSE_ON);  //
        if (RCC_WaitForHSEStartUp() == SUCCESS) //
 {
                RCC_PLLCmd(DISABLE);              //
                RCC_SYSCLKConfig(RCC_SYSCLKSource_PLLCLK); //
                RCC_HCLKConfig(RCC_SYSCLK_Div1);    //
```

```
                    RCC_PCLK1Config(RCC_HCLK_Div4);    //
                    RCC_PCLK2Config(RCC_HCLK_Div2);    //
                    RCC_PLLConfig(RCC_PLLSource_HSE, 8, 336, 2, 7);
//system_stm43f4xx.c?Line149
                    RCC_PLLCmd(ENABLE);      //
                    while (RCC_GetFlagStatus(RCC_FLAG_PLLRDY) == RESET) { } //
          }
   #if (__FPU_PRESENT == 1) && (__FPU_USED == 1)
  SCB->CPACR |= ((3UL << 10*2)|(3UL << 11*2));
 #endif
}
#ifdef  USE_FULL_ASSERT

/**
  * @brief  Reports the name of the source file and the source line number
  *        where the assert_param error has occurred.
  * @param  file: pointer to the source file name
  * @param  line: assert_param error line source number
  * @retval None
  */
void assert_failed(uint8_t* file, uint32_t line)
{
  /* User can add his own implementation to report the file name and line number,
    ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */

  /* Infinite loop */
  while (1)
  {
  }
```

```
}
#endif
AdConfig.c
#include <stm32f4xx.h>
#define ADC3_DR_ADDRESS     ((uint32_t)0x4001224C)

uint16_t ADC3ConvertedValue[20];

//__IO uint32_t ADC3ConvertedVoltage = 0;
void ADCInit(void);
void ADCInit()
{
          ADC_InitTypeDef       ADC_InitStructure;
 ADC_CommonInitTypeDef ADC_CommonInitStructure;
 DMA_InitTypeDef       DMA_InitStructure;
 GPIO_InitTypeDef      GPIO_InitStructure;

 /* Enable ADC3, DMA2 and GPIO clocks ****************************************/
 RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_DMA2 | RCC_AHB1Periph_GPIOC, ENABLE);
 RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC3, ENABLE);

 /* DMA2 Stream0 channel0 configuration **************************************/
 DMA_InitStructure.DMA_Channel = DMA_Channel_2;
 DMA_InitStructure.DMA_PeripheralBaseAddr = (uint32_t)ADC3_DR_ADDRESS;
 DMA_InitStructure.DMA_Memory0BaseAddr = (uint32_t)&ADC3ConvertedValue[0];
 DMA_InitStructure.DMA_DIR = DMA_DIR_PeripheralToMemory;
 DMA_InitStructure.DMA_BufferSize = 20;
 DMA_InitStructure.DMA_PeripheralInc = DMA_PeripheralInc_Disable;
 DMA_InitStructure.DMA_MemoryInc = DMA_MemoryInc_Enable;
```

```
DMA_InitStructure.DMA_PeripheralDataSize = DMA_PeripheralDataSize_HalfWord;
DMA_InitStructure.DMA_MemoryDataSize = DMA_MemoryDataSize_HalfWord;
DMA_InitStructure.DMA_Mode = DMA_Mode_Circular;
DMA_InitStructure.DMA_Priority = DMA_Priority_High;
DMA_InitStructure.DMA_FIFOMode = DMA_FIFOMode_Disable;
DMA_InitStructure.DMA_FIFOThreshold = DMA_FIFOThreshold_HalfFull;
DMA_InitStructure.DMA_MemoryBurst = DMA_MemoryBurst_Single;
DMA_InitStructure.DMA_PeripheralBurst = DMA_PeripheralBurst_Single;
DMA_Init(DMA2_Stream0, &DMA_InitStructure);
DMA_Cmd(DMA2_Stream0, ENABLE);

/* Configure ADC3 Channel12 pin as analog input ******************************/
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_2 | GPIO_Pin_3;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL ;
GPIO_Init(GPIOC, &GPIO_InitStructure);

/* ADC Common Init **********************************************************/
ADC_CommonInitStructure.ADC_Mode = ADC_Mode_Independent;
ADC_CommonInitStructure.ADC_Prescaler = ADC_Prescaler_Div2;
ADC_CommonInitStructure.ADC_DMAAccessMode = ADC_DMAAccessMode_1;
ADC_CommonInitStructure.ADC_TwoSamplingDelay = ADC_TwoSamplingDelay_20Cycles;
ADC_CommonInit(&ADC_CommonInitStructure);

/* ADC3 Init ****************************************************************/
ADC_InitStructure.ADC_Resolution = ADC_Resolution_12b;
ADC_InitStructure.ADC_ScanConvMode = ENABLE;
ADC_InitStructure.ADC_ContinuousConvMode = ENABLE;
ADC_InitStructure.ADC_ExternalTrigConvEdge = ADC_ExternalTrigConvEdge_None;
```

```
 ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
 ADC_InitStructure.ADC_NbrOfConversion = 2;
 ADC_Init(ADC3, &ADC_InitStructure);

 /* ADC3 regular channel12 configuration *************************************/
 ADC_RegularChannelConfig(ADC3, ADC_Channel_12, 1, ADC_SampleTime_3Cycles);//pc2
        ADC_RegularChannelConfig(ADC3, ADC_Channel_13, 2,
ADC_SampleTime_3Cycles);//pc3

 /* Enable DMA request after last transfer (Single-ADC mode) */
 ADC_DMARequestAfterLastTransferCmd(ADC3, ENABLE);

 /* Enable ADC3 DMA */
 ADC_DMACmd(ADC3, ENABLE);

 /* Enable ADC3 */
 ADC_Cmd(ADC3, ENABLE);
        ADC_SoftwareStartConv(ADC3);
}
```

DAConfig.c

```
#include "stm32f4_discovery.h"

/** @addtogroup STM32F4_Discovery_Peripheral_Examples
  * @{
  */

/** @addtogroup DAC_SignalsGeneration
  * @{
```

```
 */

/* Private typedef -----------------------------------------------------------*/
/* Private define ------------------------------------------------------------*/
#define DAC_DHR12R2_ADDRESS    0x40007414
#define DAC_DHR8R1_ADDRESS     0x40007408

/* Private macro -------------------------------------------------------------*/
/* Private variables ---------------------------------------------------------*/
DAC_InitTypeDef  DAC_InitStructure;

uint16_t DAC1Buffer = 0;
uint16_t DAC2Buffer = 0;



/* Private function prototypes -----------------------------------------------*/
void TIM6_Config(void);

void DAC2Config(void);
void DAC1Config(void);

/* Private functions ---------------------------------------------------------*/

/**
 * @brief   Main program
 * @param  None
 * @retval None
 */
```

```
void DAConfig(void)
{
  /*!< At this stage the microcontroller clock setting is already configured,
     this is done through SystemInit() function which is called from startup
     file (startup_stm32f4xx.s) before to branch to application main.
     To reconfigure the default setting of SystemInit() function, refer to
     system_stm32f4xx.c file
   */

  /* Preconfiguration before using DAC----------------------------------------*/
  GPIO_InitTypeDef GPIO_InitStructure;

  /* DMA1 clock and GPIOA clock enable (to be used with DAC) */
  RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_DMA1, ENABLE);
  RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);
  /* DAC Periph clock enable */
  RCC_APB1PeriphClockCmd(RCC_APB1Periph_DAC, ENABLE);

  /* DAC channel 1 & 2 (DAC_OUT1 = PA.4)(DAC_OUT2 = PA.5) configuration */
  GPIO_InitStructure.GPIO_Pin = GPIO_Pin_4 | GPIO_Pin_5;
  GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AN;
  GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
  GPIO_Init(GPIOA, &GPIO_InitStructure);

  /* TIM6 Configuration -------------------------------------------------------*/
  TIM6_Config();
  DAC1Config();
  DAC2Config();
```

```
}

/**
  * @brief  TIM6 Configuration
  * @note   TIM6 configuration is based on CPU @168MHz and APB1 @42MHz
  * @note   TIM6 Update event occurs each 37.5MHz/256 = 16.406 KHz
  * @param  None
  * @retval None
  */
void TIM6_Config(void)
{
  TIM_TimeBaseInitTypeDef    TIM_TimeBaseStructure;
  /* TIM6 Periph clock enable */
  RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM6, ENABLE);

  /* Time base configuration */
  TIM_TimeBaseStructInit(&TIM_TimeBaseStructure);
  TIM_TimeBaseStructure.TIM_Period = 0xFF;
  TIM_TimeBaseStructure.TIM_Prescaler = 0;
  TIM_TimeBaseStructure.TIM_ClockDivision = 0;
  TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;
  TIM_TimeBaseInit(TIM6, &TIM_TimeBaseStructure);

  /* TIM6 TRGO selection */
  TIM_SelectOutputTrigger(TIM6, TIM_TRGOSource_Update);

  /* TIM6 enable counter */
```

```
 TIM_Cmd(TIM6, ENABLE);
}

/**
 * @brief  DAC  Channel2 SineWave Configuration
 * @param  None
 * @retval None
 */
void DAC2Config(void)
{
 DMA_InitTypeDef DMA_InitStructure;

 /* DAC channel2 Configuration */
 DAC_InitStructure.DAC_Trigger = DAC_Trigger_T6_TRGO;
 DAC_InitStructure.DAC_WaveGeneration = DAC_WaveGeneration_None;
 DAC_InitStructure.DAC_OutputBuffer = DAC_OutputBuffer_Enable;
 DAC_Init(DAC_Channel_2, &DAC_InitStructure);

 /* DMA1_Stream6 channel7 configuration **************************************/
 DMA_DeInit(DMA1_Stream6);
 DMA_InitStructure.DMA_Channel = DMA_Channel_7;
 DMA_InitStructure.DMA_PeripheralBaseAddr = (uint32_t)DAC_DHR12R2_ADDRESS;
 DMA_InitStructure.DMA_Memory0BaseAddr = (uint32_t)&DAC2Buffer;
 DMA_InitStructure.DMA_DIR = DMA_DIR_MemoryToPeripheral;
 DMA_InitStructure.DMA_BufferSize = 1;
 DMA_InitStructure.DMA_PeripheralInc = DMA_PeripheralInc_Disable;
 DMA_InitStructure.DMA_MemoryInc = DMA_MemoryInc_Enable;
 DMA_InitStructure.DMA_PeripheralDataSize = DMA_PeripheralDataSize_HalfWord;
 DMA_InitStructure.DMA_MemoryDataSize = DMA_MemoryDataSize_HalfWord;
```

```
  DMA_InitStructure.DMA_Mode = DMA_Mode_Circular;
  DMA_InitStructure.DMA_Priority = DMA_Priority_High;
  DMA_InitStructure.DMA_FIFOMode = DMA_FIFOMode_Disable;
  DMA_InitStructure.DMA_FIFOThreshold = DMA_FIFOThreshold_HalfFull;
  DMA_InitStructure.DMA_MemoryBurst = DMA_MemoryBurst_Single;
  DMA_InitStructure.DMA_PeripheralBurst = DMA_PeripheralBurst_Single;
  DMA_Init(DMA1_Stream6, &DMA_InitStructure);

  /* Enable DMA1_Stream6 */
  DMA_Cmd(DMA1_Stream6, ENABLE);

  /* Enable DAC Channel2 */
  DAC_Cmd(DAC_Channel_2, ENABLE);

  /* Enable DMA for DAC Channel2 */
  DAC_DMACmd(DAC_Channel_2, ENABLE);
}

void DAC1Config(void)
{
  DMA_InitTypeDef DMA_InitStructure;

  /* DAC channel1 Configuration */
  DAC_InitStructure.DAC_Trigger = DAC_Trigger_T6_TRGO;
  DAC_InitStructure.DAC_WaveGeneration = DAC_WaveGeneration_None;
  DAC_InitStructure.DAC_OutputBuffer = DAC_OutputBuffer_Enable;
  DAC_Init(DAC_Channel_1, &DAC_InitStructure);

  /* DMA1_Stream5 channel7 configuration **************************************/
```

```
  DMA_DeInit(DMA1_Stream5);
  DMA_InitStructure.DMA_Channel = DMA_Channel_7;
  DMA_InitStructure.DMA_PeripheralBaseAddr = DAC_DHR8R1_ADDRESS;
  DMA_InitStructure.DMA_Memory0BaseAddr = (uint32_t)&DAC1Buffer;
  DMA_InitStructure.DMA_BufferSize = 1;
  DMA_InitStructure.DMA_PeripheralDataSize = DMA_PeripheralDataSize_HalfWord;
  DMA_InitStructure.DMA_MemoryDataSize = DMA_MemoryDataSize_HalfWord;
  DMA_InitStructure.DMA_DIR = DMA_DIR_MemoryToPeripheral;
  DMA_InitStructure.DMA_PeripheralInc = DMA_PeripheralInc_Disable;
  DMA_InitStructure.DMA_MemoryInc = DMA_MemoryInc_Enable;
  DMA_InitStructure.DMA_Mode = DMA_Mode_Circular;
  DMA_InitStructure.DMA_Priority = DMA_Priority_High;
  DMA_InitStructure.DMA_FIFOMode = DMA_FIFOMode_Disable;
  DMA_InitStructure.DMA_FIFOThreshold = DMA_FIFOThreshold_HalfFull;
  DMA_InitStructure.DMA_MemoryBurst = DMA_MemoryBurst_Single;
  DMA_InitStructure.DMA_PeripheralBurst = DMA_PeripheralBurst_Single;
  DMA_Init(DMA1_Stream5, &DMA_InitStructure);

  /* Enable DMA1_Stream5 */
  DMA_Cmd(DMA1_Stream5, ENABLE);

  /* Enable DAC Channel1 */
  DAC_Cmd(DAC_Channel_1, ENABLE);

  /* Enable DMA for DAC Channel1 */
  DAC_DMACmd(DAC_Channel_1, ENABLE);
}
```

```
#ifdef  USE_FULL_ASSERT

/**
  * @brief  Reports the name of the source file and the source line number
  *         where the assert_param error has occurred.
  * @param  file: pointer to the source file name
  * @param  line: assert_param error line source number
  * @retval None
  */
void assert_failed(uint8_t* file, uint32_t line)
{
  /* User can add his own implementation to report the file name and line number,
     ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */

  /* Infinite loop */
  while (1)
  {
  }
}
#endif
```

StateMachine.c

```
#include <stm32f4xx.h>
#include "StateMachine.h"
#include "ADConfig.h"
#include "DA_config.h"
#include "TimerConfig.h"
#include "usbd_cdc_vcp.h"
#include "stm32f4_discovery.h"
```

```
extern uint16_t VCP_DataTx   (uint8_t* Buf, uint32_t Len);
extern int USBInit(void);
static short int state=STATE_INIT;

int SetValue=0;

void MainStateMachine()
{
   switch(state)
   {
      case STATE_INIT:

         USBInit();
         DAConfig();
         TimerInit();
         ADCInit();

         state=STATE_RUN;
         //VCP_DataTx ("System initializtion complete",29);
      break;

      case STATE_RUN:
         //DAC1Buffer=ADC3ConvertedValue[0];
         //DAC2Buffer=ADC3ConvertedValue[1];

      default:
         ;
   }
```

```
}

void ChangeState(short int newValue)
{
   state=newValue;
}

short int GetState(void)
{
   return state;
}
```

ADRC.c

```
#include <math.h>
#include "stm32f4_discovery.h"



extern double TOTAL_DIST;

extern double TOTAL_DIST_CORRECTION;

extern double AD_DIF_RANGE;

extern int SHAPED_INPUT;
extern double GLB_DUAL_DIS;
extern unsigned char SendBuffer1[4];
extern unsigned char SendBuffer2[4];



extern double tempDistX;
```

```
extern int USB_REF_INPUT;

extern uint16_t DAC1Buffer;


extern void FillSendBuffer1(double input);
extern void FillSendBuffer2(double input);
extern void FillSendBuffer3(double input);
extern void FillSendBuffer4(double input);

void LinearESO(double e, double u);
void LinearESOTemp(double e, double u);
void ESO(double e, double u);

double fal(double e,double a,double delta);
int sign(double input);
void ADRC();
double z1=0;
double z2=0;
double z3=0;

double z1t=0;
double z2t=0;
double z3t=0;
void ADRC()
{
   double distX=0;
   double tragetDist=0;
```

```
double errorSum=0;
double adrcInput=0;
double controlEffort=0;

static double PIDEffort=0;
double PIDerror=0;
static double oldError=0;
tragetDist=(double)(USB_REF_INPUT)/10000;



TOTAL_DIST+=TOTAL_DIST_CORRECTION;
distX=GLB_DUAL_DIS;

if(distX<0)
{
   distX=0;
}
LinearESO(GLB_DUAL_DIS-z1,DAC1Buffer);
LinearESOTemp(GLB_DUAL_DIS-z1t,DAC1Buffer);
//ESO(distX-tragetDist,DAC1Buffer/10);
errorSum=-(distX-tragetDist)-z2;
PIDerror=tragetDist-distX;
//errorSum=-z1-z2;

if(errorSum<0)
{
   //adrcInput=-errorSum;
   adrcInput=pow(-errorSum,0.9)-z3;
}
```

```
PIDEffort+=1000*(PIDerror-oldError)+PIDerror*500;
if(PIDEffort<0)
{
   PIDEffort=0;
}
if(PIDEffort>120)
{
   PIDEffort=120;
}
controlEffort=sqrt((double)(PIDEffort))*(TOTAL_DIST-distX+5.3e-3*0.6667);

controlEffort*=50000;
controlEffort+=(-z2)*1600;
if(controlEffort>4095)
{
   controlEffort=4095;
}
if(controlEffort<0)
{
   controlEffort=0;
}
//FillSendBuffer2(z1);
//FillSendBuffer3(z2);
//FillSendBuffer4(z3);

FillSendBuffer2(z2/100);
FillSendBuffer3(z2t/100);
FillSendBuffer4(z3);
DAC1Buffer=controlEffort;
```

```
}
void GainSchedulingController()
{
   double distX=0;
   double controlEffort=0;

   TOTAL_DIST+=TOTAL_DIST_CORRECTION;
   distX=GLB_DUAL_DIS;

   if(distX<0)
   {
      distX=0;
   }
   LinearESO(GLB_DUAL_DIS-z1,DAC1Buffer);

   //errorSum=-z1-z2;

   controlEffort=sqrt((double)(USB_REF_INPUT))*(TOTAL_DIST-distX+5.3e-3*0.6667);

   controlEffort*=50000;
   ////controlEffort+=(-z2)*800;
   if(controlEffort>4095)
   {
      controlEffort=4095;
   }
   if(controlEffort<0)
   {
      controlEffort=0;
```

```
    }
    //FillSendBuffer2(z1);
    //FillSendBuffer3(z2);
    //FillSendBuffer4(z3);

    FillSendBuffer2(z2/100);
    FillSendBuffer3(z2t/100);
    FillSendBuffer4(z3);
    DAC1Buffer=controlEffort;
}
void GainSchedulingController_ESO()
{
    double distX=0;
    double controlEffort=0;

    TOTAL_DIST+=TOTAL_DIST_CORRECTION;
    distX=GLB_DUAL_DIS;

    if(distX<0)
    {
        distX=0;
    }
    LinearESO(GLB_DUAL_DIS-z1,DAC1Buffer);

    //errorSum=-z1-z2;

    controlEffort=sqrt((double)(USB_REF_INPUT))*(TOTAL_DIST-distX+5.3e-3*0.6667);

    controlEffort*=50000;
```

```
    controlEffort+=(-z2)*800;
    if(controlEffort>4095)
    {
      controlEffort=4095;
    }
    if(controlEffort<0)
    {
      controlEffort=0;
    }
    //FillSendBuffer2(z1);
    //FillSendBuffer3(z2);
    //FillSendBuffer4(z3);

    FillSendBuffer2(z2/100);
    FillSendBuffer3(z2t/100);
    FillSendBuffer4(z3);
    DAC1Buffer=controlEffort;
}
void GainSchedulingPIDController()
{
    double distX=0;
    double tragetDist=0;

    double errorSum=0;
    double adrcInput=0;
    double controlEffort=0;

    static double PIDEffort=0;
    double PIDerror=0;
```

```
static double oldError=0;
tragetDist=(double)(USB_REF_INPUT)/10000;


TOTAL_DIST+=TOTAL_DIST_CORRECTION;
distX=GLB_DUAL_DIS;

if(distX<0)
{
   distX=0;
}
LinearESO(GLB_DUAL_DIS-z1,DAC1Buffer);
LinearESOTemp(GLB_DUAL_DIS,DAC1Buffer);
//ESO(distX-tragetDist,DAC1Buffer/10);
errorSum=-(distX-tragetDist)-z2;
PIDerror=tragetDist-distX;
//errorSum=-z1-z2;

if(errorSum<0)
{
   //adrcInput=-errorSum;
   adrcInput=pow(-errorSum,0.9)-z3;
}
PIDEffort+=1000*(PIDerror-oldError)+PIDerror*500;
if(PIDEffort<0)
{
   PIDEffort=0;
}
if(PIDEffort>120)
```

```
    {
      PIDEffort=120;
    }
    controlEffort=sqrt((double)(PIDEffort))*(TOTAL_DIST-distX+5.3e-3*0.6667);

    controlEffort*=50000;
    //controlEffort+=(-z2)*800;
    if(controlEffort>4095)
    {
      controlEffort=4095;
    }
    if(controlEffort<0)
    {
      controlEffort=0;
    }
    //FillSendBuffer2(z1);
    //FillSendBuffer3(z2);
    //FillSendBuffer4(z3);

    FillSendBuffer2(z2/100);
    FillSendBuffer3(z2t/100);
    FillSendBuffer4(z3);
    DAC1Buffer=controlEffort;
}
void LinearESO(double e, double u)
{
      /*const double h=5e-3;
    const double b01=0.7786;
    const double b02=81.3185;
```

```
   const double b03=2.4666e+003;
   const double b=1e-5;
*/

        const double h=5e-3;
   const double b01=3;
   const double b02=600;
   const double b03=40000;
   const double b=1e-7;
   double scale=0;

   scale=6.96e-3/AD_DIF_RANGE;
   e=((int)(e/scale))*scale;
   z1=z1+h*z2+b01*e;
   if(z1>7e-3)
   {
      z1=7e-3;
   }
   if(z1<-3e-3)
   {
      z1=-3e-3;
   }
   z2=z2+h*z3+b02*e+b*u;
   if(z2>1)
   {
      z2=1;
   }
   if(z2<-1)
   {
```

```
      z1=-1;
   }
   z3=z3+b03*e;

}

void LinearESOTemp(double e, double u)
{
   const double h=5e-3;
   const double b01=3;
   const double b02=600;
   const double b03=40000;
   const double b=1e-7;
   double scale=0;

   //scale=6.96e-3/AD_DIF_RANGE;
   //e=((int)(e/scale))*scale;
   z1t=z1t+h*z2t+b01*e;
   if(z1t>7e-3)
   {
      z1t=7e-3;
   }
   if(z1t<-3e-3)
   {
      z1t=-3e-3;
   }
   z2t=z2t+h*z3t+b02*e+b*u;
   if(z2t>1)
   {
```

```
        z2t=1;
    }
    if(z2t<-1)
    {
        z1t=-1;
    }
    z3t=z3t+b03*e;

}
void ESO(double e, double u)
{
    const double h=5e-3;
    const double b01=1;
    const double b02=1/(2*pow(h,0.5));
    const double b03=2/(25*pow(h,1.2));

    double fe=0;
    double fe1=0;

    fe=fal(e,0.5,1);
    fe1=fal(e,0.25,1);

    z1=z1+h*z2-b01*e;
    z2=z2+h*(z3+u)-b02*fe;
    z3=z3-b03*fe1;
}
double fal(double e,double a,double delta)
{
    double funResult=0;
```

```
    //if(abs(e)<=delta)
    if(e<=delta && e>=(-delta))
    {
        funResult=e/(pow(delta,1-a));
    }
    else
    {
        funResult=pow(abs(e),a)*sign(e);
    }
    return funResult;
}

int sign(double input)
{
    if(input<-1e-5)
    {
        return -1;
    }
    else if(input>=-1e-5&&input<=1e-5)
    {
        return 0;
    }
    else
    {
        return 1;
    }
}
```

Appendix D

Source Codes in The Personal Computer for Solenoid Control

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.IO;
using System.IO.Ports;
using System.Threading;
using System.Drawing.Imaging;

namespace SolenoidControl
{
  public partial class Form1 : Form
  {
    private SerialPort scom;
    private bool uartOpen = false;
    private string toShow = "";

    private Graphics mapGraphics;
    private Bitmap bufMap;

    private Color backColor = Color.White;
    private Color broderColor = Color.DeepSkyBlue;

    private Color nowColor = Color.Red;
    private Color targetColor = Color.DarkSeaGreen;
```

```
private Color nowColor1 = Color.Blue;
private Color targetColor1 = Color.CadetBlue;

private Color nowColor2 = Color.DarkOrange;
private Color targetColor2 = Color.DeepPink;

private Color nowColor3 = Color.MediumPurple;
private Color targetColor3 = Color.Black;



private int[] sampleData;
private int sampleDataLen=0;

private int[] sampleData1;
private int sampleDataLen1 = 0;

private int[] sampleData2;
private int sampleDataLen2 = 0;

private int[] sampleData3;
private int sampleDataLen3 = 0;

private const UInt32 maxLen = 1000;
private const int stepLen = 1;



private int initialPos = 2080;
private int totalADDisplacement = 960;
```

```
private double totalDisplacement = 6.96e-3;
public Form1()
{
   InitializeComponent();
   bufMap = new Bitmap(1000, 500);
   mapGraphics = Graphics.FromImage(bufMap);//初始化
   ResetMap();
   sampleData = new int[maxLen];
   sampleData1 = new int[maxLen];
   sampleData2 = new int[maxLen];
   sampleData3 = new int[maxLen];
}

 ~Form1()
{
   if ((uartOpen = true) && (scom != null))
   {
      try
      {
         scom.Close();
      }
      catch (Exception ee)
      {
         MessageBox.Show(ee.Message.ToString());
      }
   }
}
private void Form1_Load(object sender, EventArgs e)
{
```

```
    string[] uartNames = getUARTNames();
    uartComboBox.Items.Clear();
    uartComboBox.Items.Add("Select a port");
    uartComboBox.SelectedIndex = 0;

    controllerComboBox.SelectedIndex = 0;
    if (uartNames.Length > 0)
    {
        for (int i = 0; i < uartNames.Length; i++)
        {
            uartComboBox.Items.Add(uartNames[i]);
        }
    }
    else
    {
        MessageBox.Show("No serial port found in this computer");
    }
}

private string[] getUARTNames()
{
    string[] names = null;
    try
    {
        names = SerialPort.GetPortNames();
    }
    catch (Exception ex)
    {
```

```
            MessageBox.Show("Serial Ports can not acquired. Detailed information:" +
ex.Message.ToString());


        }
        return names;
    }


    private void uartComboBox_SelectedIndexChanged(object sender, EventArgs e)
    {
        if(uartComboBox.SelectedIndex!=0)
        {
            try
            {
                scom = new SerialPort(uartComboBox.Text.ToString(), //port name
                                                    9600, //buad rate
                                                Parity.None, //no check point
                                                    8, //8 data bits
                                                StopBits.One);//1 stop bit
                scom.RtsEnable = false;
                scom.ReadTimeout = 3;
                if (!scom.IsOpen)
                {
                    scom.Open();
                    uartOpen = true;
                    //scom.DataReceived+=new SerialDataReceivedEventHandler(scomDataReceived);
                    scom.DataReceived += new SerialDataReceivedEventHandler(scom_DataReceived);
                    MessageBox.Show(uartComboBox.Text.ToString() + "open successfully");
                }
            }
```

```
        catch (Exception ex)
        {
            uartOpen = false;
            MessageBox.Show("serial port error:" + ex.Message.ToString());
        }
    }
}

private void scom_DataReceived(object sender, SerialDataReceivedEventArgs e)
{
    //throw new Exception("The method or operation is not implemented.");

    //int bufLength = scom.BytesToRead;

    Thread.Sleep(50);
    byte[] scomBuf = new byte[scom.BytesToRead];
    scom.Read(scomBuf, 0, scomBuf.Length);
    //string bufstring = scom.ReadLine();
    //MessageBox.Show(Convert.ToString(bufLength)+" received");

    toShow="";
    //receving and analyzing the tranmitted data
    if(scomBuf.Length%4==0 && uartOpen==true)
    {
        for (int i = 0; i < scomBuf.Length;)
        {
            toShow += scomBuf[i].ToString();

            if (scomBuf[i]=='A')
```

```
            {
               if ((scomBuf[i + 1] & 0x80 )== 0x00)
               {
                  sampleData[sampleDataLen] = scomBuf[i + 1] * 100 + scomBuf[i + 2] * 10 +
scomBuf[i + 2];
               }
               else
               {
                  sampleData[sampleDataLen] = ( scomBuf[i + 1] & 0x7f)* 100 + scomBuf[i + 2] *
10 + scomBuf[i + 2];
                  sampleData[sampleDataLen] *= -1;
               }
               if (sampleDataLen < maxLen - 1)
               {
                  sampleDataLen++;
               }
               else
               {
                  sampleDataLen = 0;
               }
            }
            else if (scomBuf[i] == 'B')
            {
               if ((scomBuf[i + 1] & 0x80) == 0x00)
               {
                  sampleData1[sampleDataLen1] = scomBuf[i + 1] * 100 + scomBuf[i + 2] * 10 +
scomBuf[i + 2];
               }
               else
               {
```

```
                    sampleData1[sampleDataLen1] = (scomBuf[i + 1] & 0x7f) * 100 + scomBuf[i + 2]
* 10 + scomBuf[i + 2];
                    sampleData1[sampleDataLen1] *= -1;
                }
                if (sampleDataLen1 < maxLen - 1)
                {
                    sampleDataLen1++;
                }
                else
                {
                    sampleDataLen1 = 0;
                }
            }
            else if (scomBuf[i] == 'C')
            {
                if ((scomBuf[i + 1] & 0x80) == 0x00)
                {
                    sampleData2[sampleDataLen2] = scomBuf[i + 1] * 100 + scomBuf[i + 2] * 10 +
scomBuf[i + 2];
                }
                else
                {
                    sampleData2[sampleDataLen2] = (scomBuf[i + 1] & 0x7f) * 100 + scomBuf[i + 2]
* 10 + scomBuf[i + 2];
                    sampleData2[sampleDataLen2] *= -1;
                }
                if (sampleDataLen2 < maxLen - 1)
                {
                    sampleDataLen2++;
                }
```

```
                else
                {
                    sampleDataLen2 = 0;
                }
            }
            else if (scomBuf[i] == 'D')
            {
                if ((scomBuf[i + 1] & 0x80) == 0x00)
                {
                    sampleData3[sampleDataLen3] = scomBuf[i + 1] * 100 + scomBuf[i + 2] * 10 +
scomBuf[i + 2];
                }
                else
                {
                    sampleData3[sampleDataLen3] = (scomBuf[i + 1] & 0x7f) * 100 + scomBuf[i + 2]
* 10 + scomBuf[i + 2];
                    sampleData3[sampleDataLen3] *= -1;
                }
                if (sampleDataLen3 < maxLen - 1)
                {
                    sampleDataLen3++;
                }
                else
                {
                    sampleDataLen3 = 0;
                }
            }
```

```
                i += 4;


            }
        }


        //toShow = scomBuf.ToString();


        this.Invoke(new EventHandler(updateTxt));
        this.Invoke(new EventHandler(UpdateMap));


    }
    private void updateTxt(object sender, EventArgs e)
    {
        double disMM = 0;
        if (textBox1.Text.Length > 500)
        {
            textBox1.Text = "";
        }
        textBox1.Text += toShow + "\r\n";
        //disMM = 1000 * totalDisplacement * ((double)(initialPos - 40*sampleData[(0 * stepLen
+ sampleDataLen)]) / totalADDisplacement);
        disMM = ((double)sampleData[(0 * stepLen + sampleDataLen)]) / 100;
        distLabel.Text = "Displacement:" +disMM.ToString("0.0000") +"mm";
    }
    private void button1_Click(object sender, EventArgs e)
    {
        string setPointString = setPointTextBox.Text.ToString();
        Boolean legalText = false;
        try
```

```
    {
        double.Parse(setPointString);
        legalText = true;
    }
    catch (Exception ee)
    {
        MessageBox.Show("error :"+ee.Message.ToString());
    }

    if (legalText = true && scom != null)
    {
        try
        {
            scom.WriteLine("set=" + setPointString + "end");
        }
        catch (Exception ee)
        {
            MessageBox.Show("error :" + ee.Message.ToString());
        }
    }
}

private void Form1_FormClosed(object sender, FormClosedEventArgs e)
{

}

public void ResetMap()
{
```

```
        int i = 0;
        mapGraphics.DrawRectangle(new Pen(broderColor), 0, 0, 999, 499);//外部边框设为海蓝色
        mapGraphics.FillRectangle(new SolidBrush(backColor), 1, 1, 998, 498);//内部填充颜色设为白色


        while (i < 1000)
        {
           mapGraphics.DrawLine(new Pen(broderColor), i, 124, i + 3, 124);
           mapGraphics.DrawLine(new Pen(broderColor), i, 249, i + 3, 249);
           mapGraphics.DrawLine(new Pen(broderColor), i, 249+125, i + 3, 249+125);
           //mapGraphics.DrawLine(new Pen(broderColor), 249, i, 249, i + 3);
           i += 10;
        }
        i = 0;
        while (i < 500)
        {
           //mapGraphics.DrawLine(new Pen(broderColor), i, 249, i + 3, 249);
           mapGraphics.DrawLine(new Pen(broderColor), 249, i, 249, i + 3);
           mapGraphics.DrawLine(new Pen(broderColor), 499, i, 499, i + 3);
           mapGraphics.DrawLine(new Pen(broderColor), 749, i, 749, i + 3);
           i += 10;
        }
        //oldX = 0;
        //oldY = 0;
        //nowX = 0;
        //nowY = 0;
        pictureBoxMap.Image = bufMap;
    }
```

```
public void UpdateMap(object sender, EventArgs e)
{
    ResetMap();
    //oldX = 0;
    //oldY = 0;
    if (checkBoxA.Checked == true)
    {
        mapGraphics.FillEllipse(new SolidBrush(nowColor), 0, 250 - (float)0.3 * sampleData[(0 *
stepLen + sampleDataLen) % maxLen], 3, 3);
        for (int i = 1; i < 1000; i++)
        {
            //mapGraphics.FillEllipse(new SolidBrush(nowColor), (i+sampleDataLen)%1000, 250 -
(float)sampleData[i], 3, 3);
            mapGraphics.FillEllipse(new SolidBrush(nowColor), i, 250 - (float)0.3 * sampleData[(i
* stepLen + sampleDataLen) % maxLen], 3, 3);


            mapGraphics.DrawLine(new Pen(targetColor), i, 250 - (float)0.3 * sampleData[(i *
stepLen + sampleDataLen) % maxLen], i - 1, 250 - (float)0.3 * sampleData[(i * stepLen +
sampleDataLen - 1) % maxLen]);


        }
    }
    if (checkBoxB.Checked == true)
    {
        mapGraphics.FillEllipse(new SolidBrush(nowColor1), 0, 375 - (float)0.3 *
sampleData1[(0 * stepLen + sampleDataLen1) % maxLen], 3, 3);
        for (int i = 1; i < 1000; i++)
        {
            //mapGraphics.FillEllipse(new SolidBrush(nowColor), (i+sampleDataLen)%1000, 375 -
(float)sampleData[i], 3, 3);
            mapGraphics.FillEllipse(new SolidBrush(nowColor1), i, 375 - (float)0.3 *
sampleData1[(i * stepLen + sampleDataLen1) % maxLen], 3, 3);
```

```
            mapGraphics.DrawLine(new Pen(targetColor1), i, 375 - (float)0.3 * sampleData1[(i *
stepLen + sampleDataLen1) % maxLen], i - 1, 375 - (float)0.3 * sampleData1[(i * stepLen +
sampleDataLen1 - 1) % maxLen]);


          }
        }
        if (checkBoxC.Checked == true)
        {
          mapGraphics.FillEllipse(new SolidBrush(nowColor2), 0, 375 - (float)0.3 *
sampleData2[(0 * stepLen + sampleDataLen2) % maxLen], 3, 3);
          for (int i = 1; i < 1000; i++)
          {
            //mapGraphics.FillEllipse(new SolidBrush(nowColor), (i+sampleDataLen)%1000, 375 -
(float)sampleData[i], 3, 3);
            mapGraphics.FillEllipse(new SolidBrush(nowColor2), i, 375 - (float)0.3 *
sampleData2[(i * stepLen + sampleDataLen2) % maxLen], 3, 3);


            mapGraphics.DrawLine(new Pen(targetColor2), i, 375 - (float)0.3 * sampleData2[(i *
stepLen + sampleDataLen2) % maxLen], i - 1, 375 - (float)0.3 * sampleData2[(i * stepLen +
sampleDataLen2 - 1) % maxLen]);


          }
        }
        if (checkBoxD.Checked == true)
        {
          mapGraphics.FillEllipse(new SolidBrush(nowColor3), 0, 375 - (float)0.3 *
sampleData3[(0 * stepLen + sampleDataLen3) % maxLen], 3, 3);
          for (int i = 1; i < 1000; i++)
          {
            //mapGraphics.FillEllipse(new SolidBrush(nowColor), (i+sampleDataLen)%1000, 375 -
(float)sampleData[i], 3, 3);
```

```
                mapGraphics.FillEllipse(new SolidBrush(nowColor3), i, 375 - (float)0.02 *
sampleData3[(i * stepLen + sampleDataLen3) % maxLen], 3, 3);


                mapGraphics.DrawLine(new Pen(targetColor3), i, 375 - (float)0.02 * sampleData3[(i *
stepLen + sampleDataLen3) % maxLen], i - 1, 375 - (float)0.02 * sampleData3[(i * stepLen +
sampleDataLen3 - 1) % maxLen]);


            }
        }
            pictureBoxMap.Image = bufMap;


    }


    private void DrawTestButton_Click(object sender, EventArgs e)
    {
        scom.WriteLine("zero");
        initialPos=sampleData[(0 * stepLen + sampleDataLen)]*40;
    }


    private void button2_Click(object sender, EventArgs e)
    {
        scom.WriteLine("max");
        totalADDisplacement =initialPos - sampleData[(0 * stepLen + sampleDataLen)] * 40;
    }


    private void controllerComboBox_SelectedIndexChanged(object sender, EventArgs e)
    {
        int tempIndex = controllerComboBox.SelectedIndex - 1;
        if (scom != null)
        {
```

```
            scom.WriteLine("ctr=" + tempIndex.ToString());
        }
    }

    private void buttonSaveData_Click(object sender, EventArgs e)
    {
        FileStream fs = new FileStream("QuickSave.txt",FileMode.Create);
        StreamWriter sw = new StreamWriter(fs);
/*
        for (int loop = 0; loop < maxLen; loop++ )
        {
            sampleData[loop] = loop*2;
            sampleData1[loop] = loop * 3;
            sampleData2[loop] = loop * 4;
            sampleData3[loop] = loop * 5;
        }
*/
            for (int loop = 0; loop < maxLen; loop++)
            {
                string tempStr = "";
                string tempStrData = "";
                string tempStrData1 = "";
                string tempStrData2 = "";
                string tempStrData3 = "";

                tempStr += loop.ToString("D3");

                tempStrData = sampleData[(loop + sampleDataLen) % maxLen].ToString("D4");
                tempStrData1 = sampleData1[(loop + sampleDataLen1) % maxLen].ToString("D4");
```

```
            tempStrData2 = sampleData2[(loop + sampleDataLen2) % maxLen].ToString("D4");
            tempStrData3 = sampleData3[(loop + sampleDataLen3) % maxLen].ToString("D4");

            sw.WriteLine(tempStr + " " + tempStrData + " " + tempStrData1 + " " + tempStrData2
+ " " + tempStrData3);

          }
        MessageBox.Show("File quick saved as QuickSave.txt");
        sw.Flush();
        sw.Close();
        fs.Close();
      }

      private void buttonSuspend_Click(object sender, EventArgs e)
      {
        if (uartOpen == true)
        {
          uartOpen = false;
          buttonSuspend.Text = "Resume";
        }
        else
        {
          uartOpen = true;
          buttonSuspend.Text = "Suspend";
        }
      }
    }
```