MULTI-DOF PRECISION POSITIONING METHODOLOGY USING TWO-AXIS HALL-EFFECT SENSORS

A Thesis

by

YUSUKE KAWATO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 2005

Major Subject: Mechanical Engineering

MULTI-DOF PRECISION POSITIONING METHODOLOGY USING TWO-AXIS HALL-EFFECT SENSORS

A Thesis

by

YUSUKE KAWATO

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

Won-jong Kim (Chair of Committee)	Andrew K. Chan (Member)
Reza Langari (Member)	Dennis O’Neal (Head of Department)

May 2005

Major Subject: Mechanical Engineering

ABSTRACT

Multi-DOF Precision Positioning Methodology
Using Two-Axis Hall-Effect Sensors. (May 2005)

Yusuke Kawato, B.Eng., Keio University
Chair of Advisory Committee: Dr. Won-jong Kim

A novel sensing methodology using two-axis Hall-effect sensors is proposed, where the absolute positioning of a device atop any magnet matrix is possible. This methodology has the capability of micrometer-order positioning resolution as well as unrestricted translational and rotational range in planar 3-DOF (degree-of-freedom) motions, with potential capability of measuring all 6-DOF motions. This research presents the methodology and preliminary experimental results of 3-DOF planar motion measurements atop a Halbach magnet matrix using two sets of two-axis Hall-effect sensors. Analysis of the Halbach magnet matrix is presented to understand the generated magnetic field. The algorithm uses the Gaussian least squares differential correction (GLSDC) algorithm to estimate the relative position and orientation from the Hall-effect sensor measurements. A recursive discrete-time Kalman filter (DKF) is used in combination with the GLSDC to obtain optimal estimates of position and orientation, as well as additional estimates of velocity and angular velocity, which we can use to design a multivariable controller.

The sensor and its algorithm is implemented to a magnetic levitation (maglev) stage positioned atop a Halbach magnet matrix. Preliminary experimental results show its position resolution capability of less than 10 μm and capable of sensing large rotations. Controllers were designed to close the control loop for the three planar degrees of freedom motion using the GLSDC outputs at a sampling frequency of 800 Hz on a Pentek 4284 digital signal processor (DSP). Calibration was done by comparing the laser interferometers'

and the GLSDC’s outputs to improve the positioning accuracy.

ACKNOWLEDGMENTS

First of all, I would like to thank my thesis advisor, Dr. Won-jong Kim. Dr. Kim has consistently given me encouragement and motivated me throughout my study at Texas A&M University. I truly admire his knowledge and experience in the field of electromechanical systems, control systems, and precision positioning devices.

I wish to thank Drs. Andrew K. Chan and Reza Langari for serving on my thesis committee. Dr. Chan's class in electromagnetic fields has helped me greatly in understanding the magnetic field analysis. Dr. Langari has provided me with the knowledge of control systems design, which was an essential part of my thesis work.

In our Precision Mechatronics Laboratory, I would like to express my special thanks to Tiejun Hu and Shobhit Verma. Tiejun has helped me in the experimental setup using the maglev stage, and in understanding in the working principles. Shobhit's expertise in circuitry and hardware and his encouragement have helped me get things going. I have learned a lot from them, a lot of things which could never be taught in class. I really enjoyed working with them, and could not have done all of this work without their continuous help.

Finally, I would like to thank my parents, Mr. Naoki Kawato and Mrs. Reiko Kawato. Without their unconditional love, I would never have come so far.

TABLE OF CONTENTS

CHAPTER Page

I INTRODUCTION 1

- A. Prior Art 1
  - 1. High-Precision Positioning-Sensing Techniques 2
  - 2. Magnetic Field Sensors 5
  - 3. Magnet Arrays 7
- B. Proposed Sensing Method 9
- C. Thesis Overview 12
- D. Contribution of Thesis 12

II ELECTROMECHANICAL DESIGN 14

- A. Instrumentation 14
  - 1. Maglev Stage 14
  - 2. Hall-Effect Sensor 14
  - 3. Laser Interferometers 16
- B. Experimental Setup 17
  - 1. Hall-Effect Sensor Mount 17
  - 2. Power-Supply Circuit 18
  - 3. Amplifier and Analog Filter Design 19
  - 4. VMEbus Connection 20
  - 5. Grounding and Shielding 22

III MAGNETIC FIELD ANALYSIS 25

- A. Halbach Magnet Matrix 25
  - 1. Design 25
  - 2. Fourier Series Analysis - Strong Side 26
  - 3. Fourier Series Analysis - Weak Side 30
- B. Three-Phase Planar Actuator Coils 35

IV ALGORITHM 39

- A. Obtaining Least Squares Model for Sensor Outputs 39
- B. Gaussian Least Squares Differential Correction (GLSDC) 45
  - 1. GLSDC Problem Formulation 45
  - 2. Algorithm for 3-DOF Positioning 45

CHAPTER Page

3. Algorithm for 6-DOF Positioning 50
C. Recursive Discrete-Time Kalman Filter (DKF)................................ 51
D. Calibration Method 55
E. Dynamical Modeling and Control System Design............................ 55
1. Dynamic Model 55
2. SISO Controller Design.. 56

V RESULTS.. 60

A. Simulation Results 60
1. GLSDC Results with No Plant 60
2. DKF Dynamics with No Plant.. 61
3. GLSDC and DKF with Plant and Controller........................ 69
B. Experimental Results 71
1. 2-DOF Positioning... 71
2. 2-DOF Positioning with Calibration.................................... 78
3. 3-DOF Positioning... 78

VI CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK.................. 95

A. Conclusions... 95
B. Applications 97
C. Suggestions for Future Work 97

REFERENCES 99

APPENDIX A : MATLAB AND REAL-TIME C CODES...................................... 102

A. MATLAB Code for Curve-Fitting... 102
B. MATLAB Code for 3-DOF GLSDC and DKF................................ 103
C. C Code for 3-DOF GLSDC 108

VITA 115

LIST OF TABLES

TABLE Page

I Specifications of 2D-VH-11SO Hall-effect sensor 17

II Hall-effect sensor connections to Pentek 6102 23

III Recursive discrete-time Kalman filter 54

IV DKF results of changing values of q_x with $q_\theta = 10^5$ rad^2/s^6 68

V DKF results of changing values of q_θ with $q_x = 10^2$ m^2/s^6 68

LIST OF FIGURES

FIGURE Page

1 Laser interferometer setup for the ATP stage 3

2 Hall-effect principle 6

3 Prior art of planar motor magnet matrices 8

4 Halbach magnet matrix 8

5 1-D sensing mechanism 9

6 Representative Hall-effect sensor setups 11

7 Photograph of the experimental setup 15

8 Photograph of the 2D-VH-11SO mounted on an IC board 16

9 Overview of the sensing circuit and connections to 2D-VH-11SO 18

10 A photograph of the power supply circuit 19

11 Designed amplifier and anti-aliasing filter 21

12 Frequency response of the anti-aliasing filter 22

13 Shield line connection 23

14 Halbach magnet array’s magnetization (strong side) 27

15 Fourier coefficients of the Halbach magnet array flux density 29

16 Magnetic flux density analysis of Halbach magnet matrix for the strong side, at the air gap of Z_0=3 mm 31

17 B_Z with respect to air gap 32

18 Fourier coefficients with respect to air gap 32

FIGURE Page

19 Halbach magnet array's magnetization (weak side) 33

20 Magnetic flux density analysis of Halbach magnet matrix for the weak side, at the air gap of Z_0=3 mm 34

21 Actuator coil dimensions of the ATP stage 36

22 Current density of coil actuator 37

23 Fourier coefficients of magnetic flux density due to actuator coil at air gap of 1 mm 37

24 Magnetic flux density analysis of coil actuators at air gap of 1 mm 38

25 GLSDC and DKF series design 40

26 An example of one Hall-effect sensor output 40

27 Fitting the phase lag to determine offsets 41

28 Definition of sensor locations and axes of measurement 41

29 Output of the curve-fitted model 46

30 Modeling error from curve-fitting 46

31 Gaussian least squares differential correction algorithm 47

32 10-Hz controller for translational motion in X and Y 58

33 Step response to a 100-μm step in X 58

34 20-Hz controller for rotational motion in θ_z 59

35 Step response to a 1° step in θ_z 59

36 Simulation GLSDC output to a 200-μm step input in Y, with no sensor noise 62

37 Simulation DKF output to a 100-μm step input in X with q_x = 100 $\mathrm{m}^2/\mathrm{s}^6$ and $q_\theta = 1 \times 10^5$ $\mathrm{rad}^2/\mathrm{s}^6$ 64

FIGURE Page

38 Response to step input of 1-mrad in θ_z with $q_x = 100$ m^2/s^6 and $q_\theta = 1 \times 10^5$ rad^2/s^6 65

39 Simulation results of X for GLSDC and DKF to a 100-μm step input in X with no sensor noise, $q_\theta = 10^5$ rad^2/s^6 66

40 Simulation results of θ_z for GLSDC and DKF to a 1-mrad step input in θ_z with no sensor noise, $q_x = 100$ m^2/s^6 67

41 Simulink block diagram for GLSDC and DKF 70

42 Simulation result of GLSDC output for a 10-μm step 72

43 Simulation result of GLSDC output for a 100-μm step 72

44 Simulation result of GLSDC output for a 100-μm step, feedback using DKF output 73

45 Simulation result of DKF output for a 100-μm step, feedback using DKF output 73

46 Experimental results when there is no motion at the origin, and FFT results 74

47 Experimental results when there is no motion at the origin using the fast controller, and FFT results 76

48 Experimental results of 10-μm steps in Y 82

49 Experimental results of positioning platen in X and Y following a zigzag trajectory 83

50 Laser interferometer and GLSDC output to 1-mm consecutive steps in Y . . 84

51 Laser interferometer and GLSDC output to a 4-mm step in Y 84

52 Laser interferometer and GLSDC output to a 1-mm step in Y with calibration 85

53 Laser interferometer and GLSDC output to a 4-mm step in Y with calibration 85

54 Experimental setup with three sensors 86

55 Experimental setup with three sets of 2D-VH-11SO 86

FIGURE Page

56 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 10-μm steps in Y 87

57 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 1-mm steps in Y 88

58 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for a 10-mm step in Y 89

59 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 100-μrad steps in θ_z 90

60 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 1-mrad steps in θ_z 91

61 Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 10-mrad steps in θ_z 92

62 Experimental results for ramp input in $+\theta_z$ rotation 93

63 Experimental results for ramp input in $-\theta_z$ rotation 94

CHAPTER I

INTRODUCTION

For high-precision positioning of any device such as wafer steppers, magnetic-suspension stages, surface motors, and long-range scanning stages, conventional methods use laser interferometers, optical sensors, or capacitance gauges [1–4]. These sensors have high resolution and low positioning noise, allowing sub-nanometer position measurements. The non-contact nature of these sensors is suitable in magnetic-levitation (maglev) applications.

This chapter presents the conventional sensing techniques for precision positioning and methods of measuring magnetic fields. Prior art of magnet matrices is explained, which will be utilized in our proposed sensing methodology.

A. Prior Art

Some conventional sensors which are used for precision positioning are described in this section. In general, such sensors can be categorized as analog and digital sensors. Digital sensors have a fixed resolution depending on the pulse width or the frequency of the received signal. Some examples are laser interferometers and encoders. Analog sensors do not have a fixed resolution due to sensor signal noise, and can be said to have infinite resolution, but there is a limitation due to linearity, noise, and quantization. Some examples of analog sensors are capacitance gauges, linear variable differential transformers (LVDTs), and potentiometers.

Another way to classify sensors is whether they give absolute or relative position measurements. Absolute position measurements tell us the position regardless of where the initial position is, and will give the same output at all times for that position. Some exam-

---

The journal model is *IEEE Transactions on Automatic Control.*

ples are potentiometers and LVDTs. On the other hand, relative position measurements tell us the position with respect to some position, usually the initial position. Some examples are incremental encoders and laser interferometers. These absolute position sensors count ‘pulses’ of signals, and hence each time it is reset, the number of counts is reset.

1. High-Precision Position-Sensing Techniques

Interferometry is a technique that uses the wavelength of light as the unit of measurement [5]. By counting the number of wavelengths of the light using the receiver, an interferometer system measures the change in distance. A detailed explanation can be found in [5]. Note that laser interferometers are digital sensors with relative position sensors.

An example of a laser-interferometer setup is shown in Figure 1. This is the setup currently in use for the ATP (Advanced Technology Program) maglev stage presented in [2], and consists of three laser interferometers and receivers manufactured by Agilent Technologies[1]. With the three readings from the sensors, an algorithm solving geometric relations is implemented to estimate the position and orientation of the platen in the XY plane (X, Y, θ_z). The laser receiver is connected to the laser axis board, which is mounted on the VMEbus. The setup is capable of resolution up to 0.6 nm at velocities up to 500 mm/s.

An advantage of using laser interferometers is the large measurement range for translational motion. As long as there is nothing interfering the laser beam and there is a mirror fixed to the object which reflects the laser beam, high-resolution measurements can be obtained.

One major downside of the use of laser interferometers is the limitation in rotation. Since the laser beam reflected off the mirror must reflect into the laser receiver, even small rotations on the order of a few milliradians can cause the laser beam to stray off the re-

---

[1] Agilent Technologies, Inc. 395 Page Mill Rd. Palo Alto, CA 94306

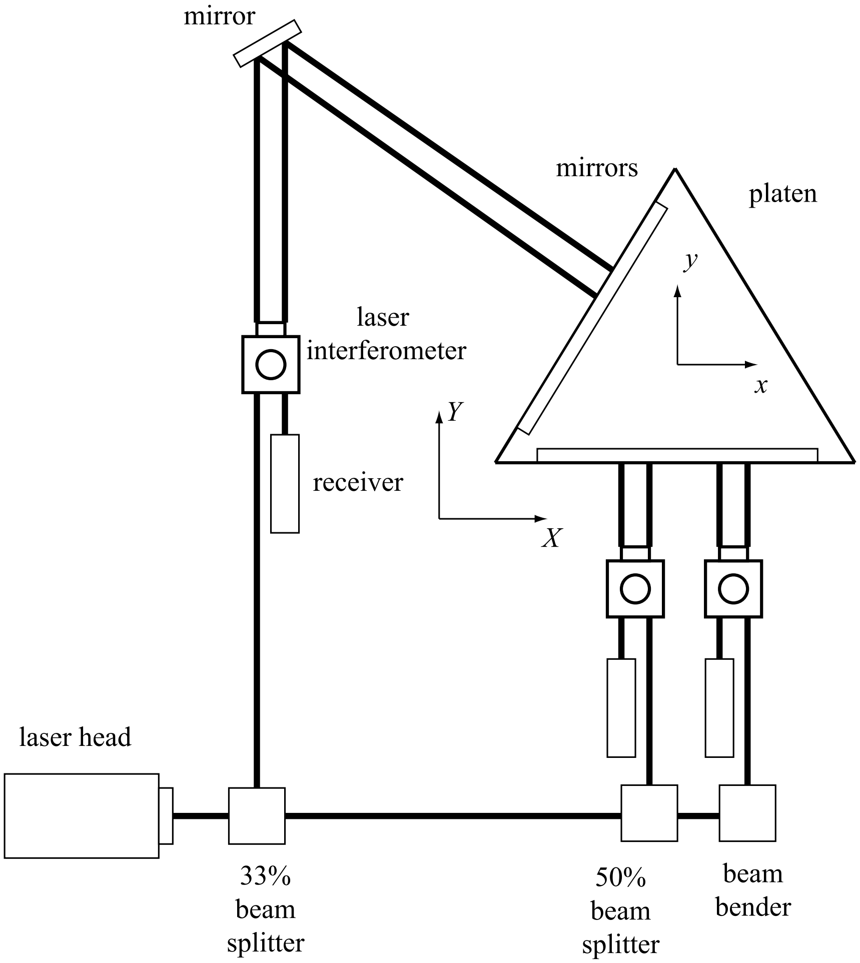


Fig. 1. Laser interferometer setup for the ATP stage

ceiver, which causes the platen to become unstable. In a practical sense, this becomes more problematic when applying large (a few millimeters) step inputs because this can generate a responsive jerk of the platen, which may cause the platen to rotate enough to cause instability.

Capacitance gauges are another alternative for precision positioning. Capacitance gauges measure the capacitance between two plates, one being the probe and the other being the object to be measured. Measurements utilizing capacitance require three basic components [6]:

- A probe that uses changes in capacitance to sense changes in distance to the target
- Driver electronics to convert these changes in capacitance into voltage changes
- A device to indicate and/or record the resulting voltage change

The capacitance between two plates is determined by three things [6]:

- Size of the plates – Capacitance increases as the plate size increases.
- Gap size – Capacitance decreases as the gap increases.
- Material between the plates (the dielectric) – Dielectric material will cause the capacitance to increase or decrease depending on the material.

An example of the use of capacitance gauges is in the MRI (Major Research Instrumentation) maglev stage presented in [7], which is a maglev stage with 6-DOF positioning capabilities. Capacitance gauges are used to detect the vertical (Z), roll (θ_x), and pitch (θ_y) motions. The levitated part (namely, the platen) is fabricated out of aluminum, which provides a flat surface with large area. The setup uses ADE[2] 2810 probe and 3800 OEM gauging module, which has 1.3-nm resolution with 0.02% linearity.

---

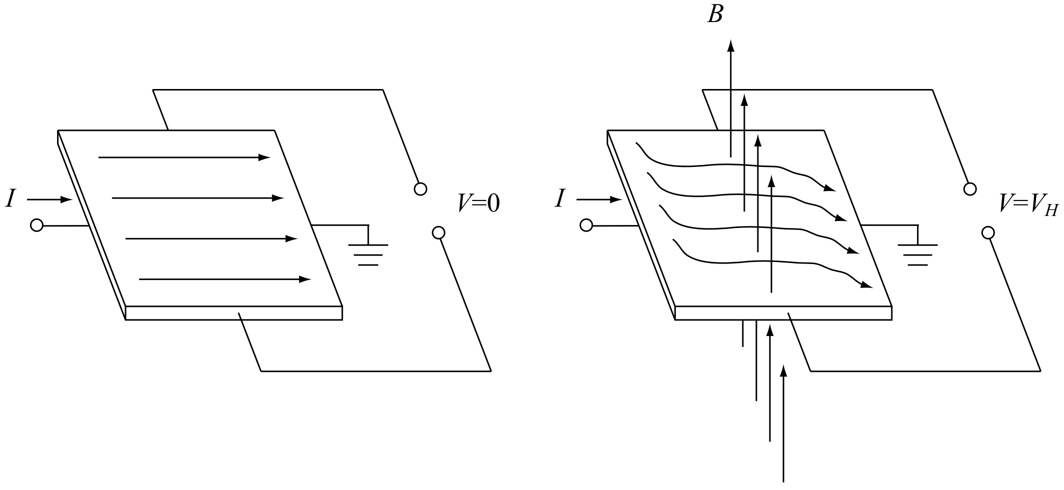
[2] ADE Technologies, 1525 McCandless Drive, Milpitas, CA 95035

Although these sensors have a very small resolution, they have downsides in cost, design, and limitations in travel range. Components of such sensors, such as mirrors, laser heads, and laser interferometers which make up the sensing system add up to be very expensive. For example, the laser interferometry setup for the maglev stage in [2] cost approximately $ 48,000, and the capacitance gauges in [7] cost approximately $15,000. In terms of design, these sensors require a very good surface finish as a flat reference. For a laser interferometer, a flat mirror is necessary for the laser beam to be reflected to the receiver, and for a capacitance gauge, a flat metal surface is necessary to have consistent measurements. The mirrors may become very large in order to achieve long travel range, which make the platen heavy, affecting its performance and its design. The use of such devices is limited to very clean, contaminant-free environments.

It must also be noted that the applications of maglev devices are not only for photolithography that requires nanometer-precision positioning, but for microassembly where nanometer resolution is not a strict requirement. Furthermore, there may be more need for larger angular displacement capabilities, which cannot be met using conventional laser interferometers. Having these considerations, we seek for inexpensive sensors with sufficient positioning capabilities allowing large rotations, with absolute outputs with respect to position. We focus on a methodology to estimate the position based on the magnetic field generated by the magnet matrices which are used to levitate and actuate the platen by using Hall-effect sensors.

2. Magnetic Field Sensors

Many magnetic field sensors utilize the Hall effect to detect the magnitude of the magnetic flux density. When a current-carrying conductor is placed in a magnetic field, a voltage will be generated perpendicular to both the current and the field. Figure 2 illustrates the basic principle of the Hall effect. It shows a thin sheet of semiconducting material (Hall



(a) (b)

Fig. 2. Hall-effect principle. (a) no magnetic field, (b) magnetic field present

element) through which a current is passed. The output connections are perpendicular to the direction of current. When no magnetic field is present, as shown in Figure 2 (a), current distribution is uniform and no potential difference is seen across the output. When a perpendicular magnetic field is present, as shown in Figure 2 (b), a Lorentz force is exerted on the current. This force disturbs the current distribution, resulting in a potential difference (voltage) across the output. This voltage is the Hall voltage (V_H). The interaction of the magnetic field and the current is shown in the following equation.

$$V_H \propto I \times B \tag{1.1}$$

Hall-effect sensors are used in various fields, for they are contactless, small in size, reliable, low-cost, and not sensitive to harsh and polluted environmental conditions, and have a linear relation between the output voltage and the measured magnetic flux density. However, they have limited accuracy due to offset, noise, temperature, and aging effects. The measured field is usually nonlinear and hence there is a nonlinear relation between the position and the sensor readings [8]. Nevertheless, Hall-effect sensors have been utilized in various measurement techniques, such as current sensing [9], sensing the movement of ferrous metal targets with Hall-effect proximity sensors, and measuring positions of rotating machinery [10]. Previous work showed their capability in sensing rotations of less than one degree [11].

An LVDT is another position device which utilizes the magnetic field to estimate the position. An LVDT consists of coils which measure the difference in inductance which results from the moving magnetic core. A detailed explanation of its working principle can be found in [12].

Either of these methods can be applied to our proposed method. Due to the simplicity and compactness, we use Hall-effect sensors in our research.

3. Magnet Arrays

There has been a number of magnet matrices which have been proposed and designed primarily for planar permanent-magnet (PM) motors. Some examples are the magnet matrix by Asakawa [13], Hinds [14], Ebihara et al. [15], Trumper et al. [16], and Hazelton and Gery [17]. The magnet matrices of Asakawa, Hinds, and Ebihara et al. are shown in Figure 3. In the figures, non-magnetic material is shown with blank spaces.

We utilize the periodicity and orthogonality of such magnet matrices to estimate the position of the platen atop a magnet matrix. In this research, we focus on the using the Halbach magnet matrix shown in Figure 4. Detailed analyses of the Halbach magnet matrix is discussed in Chapter III.

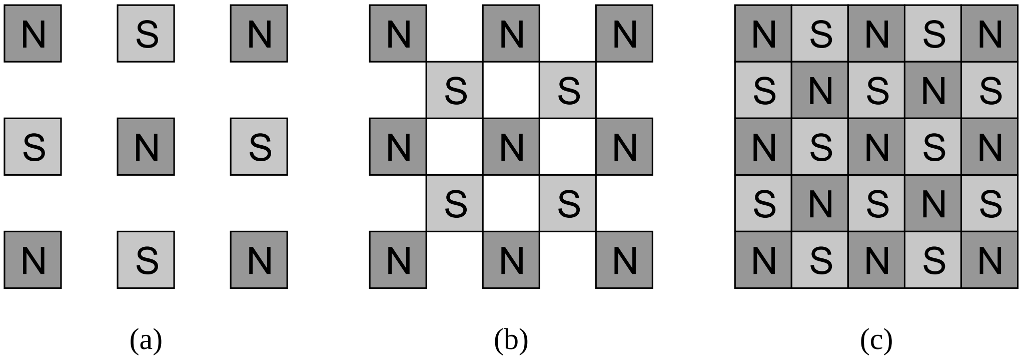


Fig. 3. Prior art of conventional planar motor magnet matrices. (a) Asakawa, (b) Hinds, and (c) Ebihara

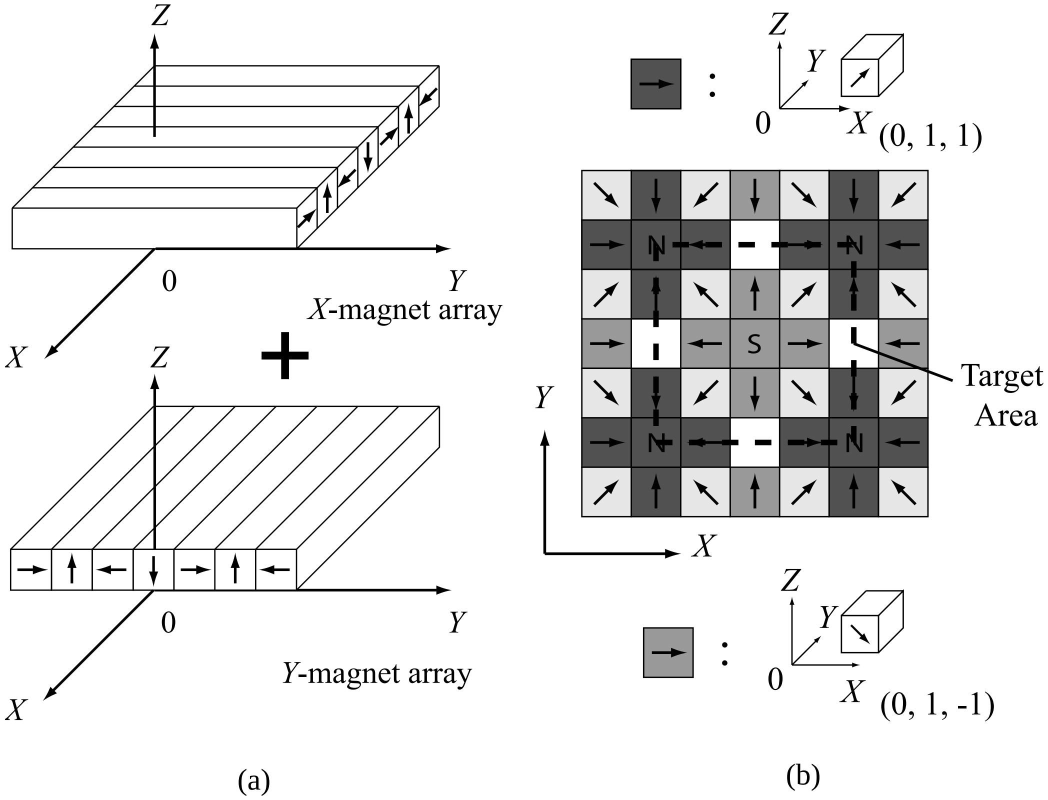


Fig. 4. Halbach magnet matrix [1,16] (a) Conceptual superimposition of the two orthogonal Halbach magnet arrays to produce a concentrated-field magnet matrix. (b) Top view of the concentrated-field magnet matrix.

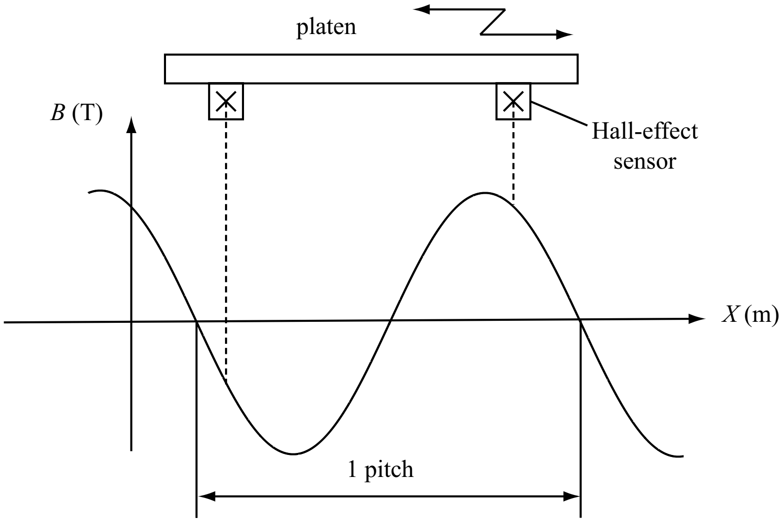


Fig. 5. 1-D sensing mechanism

B. Proposed Sensing Method

We propose a novel methodology for a 6-DOF positioning of a platen atop a magnet matrix, using multi-axis Hall-effect sensors. This methodology is capable of sensing large rotation as well as lateral motion in 6 DOFs, and is capable of positioning resolution on the order of micrometers. This methodology will allow positioning of the platen within the ‘target area,’ which is one pitch by one pitch (50.8 mm × 50.8 mm, or 2” × 2”) of the Halbach magnet matrix.

First, consider a case for positioning in 1 D, as shown in Figure 5 for positioning within one pitch. The platen will move along the *X* axis where there is some sinusoidal magnetic field *B*. On the platen are two Hall-effect sensors which can detect the magnetic flux density.

If there is only one Hall-effect sensor, it can detect the location of the platen, but it will not be unique because there are two points which have the same magnetic flux density within one pitch. Further, it is impossible to detect the direction of motion because the magnetic flux density is symmetric about the local maximum/minimum of B.

If there are two Hall-effect sensors positioned with some phase lag, it is possible to detect the unique position of the platen within one pitch, as well as the direction of motion. Further, if the Hall-effect sensors are located

$$\left(\frac{1}{4}+\frac{1}{2}n\right)\times \text{pitch}, \quad n = 1, 2, 3, \ldots \tag{1.2}$$

apart, at least one sensor is located in a ‘sensitive’ region where the gradient of B is large. This will allow positioning within the ‘target area’ to be sensitive at all locations.

Next, consider positioning in a 2-D XY plane without rotation. If there are orthogonal magnetic fields in a plane and two measurements can be taken for each axis, we can position the platen. This is precisely the reason we consider the use of magnet matrices, because magnet matrices have orthogonal magnetic fields along the X and Y axes. An example of a sensor setup for 2-D planar positioning is shown in Figure 6 (a). Note that this has only the minimum number of sensors, and having more redundant sensors will improve the accuracy of positioning.

Further, the fact that these magnetic fields are orthogonal brings one more advantage. By having four sensor outputs and two orthogonal magnetic fields, it is possible to detect the orientation θ_z of the platen as well. This approach can be extended to 6-DOF position sensing, where in that case, the coefficients of the curve-fitting will be modeled as a function of the air gap, as explained in Chapter IV.

An important part of this research is to find a way to map the nonlinear relationship between the four sensors’ outputs to the position and orientation of the platen. Candidate methods for this nonlinear mapping include neural networks [18] and Gaussian least squares differential correction (GLSDC). The GLSDC algorithm is implemented due to its more deterministic nature, and resulted in better convergence.

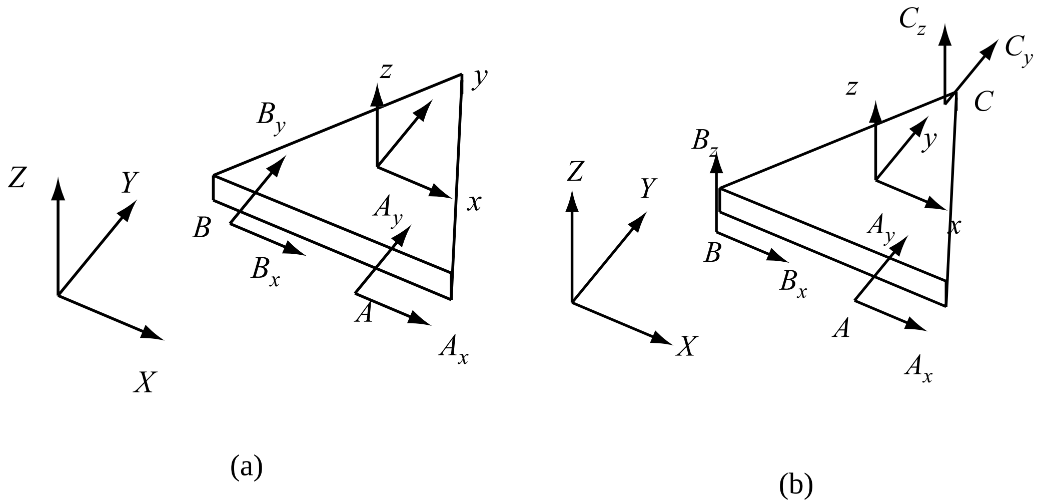


Fig. 6. Representative Hall-effect sensor setups with minimum number of sensors (a) For 3-DOF position sensing in a plane. (b) For 6-DOF position sensing.

In addition to the GLSDC, a recursive discrete-time Kalman filter (DKF) is designed and implemented. The DKF is used to obtain optimal estimates based on the GLSDC's position and orientation outputs. The DKF also outputs additional estimates of velocity, which we can use to design a multivariable controller.

Our approach has many attractive features including the following:

- very small sensor with unrestricted range in planar (X, Y) and yaw (θ_z) motion
- no costly laser-interferometer setup required (a two-axis Hall-effect sensor is $10 apiece)
- relatively simple electronic circuits with no demanding design constraints
- can be applied in various fields, not limited to magnetic fields
- capable of sensing large rotations with sub-degree resolution

The same method can be extended to positioning in 6 DOFs. A possible setup for 6-DOF positioning with minimum number of sensors is shown in Figure 6 (b). The algorithm is presented in Chapter IV.

C. Thesis Overview

This thesis encompasses the work performed on this sensing mechanism to date. In Chapter II, the electromechanical design and the experimental setup is discussed. In Chapter III, analysis of the magnetic field due to the Halbach magnet matrix is discussed. This is the basis for curve-fitting model of the Halbach magnet matrix. Chapter IV explains the algorithm which is implemented to make this sensing mechanism work. This is the most essential part of this research, where the GLSDC algorithm and the design of a DKF are discussed. Designed controllers are also discussed in this chapter. Simulations and experimental results are presented in Chapter V. Conclusions, recommendations for future work, and applications for which the sensing methodology is suitable are discussed in Chapter VI.

D. Contribution of Thesis

This masters thesis deals with the design and construction of a new high-precision sensing mechanism and empirical validation of the proposed design. The capabilities of the sensing mechanism are discussed thoroughly. The sensing methodology can be used in any situation where appropriate modeling can be done to estimate the magnetic field. The algorithm proposed in this research can serve as a basis for using this in a more complex or redundant sensing scheme.

The contributions of this thesis include: (1) design of mechanical and electrical instrumentation of the sensing device, (2) establishment of an algorithm to map nonlinear relations from four Hall-effect-sensor readings to the position and orientation of the platen, and (3) implementation of this methodology in the maglev stage as a proof of concept.

This research has been done with the extensive use of the magnetically levitated stage (ATP stage) presented in [2]. The development of the ATP stage has been done by Nikhil Bhat, a former graduate student of Won-jong Kim, and Tiejun Hu, a doctoral student. The controls software was developed by Jie Gu, a former graduate student of Won-jong Kim.

CHAPTER II

ELECTROMECHANICAL DESIGN

This chapter explains the electromechanical design and experimental setup.

A. Instrumentation

1. Maglev Stage

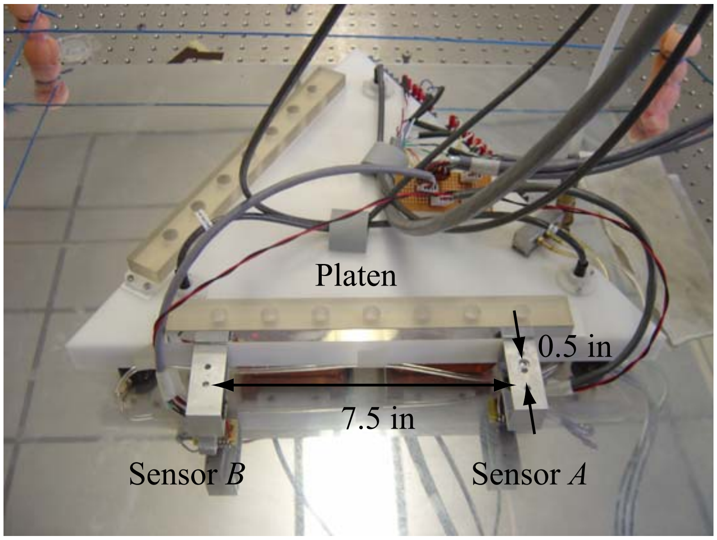
The maglev stage used in our experimental setup is shown in Figure 7. It is a magnetically-levitated positioner (namely, the platen) capable of positioning in 6 DOFs with a 20-nm positioning resolution with a travel range of 160 × 160 mm [2]. It is capable of moving at 0.5 m/s in a plane at an acceleration of 5 m/s^2. Beneath the platen is a double-axis Halbach magnet matrix covered with a mirror-finished thin aluminum plate. A detailed explanation and analysis of the Halbach magnet matrix is presented in Chapter III. The platen is currently suspended using three aerostatic bearings, and generates force to move in 6 DOFs using three planar motors, which are attached to the bottom of the platen. Three laser interferometers and three laser distance sensors are used for the 6-DOF position sensing.

2. Hall-Effect Sensor

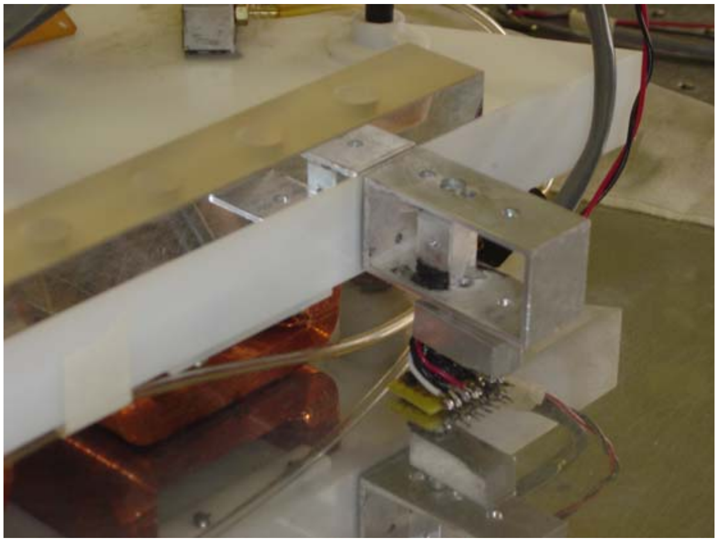
The Hall-effect sensor used in this research is a two-axis Hall-effect sensor 2D-VH-11SO manufactured by Sentron AG [3] [19]. It is an 8-pin, surface mount, small outline integrated circuit (SOIC). It can measure the magnetic flux density of two orthogonal axes about the chip's surface. A photograph of the 2D-VH-11SO mounted on an IC board is shown in Figure 8. The specifications are shown in Table I. The 2D-VH-11SO requires either a constant current source of 2 mA or a constant voltage source of 5 V.

---

[3] Sentron AG, Baarerstrasse 73 CH - 6300 Zug, Switzerland



(a)



(b)

Fig. 7. Photograph of the experimental setup (a) Multi-DOF positioner with two sets of 2D-VH-11 Hall-effect sensors mounted at the base. The triangular platen is placed atop a mirror-finished aluminum plate, and beneath the plate is the Halbach magnet matrix. (b) Close-up view of the mounted Hall-effect sensor

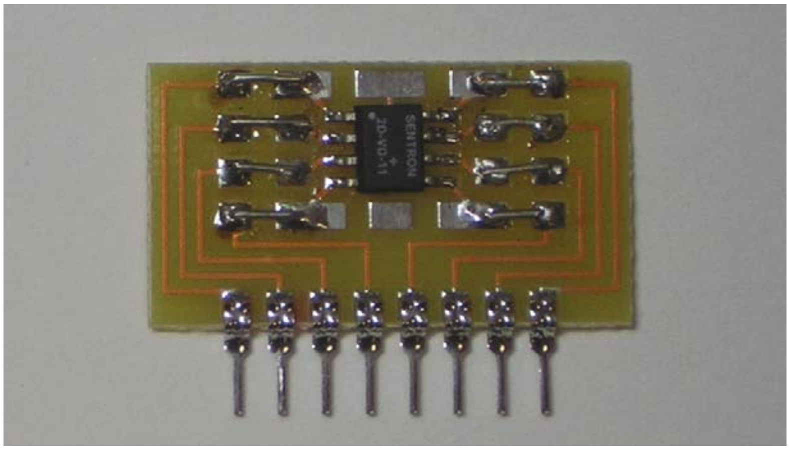


Fig. 8. Photograph of the 2D-VH-11SO mounted on an IC board

3. Laser Interferometers

Laser interferometers are used as a reference to validate the position data of the proposed method. Figure 1 shows the metrology arrangement of the laser-interferometer system. The laser-interferometer system consists of a laser head, laser interferometers, beam benders, beam splitters, laser receivers, and laser-axis boards. The Agilent 5517D laser head has a HeNe laser source at the wavelength of 632 nm. The resolution from the Agilent 10897B laser axis board is 0.6 nm. It provides 35-bit position data and 24-bit velocity data. The Pentek 4284 DSP reads the position measurement values from the three laser-axis boards through the VMEbus.

These three laser interferometers give us 3-DOF position information of the platen: translation in the X axis, translation in the Y axis, and rotation around the z axis. The laser interferometers' outputs are used to calibrate the estimates obtained by the proposed sensing method.

Table I. Specifications of 2D-VH-11SO Hall-effect sensor

Item	Conditions	Typical value
Input resistance	B = 0 mT, I_c = 2 mA	2.2 kΩ
Output resistance	B = 0 mT, I_c = 2 mA	8.5 kΩ
Output voltage	Constant Current Drive	400 mV
	B = 1 T, I_c = 2 mA	
Offset voltage	B = 0 mT, I_c = 2 mA	±3 mV
Sensitivity	I_c = 2 mA	400 mV/T
Magnetic sensitive volume		$0.25 \times 0.25 \times 0.20$ mm^3

B. Experimental Setup

An overview of the sensing circuit is shown in Figure 9. The circuit consists of a power supply circuit for a Hall-effect sensor, a Hall-effect sensor, an amplifier and low-pass-filter circuit for anti-aliasing. There are two sets of this circuit, one for each sensor. Each part is explained in detail in the following sections.

1. Hall-Effect Sensor Mount

A photograph of the Hall-effect sensor mounted on the platen is shown in Figure 7. The sensor mounts are fixed to the platen by screws, and are designed so that minor adjustments can be made to position and align the sensors. The sensors are mounted approximately 190.5 mm (7.5”) apart in y and 12.7 mm (0.5”) apart in x. The reason for this is to maximize the sensing capability, as discussed in Chapter I. The surface of the Hall-effect sensor chip is positioned close to the surface of the magnet matrix. However, placing them too close will be a source of modeling error because unmodeled higher harmonics of the Fourier

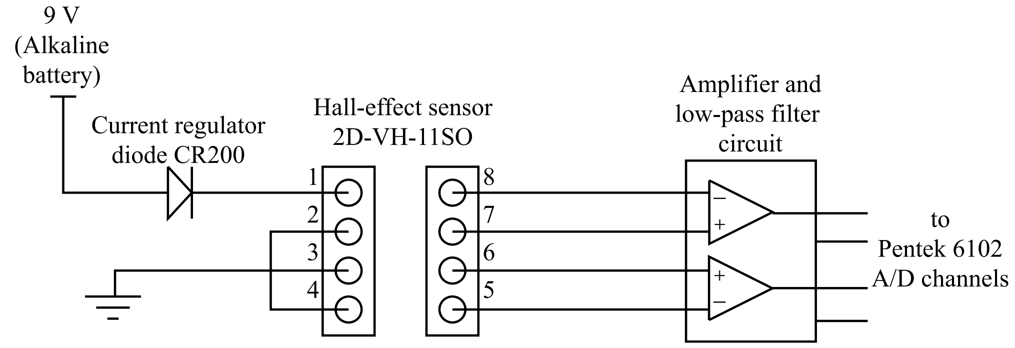


Fig. 9. Overview of the sensing circuit and connections to 2D-VH-11SO

series will come into play. The air gap between the two surfaces is set to be approximately 2 mm. The sensors' axes of measurement are adjusted to be aligned with the platen's body-fixed *xyz* axes.

2. Power-Supply Circuit

The power supply for the Hall-effect sensor can be either (1) 5-V constant voltage source, or (2) a 2-mA constant current source. Comparison was made by testing both cases, using Agilent E3646A for (1) and current regulator diodes CR200 by Vishay Siliconix[4] for (2). Both power supply circuits were tested by connecting them to a Hall-effect sensor, and the outputs were acquired. Experimental results showed that the sensor outputs' fluctuation is smaller when using a constant current source. The sample standard deviation of the sensors' output noise was 5.76 mV for the constant voltage source and 9.90 mV for the constant current source. The fast Fourier transform (FFT) plots of the outputs from the Tektronix amplifier showed peaks at 60 Hz and 180 Hz, which was the primary cause of the noise. Hence, the constant current source using CR200 is used for our experimental

---

[4] Vishay Intertechnology, Inc. 63 Lincoln Highway, Malvern, PA 19355-2120

setup.

The CR200 provides the 2-mA constant current and guarantees a tight ±10% tolerance with excellent temperature stability [20]. A photograph of two sets of power supply circuit is shown in Figure 10. The power to the diode is provided by 9-V Alkaline batteries, and one power supply circuit will supply power for one Hall-effect sensor.

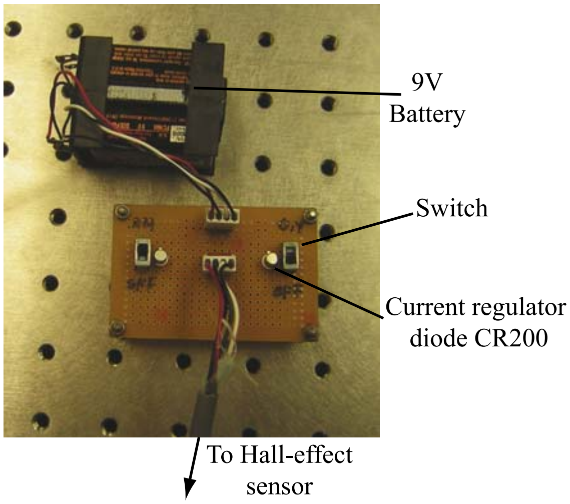


Fig. 10. A photograph of the power supply circuit

3. Amplifier and Analog Filter Design

At first, for the amplifier, ADA400A manufactured by Tektronix[5] was used. This amplifier has adjustable gain settings $(0.1\times, 1\times, 10\times, 100\times)$ with common mode rejection ratio (CMRR) of greater than 100 dB and a typical noise of $\leq 30\ \mu$V rms at 100× gain. This also has selectable filters at 100 Hz, 300 Hz, and 1 kHz which seemed appropriate for our setup. However, it was found that there is a 60-Hz peak noise which could not be attenu-

---

[5] Tektronix, Inc. 14200 SW Karl Braun Drive, P. O. Box 500, Beaverton, OR 97077

ated. Hence, an amplifier circuit and analog filters were designed and implemented, which resulted in less noise compared to the Tektronix amplifier and no 60-Hz peak noise.

The implemented circuit is shown in Figure 11. The operational amplifier (op-amp) TL072ACP manufactured by Texas Instruments[6] is used. The left half of the circuit is a differential amplifier which is designed to filter out the common-mode noise, and the right half is the amplifier. The amplifier amplifies the signal by

$$Av = -\frac{27.4k\Omega}{1.3k\Omega} = 21.07 \tag{2.1}$$

The ±15 V power supply for the op-amp is supplied by Agilent E3646A. Following the amplifier is an anti-aliasing filter with a corner frequency of 200 Hz. This filter is put before the input of the A/D channels to prevent aliasing by the A/D converters, and will also cut off the high-frequency noise. Figure 12 shows the frequency response of this anti-aliasing filter. The PCB board in Figure 11 (b) is mounted on the rack which the VME personal computer (PC) is mounted.

4. VMEbus Connection

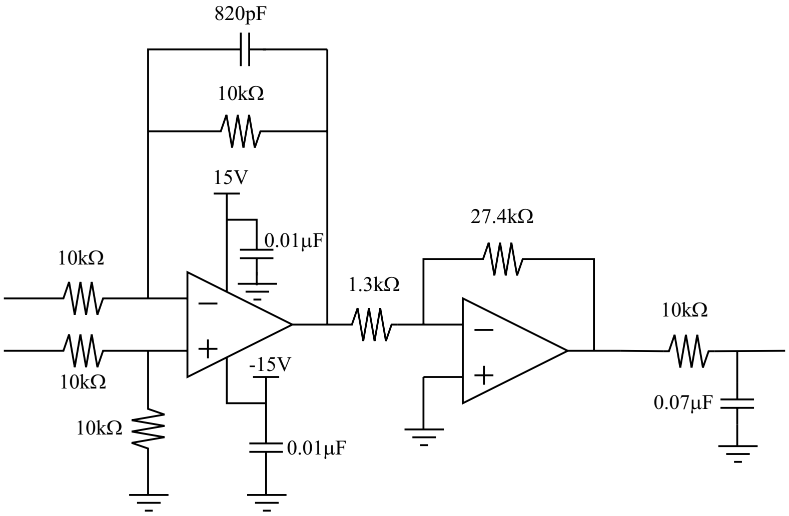
The setup uses Pentek[7] 4284 for the DSP, Pentek 6102 for the analog-to digital (A/D), and DATEL[8] DVME622 for the digital-to-analog (D/A) converter board for the control of the actuator coils. The Pentek 4284 board has a single TMS320C40 processor with 50 Million floating-point operations per second (MFLOPS) peak processing power. The real-time control routine is executed by this board, and is connected to Pentek 6102 and Datel DVME622 through the VMEbus. The Hall-effect sensor outputs are connected to

---

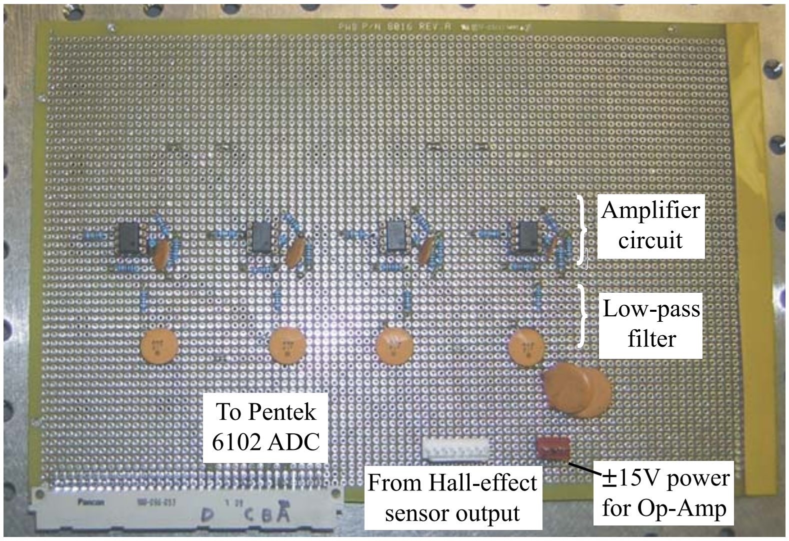
[6] Texas Instruments Inc. 13532 N. Central Expressway, M/S 3807 Dallas, Texas 75243-1108

[7] Pentek, Inc. One Park Way, Upper Saddle River, NJ 07458

[8] DATEL, Inc. 11 Cabot Blvd. Mansfield, MA 02048-1151



(a)



(b)

Fig. 11. Designed amplifier and anti-aliasing filter. (a) Circuit diagram. (b) Photograph of four sets of amplifier and anti-aliasing filter

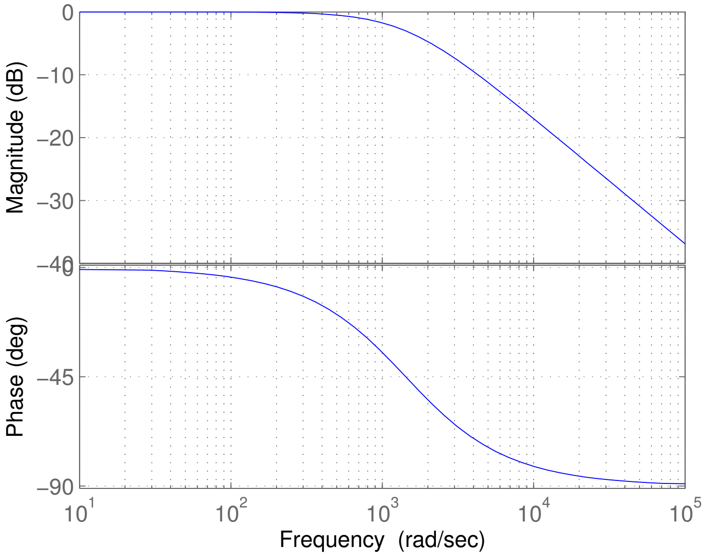


Fig. 12. Frequency response of the anti-aliasing filter

Pentek 6102's A/D converter channels. The connections are shown in Table II. The symbols correspond to the axes defined in Figure 6 (a).

5. Grounding and Shielding

Since the Hall-effect sensor is very sensitive to any external noise due to stray magnetic field, it is critically important to ground and shield all of the cables appropriately. All shielded cables are connected as shown in Figure 13. The ground and the shield lines are connected to the side closer to the signal source, and are not connected on the other side. The other end is taped and protected with insulating electrical tape so that no portion of the wire can touch a signal, ground, or a metal point. This prevents ground looping and reduces the noise. Reference [21] provides an explanation of the ideal setup to reduce the sensor noise.

Table II. Hall-effect sensor connections to Pentek 6102

Sensor	Aligned axis	Symbol	Channel on 6102
A	$-y$	a_1	A1
A	$-x$	b_1	B1
B	$-y$	c_1	C1
B	$-x$	d_2	D2

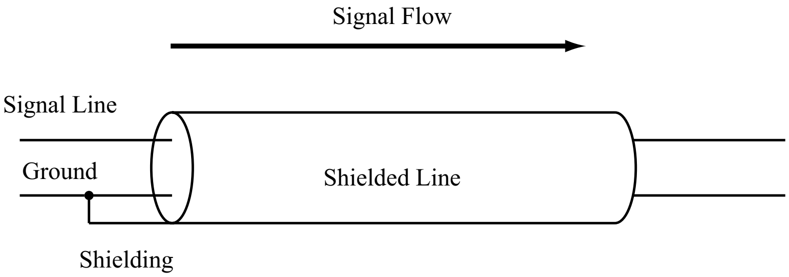


Fig. 13. Shield line connection

All wires are cut as short as possible. Shielded lines are used between (1) the power supply circuit and Hall-effect sensor, (2) between Hall-effect sensor and amplifier circuit, and (3) between amplifier circuit and 6102 A/D converter. In (1), the ground of each output is connected to its shield line at the end closer to the CR200. In (2), each sensor output pair (Hall-effect sensor outputs 5-6 pair and 7-8 pair) has its own shielded cable, and the shield lines are connected to the signal ground at the end closer to the Hall-effect sensor. Similarly, in (3), each sensor signal has its shielded cable, and the shield lines and signal grounds are connected at the end closer to the low-pass filter output.

It is important to note that connecting all of the sensors' grounds amplifies the noise significantly. Each sensor output should have a separate ground that is not connected to any other outputs' grounds. For the same reason, a separate amplifier should be used for each sensor output.

CHAPTER III

MAGNETIC FIELD ANALYSIS

A. Halbach Magnet Matrix

A conceptual construction of the Halbach magnet matrix is shown in Figure 4. The magnet matrix is a superposition of two orthogonal Halbach magnet arrays with orthogonal magnetic fields. Halbach arrays have a stronger fundamental field by a factor of $\sqrt{2}$, which allow a design of a higher power-efficient magnetic device [16, 22]. The magnetic field atop the magnet matrix is periodic, and hence can be appropriately modeled using Fourier series. In this section, a theoretical framework is developed to model the three-dimensional magnetic field generated by the Halbach magnet matrix.

1. Design

The magnetic matrix consists of two kinds of magnets. One is a strong magnet with 90° magnetization (shown with ‘N’ or ‘S’ in Figure 4), where we chose NdFeB50 material which has a remanence of $B_{r\text{strong}} = 1.43$ T. The other magnet used is a weak magnet with magnetization in 45° (shown with arrows). For an ideal Halbach array, a weak magnet will have a remanence of $\frac{B_{r\text{strong}}}{\sqrt{2}} = 1.01$ T. We chose to use NdFeB30 material with a remanence of $B_{r\text{weak}} = 1.10$ T, which was the weakest NdFeB magnet at the time of purchase. The blank areas in the figure are non-magnetic aluminum spacers. The pitch of each magnet array is 50.8 mm (2”), and the dimension of each magnet and spacer is 12.7 mm × 12.7 mm × 12.7 mm (0.5” × 0.5” × 0.5”). The Halbach magnet matrix was constructed by Nikhil Bhat, a former graduate student of Won-jong Kim.

2. Fourier Series Analysis - Strong Side

Since the Halbach magnet matrix has a periodic geometry and magnetic field, it is beneficial to use Fourier series representation. Although the magnet matrix has finite length, this analysis will provide insight on the properties of the magnet matrix. In this thesis, we follow Melcher's notation in Reference [23].

One side of the Halbach magnet matrix will have a strong fundamental field, and the other will be weak. In Figure 4, the top surface is the strong side, which will be facing the actuator coils of the platen. To model the magnetic flux density of the strong side, we first model the magnetization using Fourier series. We consider a function Φ which is dependent sinusoidally on X,

$$\Phi(X,t) = Re\left\{\tilde{\Phi}(t)e^{-jkX}\right\} \tag{3.1}$$

where $k = 2\pi/l$ is the spatial wave number, and l is the spatial wavelength of the magnet array, l = 50.8 mm (2”).

The vertical and lateral magnetization components of the Halbach magnet array are represented by the complex Fourier coefficients [1],

$$\tilde{M}_{Xn} = \frac{1}{l}\int_0^l M_X e^{jk_nX'}dX' \tag{3.2}$$

$$\tilde{M}_{Zn} = \frac{1}{l}\int_0^l M_Z e^{jk_nX'}dX' \tag{3.3}$$

where k_n is the spatial wave number,

$$k_n = \frac{2\pi n}{l}. \tag{3.4}$$

The Fourier coefficients for Halbach array whose magnet has a square cross-section

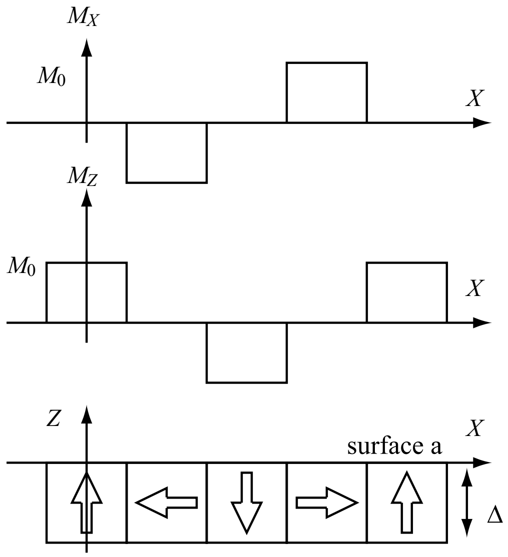


Fig. 14. Halbach magnet array's magnetization (strong side)

as shown in Figure 14 with peak magnetization M_0 are

$$\tilde{M}_{Zn} = \begin{cases} \dfrac{\sqrt{2}M_0}{\pi|n|} & n = \pm(8m+1) \text{ or } \pm(8m+3) \\ -\dfrac{\sqrt{2}M_0}{\pi|n|} & n = \pm(8m+5) \text{ or } \pm(8m+7) \\ 0 & \text{even.} \end{cases} \tag{3.5}$$

Similarly, M_{Xn} and M_{Yn} are

$$\tilde{M}_{Xn} = \tilde{M}_{Yn} = j^n M_{Zn} \cdot \frac{B_{r\text{weak}}}{B_{r\text{strong}}} \tag{3.6}$$

The total magnetization can be written as

$$M_Z = \sum_{n=-\infty}^{\infty} (\tilde{M}_{Zn} e^{-jk_nY} + \tilde{M}_{Zn} e^{-jk_nX}) \tag{3.7}$$

$$M_X = 2\sum_{n=0}^{\infty} (\tilde{M}_{Xn} e^{-jk_nX}) = 2\sum_{n=0}^{\infty} \tilde{M}_{Xn}(\cos k_nX - j\sin k_nX). \tag{3.8}$$

Next, from the magnetization components, we solve for the magnetic flux density at air gap Z_0. Computing for the magnetic flux density using magnetoquasistatic (MQS) approximation yields the following equations [1].

$$\tilde{B}_{Zn} = \frac{\mu_0}{2}\left(j\tilde{M}_{Xn} + \tilde{M}_{Zn}\right)e^{-\gamma_n Z_0}\left(1 - e^{-\gamma_n \Delta}\right) \tag{3.9}$$

$$\tilde{B}_{Xn} = \frac{\mu_0}{2}\left(\tilde{M}_{Xn} - j\tilde{M}_{Zn}\right)e^{-\gamma_n Z_0}\left(1 - e^{-\gamma_n \Delta}\right) \tag{3.10}$$

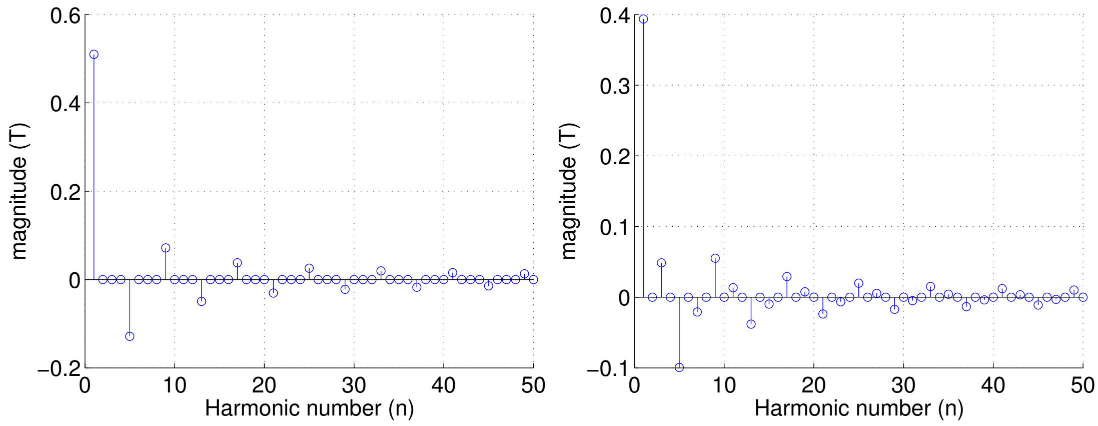
Here, γ_n is the absolute value of the spatial wave number.

Equations (3.9) and (3.10) are the field solutions for the magnet array at a plane Z_0 above the boundary (a) in Figure 14. Substituting equations (3.5) and (3.6) to these equations will yield the Fourier coefficients of the magnetic flux density generated by the magnet array, shown in Figure 15. In the ideal case (as shown in Figure 15 (a)), the generated magnetic field is a superimposition of the fundamental, 5th, 9th, . . . harmonics. The 3rd, 7th, . . . harmonics cancel out. This implies that the Halbach array has a more purely sinusoidal field on its strong side.

The actual model of the Halbach magnet matrix differs from the ideal case, as shown in Figure 15 (b). The 3rd, 7th, . . . harmonics have not canceled out because the materials used do not match the ideal case, that is

$$B_{r\text{weak}} = 1.10 \neq \frac{B_{r\text{strong}}}{\sqrt{2}} = 1.01 \tag{3.11}$$

However, it is important to note here that the fundamental and 5th harmonics are nonzero and are large compared to the other odd harmonics. From this analysis, we chose the harmonics with the largest magnitude as the basis function for the curve-fitting model, i.e. $\sin\omega X, \sin 5\omega X, \ldots$. Hence, the model of the magnetic flux density with the two highest



(a) (b)

Fig. 15. Fourier coefficients of the Halbach magnet array flux density. (a) Ideal array. (b) Modeled array

magnitude harmonics is in one of the following form.

$$\begin{aligned} B_X(X_i) &= c + \alpha\sin(\omega X_i) + \beta\sin(5\omega X_i) \\ B_Y(Y_i) &= c + \alpha\sin(\omega Y_i) + \beta\sin(5\omega Y_i) \end{aligned} \tag{3.12}$$

This is also the model for curve-fitting because Hall-effect sensors are linear sensors. The coefficients c, α, and β are the coefficients to be curve-fitted. The method for curve-fitting is explained in Chapter IV.

From the definition of the Fourier series, the total magnetic flux density due to Halbach magnet matrix on the strong side is

$$B_Z = \sum_{n=0}^{\infty} \tilde{B}_{Zn}\left(e^{-jk_nX} + e^{-jk_nY}\right) \tag{3.13}$$

$$B_X = 2\sum_{n=0}^{\infty} \tilde{B}_{Xn}e^{-jk_nX}. \tag{3.14}$$

The total magnetic flux density at air gap of $Z_0 = 3$ mm is shown in Figure 16.

Next, we compute the relation between the air gap and the maximum magnetic flux density. The results are shown in Figure 17. As expected, the magnitude of the magnetic field decays exponentially. The exponential decay of the fundamental and 5th Fourier coefficients are shown in Figure 18. Considering the previous analysis, the coefficients α and ζ decay exponentially by $e^{-\gamma_1 Z_0}$ whereas β and δ decay by $e^{-\gamma_5 Z_0}$. Therefore, the higher order harmonics decay at a faster rate.

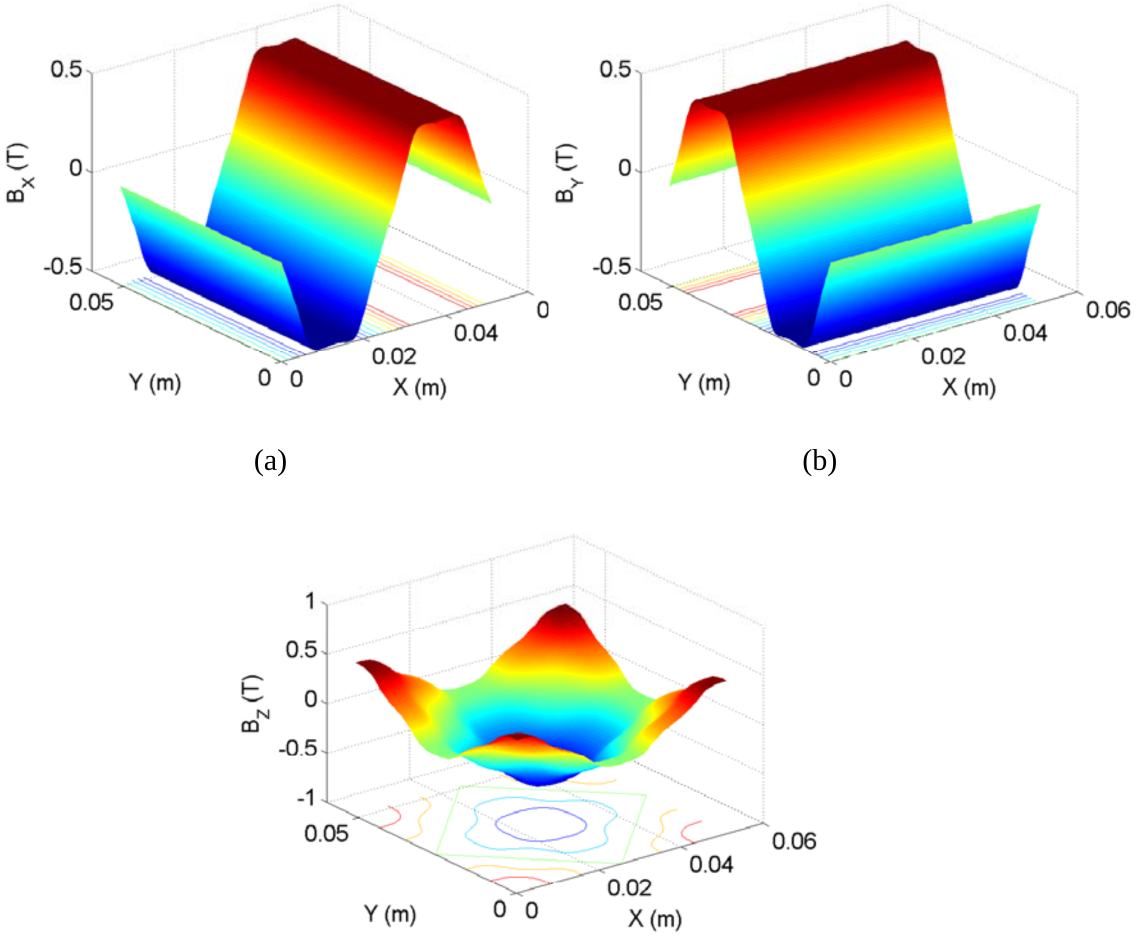
This is important to note for the case of 6-DOF positioning, because for that case, the curve-fitted coefficients of equation (3.12) can be assumed to be in the form,

$$\begin{aligned} \alpha(Z_0) &= \alpha_0 e^{-\gamma_1 Z_0} \\ \beta(Z_0) &= \beta_0 e^{-\gamma_5 Z_0} \end{aligned} \tag{3.15}$$

where c_0, α_0, and β_0 are the coefficients to be curve-fitted.

3. Fourier Series Analysis - Weak Side

The analysis of the weak side is very similar to that of the strong side. In the 2-D case, the only difference between the strong side and the weak side is in the sign of the magnetization M_X. Hence the difference in the model is the sign change for M_X (Figure 19 and Figure 14). The total magnetic flux density of the weak side at air gap of 3 mm is shown in Figure 20. Note that the magnetic flux density is significantly smaller than the strong side as expected.

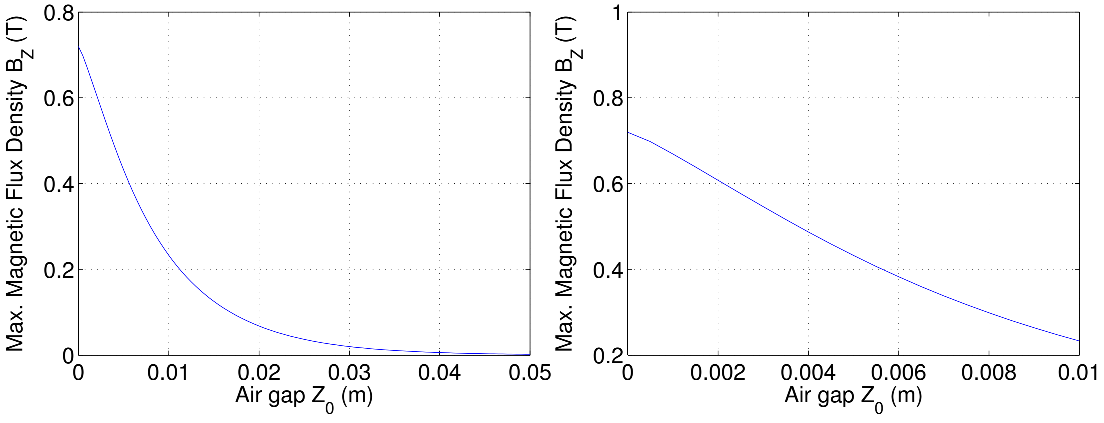


(a)

(b)

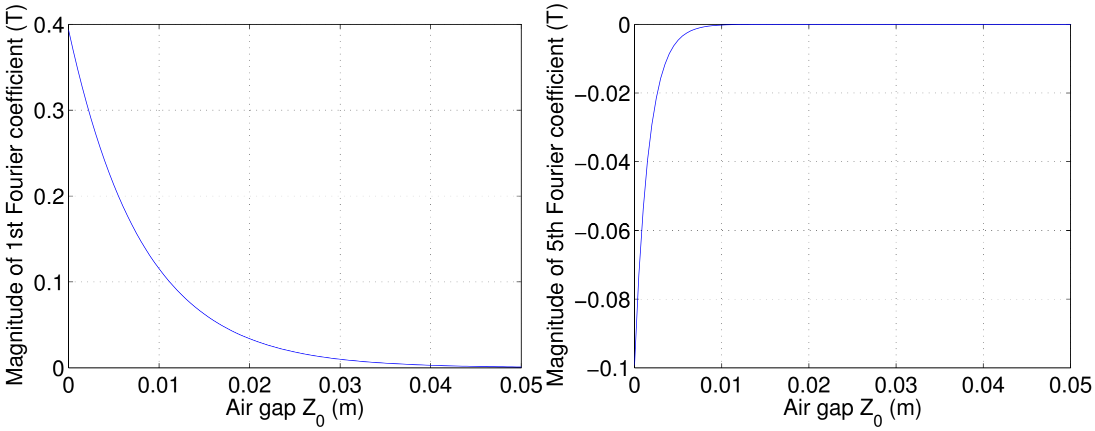
(c)

Fig. 16. Magnetic flux density analysis of Halbach magnet matrix for the strong side, at the air gap of Z_0=3 mm. (a) B_X (b) B_Y (c) B_Z



(a) (b)

Fig. 17. B_Z with respect to air gap. (b) Close-up for small air gap



(a) (b)

Fig. 18. Fourier coefficients with respect to air gap. (a) B_{Z1} (b) B_{Z5}

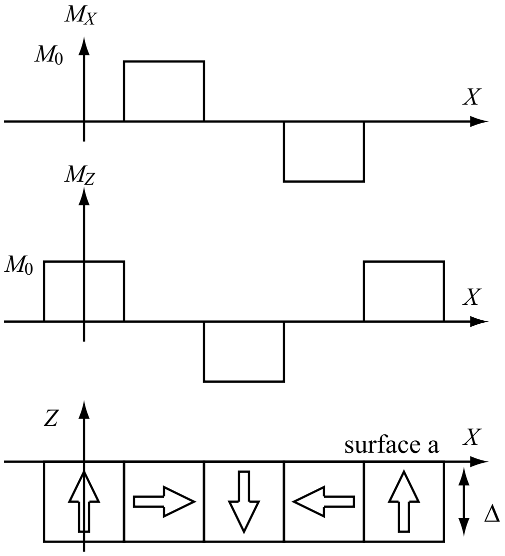
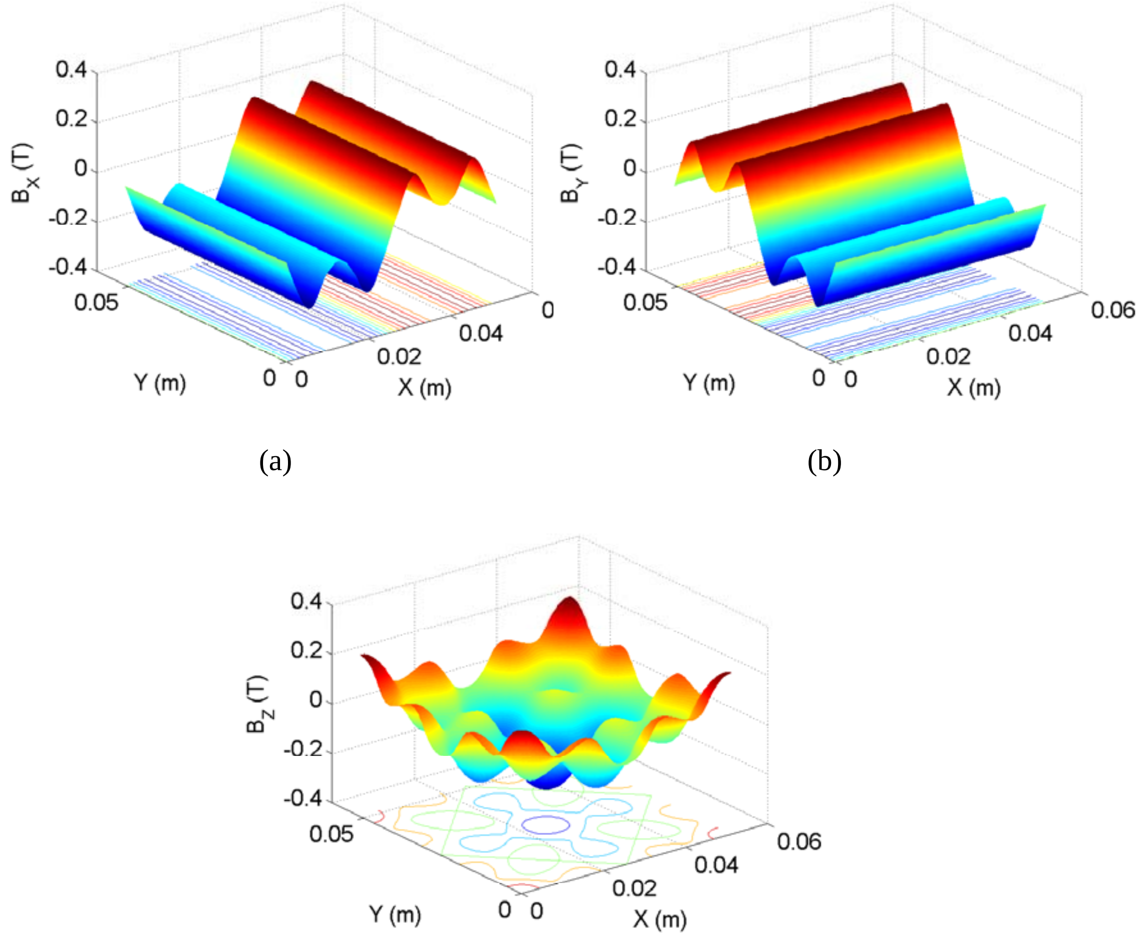


Fig. 19. Halbach magnet array’s magnetization (weak side)



(a)

(b)

(c)

Fig. 20. Magnetic flux density analysis of Halbach magnet matrix for the weak side, at the air gap of Z_0=3 mm. (a) B_X (b) B_Y (c) B_Z

B. Three-Phase Planar Actuator Coils

Although we have not used the sensing mechanism for coils, it can be applied using the same principles. This section derives the model of the magnetic flux density due to multiphase coil actuators. The coils are assumed to be infinitely long and wide to neglect fringing effects. A model of the coil with the dimensions and axes are shown in Figure 21. The model is derived for three-phase winding coil actuators. First, we obtain a model of the current density J. The Fourier coefficients are the following [1].

$$\begin{aligned} J_{yn} &= \frac{1}{l}\int_l J_{sk}(x,t)e^{-jk_nx}dx && (3.16)\\ &= \frac{1}{l}\sum_{k=0}^{2q-1}\int_{\frac{(2k-1)l}{4q}}^{\frac{(2k+1)l}{4q}} J_0\cos\left(\theta(t)+\frac{\pi k}{q}\right)e^{j\frac{2\pi nz}{l}} && (3.17)\\ &= \begin{cases} \dfrac{qJ_0}{n\pi}\sin\left(\dfrac{n\pi}{2q}\right)e^{\mp j\theta(t)} & n = 2mq\pm 1,\quad m = 1,2,\ldots\\ 0 & \text{otherwise}\end{cases} && (3.18)\end{aligned}$$

The magnetic flux density due to the coils is

$$B_{zn} = \frac{j\mu_0}{2k_n}J_{yn}(1-e^{-\gamma_n\Gamma})e^{-\gamma_n Z_0} \qquad (3.19)$$

$$B_{xn} = -\frac{\mu_0}{2k_n}J_{yn}(1-e^{-\gamma_n\Gamma})e^{-\gamma_n Z_0} \qquad (3.20)$$

and

$$\mathbf{B} = \sum_{n=-\infty}^{\infty} B_{xn}e^{-jk_nx}\mathbf{i}_x + B_{zn}e^{-jk_nx}\mathbf{i}_z. \qquad (3.21)$$

This analysis is done for an actuator coil used in our maglev stage. The design parameters have the following values: the pitch is l = 50.8 mm, the winding turn density is $\eta_0 = 3.5246\times 10^6$ turns/m^2, the absolute value of fundamental wave number is $\gamma_1 = \frac{2\pi}{l}$ = 123.7 mm, the winding thickness is $\Gamma = \frac{l}{5}$ = 10.2 mm, and the number of pitches is

$N_m = 2$. The analytical results are shown in Figure 22, where the nominal current density is set as $J_0 = 1.5 \times 10^6$ A/m^2. The Fourier coefficients of the generated magnetic flux density B_{zn} is shown in Figure 23, and the total generated magnetic flux density is shown in Figure 24. In this case, the basis function with only the fundamental wave will be sufficient to model the magnetic flux density. The magnitude of the generated magnetic flux density is small compared to that of the Halbach magnet matrix, on the order of a few milliTeslas.

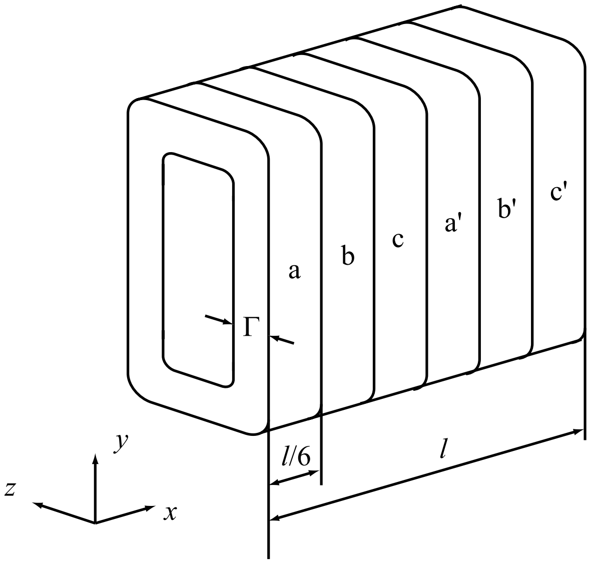


Fig. 21. Actuator coil dimensions of the ATP stage

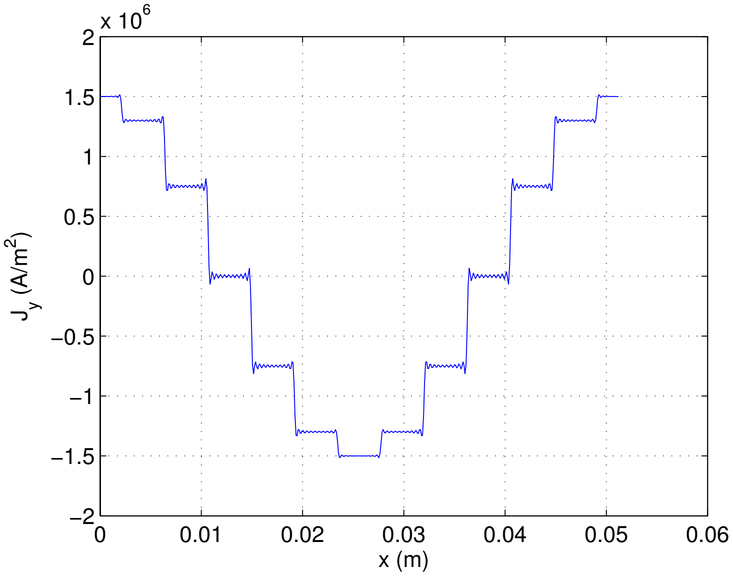


Fig. 22. Current density of coil actuator

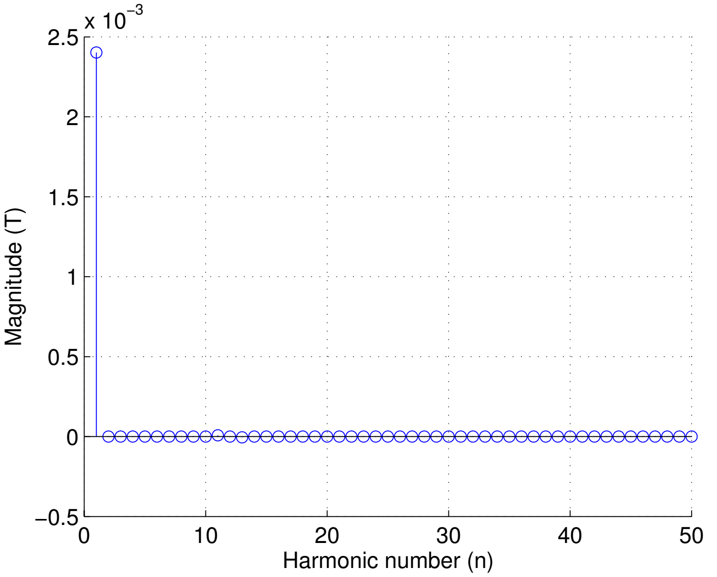
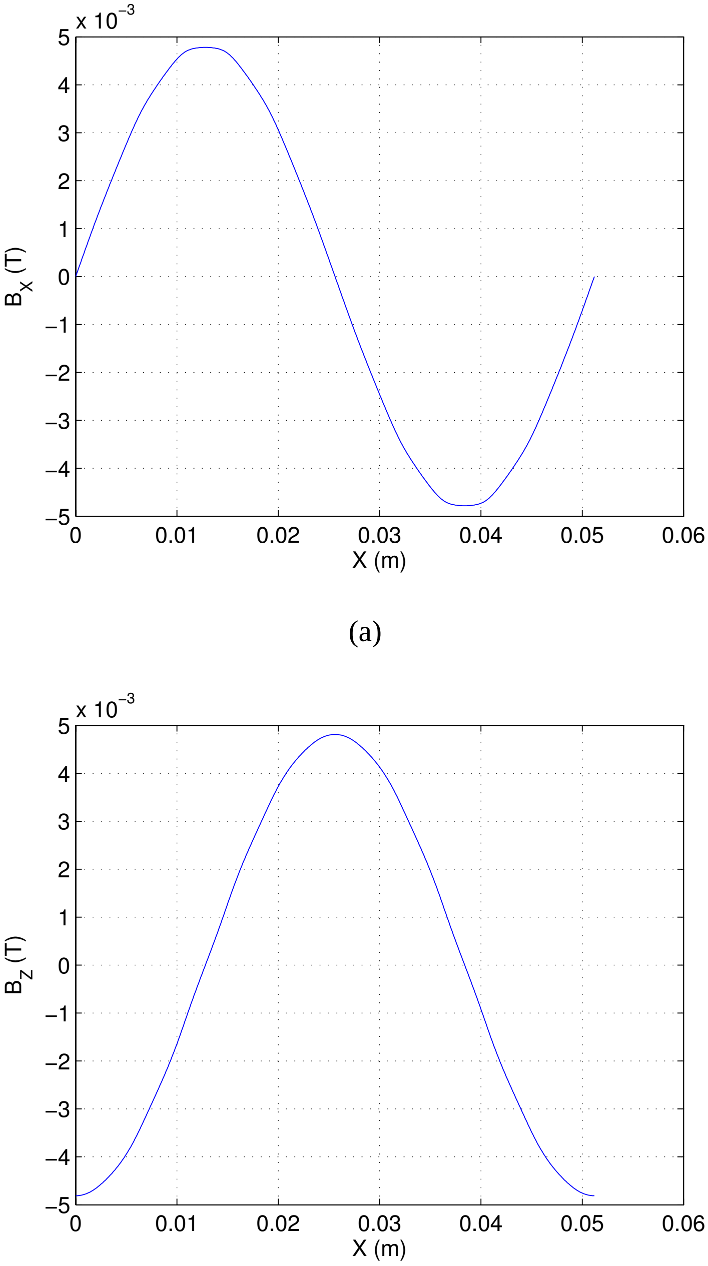


Fig. 23. Fourier coefficients of magnetic flux density due to actuator coil at air gap of 1 mm



(a)

(b)

Fig. 24. Magnetic flux density analysis of coil actuators at air gap of 1 mm. (a) B_X (b) B_Z

CHAPTER IV

ALGORITHM

This chapter explains the algorithm applied to solve for the position of the platen from the Hall-effect sensors' outputs. The algorithm consists of nonlinear least squares, or Gaussian least squares differential correction (GLSDC), followed by a recursive discrete-time Kalman filter (DKF), as shown in Figure 25. Single-input-single-output (SISO) controllers are designed to close the loop using this algorithm.

The GLSDC maps the nonlinear relation between the sensor outputs and the platen's position. The DKF is used to provide a reduced-noise signal from the GLSDC, and also to obtain estimates of the velocity and the angular velocity. These values can be used when designing a multivariable multi-input-multi-output (MIMO) controller.

The steps required to use this algorithm is the following.

1. Obtain sufficient data for each sensor output
2. Curve-fit the data using the least squares method to solve for the coefficients (*a priori*)
3. Apply coefficients in the GLSDC algorithm
4. Run the GLSDC and DKF algorithms in real-time

A. Obtaining Least Squares Model for Sensor Outputs

The outputs of the Hall-effect sensors and the position data from the laser interferometers are obtained using the “snap” function. The data is obtained keeping the platen at a constant air gap by using aerostatic bearings, and the platen is moved in the range of 52 mm by 52 mm. The plots of the sensors' outputs are shown in Figure 26. Four of such sensor output measurements are acquired to obtain the least squares model for each output.

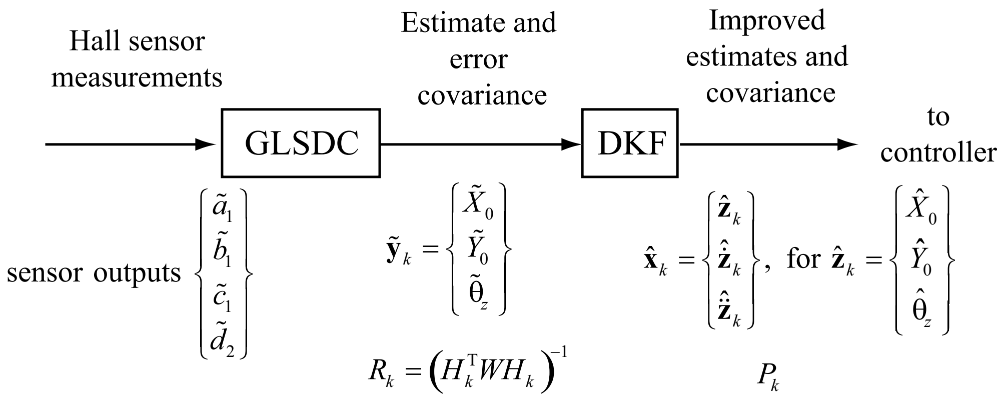

Hall sensor measurements
GLSDC
Estimate and error covariance
DKF
Improved estimates and covariance
to controller
sensor outputs $\begin{Bmatrix} \tilde{a}_1 \\ \tilde{b}_1 \\ \tilde{c}_1 \\ \tilde{d}_2 \end{Bmatrix}$
$\tilde{\mathbf{y}}_k = \begin{Bmatrix} \tilde{X}_0 \\ \tilde{Y}_0 \\ \tilde{\theta}_z \end{Bmatrix}$
$R_k = \left(H_k^{\mathrm{T}} W H_k\right)^{-1}$
$\hat{\mathbf{x}}_k = \begin{Bmatrix} \hat{\mathbf{z}}_k \\ \hat{\dot{\mathbf{z}}}_k \\ \hat{\ddot{\mathbf{z}}}_k \end{Bmatrix}$, for $\hat{\mathbf{z}}_k = \begin{Bmatrix} \hat{X}_0 \\ \hat{Y}_0 \\ \hat{\theta}_z \end{Bmatrix}$
P_k

Fig. 25. GLSDC and DKF series design

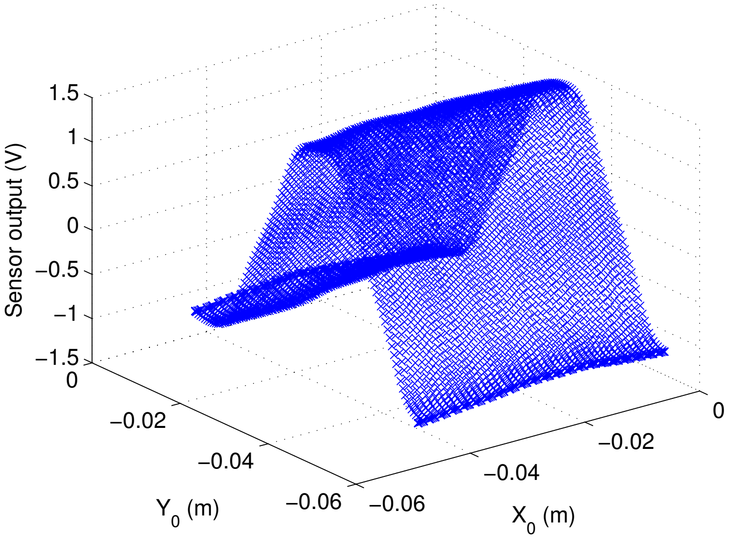


Fig. 26. An example of one Hall-effect sensor output

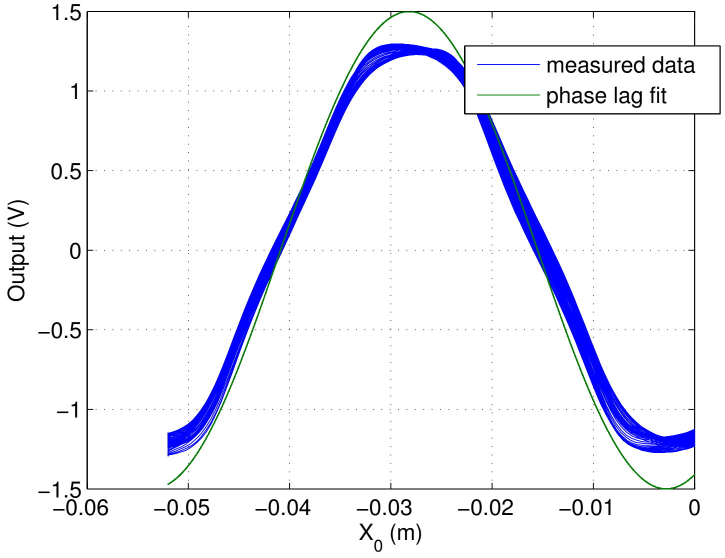


Fig. 27. Fitting the phase lag to determine offsets

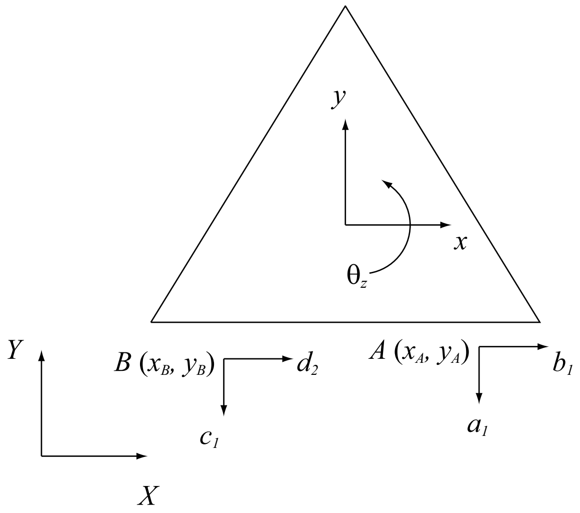


Fig. 28. Definition of sensor locations and axes of measurement

The four outputs of the Hall-effect sensors are acquired by the Pentek 6102 A/D converter board. a_1, b_1, c_1, and d_2 are the sensors' outputs which correspond to the input channels to the Pentek 6102. From the obtained data, we solve for the least squares model using method of batch least squares. Other methods such as linear sequential estimation can also be used.

It was shown in Section III-A that the periodic wave is composed of odd sine harmonics $\sin(\omega X)$, $\sin(5\omega X), \ldots$. Hence, these are chosen as the basis functions for the model along with the DC term, as shown below. For convenience, only up to 5th harmonics are included in the model.

$$\begin{aligned}\hat{a}_{10} &= c_1 + \alpha_1 \sin(\omega(Y_A - Y_{A\text{offset}})) + \beta_1 \sin(5\omega(Y_A - Y_{A\text{offset}})) \\ &\quad + \zeta_1 \sin(\omega(X_A - X_{A\text{offset}})) + \delta_1 \sin(5\omega(X_A - X_{A\text{offset}}))\end{aligned} \tag{4.1}$$

$$\begin{aligned}\hat{b}_{10} &= c_2 + \alpha_2 \sin(\omega(X_A - X_{A\text{offset}})) + \beta_2 \sin(5\omega(X_A - X_{A\text{offset}})) \\ &\quad + \zeta_2 \sin(\omega(Y_A - Y_{A\text{offset}})) + \delta_2 \sin(5\omega(Y_A - Y_{A\text{offset}}))\end{aligned} \tag{4.2}$$

$$\begin{aligned}\hat{c}_{10} &= c_3 + \alpha_3 \sin(\omega(Y_B - Y_{B\text{offset}})) + \beta_3 \sin(5\omega(Y_B - Y_{B\text{offset}})) \\ &\quad + \zeta_3 \sin(\omega(X_B - X_{B\text{offset}})) + \delta_3 \sin(5\omega(X_B - X_{B\text{offset}}))\end{aligned} \tag{4.3}$$

$$\begin{aligned}\hat{d}_{20} &= c_4 + \alpha_4 \sin(\omega(X_B - X_{B\text{offset}})) + \beta_4 \sin(5\omega(X_B - X_{B\text{offset}})) \\ &\quad + \zeta_4 \sin(\omega(Y_A - Y_{B\text{offset}})) + \delta_4 \sin(5\omega(Y_B - Y_{B\text{offset}}))\end{aligned} \tag{4.4}$$

$X_{A\text{offset}}, Y_{A\text{offset}}, X_{B\text{offset}}$, and $Y_{B\text{offset}}$ are offset values to adjust the location of the sensors with respect to the magnetic field's phase. Coupling terms ζ and δ are included to model the axes' misalignment with respect to the *xyz* axes.

The measurements are assumed that there is no rotation about the *z* axis, so the *xyz* and the *XYZ* axes are aligned. The subscript 0 represents that it is in the state with no rotation, and the circumflex (ˆ) represents estimated values from the least squares model. First, the sensors' positions are defined. The position of the sensors in XYZ are

$$X_A = x_A \cos\theta_z - y_A \sin\theta_z \tag{4.5}$$

$$Y_A = x_A \sin\theta_z + y_A \cos\theta_z \tag{4.6}$$

$$X_B = x_B \cos\theta_z - y_B \sin\theta_z \tag{4.7}$$

$$Y_B = x_B \sin\theta_z + y_B \cos\theta_z, \tag{4.8}$$

and since we assume no rotation while obtaining the data, $\theta_z = 0$. The values of $x_A, y_A, \ldots$ are known from the position which the Hall-effect sensors are mounted with respect to the center of mass of the platen (Figure 28).

Next, the offset values are adjusted such that the phase of the sensors' outputs and the model are the same. An example is shown in Figure 27, where the offset is adjusted such that the phase is nearly identical. This can be done by curve-fitting as well, but due to complexity, this value was adjusted manually. It is important that the phase is the same because this leads to large modeling errors and will affect the positioning capability. With the offset values fixed, the coefficients c, α, β, ζ, and δ are obtained using batch least squares, as described below.

The coefficients to be fitted are

$$\mathbf{x}_i = \left\{ \begin{matrix} c_i \\ \alpha_i \\ \beta_i \\ \zeta_i \\ \delta_i \end{matrix} \right\}, \tag{4.9}$$

and assume that we have n data points which we know the position of the platen (X_0 and Y_0 from laser interferometers) and the 4 sensors' outputs. The batch least squares is done for each output as the following. Hence, we have n sets of data $(X_0, Y_0, \tilde{a}_1, \tilde{b}_1\ \tilde{c}_1, \tilde{d}_2)$. To use batch least squares, we rearrange the data into the following form.

$$\tilde{\mathbf{X}} = \begin{Bmatrix} \tilde{X}_{10} \\ \tilde{X}_{20} \\ \vdots \\ \tilde{X}_{n0} \end{Bmatrix}, \quad \tilde{\mathbf{Y}} = \begin{Bmatrix} \tilde{Y}_{10} \\ \tilde{Y}_{20} \\ \vdots \\ \tilde{Y}_{n0} \end{Bmatrix}, \quad \tilde{\mathbf{a}} = \begin{Bmatrix} \tilde{a}_{10} \\ \tilde{a}_{20} \\ \vdots \\ \tilde{a}_{n0} \end{Bmatrix}, \quad \ldots, \tilde{\mathbf{d}} = \begin{Bmatrix} \tilde{d}_{10} \\ \tilde{d}_{20} \\ \vdots \\ \tilde{d}_{n0} \end{Bmatrix}, \tag{4.10}$$

Consider the case to curve-fit output a_1. We solve for the coefficients $\mathbf{x}_{a1}$ from the following equation.

$$\hat{\mathbf{x}}_{a1} = (H^T H)^{-1} H^T \tilde{\mathbf{a}} \tag{4.11}$$

where H is a $n \times m$ matrix and m is the number of basis functions. In this case, $m = 5$, and

$$H = \begin{bmatrix} 1 & \sin(\omega(Y_1 + y_A - Y_{\text{Aoffset}})) & \ldots & \sin(5\omega(X_1 + x_A - X_{\text{Aoffset}})) \\ 1 & \sin(\omega(Y_2 + y_A - Y_{\text{Aoffset}})) & \ldots & \sin(5\omega(X_2 + x_A - X_{\text{Aoffset}})) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \sin(\omega(Y_n + y_A - Y_{\text{Aoffset}})) & \ldots & \sin(5\omega(X_n + x_A - X_{\text{Aoffset}})) \end{bmatrix} \tag{4.12}$$

The same procedure is taken for the other outputs, replacing the offset values and the axes. A curve-fitted model's output for a_1 is shown in Figure 29. It can be seen that the fundamental shape is very similar to the measured data of Figure 26. From the experimental results, adding these terms did not reduce the modeling error, and hence, the coefficients ζ and δ have not been included in the curve-fitted model. The values for coefficients c, α, and β are substituted into the GLSDC model. The modeling error of the curve-fitting is shown in Figure 30. The model has a maximum error of about 200 mV, and some reasons which might cause this error are the following.

- Fabrication error of the Halbach magnet matrix
- Error in material properties of the magnets

- Modeling error of the magnet matrix using batch least squares curve-fitting
- Hall-effect sensor misalignment with respect to the platen's axes, in θ_x, θ_y, and θ_z
- Error in defining the precise location of the sensor (the sensitive volume of the Hall-effect sensor is only 250 μm × 250 μm × 200 μm)
- External magnetic field interference and sensor noise

B. Gaussian Least Squares Differential Correction (GLSDC)

1. GLSDC Problem Formulation

Gaussian least squares differential correction is a widely used successive approximation procedure, otherwise known as nonlinear least squares. An outline of the GLSDC algorithm is shown in Figure 31. At each time step, the input signals $\tilde{\mathbf{y}}_c$ are obtained, and GLSDC algorithm is executed to minimize J, the sum square of the residual errors. J decreases when the GLSDC algorithm converges, and hence an additional loop is added in the routine to check for convergence. This is a modified algorithm of that presented in [24], where α is a scaling factor which is adjusted to minimize the GLSDC output's oscillation. This value will depend on the Hall-effect sensors' noise, and the value is chosen by simulation.

2. Algorithm for 3-DOF Positioning

The objective of this section is to show how to solve for the position of the platen $\tilde{\mathbf{x}}_c = \{X_0,\ Y_0,\ \theta\}^T$ from the sensors' outputs $\mathbf{y}_c = \{a_1,\ b_1,\ c_1,\ d_2\}^T$ using the GLSDC algorithm. The relation between the position and sensor outputs can be written as

$$\tilde{\mathbf{y}}_c = H\mathbf{x_c}, \tag{4.13}$$

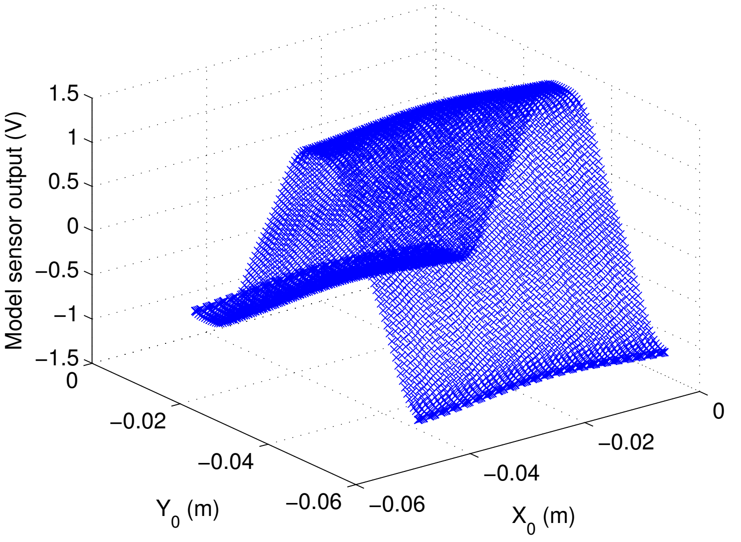


Fig. 29. Output of the curve-fitted model

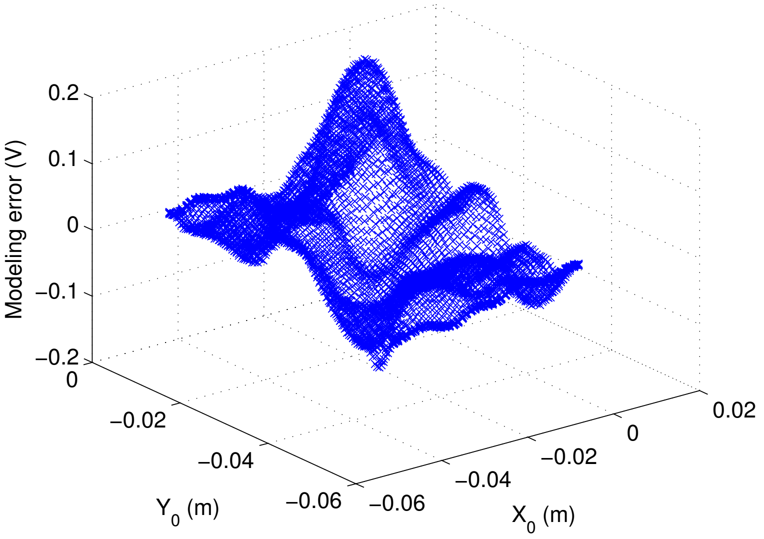


Fig. 30. Modeling error from curve-fitting

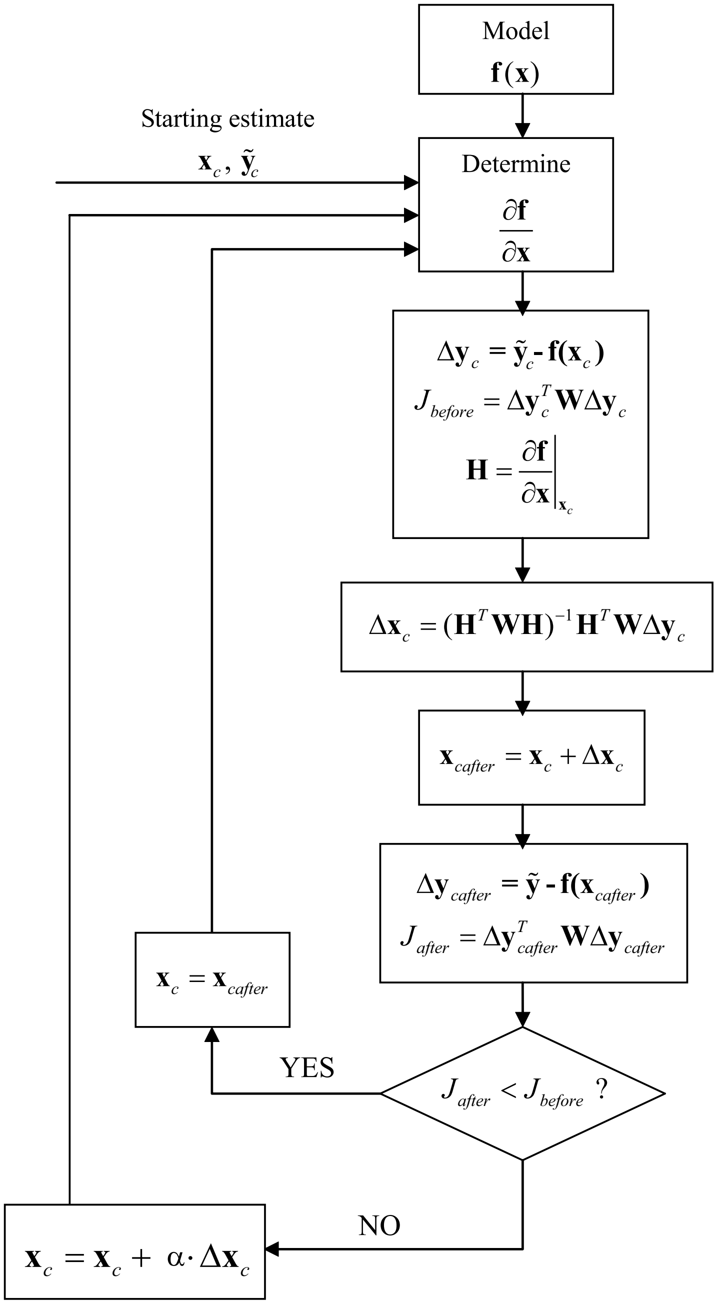

Model
$\mathbf{f}(\mathbf{x})$
Starting estimate
$\mathbf{x}_c, \tilde{\mathbf{y}}_c$
Determine
$\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$
$\Delta \mathbf{y}_c = \tilde{\mathbf{y}}_c - \mathbf{f}(\mathbf{x}_c)$
$J_{before} = \Delta \mathbf{y}_c^T \mathbf{W} \Delta \mathbf{y}_c$
$\mathbf{H} = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_c}$
$\Delta \mathbf{x}_c = (\mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W} \Delta \mathbf{y}_c$
$\mathbf{x}_{cafter} = \mathbf{x}_c + \Delta \mathbf{x}_c$
$\Delta \mathbf{y}_{cafter} = \tilde{\mathbf{y}} - \mathbf{f}(\mathbf{x}_{cafter})$
$J_{after} = \Delta \mathbf{y}_{cafter}^T \mathbf{W} \Delta \mathbf{y}_{cafter}$
$\mathbf{x}_c = \mathbf{x}_{cafter}$
YES
$J_{after} < J_{before}$?
NO
$\mathbf{x}_c = \mathbf{x}_c + \alpha \cdot \Delta \mathbf{x}_c$

Fig. 31. Gaussian least squares differential correction algorithm [24]

where H is the Jacobian matrix and

$$\hat{\mathbf{x}} = \begin{Bmatrix} \hat{X}_0 \\ \hat{Y}_0 \\ \hat{\theta}_z \end{Bmatrix}, \quad \tilde{\mathbf{y}}_c = \begin{Bmatrix} \tilde{a}_1 \\ \tilde{b}_1 \\ \tilde{c}_1 \\ \tilde{d}_2 \end{Bmatrix}. \tag{4.14}$$

The tilde (˜) represents measured values. The sensors are fixed to the platen, and their axes are aligned with the *xyz* axes, as shown in Figure 28.

From the curve-fitted model, we know the estimate values when no rotation is considered.

$$\hat{a}_{10} = c_1 + \alpha_1 \sin(\omega(X_A - X_{A\text{offset}})) + \beta_1 \sin(5\omega(X_A - X_{A\text{offset}})) \tag{4.15}$$

$$\hat{b}_{10} = c_2 + \alpha_2 \sin(\omega(Y_A - Y_{A\text{offset}})) + \beta_2 \sin(5\omega(Y_A - Y_{A\text{offset}})) \tag{4.16}$$

$$\hat{c}_{10} = c_3 + \alpha_3 \sin(\omega(X_B - X_{B\text{offset}})) + \beta_3 \sin(5\omega(X_B - X_{B\text{offset}})) \tag{4.17}$$

$$\hat{d}_{20} = c_4 + \alpha_4 \sin(\omega(Y_B - Y_{B\text{offset}})) + \beta_4 \sin(5\omega(Y_B - Y_{B\text{offset}})), \tag{4.18}$$

where, for the general case,

$$X_A = x_A \cos\theta_z - y_A \sin\theta_z \tag{4.19}$$

$$Y_A = x_A \sin\theta_z + y_A \cos\theta_z \tag{4.20}$$

$$X_B = x_B \cos\theta_z - y_B \sin\theta_z \tag{4.21}$$

$$Y_B = x_B \sin\theta_z + y_B \cos\theta_z. \tag{4.22}$$

Next, compute the estimates with rotation θ_z considered.

$$\hat{a}_1 = \hat{a}_{10} \cos\theta_z + \hat{b}_{10} \sin\theta_z \tag{4.23}$$

$$\hat{b}_1 = -\hat{a}_{10} \sin\theta_z + \hat{b}_{10} \cos\theta_z \tag{4.24}$$

$$\hat{c}_1 = \hat{c}_{10} \cos\theta_z + \hat{d}_{20} \sin\theta_z \tag{4.25}$$

$$\hat{d}_2 = -\hat{c}_{10}\sin\theta_z + \hat{d}_{20}\cos\theta_z \tag{4.26}$$

From these relations, the Jacobian matrix H is obtained as the following.

$$H = \begin{bmatrix} \dfrac{\partial a_1}{\partial X_0} & \dfrac{\partial a_1}{\partial Y_0} & \dfrac{\partial a_1}{\partial \theta_z} \\ \dfrac{\partial b_1}{\partial X_0} & \dfrac{\partial b_1}{\partial Y_0} & \dfrac{\partial b_1}{\partial \theta_z} \\ \dfrac{\partial c_1}{\partial X_0} & \dfrac{\partial c_1}{\partial Y_0} & \dfrac{\partial c_1}{\partial \theta_z} \\ \dfrac{\partial d_2}{\partial X_0} & \dfrac{\partial d_2}{\partial Y_0} & \dfrac{\partial d_2}{\partial \theta_z} \end{bmatrix} \tag{4.27}$$

From this algorithm, we also obtain the error covariance matrix, defined in Reference [24] as

$$R_k = ({H_k}^T W H_k)^{-1} \tag{4.28}$$

This becomes useful in the DKF design when defining the error covariance of the GLSDC output.

For our experimental setup, the sensor noise's standard deviation is approximately 1 mV, and selecting the scaling factor α in Figure 31 as 0.2 resulted in best convergence.

An advantage of using the GLSDC algorithm instead of neural networks (NN) is that this utilizes the periodicity of the magnet matrix, and the output converges to the local minimum, depending on the position of the platen. For example, suppose that the platen is located at $(X_0, Y_0) = (0, 0)$ and is given an input of (−1 mm, −1 mm). When NN is used, since it does not recognize the periodicity, it may recognize the reference input as (−49.8 mm, −49.8 mm). However, GLSDC will not do this because it understands that the model is continuous and periodic. Hence, this algorithm has unrestricted travel range.

This algorithm can easily be extended in case with redundant Hall-effect sensors. When having redundant sensors, each sensor output must be curve-fitted using appropriate basis functions. For n number of sensors to position in 3 DOFs, the Jacobian matrix H will be a $2n \times 3$ matrix. In the case when rotation about z is negligible and not considered, the model is simplified and H matrix becomes a 4 × 2 matrix,

$$H = \begin{bmatrix} \frac{\partial a_1}{\partial X_0} & \frac{\partial a_1}{\partial Y_0} \\ \frac{\partial b_1}{\partial X_0} & \frac{\partial b_1}{\partial Y_0} \\ \frac{\partial c_1}{\partial X_0} & \frac{\partial c_1}{\partial Y_0} \\ \frac{\partial d_2}{\partial X_0} & \frac{\partial d_2}{\partial Y_0} \end{bmatrix}. \tag{4.29}$$

Since rotation θ_z is not considered, $\theta_z = 0$ and

$$a_{10} = a_1, \;\; b_{10} = b_1, \;\; c_{10} = c_1, \;\; d_{20} = d_2. \tag{4.30}$$

This can be used when laser interferometers are used for θ_z control, assuming negligible rotation, and will result in better convergence.

3. Algorithm for 6-DOF Positioning

The same algorithm can be extended to 6-DOF positioning. To do this, we must first obtain the coefficients of the curve-fitting as a function of the air gap Z_0. For 6-DOF positioning, at least three sets of two-axis Hall-effect sensors are necessary, as shown in Figure 6 (b). For generalization, consider using n number of sensors and $2n$ number of sensor outputs $(a_1, b_1, \ldots, q)$. This is possible because the Z axis is orthogonal to the X and Y axes and has independent basis functions. As explained in the analysis in Chapter III, the Fourier coefficients decay exponentially with respect to the air gap. Hence, for one sensor output, the coefficients will be fitted as

$$\begin{aligned} \alpha_1(Z_0) &= \alpha_{10} e^{-\gamma_1 Z_0} \\ \beta_1(Z_0) &= \beta_{10} e^{-\gamma_5 Z_0} \\ \zeta_1(Z_0) &= \zeta_{10} e^{-\gamma_1 Z_0} \\ \delta_1(Z_0) &= \delta_{10} e^{-\gamma_5 Z_0}, \end{aligned} \tag{4.31}$$

and the values for $\alpha_{10}, \beta_{10}, \zeta_{10}$, and δ_{10} will be solved for by curve-fitting methods such as batch least squares. In this case, it is necessary to fit the value of Z_0, similar to the way the offsets $X_{A\text{offset}}, Y_{A\text{offset}}, X_{B\text{offset}}$, and $Y_{B\text{offset}}$ are decided.

For 6-DOF positioning, it may be better to obtain measurements of B_Z, in which case the curve-fitted model will be the following,

$$B_Z(X_i, Y_i, Z_i) = c + \Big(\alpha_x \sin(\omega X_i) + \beta_x \sin(5\omega X_i)\Big) \cdot \Big(\alpha_y \sin(\omega Y_i) + \beta_y \sin(5\omega Y_i)\Big), \tag{4.32}$$

where $\alpha_x, \alpha_y, \beta_x$, and β_y are functions of the air gap Z_0.

To position in 6 DOFs ($X_0, Y_0, Z_0, \theta_x, \theta_y$, and θ_z), the Jacobian matrix will be a $2n \times 6$ matrix,

$$H = \begin{bmatrix} \frac{\partial a_1}{\partial X_0} & \frac{\partial a_1}{\partial Y_0} & \frac{\partial a_1}{\partial Z_0} & \frac{\partial a_1}{\partial \theta_x} & \frac{\partial a_1}{\partial \theta_y} & \frac{\partial a_1}{\partial \theta_z} \\ \frac{\partial b_1}{\partial X_0} & \frac{\partial b_1}{\partial Y_0} & \frac{\partial b_1}{\partial Z_0} & \frac{\partial b_1}{\partial \theta_x} & \frac{\partial b_1}{\partial \theta_y} & \frac{\partial b_1}{\partial \theta_z} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial q}{\partial X_0} & \frac{\partial q}{\partial Y_0} & \frac{\partial q}{\partial Z_0} & \frac{\partial q}{\partial \theta_x} & \frac{\partial q}{\partial \theta_y} & \frac{\partial q}{\partial \theta_z}. \end{bmatrix} \tag{4.33}$$

As we have done for the 3-DOF case, measurements will be taken for the case without rotation ($\hat{a}_{10}, \ldots, \hat{q}_0$), for various air gaps. The derived model will involve a rotation matrix which will incorporate the rotation in θ_x, θ_y, and θ_z.

C. Recursive Discrete-Time Kalman Filter (DKF)

A recursive discrete-time Kalman filter (DKF) has been designed to filter out the noise from the GLSDC, and to provide estimates of the velocity and angular velocity which will become useful when designing a multivariable controller. I first tried to close the 3-DOF motion using only GLSDC, but found that the GLSDC output had large variation, which I assumed was the reason why I could not close all 3 DOFs using two sets of two-axis Hall-effect sensors. Description of the DKF design is presented below.

The states are defined as

$$\mathbf{x}_k = \begin{Bmatrix} \mathbf{z}_k \\ \dot{\mathbf{z}}_k \\ \ddot{\mathbf{z}}_k \end{Bmatrix}, \quad \text{for} \;\; \mathbf{z}_k = \begin{Bmatrix} X_0 \\ Y_0 \\ \theta_z \end{Bmatrix}. \tag{4.34}$$

For continuous time, the dynamic model and measurement (estimated) model are

$$\dot{\mathbf{x}}(t) = \begin{Bmatrix} \dot{\mathbf{z}}(t) \\ \ddot{\mathbf{z}}(t) \\ \dddot{\mathbf{z}}(t) \end{Bmatrix} = \begin{bmatrix} 0 & I_{3\times3} & 0 \\ 0 & 0 & I_{3\times3} \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{z}(t) \\ \dot{\mathbf{z}}(t) \\ \ddot{\mathbf{z}}(t) \end{Bmatrix} + \begin{bmatrix} 0 \\ 0 \\ I_{3\times3} \end{bmatrix} \mathbf{w}(t) \tag{4.35}$$

$$\tilde{\mathbf{y}}(t) = \hat{\mathbf{z}}(t) + \mathbf{v}(t) = \begin{bmatrix} I_{3\times3} & 0 & 0 \end{bmatrix} \mathbf{x}(t) + \mathbf{v}(t). \tag{4.36}$$

As shown in Figure 25, the inputs to the DKF are the outputs of the GLSDC,

$$\tilde{\mathbf{y}}(t) = \begin{Bmatrix} \tilde{X}_0 \\ \tilde{Y}_0 \\ \tilde{\theta}_z \end{Bmatrix}, \tag{4.37}$$

and are not the Hall-effect sensors' output. $\mathbf{v}$ is a 3×1 vector and is the noise which results from the GLSDC. It is assumed to be zero-mean Gaussian noise,

$$\mathbf{v}(t) \sim N(0, R_k), \tag{4.38}$$

where R_k is the error covariance matrix defined in equation (4.28).

The model assumes constant acceleration. The relationships between position, velocity, and acceleration are modeled exactly without any modeling error. Note that the process noise $\mathbf{w}$ is a 3×1 vector which has the same unit as the jerk and does not depend on the error due to GLSDC. Further, $\mathbf{w}$ is assumed to be a zero-mean Gaussian process, and selecting the bounds to this process noise becomes the parameter to tune the DKF.

The constant-acceleration assumption is merely for simple modeling purpose. Although this may not be a valid assumption, the output shows that the noise is reduced and the accuracy is improved. If more accurate estimates of velocity and acceleration are required, we can add the process noise to the time derivative of the jerk. This would mean the model will assume constant jerk, and the computation will be more complex and require more time.

In discrete time, the model becomes the following.

$$\begin{aligned}\mathbf{x}_{k+1} &= \begin{bmatrix} I_{3\times3} & (t_{k+1}-t_k)I_{3\times3} & \frac{1}{2}(t_{k+1}-t_k)^2 I_{3\times3} \\ 0 & I_{3\times3} & (t_{k+1}-t_k)I_{3\times3} \\ 0 & 0 & I_{3\times3} \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \frac{1}{6}(t_{k+1}-t_k)^3 I_{3\times3} \\ \frac{1}{2}(t_{k+1}-t_k)^2 I_{3\times3} \\ (t_{k+1}-t_k)I_{3\times3} \end{bmatrix} \mathbf{w}_k \\ &= \Phi\mathbf{x}_k + \Upsilon\mathbf{w}_k \end{aligned} \tag{4.39}$$

$$\tilde{\mathbf{y}}_k = \begin{bmatrix} I_{3\times3} & 0 & 0 \end{bmatrix} \mathbf{x}_k + \mathbf{v}_k = H\mathbf{x}_k + \mathbf{v}_k \tag{4.40}$$

where $(t_{k+1} - t_k)$ is the integration step size, which is constant for simulation and real-time control, and $\mathbf{v}_k$ is the measurement noise. The DKF formulation is presented in Table III, as given in [24].

The Q_k is the error covariance matrix of the process noise $\mathbf{w}$, which acts on the acceleration. This becomes the design parameter to tune the DKF by selecting a reasonable value for the acceleration noise. Q_k is chosen to be a positive constant diagonal matrix,

$$Q_k = \begin{bmatrix} q_x & 0 & 0 \\ 0 & q_y & 0 \\ 0 & 0 & q_\theta \end{bmatrix}, \tag{4.41}$$

where an assumption is made that $\mathbf{v}_k$ and $\mathbf{w}_k$ are uncorrelated, and the values of q_x, q_y, and q_θ are set by the designer. Choosing these values is an ad hoc approach, similar to selecting

Table III. Recursive discrete-time Kalman filter

MODEL	$\mathbf{x}_{k+1} = \Phi\mathbf{x}_k + \Upsilon\mathbf{w}_k\ ,\ \ \mathbf{w}_k \sim N(0, Q_k)$ $\tilde{\mathbf{y}}_k = H\mathbf{x}_k + \mathbf{v}_k\ ,\ \ \mathbf{v}_k \sim N(0, R_k)$
INITIALIZE	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0\ ,\ \ P_0 = E\{\hat{\mathbf{x}}(t_0)\hat{\mathbf{x}}(t_0)^T\}$
GAIN	$K_k = P_k^- H_k^T [H_k P_k H_k^T + R_k]^{-1}$
UPDATE	$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k[\tilde{\mathbf{y}}_k - H_k\hat{\mathbf{x}}_k^-]$ $P_k^+ = [I - K_k H_k]P_k^-$
PROPAGATION	$\hat{\mathbf{x}}_{k+1}^- = \Phi\hat{\mathbf{x}}_k^+$ $P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \Upsilon_k Q_k \Upsilon_k^T$

the bandwidth of a filter, where we iterate the design until a satisfactory performance level is achieved. The initial estimates are set to

$$\hat{\mathbf{x}}_0 = \begin{Bmatrix} \hat{\mathbf{z}}_0 \\ \mathbf{0}_{6\times1} \end{Bmatrix}, \quad \hat{\mathbf{z}}_0 = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \tag{4.42}$$

where $\hat{\mathbf{z}}_0$ is the same as the initial guess used for the GLSDC. The initial estimation error covariance matrix is specified by

$$P_0 = pI_{9\times9}. \tag{4.43}$$

The initial value of p is set by the designer to obtain a good convergence behavior of the filter. The DKF updates the P matrix at each time step, and hence the P matrix will converge if appropriate initial values are chosen. Simulation results with changing these values are shown in Chapter V.

Instead of using a recursive DKF, some digital filters such as low-pass filters were tried in both simulation and experiment, but this resulted in large oscillation at the outputs. The reason for this seems to be caused by the delay and the dynamics of the filter. There may be a way to make this work using low-order digital filters. The DKF output did not show this behavior, but the dynamics of the filter was too slow to stabilize the 3-DOF motion.

D. Calibration Method

Calibration is done for the translational motion in X and Y when considering negligible rotation in θ_z. The position error for discrete points is tabulated so that it can be interpolated to find the position error.

Although this will reduce the translational positioning error, it cannot compensate for the error when rotation is considered. In that case, the only valid method will be obtaining a more precise model and using more number of sensors. This will make the GLSDC output more likely to converge to the correct value. To obtain a more precise model, other basis functions can be incorporated which will model misalignments with respect to other axes. As of now, the coefficients ζ and δ only compensate for the misalignment in the z axis.

E. Dynamical Modeling and Control System Design

In this section, we present the dynamics and control of the maglev stage. Section 1 gives the dynamic model of the moving part (namely, the platen). We only consider the dynamic model for planar motion X, Y, and θ_z. Section 2 gives the control system designs for planar motions.

1. Dynamic Model

The moving part of the system is modeled as a pure mass because there is no friction in this maglev system. The total mass of the platen includes the platen, mirrors, actuator coils, cables, screws, and the Hall-effect sensor setup, and is measured to be 6.20 kg. Therefore, for translational motion, assuming that there are is no dynamic coupling, the system is represented by the differential equation (4.44).

$$6.20\frac{d^2x}{dt^2} = f_x \tag{4.44}$$

The system transfer function is

$$\frac{X(s)}{F_x(s)} = \frac{1}{6.20s^2}. \tag{4.45}$$

For rotation, the system is represented by

$$I_z\frac{d^2\theta_z}{dt^2} = T_z, \tag{4.46}$$

where $I_z = 0.054$ kg·m^2 is the moment of inertia around the z axis and T_z is the torque around the z axis. The value of the moment of inertia is obtained by using SolidWorks[9]. The system transfer function is

$$\frac{\Theta_z(s)}{T_z(s)} = \frac{1}{I_z s^2}. \tag{4.47}$$

2. SISO Controller Design

SISO controllers are designed to control the platen using sensor readings from the GLSDC and DKF algorithms. The rule of thumb when designing a controllers is that the dominant poles of the filter (DKF) must be 6 to 10 times faster than those of the controller. Controller are designed with crossover frequency of 20 Hz for X and Y controllers, and 10 Hz for θ_z controller. However, this puts a very severe restriction in the design of the DKF, and the DKF with such high bandwidth did not attenuate the noise, but has large overshoot.

A lead-lag controller is designed at various crossover frequencies. Controllers designed for translational motion (X and Y) with a crossover frequency of 10 Hz and controller for rotational motion (θ_z) with a crossover frequency of 20 Hz are presented.

---

[9] SolidWorks Corporation, 300 Baker Avenue, Concord, MA 01742

First, the controllers $D(s)$ are designed in the continuous-time domain, using the Matlab's 'sisotool' function, a product of Mathworks[10]. They are the controllers designed for 800-Hz sampling and are obtained by discretizing the controllers designed in the s-domain using the matched pole-zero (MPZ) method.

The lead-lag controllers for X and Y are described below. The controller's bandwidth is 9.90 Hz and phase margin is 70.1°. The root-locus plot and the open-loop Bode plot are shown in Figure 32. The step response to an ideal plant with ideal measurements is shown in Figure 33.

$$D(s) = 4.09 \times 10^5 \cdot \frac{(s + 8.31)(s + 10)}{s(s + 1134)} \tag{4.48}$$

$$D(z) = 2.236 \times 10^5 \cdot \frac{z^2 - 1.9773z + 0.977428}{z^2 - 1.2423z + 0.2423} \tag{4.49}$$

The lead-lag controller for θ_z is presented below. The controller's bandwidth is 19.7 Hz and phase margin is 75.3°. The root locus plot is shown in Figure 34, and a response to a 1° step input is shown in Figure 35.

$$D(s) = 7.6 \times 10^3 \cdot \frac{(s + 8.31)(s + 10)}{s(s + 1134)} \tag{4.50}$$

$$D(z) = 4.155 \times 10^3 \cdot \frac{z^2 - 1.9773z + 0.977428}{z^2 - 1.2423z + 0.2423} \tag{4.51}$$

These controllers have been tested on our setup using the GLSDC algorithm. Experimental results are presented in Chapter V.

---

[10] The Mathworks, Inc. 3 Apple Hill Drive, Natick, MA 01760-2098

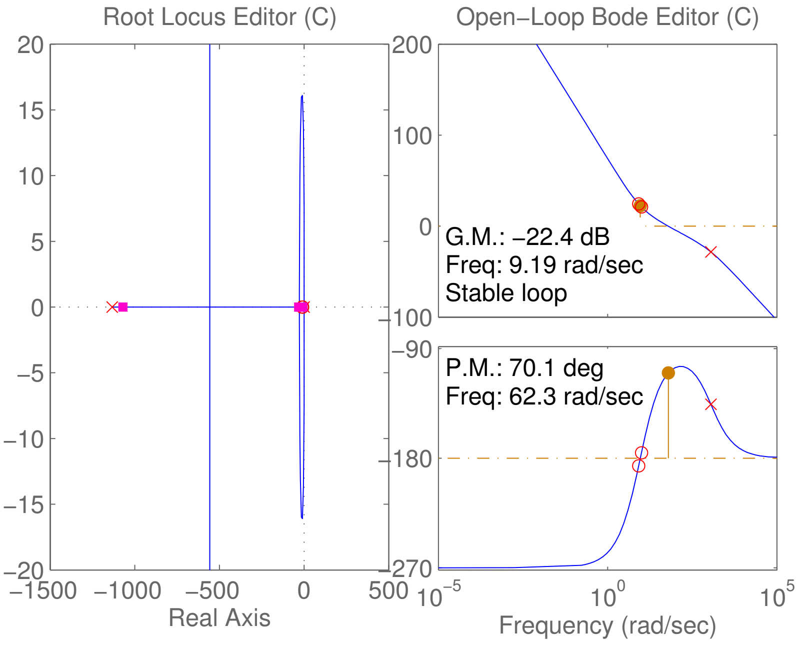


Fig. 32. 10-Hz controller for translational motion in *X* and *Y*

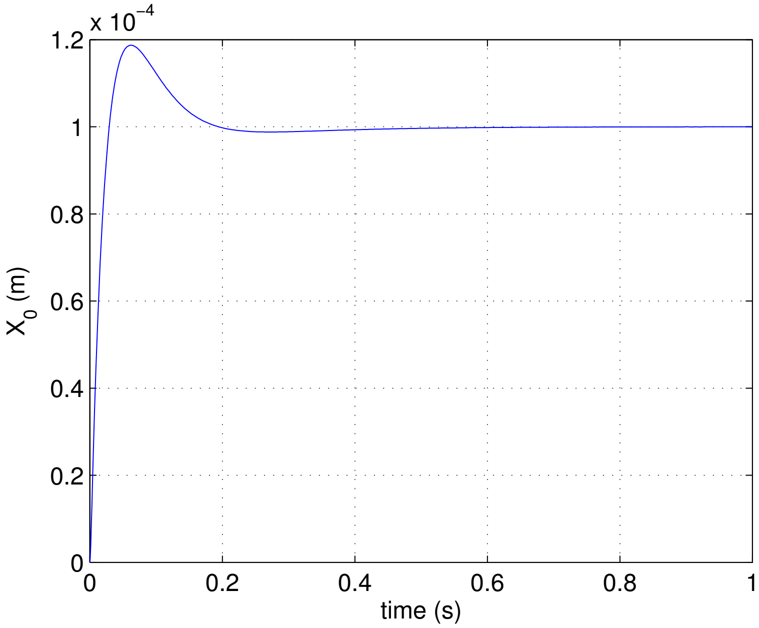


Fig. 33. Step response to a 100-μm step in *X*

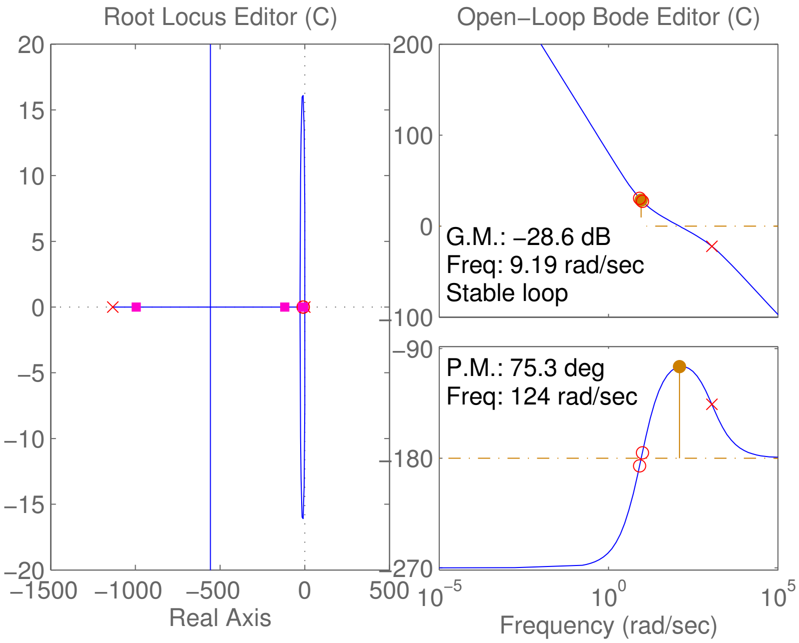


Fig. 34. 20-Hz controller for rotational motion in θ_z

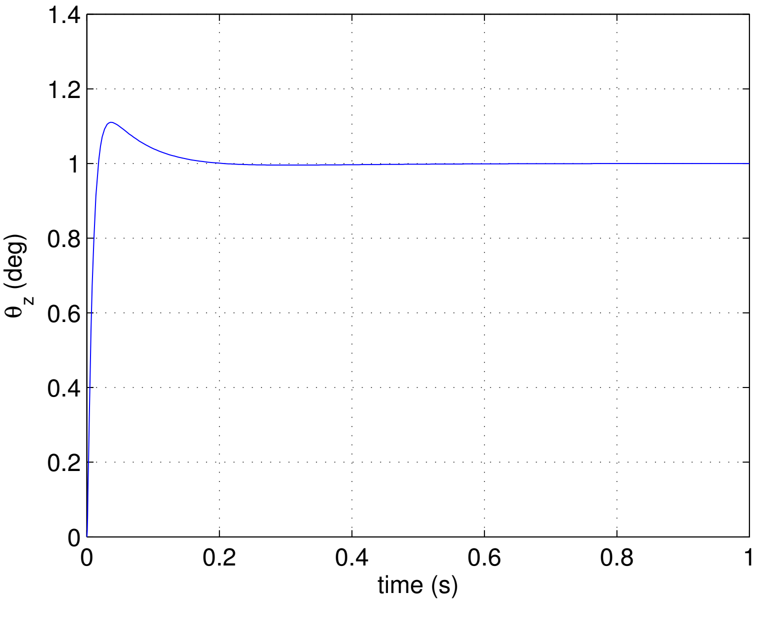


Fig. 35. Step response to a 1° step in θ_z

CHAPTER V

RESULTS

This chapter presents the simulation and experimental results. Due to the performance of the Pentek 4284 DSP, the computation of each iteration is completed in 1.25 ms, or at the sampling frequency of 800 Hz.

The simulation results are done assuming the use of two sets of two-axis Hall-effect sensors, as shown in Figure 28.

A. Simulation Results

1. GLSDC Results with No Plant

This section shows the GLSDC response to a step input without considering the plant's dynamics. The sensor noise is approximately 1 mV, and choosing $\alpha = 0.2$ resulted in the best convergence and small oscillation in the output. The initial states are set as $\{X_0, Y_0, \theta_z\} = \{0, 0, 0\}$, and no sensor noise is considered. At $t = 0$ s, or at the first time step, the estimated ideal sensor output value for $\{X_0, Y_0, \theta_z\} = \{0, 200\ \mu\text{m}, 0\}$ is given. The results are shown in Figure 36. From Figure 36 (d), it can be seen that the sum of squares error J takes approximately three steps to converge. Although the step input was in Y, there are small changes shown in the outputs for X and θ_z (Figures 36 (b) and (c)), which show that this is a coupled nonlinear relation. The same behavior can be seen when applying step inputs for θ_z, as shown later in Figure 38.

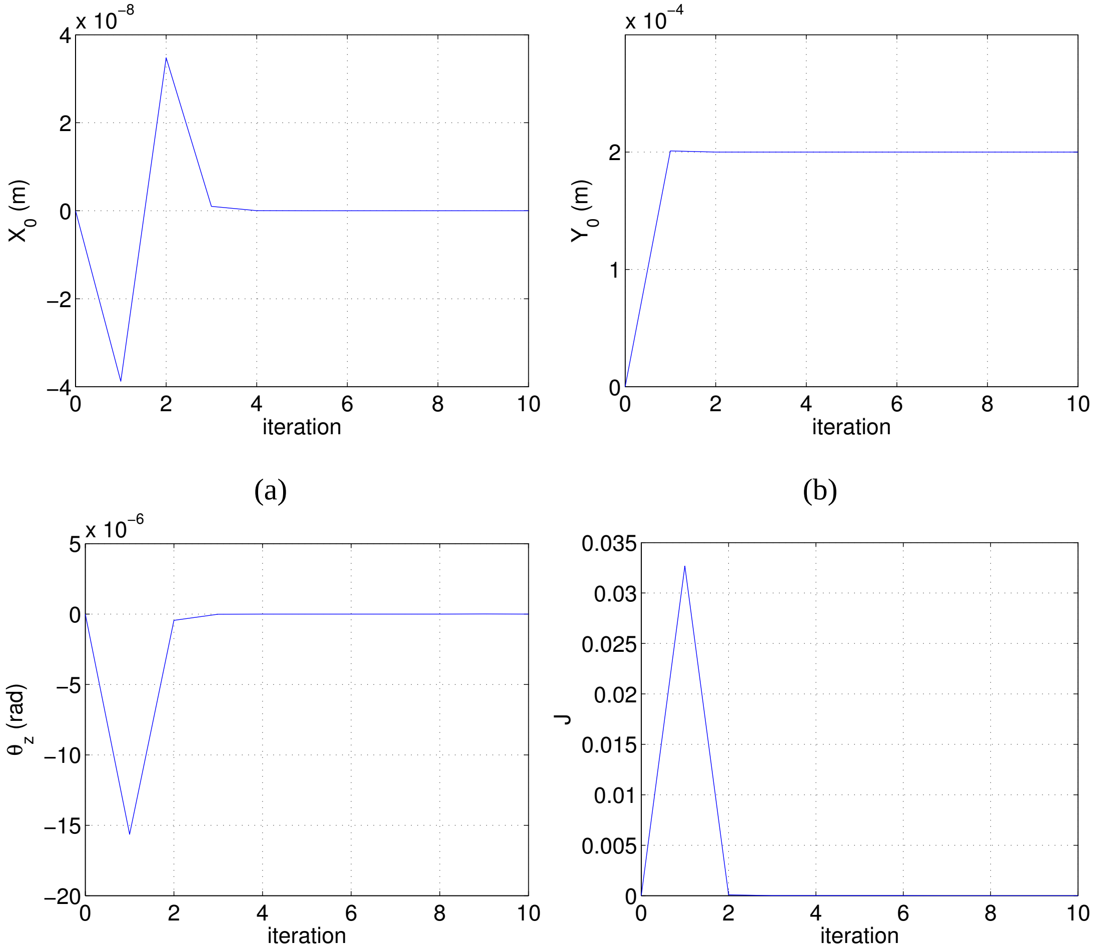
Giving a large step on the order of a few millimeters can make the GLSDC converge to an incorrect value. This can happen because the GLSDC algorithm will only converge to minimize J locally. Due to the periodic nature of the magnet matrix, it can only position within one pitch. Hence, if there is a local (incorrect) minimum closer to than the correct value, the GLSDC may converge to that incorrect value. However, in reality, the platen's dynamics will also be involved, and it is impractical for the platen to move very quickly (over 0.5 m/s), faster than the platen's design specifications. If the GLSDC algorithm routine can be executed at a faster sampling frequency, then it would result in faster convergence. The slow response of the GLSDC output may cause difficulty when designing a fast controller.

It should also be noted that using this method to position in 6 DOFs and adding more basis functions to the curve-fitting model will make it more difficult for the GLSDC to converge to the correct value. This is because by adding more basis functions, the mapped nonlinear relation will have more local minimums which the GLSDC may converge to. As of now, we have two two-axis Hall-effect sensors with which, in theory, we can obtain 3-DOF position estimates. However, due to modeling errors, we have not succeeded in closing all 3-DOF motions with only Hall-effect sensors. We believe that having redundant sensors will mitigate this problem.

2. DKF Dynamics with No Plant

The simulation results of the DKF are presented in this section. This simulation is done only to see the dynamics of the DKF, and hence the plant (platen) dynamic model is not included. The purpose of this simulation is to decide the value of q_x and q_θ so that the DKF output converges quickly and yet has attenuated the GLSDC output oscillation. Simulation is done with the value of p_0 fixed as $p_0 = 1 \times 10^{-8}$ m^2 and setting initial guess as $\{X_0, Y_0, \theta_z\} = \{0, 0, 0\}$. Values of q_x and q_θ are adjusted.

First, the input to the GLSDC are the estimated sensor output values for $\{X_0, Y_0, \theta_z\} =$ 100 μm, 0, 0 at $t = 0.1$ s, or at the 80th step. Sensor noise is modeled as a zero-mean white Gaussian noise (WGN) with a standard deviation of 1 mV. The outputs of the GLSDC and DKF are shown in Figure 37 for the case when $q_\theta = 10^5$ rad^2/s^6 and $q_x = 10^2$ m^2/s^6.



(a) (b)

(c) (d)

Fig. 36. Simulation GLSDC output to a 200-μm step input in Y, with no sensor noise (a) X_0, (b) Y_0, (c) θ_z, and (d) J

Similarly, the response to an input of $\{X_0, Y_0, \theta_z\} = \{0, 0, 1 \text{ mrad}\}$ is shown in Figure 38. The GLSDC output converges in approximately 3 steps, but the DKF dynamics cause the overshoot and oscillation, taking about 0.1 s to converge. This is due to the choice of q_θ and q_x, and there is a design compromise to choose whether we want the fast response or better noise reduction.

To have a better understanding of the effects of adjusting q_z and q_θ, simulations are run by first fixing q_θ as 10^5 rad^2/s^6 and changing q_x. The results to a step input of 100 μm are shown in Figure 39. Next, simulations are run by fixing q_x as 10^2 m^2/s^6 and changing q_θ, shown in Figure 40. The results are tabulated in Tables IV and V. The corner frequency (bandwidth) ω_{BW} of the DKF are calculated using second-order approximation from the rise-time t_r using the following equation [25].

$$t_r = \frac{1.8}{\omega_{\text{BW}}} \tag{5.1}$$

A conventional way of finding the poles of an estimator modeled as

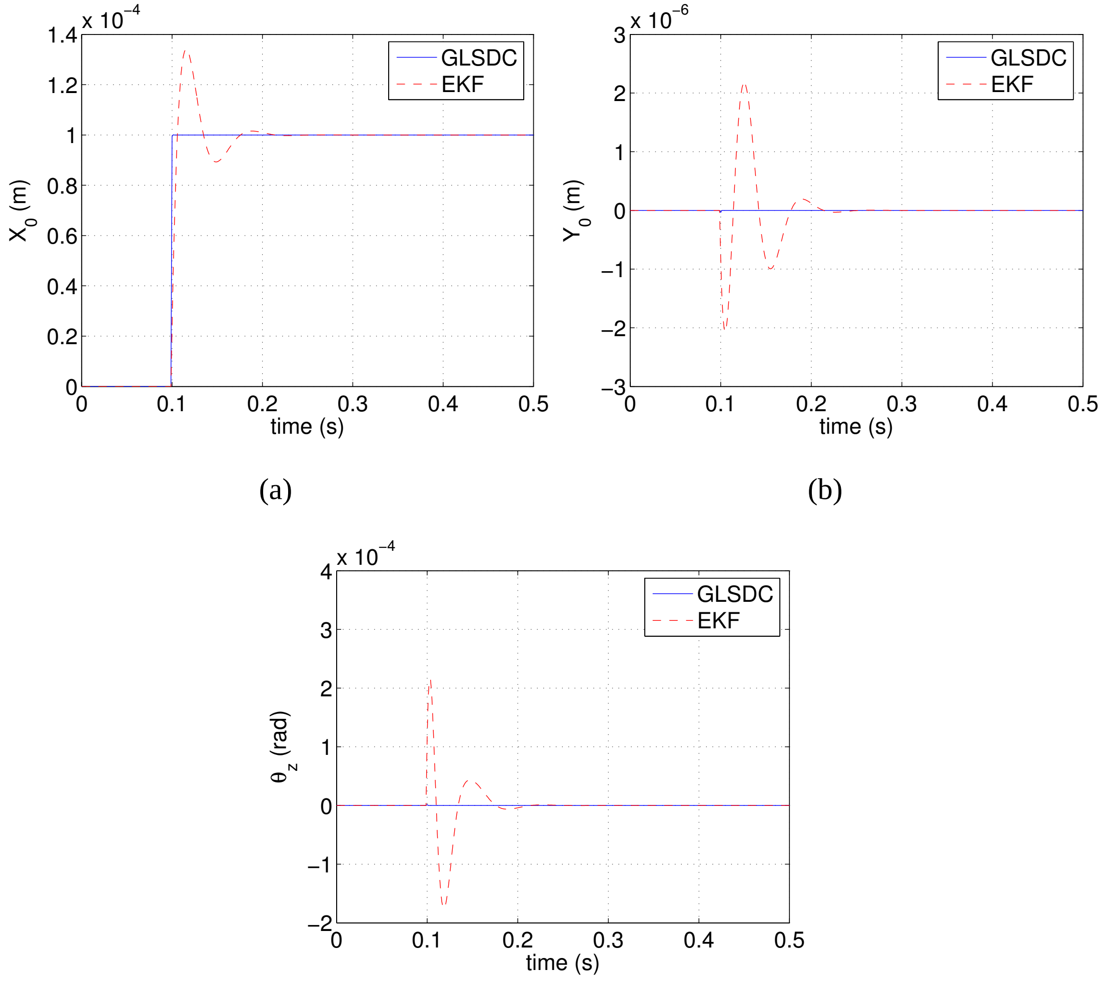
$$\begin{aligned} \dot{\hat{\mathbf{x}}}(t) &= F\hat{\mathbf{x}}(t) + B\mathbf{u}(t) + K(\tilde{\mathbf{y}}(t) - H\mathbf{x}(t)) \\ \hat{\mathbf{y}}(t) &= H\hat{\mathbf{x}}(t) \end{aligned} \tag{5.2}$$

where one would solve for the eigenvalues of E, defined as

$$E = F - KH. \tag{5.3}$$

In the case of the DKF, the K matrix varies in each time step because the P matrix is updated. Hence, although the P matrix will converge after some time, the E matrix is not constant and will change when a step input is applied. The simple second-order approximation is used to compute the bandwidth because it provides a rough estimate of the performance of the DKF, and seeing the response to a step input served its purpose.

Small q values filter out the noise, but result in slow convergence because the band-

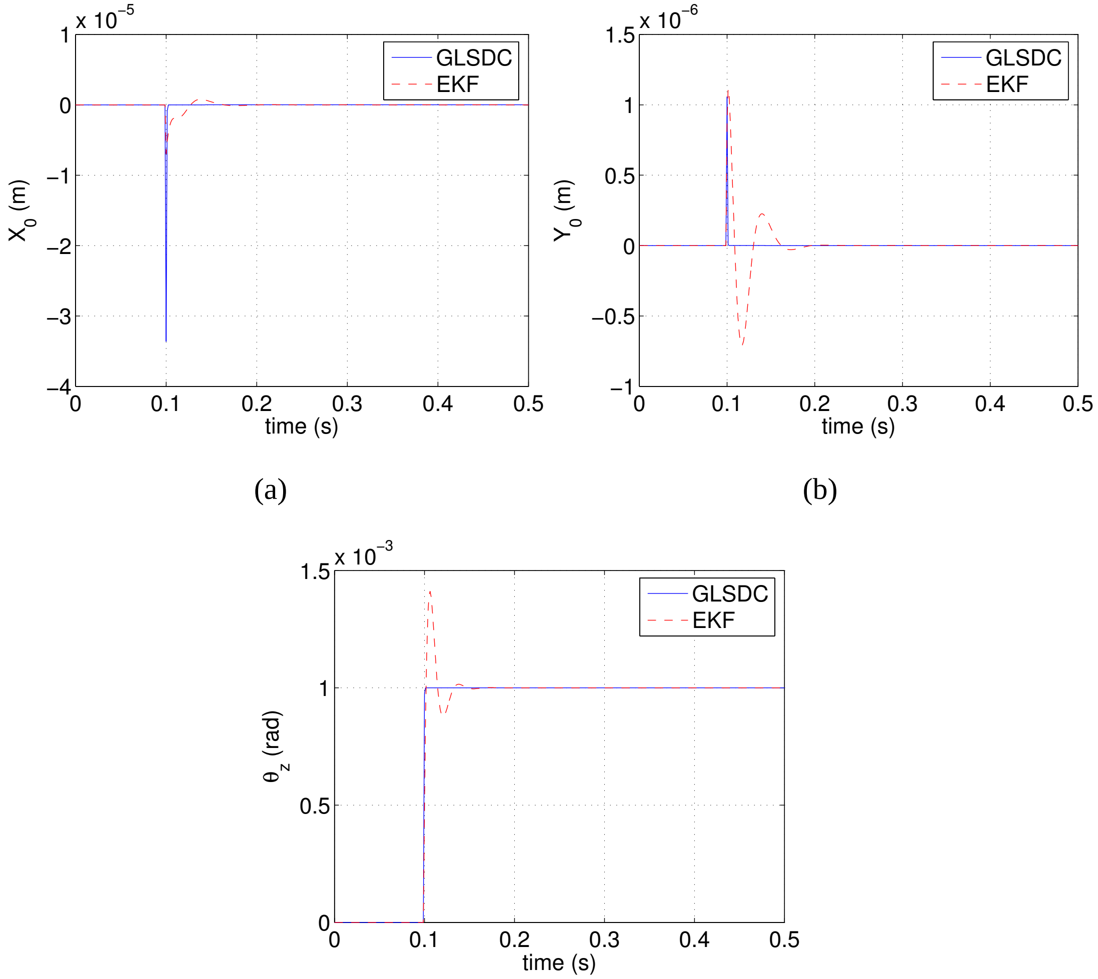


(a)

(b)

(c)

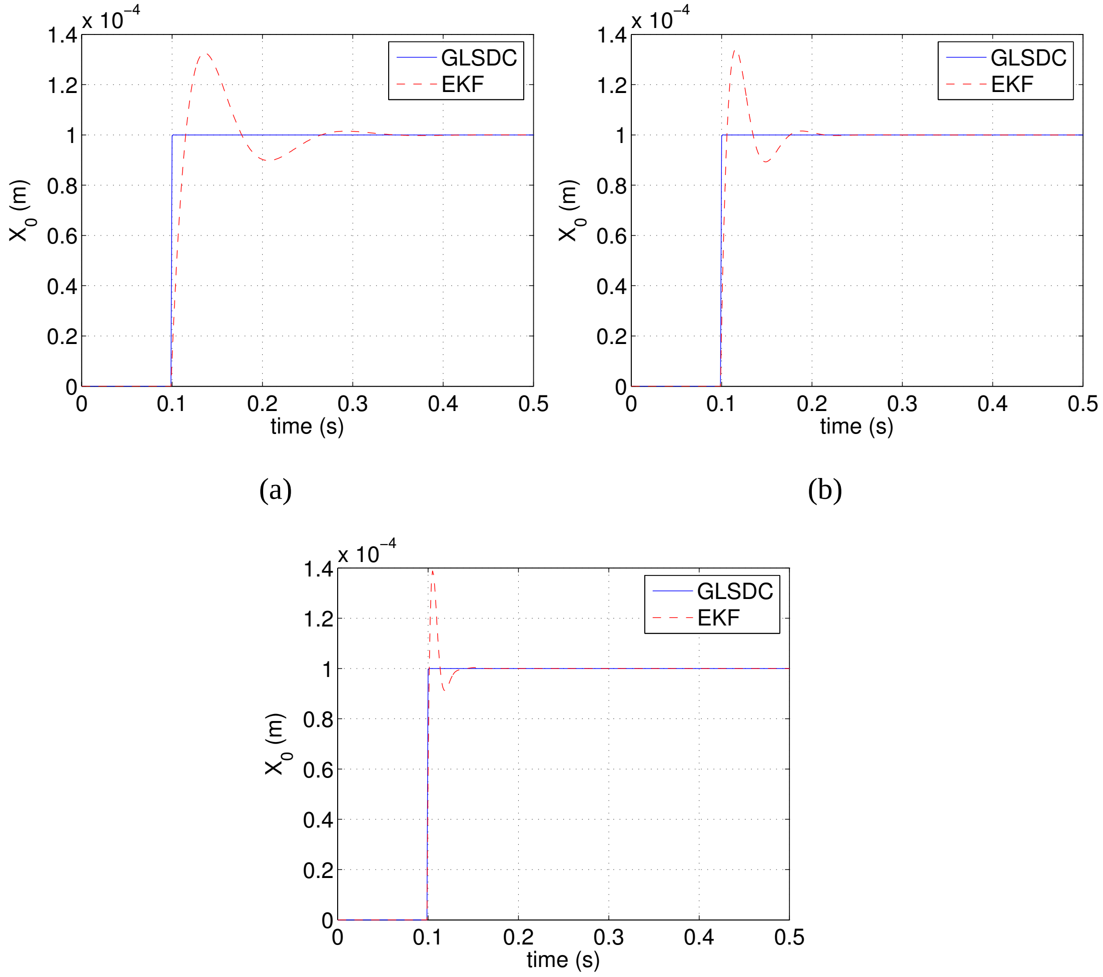
Fig. 37. Simulation DKF output to a 100-μm step input in X with $q_x = 100$ m^2/s^6 and $q_\theta = 1 \times 10^5$ rad^2/s^6 , (a) X_0, (b) Y_0, (c) θ_z



(a) (b)

(c)

Fig. 38. Response to step input of 1-mrad in θ_z with $q_x = 100\ \mathrm{m}^2/\mathrm{s}^6$ and $q_\theta = 1{\times}10^5\ \mathrm{rad}^2/\mathrm{s}^6$, (a) X_0, (b) Y_0, (c) θ_z

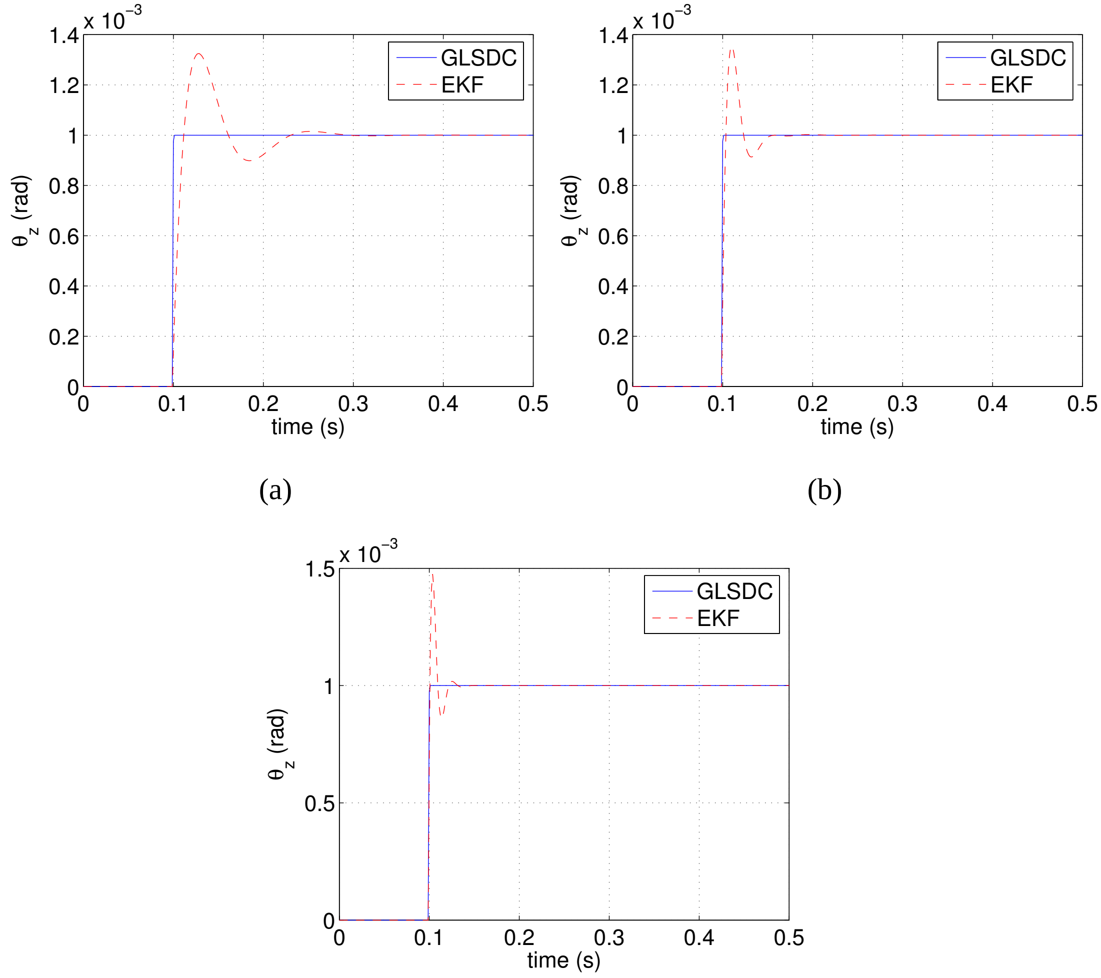


(a)

(b)

(c)

Fig. 39. Simulation results of X for GLSDC and DKF to a 100-μm step input in X with no sensor noise, $q_\theta = 10^5$ rad^2/s^6 , (a) $q_x = 1$ m^2/s^6 , (b) $q_x = 10^2$ m^2/s^6 , (c) $q_x = 10^4$ m^2/s^6



(a)

(b)

(c)

Fig. 40. Simulation results of θ_z for GLSDC and DKF to a 1-mrad step input in θ_z with no sensor noise, $q_x = 100\ \mathrm{m}^2/\mathrm{s}^6$, (a) $q_\theta = 1 \times 10^2\ \mathrm{rad}^2/\mathrm{s}^6$, (b) $q_\theta = 1 \times 10^4\ \mathrm{rad}^2/\mathrm{s}^6$, (c) $q_\theta = 1 \times 10^6\ \mathrm{rad}^2/\mathrm{s}^6$

Table IV. DKF results of changing values of q_x with $q_\theta = 10^5$ rad^2/s^6

q_x (m^2/s^6)	rise time t_r (s)	Overshoot M_p	$\omega_{\mathrm{B}W}$ (Hz)
1×10^{-1}	0.0178	32.4%	16.1
1×10^{0}	0.1021	32.9%	23.7
1×10^{1}	0.0080	33.6%	35.8
1×10^{2}	0.0054	34.4%	53.1
1×10^{3}	0.0033	35.2%	86.8
1×10^{4}	0.0020	40.0%	143.2

Table V. DKF results of changing values of q_θ with $q_x = 10^2$ m^2/s^6

q_θ (rad^2/s^6)	rise time t_r (s)	Overshoot M_p	$\omega_{\mathrm{B}W}$ (Hz)
1×10^{2}	0.0100	32.8%	28.6
1×10^{3}	0.0064	32.1%	44.8
1×10^{4}	0.0039	34.2%	73.5
1×10^{5}	0.0025	38.8%	115
1×10^{6}	0.0017	44.4%	169
1×10^{7}	0.0010	51.0%	286

width is small. Large q values will converge faster because the bandwidth of the filter is higher, but will not filter out the noise well. Also, increasing q will cause larger overshoot of the DKF output, which is not favorable. There must be a design compromise, and we choose $q_x = 10^2$ m^2/s^6 and $q_\theta = 10^5$ rad^2/s^6 . This will make the bandwidth of the DKF to be approximately 50 Hz for X and Y and approximately 100 Hz for θ_z, and overshoot be less than 40%. The bandwidths are different for X and θ_z because the controller requires higher crossover frequency in θ_z in order to stabilize it. However, for our current setup, the slow response of the DKF makes it impractical and unfeasible to use in the real-time control routine. Designing a faster DKF will result in faster convergence but will not serve its purpose to filter out the noise. Since the bandwidth of the filter should be 6 to 10 times faster than the controller, the controllers must be slow, but slow controllers may make the system unstable.

Simulations were done using other filtering techniques such as low-pass filters and averaging. However, these resulted in large oscillation outputs for both simulation and experimental results and could not stabilize the platen using it. This is probably due to the time delay which comes from discretization.

3. GLSDC and DKF with Plant and Controller

The simulink block diagram used for the simulation is shown in Figure 41. The plant is a 3-DOF linearized model. To emulate the sensor signal, the ‘sensor output with noise’ block outputs are the four sensor outputs with noise of 1-mV standard deviation. The ‘GLSDC’ and ‘GLSDC and DKF’ blocks are s-function blocks which compute the GLSDC and DKF algorithms, respectively. Various controllers have been tested using this simulation, both SISO and MIMO controllers. Simulation results using the controllers designed in Chapter IV are presented.

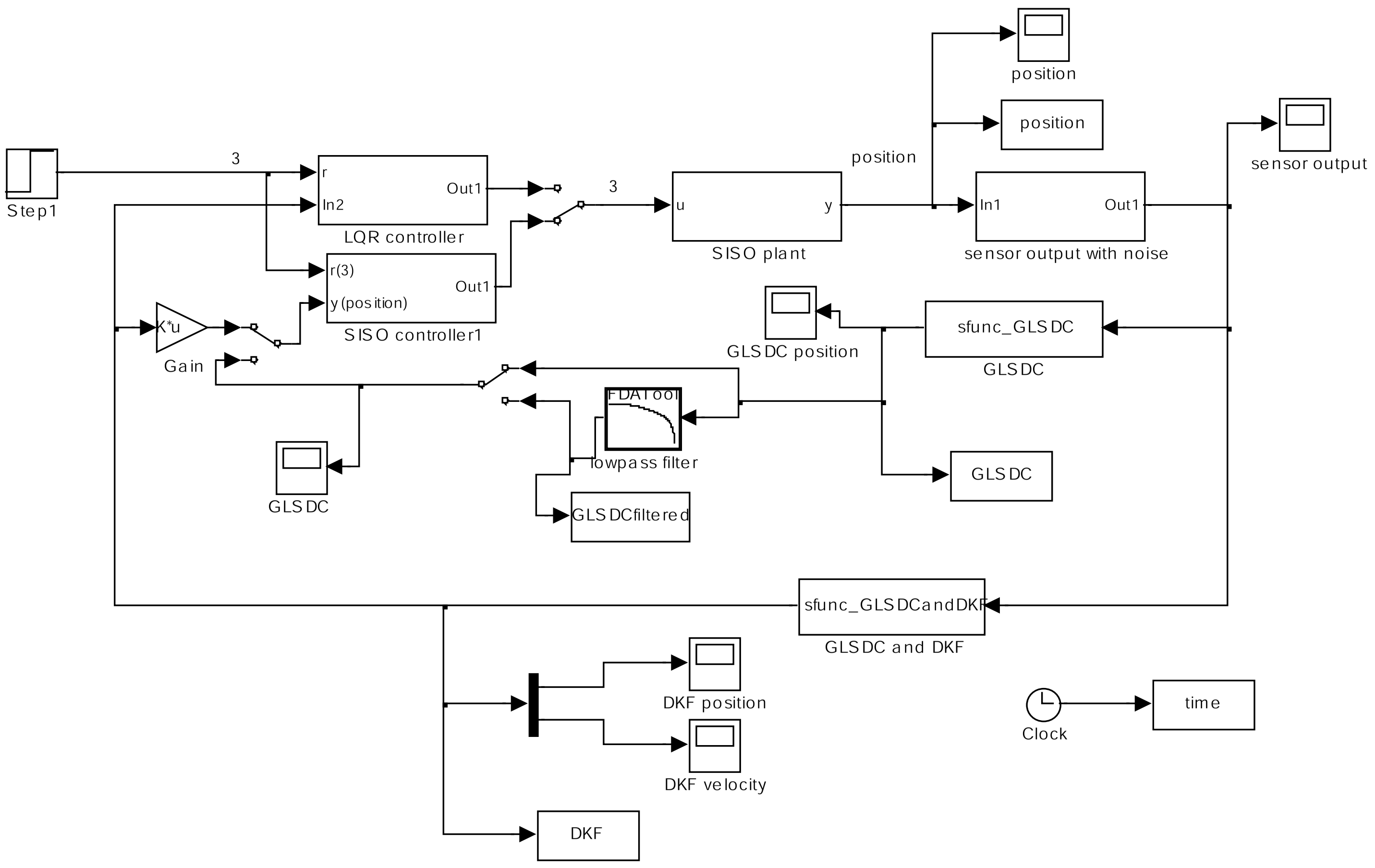


Fig. 41. Simulink block diagram for GLSDC and DKF

Simulation results to 10-μm and 100-μm step inputs when GLSDC outputs are used are shown in Figures 42 and 43, respectively. The results show that this sensing method is capable of 10-μm resolution positioning. Experimental results in the proceeding section confirms this result.

Next, simulation is done closing the loop with DKF outputs. The GLSDC and DKF outputs are shown in Figures 44 and 45, respectively. It can be seen that the system is stabilized although the response is slow and there is some oscillation. The DKF output shows improvement in the estimates compared to those of the GLSDC by attenuating the noise.

B. Experimental Results

First, experimental results present the GLSDC outputs when controlling X and Y control using the Hall-effect sensors, and θ_z control using laser interferometers. This is done to compare the performance of the proposed method with the laser interferometers. By closing the θ_z control using laser interferometers, the X and Y readings of the laser interferometers can be acquired, and are compared with the GLSDC outputs. Next, the results when closing all 3-DOF motion using the proposed method is presented.

The SISO controllers presented in Chapter IV are implemented.

1. 2-DOF Positioning

Experimental results for 2-DOF positioning is presented. These experiments were performed using two sets of two-axis Hall-effect sensors, and the X and Y control was closed using these sensors, and the θ_z control was closed using laser interferometers.

First, an experiment was conducted to measure the noise output of the GLSDC algorithm. The platen was controlled to be still at the origin, and outputs of the GLSDC and

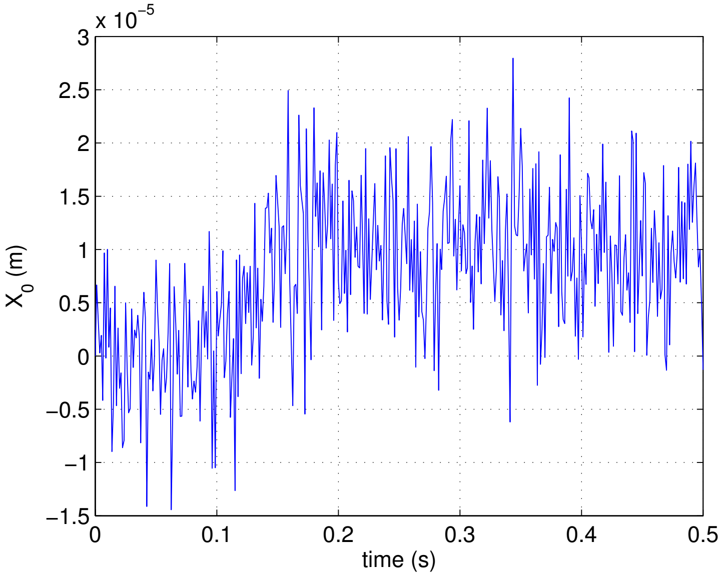


Fig. 42. Simulation result of GLSDC output for a 10-μm step

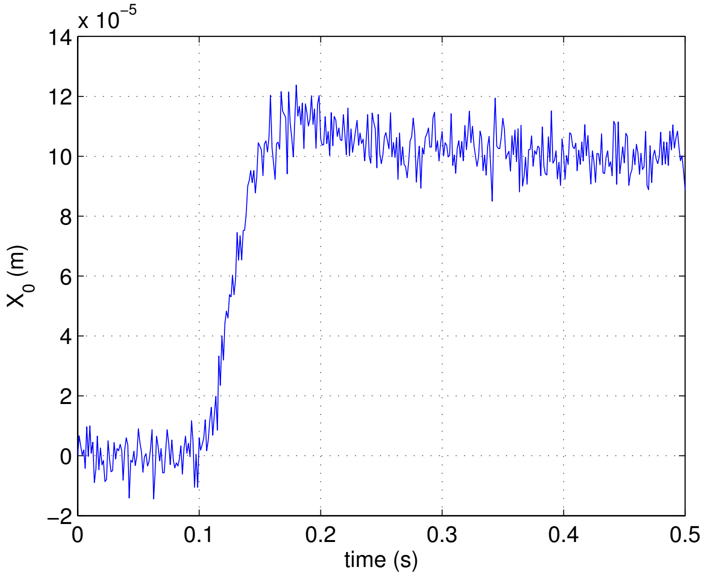
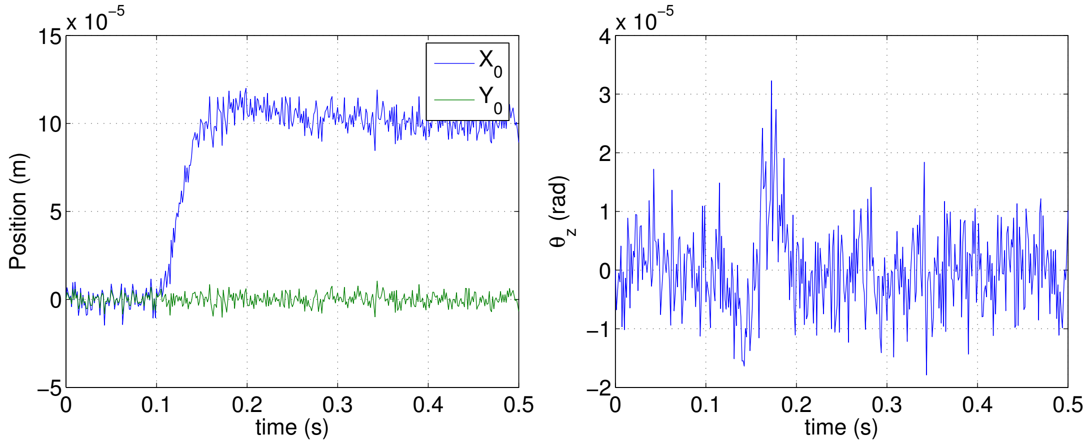
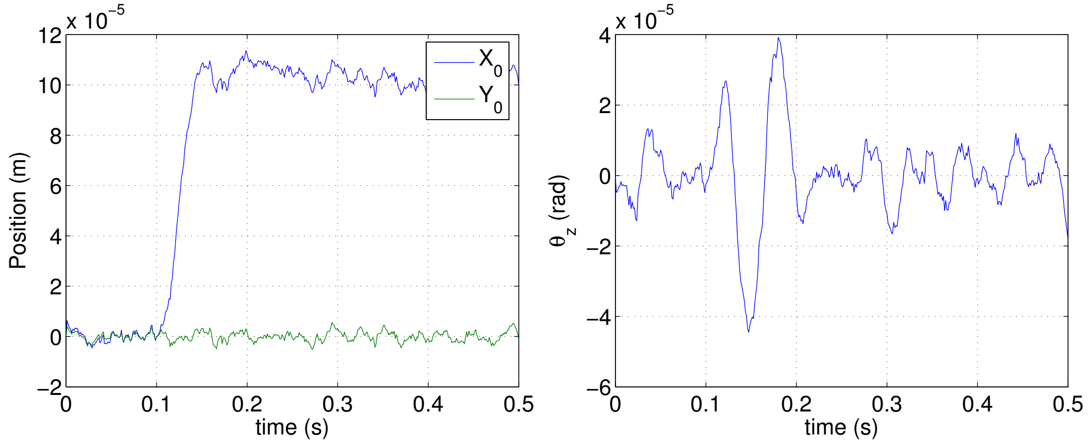


Fig. 43. Simulation result of GLSDC output for a 100-μm step



(a) (b)

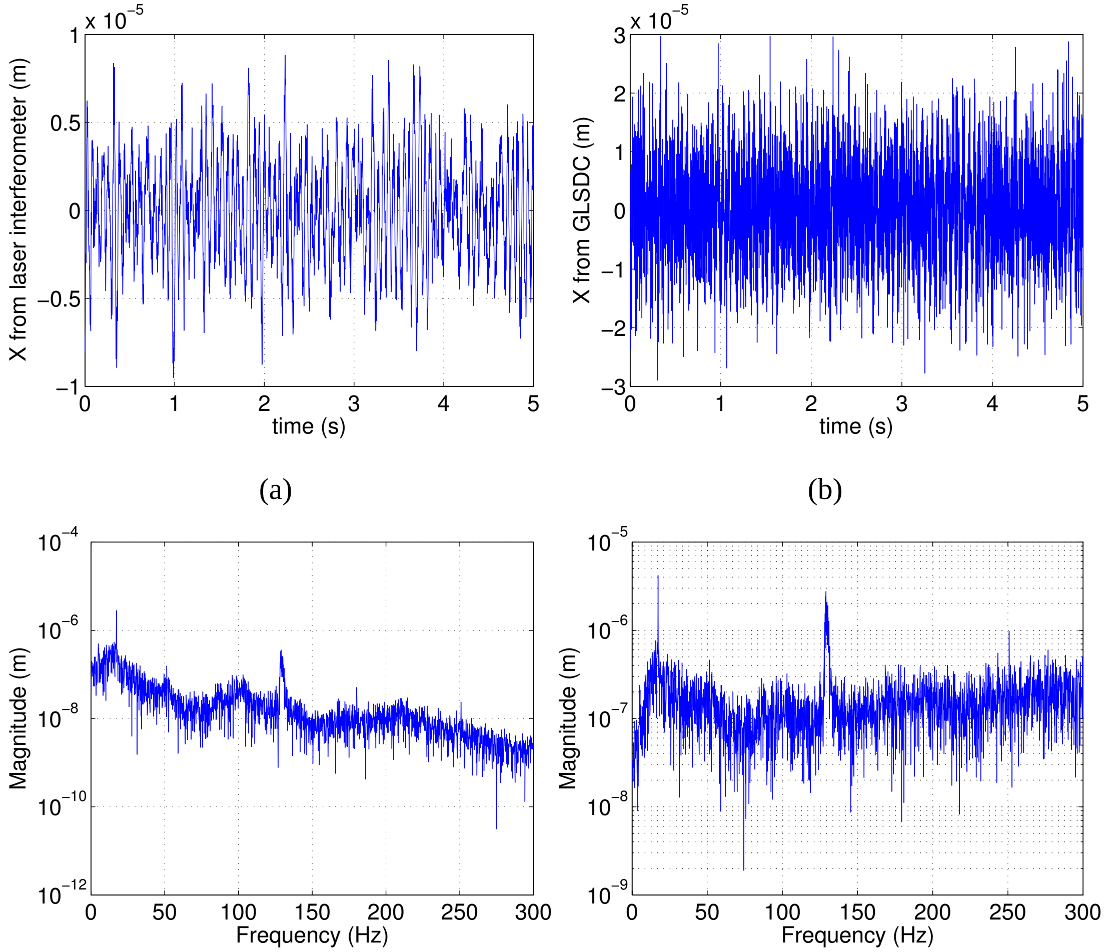
Fig. 44. Simulation result of GLSDC output for a 100-μm step, feedback using DKF output (a) X_0 and Y_0. (b) θ_z.



(a) (b)

Fig. 45. Simulation result of DKF output for a 100-μm step, feedback using DKF output (a) X_0 and Y_0. (b) θ_z.

laser interferometers were recorded. The outputs were sampled at 800 Hz for 5 s, and the results are shown in Figure 46 (a) and (b). The fast Fourier transform (FFT) results of the outputs are shown in Figure 46 (c) and (d), respectively. The sample standard deviation of the GLSDC algorithm outputs were 9.02 μm for X_0 and 7.6 μm for Y_0.



(a) (b)

(c) (d)

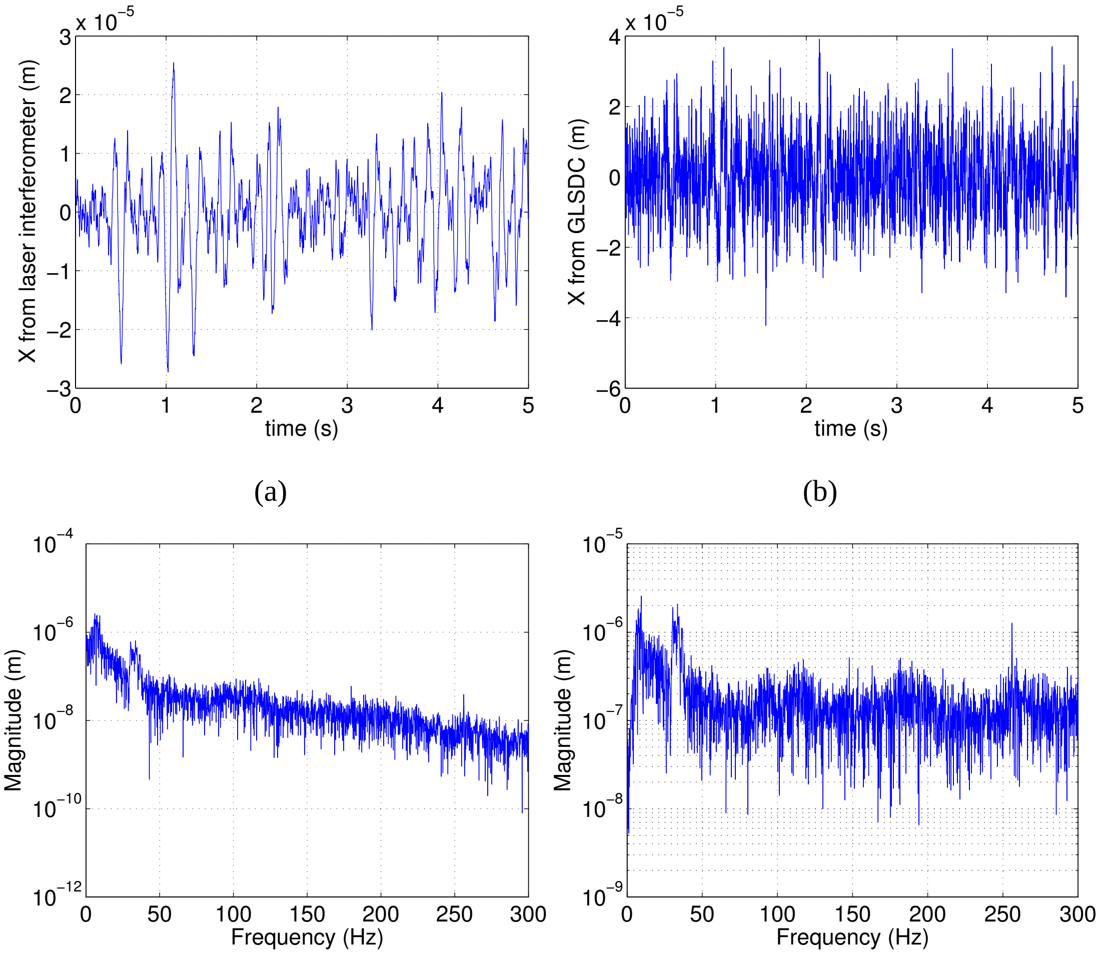
Fig. 46. Experimental results when there is no motion at the origin, and FFT results. (a) X_0 from laser interferometers, (b) X_0 from the GLSDC algorithm, (c) FFT of the laser interferometer output, and (d) FFT of the GLSDC output

To better understand the reason for the peak noise at 17 Hz and 230 Hz, compare the results with Figure 47. The results of Figure 47 were obtained under the same conditions, but using a lead-lag controller with a higher crossover frequency (20 Hz) for the X and Y controllers. The following digital controllers were implemented for X, Y, and θ_z are respectively the following,

$$\begin{aligned} D_X(z) &= 3.57 \times 10^5 \cdot \frac{z^2 - 1.9204z + 0.921233}{z^2 - 1.2423z + 0.2423} \\ D_Y(z) &= 3.57 \times 10^5 \cdot \frac{z^2 - 1.9204z + 0.921233}{z^2 - 1.2423z + 0.2423} \\ D_{\theta_z}(z) &= 7.35 \times 10^3 \cdot \frac{z^2 - 1.9204z + 0.921233}{z^2 - 1.2423z + 0.2423} \end{aligned} \tag{5.4}$$

Note the difference in the output and the FFT plots. The peaks of the FFT can vary depending on the controller used. The FFT plots show that high-frequency noise is present in the GLSDC outputs, which is why we have considered designing digital filters and the DKF. The sample standard deviation of the GLSDC algorithm outputs were 10.81 μm for X_0 and 10.76 μm for Y_0, which are larger than the above results. This experiment implies that the design of the controllers can affect the performance of this sensing method. If the noise profile of the GLSDC output can be known, then digital filters can be designed to prevent the noise. However, these results show that the noise profile depends on the controller. Hence, instead of designing a filter for each controller, we focus on the implementation of the DKF.

Responses to 10-μm consecutive steps in Y are shown in Figure 48. The outputs of both the laser interferometer and the GLSDC are shown. This shows that the Hall-effect sensors are capable of detecting the change in magnetic flux density for position changes of less than 10 μm, and hence this method has at least 10-μm position resolution. The results are comparable to the simulation results in Figure 42. The laser interferometer's outputs are not exactly 10 μm as the GLSDC output shows. This is due to the modeling error of the magnetic flux density, which comes from the reasons discussed in Chapter III.



(a) (b)

(c) (d)

Fig. 47. Experimental results when there is no motion at the origin using the fast controller, and FFT results. (a) X_0 from laser interferometers, (b) X_0 from the GLSDC algorithm, (c) FFT of the laser interferometer output, and (d) FFT of the GLSDC output

Next, an experiment was done to see the positioning accuracy with respect to the laser interferometer's outputs. A zigzag reference trajectory is given to the controller, and Figure 49 shows the result of the zigzag motion. The trajectory is identical to Figure 49 (a), and the platen has been controlled to follow the preplanned path very well. However, the actual path of the platen detected from the laser interferometers' outputs is shown in Figure 49 (b). There is a significant deviation between the measurements from the laser interferometers and the output of the GLSDC based on the Hall-effect sensors' outputs. The maximum positioning error can be as great as 1.4 mm, as shown in Figure 49 (c). Sensor calibration and use of redundant sensors may reduce this error.

Experimental results for large step inputs are presented next. Figure 50 shows the outputs to 1-mm step inputs, and Figure 51 shows the response to a 4-mm step. The GLSDC output follow the reference, but there is a deviation in the laser interferometer output. Since the laser interferometer outputs are closer to the true position, it should be that laser interferometer and GLSDC outputs are identical and the laser interferometer outputs show the 1-mm or 4-mm step.

It is interesting to note that either using very fast or slow controllers will destabilize this system. For X and Y control, when fast controllers (crossover frequencies higher than 20 Hz) are implemented, the response becomes more oscillatory. This is probably due to computational delay required for the GLSDC to converge, which takes about 3 iterations (3.75 ms). On the other hand, slow controllers (crossover frequencies lower than 5 Hz) will easily become unstable because this is a marginally stable system.

The same applies for θ_z control. The θ_z controller should be designed such that its crossover frequency is higher with a larger gain compared with the X and Y controllers. I believe this is due to the unmodeled external torque applied by the hung cables which connect the platen to the VME PC. In our experiment, the θ_z control became unstable with a controller with a crossover frequency of 10 Hz.

From these results, we found that there are performance limitations due to the DSP and our mechanical design. Further, implementing a DKF will impose stricter limitations, which made it very difficult to stabilize the platen in 3 DOFs using only Hall-effect sensors.

2. 2-DOF Positioning with Calibration

Calibration is performed to reduce the error between laser interferometers' outputs and the GLSDC outputs. The positioning error due to the GLSDC is predefined from the previous measurements (such as Figure 49 (c)), and the positioning error is then corrected by linear interpolation. Calibrated results for a 1-mm step and 4-mm step are shown in Figures 52 and 53, respectively. Comparing Figures 51 and 53, the error between the laser interferometer and the GLSDC has been reduced significantly. However, this calibration method is only feasible for 2-D positioning because this relies on the tabulated position error, and laser-interferometer outputs are necessary for the comparison. It would be increasingly difficult and cumbersome to tabulate when positioning in 3 DOFs or more. When using this method for positioning in higher DOFs, the only feasible solution is having redundant sensors and obtaining more precise model for the magnetic flux density as well as the sensor locations.

3. 3-DOF Positioning

The experimental results for 3-DOF positioning are presented in this section. As presented in Chapter IV, 3-DOF positioning can be done using two sets of two-axis Hall-effect sensors. However, it was difficult to control all 3-DOF motion using Hall-effect sensors. This is due to the variation between the measured and the modeled magnetic flux density, which results in large positioning error, and may also result in converging to the wrong values. Hence, one additional, theoretically redundant Hall-effect sensor was mounted to overcome this problem, as shown in Figure 54. The third Hall-effect sensor (sensor C) was mounted as shown in Figure 55. The notation a_2 and b_2 correspond to the connections to the Pentek 6102 A/D channels. Because the θ_z control is closed using Hall-effect sensors, the laser beam may stray off from the laser receiver. The following are the experimental results of the GLSDC without the laser interferometers.

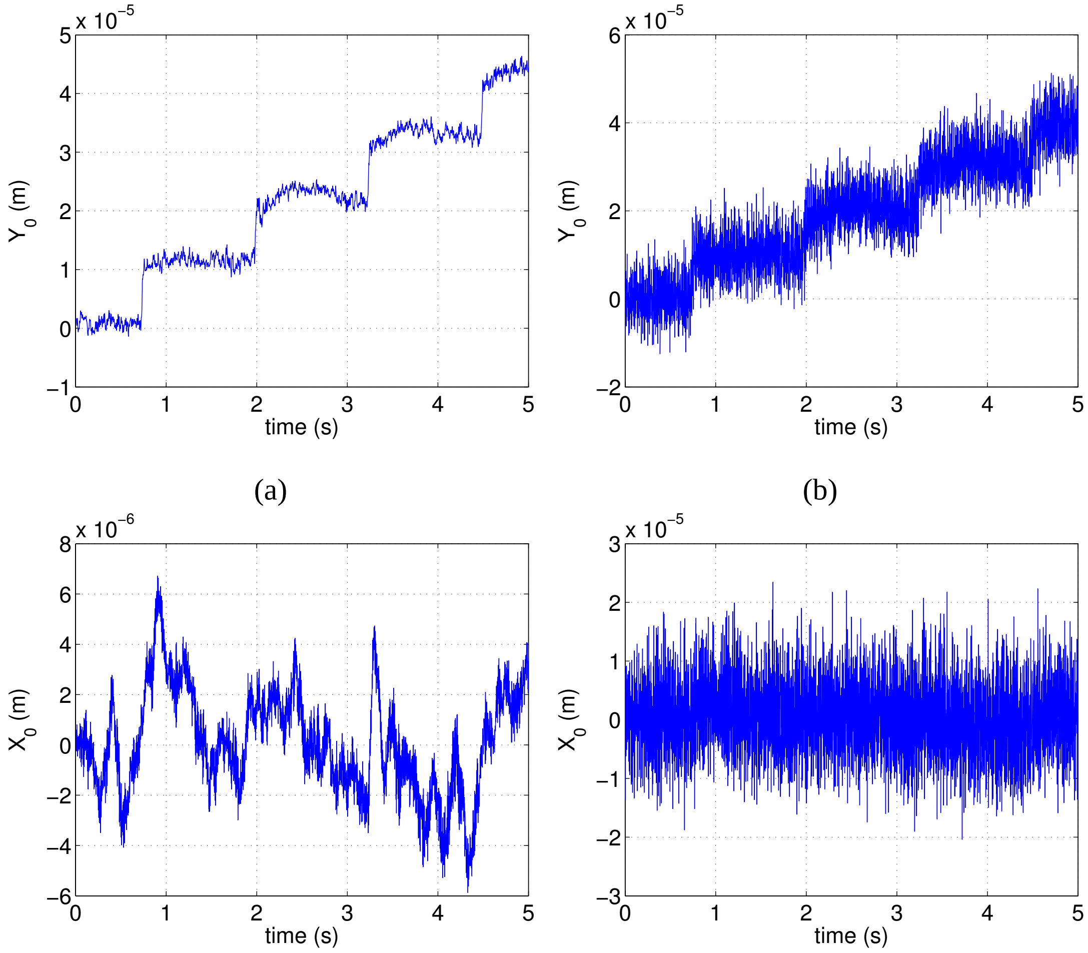
First, experiments were done to see the positioning resolution of the sensing method for 3 DOFs. The GLSDC outputs for 10-μm steps are shown in Figure 56. As in the 2-DOF case, the sensing method shows a positioning resolution of better than 10 μm. Next, the GLSDC outputs to 1-mm steps in Y are shown in Figure 57. The response shows some motion in X and θ_z, which are due to the coupled dynamics and the nonlinear nature of the GLSDC. The X_0 output shows a trend that as Y_0 increases, the positioning error at the instant of the step input increases. However, this is not always the case, for this depends on the modeling error for each sensor output at a particular position. The outputs for θ_z show an opposite trend, but this is also not always the case. Large positioning error implies that there is a large difference between the modeled and measured values for the sensor outputs. For this reason, having redundant sensors is effective in improving the accuracy and convergence to the correct local minimum.

Figure 58 shows the response to a 10-mm step input in Y. At 0.4 s, the outputs of X_0, Y_0, and θ_z did not converge immediately to the desired values. This is due to the GLSDC, for it solves for the local minimum value that minimizes the sum of square error. This error may happen when large step (as large as 30 mm) inputs are applied which result in the GLSDC output to converge to the wrong local minimum value. Also, steps as large as the magnet pitch (50.8 mm) will not be feasible because the magnetic field is periodic and the sensing method is only capable of positioning within one pitch. Hence, large step inputs are problematic although this problem is significantly attenuated compared to using laser interferometers. When using laser interferometers, steps as large as 1 mm can cause a responsive jerk where the laser beam can stray off from the laser receiver resulting in instability. This will not be the problem when using the Hall-effect sensors because it has unrestricted rotational range. The 22 rad/s oscillation in the X_0 output is primarily due to the actuators' setup for the platen. Because there is only one actuator which generates force to move in X, there is asymmetry in the force which causes some moment about the z axis, which causes the oscillation. The slow controller and the computation for the GLSDC are also the factors of this oscillation.

Next, experiments for rotational motion are presented. The GLSDC outputs for 100-μrad steps are shown in Figure 59. The steps are recognizable, and hence this sensing methodology has a 100-μrad resolution. Experimental results of 1-mrad and 10-mrad step inputs are shown in Figures 60 and 61, respectively. Some coupled motion in X and Y can be observed. In Figure 61, the GLSDC outputs for a large rotation of 40 mrad (approximately 2.3 deg) show a large oscillation in all X_0, Y_0, and θ_z outputs. Some possible reasons for this oscillation may be (1) the linearized models of the actuators are not appropriate for large rotations, (2) modeling error of the magnetic flux density, and (3) precise locations of the Hall-effect sensors are not known. (3) is especially problematic because the location of the platen center of mass is only known from the design, and large rotation in θ_z can result in a large difference between the actual and the modeled values. Because the sensor output model is obtained for the case with no rotation, the modeling error is likely to increase as the rotation in θ_z becomes large.

This trend is also present in the large ramp inputs shown in Figures 62 and 63. The results for $+\theta_z$ motion in Figure 62 shows large oscillation compared to $-\theta_z$ rotation shown in Figure 63. This 22 rad/s oscillation is similar to the results of Figure 58, and is due to various modeling error which will become large as each sensors' measurements differ from the modeled values.

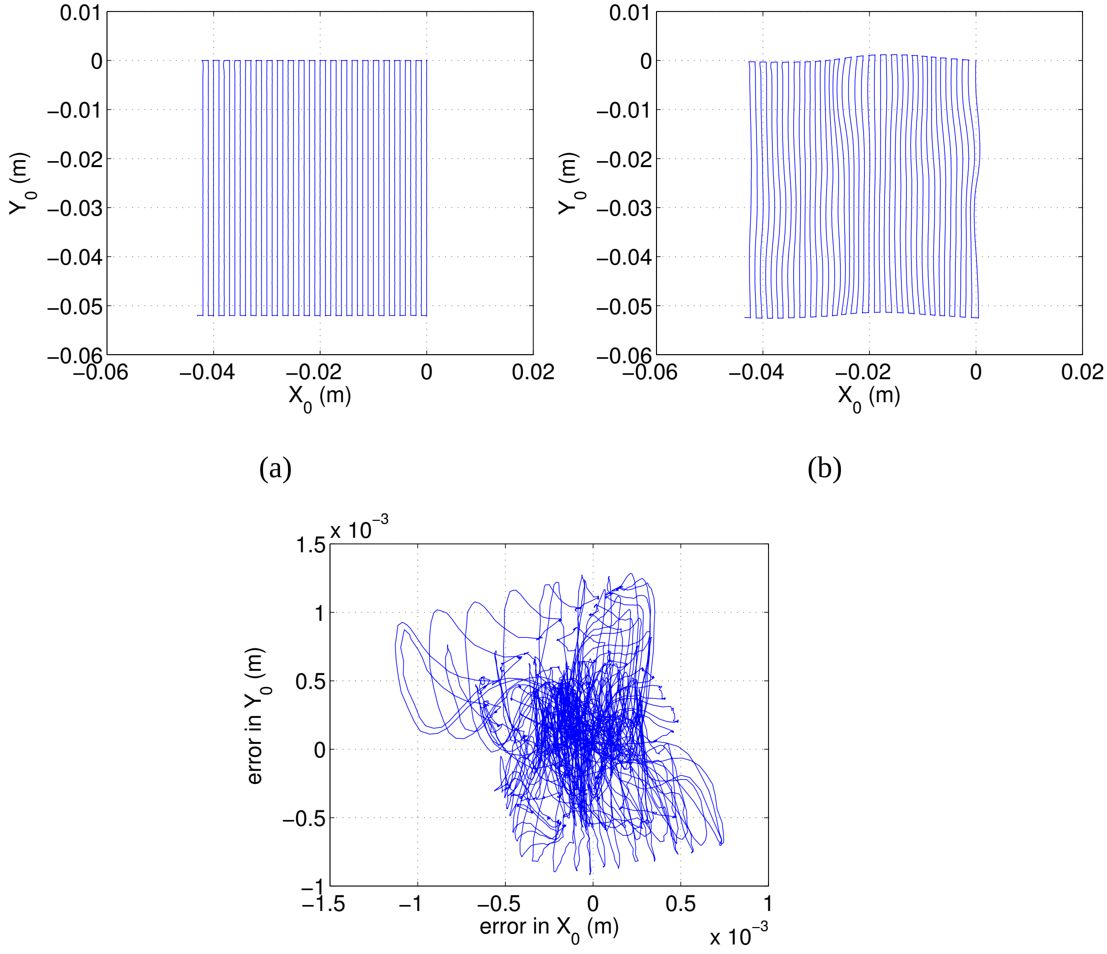
Despite these positioning error due to the modeling error, the results show that it is capable of rotating -0.12 rad to 0.16 rad (approximately -6.9° to 9.2°). Hence, this sensing method shows its capability of sensing large rotation and the results confirm that this sensing methodology has large, unrestricted rotational range, which laser interferometers and any conventional translational positioning sensors are capable of.



(a) (b)

(c) (d)

Fig. 48. Experimental results of 10-μm steps in Y. (a) Y-outputs of laser interferometer, (b) Y-outputs of the GLSDC, (c) X-outputs of laser interferometer, and (d) X-outputs of the GLSDC

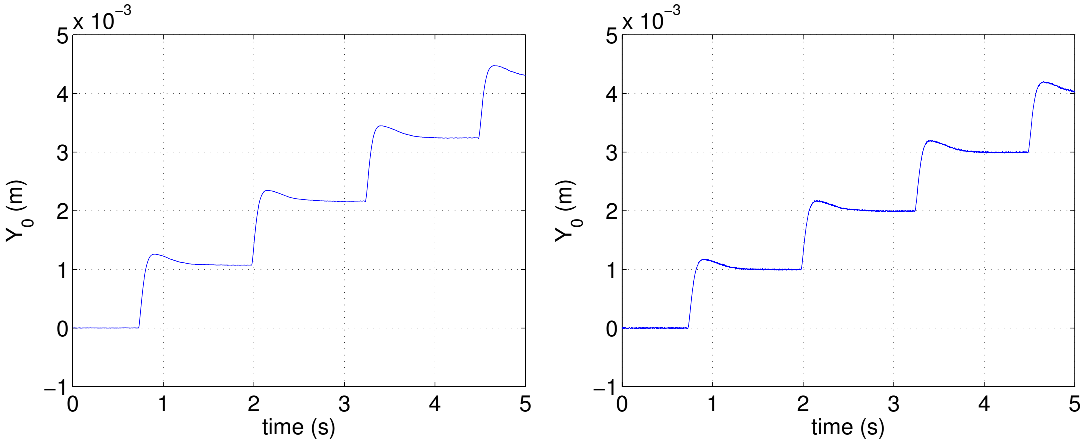


(a)

(b)

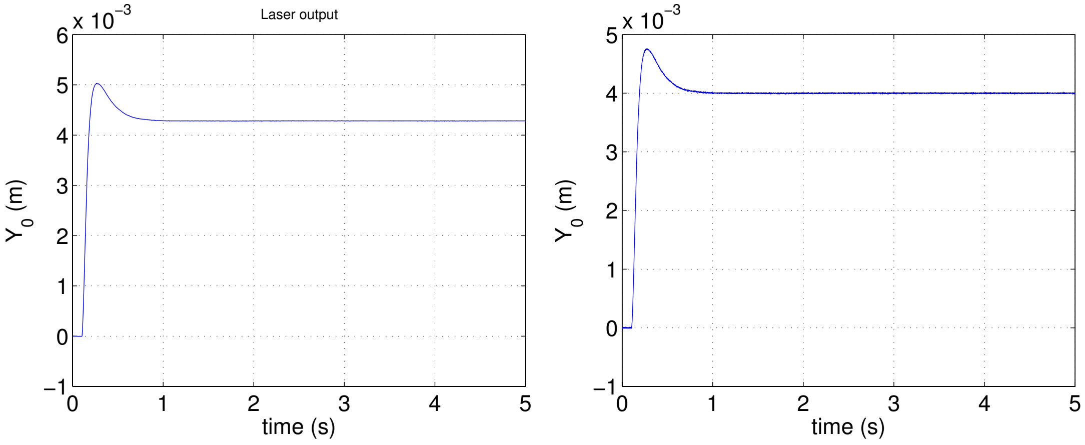
(c)

Fig. 49. Experimental results of positioning platen in *X* and *Y* following a zigzag trajectory. (a) Measured position from Hall-effect sensors. (b) Measured position from laser interferometers used for Hall-effect-senor calibration purpose. (c) Error between the two measured values



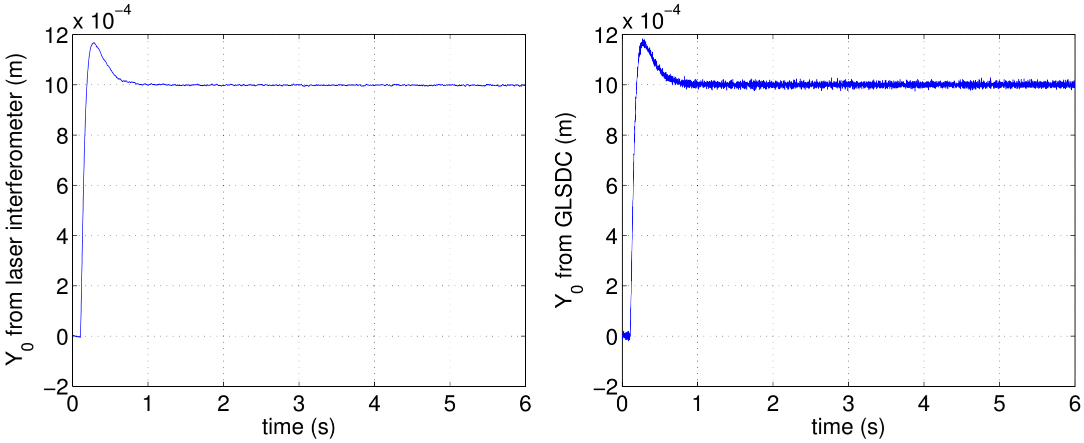
(a) (b)

Fig. 50. (a) Laser interferometer and (b) GLSDC output to 1-mm consecutive steps in *Y*



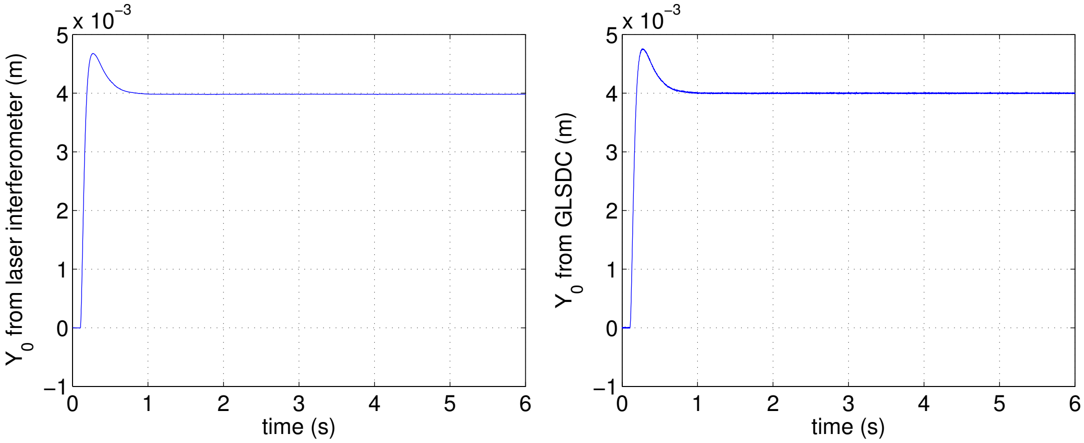
(a) (b)

Fig. 51. (a) Laser interferometer and (b) GLSDC output to a 4-mm step in *Y*



(a) (b)

Fig. 52. (a) Laser interferometer and (b) GLSDC output to a 1-mm step in *Y* with calibration



(a) (b)

Fig. 53. (a) Laser interferometer and (b) GLSDC output to a 4-mm step in *Y* with calibration

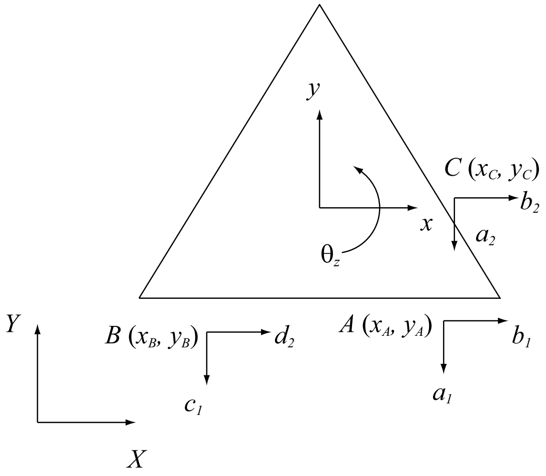


Fig. 54. Experimental setup with three sensors

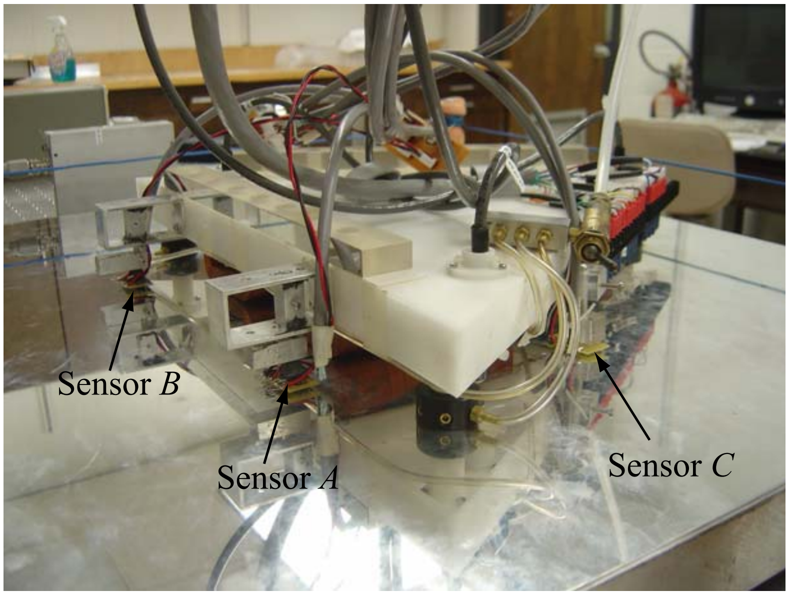
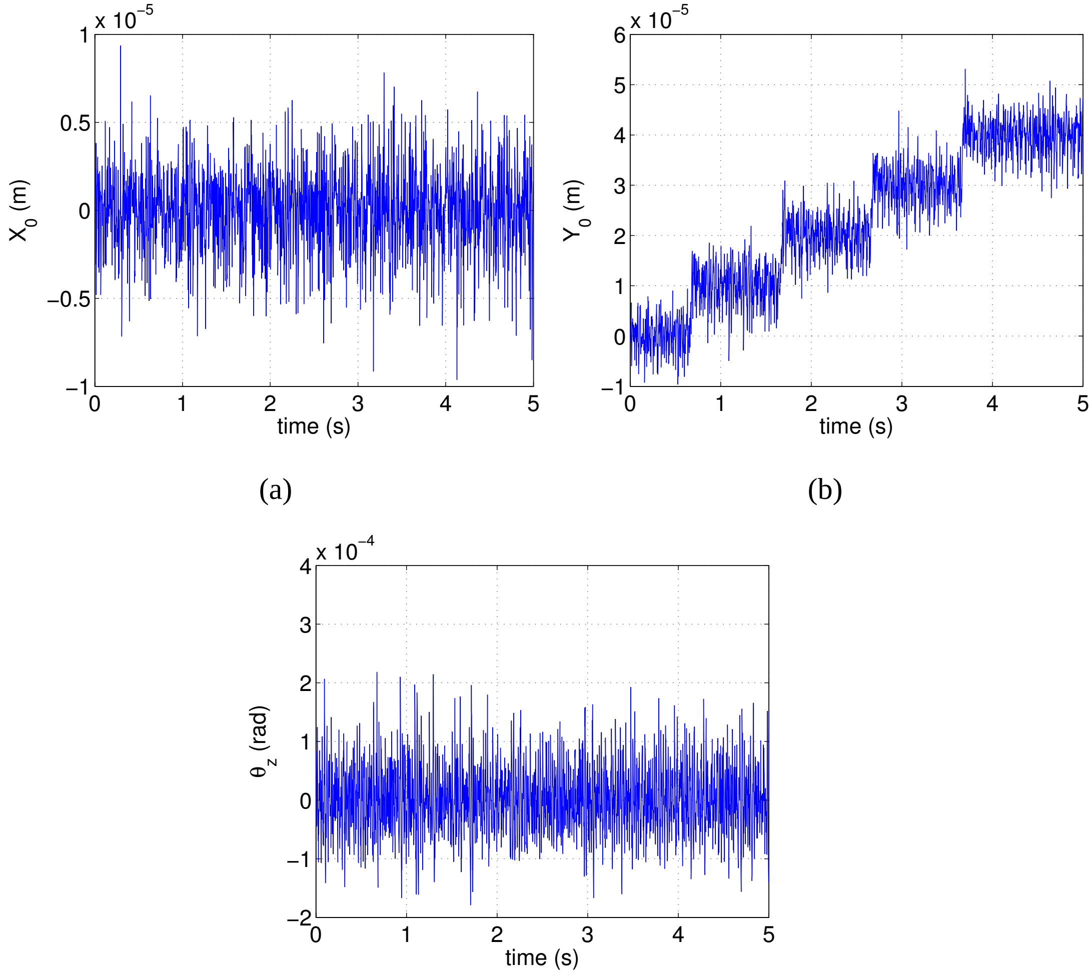


Fig. 55. Experimental setup with three sets of 2D-VH-11SO

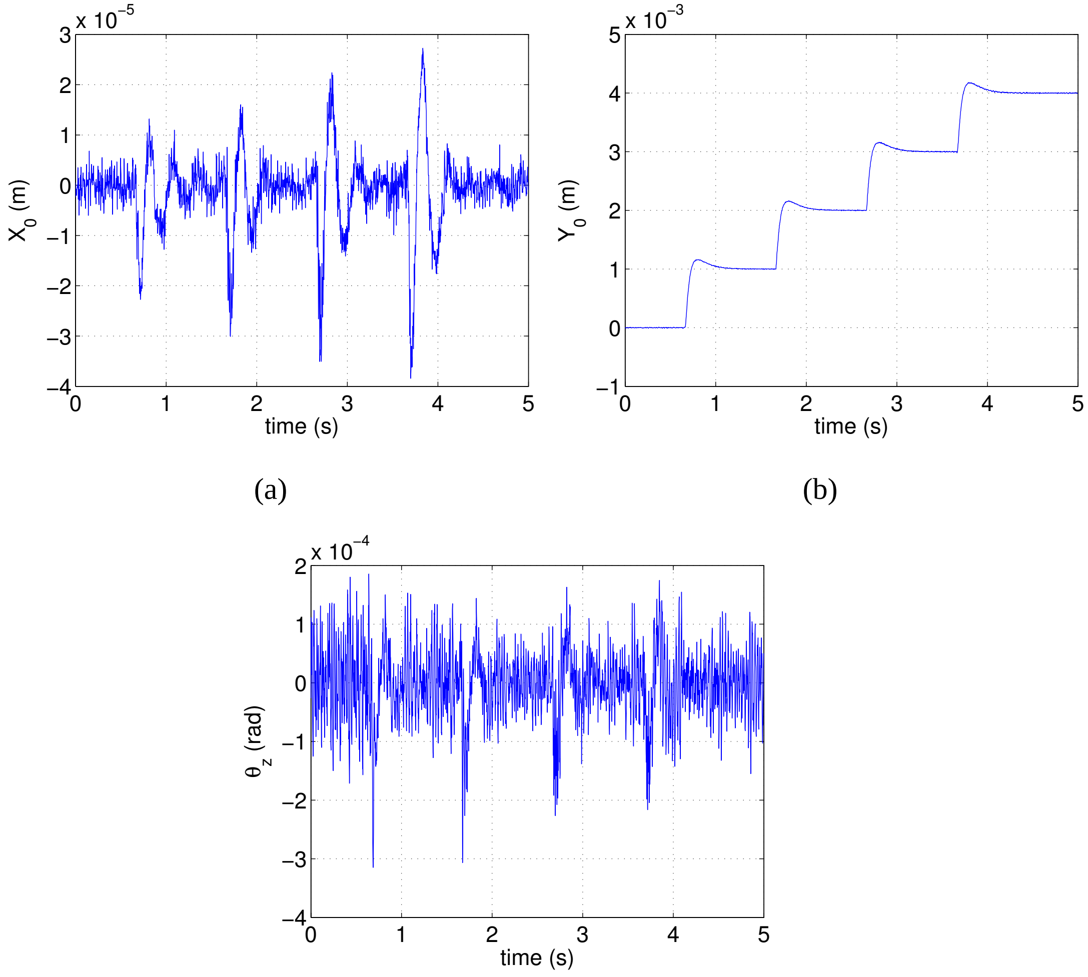


(a)

(b)

(c)

Fig. 56. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 10-μm steps in Y (a) X_0, (b) Y_0, and (c) θ_z

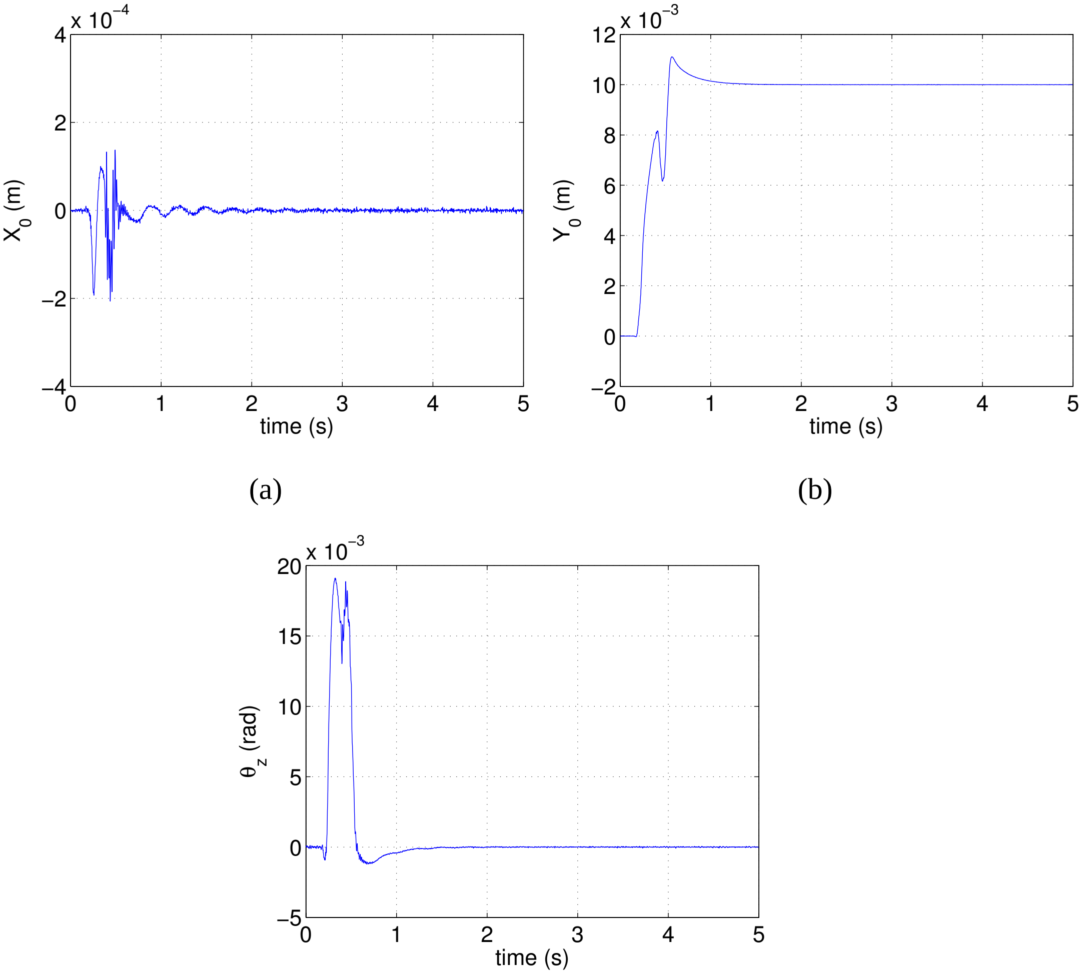


(a)

(b)

(c)

Fig. 57. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 1-mm steps in Y (a) X_0, (b) Y_0, and (c) θ_z

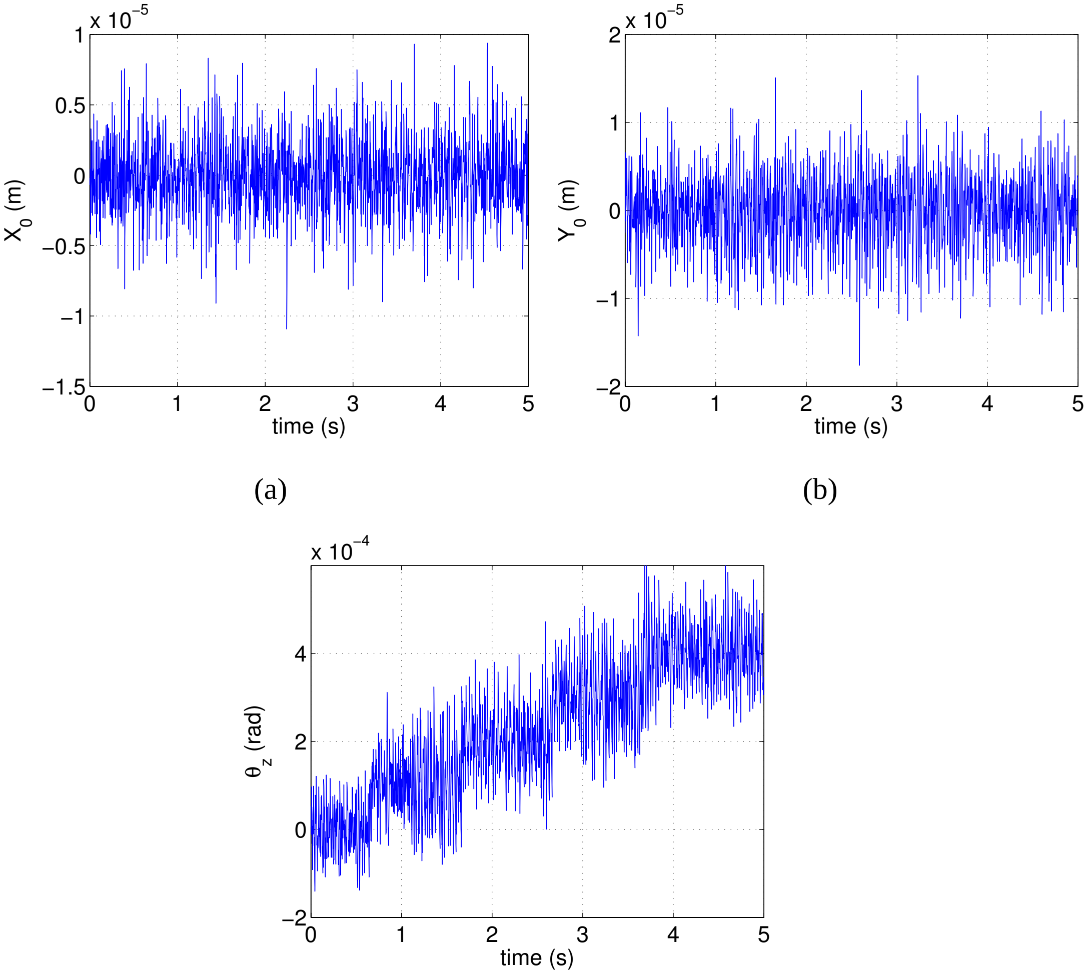


(a)

(b)

(c)

Fig. 58. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for a 10-mm step in *Y* (a) X_0, (b) Y_0, and (c) θ_z

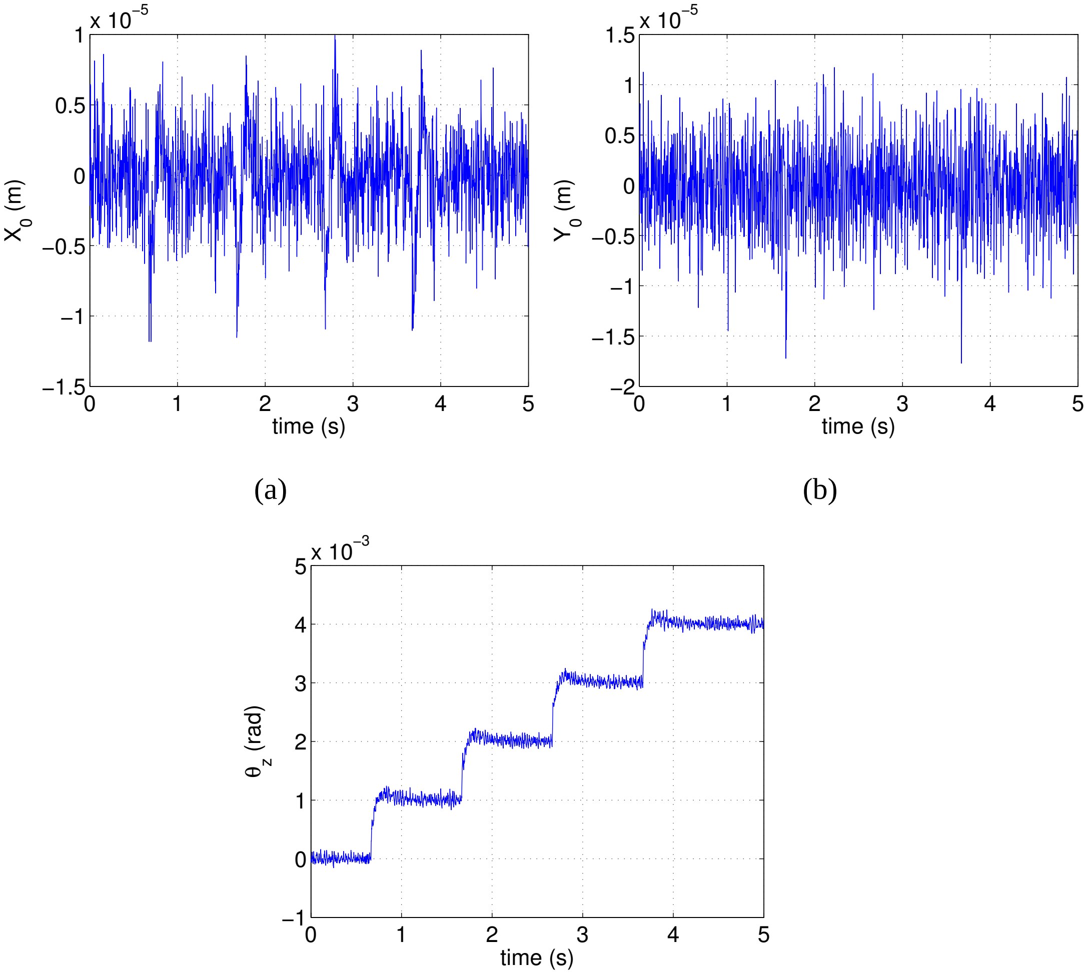


(a)

(b)

(c)

Fig. 59. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 100-μrad steps in θ_z (a) X_0, (b) Y_0, and (c) θ_z

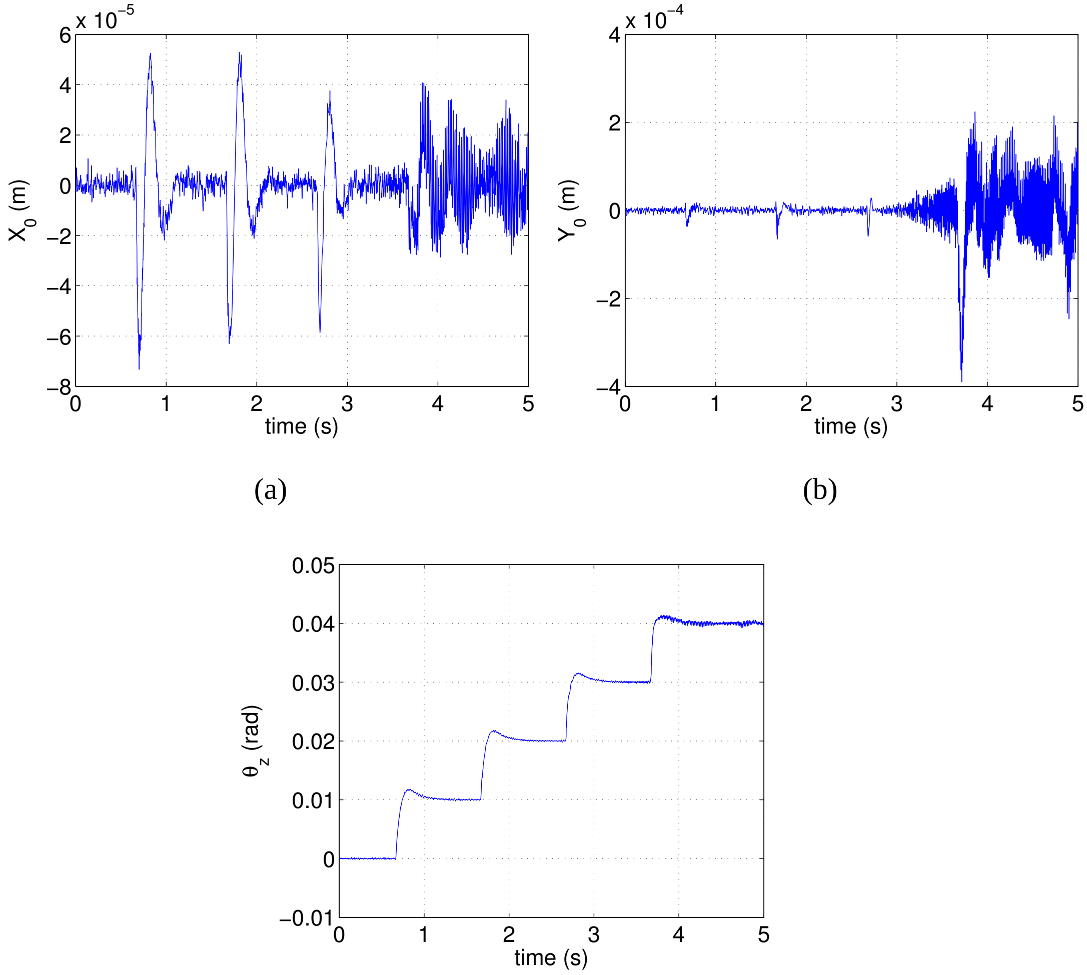


(a)

(b)

(c)

Fig. 60. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 1-mrad steps in θ_z (a) X_0, (b) Y_0, and (c) θ_z

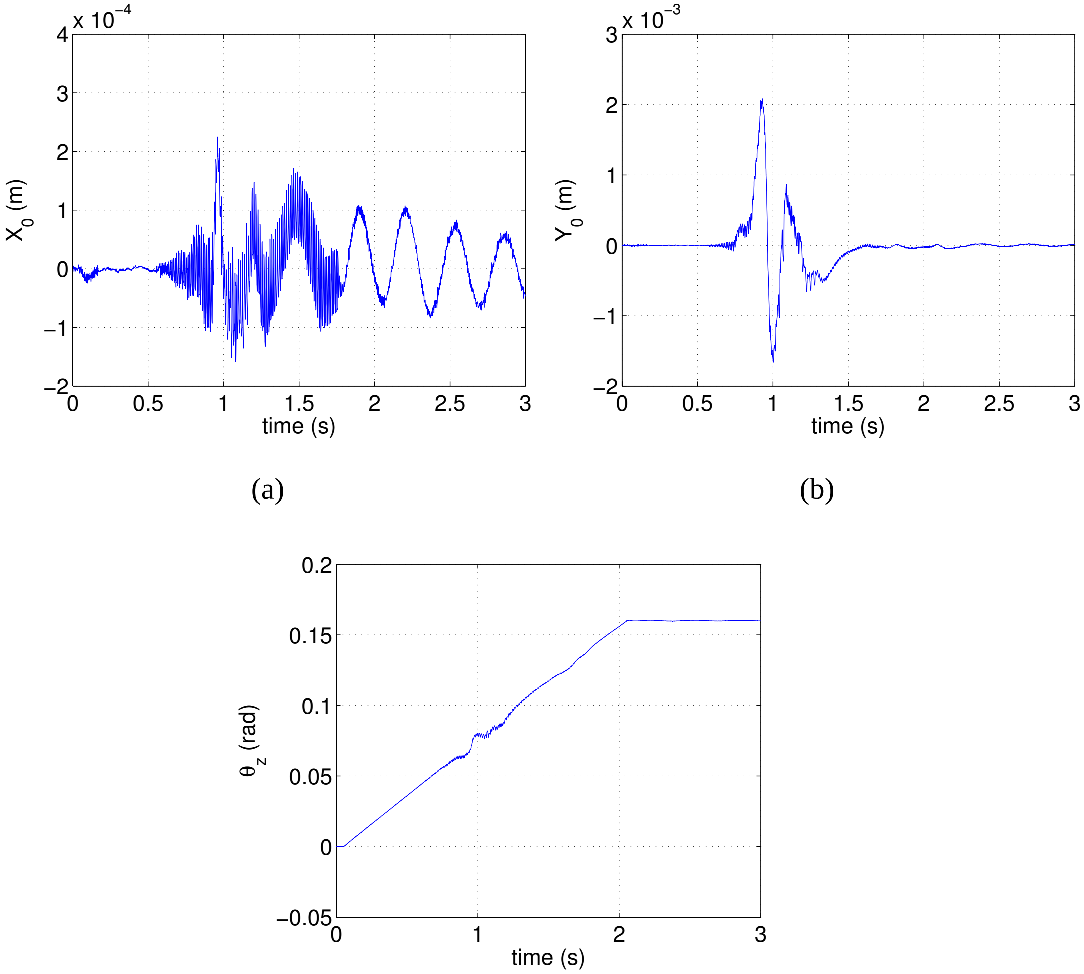


(a)

(b)

(c)

Fig. 61. Experimental results of positioning platen in 3 DOFs using the proposed algorithm for 10-mrad steps in θ_z (a) X_0, (b) Y_0, and (c) θ_z

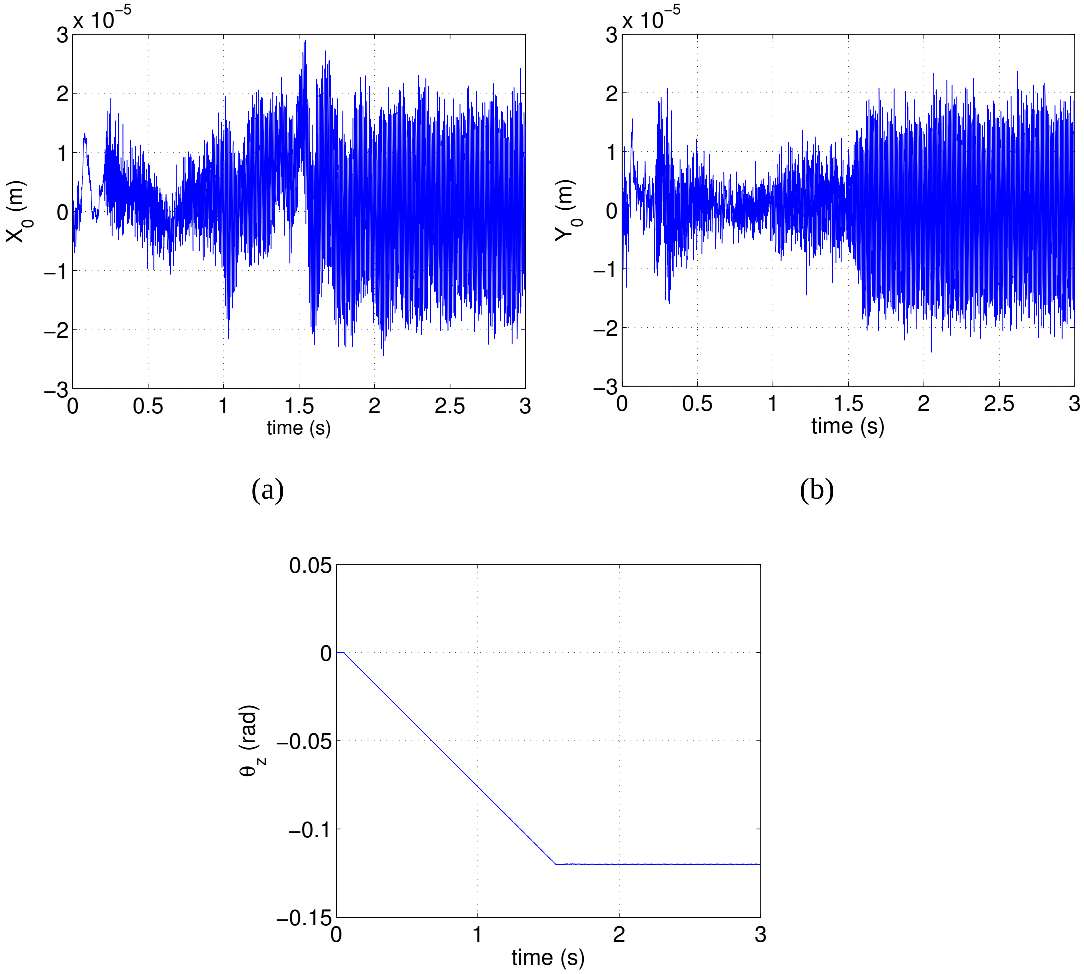


(a)

(b)

(c)

Fig. 62. Experimental results for ramp input in $+\theta_z$ rotation. (a) X_0, (b) Y_0, and (c) θ_z



(a)

(b)

(c)

Fig. 63. Experimental results for ramp input in $-\theta_z$ rotation (a) X_0, (b) Y_0, and (c) θ_z

CHAPTER VI

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

This chapter contains the conclusions regarding the design, construction, and performance of the sensing methodology utilizing two-axis Hall-effect sensors. Some applications for which this sensing methodology is appropriate are discussed. This chapter also includes suggestions for future work to improve the performance of the sensing and state estimation.

A. Conclusions

In this thesis, a novel multi-DOF precision position sensing methodology was proposed and implemented. The contribution of this thesis includes the analytical framework of the nonlinear mapping between the Hall-effect sensors' outputs and the position of the platen. A Gaussian least squares differential correction algorithm was implemented and modified to achieve this, and the algorithm proved to be very effective in solving the nonlinear relationship. The basis functions of the magnetic flux density generated by the Halbach magnet matrix were derived from the analysis of the Halbach magnet matrix. Since the Hall-effect sensors are very sensitive to external noise, the power-supply and amplifier-circuit design was explained in detail. The key advantage in using this proposed sensing method is that absolute position sensing is possible atop a magnet matrix with unrestricted translational and rotational motion range. It only requires small, inexpensive Hall-effect sensors to be mounted with a simple analog interface circuit.

As a proof of concept, this method has been implemented on our maglev stage, and shows positioning resolution of 10 μm and 100 μrad, and positioning accuracy of better than 1.4 mm, which can reduced with error mapping and correction. This sensing method also shows good repeatability. These specifications are obtained experimentally and can be improved for more demanding applications. The position accuracy primarily depends on the noise from the Hall-effect sensors and the modeling error between the measured and the curve-fitted model.

The unrestricted translational motion range is a result of utilizing the periodicity and orthogonality of the Halbach magnet matrix's magnetic field, and was an ideal platform for testing this methodology. However, the obtained model has significant modeling error which comes from (as presented in Chapter IV):

- Fabrication error of the Halbach magnet matrix
- Error in material properties of the magnets
- Modeling error of the magnet matrix using batch least squares curve-fitting
- Hall-effect sensor misalignment with respect to the platen's axes, in θ_x, θ_y, and θ_z
- Error in defining the precise location of the sensor (the sensitive volume of the Hall-effect sensor is only 250 μm × 250 μm × 200 μm)
- External magnetic field interference and sensor noise.

If the reference model of the GLSDC has large deviations with the actual measurements, it will result in large positioning error, and may even result in converging to a wrong local minimum position. These are the reasons which make this difficult to achieve high accuracy.

The use of a recursive discrete-time Kalman filter is one way to obtain better estimates of a noisy signal. However, the filter's dynamics was not fast enough compared with the plant dynamics, and hence there was a difficulty in stabilizing the platen while implementing the DKF in real time. The reason why the DKF was designed was because the platen could not be controlled in 3-DOF using only two sets of two-axis Hall-effect sensors, and my guess was that the noisy output of the GLSDC was the reason for the instability. This problem was overcome by having an additional, theoretically redundant, Hall-effect sensor mounted.

B. Applications

Through this research, the fundamental sensing methodology and algorithm has been developed and can readily be applied for various applications. This sensing methodology will allow estimating position where there is any periodicity in the magnetic (or any) field. This is not only limited to permanent magnets, but can also be used directly for actuator coils. In that case, the rotor (actuator coil) will move with respect to the stator (magnet array), and the Hall-effect sensors can be mounted on the stator.

One possible application is to have this sensing method as an inexpensive backup in case the primary laser-interferometer sensing becomes unavailable due to signal loss. Also, when nanometer-precision is not a strict requirement, but requires large rotational motion sensing, this sensing method is very effective and can be easily implemented.

C. Suggestions for Future Work

The resolution and accuracy of the proposed method can be improved in a number of ways. The primary reason for the error is the difference between the measured magnetic flux density and the modeled, and aligning the sensors' axes with the platen's body-fixed axes is most critical. Having more basis functions in the model to compensate for such misalignments is a possible solution to improve sensing resolution and accuracy. Further adding redundant sensors is a possibility for better positioning accuracy, and this algorithm can also be expanded for 6-DOF positioning.

Theoretically, the 3 DOFs can be controlled using only two sets of two-axis Hall-effect sensors. Further investigating the reason why this did not work is for suggested future work.

Fine-tuning the DKF or any estimator is a possibility to obtain a noiseless signal. This will allow design of multivariable controller and also may attenuate the oscillation due to the noisy GLSDC output.

REFERENCES

[1] W.-J. Kim, “High-precision planar magnetic levitation,” Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, June 1997.

[2] W.-J. Kim, N. Bhat, and T. Hu, “Integrated multidimensional positioner for precision manufacturing,” *Journal of Engineering Manufacture*, vol. 218, no. 4, pp. 431–442, April 2004.

[3] X. Shan, S.-K. Kuo, J. Zhang, and C.-H. Menq, “Ultra precision motion control of a multiple degrees of freedom magnetic suspension stage,” *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 1, pp. 67–78, Mar. 2002.

[4] M. Holmes, R. Hocken, and D. Trumper, “The long-range scanning stage: A novel platform for scanned-probe microscopy,” *Precision Engineering*, vol. 24, no. 3, pp. 191–209, July 2004.

[5] Renishaw website. [Online]. Available: http://www.renishaw.com/userfiles/htm/LASWHI-0008.html [Accessed October 2004]

[6] Lion Precision website. [Online]. Available: http://www.lionprecision.com/theory [Accessed October 2004]

[7] S. Verma, W.-J. Kim, and J. Gu, “Six-axis nanopositioning device with precision magnetic levitation technology,” *IEEE/ASME Transactions on Mechatronics*, vol. 9, no. 2, pp. 384–391, June 2004.

[8] C. Schott, R. Racz, F. Betschart, and R. S. Popovic, “A new two-axis magnetic position sensor,” in *Sensors, 2002 Proceedings of IEEE*, vol. 2, June 2002, pp. 911–915.

[9] L. Law, “Measuring current with IMC Hall effect technology,” *Sensors*, vol. 3, no. 11, pp. 29–32, Nov. 2003.

[10] P. Driljace, M. Demierre, C. Schott, and R. S. Popovic, “Nonlinear effects in

magnetic angular position sensor with integrated flux concentrator," in *Proc. 23rd International Conference on Microelectronics*, vol. 1, May 2002, pp. 223–226.

[11] J. Trontelj, Jr., "Functionality test for magnetic angular positioning integrated circuit," *Informacije MIDEM*, vol. 31, no. 4, pp. 287–289, 2001.

[12] Macrosensors website. [Online]. Available: http://www.macrosensors.com [Accessed March 2004]

[13] T. Asakawa, "Two dimensional precise positioning devices for use in a semiconductor apparatus," U.S. Patent 4 535 278, Aug. 13, 1995.

[14] W. Hinds, "Single plane orthogonally movable drive system," U.S. Patent 4 654 571, Mar. 31, 1987.

[15] D. Ebihara and M. Watada, "Study of a basic structure of surface actuator," *IEEE Transactions on Magnetics*, vol. 25, no. 5, pp. 3916–3918, Sept. 1989.

[16] D. L. Trumper, W.-J. Kim, and M. E. Williams, "Magnetic arrays," U.S. Patent 5 631 618, May 20, 1997.

[17] A. J. Hazelton and J.-M. Gery, "Planar electric motor and positioning device having transverse magnets," U.S. Patent 6 285 097, Sept. 4, 2001.

[18] Y. H. Hu and J.-N. Hwang, *Handbook of Neural Network Signal Processing*, 1st ed. Boca Raton, FL: CRC Press, 2002.

[19] Sentron AG website. [Online]. Available: http://www.sentron.ch [Accessed March 2004]

[20] Vishay Siliconix website. [Online]. Available: http://www.vishay.com [Accessed August 2004]

[21] H. W. Ott, *Noise Reduction Techniques in Electronic Systems*, 2nd ed. New York, NY: John Wiley & Sons, 1988.

[22] K. Halbach, "Design of permanent multipole magnets with oriented rare earth cobalt material," *Nucl. Instrum. Methods*, vol. 169, no. 1, pp. 1–10, Feb. 1980.

[23] J. R. Melcher, *Continuum Electromechanics*. Cambridge, MA: MIT Press, 1981.

[24] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*, 2nd ed. Boca Raton, FL: Chapman & Hall / CRC Press, 2004.

[25] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 4th ed. Upper Saddle River, NJ: Prentice Hall PTR, 2002.

APPENDIX A

MATLAB AND REAL-TIME C CODES

A. MATLAB Code for Curve-Fitting

This is the Matlab program is to obtain the least squares model of sensor output a_1 using batch least squares. This is executed for each sensor output, and the model is in the form,

```
 a1hat=c+alpha*sin(w*(Y0+yA-YAoffset))+beta*sin(5*w*(Y0+yA-YAoffset))...
     +gamma*sin(w*(X0+xA-XAoffset))
```

The curve-fit result is saved as trycoefs.mat and used in the GLSDC algorithm.

```
clc;clear;close all;
tryfit; % determines the value of offset to minimize modeling error
ytilde=a1;
I=eye(4);
H=[sin(w*(Y0+yA-YAoffset)) sin(5*w*(Y0+yA-YAoffset)) ...
    sin(w*(X0+xA-XAoffset)) sin(5*w*(X0+xA-XAoffset))];
H=sbs(ones(length(ytilde),1),H);
xhat=inv(H'*H)*H'*ytilde;
% xhat=[alpha beta gamma delta]
c=xhat(1);alpha=xhat(2); beta=xhat(3);gamma=xhat(4); delta=xhat(5);
% model output
a1hat=c+alpha*sin(w*(Y0+yA-YAoffset))+beta*sin(5*w*(Y0+yA-YAoffset))...
     +gamma*sin(w*(X0+xA-XAoffset));
```

B. MATLAB Code for 3-DOF GLSDC and DKF

```
clc;clear;close all;

trycoefs; % load coefficients data obtained from batch LS

numcount=500;

% tweak these

qx=1e2; qtheta=1e5; p0=1e-8;

dt=0.00125; %800Hz

time=0:dt:dt*numcount;

zkresult=zeros(numcount+1,3);   % result of GLSDC

xkresult=zeros(numcount+1,9);   % result of EKF

xbound=1e-3;

thetabound=pi/180;

% initial guess xc

theta=0*pi/180; X0=0; Y0=-0e-3;

zk=[X0; Y0; theta];

zkresult(1,:)=zk’;

w1=1e-3;   % std of each signal from Hall sensors

W=eye(4)*(w1ˆ(-2));

% initial values for EKF

x0=[0; 0; 0]; xk=[x0; zeros(6,1)]; xkresult(1,:)=xk’;

Pk=p0*eye(9); %initial Kalman gain

Qk=zeros(3);Qk(1,1)=qx;Qk(2,2)=qx;Qk(3,3)=qtheta;

Hk=sbs(eye(3), zeros(3,6));

Phik=[eye(3)    dt*eye(3)   1/2*dtˆ2*eye(3);

    zeros(3)    eye(3)      dt*eye(3);
    zeros(3)    zeros(3)    eye(3)];
Upsk=[dt^3/6*eye(3); dt^2/2*eye(3); dt*eye(3)];
Hk=[eye(3) zeros(3,6)];
xk=zeros(9,1);

ytilde=zeros(4,1);
y=pos_to_output(0,0,0)';
J=zeros(numcount+1,1);
for count=1:numcount,
    if count>=80, y=pos_to_output(00e-6,0,1e-3)';end;
    ytilde=y+w1*randn(4,1);
    zkresult(count+1,:)=zk';

    Ct=cos(theta); St=sin(theta);
    XA = X0 + xA*Ct - yA*St; YA = Y0 + xA*St + yA*Ct;
    XB = X0 + xB*Ct - yB*St; YB = Y0 + xB*St + yB*Ct;
    wXA=omega*(XA-XAoffset); wYA=omega*(YA-YAoffset);
    wXB=omega*(XB-XBoffset); wYB=omega*(YB-YBoffset);

    CwXA=cos(wXA); CwYA=cos(wYA);
    C5wXA=cos(5*wXA); C5wYA=cos(5*wYA);
    CwXB=cos(wXB); CwYB=cos(wYB);
    C5wXB=cos(5*wXB); C5wYB=cos(5*wYB);
    SwXA=sin(wXA); SwYA=sin(wYA);
    S5wXA=sin(5*wXA); S5wYA=sin(5*wYA);
    SwXB=sin(wXB); SwYB=sin(wYB);
```

```
S5wXB=sin(5*wXB); S5wYB=sin(5*wYB);

da10dX0=gamma1*omega*CwXA + 5*delta1*omega*C5wXA;
da10dY0=alpha1*omega*CwYA + 5*beta1*omega*C5wYA;
da10dtheta=omega*(xA*Ct-yA*St)*(alpha1*CwYA+5*beta1*C5wYA)
    - omega*(xA*St+yA*Ct)*(gamma1*CwXA+5*delta1*C5wYA);
db10dX0=alpha2*omega*CwXA + 5*beta2*omega*C5wXA;
db10dY0=gamma2*omega*CwYA + 5*delta2*omega*C5wYA;
db10dtheta=-omega*(xA*St+yA*Ct)*(alpha2*CwXA+5*beta2*C5wYA)
    + omega*(xA*Ct-yA*St)*(gamma2*CwYA+5*delta2*C5wXA);
dc10dX0=gamma3*omega*CwXB + 5*delta3*omega*C5wXB;
dc10dY0=alpha3*omega*CwYB + 5*beta3*omega*C5wYB;
dc10dtheta=omega*(xB*Ct-yB*St)*(alpha3*CwYB+5*beta3*C5wYB)
    - omega*(xB*St+yB*Ct)*(gamma3*CwXB+5*delta3*C5wYB);
dd20dX0=alpha4*omega*CwXB + 5*beta4*omega*C5wXB;
dd20dY0=gamma4*omega*CwYB + 5*delta4*omega*C5wYB;
dd20dtheta=-omega*(xB*St+yB*Ct)*(alpha4*CwXB+5*beta4*C5wYB)
    + omega*(xB*Ct-yB*St)*(gamma4*CwYB+5*delta4*C5wXB);

a10 = dc1 + alpha1*SwYA + beta1*S5wYA + gamma1*SwXA +delta1*S5wXA;
b10 = dc2 + alpha2*SwXA + beta2*S5wXA + gamma2*SwYA +delta2*S5wYA;
c10 = dc3 + alpha3*SwYB + beta3*S5wYB + gamma3*SwXB +delta3*S5wXB;
d20 = dc4 + alpha4*SwXB + beta4*S5wXB + gamma4*SwYB +delta4*S5wYB;

da1dX0=da10dX0*Ct+db10dX0*St;
da1dY0=da10dY0*Ct+db10dY0*St;
```

```
da1dtheta=da10dtheta*Ct-a10*St + db10dtheta*St+b10*Ct;
db1dX0=db10dX0*Ct-da10dX0*St;
db1dY0=db10dY0*Ct-da10dY0*St;
db1dtheta=db10dtheta*Ct-b10*Ct - da10dtheta*St-a10*Ct;
dc1dX0=dc10dX0*Ct+dd20dX0*St;
dc1dY0=dc10dY0*Ct+dd20dY0*St;
dc1dtheta=dc10dtheta*Ct-c10*St + dd20dtheta*St+d20*Ct;
dd2dX0=dd20dX0*Ct-dc10dX0*St;
dd2dY0=dd20dY0*Ct-dc10dY0*St;
dd2dtheta=dd20dtheta*Ct-d20*Ct - dc10dtheta*St-c10*Ct;

H= [da1dX0 da1dY0 da1dtheta;
    db1dX0 db1dY0 db1dtheta;
    dc1dX0 dc1dY0 dc1dtheta;
    dd2dX0 dd2dY0 dd2dtheta];

a1 = a10*Ct + b10*St; b1 = -a10*St + b10*Ct;
c1 = c10*Ct + d20*St; d2 = -c10*St + d20*Ct;

fx=[a1;b1;c1;d2];
deltay=ytilde-fx;
Jbefore=deltay'*W*deltay;
Rk=inv(H'*W*H);  % covariance matrix, used in EKF
deltax=Rk*H'*W*deltay;

zk=zk+deltax;
```

```
X0=zk(1); Y0=zk(2); theta=zk(3);
St=sin(theta); Ct=cos(theta);
XA = X0 + xA*Ct - yA*St; YA = Y0 + xA*St + yA*Ct;
XB = X0 + xB*Ct - yB*St; YB = Y0 + xB*St + yB*Ct;
wXA=omega*(XA-XAoffset); wYA=omega*(YA-YAoffset);
wXB=omega*(XB-XBoffset); wYB=omega*(YB-YBoffset);
SwXA=sin(wXA); SwYA=sin(wYA);
S5wXA=sin(5*wXA); S5wYA=sin(5*wYA);
SwXB=sin(wXB); SwYB=sin(wYB);
S5wXB=sin(5*wXB); S5wYB=sin(5*wYB);

a10 = dc1 + alpha1*SwYA + beta1*S5wYA + gamma1*SwXA +delta1*S5wXA;
b10 = dc2 + alpha2*SwXA + beta2*S5wXA + gamma2*SwYA +delta2*S5wYA;
c10 = dc3 + alpha3*SwYB + beta3*S5wYB + gamma3*SwXB +delta3*S5wXB;
d20 = dc4 + alpha4*SwXB + beta4*S5wXB + gamma4*SwYB +delta4*S5wYB;
a1 = a10*Ct + b10*St;
b1 = -a10*St + b10*Ct;
c1 = c10*Ct + d20*St;
d2 = -c10*St + d20*Ct;
fx=[a1;b1;c1;d2];
deltay=ytilde-fx;
Jafter=deltay'*W*deltay;

if Jafter>Jbefore,
    zk = zk-0.8*deltax;
    X0 = zk(1); Y0 = zk(2); theta = zk(3);
    end;
    J(count+1)=deltay'*W*deltay;
    zkresult(count+1,:)=zk';

% Extended Kalman Filter
    ytildek=zk;
    % gain
    matA=Hk*Pk*Hk'+Rk;
    Kk=Pk*Hk'*inv(Hk*Pk*Hk'+Rk);
    % update
    Pkupdate=(eye(9)-Kk*Hk)*Pk;
    Pk=Pkupdate;
    xk=xk+Kk*(ytildek-Hk*xk);
    %Propagation
    xk=Phik*xk;
    Pk*Phik';
    Pk=Phik*Pk*Phik'+Upsk*Qk*Upsk';
    xkresult(count+1,:)=xk';
end;
```

C. C Code for 3-DOF GLSDC

The C codes for the GLSDC are presented. The curve-fitted values are saved in a header file hall.m. GLSDC_flag enables the GLSDC algorithm, and filter_flag enables the DKF.

```
if(GLSDC_flag==1)
{
```

```
Ct=cos(theta); St=sin(theta);

XA = X0 + xA*Ct - yA*St; YA = Y0 + xA*St + yA*Ct;

XB = X0 + xB*Ct - yB*St; YB = Y0 + xB*St + yB*Ct;

wXA=omega*(XA-XAoffset); wYA=omega*(YA-YAoffset);

wXB=omega*(XB-XBoffset); wYB=omega*(YB-YBoffset);


CwXA=cos(wXA); CwYA=cos(wYA);

C5wXA=cos(5*wXA); C5wYA=cos(5*wYA);

CwXB=cos(wXB); CwYB=cos(wYB);

C5wXB=cos(5*wXB); C5wYB=cos(5*wYB);

SwXA=sin(wXA); SwYA=sin(wYA);

S5wXA=sin(5*wXA); S5wYA=sin(5*wYA);

SwXB=sin(wXB); SwYB=sin(wYB);

S5wXB=sin(5*wXB); S5wYB=sin(5*wYB);


da10dX0=gamma1*omega*CwXA + 5*delta1*omega*C5wXA;

da10dY0=alpha1*omega*CwYA + 5*beta1*omega*C5wYA;

da10dtheta=omega*(xA*Ct-yA*St)*(alpha1*CwYA+5*beta1*C5wYA)

       - omega*(xA*St+yA*Ct)*(gamma1*CwXA+5*delta1*C5wYA);

db10dX0=alpha2*omega*CwXA + 5*beta2*omega*C5wXA;

db10dY0=gamma2*omega*CwYA + 5*delta2*omega*C5wYA;

db10dtheta=-omega*(xA*St+yA*Ct)*(alpha2*CwXA+5*beta2*C5wYA)

       + omega*(xA*Ct-yA*St)*(gamma2*CwYA+5*delta2*C5wXA);

dc10dX0=gamma3*omega*CwXB + 5*delta3*omega*C5wXB;

dc10dY0=alpha3*omega*CwYB + 5*beta3*omega*C5wYB;

dc10dtheta=omega*(xB*Ct-yB*St)*(alpha3*CwYB+5*beta3*C5wYB)
```

```
       - omega*(xB*St+yB*Ct)*(gamma3*CwXB+5*delta3*C5wYB);
dd20dX0=alpha4*omega*CwXB + 5*beta4*omega*C5wXB;
dd20dY0=gamma4*omega*CwYB + 5*delta4*omega*C5wYB;
dd20dtheta=-omega*(xB*St+yB*Ct)*(alpha4*CwXB+5*beta4*C5wYB)
       + omega*(xB*Ct-yB*St)*(gamma4*CwYB+5*delta4*C5wXB);

a10 = dc1 + alpha1*SwYA + beta1*S5wYA + gamma1*SwXA +delta1*S5wXA;
 b10 = dc2 + alpha2*SwXA + beta2*S5wXA + gamma2*SwYA +delta2*S5wYA;
 c10 = dc3 + alpha3*SwYB + beta3*S5wYB + gamma3*SwXB +delta3*S5wXB;
 d20 = dc4 + alpha4*SwXB + beta4*S5wXB + gamma4*SwYB +delta4*S5wYB;

da1dX0=da10dX0*Ct+db10dX0*St;
da1dY0=da10dY0*Ct+db10dY0*St;
da1dtheta=da10dtheta*Ct-a10*St + db10dtheta*St+b10*Ct;
db1dX0=db10dX0*Ct-da10dX0*St;
db1dY0=db10dY0*Ct-da10dY0*St;
db1dtheta=db10dtheta*Ct-b10*Ct - da10dtheta*St-a10*Ct;
dc1dX0=dc10dX0*Ct+dd20dX0*St;
dc1dY0=dc10dY0*Ct+dd20dY0*St;
dc1dtheta=dc10dtheta*Ct-c10*St + dd20dtheta*St+d20*Ct;
dd2dX0=dd20dX0*Ct-dc10dX0*St;
dd2dY0=dd20dY0*Ct-dc10dY0*St;
dd2dtheta=dd20dtheta*Ct-d20*Ct - dc10dtheta*St-c10*Ct;

matH[0][0]=da1dX0; matH[0][1]=da1dY0; matH[0][2]=da1dtheta;
matH[1][0]=db1dX0; matH[1][1]=db1dY0; matH[1][2]=db1dtheta;
```

```
matH[2][0]=dc1dX0; matH[2][1]=dc1dY0; matH[2][2]=dc1dtheta;
matH[3][0]=dd2dX0; matH[3][1]=dd2dY0; matH[3][2]=dd2dtheta;

a1 = a10*Ct+b10*St; b1 = b10*Ct-a10*St;
c1 = c10*Ct+d20*St; d2 = d20*Ct-c10*St;
fx[0]=a1; fx[1]=b1; fx[2]=c1; fx[3]=d2;

deltay[0]=ytilde[0]-fx[0];
deltay[1]=ytilde[1]-fx[1];
deltay[2]=ytilde[2]-fx[2];
deltay[3]=ytilde[3]-fx[3];

Jbefore=deltay[0]*deltay[0]+deltay[1]*deltay[1]
       +deltay[2]*deltay[2]+deltay[3]*deltay[3];

/* solve for matH'*matW*matH=HTWH */
for(i=0;i<=2;i++){
   for(j=i;j<=2;j++){
      HTWH[i][j]=0;
      for(k=0;k<=3;k++){
         HTWH[i][j]=HTWH[i][j]+matH[k][i]*matH[k][j]*W[k];
      }
      HTWH[j][i]=HTWH[i][j];
   }
}
/* solve for inv(matH'*matW*matH)=Rk */
```

```
det=HTWH[0][0]*HTWH[1][1]*HTWH[2][2]
   + HTWH[1][0]*HTWH[2][1]*HTWH[0][2]
   + HTWH[2][0]*HTWH[0][1]*HTWH[1][2]
   - HTWH[0][0]*HTWH[2][1]*HTWH[1][2]
   - HTWH[2][0]*HTWH[1][1]*HTWH[0][2]
   - HTWH[1][0]*HTWH[0][1]*HTWH[2][2];
Rk[0][0]=(HTWH[1][1]*HTWH[2][2]-HTWH[1][2]*HTWH[2][1])/det;
Rk[0][1]=(HTWH[0][2]*HTWH[2][1]-HTWH[0][1]*HTWH[2][2])/det;
Rk[0][2]=(HTWH[0][1]*HTWH[1][2]-HTWH[0][2]*HTWH[1][1])/det;
Rk[1][1]=(HTWH[0][0]*HTWH[2][2]-HTWH[0][2]*HTWH[2][0])/det;
Rk[1][2]=(HTWH[0][2]*HTWH[1][0]-HTWH[0][0]*HTWH[1][2])/det;
Rk[2][2]=(HTWH[0][0]*HTWH[1][1]-HTWH[0][1]*HTWH[1][0])/det;
Rk[1][0]=Rk[0][1];
Rk[2][0]=Rk[0][2];
Rk[2][1]=Rk[1][2];

/* calculate inv(matH'*matW*matH)*matH'*matW=Rk*matH'*matW=RHTW */
for(i=0;i<=2;i++){
   for(j=0;j<=3;j++){
      RHTW[i][j]=0;
      for(k=0;k<=2;k++){
         RHTW[i][j]+=Rk[i][k]*matH[j][k]*W[j];
      }
   }
}
```

```
deltax[0] = RHTW[0][0]*deltay[0] + RHTW[0][1]*deltay[1]
          + RHTW[0][2]*deltay[2] + RHTW[0][3]*deltay[3];
deltax[1] = RHTW[1][0]*deltay[0] + RHTW[1][1]*deltay[1]
          + RHTW[1][2]*deltay[2] + RHTW[1][3]*deltay[3];
deltax[2] = RHTW[2][0]*deltay[0] + RHTW[2][1]*deltay[1]
          + RHTW[2][2]*deltay[2] + RHTW[2][3]*deltay[3];

/* solve for Jafter */
X0 = X0 + deltax[0]; Y0 = Y0 + deltax[1];
theta = theta + deltax[2];

Ct=cos(theta); St=sin(theta);
XA = X0 + xA*Ct - yA*St; YA = Y0 + xA*St + yA*Ct;
XB = X0 + xB*Ct - yB*St; YB = Y0 + xB*St + yB*Ct;
wXA=omega*(XA-XAoffset); wYA=omega*(YA-YAoffset);
wXB=omega*(XB-XBoffset); wYB=omega*(YB-YBoffset);

SwXA=sin(wXA); SwYA=sin(wYA);
S5wXA=sin(5*wXA); S5wYA=sin(5*wYA);
SwXB=sin(wXB); SwYB=sin(wYB);
S5wXB=sin(5*wXB); S5wYB=sin(5*wYB);
a10 = dc1+alpha1*SwYA+beta1*S5wYA+gamma1*SwXA+delta1*S5wXA;
b10 = dc2+alpha2*SwXA+beta2*S5wXA+gamma2*SwYA+delta2*S5wYA;
c10 = dc3+alpha3*SwYB+beta3*S5wYB+gamma3*SwXB+delta3*S5wXB;
d20 = dc4+alpha4*SwXB+beta4*S5wXB+gamma4*SwYB+delta4*S5wYB;
a1 = a10*Ct+b10*St; b1 = b10*Ct-a10*St;
```

```
   c1 = c10*Ct+d20*St; d2 = d20*Ct-c10*St;
   fx[0]=a1; fx[1]=b1; fx[2]=c1; fx[3]=d2;

   deltay[0]=ytilde[0]-fx[0];
   deltay[1]=ytilde[1]-fx[1];
   deltay[2]=ytilde[2]-fx[2];
   deltay[3]=ytilde[3]-fx[3];
   Jafter=deltay[0]*deltay[0]+deltay[1]*deltay[1]
          +deltay[2]*deltay[2]+deltay[3]*deltay[3];

   if(Jafter>Jbefore){ /*alpha=0.2*/
      X0 = X0 - 0.8*deltax[0];
      Y0 = Y0 - 0.8*deltax[1];
      theta = theta - 0.8*deltax[2];
   }
} /* end GLSDC */
```

VITA

The author, Yusuke Kawato, was born on May 30, 1980 in Yokohama, Japan. He graduated from Keio University, Yokohama, Japan with the Bachelor of Engineering degree in mechanical engineering in March 2003. He began his graduate research at Texas A&M University in September 2003.

His permanent address is:

2-25-1 Terabun
Kamakura, Japan 247-0064

His e-mail address is:

ykawato@2003.jukuin.keio.ac.jp