5-2011

System integration for a novel positioning system using a model based control approach

Chan yet Wong
Clemson University, chan.wong@gmail.com

SYSTEM INTEGRATION FOR A NOVEL POSITIONING SYSTEM USING A MODEL BASED CONTROL APPROACH

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Automotive Engineering

---

by
Chan Yet Wong
May 2011

Accepted by:
Dr. Laine Mears, Committee Chair
Dr. John Ziegert
Dr. Ardalan Vahidi
Dr. Mary Beth Kurz

ABSTRACT

This dissertation presents a model-based approach to perform system integration of a novel positioning sensing method, termed “Direct Position Sensing.” Direct Position Sensing can actively monitor the planar position changes of motion control devices without the dependency of the conventional position sensor combined with kinematic model to estimate the planar position. Instead, Direct Position Sensing uses the technology of computer vision and digital display to directly monitor the planar position displacement of a motion control device by actively tracking the desired position of the device based on the displayed target showed on the digital screen. The integration of the computer vision as the feedback system to the motion controller, introduces intermittency and latency in the controller’s feedback loop.

In order to integrate the slower computer vision sensor to the motion controller, a model-based controller architecture, Smith Predictor approach was first implemented to the Direct Position Sensing system. The Smith Predictor uses a mathematical plant model that is running in parallel with the actual plant so that the model predicts the plant output when the actual output of the system is unavailable. Due to the intermittency feedback of the system, a path prediction algorithm was developed to minimize the model residual during the intermittent feedback so that the tracking performance of the system can be improved. Furthermore, a model input corrector was also developed to correct the control action to the plant model based on the model residual to enhance the path prediction.

Simulations and hardware experiments results show that the model-based strategy provides improved tracking performance of the system when latency and intermittency exist in the controller feedback loop.

DEDICATION

I would like to dedicate this dissertation to my parents, wife and friends who have consistently provided support, love and encouragement towards helping me to pursue my dreams and goals.

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor Dr. Laine Mears for his energetic, positive and patience guidance not only in research, but also in my attitude towards problem solving and project management.

My thanks go to my committee members, Dr. John Ziegert, Dr. Ardalan Vahidi, and Dr. Mary Beth Kurz for their suggestions and teaching during my studies at Clemson University. I would like to thank my friends and peers: Dr. Laurence Snider, Carlos Montes, Robert Clippard, Joshua Tarbutton, John Limroth, Vincent Lee, Cristina Bunget and the entire AMMIL Group.

I would also like to acknowledge National Science Foundation for funding the project, National Instruments for their sponsors, and the entire faculty and staff of Clemson University International Center for Automotive Research (CU ICAR) for their great help and company.

Last but not least, I would like to express my thanks to Clemson University and the Automotive Department for giving me such a wonderful experience in pursuing my degree at a wonderful facility.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

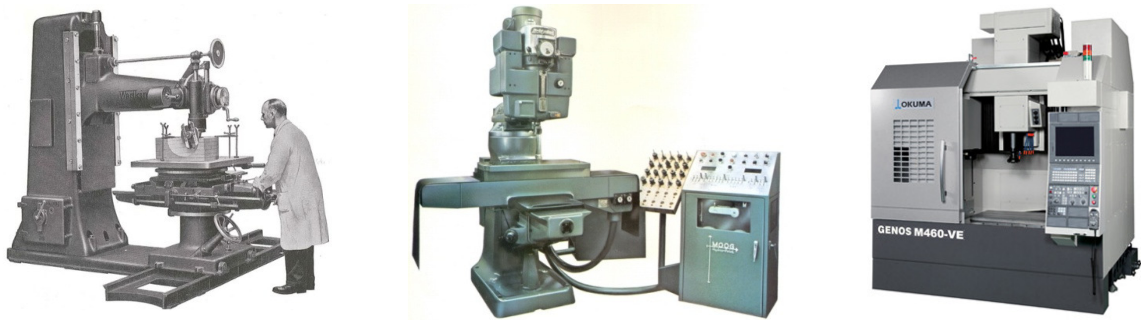
INTRODUCTION

1.1 Objective

The objective of this research is to use model based approaches to perform system integration for a novel positioning sensing system. Instead of using conventional position sensors, this novel positioning system uses computer vision together with a digital display to actively track the planar position of a motion control devices. Thus, system integration between the newly developed vision sensor and the motion controller must be performed. This research uses the Computer Numerical Control (CNC) XY table as an application example to show the proof of concept of the novel positioning system.

1.2 Motivation

Machining technology has advanced from manual machines, through Numerical Control (NC) machines and to the CNC machines of today, as represented in Figure 1-1 respectively. The main goal of advancement and continuing development of machine tools is to create more accurate and faster machines so that complicated parts can be manufactured efficiently and with greater precision. Almost all CNC machines in the manufacturing industry use position sensors such as rotary encoder or linear scale to monitor the position of each machine's axis to provide position and velocity feedback to the motion controller so that the desire path of the part can be tracked accurately. However, machine errors related to geometric inconsistencies, kinematic errors and thermal distortion exist during the machining operation, affecting the machine accuracy.



A) Manual machine

B) NC machine

C) CNC machine

Figure 1-1: Machine tool advancements [1-3]. The control of metal cutting has evolved from a primarily A) manual operation to B) semi-automatic axis control and then to C) fully automatic axis control machines, which has improved accuracy and faster productivity

CNC machines mainly operate in an offline error compensation architecture in which the inaccurate machine has to be shut down in order to be re-calibrated to factory's design specification. This process is time consuming, expensive and inefficient for any manufacturing facility. In addition, most of the previous and current research related to machine accuracy improvement involves adding more sensors and employing additional compensation algorithms over the existing position control schemes. However, such schemes are still unable to directly locate the actual toolpoint of the system. Although each machine axis’s position sensing system operates using closed-loop control, the position sensing system of the planar or spatial location is still operated in an open-loop manner. This is because the planar or spatial positions are estimated through a kinematic model of the machine based on the feedback from each axis’s position sensor. Such a design is subject to tool point position error which leads to inaccuracies in producing high precision parts, resulting in higher loss to the company.

Thus, this research presents a new position sensing architecture termed “*Direct Position Sensing”* that can actively monitor the actual planar toolpoint of an automatically- controlled positioning machine without relying on a kinematic model. The success of this new algorithm will enable the machine to accurate track the desired trajectory while eliminating the offline error compensation and mapping process commonly used to calibrate the machine. This approach will help the manufacturing industry save cost in terms of labor, energy consumption, and material resources in producing accurate and high quality parts.

1.3 Research Challenges

Instead of using an optical sensor, Direct Position Sensing utilizes a computer vision sensor to monitor the planar position of the CNC machine. This computer vision sensor consists of a digital display screen, a digital camera and an image processing micro-controller to track the planar location of the machine as shown in Figure 1-2.

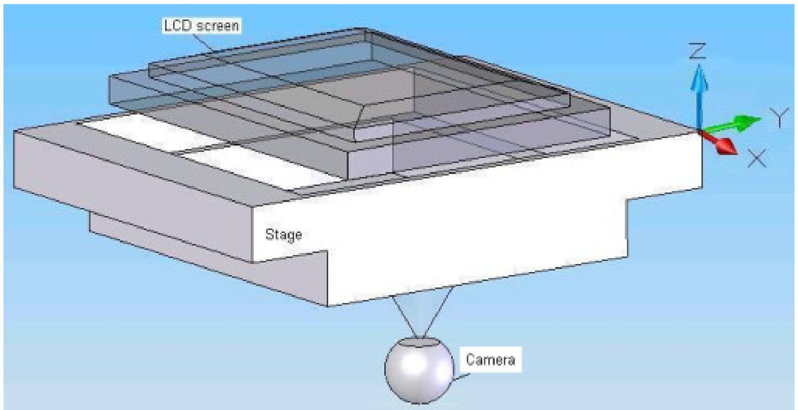


Figure 1-2: Vision sensor configuration[4]. A digital screen that is located on top of the XY table, is used to display desired trajectory of the system and the digital camera that is mounted at the center of the table, is used to monitor the planar displacement

In this research, it is assumed that 1) there exists a constant time delay, τ within the feedback loop owing the long image processing time of the vision sensor, and 2) no output will be generated during the image processing period, which leads to an intermittent feedback. Therefore, the feedback of the vision sensor to the motion controller is assumed to be delayed and intermittent, which is not only detrimental to the path tracking of Direct Position Sensing but also can cause the system to be unstable.

The primary objective is to develop a model-based control approach to predict the actual tool path of the machine during periods between intermittent feedbacks of position, and to correct the prediction when feedback is received. This approach is phased in time to account for the delay in feedback owing to image processing time.

1.4 Dissertation organization

The remainder of the dissertation is organized as follows:

- Chapter 2 presents the background and literature reviews of machine positioning error and error compensation techniques
- Chapter 3 presents the system design of the newly-developed positioning system, the fundamental intellectual questions to be answered.
- Chapter 4 presents the literature survey of model-based control, and its application to CNC and multiaxis positioners.
- Chapter 5 presents the Smith Predictor approach to integrate both the vision system and the motion controller
- Chapter 6 presents the Modified Smith Predictor that was developed to minimize the model residual that exists in the original Smith Predictor
- Chapter 7 provides the conclusion and future directions necessary in this research.

CHAPTER TWO

BACKGROUD AND LITERATURE REVIEW

2.1 Computer Numerical Control Machine

This chapter first presents a brief introduction of CNC machine, machine errors, and error compensation methods to overcome the machine error. The CNC machine consists for three main components: 1) machine structures, 2) driving system and 3) controller [5].

2.1.1 Machine Structures

Typically, a CNC machine has static machine structures such as machine columns and beds that serve as the chassis of the machine to ensure the stiffness of the machine so that the dynamic structures such as machine axis, spindle, table and other moveable structures can be sturdily mounted on the static structures as presented in Figure 2-1.

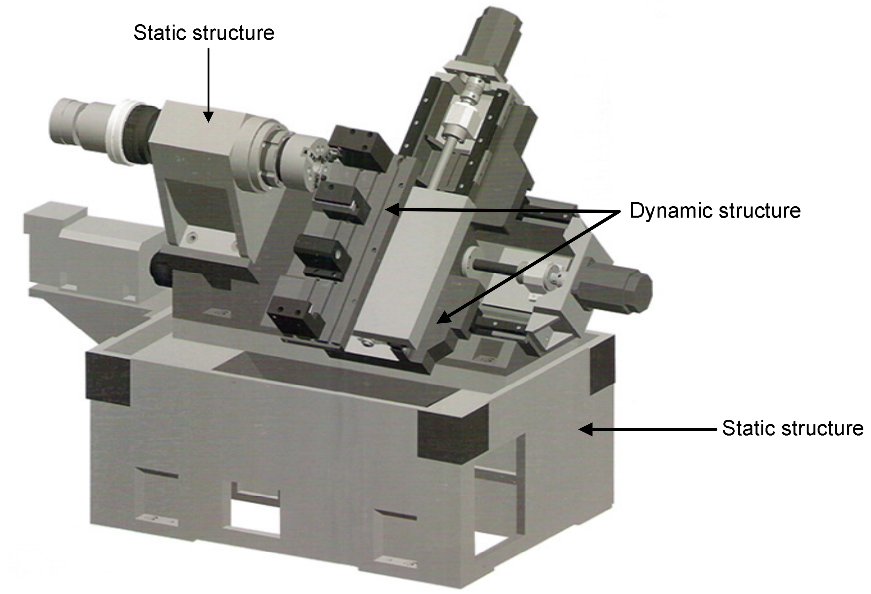


Figure 2-1: Machine structures [6]. Static structures serve as a rigid framework on which to mount the dynamic structures such as spindle and articulating axes

2.1.2 Drive System

The CNC's drive system has three main components: motor, amplifier, and sensor. The motor is used to drive not only the axis of the machine but also the spindle of the machine. In order to provide an appropriate current to the motor, an amplifier is needed to monitor the transmitted current, *i (t)* to the motor based on the command signal from the controller, *u(t)* as seen in Figure 2-2. Then, the position sensor is used to measure the position, $\theta(t)$ or velocity, $\omega(t)$ of axes' motor and feed the measured signal back to the controller so that the tracking error, *e(t)* of each axis can be obtained [5].

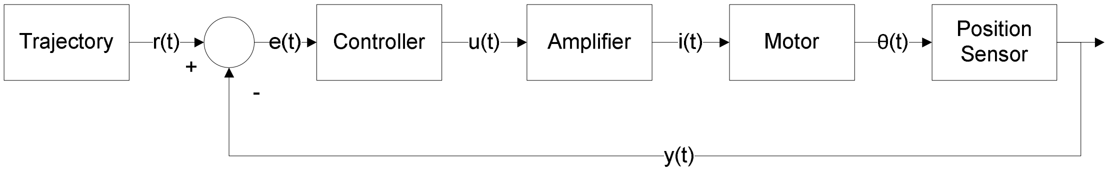


Figure 2-2: Closed loop control of a single axis. Based on the error between the trajectory, r(t) and position feedback, y(t), the controller output a proper control action, *u(t)* to the amplifier which generate the current, *i(t)* to actuate the motor

2.1.3 Controller

The controller is used to ensure the drive system follows the pre-defined trajectory or setpoints of the system with minimal error. Although various control architectures have been developed, the Propotional Integral Derivative (PID) controller is still the most commonly used control architecture in the industy due to its simplicity and easy implimentation. Eq. (2.1) shows the general PID equation and can be rewritten as Eq.(2.2) which is the more commonly used form in the industry. T_i represents the integral time and T_D represents the derivative time , their relationship with the proportional gain, K_P are shown in Eq. (2.3) and (2.4) respectively.

$$C(s) = K_p + \frac{K_I}{s} + K_D s \qquad (2.1)$$

$$C(s) = K_p(1 + \frac{1}{T_I s} + T_D s) \qquad (2.2)$$

$$T_I = {K_P}/{K_I} \qquad (2.3)$$

$$T_D = {K_D}/{K_P} \tag{2.4}$$

The output of the control action of the controller depends on both the gains of the PID controller and also the values of the measurement's error obtained from the sensor. In general, K_p is proportional to the measured error, so the bigger the K_P, the faster the output response is going to be generated. However, if the K_P value is too big, this can lead to instability and resultant oscillatory behavior. The K_I is the integral control that is used to sum up the error during the integral time so that the steady state error can be reduced. The larger the K_I, the faster the response reaches the setpoints, but improper settings of the K_I will lead to overshoot or undershoot responses. Lastly, K_D is proportional to the rate of change of the process variable [7]. Large K_D can cause vibration in the system because the first derivative of the position error is sensitive to noise [8]. Typically in the industry, the PI controller is widely used because there is difficulty in tuning the K_D; improper tuning or system temporal change can cause heavy vibration [5].

2.2 Feedforward Control

Although the conventional position control method is able to track the desired position of each axis, following error still exists. Following error in this context represents the deviation between the setpoints position and the actual position that is due to servo lag. As seen in Figure 2-3, servo lag happens when the motor does not have sufficient speed of response to reach the desired velocity when it starts running from a stop position. The following error impact to path tracking is illustrated in Figure 2-4.

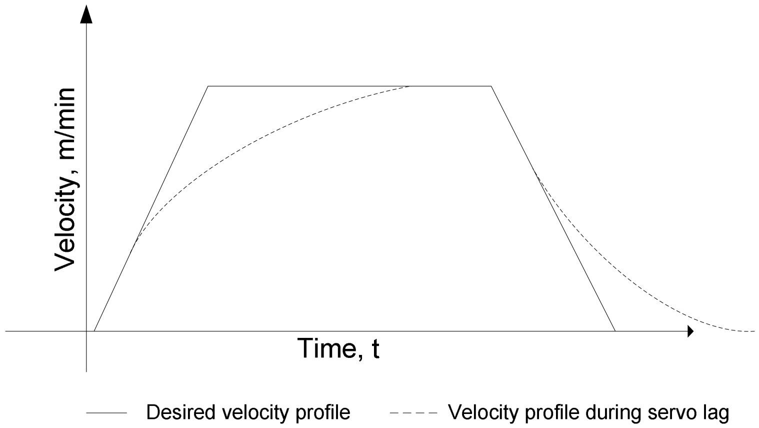


Figure 2-3: Velocity profile during servo lag. As seen in the diagram the broken line represent the servo lag of the system when the motor starting from a static position, the lag of not reaching the velocity profile fast enough causes the following error to happen

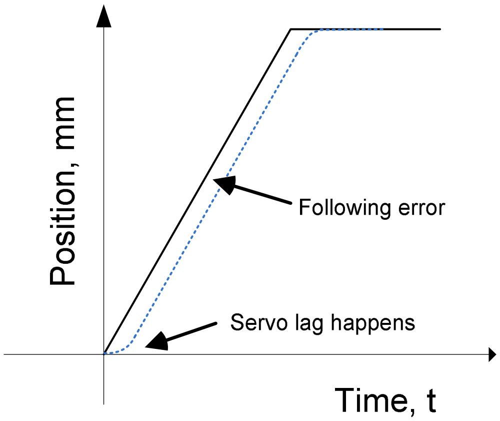


Figure 2-4: Following error due to servo lag. The solid line represents the setpoints of the system and the dotted line represents the response of the motor and the following error of the system is the vertical distance between the setpoints and response

Often, a velocity feedforward control algorithm will be added to the conventional motion controller to speed up the response, allowing the motor to reach its velocity profile faster and minimize the servo lag. Unlike conventional position control that generates control action based on the obtained feedback, the feedforward operates based on a pre-specific model [8]. Typically, the inverse transfer functions of the plant will be used as the model for the velocity feedforward controller. With the model based approach, model discrepancy is expected, so the feedforward controller is frequently used together with the feedback control architecture so that the model discrepancy from the feedforward controller can be corrected in the feedback loop as shown in Figure 2-5.

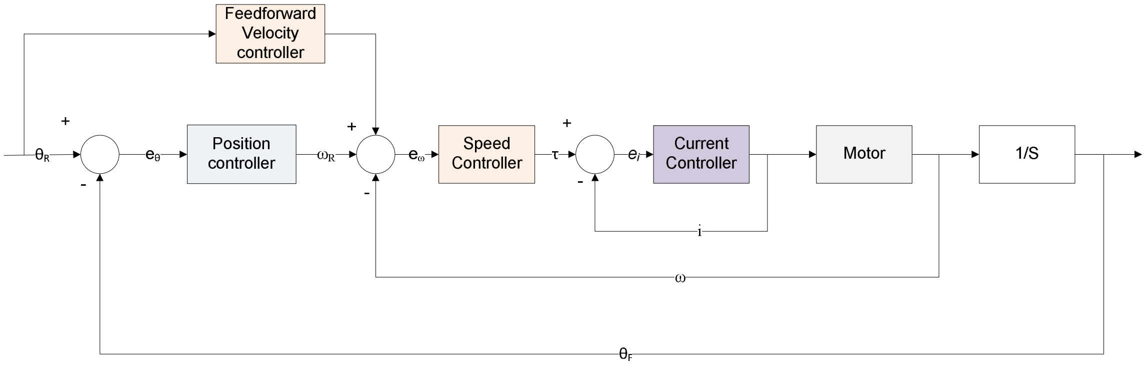


Figure 2-5: Velocity feedforward control. The feedforward controller added to the conventional controller to improve the system response of the system minimizing the following error.

2.3 Machine Error

Accuracy is one of the most crucial performance quantification metric of a machine tool in order to produce high precision and good surface finish products. In the manufacturing industry, accuracy is defined as the degree of conformance of a finished part with the required dimensional and geometrical accuracy [9]. Error in this context is described as the position deviation of the cutting edge from the theoretical value to the desired tolerance of the workpiece [10]. However, the minimum error that can exist in a system is only depends on the system's resolution. Furthermore, a more important factor than the system resolution is relied on accuracy of the sensor that is used to measures the actual point [11]. Hence, one of the suggested methods to keep track of machine error is to formulate an error budget which allocate and predict the error source of a machine [12].

There are two fundamental categories of machine errors: quasi-static and dynamic errors [10]. Quasi-static errors involve errors generated by wears and misalignment of the tool and workpiece which gradually changes with time [10]. Geometric errors, kinematic errors, and thermal errors are the most commonly known quasi-static errors. These errors may not be sensed by independent axis encoders as the axes are subject to the same error, causing the workpiece to be inaccurate. Dynamic errors are mainly related to the error generated due to the operating condition and the configuration of the machine tools, such as vibration of the machine structure due to improper setting of the spindle rate or feedrate of the machine [10].

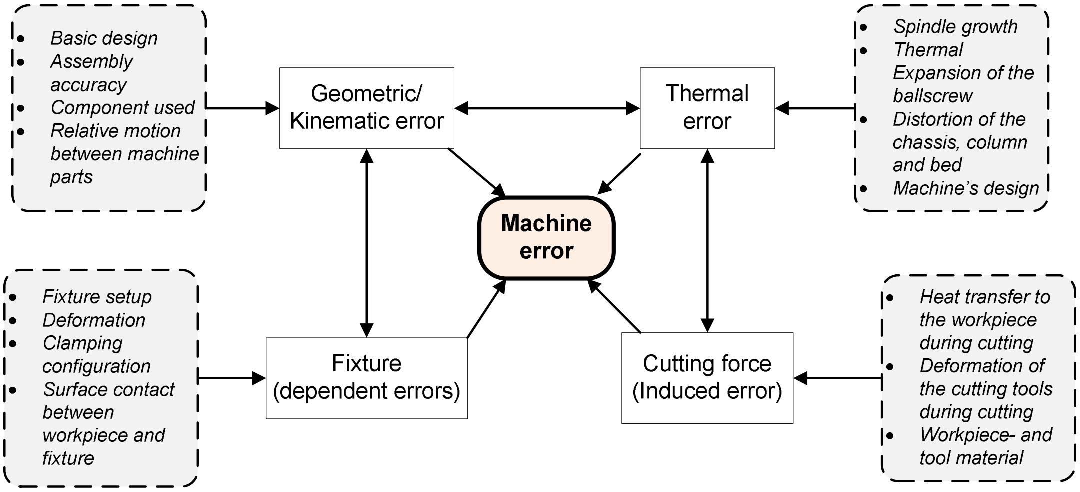

Basic design
Assembly accuracy
Component used
Relative motion between machine parts
Geometric/ Kinematic error
Thermal error
Spindle growth
Thermal Expansion of the ballscrew
Distortion of the chassis, column and bed
Machine's design
Machine error
Fixture setup
Deformation
Clamping configuration
Surface contact between workpiece and fixture
Fixture (dependent errors)
Cutting force (Induced error)
Heat transfer to the workpiece during cutting
Deformation of the cutting tools during cutting
Workpiece- and tool material

Figure 2-6: Error budget [10]. Machine errors of a machine tool and the factor affecting it

2.3.1 Geometric Error

Geometric error is the error associated with the misalignment of a machine component such as the straightness and flatness of a machine axis, e.g. the straightness and angular error of the X axis with respect to the X, Y and Z direction, as shown in Figure 2-7 [13]. In a typical 3-axis Cartesian machine such as a milling machine, the tool coordinates are directly obtained from the readings from each of the X, Y, and Z axis's position sensor. However in the actual machine, each of these axes is not perfectly accurate; they pose six components of small positioning errors on each axis as shown in Figure 2-7 [13]. Thus, for a three axis CNC machine, there are eighteen small geometrical errors.

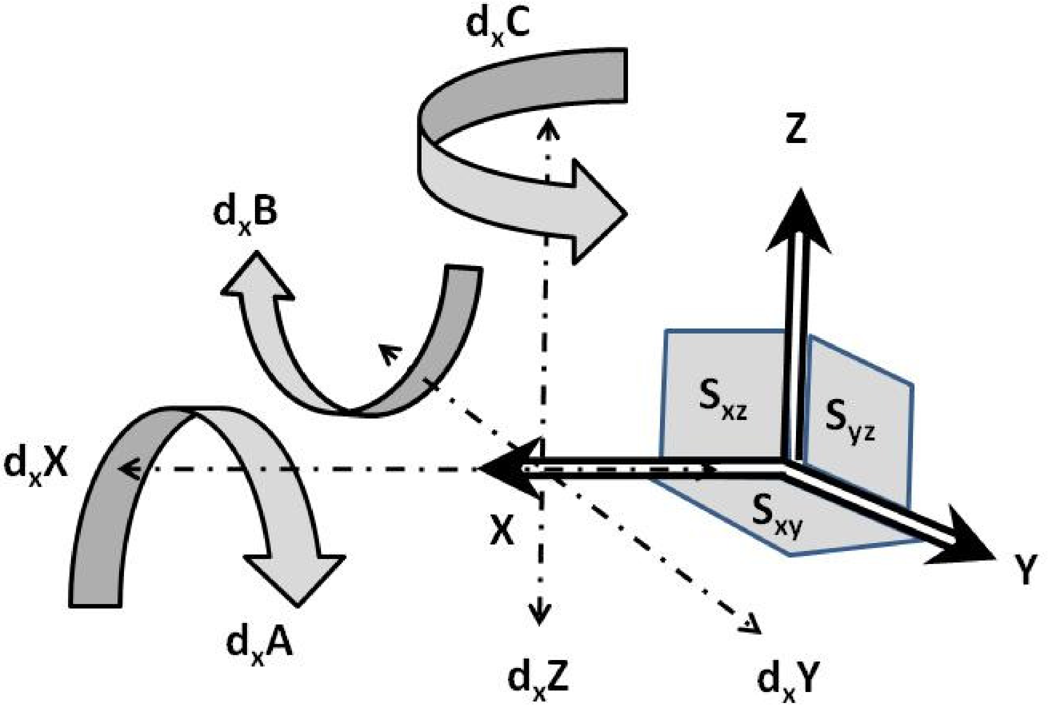


Figure 2-7: Geometrical and kinematics error. This diagram shows the associated geometric and kinematic error of the X-axis of a three axis machine

Table 2-1: Nomenclatures of the geometric error shown in Figure 2-7

Variables	Definition	Types of error
d_xX	• Straightness error of X axis with respect to X axis	• Geometric error
d_xY	• Straightness error of X axis with respect to Y axis	• Geometric error
d_xZ	• Straightness error of X axis with respect to Z axis	• Geometric error
d_xA	• Angular error of X axis with respect to X axis	• Geometric error
d_xB	• Angular error of X axis with respect to Y axis	• Geometric error
d_xC	• Angular error of X axis with respect to Z axis	• Geometric error
S_{xy}	• Squareness error between X- and Y- axis	• Kinematic error
S_{xz}	• Squareness error between X- and Z- axis	• Kinematic error
S_{yz}	• Squareness error between Y- and Z- axis	• Kinematic error

2.3.2 Kinematic Error

Kinematic error is related to the motion error when the two or more axes are used to position the desired planar or spatial location, e.g., squareness errors, S_{XY}, S_{XZ} and S_{YZ} shown in Figure 2-7 [13]. Kinematic error is also highly affected by the geometric error. In the as-built machine, the axes directions will not be perfectly orthogonal and cause squareness errors between the coupled axes. Some of these errors are caused by backlash error, machine deflection error and assembly flaws [13, 14].

2.3.3 Thermal Error

Besides the geometric and kinematic errors, thermal error is also one of the main factors for machine tool accuracy [15, 16]. The continuous operation of machines during the manufacturing process generates heat which can causes expansion on various machine tool components. The machine structures' expansion causes the machine positioning system to be inaccurate. There are six major known thermal sources within the machine tool: 1) cutting process heat, 2) heat generated by the machine, 3) machine's cooling system, 4) work cell's temperature, 5) heat generated by operator and people around the machine, 6) thermal memory of the previous environment [16].

There are two types of thermal errors: 1) position independent thermal error and 2) position dependent thermal error [15, 17]. The position independent thermal error is related to the thermal expansion of the static structure such as the beds, and columns of the machine structure, that are mainly caused the by the surroundings temperature. The position dependent thermal error is the error which occurs on the travel axis of the machine in which the thermal error changes as a function of axis position and temperature [14, 18]. As an example of position dependent thermal error, Figure 2-8 presents the thermographic snapshot of a moving ball screw during a multipass milling operation at a mean feed rate of 10 m/min over 20 minutes[19]. Based on the captured snapshot, it can be seen that there was a significant and heterogeneous temperature changes along the ballscrew ranging from approximately 28 ^{o}C to 50 ^{o}C [19]. In addition, it can also be observed that the temperature increment was only located at one significant portion of the ballscrew, which is the path where the machine travels most frequent.

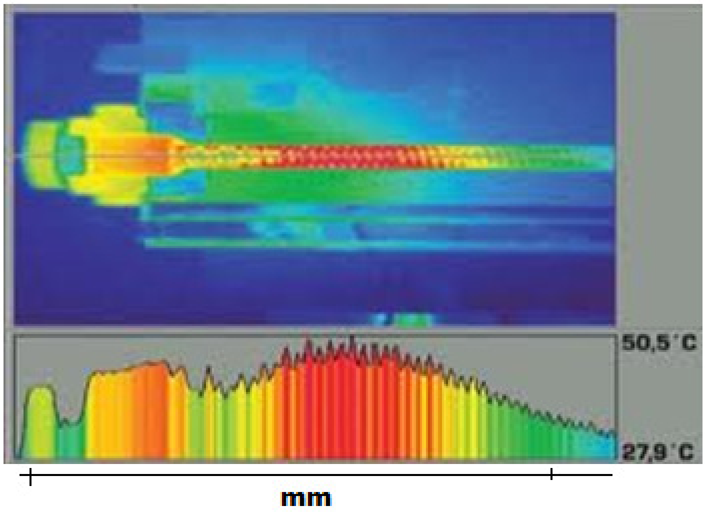
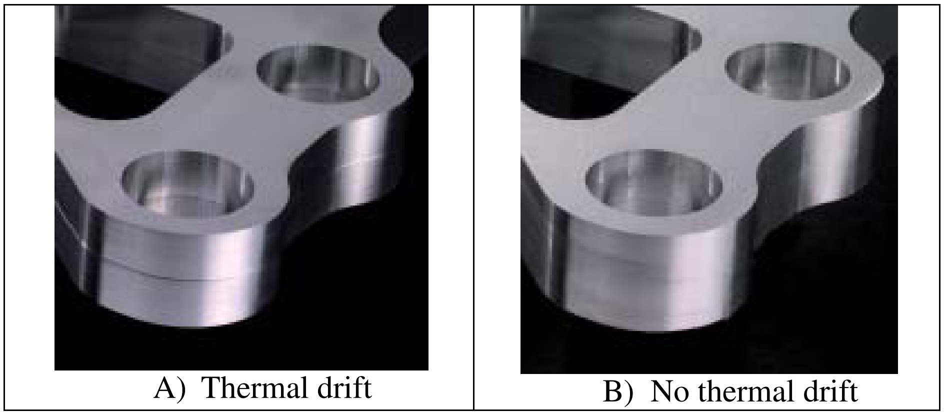


Figure 2-8: Thermal expansion of moving ball screw. The temperature of the ballscrew with respect to its absolute location during a multipasses milling process at a mean feedrate of 10 m/min within 20 minutes [19]



A) Thermal drift

B) No thermal drift

Figure 2-9: Parts comparison due to the effect of thermal drift. The two parts are presented which were manufactured before and after thermal stabilization of the CNC, the machined part shown in A) has an error of 44 μm and is mainly due to the thermal expansion of the machine structure [19]

The impact of thermal expansion is unfavorable to machining processes. As seen at Figure 2-9, the contour of the thermal affected workpiece in Figure 2-9A deviated by 44μm from the non-affected workpiece in Figure 2-9B. Consequently, the tool point coordinates estimated from the axis positions and nominal kinematic model will be slightly incorrect due to the expansion of the axes and the impact of the thermal expansion.

2.3.4 Cutting Force Induced Error

Although much research has been performed on the compensating error caused by geometric, kinematic and thermal sources, not much has been carried out in compensating the error caused by the cutting force [10]. During the machining process for hardened steel, the part is commonly machined to its final form. This type of machining process operates at a very high cutting force in which the amount of force acting on the workpiece need to be taken into consideration [20]. Due to the high cutting force used during the machining process, deformation of the workpiece and tool can happen and affects the workpiece geometry accuracy.

2.3.5 Fixture Error

The fixture-tool workpiece system is also one of the most important factors to ensure the overall accuracy of the workpiece. During a machining operation, a fixture is used to position and hold the workpiece from moving due to the excessive force acting on it. Hence, the accuracy of the machine workpiece is depending on the sturdiness of the fixture to restrain the workpiece from moving during the machining process [21]. Workpiece displacement can be caused by inadequate clamping force of the fixture, which also can generate the geometric error of the workpiece. Studies also show that the workpiece displacement can be affected by the clamping sequence, clamp's geometry and location [22].

2.4 Error Compensation

During the machining operation, the cutting motion and thermal condition of the machine tools varies with time, making the error source also changes with time [10]. The time-varying factor of the error is one of the barriers to accurately track and compensate the machine error, leading to machine inaccuracy [23]. There are two alternatives to improve the accuracy of the CNC machine: 1) error avoidance and 2) error compensation [14]. The error avoidance method is to build a machine that is mechanically accurate in term of all the structures of the machine [10, 24, 25]. However, this method is expensive owing to the use of high-precision components to build the machine, and operation in a temperature-controlled environment in order for the accuracy to be maintained. In addition, it is impossible to build a machine that is error free.

On the other hand, error compensation is a more commonly used method that is in the industry to calibrate the machine while improving the machine's accuracy. The goal of error compensation is to first analyze and inspect the error source of the machine and then correct the measured error using suitable algorithms based on the types of machine error sources [14]. However, machine accuracy can be affected not only by a single error source, but by the combination of various error sources mentioned above. As a result, all the existing machine tool errors have to be taken into consideration when designing the error compensation algorithms.

There are two categories: "pre-calibrated error compensation" and "active error compensation [14]." Pre-calibrated error compensation also termed offline error compensation, is a method to calibrate the machine tool when the machine is not in operation, either before or after the operation. However, the machining process and measurement has to be repeatable for this algorithm to accurately compensate the error. On the other hand, the active error compensation method is used to correct the measured error during the machine operation [26]. The benefit of this method is that the workpiece's accuracy can be improved by implementing the error compensation algorithms on the machines, which is why this method is attractive to the industry. In addition, active error compensation method not only enhance the machine accuracy but also has justified installation and calibration cost [14]. Another alternative is to compensate the error in real time during the machining processes by observing the interaction between ranges of error components within the machining processes, termed real-time error compensation or active error compensation [27]. There are two basic real-time error compensation approaches: parametric error measurement approach and master part tracking approach [14].

2.4.1 Parametric Error Measurement

The parametric error measurement approach consists of three major areas of activity: 1) Error identification and modeling, 2) Error mapping and 3) Error compensation via add-on control algorithms [10].

2.4.1.1 Error Identification and Modeling

There are five commonly known error identification methods used: 1) error synthesis, 2) grid calibration, 3) design artifact, 4) metrology frame, and 5) finite element [10]. Only the error synthesis model, in which the method acquires the total error in terms of individual error components, and then provide the compensation scheme to compensate the quasi-static error [28]. It is also known that the empirical modeling approaches: the regression analysis and feedforward using neural-network could be used to perform error mapping. Due to this model-based approach, the error synthesis modeling is sensitive to the location of the sensor and also requires more time to perform data acquisition and data training in order to formulate a good machine tool model [10].

2.4.1.2 Error Mapping and Compensation

Error mapping is used to quantify each error source of the machine. The laser interferometer as shown in Figure 2-10 is the more precise and commonly used method to measure most error components on a CNC or a CMM machine, such as the straightness, flatness and squareness of the axes. When the error map of the axis is obtained, the measured error will be inverted so that the error of each absolute position can be corrected though the pitch compensation algorithm of CNC controller as shown in Figure 2-11.

However, the laser interferometer is not able to measure the rotating component's position, such as the machine spindle; only the non-contact capacitance sensor can be used to quantify the spindle errors [28]. Although the laser interferometer can also be used to measure the position error cause by the thermal expansion of the machine, the measurements of the affected area is not going to be repeatable due to the non-linearity of the thermal source. Thus, laser interferometer is not suitable to be used to calibrate the thermally induced error on a machine. In general, a mathematical model is needed to provide the relationship between thermal displacement and temperature [9].

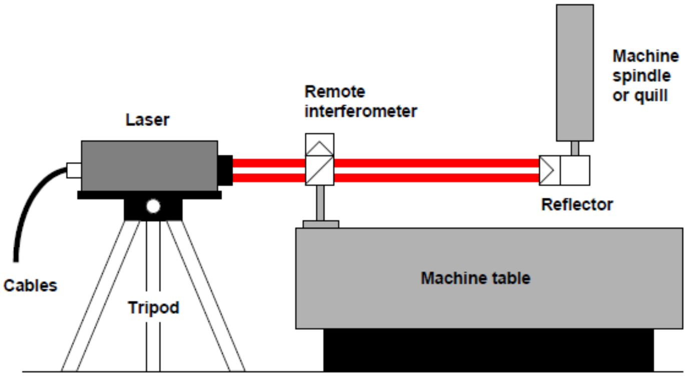


Figure 2-10: Laser Interferometer [29]. Configuration of a laser interferometer to perform error mapping on a machine

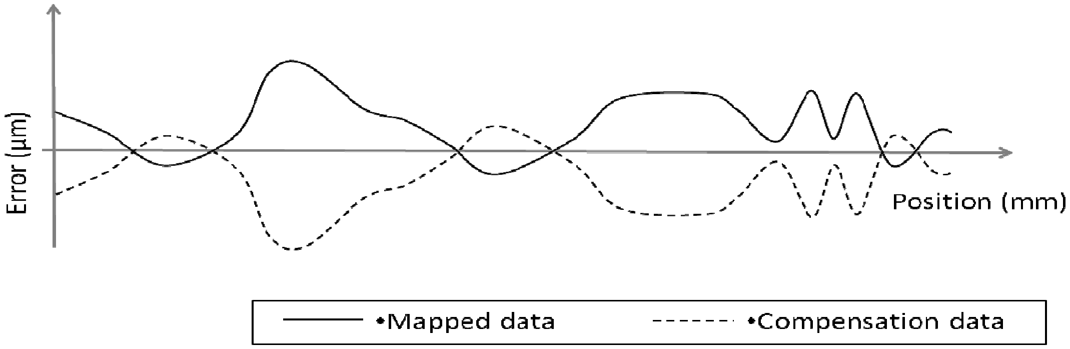


Figure 2-11: Overview of error mapping and compensation process: The solid line represents the mapped error from the laser interferometer of each absolute position along a ballscrew and the broken line represents the compensated value generated by the error compensation software to compensate the error of each position so that the error can be cancelled off

2.4.2 Master Path Tracking

The master path tracking method requires the machine tool point to track the master component such as a ball-bar [10]. This method is mainly targeted to measure the volumetric error of the machine [26]. The tracing error of the workpiece can be measured using both the laser interferometer and the Coordinate Measuring Machine (CMM) and then be compensated using real time error compensation algorithms mentioned earlier to correct the motion's discrepancy of the machine's structures [30]. However, measurement criteria such as: 1) repeatable machining process, 2) lower spatial rate of change, and 3) reasonable cost to compensate the machine error, need to be met in order for the master path tracking method to be used effectively [30].

2.4.3 Geometric Error Compensation

70 percent of the machine errors are composed of geometric and kinematic error [16]. Much research has been performed to mitigate machine errors and improve the accuracy of the machine tool, as well as the CMM that uses the same type of position control system [17, 28, 31-36] .Weekers performed research in software error compensation for dynamic error of the CMM [37]. His approach uses two sets of additional inductive sensors to monitor carriage motion errors. Although the result of the approach is able to improve the machine accuracy, experiments were performed based on an one axis architecture and has not being applied to multiple axes [37]. Mu's approach to CMM compensation, which used a software data fitting method could also only partially compensate for dynamic errors [38]. Donmez first proposed a system to intercept the encoder feedback signals by a computer software compensation [32]. The proposed algorithm used the measured quadrature signals to compute the volumetric error of the machine, and then compensates the position error in real time, which does not required extra modification to the CNC controller software and it can be integrated to any CNC machines [32].

In 2000, Wang and Janeczko designed a new type volumetric positioning errors measurement method that is capable to compute the linear displacement errors and straightness errors concurrently, instead of using laser interferometer [39]. However in 2003, Chapman emphasized that Wang's method still consist of some vector's constraints or sequential diagonal methods [40]. Furthermore, in 2006 Svoboda had shown that the magnitude of the linear displacement error of Wang's method is big after performing multiple experiments and test [41].

2.4.4 Thermal Error Compensation

There are many thermal sources that influence the machines structures: 1) cutting process, 2) cooling system, 3) operating environment, and 4) machine energy loss [16]. Thus, thermally-induced machine error is categorized as the most difficult machine error source for researchers to compensate [42]. Thermal error avoidance research such as repositioning the heat source, and using different types of material to build stiffer structures to minimize expansion were performed but these methods are expensive and still unable to greatly eliminate the thermal expansion error [5].

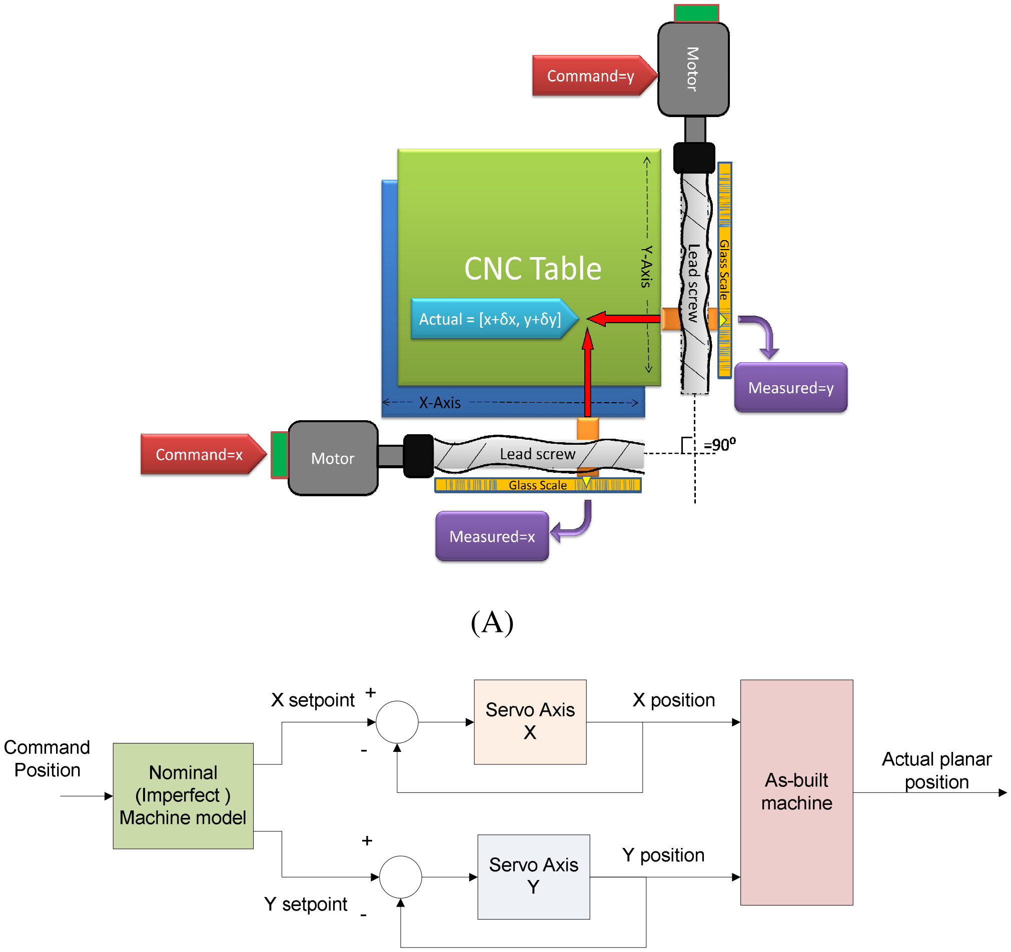
Although thermal error compensation is difficult, much research has been carried out to help to minimize this error. Most of the compensation methods utilize artificial intelligence and a model-based approach to predict the thermally-induced errors [15]. For instance, different types of neural network were used to perform thermal error modeling [43], including Cerebellar Model Articulation Controller (CMAC) neural network [44], fuzzy ARTMAP neural network [45]. Srinivasa and Ziegert approached thermal error compensation using artificial neural networks coupled with a rapid machine measurement scheme [45]. Even with this approach, characterization of the thermal behavior of the machine required several days for data collection. Although the thermally-induced error was improved with this method, accuracy of the overall machine was degraded slightly while at constant temperature [46].

In 1995, Chen and Chiou compared the thermal error modeling effects by using multiple regression analysis and artificial neural network [47]. In 2003, Ramesh utilized the Bayesian network and support vector machine model to perform error mapping during the machining operation based on the machine tool temperature profile [48]. At the same year, Yang and Ni also developed a new type of thermal error mapping model, termed Output error model that can predict the thermal error based on the time series of the operating temperature inputs and the thermal deformation outputs [49]. Although the output error model approach can formulate an accurate thermal error model, it is also heavily depends on the training condition, and parameter tuning of the compensation algorithms used. Yet, the thermal error is still unable to be eliminated [44].

2.4.5 Drawbacks of the Current Positioning System

Based on the up-to-date error compensation algorithms, the actual toolpoint position of the system is still unable to be tracked accurately. In addition, nearly all the methods are still relying on the kinematic model of the machine. For a conventional positioning system, the machine error impact on machines can be further illustrated as shown in Figure 2-12A where the nominal machine kinematic model assumes perfectly straight axes and exact squareness between the two axes. However, the as-built machines guideways for both the X and Y direction are not perfectly flat and straight. Therefore, when one wishes to position the tool at some arbitrary spatial coordinate, the commands to the individual axes are obtained using the nominal kinematic model [4].

Typically, the position setpoints of each axis of the XY table are pre-generated by the Computer Aided Manufacturing (CAM) software before deploying the parts program in the CNC machine's controller. During the operation, the controller will actuate the servo motors based on the generated position setpoints and also to ensure that the position error between the desired setpoints and measured position is minimal. The XY table's planar positions are estimated using the kinematic model based on the measured position of the position sensors of the X and Y axis as shown in Figure 2-12B. Based on the block diagram shown in Figure 2-12B, it can be seen that the individual position feedback of each axis can be accurate with the use of the linear glass scale, but the "As-built" geometrical error such as squareness error cannot be detected via these sensors. Thus, the imperfection in the nominal kinematic model lies outside the feedback loop, and so cannot be detected or corrected in real time [4].



(A)

(B)

Figure 2-12: CNC controller block diagram: A) shows the schematic of a XY table which unable to estimate the actual planar position of the table due to the usage of the kinematic model, B) shows the block diagram of the current CNC controller architecture in which the error feedback lies within feedback loop of each motor.

CHAPTER THREE

SYSTEM DESIGN AND CHALLENGES

This chapter introduces a new type of positioning sensing system that can actively monitor the actual position of a XY table without the dependency of the kinematic model, termed “*Direct Position Sensing*”. First, the concept and system design of the system is introduced. Then, the associated research challenges of the design are given.

3.1 Direct Position Sensing Method

The Direct Position Sensing provides a two-dimensional position sensor to directly monitor planar tool position. To prove the concept of Direct Position Sensing, a prototype to directly monitor the two dimensional planar position was developed using a digital camera and a flat panel display (e.g. Liquid Crystal Display-LCD). The goal of the system is to drive each X- and Y- axis of the system from its current location (Black star in Figure 3-1A) toward the desired location displayed on the LCD. (Black “X” in Figure 3-1A).

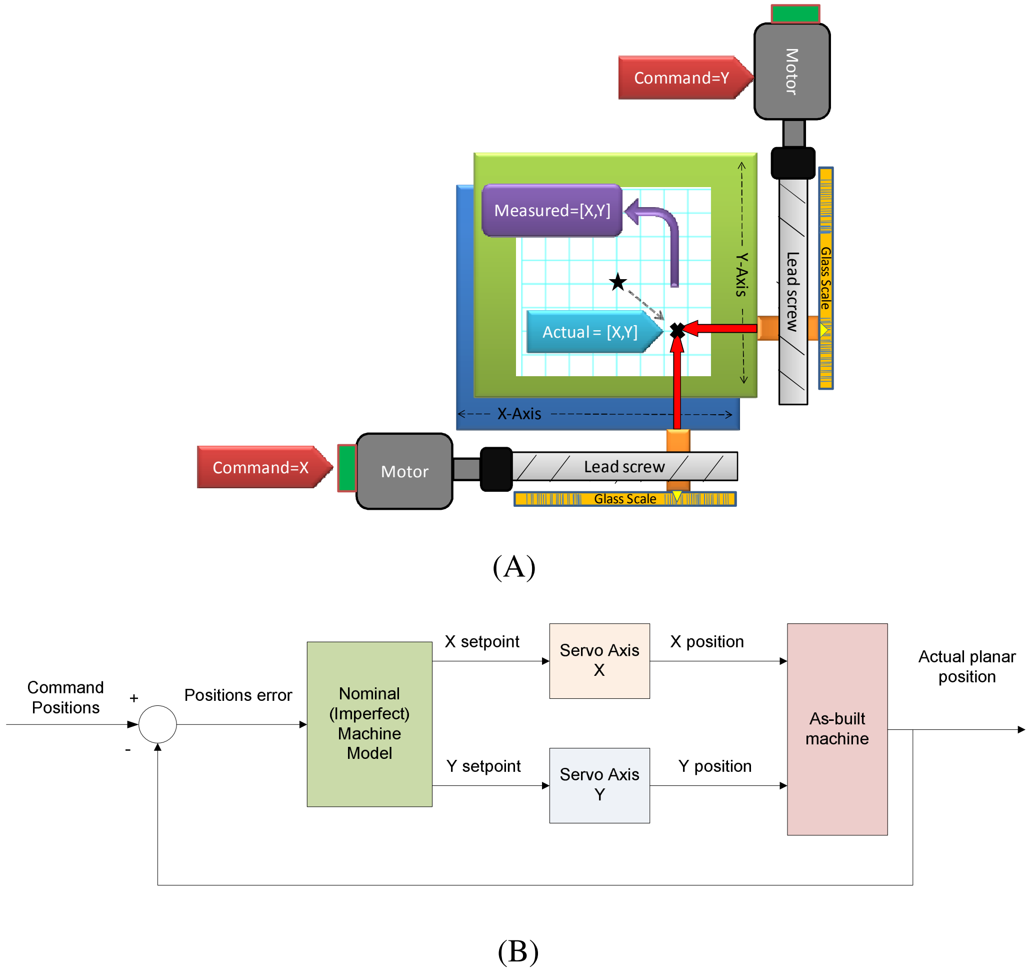


Figure 3-1: Direct Position Sensing system: A) shows the schematic of Direct Position Sensing where the actual toolpoint is sensed via vision sensor instead of conventional position sensor, B) shows the new control scheme where the machine error is located outside the as-built machine, allowing the machine error to be compensated

The error vector of the tool point is measured using the computer vision system and is then decomposed into individual axis position errors to be fed back to the motion controller in order to drive the tool towards the correct position. The motion controller obtains the feedback position from the image processing algorithm that actively monitors the displacement changes of the dynamic display target rather than through the feedback from the kinematic model that is obtained using the position sensors from each axis, as shown in Figure 2-12A. Thus, the travel path of the XY table will not be affected by the geometrical error and thermal expansion on the two axes used for X and Y direction [4].

Unlike the conventional system where the feedback loop is located in the servo drives as seen in Figure 3-1B, Direct Position Sensing's feedback loop is located outside servo loop, permitting the machine error to be included into the control loop. Therefore, Direct Position Sensing will take account of machine error directly to control of the desired path regardless of the machine error that occurs in the machine.

3.2 System Design

In order to show the proof of concept, the Direct Position Sensing prototype was built where the LCD screen that is used to display the active array target is positioned on top of the XY table and a digital camera is mounted below the XY table as shown in Figure 3-2. Figure 3-3 shows the location of the digital camera at the center of the table. This prototype system uses a National Instrument's CompactRIO real time controller and two brushed motor drive modules for motion control purposes. Meanwhile, an IEEE-1394 firewire camera and a National Instrument's Compact Vision System were used for image acquisition and image processing respectively [4].

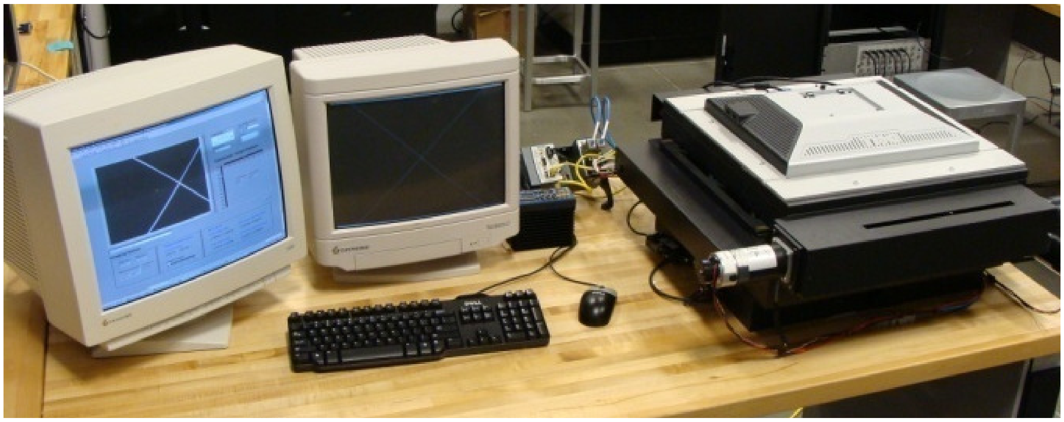


Figure 3-2: Direct Position Sensing's prototype. The prototype configuration of Direct Position Sensing where the LCD screen is located on top of a XY table for target display and tracking purposes

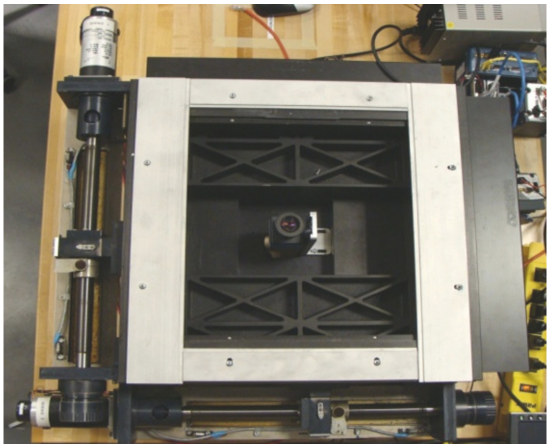


Figure 3-3: Camera location of the prototype. The digital camera is located at the center of the table to track the display target on the LCD screen

3.3 Research Challenges

The Direct Position Sensing consists of two main subsystems: image processing system and motion control system. The main research of this dissertation concentrates on the system integration of the motion controller with the computer vision system. The goal of system integration is to create a robust coupling of subsystems so that they do not have adverse effects on one another. Each system must be designed with the effect on operation of coupled systems in mind so that Direct Positing Sensing via vision sensor can be achieved. There are three subsystems:

1. dynamic path planning which generates the motion controller setpoints for display on the LCD,
2. vision sensor that calculates the position error based on the current position and the displayed target position,
3. motion controller which drives each axis to the displayed location based on the position error obtained by the vision system.

Unlike a conventional motion controller that uses a rotary encoder or a linear glass scale which has high feedback frequency ranging from 10kHz to 100kHz [19], the vision sensor has slower feedback frequency approximately in the range of 2 to 10Hz, while the motion controller's loop rates ranges from 500Hz to 1kHz. The slower feedback in the motion controller's feedback loop leads to an intermittent feedback, in which the motion controller only obtains the actual feedback from the vision sensor in a fixed period of time instead of continuously. In addition, it is assumed that it takes τ milliseconds for the image processing algorithm to output the calculated position at each iteration, the intermittent feedback signal to the motion controller is also time-delayed. The resultant time delay, τ in the intermittent feedback of the Direct Position Sensing as illustrated in Figure 3-4 is detrimental to the path tracking performance of the system. This kind of feedback is unacceptable in the motion control discipline as the traditionally-controlled system response will become sluggish and oscillatory. While the control loop rate can be reduced to match the image processing algorithm's frequency, the response of the proposed positioning system will be unacceptably slow.

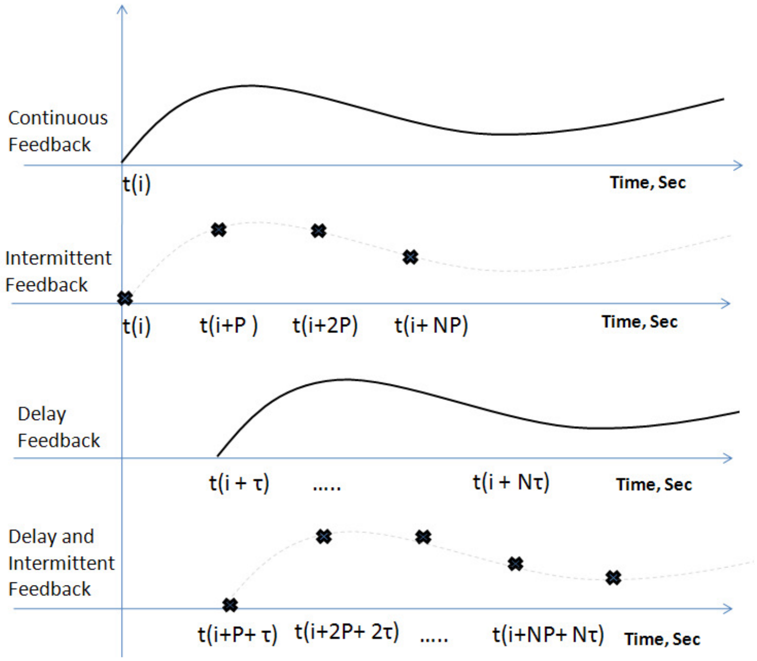


Figure 3-4: The comparison of feedback scenario. From top, continuous signal, intermittent feedback, delayed feedback and delay and intermittent feedback

3.4 Research Objectives and Questions

This dissertation presents the solutions to mitigate the above mentioned challenges. There are two main objectives of this research. The first objective is to establish a fundamental understanding of the effect and impact of the time delay and intermittent feedback to the path tracking performance of Direct Position Sensing. This goal gives rise to the following research questions:

3.4.1 Research Questions A (Time Delay and Intermittent System Behavior)

A 1. What are the detrimental effects of the time delay and intermittent feedback to the proposed system?

A 2. What are the approaches that can be used to mitigate the impact of the time delay and intermittent feedback?

A 3. What are the tradeoffs between the delay and intermittent period as compared with the resolution of the vision sensor?

The second objective is to develop system integration solutions to integrate the motion controller and computer vision system with the available information from the controller and sensor. The following research questions are studied:

3.4.2 Research Questions B (System Integration Challenges)

B 1. What are the available known data that can be obtained by the motion controller during the operation?

B 2. Can a model-based approach be effectively used in the control architecture to predict the path of the actual system? If yes, how accurate can the model be made, and what are the tradeoffs between model accuracy and system controllability?

B 3. What are the modelings approaches to estimate the plant model so that modeling discrepancy can be minimized?

CHAPTER FOUR

MODEL BASED CONTROLLER

This chapter presents the literature and background related to model-based approaches to improve the performance of the system that has time delay and intermittent feedback.

4.1 Model-Based Control

Model-Based Control (MBC) architecture in this context describes a control system that explicitly uses a plant model in the control algorithm [50]. Figure 4-1 shows the generic MBC architecture where the model is used to predict the process output, and the disturbance estimation block is used to adjust and update the estimated disturbance so that the predicted outcome is closer to the actual measurements. The concept of MBC started in the 1970s primarily in the process industry where the process model was use to predict the future system behavior, so that the controller could generate a set of optimal control actions based on the given process constraints [50].

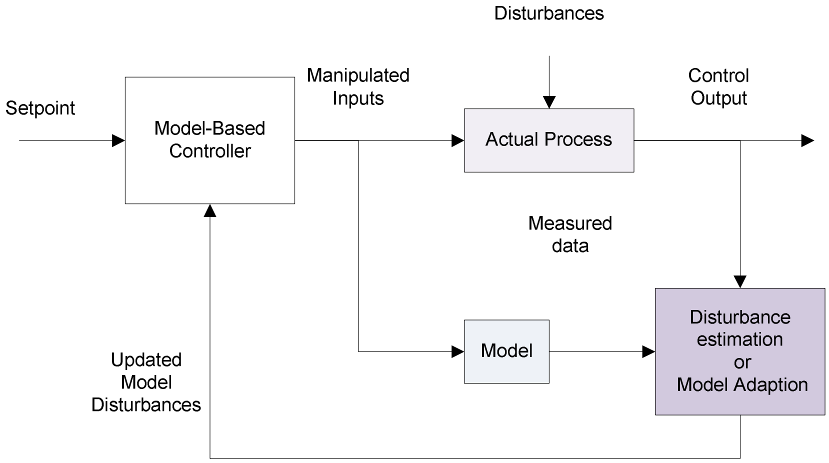


Figure 4-1: Generic Model-Based Control strategy [50]. A model is used to predict the process output, and the disturbance estimation block is used to adjust and update the estimated

The accuracy of the plant model is critical in MBC where the performance and stability of the controller is heavily relying on the model's predictive capability. Typically, the plant model is obtained using a system identification process. The MBC procedure is carried out as follows:

1. formulate the plant model of the system using system identification process,
2. validate the plant model together with the actual plant via open loop stimulus signal,
3. tune the controller using the model in simulation,
4. analyze the performance of the controller based on the design requirement such as percentage of overshoot, rise time, steady state error , and the root mean square of the tracking error,
5. fine tune the controller in the real time system.

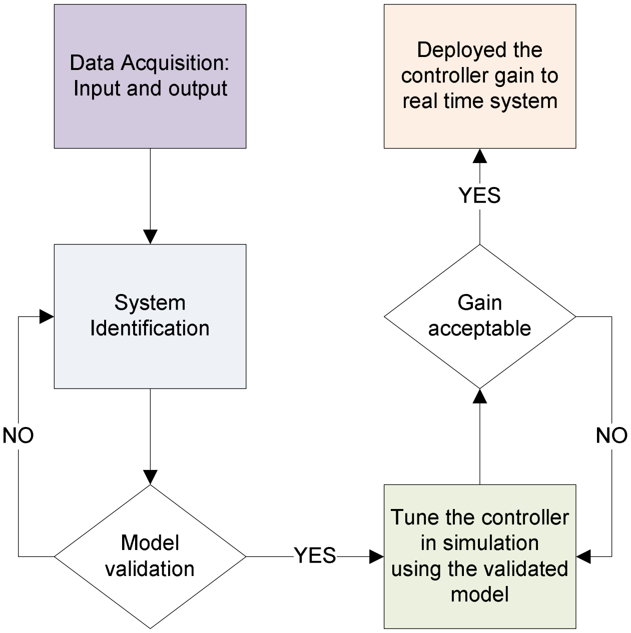

Data Acquisition: Input and output
Deployed the controller gain to real time system
YES
System Identification
Gain acceptable
NO
NO
Model validation
YES
Tune the controller in simulation using the validated model

Figure 4-2: Model-Based Control. Overview of the MBC design from system identification till the deployment of the model to the controller

The selection of an MBC approach in this research is mainly due to the existence of intermittent feedback and time delay in the feedback loop of the system. Intermittent feedback in this context is defined as the period when there are no feedback signals provided by the sensor back to the controller due to the slow sampling time of the sensor. Therefore, the plant model of Direct Position System is used to serve as the path predictor to provide the controller the system output during the intermittent period until the next available actual vision feedback is generated by the image processing algorithm.

4.2 System Modeling

To address the first step for model based controller design, system modeling for the plant used in the proposed system was performed. First the theoretical modeling of the plant model was carried out to estimate the order of the model, and then system identification process was performed to obtain the plant model of the actual system.

4.2.1 Theoretical Modeling

Two servo motors are used in this positioning system, where each motor is coupled to a lead screw of one of the axis of the XY table. Figure 4-3 shows the schematic of the simple servo motor that is modeled in using electric circuit consisting of resistance, *R* representing the resistance within the electric circuits, inductance, *L* representing the inductance within the armature windings and back-electromotive force or back-emf, e_b [7]. Based on Kirchhoff's voltage law, the equation of the electrical circuit can be written as Eq. (4.1).

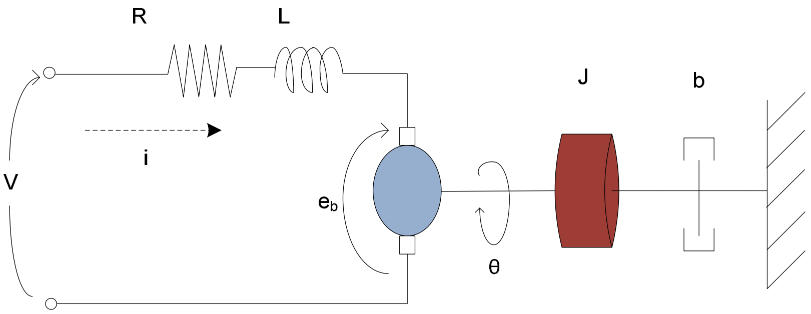


Figure 4-3: DC Motor model [7]. Theoretical modeling for a basic servo motor

$$V = R \cdot i + L\frac{di}{dt} + e_b \tag{4.1}$$

$$e_b = K_{BEMF}\frac{d\theta}{dt} = K_{BEMF}\omega \tag{4.2}$$

The back-emf of the system is proportional to the angular velocity, ω of the motor and can be expressed as Eq. (4.2) and the back-emf constant, K_{BEMF} is used to convert the rotational rate to voltage. The current of the circuit, i is proportional to the applied torque, and the torque constant, K_{motor}. The relationship between the current and the torque is formulated as Eq.(4.3) and the summation of torque of the motor is shown in Eq.(4.4) where b is the viscous damping coefficient representing the mechanical loss and J represents the moment of inertia of the entire rotating system, including the machine rotor, load, coupling and shaft [7].

$$T = K_{motor}i \tag{4.3}$$

$$K_{motor}i = J\frac{d^2\theta}{dt^2} + b\frac{d\theta}{dt} \tag{4.4}$$

$$\frac{\dot{\theta}}{V} = \frac{K_{Motor}}{JLs^2 + (bL + JR)s + (bR + K_{Motor}K_{BEMF})} \tag{4.5}$$

$$\frac{\theta}{V}=\frac{K_{Motor}}{JLs^3+(bL+JR)s^2+(bR+K_{Motor}K_{BEMF})s} \tag{4.6}$$

By solving for the input current *i* of Eq. (4.1) and (4.4) using Laplace Transform with zero initial condition, the transfer function of the angular velocity over the input voltage can be obtained, as shown in Eq.(4.5). In addition, the angular position over input voltage's transfer function shown in Eq. (4.6) is formulated by integrating Eq.(4.5). Based on the theoretical derivation shown, the plant model can be formulated as a third order model if the output is to be the position over voltage relationship. However, the model can be further reduced to a second order model if the inductance of the motor is small and negligible [7].

4.2.2 System Identification

Once the model order of the plant is derived via theoretical modeling, it can serve as a guideline to perform system identification. System identification is a process to construct a model of an actual plant using an estimation algorithm based on the measured data of input and output signal to the plant to be characterized. In brief, Figure 4-4 shows the flow chart of the system identification process where data acquisition of the required data of the servo motor will first be initiated. Then, system identification will estimate the plant model based on the recorded input and output of the plant. After that, validation based on the error between the output response of the predicted model and the measured output is performed and the model is refined until this error falls below a user-defined threshold.

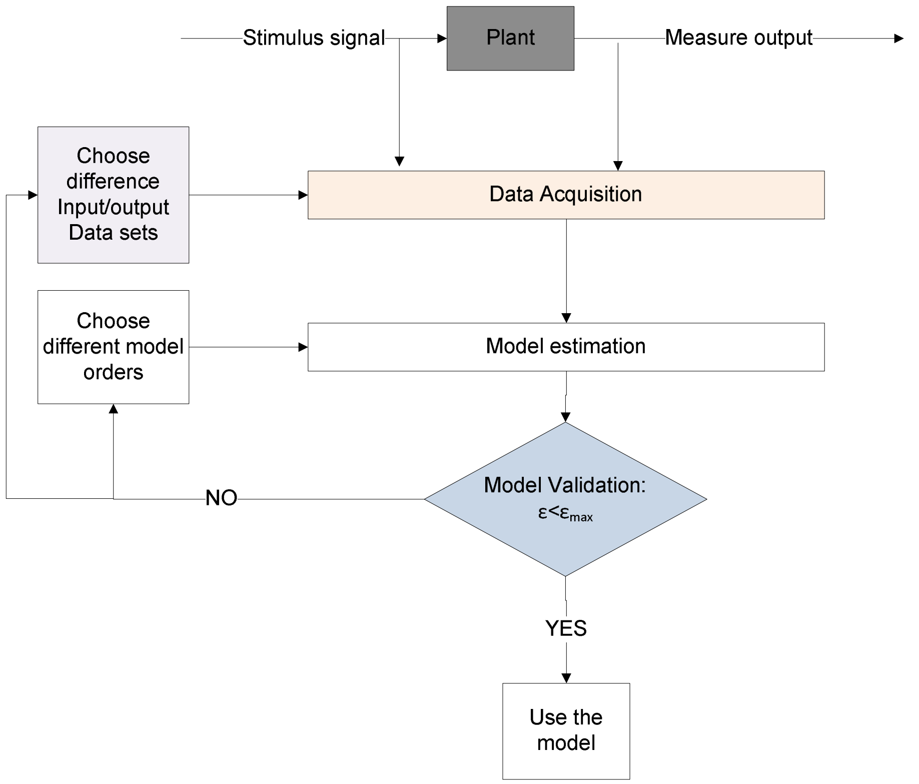

Stimulus signal
Plant
Measure output
Choose difference Input/output Data sets
Data Acquisition
Choose different model orders
Model estimation
NO
Model Validation: $\varepsilon<\varepsilon_{max}$
YES
Use the model

Figure 4-4: System identification procedure. The input and output of the plant are used to estimate the plant model by the model estimation algorithm, and the selection of the model is based on residual analysis of the process and also the max acceptable model error, ε

Traditionally system identification has been an estimation process of a dynamic system based on historical measured data, and can be categorized into parametric and non-parametric methods. Although the non-parametric model estimation method is a more simple method, the estimated model of the system is not as accurate as the parametric model [51]. Most non-parametric estimated models was mainly used prior to the 1960's to estimated the system response based on the impulse response and frequency response of the system in time domain [51-53]. As a result, the non-parametric estimation can only provide partial information to the controller designer such as the stability and time constants of the system but not the model parameters coefficients, which is important in creating model to be used in MBC. On the other hand, the parametric method uses mainly curve-fitting algorithm to predict the model parameters to a pre-selected model, that will be discussed later, based on the measured input and output of a system identification process [51]. The parametric model is formulated in the form of differential equations in both continuous and discrete model that can also be converted to transfer function or state space form depending on the user preference and also system requirement. Generally, the parametric model is represented in a general-linear polynomial form as shown in Eq.(4.7) [51].

$$y(k) = z^{-n}G(z^{-1},\theta)u(k) + H(z^{-1},\theta)e(k) \tag{4.7}$$

where

$$G(z^{-1},\theta) = \frac{B(z,\theta)}{A(z,\theta)F(z,\theta)} \tag{4.8}$$

$$H(z^{-1},\theta) = \frac{C(z,\theta)}{A(z,\theta)D(z,\theta)} \tag{4.9}$$

u(k) and *y(k)* are the input and output of the system respectively, *e(k)* corresponds to the system's disturbance, $G(z^{-1},\theta)$ is the deterministic part of the system, $H(z^{-1},\theta)$ represents the stochastic part of the system, z^{-1} is the backward shift operator in the discrete domain and θ is the set of model parameters [51]. Eq. (4.10), (4.11), (4.12), (4.13) and (4.14) shows the representation of *A(z)* , *B(z)*, *C(z)* , *D(z)* and *F(z)* respectively that are used in the general-linear polynomial form. The deterministic part of the system represents the relationship between the output and input signal whereas the stochastic part represents the unpredictable disturbance that affects the output signal [51].

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_{ka} z^{-ka} \tag{4.10}$$

$$B(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_{kb-1} z^{-(kb-1)} \tag{4.11}$$

$$C(z) = 1 + c_1 z^{-1} + c_2 z^{-2} + \ldots + c_{kc} z^{-kc} \tag{4.12}$$

$$D(z) = 1 + d_1 z^{-1} + d_2 z^{-2} + \ldots + d_{kd} z^{-kd} \tag{4.13}$$

$$F(z) = 1 + f_1 z^{-1} + f_2 z^{-2} + \ldots + f_{kf} z^{-kf} \tag{4.14}$$

ka, *kb*, *kc*, *kd* and *kf* are the model orders. Eq. (4.15) shows the general representation of a general-linear polynomial model; the block diagram of the model is shown in Figure 4-5.

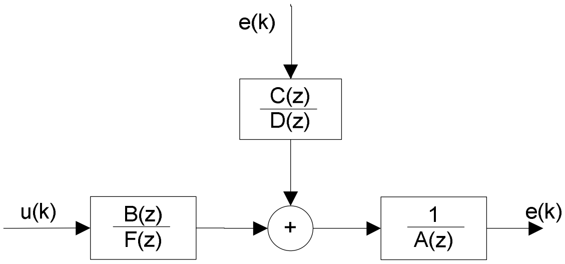


Figure 4-5: General-linear polynomial model's block diagram [51] Allow user to select the types of model to be used in the system identification process

$$A(z)y(k) = \frac{z^{-n}B(z)}{F(z)}u(k) + \frac{C(z)}{D(z)}e(k) \qquad (4.15)$$

The used of general-linear polynomial model, allows controller designer to have the options to set one or more of *A(z)*, *C(z),D(z)* and *F(z)* to 1 for both system dynamics and stochastic dynamic. There are four commonly-used configurations used in real world applications: autoregressive with exogenous terms (ARX), autoregressive-moving average with exogenous terms (ARMAX), output error and box-jenkins. A summary of these four types of general linear polynomial model methods is shown in Table 4-1.

Table 4-1: Types of parametric models algorithms [51]

Model type	Criteria that set to 1	Summary
ARX	*C(z), D(z)* and *F(z)*	The model includes the stochastic dynamics and is useful when disturbances enter the process early. The stochastic and dynamic system share the same poles.
ARMAX	*D(z)* and *F(z)*	The model includes the stochastic dynamics and is useful when disturbances enter the process early and have more flexibility than ARX in handling models that contain disturbances.
Output Error	*A(z), C(z),* and *D(z)*	The model describes the system dynamics separately from the stochastic dynamics.
Box Jenkins	*A(z)*	This model represents the disturbances properties separately from the system dynamics. It is useful when disturbance enters late in the process.

For instance in the case of ARX, the system identification algorithm will need to fit the parameters to the data criterion so that the prediction error of the Least-square method, $V_n(\theta)$ shown in Eq. (4.16) can be minimized. N is equal to the amount of measured data, $y(t)$ is the measured data and $\widehat{y}(t)$ represents the predicted output of the model, that is a scalar of the known data vector, φ and parameter vector, θ [51]. The Least-square solution is formulated as in Eq.(4.18) [50].

$$V_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \widehat{y}(t)) \tag{4.16}$$

$$\widehat{y}(t) = \varphi^T(t)\theta(t) \tag{4.17}$$

$$\theta = (\phi^T \phi)^{-1} \phi^T y \tag{4.18}$$

According to the ARX model criteria, the gain of *C(z)*, *D(z)* and *F(z)* are configured to 1, generating a general-linear polynomial model in the form shown in Eq.(4.19) and the parameter vector and data vectors of the model are shown in Eq. (4.20) and (4.21) respectively.

$$A(z)y(k) = B(z)u(k) + e(k) \tag{4.19}$$

$$\varphi(t) = [-y(t-1) ... - y(t-n_a) \quad u(t-1) ... u(t-n_a)] \tag{4.20}$$

$$\theta_{ARX} = [a_1 \quad a_2 .... a_{na} \quad b_1 .... b_{nb}]^T \tag{4.21}$$

Depending on the types of model that the controller designer chooses to formulate the model, the estimated model parameters will not be identical and can varies immensely depending on the selected model types. In order to select a better model to be used in the system, model validation needs to be carried out to quantify the goodness of the model.

4.2.3 Model Selection and Validation

The actual system can be represented by different models. As mentioned in the previous section, there are many system identification estimation methods used in formulating the model and each type of model representation has its own pro and cons. Although these estimation methods can help to formulate the best model of the system, the model might not be able to represent the actual system accurately [51]. Even with a higher order model, a mathematical model will not have a perfect representation of an actual plant of a system but if the selected model can have close estimation of the actual plant, then the model is considered “good enough” to be used [51]. In addition, most process change with time, which restrict the obtained plant model to be able to accurately represent the process during the operation. Instead of trying to perform multiple system identification processes to obtain the most accurate model for the actual plant, it is suggested to design a control system or disturbance estimation algorithm to compensate the model uncertainties effectively [51].

Model validation needs to be performed to select a heuristic model that best fit the actual system; this should be performed in both software and hardware experiments [8]. After the model is built, model simulation can be performed by using the same stimulus input signal so that the residual of the model can be obtained. Residuals in this context is described as the difference between the measured output, *y(k)* and the predicted output of the model, *y’(k)* as shown in Eq. (4.22) [51].

$$e(k) = y(k) - y'(k) \tag{4.22}$$

Residual analysis can be performed in the hardware validation. Once the model is validated in simulation, it will be programmed in the control algorithm of the prototype so that the model is running parallel with the actual plant. Therefore, same input signal will be fed to both the actual plant and model, and the residual analysis can be carried out based on the captured outputs from both the plant and model. Once the model is finalized, then it is ready to be used in MBC.

4.3 Smith Predictor

One of the MBC algorithms used in this research is the Smith Predictor. In 1957, O. J. M Smith presented a control algorithm using a plant model running parallel with the actual plant shown in Figure 4-6 to improve the system that has a long time delay, with the base design becoming known as the Smith Predictor. This structure is shown in Figure 4-6, and has a mathematical plant model, $G_e(s)$ that is created parallel to the actual plant $G_{e0}(s)$ in the system. The objective of the mathematical plant model is to serve as a predictor for the actual model when time delays occur [54]. The prediction error, d_y is used to correct the system response when the actual feedback is obtained after the time delay. Table 4-2 shows the nomenclature of the block diagram.

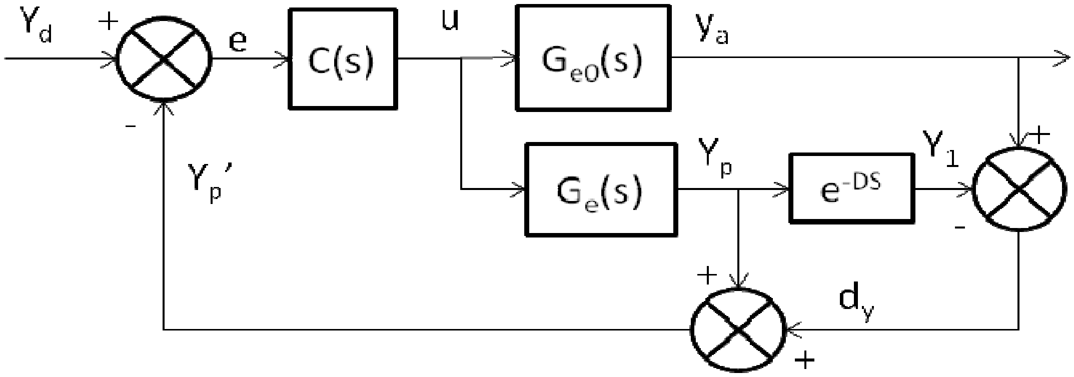


Figure 4-6: Smith Predictor block diagram: The inner loop consists of the system model to predict the actual system's output. The model is delayed by a time step *D* representing the actual time delay of the system. The residual is added to the model prediction to compensate the predicted value for feedback.

Table 4-2: Nomenclature for Smith Predictor block diagram

C(s)	Controller
$G_{eo}(s)$	Actual Plant
$G_{eo}(s)$	Mathematical Model
e^{-Ds}	Time Delay Block
Y_d	Desired Setpoints
e	Setpoints Error
u	Control Input
Y_a	Plant Response
Y_p	Mathematical Model Response
Y_1	Delayed Mathematical Model Response
d_y	Error Between Actual And Mathematical Model
Y_p'	Response Feedback

Initial implementations were mainly undertaken in the process control sector (*e.g.*, chemical plants, petrochemical refinery). However, the Smith Predictor structure has been widely used in many applications such as the communication, and motion control sectors. Although the theoretical analyses of the Smith Predictor form shows that it can effectively mitigate time delay in closed loop control, there are still some imperfections to be taken into consideration especially in dealing with disturbance rejection. Thus, many modified versions of Smith Predictor have been created to meet the requirement of the related applications.

4.3.1 Literature of Modified Smith Predictor

The original Smith Predictor architecture will have a steady state error if the time delay of the process is unknown or varying. Consequently, Watanabe and Ito had proposed a Modified Smith Predictor to reduce the steady state error by incorporating an integral function in the plant model to stabilize the plant [55]. However, analysis and experiments done by Åström had show that the Watanabe's design has slow setpoints response and the disturbance rejection response is tends to be oscillating. [56]. Åström on the other hand had proposed an approach to enhance Watanabe's design. Åström's approach is to decouple the disturbance or load response from the setpoints response so that both of these responses can be solved individually. Although the setpoints tracking of the new design is faster, the tuning of the controller is more complicated [56, 57]. Then, Matausek and Micic performed research to further improve the Åström's Modified Smith Predictor by having an additional controller in the Smith Predictor Inner loop to reduce the disturbance of decoupled load response of the system. [57]. Moreover, the Matausek and Micic's Modified Smith predictor was proven to have better system response and path tracking performance as compare to the Åström's Modified Smith Predictor architecture based on the simulation results done by Tian and Gao [58].

4.4 Adaptation of Smith Predictor in the Visual Servo-ing Applications

One of the applications that Smith Predictor is often being implemented is in the visual-servoing application. Visual servoing is also known as vision-based robot control where the feedback from a vision sensor is used to perform closed loop control of a system such as robots, and unmanned vehicle.

4.4.1 Overview of Visual Servo-ing

Most factories are equipped with multiple kinds of robots replacing manual labor, with the intention to produce high repeatability and accurate products. However, the control algorithm of a robot is complicated and has limited position accuracy due to the used of the kinematic model of the framework that resist the robot to be controlled accurately. At the same time, a lot of companies also realized that in order to improve the robot accuracy, more sensors need to be used but the sensors integrations can increase the cost and the complexity of the control architecture [59]. This might cause the control system to be less robust and stable. Instead of adding more sensors to the robot, computer vision is proposed as an alternative solution to improve the position accuracy of the robot.

In the robotics world, computer vision is used to mimic the human eye in most applications. Shirai and Inoue have performed earlier research on using a computer vision system to not only control the robot end-effectors but to improve the robot accuracy [60]. It can also been seen in today's technology where commercial robot companies such as

Kuka , Fanuc, Stäubli, and ABB have integrated computer vision in most of their robot models, either in an open loop function or a "Look-and-move" system. Thus, the accuracy of the robot is heavily depends on the visual sensor and also on the joint position sensor of the robot. Corke performed a literature search on visual servoing and found that there has been a tremendous improvement of the integration of visual servoing robotics in multiple industries [61]. In addition, Corke also describes that there are two main visual servo-ing architectures: dynamic look-and-move and direct visual-servo as shown in Figure 4-7 and Figure 4-8 respectively used in the industry [61, 62].

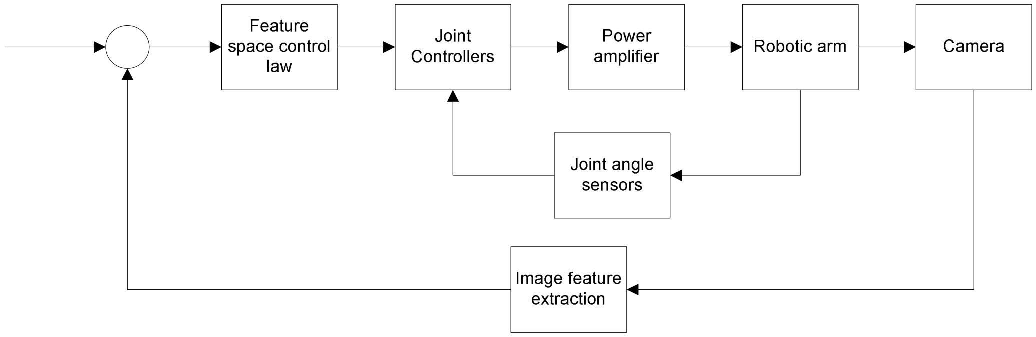


Figure 4-7: Dynamic look and move [63]: Having the computer vision as position enhancer but still relying on the joint angle sensors to provide the position of the system to the controller.

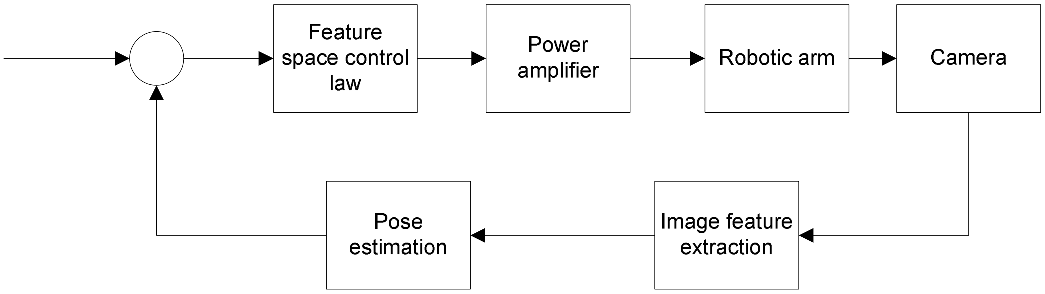


Figure 4-8: Direct visual-servo [63]. Uses the computer vision as the main feedback of the control system.

As seen in the block diagram, the dynamic look-and-move uses both the joint angle sensor and the computer vision as the feedback for the system. On the other hand, the direct visual-servo eliminates the use of the joint angle sensor and purely relies on the vision sensor to provide the feedback to the controller. Hutchingson, and Corke also point out that nearly all the visual servoing application adapt to the Dynamic look-and-move architecture [59]. This is because the slow feedback sampling rate of the vision sensor will cause the controller to be unstable, and also that the controller gain will have to be de-tune to much lower gain in order to compensate the time delay that occurs in the system. Therefore, with the assistance of the joint angle sensor, the controller can still maintain the stability of the system while using the vision sensor to further improve the robot's position accuracy [61].

The "Look-and-move" algorithm has been widely adopted by the manufacturing industry. Corke and Hutchingson also mentioned that kinematic singularities of the Look-and-move algorithm is separated from the visual controller, which permits the visual control robot becomes an ideal Cartesian motion device to accurately position the end effectors of the robot [59]. However, the low sampling rate of the vision sensor in the direct visual servo algorithm created a complex and nonlinear control problem for the controller designer as the feedback of the system is too slow, resisting a robot to have a good system response. Therefore, Corke and Hutchingson emphasized that the internal joint position sensor cannot be eliminated from the controller in order to have fast system response while maintaining the stability of the robot [61].

4.4.2 Smith Predictor in Visual-servo Application

For the system that does not require a fast system response, the dynamic look-and-move architecture is able to maintain stability and provide feedback to the controller to move accordingly. However, for a system that requires faster system response such as path tracking visual servoing application, there are still some system integration challenges specifically related to the image processing time of the vision sensor which is also the feedback delay within the system. Feedback delay can cause the system to have position error and also potentially create the wind-up situation causing the system to be unstable. In order to address such research challenges, the Smith predictor has been applied to some of the dynamic look-and-move visual servoing application.

Sim, Hong and Lim modified the Smith Predictor architecture for a 3D visual servoing application in an AdeptOne robotic arm and they added a path predictor, *F* in the original Smith Predictor architecture as presented in Figure 4-9, in which this predictor was programmed to predict the future path of the system one unit time step ahead. Based on their results, they concluded that their modified version improved the path tracking performance of the robot significantly and Figure 4-10 shows the RMS of the tracking deviation of a linear motion [64].

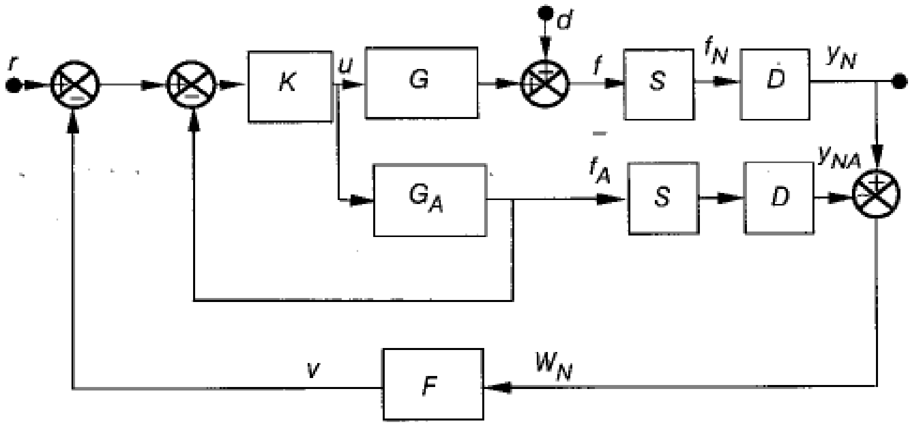


Figure 4-9: Multi rate predictor control scheme for Visual Servo[64]. System block diagram of Sim, Hong and Lim Modified Smith Predictor

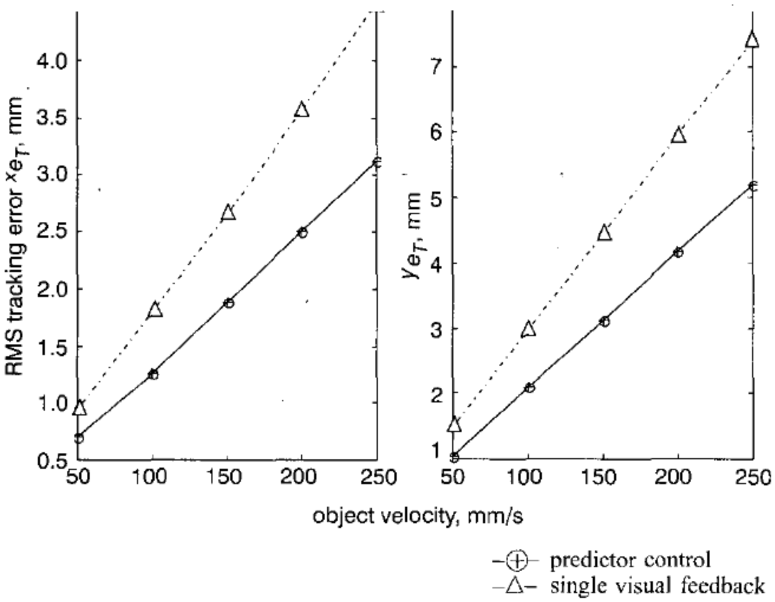


Figure 4-10: Results of the multi rate predictor control scheme for visual servo[64]. Results of Sim, Hong and Lim Modified Smith Predictor

Xie, Sun, Rong and Yuan applied a Modified Smith Predictor in a micromanipulation robot to perform point-to-point motion of their robot where they modeled the vision sensor of the system with dual modeling loops as shown in Figure

4-11, instead of model the actual plant of the micromanipulator. According to their findings, their modified architecture is able to reduce the overshooting of the system that is caused by the time delay and also the micromanipulator has better tracking performance and disturbance rejection[65].

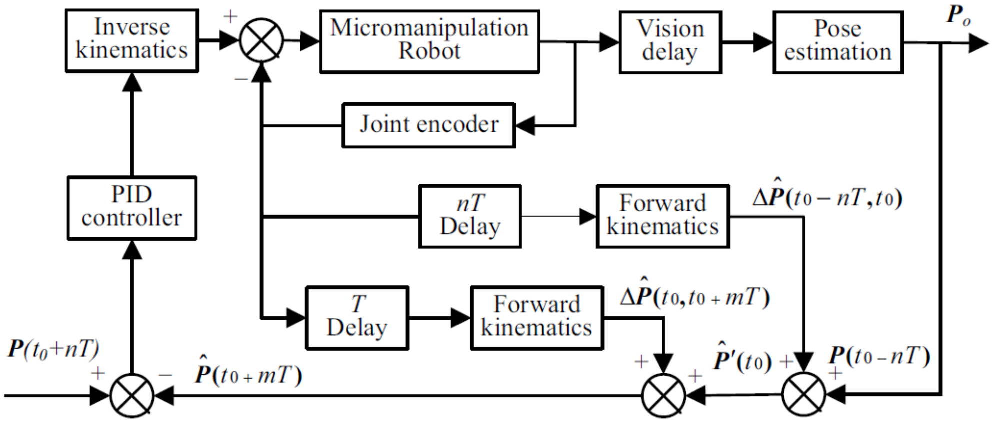


Figure 4-11: Visual servoing with Modified Smith Predictor [65]. System block diagram of the micromanipulator

Zeng, Huang and Wang implemented a Fuzzy adaptive PID with Modified Smith Predictor to a micromanipulation robotic hand and the control block diagram of their proposal is shown in Figure 4-12. They were using a Modified Smith Predictor that consists of an integrator, *M* to minimize the steady state error of the system. Then, they used a Fuzzy logic algorithm to tune the PID gains of the manipulator controller. [66]. They concluded that their Modified Smith Predictor as compare to the single PID controller has more robustness and better disturbance rejection of their micro manipulator that was running at the feedrate of 1.2 mm/s.

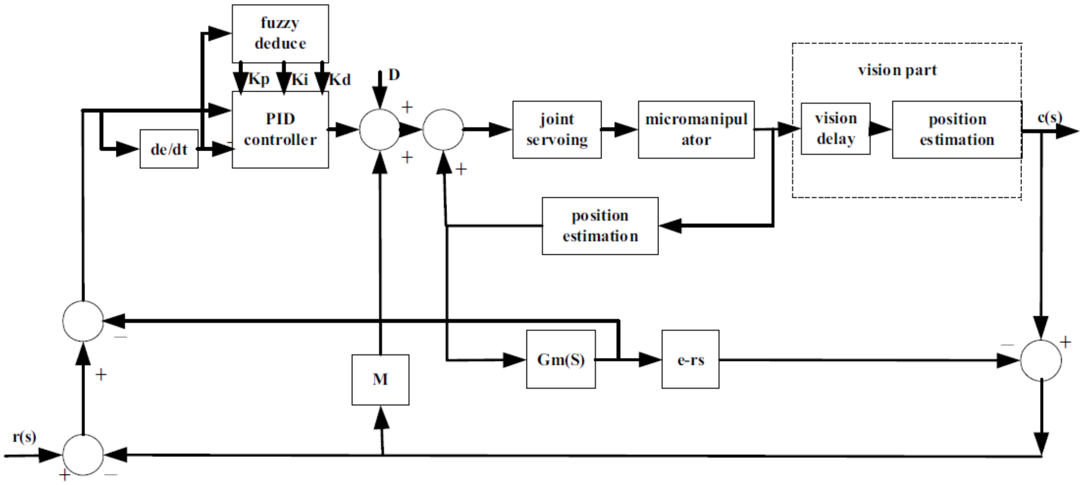


Figure 4-12: Fuzzy adaptive PID with Modified Smith Predictor [66]. System block diagram of micromanipulation robotic hand

4.5 Summary Remarks

Although Smith Predictors have been successfully implemented in many visual servoing applications, most of these applications are still rely on the primary position sensor to stabilize the controller, and the goal of these Modified Smith Predictor(s) is to mitigate the time delay of the system only. However, the Direct Position System has one more challenge than some of the previous applications that were mentioned in the literature: the intermittent feedback. This is because the Direct Position System is not relying to the conventional position sensors to provide the feedback but just only the vision sensor. Thus, the Modified Smith Predictor that will be discussed in chapter six will shows the uniqueness of the Direct Positing System version of Smith Predictor in improving the path tracking of the system by creating a prediction algorithm for the during the intermittent period to minimized the model residual and also update the control action of the model so that the model output is always close to the plant output to have a better prediction of the system.

CHAPTER FIVE

ADAPTATION OF SMITH PREDICTOR TO DIRECT POSITION SENSING

This chapter presents the procedures and results of the system identification carried out to obtain a heuristic plant model for the model-based controller used in both simulation and hardware experiments.

The simulation and hardware validation of the Smith Predictor approaches is presented for the cases when the feedback is 1) continuous, 2) delayed, 3) intermittent, and 4) combined delayed and intermittent. The objective of the simulation and hardware validation is to observe the tracking performance of the Smith Predictor controller of each feedback type and quantify the error. Root Mean Square (RMS) error between the setpoints of the system and the output of the plant was used as the metric to quantify the performance. Error of the adapted system for each case is compared with the baseline continuous feedback case.

5.1 System Identification

System identification for the servo motor used in the prototype was performed. First, data acquisition to record the input and output of the servo motor was performed. A sine sweep signal as shown in Figure 5-1 with amplitude of 20V peak-to-peak was used as the stimulus signal for the servo. At the same time, the motor linear position, in mm based on sine sweep stimulus signal were measured.

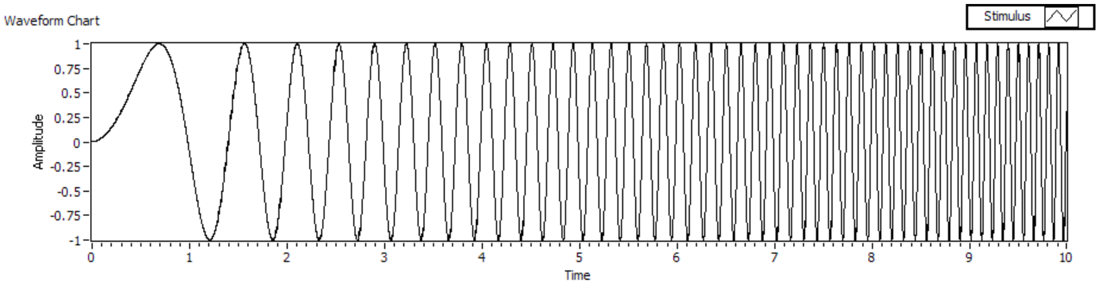


Figure 5-1: Sine sweep signal used as the stimulus signal for system identification: This is an example of the sine sweep signal continuously from 0.01 to 10 Hz, allowing stimulus-response analysis across a range of relevant frequencies.

Both the recorded stimulus signal and servo position were fed into the system identification algorithm. The ARX parametric model estimation algorithm was used to estimate the model of the servo because it is one more the most commonly used algorithm and also that it provide better prediction based on the residual analyses. Eq. (5.1) shows the discrete transfer functions estimated by the ARX model.

$$G(z) = z^{-2} \frac{0.000190997z^2 + 0.00038199z + 0.000190997}{z^2 - 1.88958z + 0.889583} \tag{5.1}$$

5.2 Model Validation

First, model validation was performed by checking RMS residual between the measured signal and the model output signal. The model was also proven stable as all the poles of the system stay within the stability circle. Once the mathematical model of the plant was selected, the same stimulus signal that was used to actuate the actual plant was fed into the selected model to generate a set of model output so that the waveform of the model and the actual measured data can be compared.

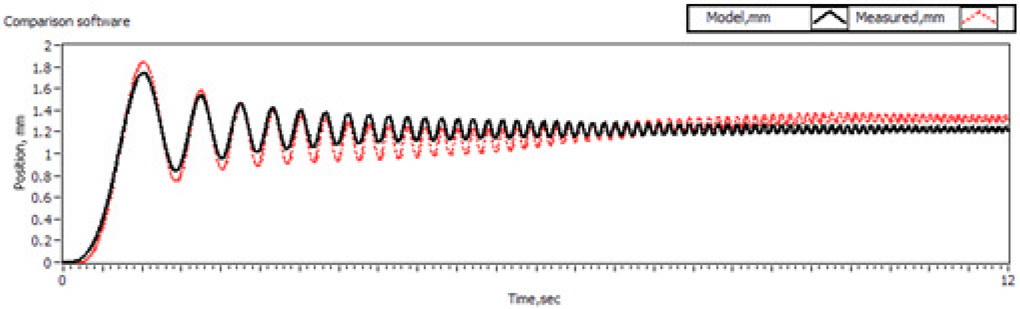


Figure 5-2: Model validation in software. The black solid line is output of the model and red dotted is the output of the motor

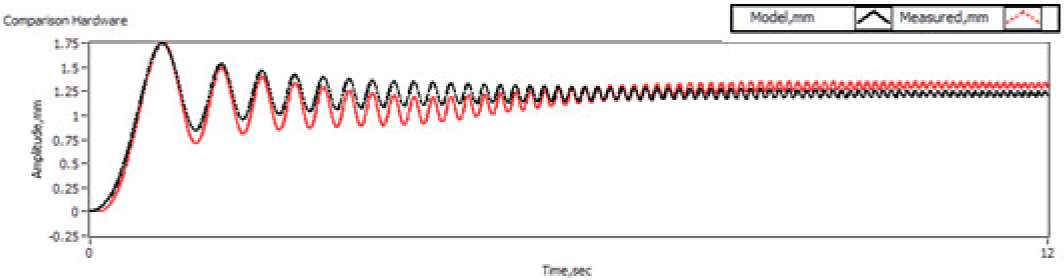


Figure 5-3: Model validation in hardware. The black solid line is output of the model and red dotted is the output of the motor

For the software validation, the model was proven stable based on the root locus plot, and the RMS residual, ε was 0.09mm, considered small for an open loop system. After that, the model was also validated in the prototype in which the model was programmed to be running parallel with the actual servo motor. With the same stimulus signal, the position of both the servo and model were captured as shown in Figure 5-3. The RMS residual, ε between the model and the actual servo motor position was 0.11mm which is also considered small in an open loop manner. Based on the validation of the stability and the RMS residual, the model is incorporated to the system architecture.

5.3 Smith Predictor

Before the Smith Predictor was deployed in the real time controller, simulations were carried out to observe and analyze the behavior of the system when the feedback is 1) continuous, 2) delay, 3) intermittent and 4) combined delay and intermittent. Figure 5-4 shows the test plan for both the simulation and experimental tests that were performed to analyze the characteristic of the system.

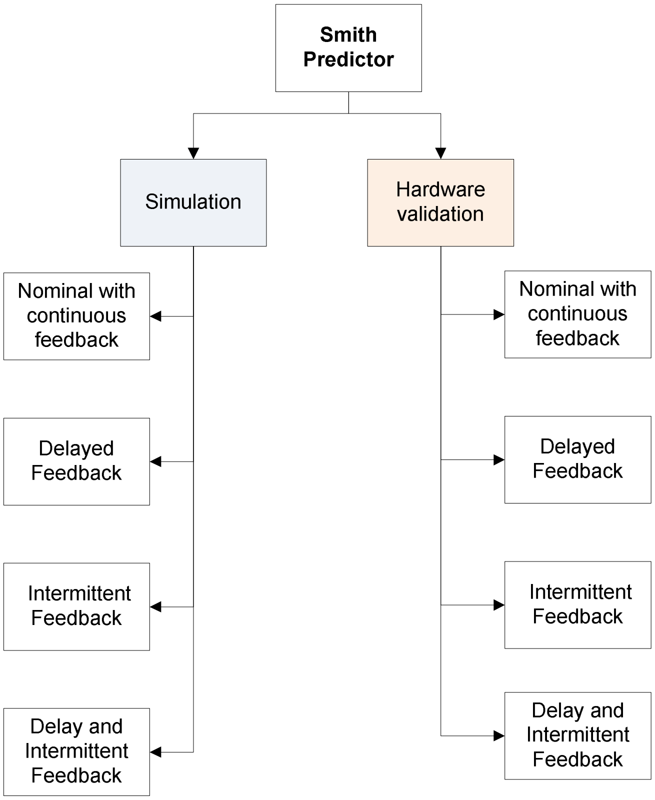

Smith Predictor
Simulation
Hardware validation
Nominal with continuous feedback
Delayed Feedback
Intermittent Feedback
Delay and Intermittent Feedback
Nominal with continuous feedback
Delayed Feedback
Intermittent Feedback
Delay and Intermittent Feedback

Figure 5-4: Smith Predictor's test plan. For simulation and experimental validation

5.3.1 Simulation

Simulation of the Smith Predictor to mitigate the mentioned three different feedback scenarios was performed using LabVIEW Control and Simulation toolkit. The goal of these simulations is to study the detrimental effect of the delay intermittent feedback to the motion control of Direct Position Sensing and also to analyze the performance and limit of the Smith Predictor in path tracking.

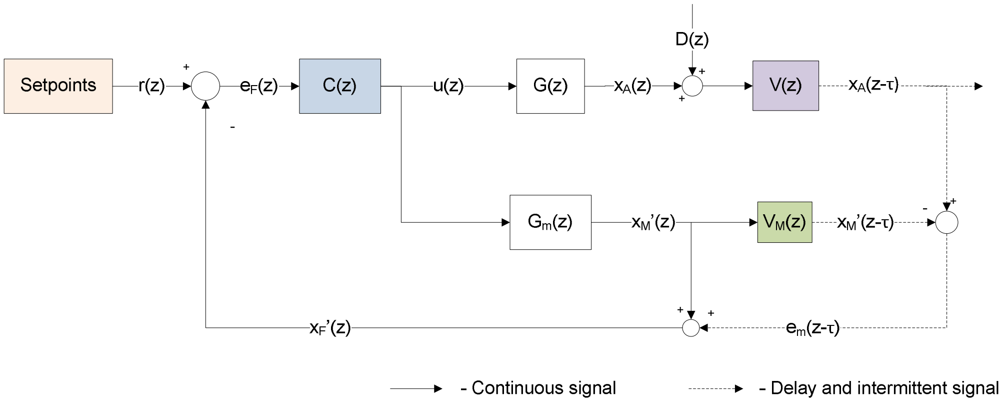


Figure 5-5: Smith Predictor architecture. Two vision sensor emulators were added to the controller to simulate the feedback of the vision sensor

Figure 5-5 shows the block diagram of the Smith Predictor during simulation where *C(s)* is the controller, *G(z)* is the plant, and *$G_m(z)$* is the plant model. In order to perform analysis of the effect that the time delay and intermittent feedback have on the path tracking performance of Direct Position Sensing, vision sensor emulators, *V(z)* and *$V_M(z)$* were created for both the plant and the model so that the time delay (0-500ms) and intermittent cycles (0-500 cycles) can be manipulated across the stated test ranges in order to observe the behavior of the control system in responding to different time delay and intermittent configurations.

Due to the computation power of the micro processor of the prototype, the actual controller of the prototype is operating at the rate of 500Hz instead of 1kHz. Thus, during the simulation, it was assumed that the micro controller was operating at 500Hz (closing the loop at 2ms per cycle), and the intermittent cycle was varied from 0 to 250 intermittent cycles to synchronize with the microcontroller loop time.

Since the mathematical model is not 100% accurate, the impact of model residual for Smith Predictor was also tested in simulation. The model residual between the plant model, *G_m(z)* and the actual plant, *G(z)* as illustrated in Figure 5-5, was simulated by using two different plant models that were obtained via system identification as shown in Eq. (5.2) and Eq.(5.3), respectively.

$$G(z) = z^{-2}\frac{0.000190997z^2 + 0.00038199z + 0.000190997}{z^2 - 1.88958z + 0.889583} \tag{5.2}$$

$$G_m(z) = z^{-2}\frac{0.000175z^2 + 0.000351z + 0.000175}{z^2 - 1.90535\mathrm{z} + 0.905346} \tag{5.3}$$

5.4 Results

A 0.2Hz sine wave with amplitude of 2mm peak-to-peak of was used as a reference trajectory of the test. In order to quantify the path tracking performance of the Smith Predictor, the RMS position error between the reference trajectory and the plant output was used as the performance metric of the system. Note that the goal of this simulation and hardware experiments in this chapter was not focused on obtaining the best gain to achieve the best tracking performance of the system but to analyze the effects of each test case: 1) continuous, 2) delay, 3) intermittent and 4) delay and intermittent feedback to the Smith Predictor. Thus, the Proportional-Integral (PI) controller used in Smith Predictor architecture in simulation and hardware experiments had the same *PI* gain which K_c=10 and T_i=0.05 that was tuned by trial and error, so that the behavior of each case can be compared.

5.4.1 Continuous Feedback

First, the nominal case in which the system has continuous feedback was run so that response and the RMS position value of this case can be used as a reference for the performance comparison of the system. In the nominal case, the RMS position error for the simulation was 6.47µm and the hardware RMS error was 17.03µm.

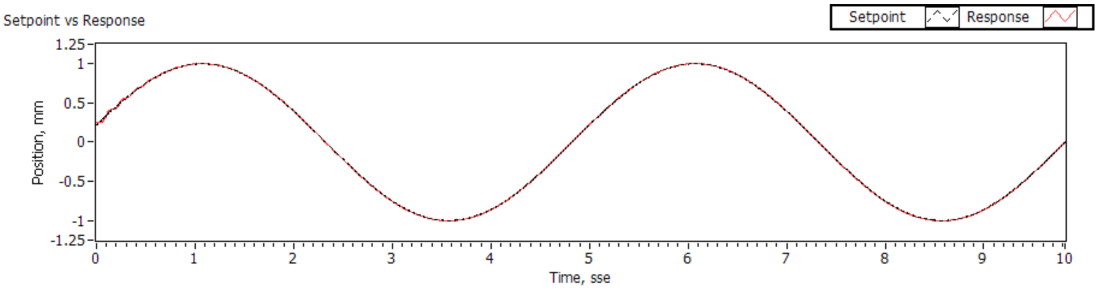
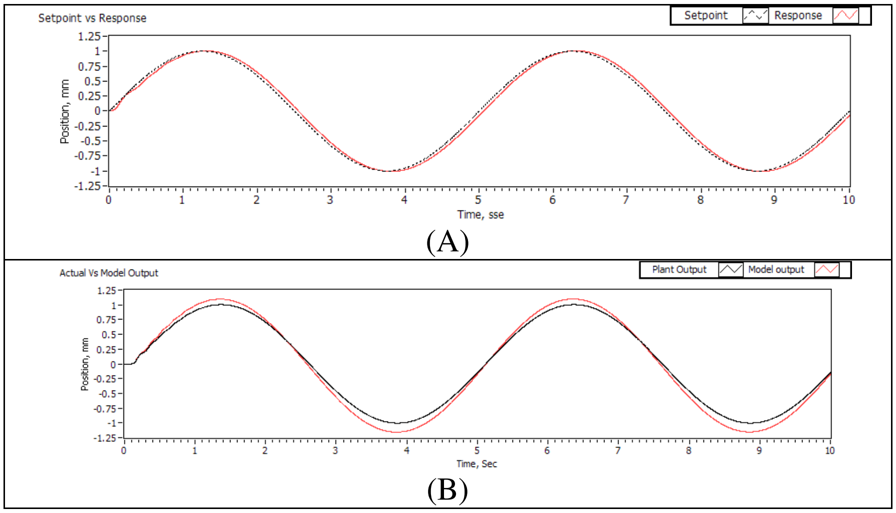


Figure 5-6: Simulation result when the feedback is continuous. This simulation was run to obtained a nominal value for the tracking performance

5.4.2 Delay Feedback

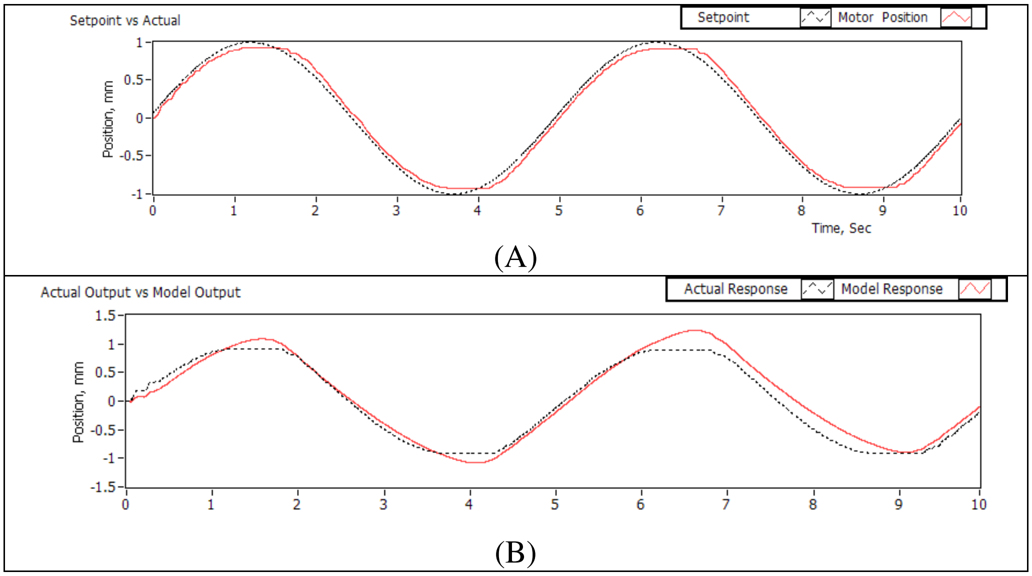
Simulation and hardware testing of a system that has time delay were performed. Figure 5-7 and Figure 5-8 shows the simulation and hardware result when there is 100ms delay, respectively.



(A)

(B)

Figure 5-7: Simulation results when system has 100ms delay feedback: (A) Plant output vs. reference, illustrating following error, (B) Plant output vs. model, illustrating model residual



(A)

(B)

Figure 5-8: Hardware experiment results when system has 100ms delay feedback: (A) Plant output vs. reference, illustrating following error, (B) Plant output vs. model, illustrating model residual.

As seen in Figure 5-7(B), the model residual of the system is quite linear and it is not drifting from the plant output, but this is not the case in the prototype as shown in Figure 5-8(B), in which the model drifted toward one direction. This is detrimental to the system when the model output is not corrected. An algorithm to address this problem is presented in chapter six.

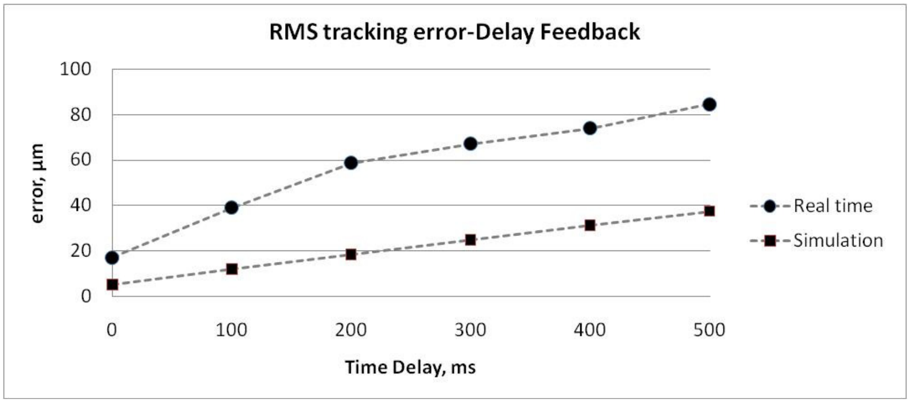


Figure 5-9: RMS position error of Smith Predictor with different time delay scenario for both simulation and experimental testing

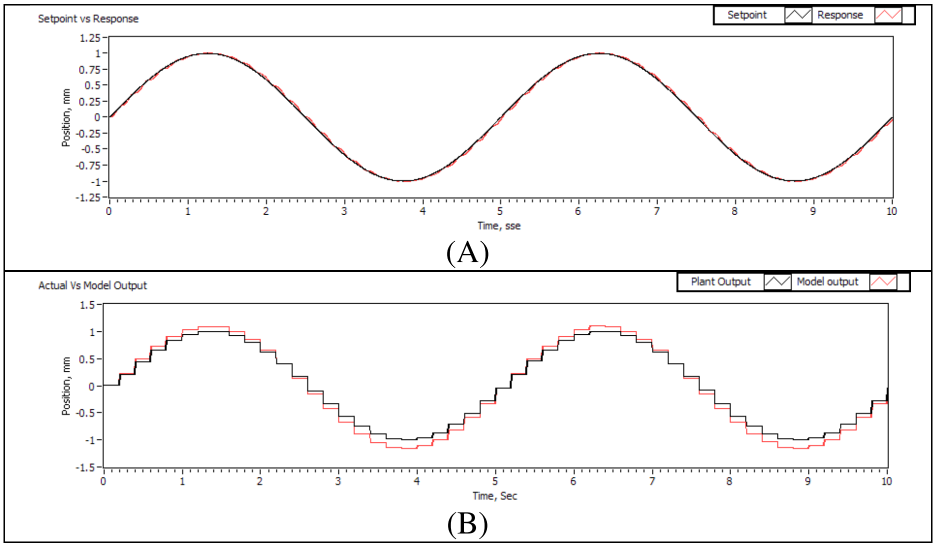
Table 5-1: RMS position error of Smith Predictor with different time delay scenario for both simulation and experimental testing

	RMS Position Error,(µm)	
Time delay,(ms)	**Real time**	**Simulation**
0	17.03	5.27
100	38.92	11.89
200	58.80	18.47
300	67.16	24.96
400	73.91	31.29
500	84.68	37.44

Figure 5-9 and Table 5-1 summarize the RMS position error of both the simulation and experimental results run from 100ms to 500ms. Based on the tests, it can be observed that the tracking position error increases with respect to the increment of the time delay. When comparing with the nominal tracking error, both the simulation results and hardware experiments results show the increment of tracking error. In the case of 100ms, the tracking error of the hardware experiment was 38.92µm but when the delay increased to 500ms, the tracking error of the system increases to 84.68µm. One of the factors that affect this error is the accuracy of the model; the other is due to the time delay feedback that occurs in the system in which the modeling error of the system also cannot be corrected instantaneously but after the delay.

5.4.3 Intermittent Feedback

The intermittent feedback simulation and hardware testing were also performed to observe and analyze the system if such type of feedback exist the system without the time delay. The result of the simulation and hardware testing that had 200 intermittent cycles is shown in Figure 5-10 and Figure 5-11.



(A)

(B)

Figure 5-10: Simulation results for 200 intermittent cycles: (A) Setpoints vs. plant response and (B) actual output vs. model output illustrating model residual

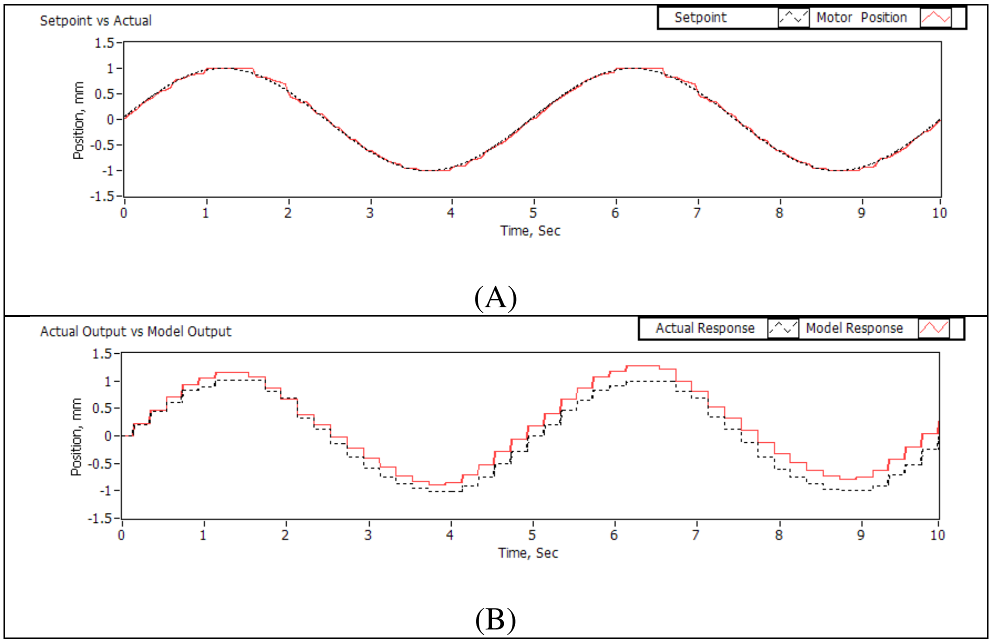


Figure 5-11: Prototype results when system has 100 cycles: (A) Setpoints vs. plant response and (B) the intermittent feedback of the plant and model illustrating model residual

Both the simulation results from the simulation and hardware show the affected feedback waveform of the system, in which the position of the system was assumed constant during the intermittent period. In addition, the output response of the plant in the hardware experiment also showed the drifting effect of the model as mentioned in the delay case. It can also be observed that plant output of the system was not smooth as every time the feedback sensor update the controller after each intermittent period, it created a small step input to feedback signal.

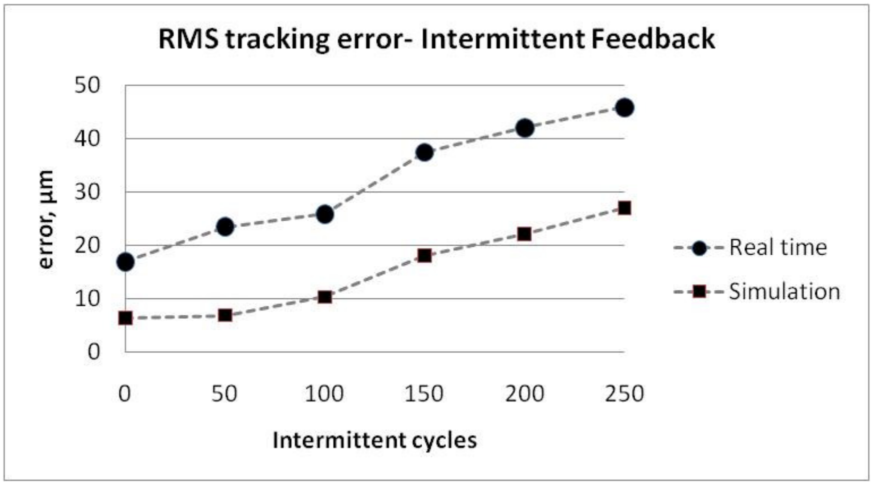


Figure 5-12: RMS position error of Smith Predictor with different intermittent cycle's scenario for both simulation and experimental testing

Table 5-2: RMS position error of Smith Predictor with different intermittent cycle's scenario for both simulation and experimental testing

	RMS Position Error,(µm)	
Intermittent feedback,(cycle)	**Real time**	**Simulation**
0	17.03	6.42
50	23.52	6.95
100	25.93	10.39
150	37.47	18.14
200	42.11	22.20
250	45.97	27.05

Table 5-2 and Figure 5-12 shows the RMS tracking error of the simulation and experimental results for the intermittent feedback case and it can be seen that the RMS tracking error of the system increase with the intermittent cycles. For a system that has 50 intermittent cycles, the RMS tracking error of the hardware experiments data is 23.52µm, but when is increased to 250 intermittent cycles, the RMS tracking error increased to 45.97µm. Based on both the simulation and hardware experiments, it can see that the

RMS tracking error increase with respect to the length of the intermittent cycles. During the intermittent period, there were lack of actual position feedback from the sensor to correct the path, and at the same time the controller was relying on the model to predict the path until the actual position from the sensor is obtained. Thus, it is critical that the model of the plant can be modeled as close as the plant to minimize the tracking error. As compare with the delay feedback case, the RMS tracking error increment is smaller because during intermittent feedback because the obtained error can be corrected instantaneously, but not in the delay feedback.

5.4.4 Delay and Intermittent Feedback

The simulation and hardware testing for the delay and intermittent feedback were performed and the results of the both the simulation and hardware testing when the system has 200ms delay and 100 intermittent cycles are shown in Figure 5-13 and Figure 5-14.

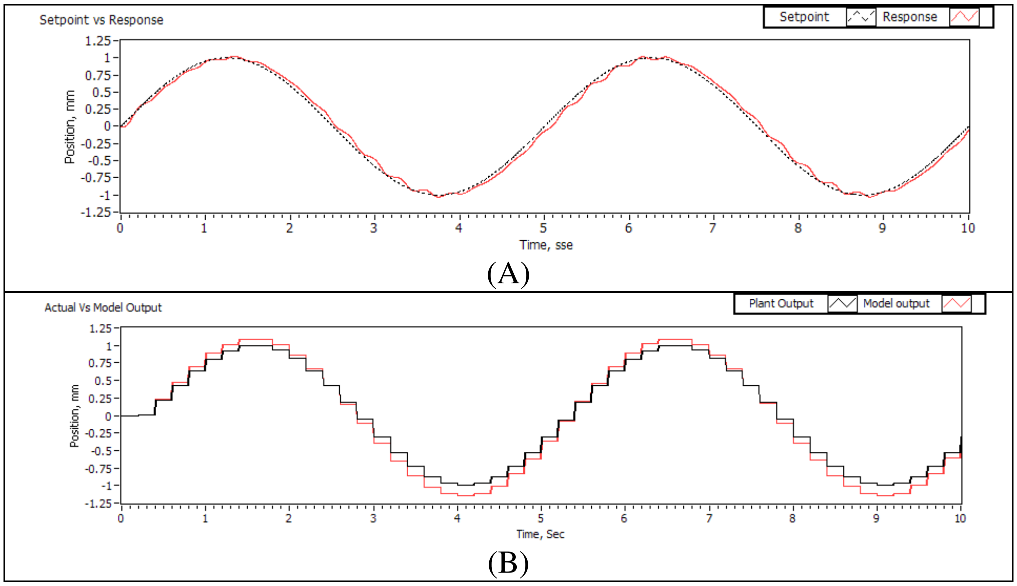


Figure 5-13: Simulation results when system has 200ms delay and 100 intermittent cycle's feedback: (A) Setpoints vs. plant response and (B) the delay output of the plant and model illustrating model residual.

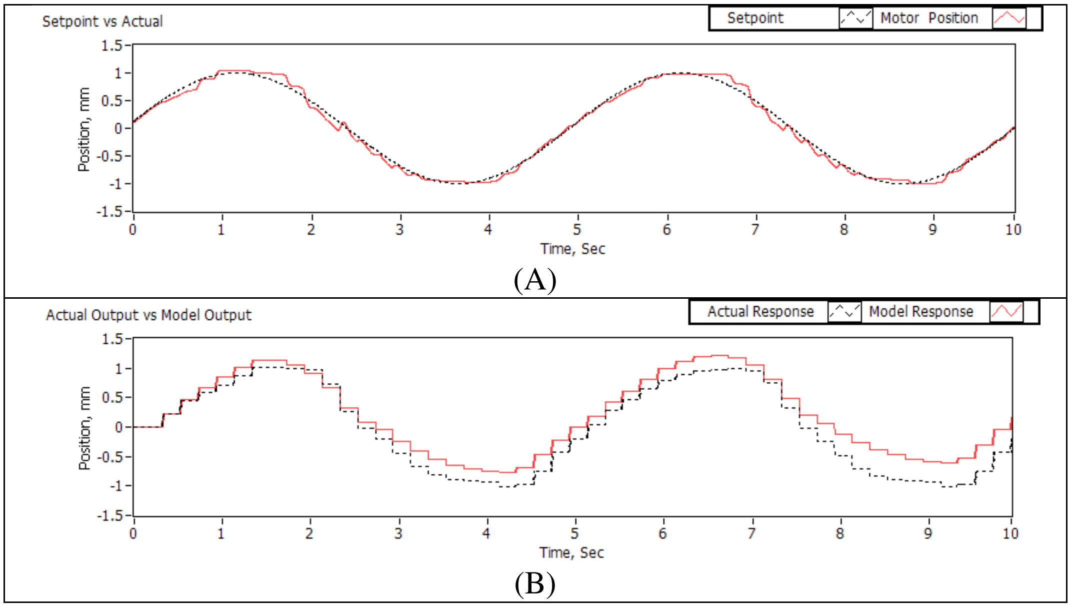


Figure 5-14: Prototype results when system has 200ms delay and 100 intermittent cycle's feedback: (A) Setpoints vs. plant response and (B) the delay output of the plant and model that shows the drifting for the model output from the actual plant output

As seen in the hardware experiment's results, the model residual as shown in Figure 5-4 can be increasing with respect to time, so correction to the model must be perform to enhance the tracking performance of the system. At the same time, it can be seen that the delay and intermittent feedback affect the path tracking performance of the simulation and hardware experiments as shown in both Figure 5-13(A) and Figure 5-14(A) respectively.

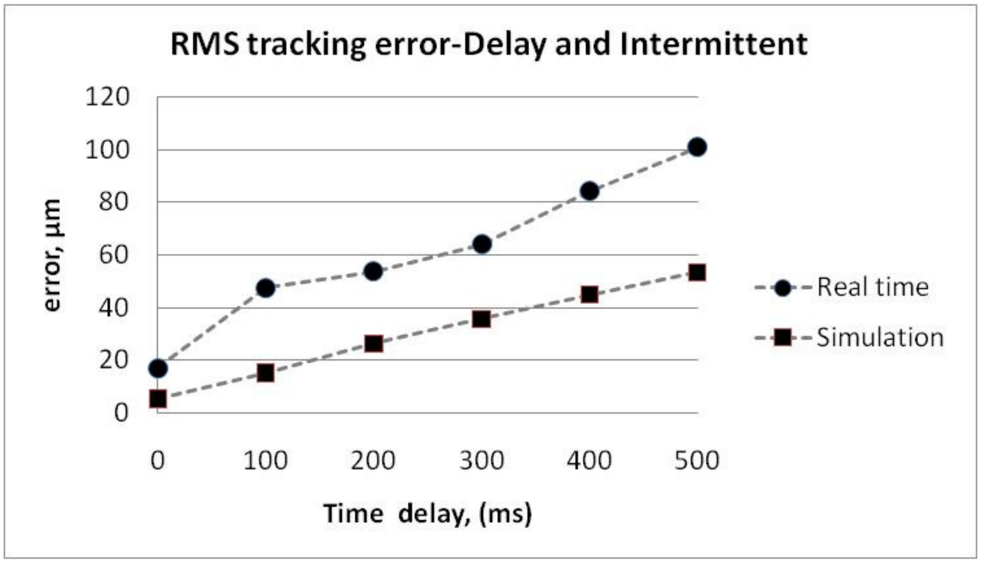


Figure 5-15: RMS tracking error for delay and intermittent feedback. Position error of Smith Predictor with different delay and intermittent cycle's scenario for both simulation and experimental testing

Table 5-3: position error of Smith Predictor with different delay and intermittent cycle's scenario for both simulation and experimental testing

		RMS Position Error,(µm)	
Intermittent feedback, (cycles)	**Time delay,(ms)**	**Real time**	**Simulation**
0	0	17.03	5.27
50	100	47.62	15.13
100	200	53.76	26.39
150	300	63.94	35.65
200	400	84.16	44.90
250	500	100.96	53.40

Figure 5-15 and Table 5-3 shows the summary of the simulation and experimental results of the Smith Predictor when delay and intermittent exist in the feedback. As seen in the results when there were 500ms time delay and 250 intermittent cycle's feedback, the RMS tracking error was 100.96µm as compare to 47.62µm when there were 100ms time delay and 50 intermittent cycles. Similar like the delay feedback case and the intermittent feedback case, the RMS tracking error of the system that has both the delay and intermittent feedback increases with the length of the delay and intermittent cycles Furthermore, the RMS error of this case is the highest among all three cases mentioned above because the feedback data is not only intermittent but delay, causing the controller not able to correct the modeling error and the actual measured error instantaneously.

5.5 Summary Remarks

When the delay or intermittent cycles of the system increases, the RMS tracking position error also increases. For the system that has only time delay the simulation results showed that there was at least 2μm of position error increment for every 100ms time delay increment whereas the experimental results also shows the increment of error but not linearly as seen in results. For the system that has intermittent cycles only, the simulation results also shows that the increment of errors with respect to the intermittent cycles. Based on the observation, the tracking error of the system in the hardware experiment is smaller than the tracking error in the delay feedback case. This is because the model residual of the system for the intermittent feedback case was corrected instantaneously when the residual was obtained. However in the delay feedback case, the obtained residual at time *t* is not the current model residual of the system but instead the previous residual at time t-τ, preventing the system from correcting the residual immediately. When comparing the worst case scenario of the delay feedback case and intermittent feedback case, the RMS tracking error difference between both cases was 18μm in the hardware experimental result. However, when comparing the system that has both delay and intermittent feedback, both the simulation and experimental results show a bigger increment in the RMS tracking position error. For example, in the case of 500ms delay and 250 intermittent cycles, the error was 100.96μm, almost three times larger than the tracking error when only 250 intermittent cycles feedback exists in the system.

One of the main underlying reasons for position error is model discrepancy between the model and the plant. The plant model is unable to accurately represent the actual plant, particularly un-modeled dynamics that were not included in the model used in the Smith Predictor. Thus, when model discrepancy error builds up, the error needs to be corrected instantaneously, but when the feedback is delay and also intermittent, the error can only be corrected when the actual plant data is obtained. This can cause large drift error and large compensation at each data point. Thus, research involving correcting and predicting the model residual during the intermittent feedback was performed to enhance the tracking performance of Direct Position Sensing.

CHAPTER SIX

MODIFIED SMITH PREDICTOR

6.1 Introduction

Based on the findings from the previous chapter, the model residual buildup during the intermittent path is shown to be detrimental to the path tracking performance of the system. Thus, this chapter presents augmentations to the Smith Predictor to minimize the model residual.

For a system that has intermittent feedback, the modeling residual can only be corrected when the actual feedback is obtained. In addition, the modeling residual may increase during the intermittent period, which potentially causes instability in the system. Hence, this chapter shows algorithms that were developed to augment the original Smith Predictor to address these challenges.

First, a prediction algorithm of the intermittent path based on the historical obtained information was created to minimize the modeling residual. Second, a Proportional controller is added in the inner loop of the Smith Predictor to update the control action to the plant model based on the model residual obtained. Third, the combined intermittent path prediction together with the inner Proportional controller was also implemented to enhance the tracking performance of the system.

6.2 Intermittent Path Prediction Algorithms

As seen in Figure 6-1, the measured output signal, *x(z-τ)* from the vision sensor, *V(z)* is delayed and intermittent. During the delay and intermittent cycles, the plant model, $G_m(z)$ is used to provide the estimated plant output based on the plant model. In order to correct the prediction, the output of the plant model was also modeled to have delay and intermittent feedback so that the prediction error, $e_m(z+ \tau)$ can be obtained to correct the estimated output from the model.

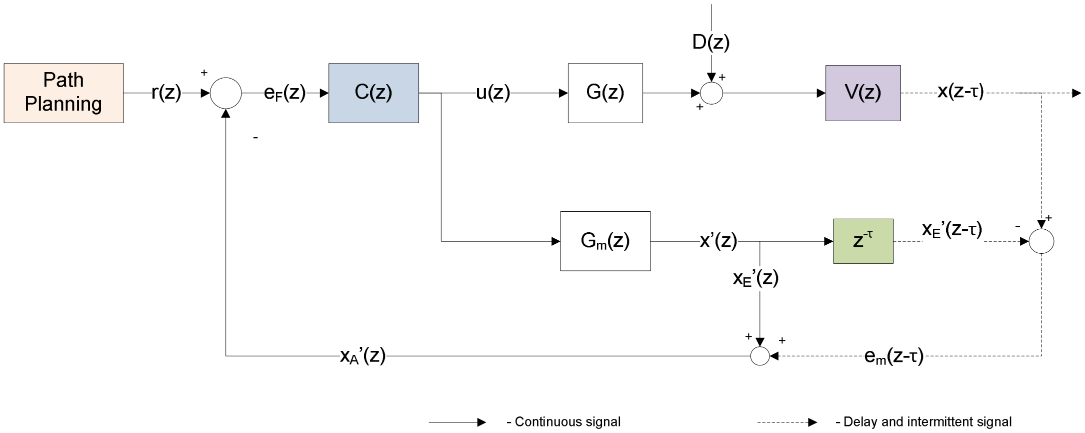


Figure 6-1: Smith Predictor block diagram: The solid line is a continuous signal and the broken line is the intermittent signal

Based on the original Smith Predictor architecture, it was assumed that the current position maintains a constant value until the next obtained position was available, a signal equivalent to a Zero Order Hold (ZOH) model as shown in Figure 6-2(A). Hence, the model residual, $e_m(z)$ also has the ZOH type of waveform as shown in Figure 6-2(B). In addition, every time the system acquires the updated feedback, it creates a step input of 10 to 40µm ,as seen in Figure 6-2(B), to feedback signal of the controller, $x_A'(z)$.and cause $x_A'(z)$ to be noisy as shown in Figure 6-3.

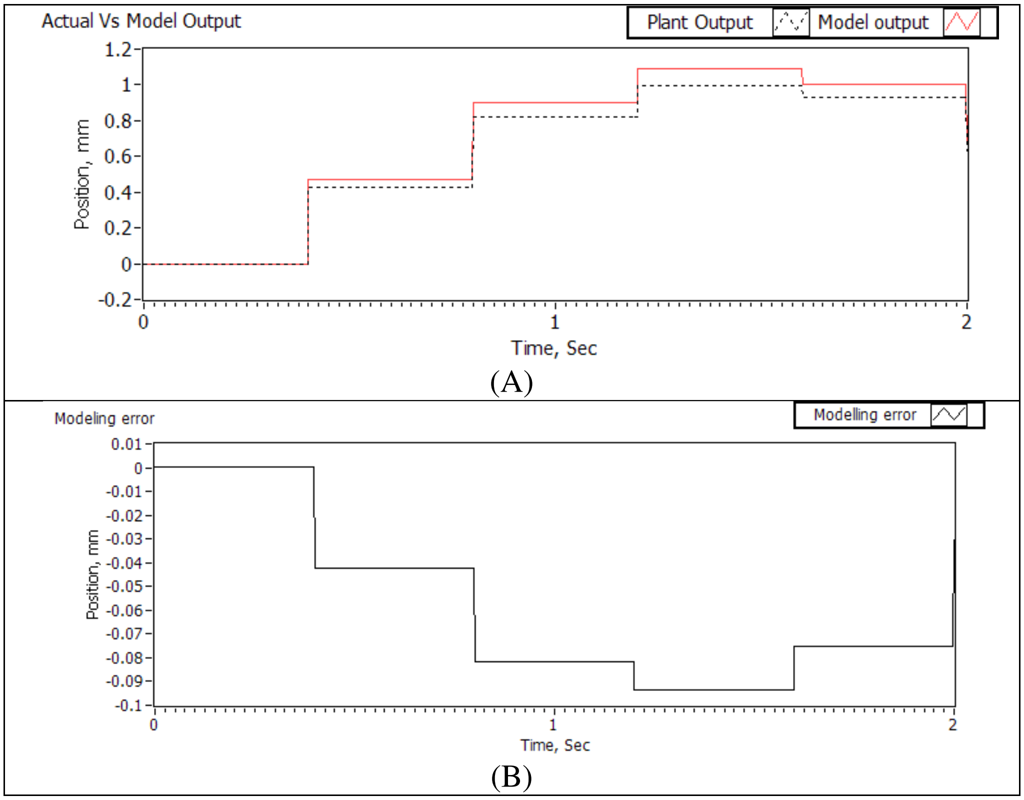


Figure 6-2: Intermittent feedback (A) actual and model output, (B) modeling discrepancy between the actual and model output.

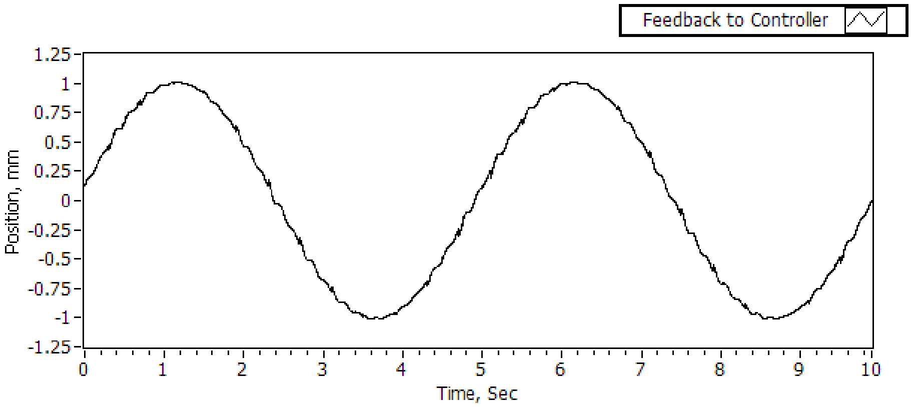


Figure 6-3: Feedback signal to the controller, *x_A'(z)*.The unevenness of the curves was caused by the update of the model residual every time the system obtained the intermittent update

Thus, the goal of the intermittent path prediction algorithm is to estimate the path of the plant during the intermittent period by extrapolating the historical obtained position. This prediction path provides closer estimation of the model residual as compared to the ZOH model used in the Smith Predictor. Three extrapolation algorithms based on the obtained historical actual data were added to the Smith Predictor architecture to estimate the intermittent path: First Order Hold (FOH), Second Order Hold (SOH), and Third Order Hold (TOH) models. Hence, by predicting the intermittent path rather than holding the position at a static position, a smoother model residual signal is provided as compared with a ZOH signal.

6.2.1 Extrapolation Method

As seen in Figure 6-4, instead of assuming that the position is maintained at its previous location, the prediction algorithm uses points of the historical feedback to estimate the intermittent path. For the FOH extrapolation, the algorithm used the current obtained position, *y(t)* and the previous obtained position, *y(t-P)* to extrapolate the path for *y(t+P)* as shown in Figure 6-4. Eq. (6.1) shows the equation used to extrapolate the path from *t* to *t+P* and A_1 and B_1 were calculated based on *y(t)* and *y(t+P)*.

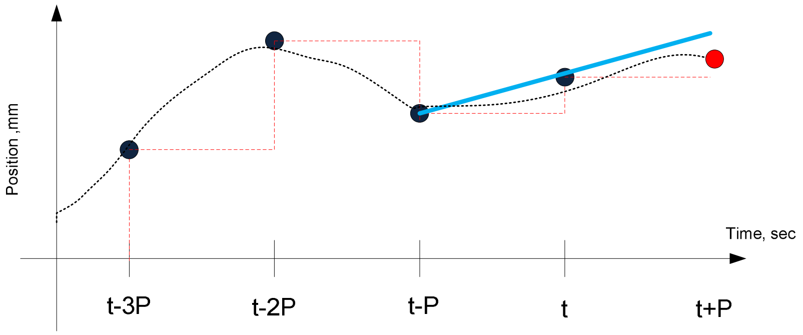


Figure 6-4: First Order Hold. Use the current point, *y(t)* and previous point, *y(t-P)* where *P* is the intermittent interval to extrapolate the path from *t* to *t+P* shown in solid blue line. The black dotted line represent the setpoints, and the red broken line show the ZOH

$$y(t) = A_1 t + B_1 \Big|_{t=t:t+P} \tag{6.1}$$

where

$$\begin{aligned} y(t) &= A_1 t + B_1 \\ y(t-P) &= A_1 (t-P) + B_1 \end{aligned}$$

For the SOH, the algorithm uses the current obtained position, *y(t)* and the previous two obtained positions, *y(t-P)* and *y(t-2P)* to extrapolate the path for *y(t+P)* and the Eq.(6.2) shows the equations used to extrapolate the path from *t* to *t+P* by solving the parameters A_2, B_2 and C_2 based on *y(t)*, *y(t-P)* and *y(t-2P).*

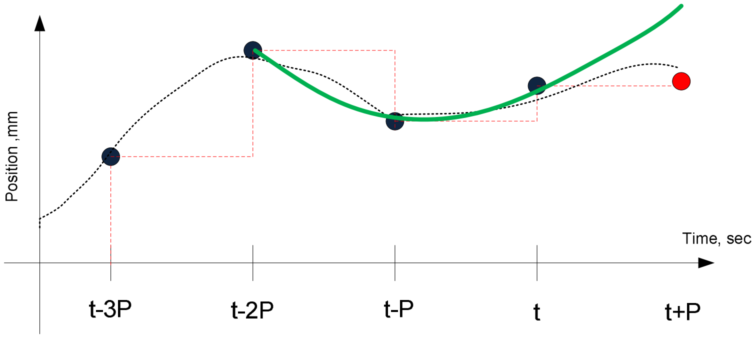


Figure 6-5: Second Order Extrapolation: Use the current point, *y(t)* and previous point, *y(t-P)* and *y(t-2P)* where *P* is the intermittent interval to extrapolate the path from *t* to *t+P* shown in solid green line. The black dotted line represent the setpoints, and the red broken line show the ZOH

$$y(t) = A_2 t^2 + B_2 t + C_2 \Big|_{t=t:t+P} \tag{6.2}$$

where

$$\begin{aligned}
&y(t) = A_2 t^2 + B_2 t + C_2 \\
&y(t-P) = A_2 (t-P)^2 + B_2 (t-P) + C_2 \\
&y(t-2P) = A_2 (t-2P)^2 + B_2 (t-2P) + C_2
\end{aligned}$$

Similarly for the TOH model, three previously obtained positions, *y(t-P)*, *y(t-2P)* and *y(t-3P)* together with the current obtained position, *y(t)* will be used to extrapolate the path for *y(t+P)* as shown in Figure 6-6. Eq. (6.3) was used to extrapolate the path from *t* to *t+P* by solving the parameters A_3, B_3, C_3 and D_3 based on *y(t)*, *y(t-P)*,*y(t-2P)* and *y(t-3P).*

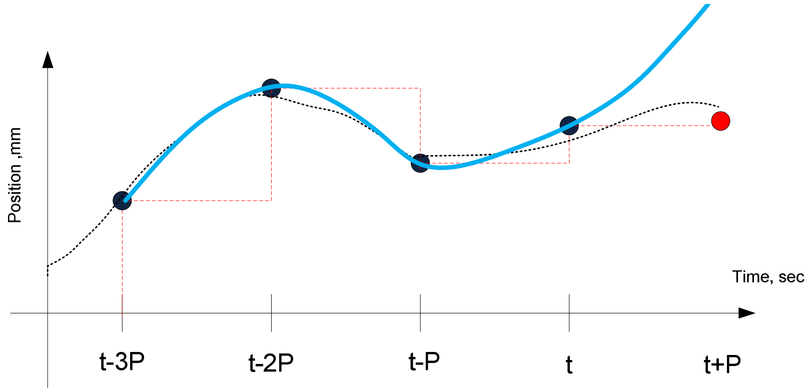


Figure 6-6: Third Order Extrapolation: Use the current point, *y(t)* and previous point, *y(t-P), y(t-2P)* and *y(t-3P)* where *P* is the intermittent interval to extrapolate the path from *t* to *t+P* shown in solid blue line. The black dotted line represent the setpoints, and the red broken line show the ZOH

$$y(t:t+p) = A_3t^3 + B_3t^2 + C_3t + D_3\Big|_{t=t:t+P} \tag{6.3}$$

where

$$\begin{aligned}
y(t) &= A_3t^3 + B_3t^2 + C_3t + D_3 \\
y(t-P) &= A_3(t-P)^3 + B_3(t-P)^2 + C_3(t-P) + D_3 \\
y(t-2P) &= A_3(t-2P)^3 + B_3(t-2P)^2 + C_3(t-2P) + D_3 \\
y(t-3P) &= A_3(t-3P)^3 + B_3(t-3P)^2 + C_3(t-3P) + D_3
\end{aligned}$$

As seen in Figure 6-4, Figure 6-5 and Figure 6-6, the FOH has the more accurate extrapolated path as compare to the higher order methods like the SOH and TOH. The prediction of these algorithms is not only dependent on the setpoints' underlying waveform but also the measurement data from the sensor. This effect can be further observed in both the simulation and hardware experimental results

6.2.2 *Results*

Simulation and hardware experiments of the extrapolation algorithms were performed and the extrapolation algorithm of the plant and the model, $P_A(z)$ and $P_M(z)$ respectively, were integrated to the Smith Predictor architecture as shown in Figure 6-7. Similar to chapter five, a 0.2Hz sine wave was used as the setpoints for the system, and the FOH, SOH and TOH algorithm are tested individually with respect to different feedback cases. The RMS tracking error of the simulation was obtained so that the performance of each algorithm can be compared. This extrapolation algorithm is applied to the system with intermittent feedback. Two simulations were performed: 1) system with intermittent feedback only and 2) system with delay and intermittent feedback.

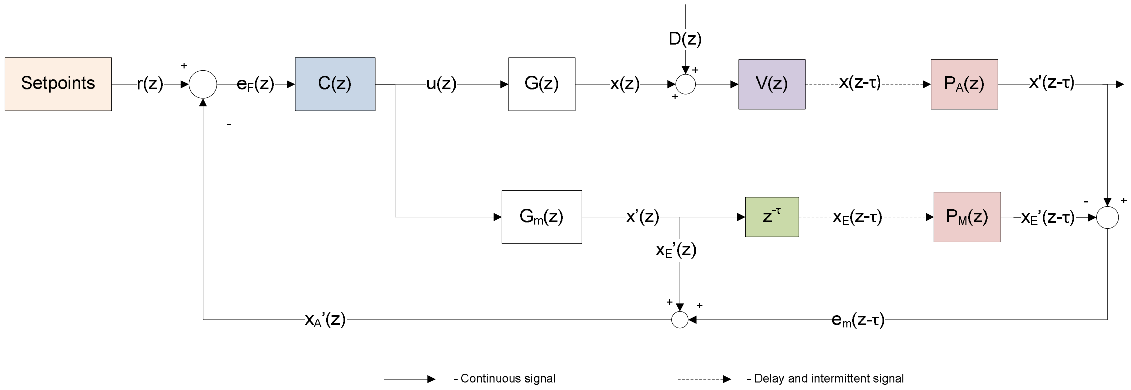


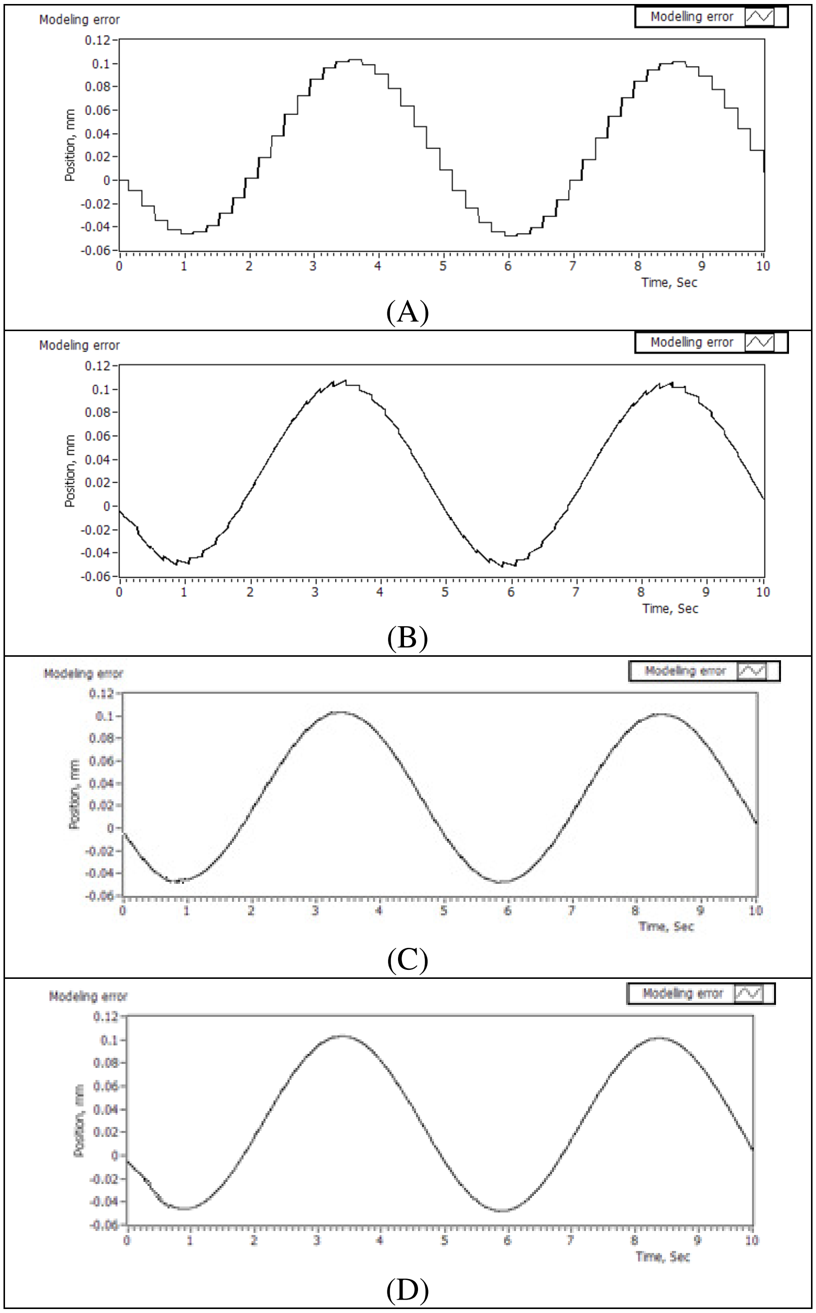
Figure 6-7: Smith Predictor with intermittent path prediction algorithm. The Intermittent path prediction algorithm, $P_A(z)$ and $P_M(z)$ were added to the original Smith Predictor architecture

6.2.2.1 Intermittent Feedback

Simulations and hardware experiments using the three described extrapolation methods in the intermittent feedback case were performed. The same reference trajectory used in the hardware testing was used in the simulation. The goal of the simulation is to compare the performance of the extrapolation algorithms at equivalent controller setups. Therefore, the gains of the PI controller's gains for all cases were configured having the same gains to compare the behaviors of the system. In the intermittent case, the time delay τ will be equal to zero.

Figure 6-8 shows the comparison of the modeling error waveform, $e_m(z)$ using the Smith Predictor together with the intermittent path prediction algorithms and it can be seen that the extrapolation algorithms a smoother signal, $e_m(z)$ as compare to the ZOH signal from the original Smith Predictor architecture, especially when the SOH and TOH were used.

Based on the simulation results, the tracking performance of the Smith Predictor with intermittent path prediction has smaller RMS tracking error than the original Smith Predictor as seen in Table 6-1 and Figure 6-9. The extrapolation algorithms help to reduce the tracking error As seen in Table 6-2, there was at least 32% RMS tracking error reduction by using the intermittent path prediction algorithm when there was 50 intermittent cycles feedback. For the 100 intermittent cycles feedback case, there was at least 53% RMS tracking error reduction.



(A)

(B)

(C)

(D)

Figure 6-8: Error waveform *e_m* of the Smith Predictor with intermittent path prediction algorithm in simulation . (A) ZOH, (B) FOH, (C) SOH and (D) TOH

Table 6-1: Simulation of the extrapolation algorithms in Smith predictor during intermittent feedback

	RMS tracking error, µm			
Intermittent Feedback(cycles)	**ZOH**	**FOH**	**SOH**	**TOH**
0	4.67	4.67	4.67	4.67
50	7.47	5.02	4.75	4.73
100	13.74	6.35	4.64	4.63

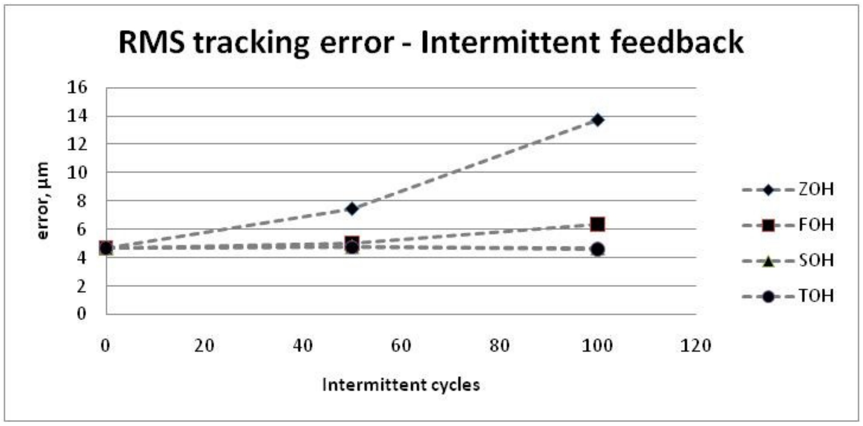


Figure 6-9: Simulation results of the extrapolation algorithms during intermittent feedback. Based on the simulation results, the SOH has the lowest RMS tracking error for both the 50 intermittent cycles and 100 intermittent cycles

Table 6-2: Percentage of error reduction as compare to ZOH

	Error comparison, %		
Intermittent Feedback(cycles)	**FOH**	**SOH**	**TOH**
50	-32.78%	-36.40%	-36.67%
100	-53.81%	-66.19%	-66.32%

Based on the simulation results when the system has 100 intermittent cycles, the intermittent path prediction helps to improve the tracking performance of the system by at least 6μm. However, the selection of the algorithms used in the Smith Predictor will depend on the setpoints and also the amount of noise and disturbance in the system. As the intermittent cycles increase, the prediction error of these extrapolation algorithm will also increase, which can lead to instability in the system.

Hardware experiments were also performed and the results are presented in Table 6-3 and Figure 6-10. For a system that has 50 intermittent cycles, the SOH method has the best performance in which the RMS tracking error of the system was reduced by 19.76% and for a system that has 100 intermittent cycles, the FOH has a error reduction of 39.66%. Thus, it can be seen that TOH is not always the better method for the prediction, and it is based on influences such as the length of the intermittent cycles, the measurement noises, and other disturbances.

Table 6-3: Hardware experimental results for intermittent feedback

	RMS tracking error, µm			
Intermittent, cycles	**ZOH**	**FOH**	**SOH**	**TOH**
0	14.2	14.2	14.2	14.2
50	20.19	16.2	16.76	15.55
100	31.64	19.09	43.92	33.93

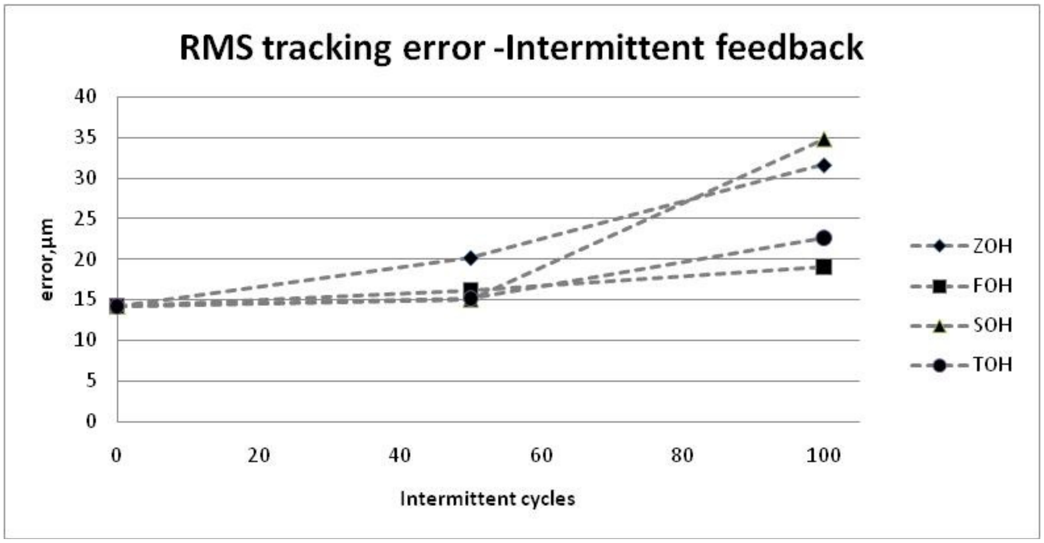


Figure 6-10: Hardware experiment results for intermittent feedback For 50 intermittent cycles feedback, the TOH has the lowest RMS value and for the 100 intermittent cycles feedback , the FOH has the lowest RME tracking error

Table 6-4: Error comparison for intermittent feedback case in hardware experiment with the ZOH

	Error reduction, %		
Intermittent, cycles	**FOH**	**SOH**	**TOH**
50	-19.76%	-16.99%	-22.98%
100	-39.66%	38.81%	7.24%

6.2.2.2 Delay and Intermittent Feedback

Simulations were also performed for the case of delay and intermittent feedback. In this case, the obtained feedback at time *t*, is a delay feedback but the goal of this algorithm is to predict the intermittent path so that the model residual signal is not in a ZOH type of waveform. Based on the simulation result showed in Table 6-5 and Figure 6-11, it can be observed that the RMS tracking error of the FOH, SOH and TOH reduced error as compared with ZOH. For the case of 100 ms delay and 50 intermittent cycles, the

RMS tracking error is reduced by 20.44% using FOH, by 21.86% using SOH and by 22.07% using TOH.

Table 6-5: Simulation of the extrapolation algorithm in Smith predictor during delay and intermittent feedback

	RMS tracking error, µm			
Delay (ms) ǀ Intermittent feedback (cycles)	**ZOH**	**FOH**	**SOH**	**TOH**
0	4.78	4.78	4.78	4.78
100 ǀ 50	13.65	10.86	10.66	10.63
200ǀ100	24.75	17.94	16.75	16.75

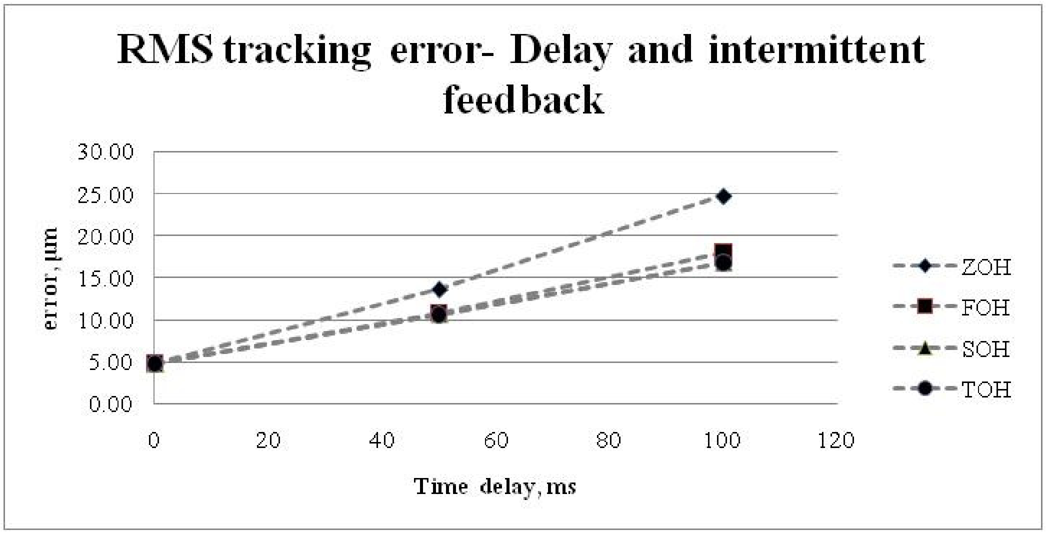


Figure 6-11: RMS tracking error for delay and intermittent feedback. The simulation results shows that both the SOH and TOH have lower RMS tracking error for both cases

Table 6-6: Error comparison when intermittent path prediction is added to the Smith Predictor during delay and intermittent feedback

	Error reduction, %		
Delay (ms) | Intermittent feedback (cycles)	**FOH**	**SOH**	**TOH**
100 | 50	-20.44%	-21.86%	-22.07%
200|100	-27.53%	-32.32%	-32.32%

Hardware experiments for the intermittent path prediction were also performed and the results are presented in Figure 6-12. The percentage different between the RMS tracking error of each extrapolation algorithm to the original Smith Predictor algorithm's RMS tracking error is shown in Table 6-7.

Table 6-7: Hardware experimental results for delay and intermittent feedback

	RMS tracking error, µm			
Delay (*ms*)| intermittent , cycles	**ZOH**	**FOH**	**SOH**	**TOH**
0	14.2	14.2	14.2	14.2
100|50	57.8	73.67	69.5	77.46
200|100	70.28	102.74	84.43	99.33

Table 6-8: Error comparison for delay and intermittent feedback in hardware

	Error comparison, %		
Delay (*ms*)| intermittent cycles	**FOH**	**SOH**	**TOH**
100|50	27.46%	20.24%	34.01%
200|100	46.19%	20.13%	41.33%

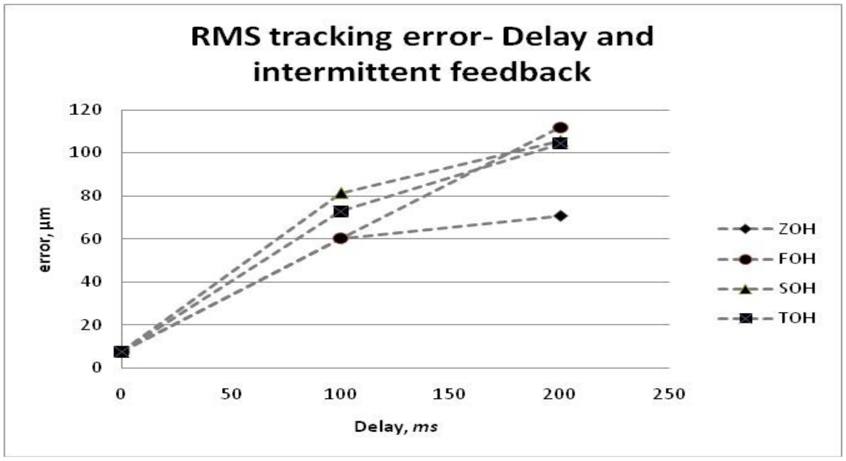


Figure 6-12: Hardware experimental results for delay and intermittent feedback . The FOH has lowest RMS tracking error during the 100*ms* delay and 50 intermittent cycles case, but the ZOH has the lowest when the time delay and intermittent cycles increase to 200*ms* and 100 intermittent cycles

It can be observed that the intermittent prediction algorithm did not perform as well as expected in the simulation. For the case of 100ms delay and 50 intermittent cycles, the RMS tracking error increases 27.46% for the FOH, 20.24% for the SOH and 34.01% for the TOH. For the case of 200ms delay and 100 intermittent cycles, the RMS tracking error increase by 46.19% using FOH, 20.13% using SOH and 41.33% using TOH. Thus, it can be seen that the Smith Predictor with ZOH works better in the system that has both delay and intermittent.

6.2.2.3 Summary

Based on the simulation results, the intermittent path prediction algorithm also works in the system that has both the delay and intermittent feedback but just that the tracking error improvement of the system is not as significant as compared with the system that has intermittent feedback alone.

Based on the hardware experimental results, the intermittent prediction algorithm works well when the system has intermittent feedback only, but when the algorithm is used in the delay and intermittent feedback case, the RMS tracking error increases. Potential reason for the error increment can due to the delay feedback of the system , as when the delay feedback is obtained, it was also fed back to the system causing a small step input to the feedback, which also making the path to be uneven. Since the intermittent path prediction uses historical data to extrapolate the path, the small bump in the feedback signal might cause the next prediction path to be inaccurate. In addition, the choice of extrapolation algorithm used in the system will also depend on the setpoints signal and the amount of noise of the system. This is because this intermittent path prediction algorithm is depending on the historical points, so if the previous obtained value has sufficiently high position error, then this will cause the intermittent predicted path to be inaccurate

6.3 Model Input Corrector

The second approach was to include a model input corrector to update the plant model. As seen in chapter five, the validation of the plant model with the actual plant shows that the model residual will always exist in a model based control system, and the model residual can increase with respect to time or operation conditions if the model is not updated. Thus, the objective of adding the model input corrector, $C_m(z)$ is to update the control action of the primary controller to the model $G_m(z)$ based on the obtained model residual, $e_m(z)$ as shown in Figure 6-13.

As seen in chapter five in both the simulation and experimental results, that the plant model's output has the tendency to drift away from the actual plant output. Although the model residual, $e_m(z)$ was obtained and added to the feedback signal to the controller, $x_A(z)$ to be corrected by the primary controller of the system, $C(z)$, the model output, $x'(z)$ of the system was still uncorrected. Thus, the goal of adding a additional Proportional controller, $C_m(z)$ is to update the control action to the model, $u_m(z)$ based on the obtained residual as illustrated in so that the model output of the system is close to the actual plant output to have a better predictor. Stability analysis of adding $C_m(z)$ into the original Smith Predictor was performed, and the eigenvalues of the system were all located on the left side of the plane, which shows this to be a stable control architecture. (Refer to Appendix B).

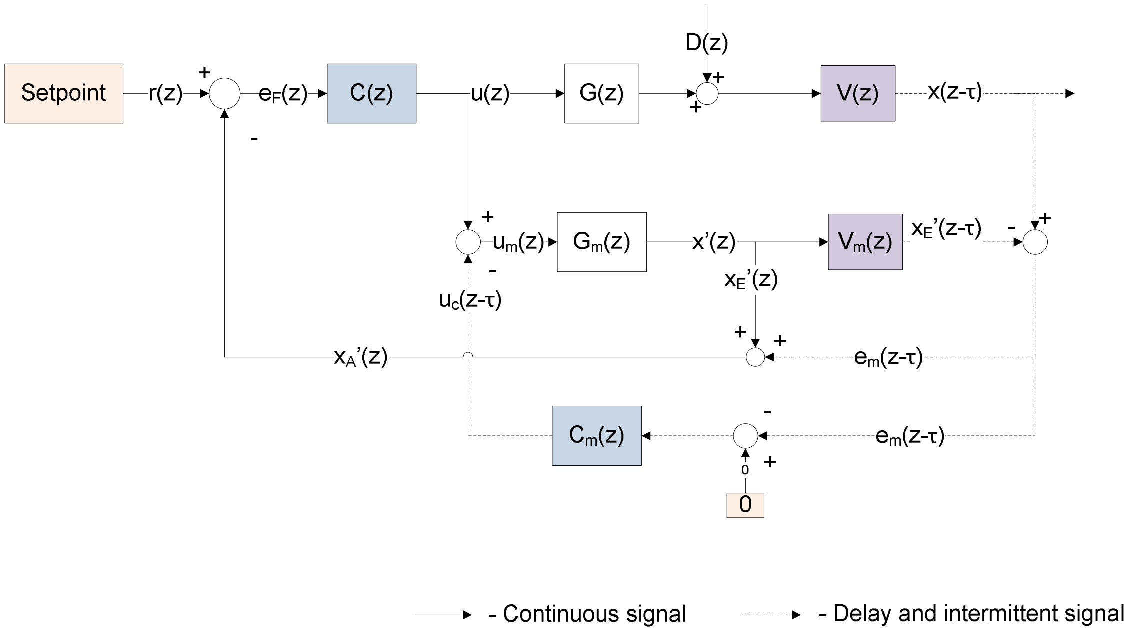
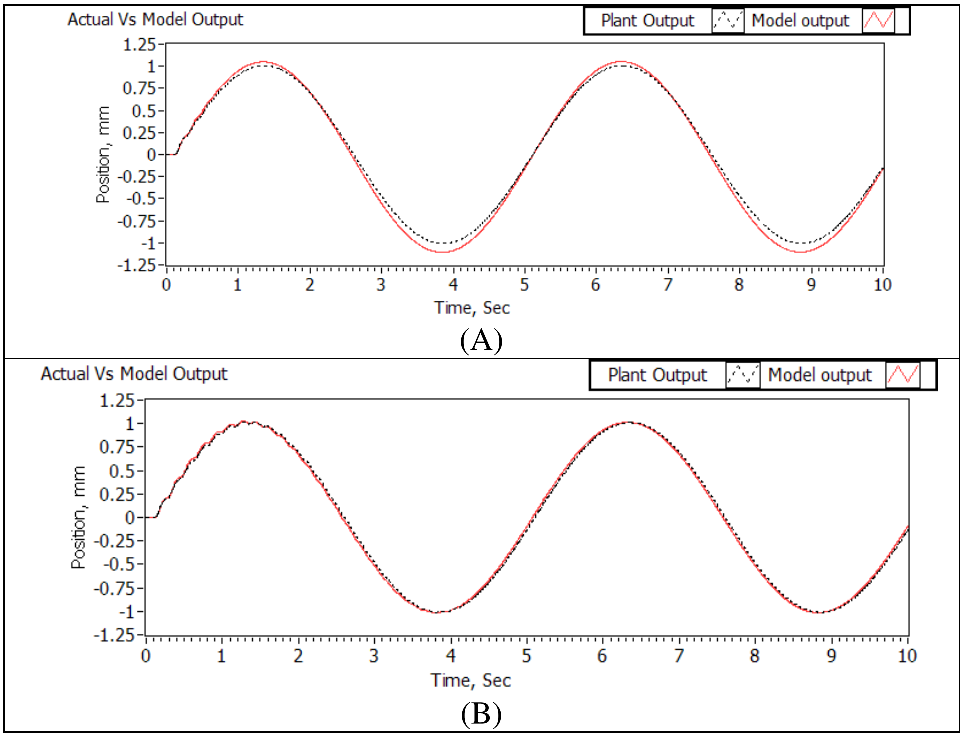


Figure 6-13: Model input corrector: An additional controller $C_m(z)$ was added to the Smith Predictor

6.3.1 Results

Simulations and hardware experiments of the Smith Predictor with model input corrector were performed. The RMS tracking error between the setpoints and the system response was used as the performance metric. Thus, the performance of the path tracking and the performance of the model input corrector can be quantified. The model input corrector algorithm was tested with all three different feedback scenarios: 1) delay, 2) intermittent and 3) delay and intermittent feedback. Figure 6-14 shows the comparisons of the output signal of the plant and model with and without the model input corrector. Based on the simulation results, the model input corrector helps to maintain the model output closer to the actual plant output as illustrated in Figure 6-14B. Figure 6-15 and

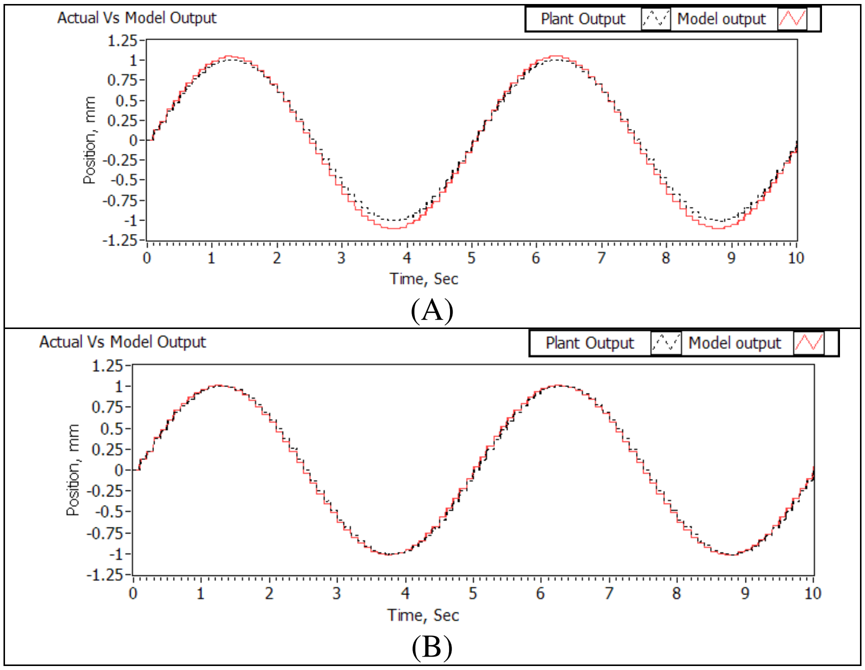
Figure 6-16 shows the comparison of the model input corrector performance's simulation results for the intermittent feedback case, and the delay and intermittent feedback case respectively.



(A)

(B)

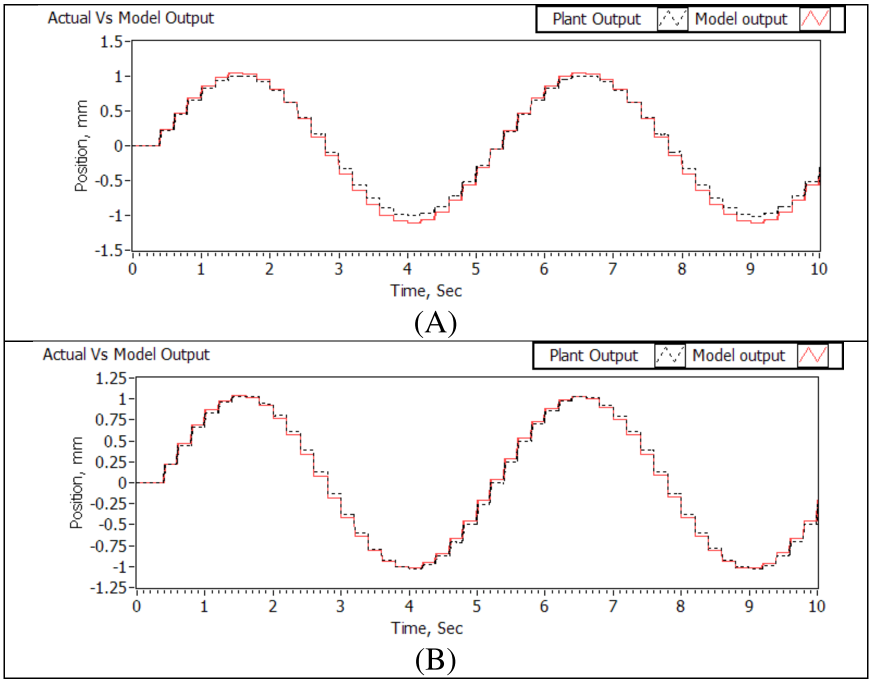
Figure 6-14: Simulation results of the plant and model output when there is 100ms delay: (A) without Model input corrector and (B) with Model Discrepancy Corrector.



(A)

(B)

Figure 6-15: Simulation results of the plant and model output when there is 100ms delay: (A) without Model input corrector and (B) with Model Discrepancy Corrector.



(A)

(B)

Figure 6-16: Simulation results of the plant and model output when there is 200ms delay and 100 intermittent cycles : (A) without Model input corrector and (B) with Model Discrepancy Corrector.

Table 6-9 shows the RMS position tracking error of the Smith Predictor with model input corrector for both the simulation and hardware experiments. It can be seen that the Smith Predictor with model input corrector improved the tracking error and the error reduction percentage of each case is presented in Table 6-10.

Table 6-9: Comparison of the Smith Predictor with and without the model input corrector in both simulation and hardware experiment

		Simulation		Hardware	
		RMS Position Error, *µm*		**RMS Position Error, *µm***	
Time delay, *ms*		**Without Model Updater**	**Model Updater**	**Without Model Updater**	**Model Updater**
100		10.80	8.20	33.85	26.74
200		17.09	12.01	59.43	37.77
Intermittent, cycles		**Without Model Updater**	**Model Updater**	**Without Model Corrector**	**Model Updater**
50		8.05	6.99	23.4	21.72
100		14.67	10.75	33.28	27.9
Intermittent, cycles	**Time delay, ms**	**Without Model Updater**	**Model Updater**	**Without Model Corrector**	**Model Updater**
50	100	14.09	10.27	57.8	45.5
100	200	25.49	18.58	70.28	56.66

Table 6-10: RMS tracking error reduction, for simulation and hardware experiment

		Error reduction, %	
Time delay, *ms*		**Simulation**	**Hardware**
100		-24.13%	-21.00%
200		-29.70%	-36.45%
Intermittent, cycles		**Simulation**	**Hardware**
50		-13.11%	-7.18%
100		-26.70%	-16.17%
Intermittent, cycles	**Time delay, ms**	**Simulation**	**Hardware**
50	100	-27.10%	-21.28%
100	200	-27.13%	-19.38%

For the system that has 100ms delay, the simulation results show a 24.13% reduction in RMS tracking error with the implementation of the model input corrector, and 21.00% of reduction in the hardware experiment. Similarly, for system that has 100 intermittent cycles, the simulation result shows a 13.11% reduction in the RMS tracking error; and 7.18% reduction in the hardware experiment. For the case that has 200ms time delay and 100 intermittent cycles, the simulation results shows a 27.13% reduction in the RMS tracking error; and 19.38% reduction in the hardware experiment.

6.3.2 Summary Remarks

The result of the simulation and hardware experiments of implementing model input corrector to the Smith Predictor shows that this scheme helps to improve the tracking performance of the system as seen in the data and analyses mentioned above. Since the goal of this test is to analyze and observe the behavior of the system so the gain of the controller and the model input corrector was not optimally tuned yet. Thus, the performance of the model input corrector can further be optimized with better gain turning. At the moment, the model input corrector uses only a Proportional controller and the gain of the controller, *Kp* was tuned heuristically so that the model output maintain as close as the plant output.

6.4 Combining the Intermittent Path Predictor via Extrapolation with Model Input Corrector

Based on the simulation and experimental results of the two additional method mentioned above, this chapter shows the results of the Smith Predictor when the intermittent path prediction algorithm: $P_A(z)$ and $P_m(z)$ and the model input corrector, $C_m(z)$ were integrated together to the Smith Predictor architecture as described in Figure 6-17.

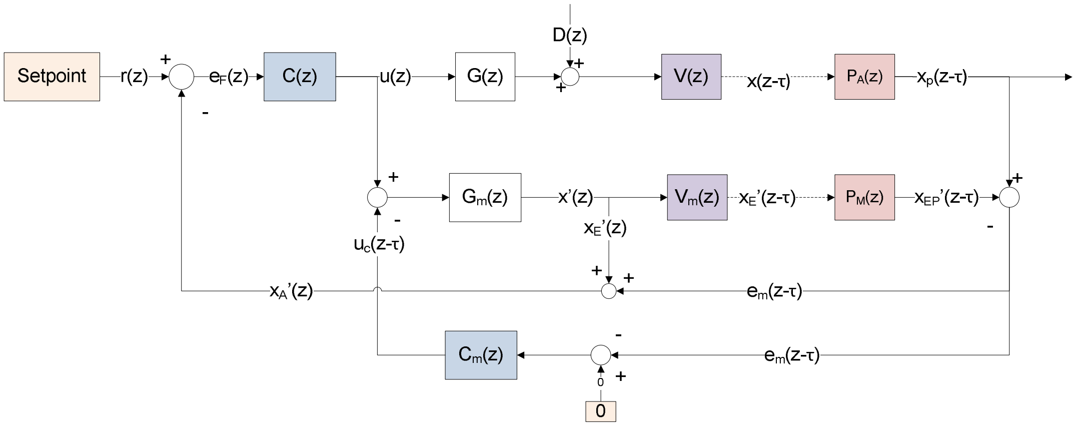


Figure 6-17: Smith Predictor with intermittent path prediction and model input corrector. When both the model input corrector, $C_m(z)$and the intermittent path predictions for the plant and model*, $P_A(z)$and $P_m(z)$* respectively were added to the Smith Predictor architecture

6.4.1 Results

Simulation and hardware experiments of the combined algorithms into the Smith Predictor architecture were performed for two feedback cases: 1) intermittent and 2) delay and intermittent feedback. A 0.2Hz sine wave was used to generate the setpoints of the system. The RMS tracking error of the system was used to as the performance metric.

First, simulation of original Smith Predictor architecture was performed indicated by ZOH. Then, the intermittent path prediction with all three extrapolation methods: first order hold, second order hold and third order hold was performed, denoted as FOH, SOH and TOH respectively. After that, these four mentioned cases were re-simulated again by integrating with the model input corrector, denoted by ZOH_M, FOH _M, SOH _M and TOH_M respectively.

6.4.1.1 Intermittent Feedback

Simulation and hardware experiments for the Smith Predictor with the combined algorithms were performed and the results are presented in Table 6-11 and Table 6-12 respectively. Based on the simulation results, it can be observed that the Smith Predictor with the combined algorithms, denoted as ZOH_M, FOH_M, SOH_M, and TOH_M minimized the RMS tracking error of the system as compare to using the intermittent prediction algorithms without the model input corrector. For the case of 100 intermittent cycles as presented in Table 6-11, TOH_M has the best tracking performance and the RMS tracking error for this scheme is 4.43μm.

Table 6-11: Simulation result for intermittent feedback

	RMS tracking error, *μm*							
Intermittent Feedback(cycles)	**ZOH_M**	**ZOH**	**FOH_M**	**FOH**	**SOH_M**	**SOH**	**TOH_M**	**TOH**
0	4.78	4.67	4.78	4.67	4.78	4.67	4.78	4.67
50	6.12	7.47	4.69	5.02	4.81	4.75	4.70	4.73
100	9.70	13.74	4.76	6.35	4.57	4.64	4.43	4.63

In the hardware experiment, the Smith Predictor with the combined algorithms has lower RMS tracking error as compare to the intermittent path prediction without the model input corrector as seen in Table 6-12. In the case of 100 intermittent cycles, the ZOH_M has best tracking performance among other schemes and the RMS tracking error for this scheme was 19.06µm. When comparing with control scheme, the FOH was also performing well in which the RMS tracking error is 19.09µm. However, when comparing with SOH and TOH, the RMS tracking error increases to 43.32µm and 33.93µm respectively. However, with model input corrector, the RMS tracking error of both algorithms was reduced to 34.77µm and 22.62µm respectively. Thus, the model input corrector helps to improve the tracking performance of the intermittent path prediction.

Table 6-12: Hardware experimental result for intermittent feedback

	RMS tracking error, µm							
Intermittent Feedback(cycles)	**ZOH_M**	**ZOH**	**FOH_M**	**FOH**	**SOH_M**	**SOH**	**TOH_M**	**TOH**
0	14.2	14.2	14.2	14.2	14.2	14.2	14.2	14.2
50	18.04	20.19	15.73	16.2	15.08	16.76	15.13	15.55
100	19.06	31.64	18.37	19.09	34.77	43.92	22.62	33.93

6.4.1.2 Delay and Intermittent Feedback

Simulation and hardware experiments for the Smith Predictor with the combined algorithms were also performed for the delay and intermittent feedback case. Based on the simulation results as seen in Table 6-13, the Smith Predictor with the combined algorithms has lower RMS tracking error as compare to using the intermittent prediction algorithms without the model input corrector. For instance, in the case of 200ms delay and 100 intermittent cycles as presented in Table 6-13, TOH_M has the better tracking performance as compare to other scheme and the RMS tracking error for this scheme was 11.49µm.

Table 6-13: Simulation results for delay and intermittent feedback

	RMS tracking error, *µm*							
Delay (*ms*) | Intermittent feedback (cycles)	**ZOH_M**	**ZOH**	**FOH_M**	**FOH**	**SOH_M**	**SOH**	**TOH_M**	**TOH**
0	4.78	4.78	4.78	4.78	4.78	4.78	4.78	4.78
100 | 50	9.66	13.65	7.99	10.86	8.01	10.66	7.99	10.63
200|100	17.80	24.75	11.95	17.94	11.68	16.75	11.49	16.75

For hardware experiments, the Smith Predictor with the combined algorithms also shows the same performance in which this control scheme has lower RMS tracking error as compare to the intermittent path prediction without the model input corrector. Table 6-14 shows the experimental results obtained from the prototype. When the system has 200ms delay and 100 intermittent cycles, TOH_M has the least RMS tracking error, and also RMS tracking error was reduced from nominal ZOH tracking error of 70.28µm to 56.63µm. Similarly to the SOH case in which the RMS tracking error was 84.43µm was reduced to 61.1µm after implementing the model input corrector.

Table 6-14: Hardware experimental results for delay and intermittent feedback

RMS tracking error, µm								
Delay (*ms*) | Intermittent feedback (cycles)	**ZOH_M**	**ZOH**	**FOH_M**	**FOH**	**SOH_M**	**SOH**	**TOH_M**	**TOH**
0	14.2	14.2	14.2	14.2	14.2	14.2	14.2	14.2
100 | 50	45.5	57.8	37.4	73.67	51.99	69.5	36.22	77.46
200|100	56.66	70.28	58.22	102.74	61.1	84.43	56.63	99.33

6.5 Combining the Intermittent Path Predictor via Interpolation with Model Input Corrector

Based on the simulation and hardware experiments results from section 6.4, it can be observed that the extrapolation method is not robust enough to perform the path prediction of the Modified Smith Predictor. Figure 6-18(A) shows the scenario when the feedback is delayed and intermittent, and Figure 6-18(B) shows predicted intermittent path using the extrapolation method based on the obtained positions from the sensor. It can be seen that the prediction error using extrapolation method can be large and unbounded because the extrapolation method is sensitive to the noise, disturbance and also the measured data.

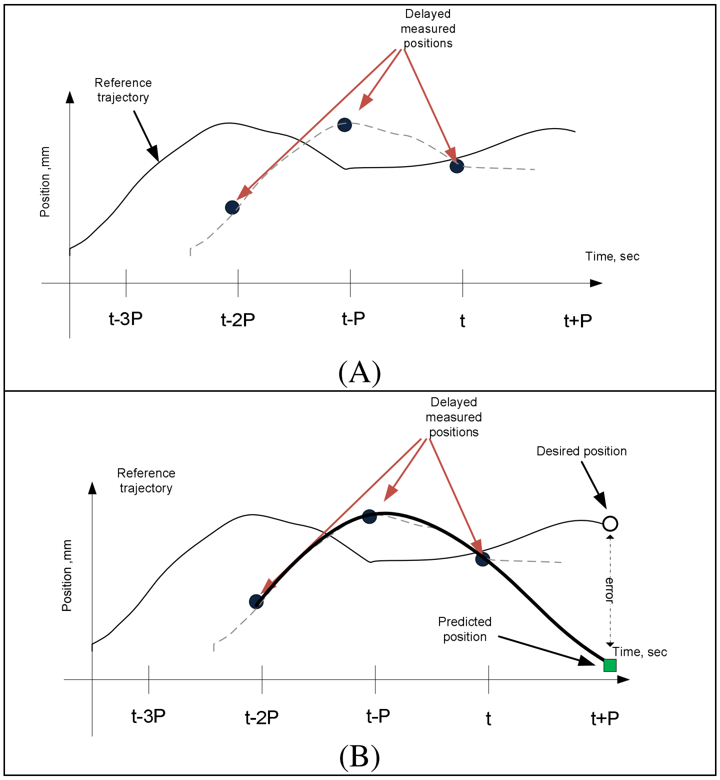


Figure 6-18: Extrapolation method. (A) shows the illustration when the feedback is delayed and intermittent, (B) shows the extrapolation method to predict the intermittent path using historical data only and it can be seen that the prediction error of the intermittent path is very large.

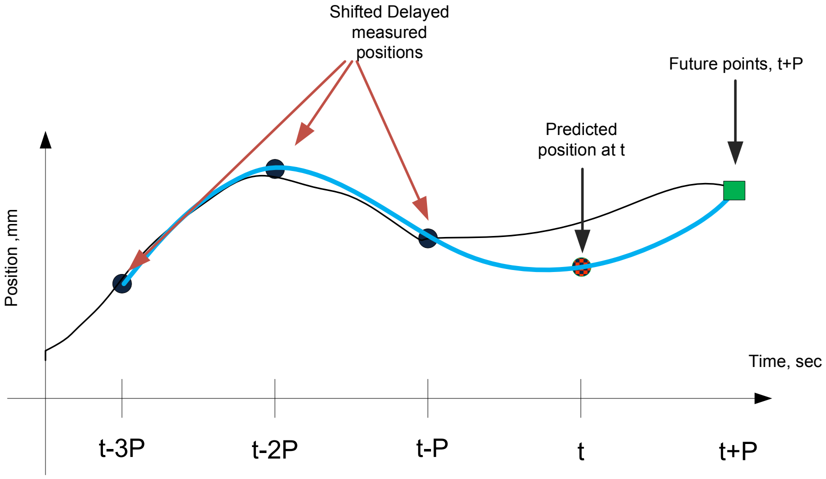


Figure 6-19: Intermittent path prediction. Interpolation path was predicted by using the delayed measured positions that were shifted backwards by image processing time, *P*, and one future point, *Y(t+P)* that was known ahead of time

Thus, an interpolation method based on the historical data and the future data was implemented to the Modified Smith Predictor to predict the intermittent path during the delayed and intermittent period. Since the *a priori* reference trajectory of system was known, the future point of the trajectory, *Y(t+P)* can also be used as one of the interpolation points of the algorithm. Therefore, all the delayed and intermittent data obtained by the sensor shown in Figure 6-18(A), were shifted backwards by the image processing time, *P* as shown in Figure 6-19. Then, interpolation using the shifted data, *Y(t-nP)* and the future point of the trajectory, *Y(t+P)* was performed not only to estimate the position at time *t* but to predict the intermittent path of the system from time *t* to *t+P* as seen in Figure 6-19. Three different interpolation methods were tested: first order, second order and third order. First order interpolation used only one obtained data point, *Y(t-P)* and the future point, *Y(t+P)* to obtain the intermittent path as shown in Eq.(6.4).

$$y(t) = A_1 t + B_1 \Big|_{t=t:t+P} \tag{6.4}$$

where

$$\begin{aligned} y(t+P) &= A_1(t+P) + B_1 \\ y(t-P) &= A_1(t-P) + B_1 \end{aligned}$$

Second order interpolation used two obtained positions, *Y(t-P)* and *Y(t-2P),* and the future point, *Y(t+P)* to obtain the intermittent path as shown in Eq.(6.5).

$$y(t) = A_2 t^2 + B_2 t + C_2 \Big|_{t=t:t+P} \tag{6.5}$$

where

$$\begin{aligned} y(t+P) &= A_2(t+P)^2 + B_2(t+P) + C_2 \\ y(t-P) &= A_2(t-P)^2 + B_2(t-P) + C_2 \\ y(t-2P) &= A_2(t-2P)^2 + B_2(t-2P) + C_2 \end{aligned}$$

Third order interpolation used three obtained positions, *Y(t-P) ,Y(t-2P)* and *Y(t-3P),* and the future point, *Y(t+P)* to obtained the intermittent path as shown in Eq.(6.6).

$$y(t) = A_3 t^3 + B_3 t^2 + C_3 t + D_3 \Big|_{t=t:t+P} \tag{6.6}$$

where

$$\begin{aligned} y(t+P) &= A_3(t+P)^3 + B_3(t+P)^2 + C_3(t+P) + D_3 \\ y(t-P) &= A_3(t-P)^3 + B_3(t-P)^2 + C_3(t-P) + D_3 \\ y(t-2P) &= A_3(t-2P)^3 + B_3(t-2P)^2 + C_3(t-2P) + D_3 \\ y(t-3P) &= A_3(t-3P)^3 + B_3(t-3P)^2 + C_3(t-3P) + D_3 \end{aligned}$$

Results of the interpolation approach and its comparison with the previously-described extrapolation approach are given in Table 6-15, and represented graphically in Figure 6-20.

Table 6-15: Hardware experiments results. This table shows the comparison of the path tracking performance of the Modified Smith Predictor when the interpolation method was used in the intermittent path prediction instead of the extrapolation method.

	RMS tracking error, µm							
			Extrapolation			Interpolation		
Delay (*ms*) | Intermittent feedback (cycles)	**ZOH**	**ZOH_M**	**FOH_M**	**SOH_M**	**TOH_M**	**FOH_M**	**SOH_M**	**TOH_M**
0	14.20	14.20	14.20	14.20	14.20	14.20	14.20	14.20
100 | 50	57.80	45.50	37.40	51.99	36.22	28.59	24.19	28.16
200 | 100	70.28	56.66	58.22	61.10	56.63	38.88	36.57	37.46

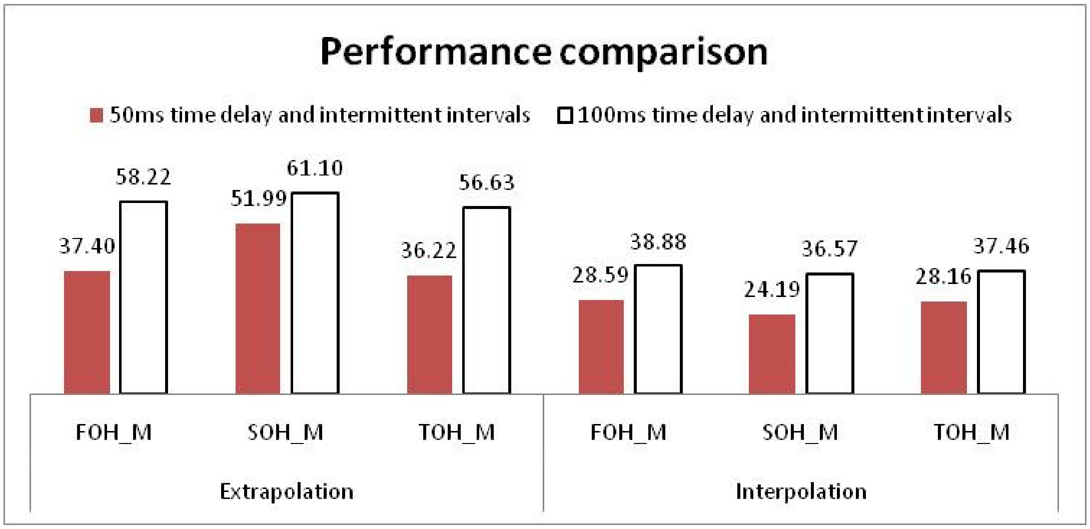


Figure 6-20: Performance comparison. The chart shows the hardware experimental results when the Modified Smith Predictor used the extrapolation method and interpolation method together with the model input corrector to perform path tracking of a 0.2 Hz sine wave reference trajectory

Hardware validation of Modified Smith Predictor with model input corrector and the intermittent path prediction was performed. Table 6-15 shows the comparison of the path tracking performance of the Modified Smith Predictor during the hardware experiments when the interpolation method was used in the intermittent path prediction algorithm instead of the extrapolation method. The results show that the interpolation method reduces the RMS tracking error by at least 20% as compared with the extrapolation method. For instance, when the system has 100ms time delay and intermittent intervals, the RMS tracking error was reduced by 53%, using the second order interpolation method. In addition, this Modified Smith Predictor was also tested in the prototype system to track reference trajectories different than a sine as presented in Figure 6-21 and Figure 6-22. The hardware experimental results showed that the path tracking performance of the Modified Smith Predictor works well not only with the test reference trajectory: 0.2Hz sine wave but also applicable to other reference trajectories.

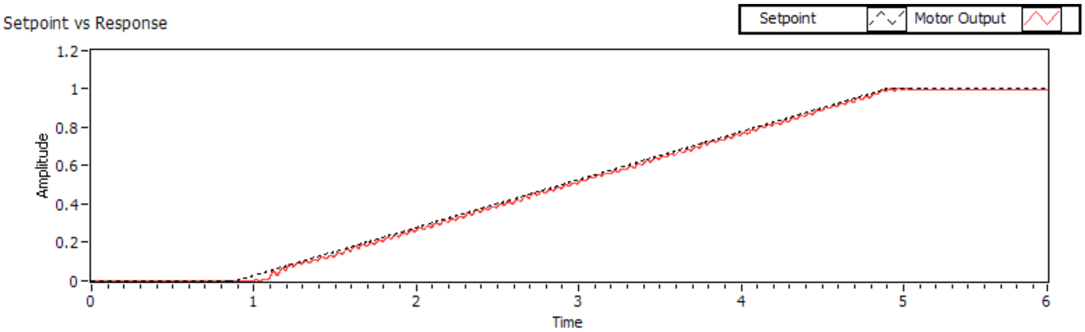


Figure 6-21: Hardware experiment for tracking a ramp-like waveform. The RMS tracking error was 10.5μm

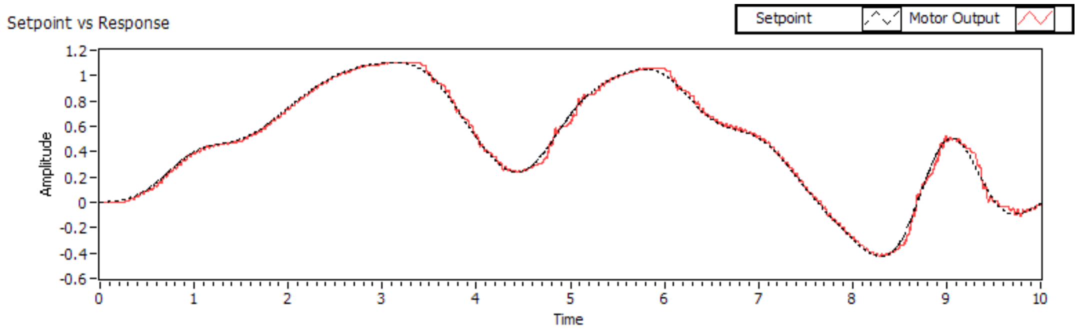


Figure 6-22: Hardware experiment for tracking a random waveform. The RMS tracking error was 26.68μm

Based on the experimental results, it can be concluded that the intermittent path prediction via interpolation together with the model input corrector not only improves the path tracking performance of the system, but also increases the robustness of the prediction algorithm.

6.6 Frequency Analysis Comparison

Frequency analysis of the system with different control architectures and feedback scenarios were performed: 1) Baseline system - PI controller with continuous feedback from encoders, and 2) Direct Position Sensing system using a Modified Smith Predictor with intermittent path prediction via interpolation together with a model input corrector, tested for the case when delayed and intermittent feedback occurs in the system.

Frequency analysis of both the baseline system and Direct Position Sensing system were performed to compare the dynamic response of both systems and resultant usable bandwidth. Bandwidth in this context is defined as the frequency range from 0 to the cut-off frequency f_c where the amplitude response dropped by -3dB. In order to obtain the output response of the system at varying frequencies, sine waves of 2mm peak-to-peak magnitude ranging from 0.01Hz to 10Hz were run to obtain the Bode plot for each system.

6.6.1 Classical System

Figure 6-23 and Figure 6-24 show the amplitude and phase Bode plots of the baseline system; the bandwidth of this system was found to be 4.5Hz.

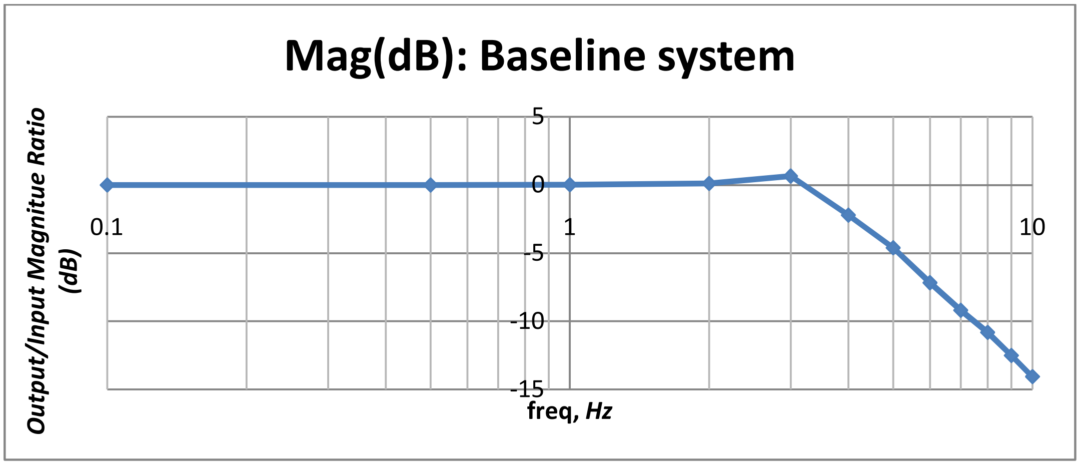


Figure 6-23:Bode plot Mag. Bode magnitude ratio plot for the classical system with continuous feedback. The bandwidth of the system is estimated around 4.5Hz based on the plot.

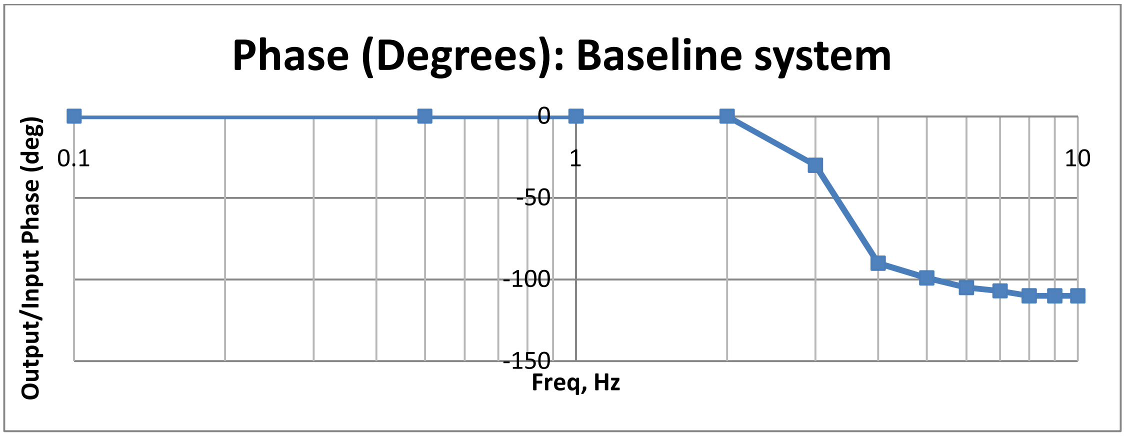


Figure 6-24: Bode plot Phase. Bode phase plot for the classical system with continuous feedback; the phase shift begins at approximately 3Hz.

6.6.2 Direct Position Sensing

Frequency analysis of the Modified Smith Predictor with model input corrector and intermittent path prediction using interpolation was performed on the prototype to obtain the bandwidth of the system under test. In addition to the 2-mm-amplitude input reference, a 100-ms time delay and 100-ms intermittent interval was included in the system. As the loop closure time or the control is 2ms, this intermittent behavior translates to an actual feedback data point of 100-ms delayed data every 50th control cycle. Figure 6-25 and Figure 6-26 show the Bode plots of the system; bandwidth is around 4.5Hz, equivalent to the bandwidth of the baseline system.

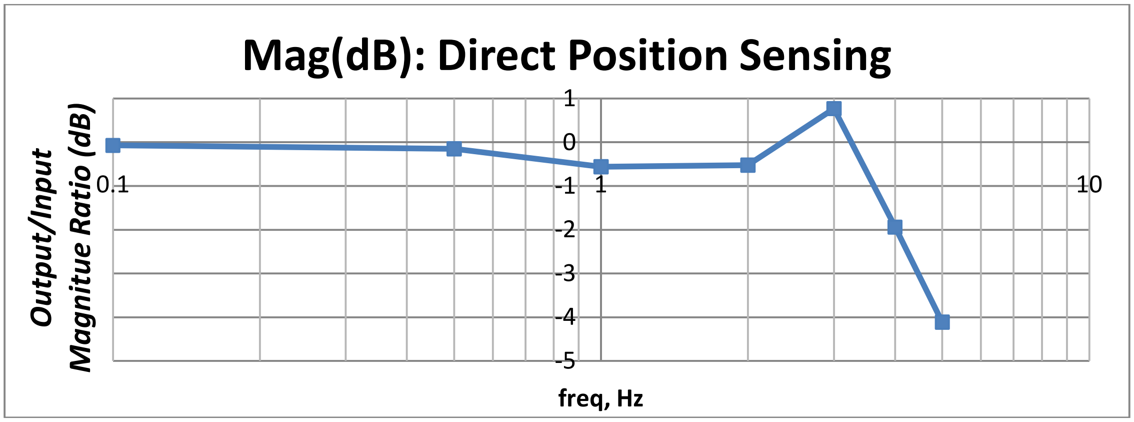


Figure 6-25: Bode plot Mag. Bode plot for the Direct Position Sensing System that has 100ms delay and 50-cycle intermittent interval feedback, using intermittent path prediction via interpolation method and model input corrector structure. The bandwidth of this system is 4.5Hz.

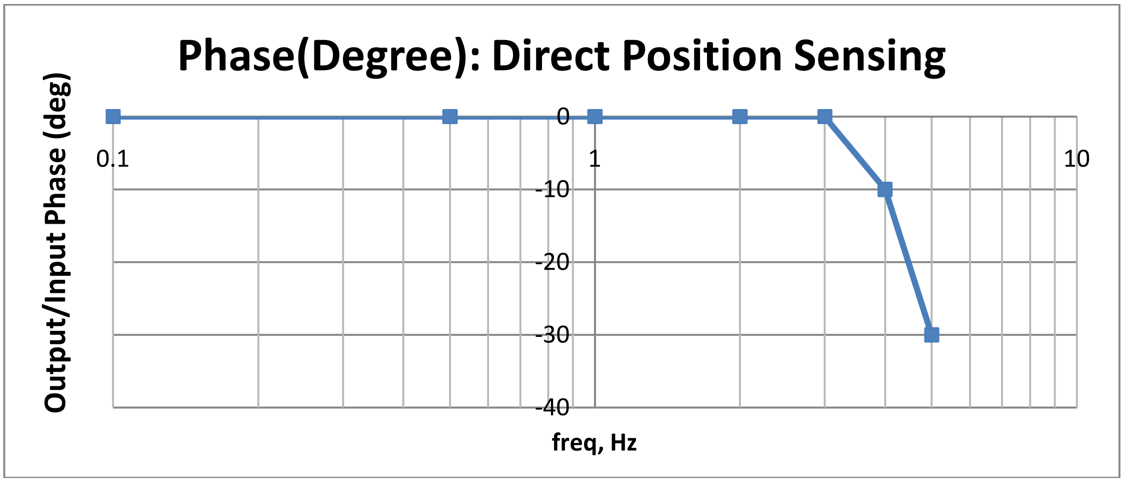


Figure 6-26: Bode plot Phase. Bode plot for the Direct Position Sensing System with 100ms delay and 50-cycle intermittent interval feedback using intermittent path prediction via interpolation method and model input corrector structure.

6.6.3 Recommendation

A primary assumption in this analysis is that the time delay is equal to the intermittent interval, which is also equivalent to the processing time of the image processing algorithm. Although the processing time of the image processor is not always equivalent to the intermittent interval, this research assumed both the processing time and the intermittent interval to simplify the feedback scenario of the system during the system integration process of the Modified Smith Predictor control architecture and the vision sensors. However, when the actual time delay and intermittent interval of a system is known, those values can be used to tune the Modified Smith Predictor to compensate the delayed and intermittent feedback.

Based on the test run on the hardware, the baseline system with continuous feedback has a bandwidth of 4.5Hz. The Direct Position Sensing System using the intermittent path prediction via interpolation method and the model input corrector with the emulation of 100ms image processing time and 50-cycle intermittent feedback also has a bandwidth of 4.5Hz. Similar testing was performed at a 200-ms delay value on the prototype, and the system found to be significantly degraded especially the path tracking performance of the system in which the intermittent path is too long, causing the path prediction error to increase. Therefore, it is recommended that the image processing time of the vision sensor should not be longer than 100ms in order to match the baseline system response.

In addition, the aliasing effect is another factor that needs to be considered in this research due to the long cycles of intermittent feedback which significantly decrease the system sampling frequency. For the case under study in this analysis, when the system has 100ms intermittent interval, the equivalent sampling frequency is 10Hz. According to

Shannon's theorem, the maximum frequency to be resolved, f_i cannot exceed ½ of the sampling frequency, f_s as presented in Eq.(6.7).

$$f_i \leq \frac{1}{2} f_s \tag{6.7}$$

If this criterion is violated, the sampling system cannot resolve the higher frequencies, and false signals will be introduced, a phenomenon known as *aliasing*. Figure 6-27 illustrates a hardware test of the intermittent system sampling at 10Hz, and trying to resolve a 7Hz signal. This violates Shannon's theorem, and aliasing occurs in the system, notably the beat frequency observed across multiple cycles.

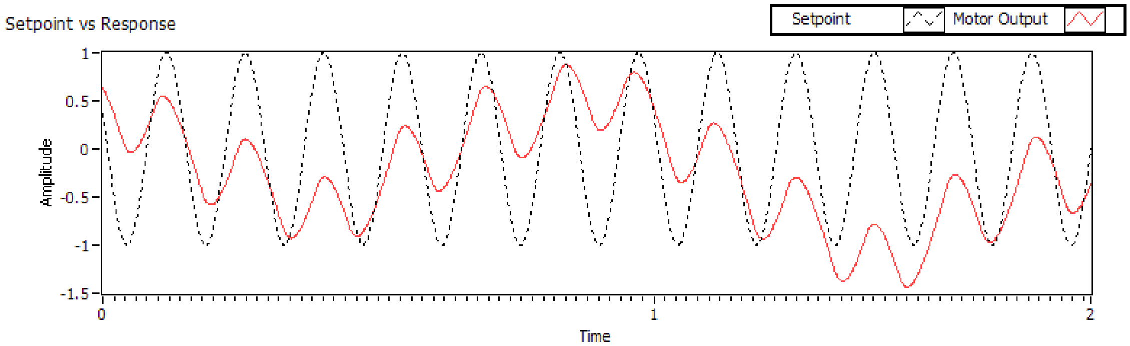


Figure 6-27: Aliasing effect when the input signal of the system is larger than 5 Hz. The input signal of this test is 7Hz, and the restricted sampling frequency cannot resolve the input, resulting in a false signal.

CHAPTER SEVEN

CONCLUSION

7.1 Conclusion

This dissertation presents a new type position sensor: Direct Position Sensing that can actively monitor the planar position of a device using computer vision technology Instead of using a set of conventional position sensor, Direct Position Sensing uses computer vision technology: a digital camera and a digital display screen, to track the planar position of the system. The objective of Direct Position Sensing is to be able to perform planar motion control of the system by actively tracking the display target on the digital display screen. By doing so, the associated machine errors of the device will not affect the path tracking of the system. In addition, Direct Position Sensing also eliminates the need of the kinematic model to estimate the planar position as Direct Position Sensing is capable to actively track the actual planar position based on the display target on the digital screen. Since Direct Position Sensing is actively monitoring the planar position of the system and not relying on the kinematic model, the error mapping and error compensation process can be eliminated. Therefore, the production cost of a part also can be reduced because the machined part is more accurate, which reduces scrap attributable to machine error and improves overall product quality, reducing scrap and rework.

In addition, this dissertation also shows a unique system integration process to integrate a motion controller with a slow sensor system. Although the feedback latency of using a computer vision system is a well known issue, the feedback intermittent behavior that exists in this system posed a new research challenge. Intermittent feedback actually occurs in many applications, but typically in shorter intermittent cycles. This research investigated the system that has long intermittent cycles, which causes fundamental control issues using traditional methods. This dissertation pioneers new approaches in control of long intermittent feedback systems with significant time delay behavior, and also evaluates the system integration approach to bridge a system that not only has delay feedback but also intermittent feedback to a motion controller using model based approach.

Based on the trade-off analysis of the factor that affects the controllability and the accuracy of Direct Position Sensing is presented in Figure 7-1. If the high resolution is desired in the system, then the number of images that need to be processed increases which also increases the image processing time and also the accuracy of the tracking system. However, it affects the controllability of the system because when the image processing time of the system increases, and the time delay and intermittent cycles of the system also increases. As seen in this dissertation, when the time delay and intermittent cycles increases, the controllability of the system decreases, detrimentally affecting tracking capability of the system. However, controllability of the system can still be achieved by finding the best balance between the controller frequency and the vision sensor frequency.

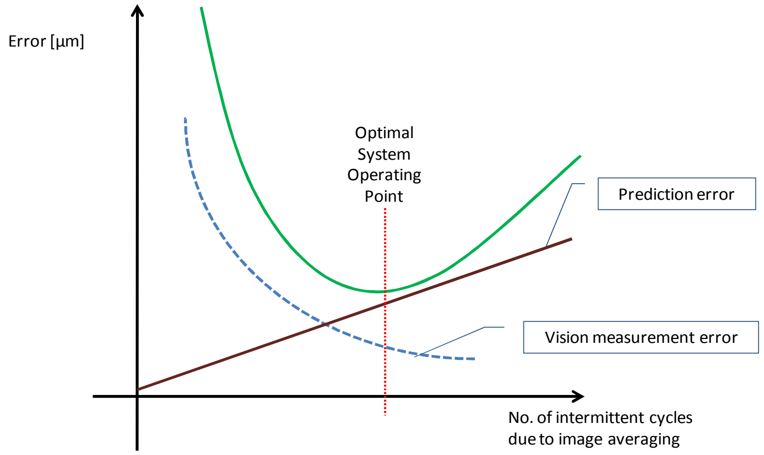


Figure 7-1: Trade-off analysis. This diagram shows the trade-off analysis between the system resolution, accuracy image processing time and the controllability of the system

7.2 Contributions

Below are the academic contributions as a result of this research:

1. Design of a novel active tracking position sensing system
    a. planar position of a device can be directly measured accurately without using a kinematic model
    b. eliminate the needs for error mapping and compensation that improve accuracy and are robust to time-dependent system changes

2. Integrated high-rate motion control system with low-rate vision acquisition and processing system.
    a. Design of a new type of feedback control architecture that controls a motion system with both time delay and long intermittent cycle behavior using a model based control algorithm
    b. Development of a control algorithm to address the delay and long intermittent cycles by using model based control in an augmented Smith Predictor architecture.
    c. Creation of an automated model residual corrector algorithm: Model input corrector, emulating a feedback controller to correct the plant model during the motion control process.
    d. Development of an estimation algorithm based on the historical data and *a priori* data to minimize the model residual of the model based control during the intermittent period , enhancing the tracking performance of the Direct Position System
    e. Investigation of the types of possible model-based approaches to address this new feedback control system and also provide the feedback of the findings
    f. Recommendation of the image processing time so that the system response of the Direct Position Sensing system performance is compatible with the baseline system with continuous feedback

7.3 Future Work

Most model-based control systems use only one fixed model in the controller that is obtained via offline system identification method. Thus, during the continuous operation of a system, the plant model might be varying with time which leads to the increment of model discrepancy, which directly affects the performance of the controller (See Appendix F). Thus, research related to investigate real time system identification either periodically or continuously to update the model for a model-based controller can be performed to enhance the controller performance[67].

In addition, this dissertation only uses a linear model to represent a servo motor. Hence, the plant model can be extended to incorporate other involved factors such as backlash and friction to further improve the model accuracy of the plant. As a first prototype of Direct Positing Sensing, the camera, digital screen and the embedded micro controller was limited in terms of processing power and capturing speed. As seen in the simulation and experiment results, the RMS position error of the tracking system increases with the time delay and intermittent cycles which is caused by the processing time of the computer vision system. Thus, further testing can be pursued to decrease the image processing time by improving the algorithm and also faster processor so that the delay and intermittent cycles of the system can be minimized.

REFERENCES

1. Wadkin, *Pattern Milling Machine*: London, England.
2. Machining, P., *Model 83-1000*. Dec 2000.
3. Okuma, *M 400 Milling Machine*.
4. Wong, C., et al., *A New Position Feedback Method for Manufacturing Equipment*, in *Proceedings of the ASME International Manufacturing Science and Engineering Conference 2008*. 2008 Evanston, IL. p. 111-120.
5. Suh, S.-H., S.-K. Kang, and D.-H. Chung, *Theory and Design of CNC Systems*. 2008.
6. Macpower, *GX-100 CNC lathe* 2010.
7. Ogata, K., *Modern Control Engineering*. 2001: Prentice Hall PTR.
8. MacCleery, B., *Adaptive PID Tuning with LabVIEW 8.6*. 2008, National Instruments.
9. Jedrzejewski, J. and W. Modrzycki, *Intelligent supervision of thermal deformations in high precision machine tools*, in *Proc. 32nd Int. MATADOR Conf.* 1997: Manchester, UK,.
10. Ramesh, R., M.A. Mannan, and A.N. Poo, *Error compensation in machine tools -- a review: Part I: geometric, cutting-force induced and fixture-dependent errors.* International Journal of Machine Tools and Manufacture, 2000. 40(9): p. 1235-1256.
11. Morris, T.J., *The REAL accuracy of machine tools, Proc. 3rd Int. Conf. on Laser Metrology and Machine Performance.* LAMDAMAP, 1997: p. T.J.113-122.
12. Evans, c.j. and R.J. Kocken, *Self-calibration: reversal, redundancy, error separation and absolute testing.* Annals of the CIRP 45, 1996.
13. Tlusty, J., *Manufacturing processes and equipment*. 2000, Upper Saddle River, NJ: Prentice-Hall. xxiii, 928 p.
14. Ni Jun, *CNC machine accuracy enhancement through real time error compensation.* Transactions of the ASME. Journal of Manufacturing Science and Engineering, 1997. 119(4B): p. 717-25.
15. Zhu, J., *Robust thermal error modeling and compensation for CNC machines.* Ph.D Thesis, 2008.
16. Bryan, J., *Interaction status of thermal error research.* Annals of the CIRP, 1990: p. 645-656.
17. Chen, J.S., et al., *Thermal error modeling for volumetric error compensation.* ASME PROD ENG DIV PUBL PED, 1992.

18. Xie, C., et al., *The Analysis and Research about Temperature and Thermal Error Measurement Technology of CNC Machine Tool.* Manufacturing Automation Technology, 2008.

19. Heidenhain_Corporation. *208_960-24*. 2003 [cited; Available from: http://filebase.heidenhain.de/doku/brochures/pdf/208_960-24.pdf.

20. Liang, J.C., et al., *A comprehensive error compensation system for correcting geometric, thermal, and cutting force-induced errors.* The International Journal of Advanced Manufacturing Technology, 1997. 13(10): p. 708-712.

21. De Meter, E.C. and M.J. Hockenberger, *The application of tool path compensation for the reduction of clamping induced geometric errors.* International Journal of Production Research, 1997. 35(12): p. 3415 - 3432.

22. Hockenberger, M.J. and E.C. De Meter, *The Application of Meta Functions to the Quasi-Static Analysis of Workpiece Displacement Within a Machining Fixture.* Journal of Manufacturing Science and Engineering, 1996. 118(3): p. 325-331.

23. Ramesh, R., M.A. Mannan, and A.N. Poo. *Development of an in-situ machine tool axis positioning accuracy measurement device*. 2004. Charlotte, NC, United States: Society of Manufacturing Engineers, Dearborn, 48121-0930, United States.

24. Kim, S.C., O.H. Kim, and E.S. Lee. *A study of on-machine measurement for corrective manufacturing using a calibrated machining center*. 2002. Hefei, China: Harbin Institute of Technology.

25. Kobayashi, R., et al., *Development and evaluation of a non-contact on-machine profile measurement system using a compact laser probe.* Key Engineering Materials, 2008. 381-382(1): p. 187-190.

26. Ni, J. and S.M. Wu, *An On-Line Measurement Technique for Machine Volumetric Error Compensation.* Journal of Engineering for Industry, 1993(115): p. 85–92.

27. Yuan, J. and J. Ni, *The real-time error compensation technique for CNC machining systems.* Mechatronics, 1998. 8(4): p. 359-380.

28. Lo, C.H., *Real-time error compensation on machine tools through optimal thermal error modeling*, in *PhD dissertation*. University of Michigan: University of Michigan.

29. Renishaw, *The benefits of remote interferometers in laser measurement.*

30. Sartori, S. and G.X. Zhang, *Geometric Error Measurement and Compensation of Machines.* CIRP Annals - Manufacturing Technology, 1995. 44(2): p. 599-609.

31. Zhang, G., et al. *Error compensation of Coordinate measurement machine S.* 1985. Palermo, Italy: Technische Rundschau, Berne, Switz.

32. Donmez, M.A., Blomquist, D. S., Hocken, R. J., Liu, C. R. ,Barash, M. M., *A general methodology for machine tool accuracy enhancement by error compensation.* Precision Engineering, 1986. 8(4): p. 187-196.

33. Duffie, N.A. and S.J. Malmberg, *Error Diagnosis and Compensation Using Kinematic Models and Position Error Data.* CIRP Annals - Manufacturing Technology, 1987. 36(1): p. 355-358.

34. Balsamo, A., D. Marques, and S. Sartori, *A Method for Thermal-Deformation Corrections of CMMs.* CIRP Annals - Manufacturing Technology, 1990. 39(1): p. 557-560.

35. Mou, J. and C.R. Liu, *A method for enhancing the accuracy of CNC machine tools for on-machine inspection.* Journal of Manufacturing Systems, 1992. 11(4): p. 229-37.

36. Weck, M. and U. Herbst. *Compensation of thermal errors in machine tools with a minimum number of temperature probes based on neural networks*. 1998. Fairfield, NJ, USA: ASME.

37. Weekers, W.G. and P.H.J. Schellekens, *Compensation for dynamic errors of coordinate measuring machines.* Measurement: Journal of the International Measurement Confederation, 1997. 20(3): p. 197-209.

38. Mu, Y.H. and B.K.A. Ngoi, *Dynamic error compensation of coordinate measuring machines for high-speed measurement.* International Journal of Advanced Manufacturing Technology, 1999. 15(11): p. 810-814.

39. Janeczko, J., B. Griffin, and C. Wang, *Laser vector measurement technique for the determination and compensation of volumetric position errors. Part II: Experimental verification.* Review of Scientific Instruments, 2000. 71(10): p. 3938-41.

40. Chapman, M.A.V., *Limitations of laser diagonal measurements.* Precision Engineering, 2003. 27(4): p. 401-406.

41. Svoboda, O., *Testing the diagonal measuring technique.* Precision Engineering, 2006. 30(2): p. 132-44.

42. Donaldson, R.R., D.C. Thompson, and E.G. Loewen, *Design and Performance of a Small Precision CNC Turning Machine.* CIRP Annals - Manufacturing Technology, 1986. 35(1): p. 373-376.

43. Veldhuis, S.C. and M.A. Elbestawi, *A Strategy for the Compensation of Errors in Five-Axis Machining.* CIRP Annals - Manufacturing Technology, 1995. 44(1): p. 373-377.

44. Yang, J., J. Yuan, and J. Ni, *Thermal error mode analysis and robust modeling for error compensation on a CNC turning center.* International Journal of Machine Tools and Manufacture, 1999. 39(9): p. 1367-1381.

45. Srinivasa, N. and J.C. Ziegert, *Prediction of Thermally Induced Time-Variant Machine Tool Error Maps Using a Fuzzy Artmap Neural Network.* Journal of Manufacturing Science and Engineering, 1997. 119(4A): p. 623-630.

46. Narayan Srinivasa, J.C.Z., *Application of fuzzy artmap neural network to real-time learning and prediction of time-variant machine tool error maps*, in *IEEE International Conference on Fuzzy Systems,*. 1994,.

47. Chen, J.S. and G. Chiou, *Quick testing and modeling of thermally-induced errors of CNC machine tools.* International Journal of Machine Tools & Manufacture, 1995. 35(7): p. 1063-1074.

48. Ramesh, R., M.A. Mannan, and A.N. Poo, *Error compensation in machine tools -- a review: Part II: thermal errors.* International Journal of Machine Tools and Manufacture, 2000. 40(9): p. 1257-1284.

49. Yang, H. and J. Ni, *Dynamic modeling for machine tool thermal error compensation.* Journal of Manufacturing Science and Engineering, Transactions of the ASME, 2003. 125(2): p. 245-254.

50. Brosilow, C. and B. Joseph, *Techniques of Model Based Control with Cdrom.* 2001.

51. Ljung, L., *System Identification: Theory for the User* 1999: Prentice Hall.

52. Rake, H., *Step response and frequency response methods.* Automatica, 1980. 16(5): p. 519-526.

53. Wellstead, P.E., *Non-parametric methods of system identification.* Automatica, 1981. 17(1): p. 55-69.

54. Brown, C., *Gaze controls with interactions and delays.* IEEE Transactions on Systems, Man and Cybernetics, 1990. 20(2): p. 518-527.

55. Watanabe, K. and M. Ito, *A process-model control for linear systems with delay.* IEEE Transactions on Automatic Control, 1981. AC-26(6): p. 1261-9.

56. Astrom, K.J., C.C. Hang, and B.C. Lim, *A new Smith predictor for controlling a process with an integrator and long dead-time.* IEEE Transactions on Automatic Control, 1994. 39(2): p. 343-5.

57. Matausek, M.R. and A.D. Micic, *On the modified Smith predictor for controlling a process with an integrator and long dead-time.* IEEE Transactions on Automatic Control, 1999. 44(8): p. 1603-6.

58. Tian, Y.-C. and F. Gao, *Compensation of Dominant and Variable Delay in Process Systems.* Industrial & Engineering Chemistry Research, 1998. 37(3): p. 982-986.

59. Corke, P.I. *High-performance visual servoing for robot end-point control*. 1993. USA.

60. Shirai, Y. and H. Inoue, *Guiding a robot by visual feedback in assembling tasks.* Pattern Recognition, 1973. 5(2): p. 99-106, IN3, 107-108.

61. Corke, P.I. and S.A. Hutchinson. *Real-time vision, tracking and control*. 2000. San Francisco, CA, USA: Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, USA.

62. Weiss, L.E., A.C. Sanderson, and C.P. Neuman. *Dynamic visual servo control of robots: An adaptive image-based approach*. in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*. 1985.

63. Corke, P.I. *High-performance visual servoing for robot end-point control*. in *Intelligent Robots and Computer Vision XII: Active Vision and 3D Methods*. 1993. Boston, MA, USA: SPIE.

64. Sim, T.P., G.S. Hong, and K.B. Lim, *Multirate predictor control scheme for visual servo control.* IEE Proceedings: Control Theory and Applications, 2002. 149(2): p. 117-124.

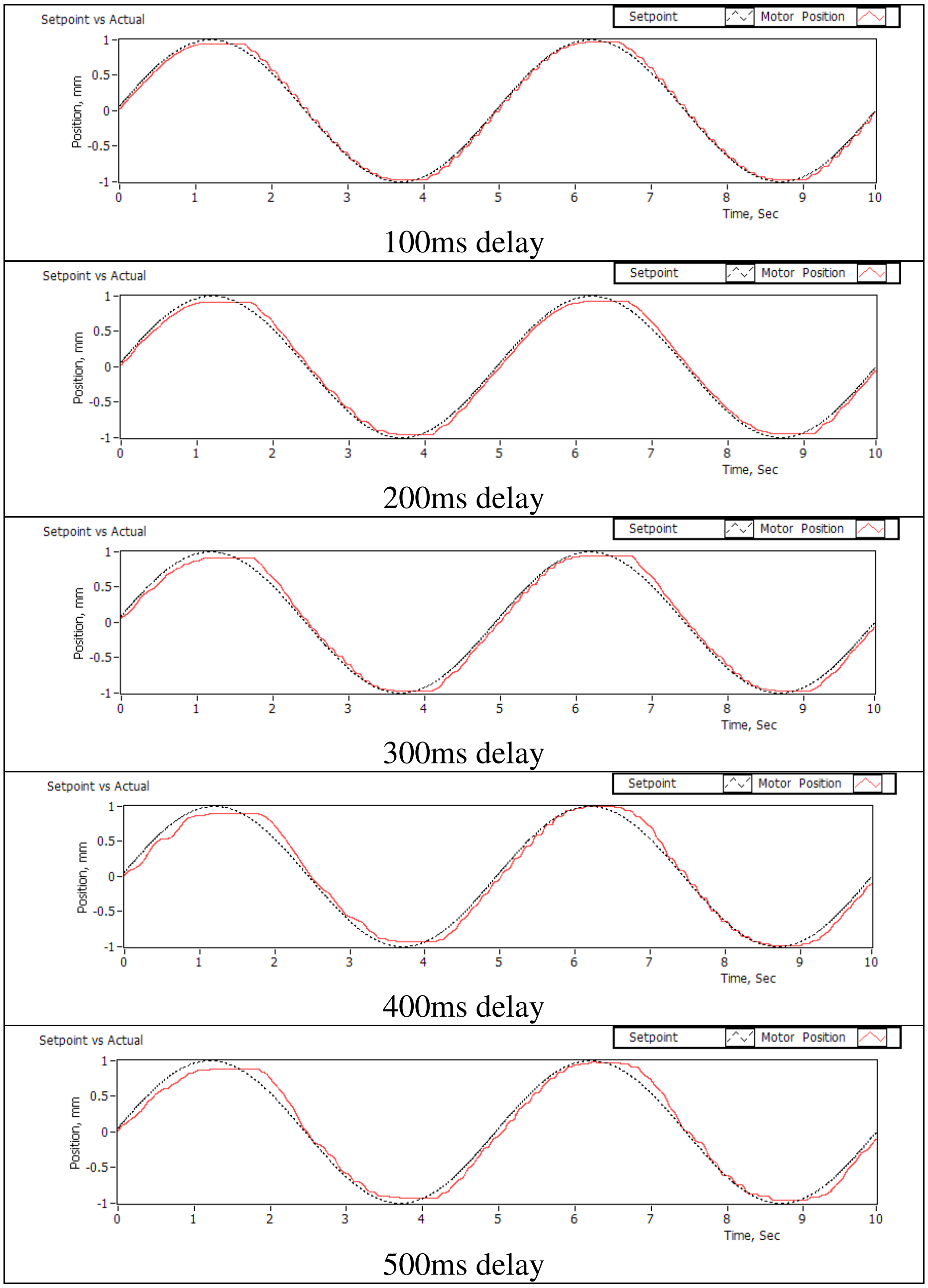
65. Xie, H., et al. *Visual servoing with modified Smith predictor for micromanipulation tasks*. 2005. Niagara Falls, Ont., Canada: IEEE.

66. Xiangjin, Z., H. Xinhan, and W. Min. *Visual servoing based on fuzzy adaptive PID with Modified Smith Predictor for micromanipulation*. in *Advanced Intelligent Mechatronics, 2008. AIM 2008. IEEE/ASME International Conference on*. 2008.

67. Wong, C.Y., L. Mears, and J. Ziegert, *Model Based Control to Enhance a Novel Visual Control Position System* in *International Conference on Control, Automation and Systems 2010*. 2010: Gyeonggi-do, Korea.

APPENDICES

APPENDIX A: Hardware Experiment Data for Smith Predictor



100ms delay

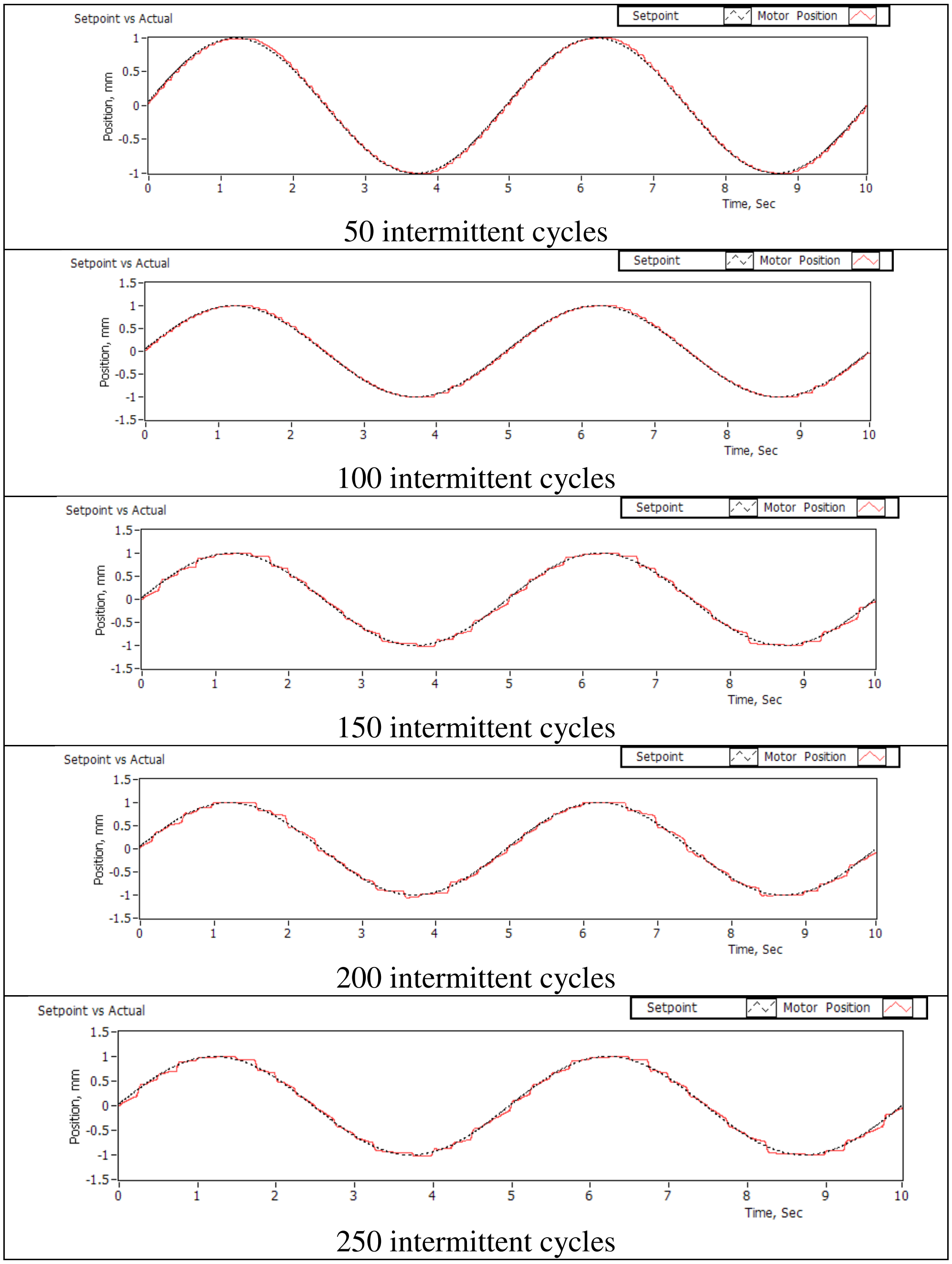
200ms delay

300ms delay

400ms delay

500ms delay

Hardware experiments results for delay feedback



50 intermittent cycles

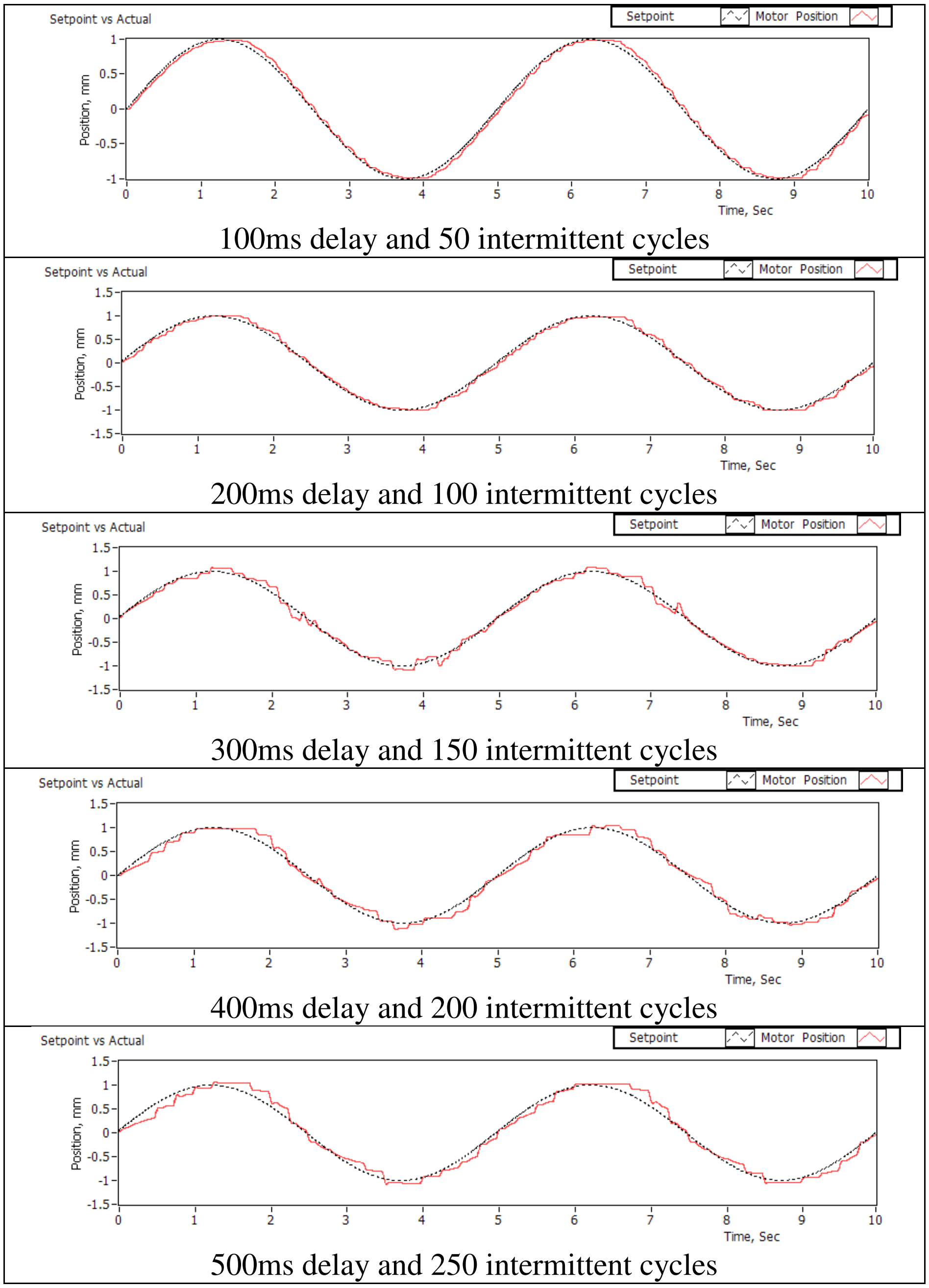
100 intermittent cycles

150 intermittent cycles

200 intermittent cycles

250 intermittent cycles

Hardware experiments results for intermittent feedback



100ms delay and 50 intermittent cycles

200ms delay and 100 intermittent cycles

300ms delay and 150 intermittent cycles

400ms delay and 200 intermittent cycles

500ms delay and 250 intermittent cycles

Hardware experiments results for delay and intermittent feedback

APPENDIX B: Stability Analysis for Model Input Corrector

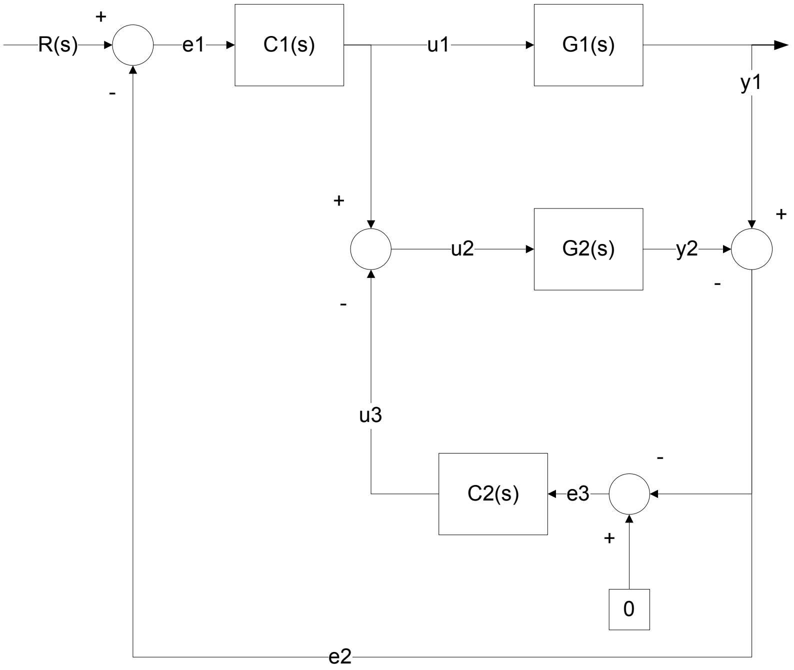


Figure B-1: Simplified block diagram from Modified Smith Predictor with model input corrector

Variables	**Description**
C_1	PI controller
C_2	P controller for Model input corrector
G_1	Plant
G_2	Plant's model

The goal of the stability analysis is to observe the stability of the Modified Smith Predictor when another controller C_2 was integrated to the system as shown in Figure B-1 and also obtain the range of the K_{p2} gain of C_2 before the system become unstable. The plant, $G_1$and the plant model, G_2 in the continuous domain as shown in Eq. (A.1) and (A.2) were used in the simulation. C_1 and C_2 show the transfer function of the PI controller and the P-controller used in the model input corrector, as shown in Eq.(A.3) and (A.4) respectively.

$$G_1 = \frac{0.000275s^2 + 0.00038255s + 0.002}{s^2 + 0.0455s + 1.048e - 7} \tag{A.1}$$

$$G_2 = \frac{0.0003s^2 + 0.000361s + 0.002}{s^2 + 0.0455s + 1.048e - 7} \tag{A.2}$$

$$C_1 = \frac{10s + 1000}{s} \tag{A.3}$$

$$C_2 = K_{p2} \tag{A.4}$$

The system block diagram of Figure B-1 was reduced to a single transfer function shown in Eq. (A.5) and the derivation of the reduced transfer function is shown below:

$$
\begin{aligned}
u_1 &= e_1 \times C_1 \\
&= (R - e_2)C_1 \\
&= (R - (y_1 - y_2))C_1 \\
&= R - u_1C_1G_1 + y_2G_2 \\
&= R - u_1C_1G_1 + u_2C_1G_2 \\
&= R - u_1C_1G_1 + (u_1 - u_3)C_1G_2 \\
&= R - u_1C_1G_1 + u_1C_1G_2 - e_3G_2C_1C_2 \\
&\because e_3 = 0 - e_2 \\
u_1 &= R - u_1C_1G_1 + u_1C_1G_2 + e_2G_2C_1C_2 \\
&= R - u_1C_1G_1 + u_1C_1G_2 + \frac{u_1 - Rc_1}{c_1}G_2C_1C_2 \\
&= R - u_1C_1G_1 + u_1C_1G_2 + u_1G_2C_2 - RC_1C_2G_2 \\
&= R(1 - C_1C_2G_2) - u_1(C_1G_1 - C_1G_2 - G_2C_2)
\end{aligned}
$$

$$
\begin{aligned}
u_1(1 + C_1G_1 - C_1G_2 - G_2C_2) &= R(1 - C_1C_2G_2) \\
\frac{u_1}{R} &= \frac{(1 - C_1C_2G_2)}{(1 + C_1G_1 - C_1G_2 - G_2C_2)}
\end{aligned}
$$

$$\frac{y_1}{G_1} = u_1$$

$$\frac{y_1}{R} = \frac{G_1(1 + C_1C_2G_2)}{1 + G_1C_1 - G_2C_1 - C_2G_2} \tag{A.5}$$

The Proportional and Integral gains of C_1 were configured to be constant throughout this simulation, in which K_p=10, and K_i=1000. These gains were the same gains used in the simulation and hardware experiment of the Modified Smith Predictor without the model input corrector that was tuned by trial and error. Thus, in order to obtain the range of K_{p2} of C_2 before the system becomes unstable, the eigenvalues of the system based on Eq.(A.5) in term of K_{p2} were computed as shown at Eq.(A.5)

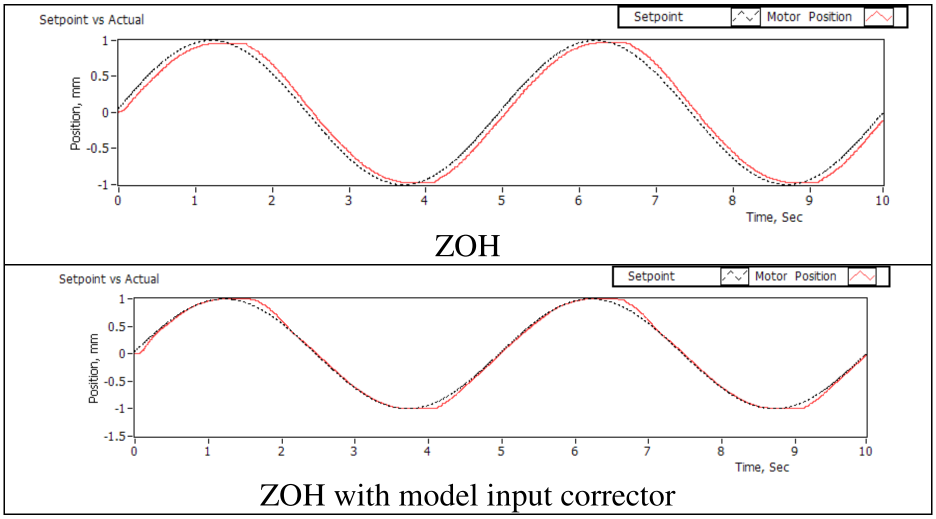
$$\begin{array}{l}
2.200802000 \cdot 10^{-17} K_{p2} - 2.371375698 \cdot 10^{-16} \\
1.911278397 \cdot 10^{-11} K_{p2} - 2.059413255 \cdot 10^{-10} \\
0.000004150028051 K_{p2} - 0.4471683564\text{e-}4_{p2} \\
0.0001829490824 K_{p2} - 0.001985562654 \\
0.002033530354 K_{p2} - 0.02460512800 \\
0.000389241 K_{p2} - .1018081150 \\
0.00031 K_{p2} - 0.9996500002
\end{array} \tag{A.5}$$

Based on the control theory, the system is considered unstable, if the eigenvalues of the system are positive. Thus, by solving all the eigenvalues obtained in Eq. (A.5) equal to zero, Eq. (A.6) shows all the K_{p2} values of the calculated results. Thus, it can be concluded that the system is stable as long as K_{p2} is smaller than 10.775

$$\begin{array}{l}
K_{p2} = 10.77505245 \\
K_{p2} = 10.77505641 \\
K_{p2} = 12.09971022 \\
K_{p2} = 10.77506829 \\
K_{p2} = 10.85308889 \\
K_{p2} = 12.09971022 \\
K_{p2} = 261.5554759 \\
K_{p2} = 3224.677420
\end{array} \tag{A.6}$$

APPENDIX C: Hardware Experiments Data for Modified Smith Predictor

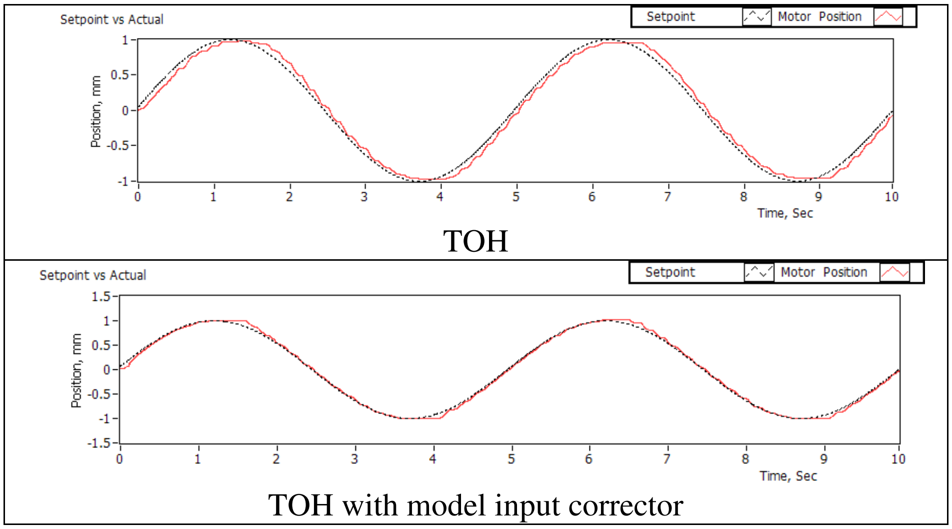
Hardware experiment data for intermittent path prediction together with model input corrector



ZOH

ZOH with model input corrector

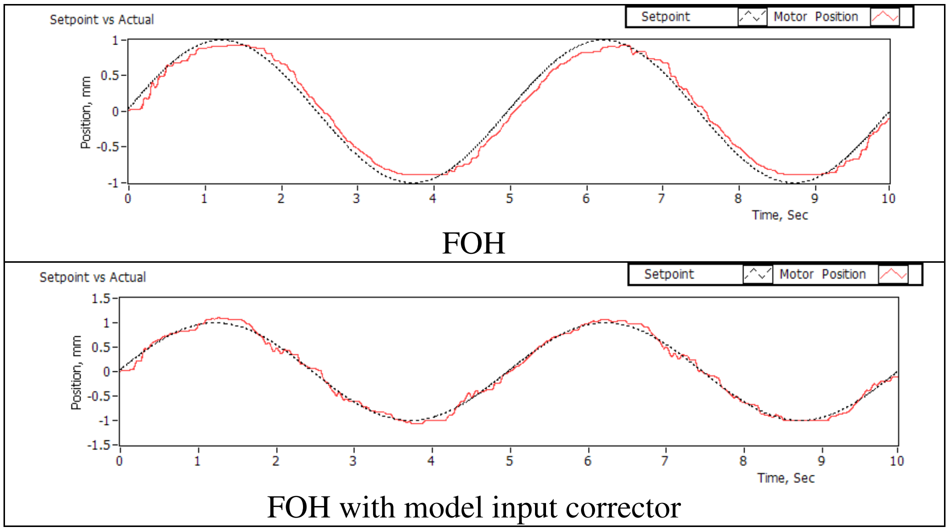
Delay 100ms and 50 intermittent cycles for the ZOH case



TOH

TOH with model input corrector

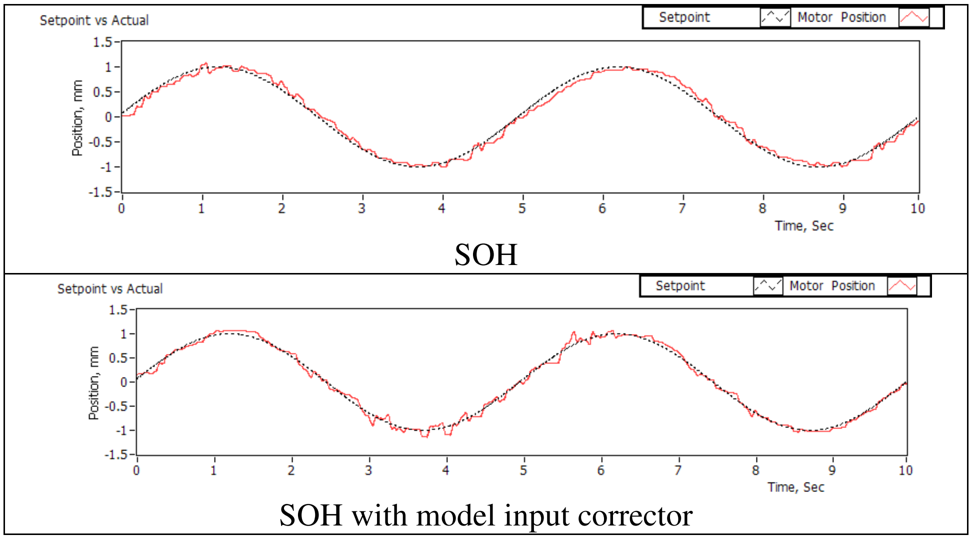
Delay 100ms and 50 intermittent cycles for the TOH case



FOH

FOH with model input corrector

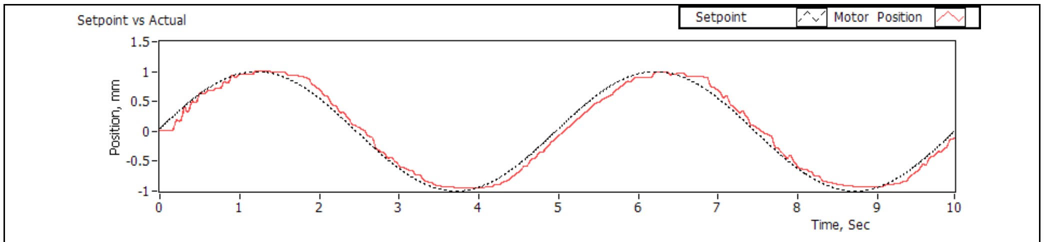
Delay 200ms and 100 intermittent cycles for the FOH case



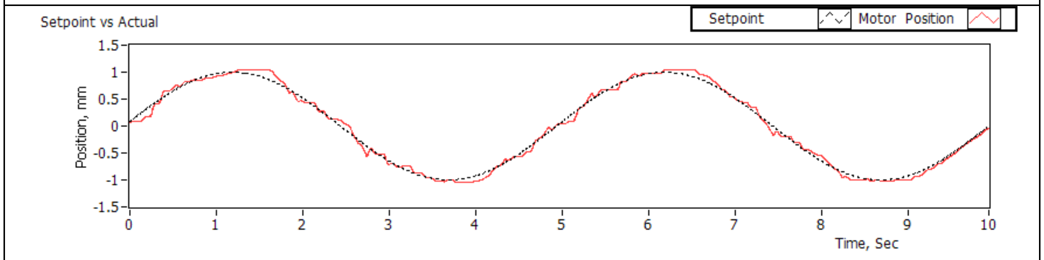
SOH

SOH with model input corrector

Delay 200ms and 100 intermittent cycles for the SOH case



TOH

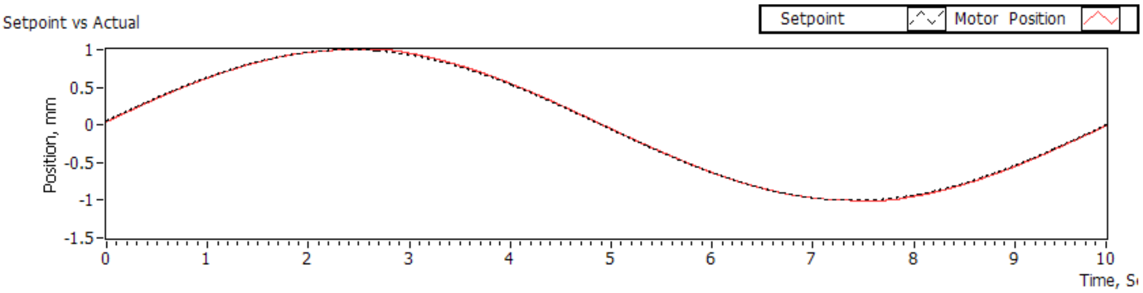


TOH with model input corrector

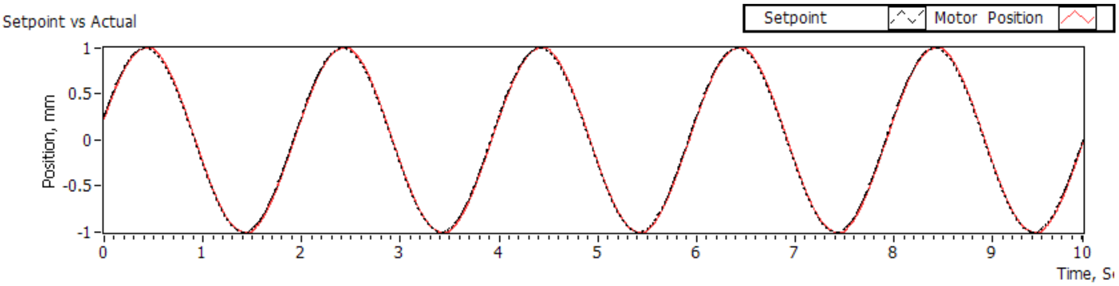
Delay 200ms and 100 intermittent cycles for the TOH case

APPENDIX D: Waveform of Baseline System with Continuous Feedback during Frequency Analysis

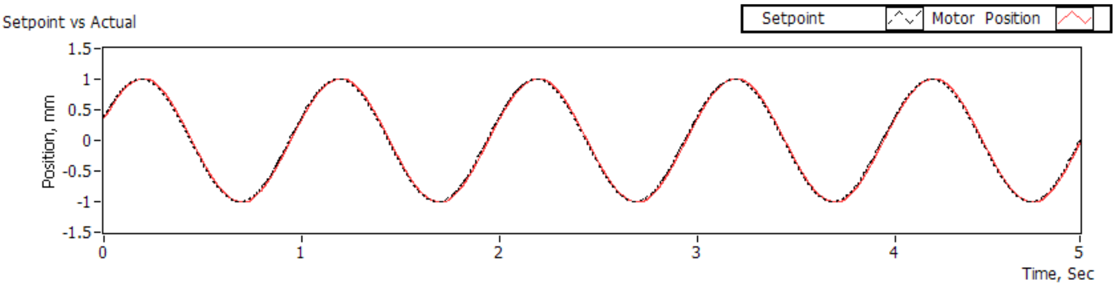
Baseline 0.1*Hz*



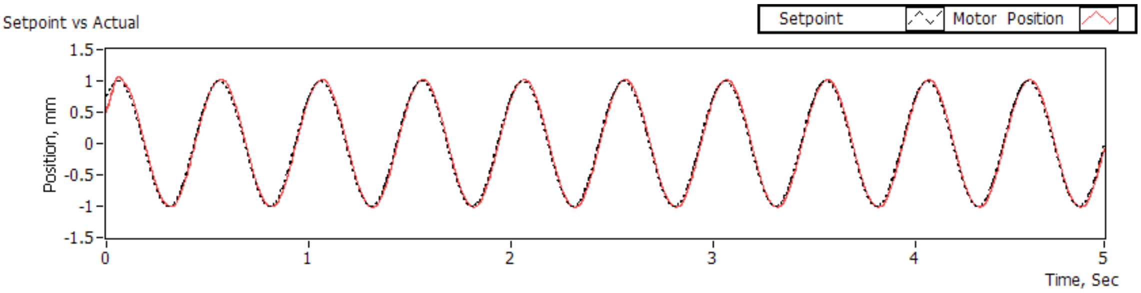
Baseline 0.5*Hz*



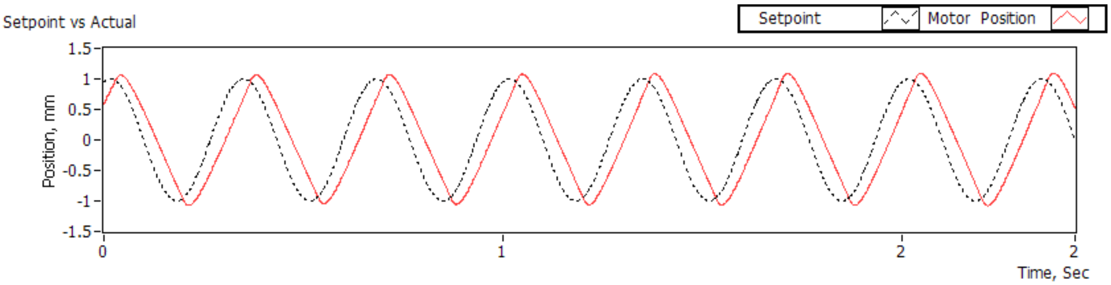
Baseline 1*Hz*



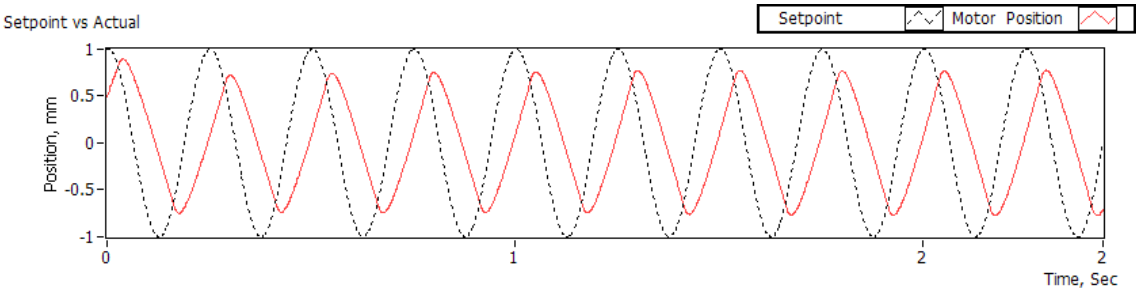
Baseline 2*Hz*



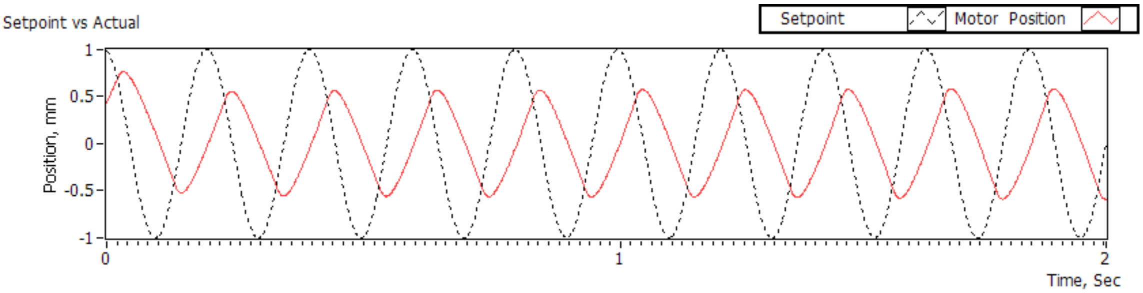
Baseline 3*Hz*



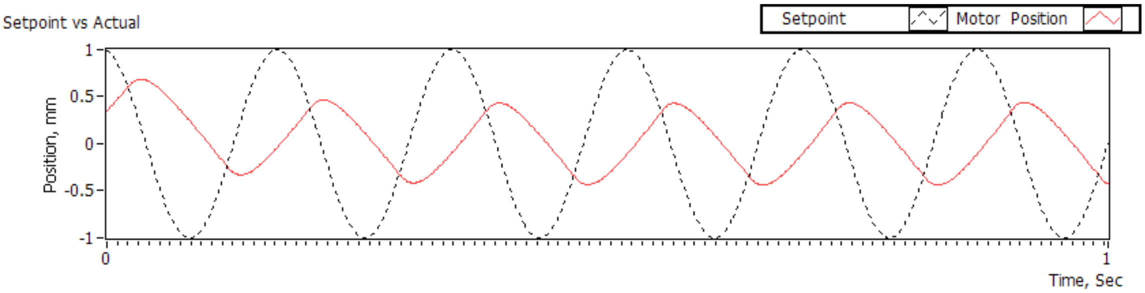
Baseline 4*Hz*



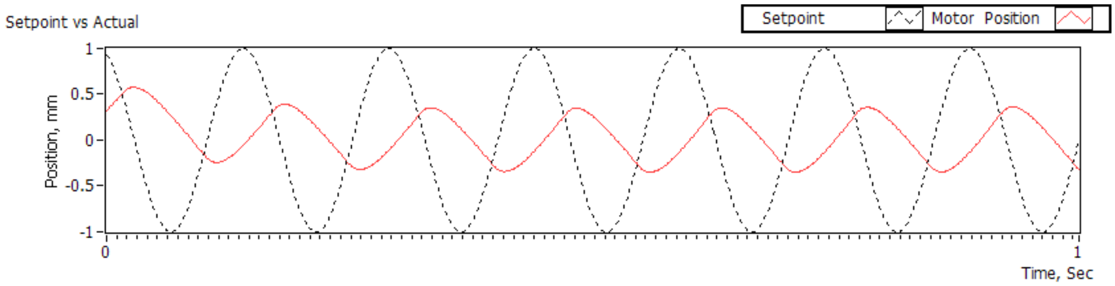
Baseline 5*Hz*



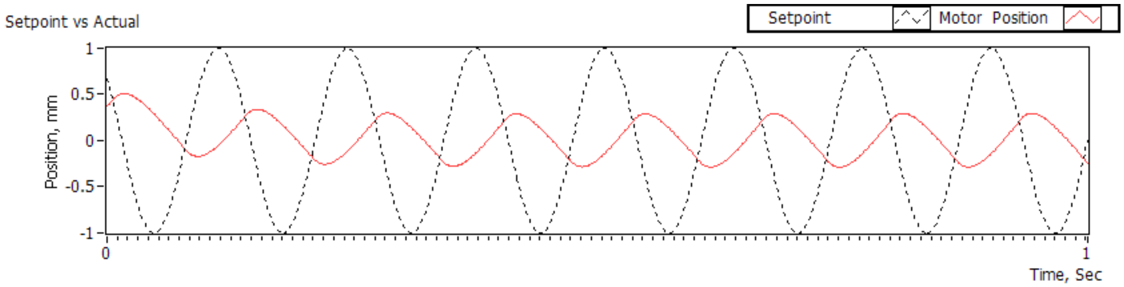
Baseline 6*Hz*



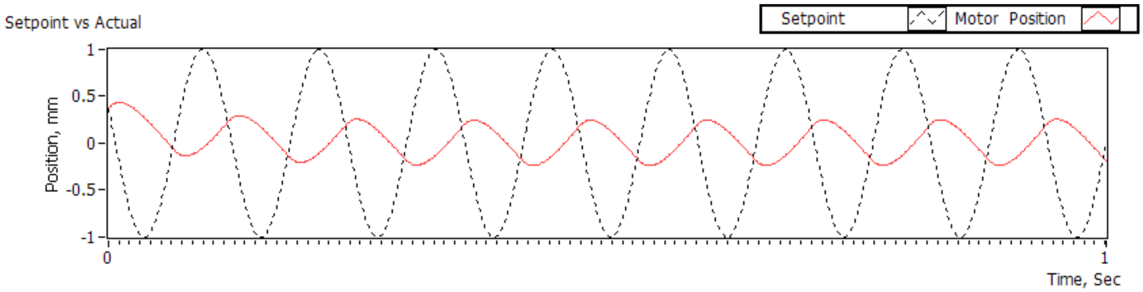
Baseline 7*Hz*



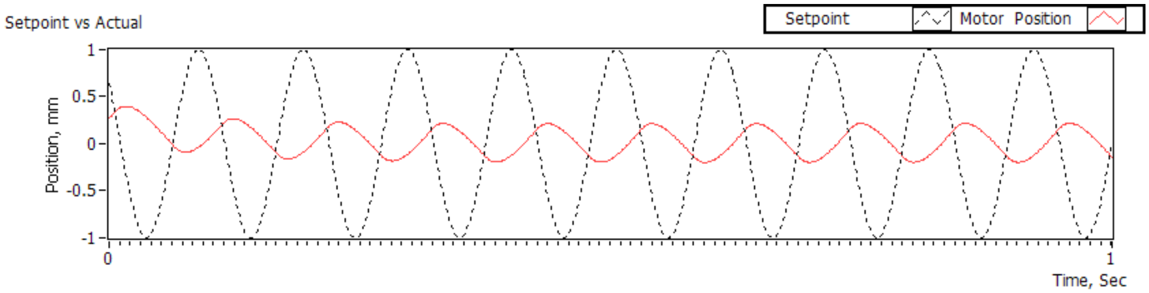
Baseline 8*Hz*



Baseline 9*Hz*

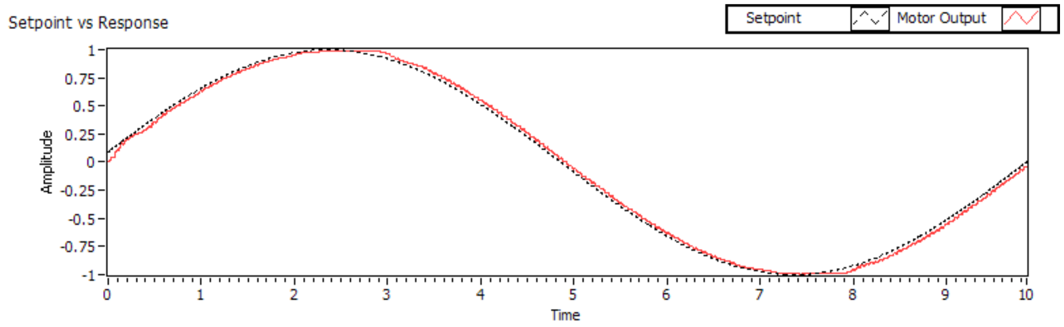


Baseline 10*Hz*

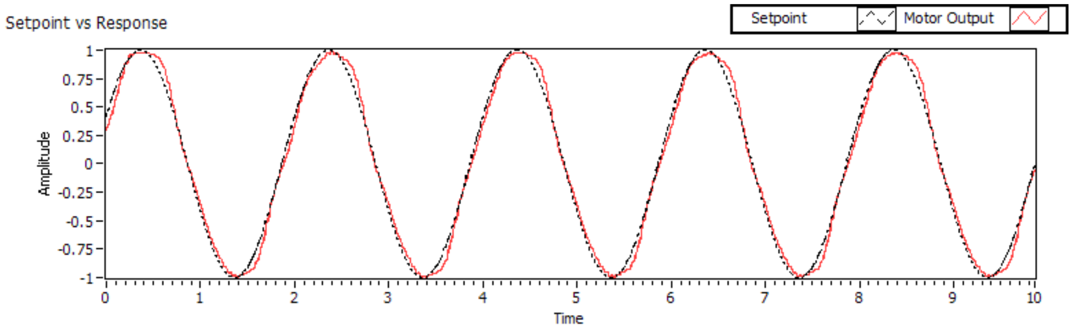


APPENDIX E: Smith Predictor with Intermittent Path Prediction using Interpolation Method and also Model Input Corrector for Frequency Analysis

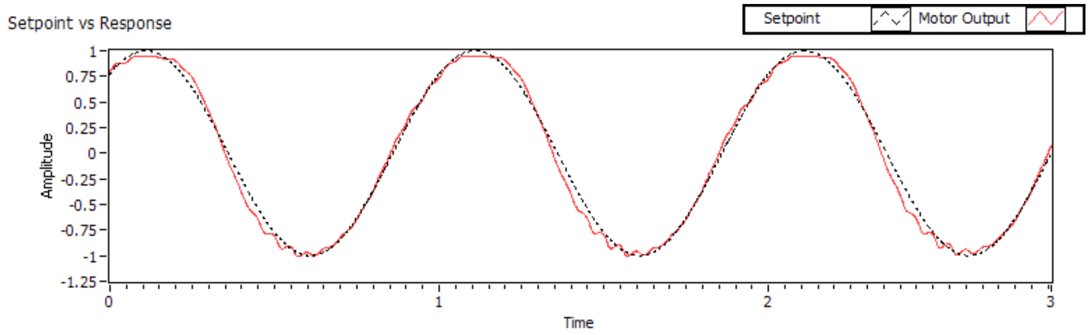
0.1Hz



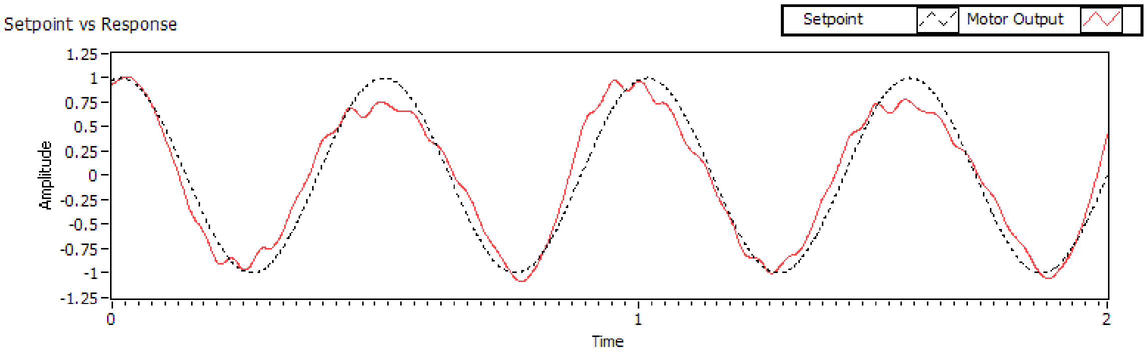
0.5 Hz



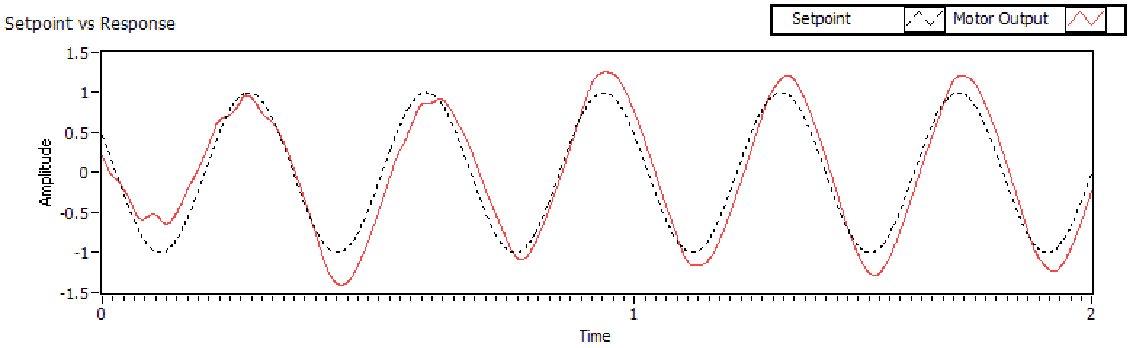
1 Hz



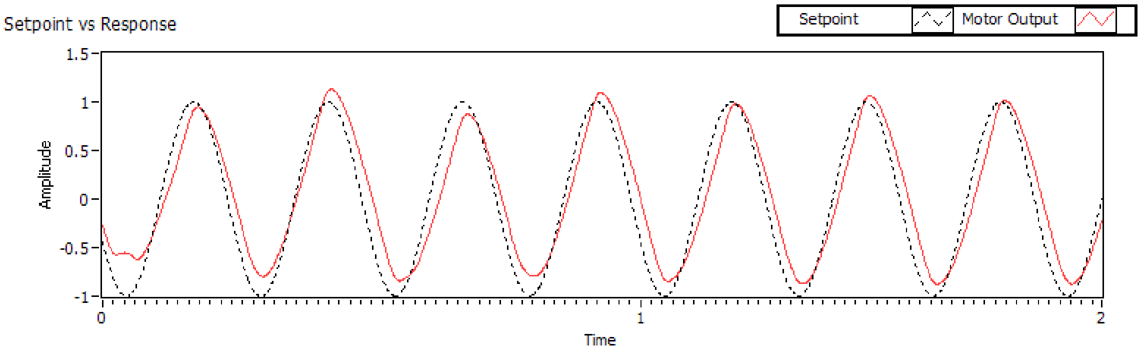
2 Hz


Setpoint vs Response
Setpoint
Motor Output
1.25
1
0.75
0.5
0.25
0
-0.25
-0.5
-0.75
-1
-1.25
Amplitude
0
1
2
Time

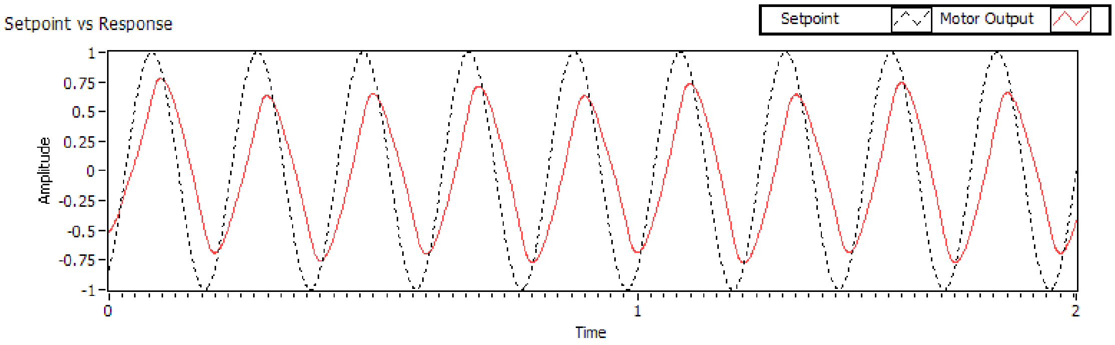
3 Hz


Setpoint vs Response
Setpoint
Motor Output
1.5
1
0.5
0
-0.5
-1
-1.5
Amplitude
0
1
2
Time

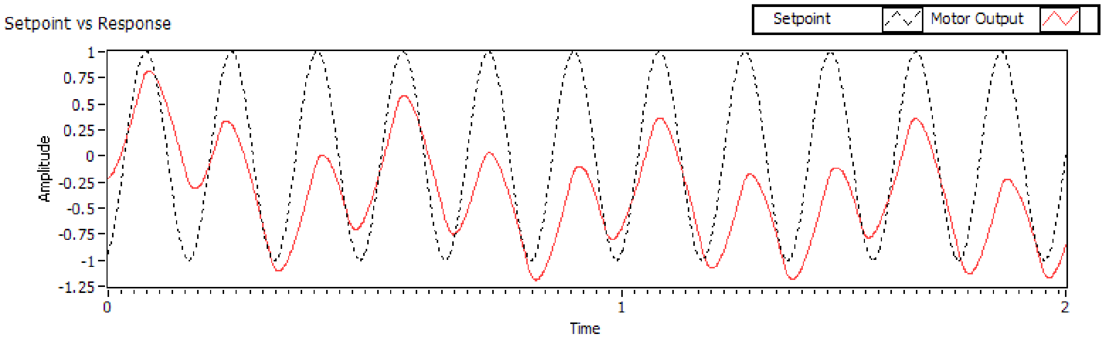
4Hz


Setpoint vs Response
Setpoint
Motor Output
1.5
1
0.5
0
-0.5
-1
Amplitude
0
1
2
Time

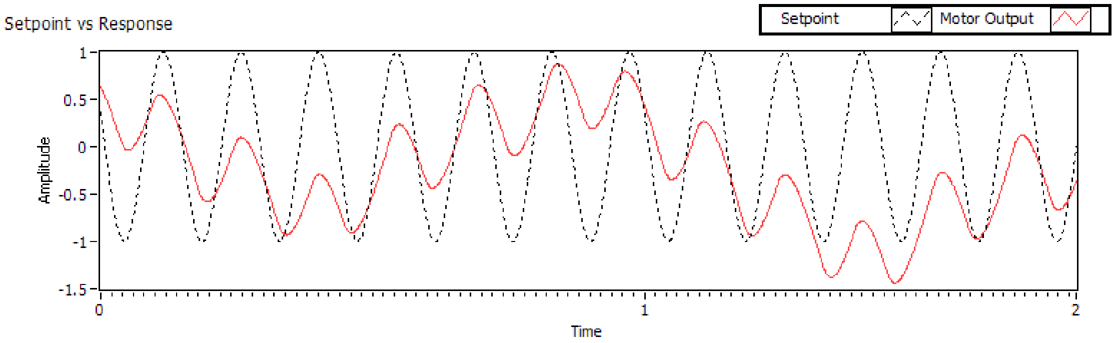
5 Hz



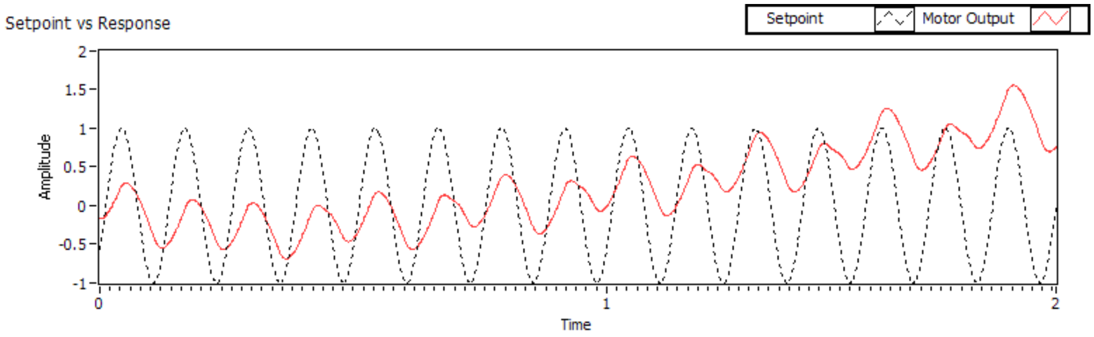
6 Hz



7 Hz



8 Hz



APPENDIX F: Simulation vs. Hardware Results

One of the reasons that the simulation results of the Modified Smith Predictor is dissimilar to the actual hardware experiment results is that the uncertainties and disturbance of the actual plant are not accurately modeled in the simulation environment. Since most of the disturbances are time-varying, it is difficult to simulate these conditions to accurately match the actual real time application. Although Gaussian noise is integrated in the simulation and separate stationary linear models are used for the plant and plant model representations, there will always be some other un-modeled elements of the actual plant missing in the simulation, restricting the simulation accuracy.

To show the impact of the time varying disturbance of a system, a hardware experiment was perform by observing the leadscrew position displacement with respect to the changes in the leadscrew's friction. The experiment measured the displacement of the leadscrew using the same input open-loop setpoints (20V peak-to-peak 0.5Hz sine wave) under three lubrication conditions: light, medium and heavy oil layer. Results as given in Figure F-1 show:

1. the baseline leadscrew, with no lubricant added, had a 6mm peak-to-peak without much position drift,
2. the light-lubed leadscrew had 8mm peak-to-peak with 1mm drift per period,
3. the medium lubed leadscrew had 13mm peak-to-peak with 1mm drift per period,
4. the heavy lubed leadscrew had 14mm peak-to-peak with 1.5mm drift per period.

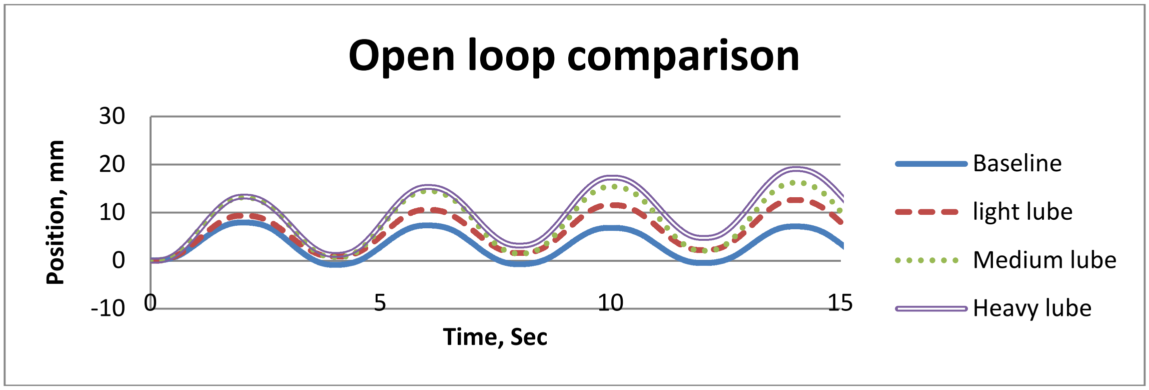


Figure F-1: Open loop comparison of the motor movement. This graph shows the influence of the friction to the leadscrew displacement with the same input command**:** 20V peak-to-peak 0.5Hz sine wave

Based on the results, it can be seen that the plant model will have to be updated at points in time in order to more closely represent the actual plant. This is because that the unmodeled elements of the plant, and also the time-varying disturbance will affect the output and behavior of the plant during the operation. This will cause the model discrepancy of the system to increase with time, degrading the path prediction and ultimate behavior of the Modified Smith Predictor. On the other hand, the simulation of the Modified Smith Predictor used two different linear plant models to replicate the real time system of having model discrepancy. By doing so, the model discrepancy of the system is linear and predictable, giving the simulation results better accuracy than the hardware validation results. Thus, a real time system identification algorithm has already been proposed for future work so that the plant model used in the real time application can be updated consistently to improve the controller performance especially when Model-based controller is used in the application.