University of Southampton

Control of High Speed Chain Conveyor Systems

by

Andrew Dennis Barton BEng(Hons) AMIEE

A thesis submitted for the Degree of

Doctor of Philosophy

in

The Department of Electrical Engineering

March 1999

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRICAL ENGINEERING

Doctor of Philosophy

CONTROL OF HIGH SPEED CHAIN CONVEYOR SYSTEMS

by Andrew Dennis Barton

This thesis is concerned with the design, development and comparison of position controllers for multi-axis manufacturing systems. The main objective is to determine a control algorithm that provides accurate and repeatable position control over a range of operating conditions. Systems based around a chain conveyor and dispenser, and operating in either indexing or synchronising modes have been considered. Due to the nature of their construction these systems may not be adequately represented by a linear model. To experimentally assess controller performance a 2 axis system has been developed which consists of a short section of chain conveyor and a dispenser. From this experimental apparatus an approximate linear model and a theoretical non-linear model have been obtained. These models have been used for design and verification of controllers before implementation.

Proportional, integral and differential (PID), model and knowledge based controllers have been developed and their performance compared experimentally. Standard PID can be improved by the addition of a velocity feedforward term. Further improvement in performance can be obtained by the implementation of model based controllers such as optimal and direct digital. Far less favourable comparisons were obtained with lag/lead compensators and fuzzy logic controllers. The greatest performance improvement was obtained through the use of a sufficient iterative learning controller. This algorithm has the dual advantages of only requiring the addition of a single tuning parameter to the standard PID system and does not require knowledge of the system dynamics. The controller monitors the error and adapts to changes in system dynamics to ensure a sufficiently small tracking error. Computational requirements are small enough to allow implementation on existing programmable logic controller(PLC) based systems.

Contents

List of Figures

To Paul, Martin and especially Rachel without whose belief,
cajoling, help and support, this thesis
may never have been finished.

Chapter 1

Introduction

Chain conveyor systems are prevalent in a number of industries, particularly food manufacturing. This highly competitive industry needs to reduce manufacturing costs by obtaining greater production rates. However, the combination of simple mechanical components and standard three term controllers can make accurate position control difficult as the throughput is increased. Improvements in production rate by increasing the speed of operation are limited as the acceleration of the product quickly becomes unacceptable. Further improvements must be obtained by minimising the deadtime required to allow the system to settle. This can only be achieved by reducing following errors. To obtain this desired improvement in control, it is necessary to look at methods beyond that of the traditional three term controller.

The type of conveyor systems found in the food industry are constructed from components that contain a number of non-linearities. Generally, motion is provided by an induction motor which has a non-linear characteristic, and encoder feedback produces quantisation of the position signal. Simple mechanical components are used which lead to significant Coulomb friction, and there is variation in friction and stiffness values as the position of the conveyor changes. Analysis of control methods based entirely on simula-

tion may therefore suggest solutions that are not valid experimentally. To allow experimental analysis of controller algorithms, a section of a typical conveyor and dispenser system has been constructed[1], and linear and non-linear[2] models obtained for design purposes. Model and knowledge based controllers have been developed and compared to the performance obtained by using standard three term control.

Due to operational constraints there is typically no position sensor on the conveyor or dispenser. Feedback is only available at the motor shaft, placing the conveyor and dispenser outside the control loop. Closing this loop, without adding to the sensor requirements, has been considered. The conveyor and dispenser form a two axis system that is required to remain synchronised. Methods of controlling the relative position of the two axes rather than the absolute position of each individual axis have also been investigated.

1.1 Current Manufacturing Technology

As the level of factory automation increases, the requirement to move product swiftly, accurately and reliably around the production site becomes more critical. Recent advances in technology have seen the introduction of automatic remote guided vehicles (ARGV's) to move product, but the conveyor still dominates in the majority of applications. It's fundamental simplicity, combined with the advances in power electronics and motor design make it the ideal material handling solution from automation of warehouse storage to bulk handling at coal mines, quarries and power stations. The very different applications to which conveyors are suited also exhibit very different requirements. Bulk handling applications generally require constant speed operation, capable of moving product immense distances. The principal difficulties with these form of conveyors is starting and the safety of personnel working near them, especially important in the confines of a

mine[3, 4]. These systems are frequently based around belt conveyors and it has been shown that under certain conditions a loaded belt can exhibit chaotic behaviour[5, 6].

A wide range of conveyors are used in the manufacturing industry and are described as either overhead, above floor or below floor, where the above floor type is the most common[7]. These are then divided into three groups based on their construction; belt, roller or chain. Each type of conveyor has advantages that make it suitable to a particular application.

Belt conveyors are ideal where a workpiece is to be removed and replaced on the conveyor, or where the workpiece is of a complex shape. Usually they are quieter than other types and require little maintenance. However, as there is only a friction connection between the conveying surface and the drive they tend to be restricted to applications where acceleration is limited.

Roller conveyors consist of a large number of consecutive rollers. To work satisfactorily the items placed on the conveyor must have an underside that is flat and sufficiently large to extend over a minimum of three rollers, otherwise some form of carrying tray is required. If the application requires a drop in height then unpowered rollers provide a simple solution. A roller conveyor can be powered by using a belt to connect the rollers to a shaft placed beneath them. A recent development in roller conveyor technology is to construct the roller as a small induction motor. The outside of the roller forms the rotor with power fed to the stator via fixed points at the ends. Some applications, especially in flexible manufacturing systems, require that the products on the conveyor are delayed before reaching the next process. Roller conveyors achieve this by slipping or using spring loaded clutches that disengage from the drive when the load becomes too high. Queues can be created by blocking the progress of one unit causing the rollers under the load to slip or disengage. As the conveyor upstream still operates further product will be brought up behind the first, thereby creating an accumulator.

Where the conveyor is required to follow a stop/start profile, then a positive connection between the drive and the conveying surface is needed. Chain conveyors fulfil this requirement and are used in a number of industries where either high loads or high accelerations are demanded. Their robust construction makes them eminently suitable for systems dealing with corrosive chemicals or where hygiene requirements mean regular cleaning. Simplicity of construction allows straightforward and inexpensive design of conveyors suited to a specific task, and chain manufactures have developed a wide range of attachments to expedite this.

Recent developments have considered the use of plastics in the construction of the chain and associated components[8]. Sprockets made from nylon appear to provide an increase in lifetime over their steel counterparts[9]. Plastic components are obviously of interest where there may be concerns about chain lubricant contaminating the product. The small amounts of lubricant on most chains makes this concern unnecessary unless there is direct contact between the chain and the product.

The control of conveyors that require a repeatable index can be accomplished either electronically or mechanically. Systems based on cams or a hydraulic walking beam[10] can offer a solution that achieves an accurate and repeatable index at minimal complexity. Cam systems do provide a very high speed, accurate index and are used in lightweight, high volume systems such as PCB assembly. However, they are susceptible to wear and require regular maintenance. Alteration of the indexing profile can only be achieved by replacement of the cam.

Detailed reviews of conveyor types and their selection have been published[11, 12, 13]. The simplicity of conveyor construction means that there is very little published material in this area with conveyors described as a product that any small engineering company is capable of constructing[14]. The majority of published material concerning conveyors deals with statis-

tical modelling for determining the most efficient layout of a manufacturing plant[15, 16]. The reason for the lack of research into engineering aspects of conveyors, particularly control, may be due in part to a lack of investment in advanced manufacturing technology. UK companies require a payback time of, on average, 2.1 years and easily quantifiable justification for investment[17]. A Department of Trade and Industry report[18] criticises the food industry in particular for not being more alert to the opportunities presented by advanced technology. Though having a turnover of about £40 billion per annum, the food industry has a very small level of automatic control, considerably behind that found in the chemical industry[19]. A recent article on a state of the art food production line indicates that the distance between product on the conveyor is controlled by an operator using a potentiometer to adjust the speed of the conveyor[20]. This apparent apathy on the part of the food industry to encourage research may account for the lack of published works relating specifically to food manufacturing applications.

The type of system studied in this project is a chain conveyor of a type frequently found in the food industry. Typically they are constructed from very simple mechanical components, the chain and sprocket being based on standard roller chain, driven from either a helical gearbox or via a belt drive. Originally, motion was provided by a d.c. servo system, but the requirement of a high level of hygiene necessitates the use of high pressure water or steam hoses in cleaning. A d.c. motor capable of withstanding such environmental demands is expensive and requires a long period of downtime for maintenance. With the advent of inexpensive power electronics the use of induction motors became viable. Position feedback is provided by an optical encoder placed on the motor shaft with control provided by the industry standard proportional, integral and differential (PID) controller also known as a three term controller.

This approach has a number of limitations. The tuning of the controller

is achieved manually which is unlikely to produce an optimum solution. Alteration of system dynamics over time will cause a decrease in performance unless the controller is manually re-tuned. These perform adequately but are unable to account directly for higher order effects that may be introduced by non-linearities. The use of modern drives could be used to overcome some of these problems. Flux vector control of induction motors claims to provide a performance that is comparable to a d.c. servo system, and the supply of inverters based around this philosophy is becoming more widespread. Replacement of existing drives with flux vector versions is often undesirable due to the expense and re-training involved.

Feedback for the controller is provided by an encoder mounted on the motor shaft. This leaves the mechanical components of the system outside the control loop which causes a number of control difficulties. As the chain ages, the bushings wear causing the chain to effectively become longer. Teeth in the gearbox and the sprockets suffer wear, and drive belts stretch introducing backlash into the system. The net effect is to decrease the accuracy of the indexing motion and this can be a reason for not using chain conveyors[21]. Currently the effect of open loop disturbances is overcome by regular manual resetting or automatic homing of the system once every revolution of the conveyor. Both methods are undesirable as while the conveyor is being homed, production is delayed and throughput decreased. Automatic compensation for position disturbances, commonly referred to as drift, would increase the production efficiency of the system but must be achieved without adding significantly to the sensing requirement. Any sensing must be non-invasive and impervious to contamination by environmental factors such as water, steam or electromagnetic interference. It should also not require long runs of expensive multi-core screened cable. An encoder placed on a mechanical output shaft is unacceptable for cabling reasons and the difficulty in achieving accurate mounting. A system based on optics is

also unacceptable as it may easily become obscured by spilt product.

1.2 Research Objectives

The objective of this research is to address some of these problems to produce an improvement in the performance of the system. Performance is measured in industry by the number of units completed per hour. Improvement could be achieved by increasing the speed of operation, however this is limited by acceleration. Accelerations that are too rapid make it difficult to maintain the position of the product on the conveyor, though slight increases in speed are possible if position accuracy is not degraded. To obtain the most significant increase in throughput, the accuracy of the position trajectory following should be improved. Currently it is necessary to introduce a time delay to allow the conveyor to settle. For example, if the time taken to dispense the product is 0.3s, a delay of 0.1s may be added before starting dispensing to ensure that the system is in the correct position. If the accuracy with which the conveyor followed the demand profile was improved then this delay time could be reduced. For each individual unit the reduction may only be a fraction of a second, but over the course of a 15 hour day this would accumulate to a significant number of additional units completed. The control problem is therefore one of accurate position tracking with the current solution, PID, relying on technology that is in effect 60 years old. The application of modern control techniques to the problem may produce a performance improvement.

To be able to determine the performance of any new controller it is first necessary to determine the performance of the PID system. This requires construction of a multi-axis test facility which consists of a short section of chain conveyor with a dispenser above driven through belt drives by induction motors. Control of both axes is achieved by a single PC equipped with suitable I/O devices. The construction is described in Chapter 2.

To allow the design of more advanced controllers the experimental apparatus has been used to develop two separate models as described in Chapter 3. An experimentally obtained linear model allows the design of controllers using standard control techniques, but does not account for non-linearities in the system. Controllers are therefore veriffed with a theoretically derived model which includes non-linearities before implementation on the experimental apparatus.

In Chapter 4, the standard PID controller is implemented and tuned. A measure of it's performance is obtained. A number of improvements that are based around the PID controller are developed and investigated.

The design of model based controllers is investigated in Chapter 5, specifically direct digital and optimal control. Fuzzy and iterative learning controllers[22], considered to be knowledge based controllers, are described in Chapter 6.

The conveyor and dispenser are effectively open loop and as such their position will tend to drift away from that of the motor. Chapter 7 considers a method of overcoming this difficulty by only using existing low resolution homing sensors. Methods of controlling the relative position of the conveyor and dispenser to ensure synchronisation[23] are also investigated here.

Conclusions, Further Work and the most suitable method of control for chain conveyor systems are presented in Chapter 8.

Chapter 2

Experimental Apparatus

2.1 Introduction

The experimental apparatus is representative of manufacturing systems frequently found in the food industry. It consists of a short section of chain conveyor operating in conjunction with a product dispenser. Such systems are designed to have two modes of operation, either indexing or synchronising. In indexing, the conveyor positions items placed upon it consecutively under the product dispenser in a stop/start motion. Product is only dispensed while the conveyor is stationary. Synchronising requires the conveyor to move at constant velocity with the dispenser accelerating to match, product being dispensed when the velocities are synchronised. The dispenser then rapidly returns to the origin in readiness for the next unit. Figure 2.1 shows the components of both synchronising and indexing demand profiles, both consisting of positioning and dispensing sections. In both operating modes the requirement is for accurate position control, especially during the dispensing time. Typical specifications are for $\pm$1mm which equates to approximately $\pm$0.05rad at the motor shaft with a dispensing time of 0.4s.

Each axis of the apparatus is constructed from the same basic components (Figure 2.2). An IBM PC is linked to an a.c. variable voltage, variable

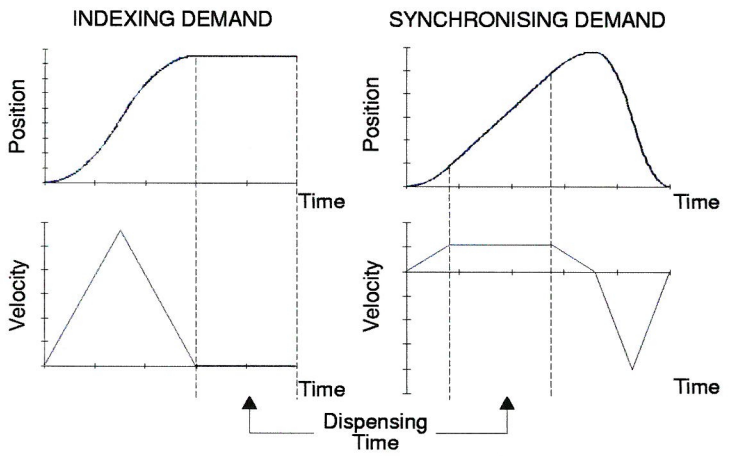

Figure 2.1: Indexing and synchronising demand profiles

frequency drive (VVVF) controlling induction motors with optical encoders to provide position feedback. The mechanical components of each axis are then driven through a belt drive.

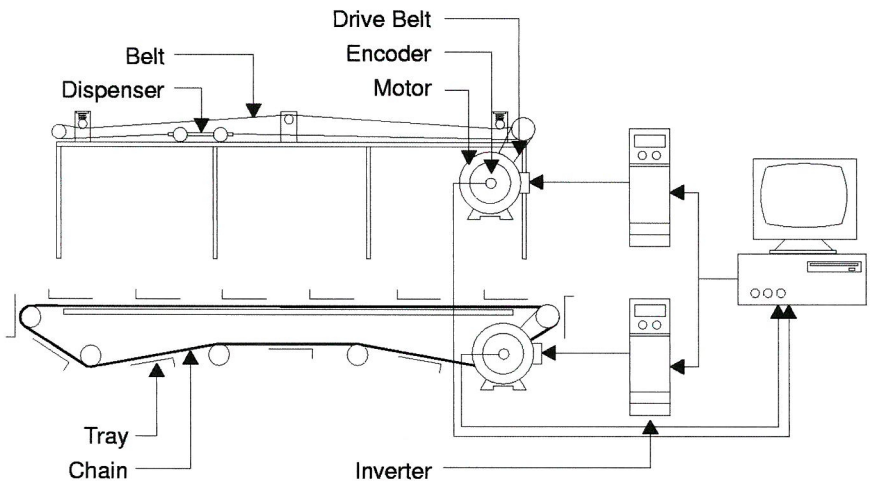

Figure 2.2: System overview

2.2 Electrical Systems

Figure 2.3 shows a schematic of the electrical connections in the system. The inverters provide three phase pulse width modulated (PWM) power to

the motors, with a constant voltage to frequency ratio. This has the effect of linearising the torque/speed characteristic of the induction motor. The inverters employed do not accept a bipolar demand signal, the analogue demand is therefore unipolar. Direction control is achieved by connecting the direction input of the inverter to a line on the PC's parallel port.

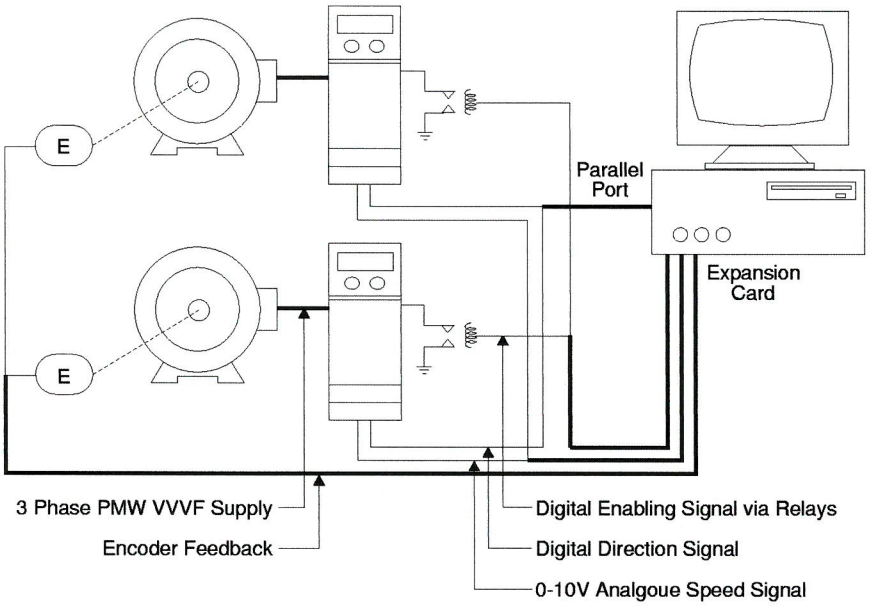

Figure 2.3: Experimental apparatus schematic

Interfacing between software and hardware is performed by a commercial ISA expansion card. This provides two $\pm$10V, 14 bit analogue outputs, two opto-isolated digital outputs for axes enabling and two 32 bit counters for encoder input. The encoder counters utilise edge triggering, therefore producing a fourfold increase in resolution. To minimise any effect of electromagnetic interference, all signal cabling employs screened and braided cable with the screen connected to 0V and earth at the PC's case.

2.3 Conveyor Axis

The conveyor (Figure 2.4) is constructed from two parallel lengths of steel roller chain with the conveying surface formed from aluminium plates attached to cross pieces at regular intervals. Power is provided by a delta connected squirrel cage induction motor driving through a 5:2 reduction timing belt drive. Additional bracing of the belt drive shafts minimise the nonlinear effects caused by variations in the drive pulley centre distances[24]. Position feedback is provided by a 500 pulse per revolution (PPR) incremental quadrature encoder with differential outputs fitted directly to the motor shaft.

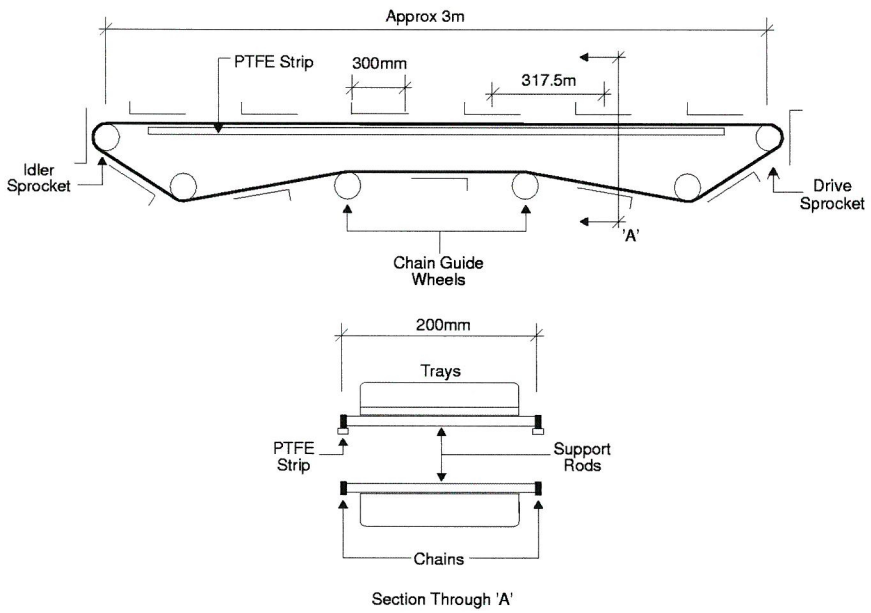

Figure 2.4: Conveyor axis schematic

A chain is in effect a finite length linkage. As it moves around the sprocket, periodic fluctuations in the chain forces are created. Dynamics due to this effect, known as polygonal action have been minimised by suitable selection of the drive sprocket[25, 26].

2.4 Dispenser Axis

The dispenser consists of a welded carriage running above the conveyor, guidance being provided by aluminium runners (Figure 2.5). The carriage is connected to a timing belt loop driven from one end by a motor/belt drive system similar to that of the conveyor. Sprung rollers at the carriage corners bear on the edges of the runners and maintain the carriage alignment when in motion.

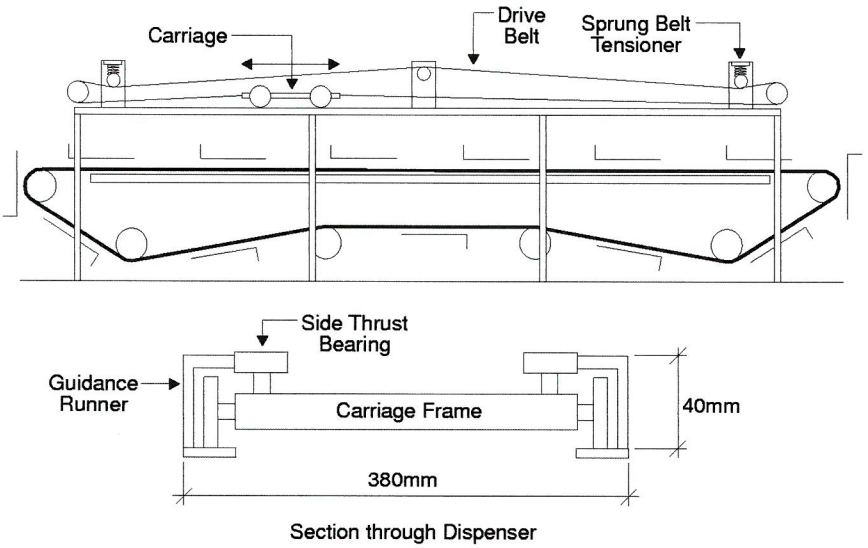

Figure 2.5: Dispenser axis schematic

2.5 Software

Control software is implemented on a 100MHz Pentium PC running DOS 6.2. A simple menu driven interface allows limited operator run-time configuration of the system. The software has been partitioned such that implementation of most controllers requires replacement of a single file.

Controllers have been implemented as interrupt service routines (ISR), the interrupt being generated from a programmable interrupt controller on

the I/O card. Step response data for the motor indicates a time constant of approximately 0.05s. It is generally accepted that the sample frequency should be between 5 and 20 times the time constant suggesting a sample frequency of only 400Hz[27]. In practice the power of modern microprocessors allows significantly higher sample frequencies to be employed, with commercial systems typically operating around 1kHz. As the sample frequency is increased then the demand profile will become smoother and approximate more to a continuous function. Selection of a sample frequency of 2.5kHz balances the requirement for a high sample frequency for good control and avoidance of aliasing while not introducing an excessive computational workload.

2.5.1 Demand Generation

Demands can be calculated in their entirety before initiation of the controller, and stored in a lookup table. A more memory efficient method is to determine the required demand at each sample instant. Multiplications should be avoided to ensure the maximum amount of time in each sample period for controller execution and housekeeping . If the measurement units are defined in terms of encoder counts and sample periods, then it is possible to generate the position demand at each sample instant by successive summations. To achieve this, values of accelerations, velocities, positions and times of key points in the demand profile are calculated prior to initiation of the controller and stored in a 2 dimensional array (Figure 2.6). At each sample instant the required value of acceleration is added to the previous velocity demand, which in turn is added to the previous position demand. In this way position and velocity demand information can be obtained with the minimum of computational overhead. To ensure no drift of the demand signal due to rounding errors, at the penultimate sample instant, the algorithm determines the velocity that must be returned to ensure that the total

position demand is equal to the final desired position per unit.

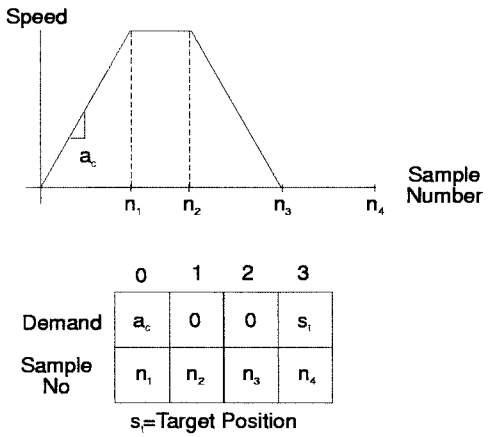

Figure 2.6: Typical indexing demand and calculation array

2.6 Analysis of Errors

The apparatus has been constructed to represent the type of system found in industry and this has required the use of simple mechanical components. The properties of the belts and chain will vary as the conditions around the apparatus alter. Over time the chain will stretch further altering the tension. This produces a system where it is difficult to accurately repeat experiments. However, over the duration of a single experiment, variation of system parameters will not introduce significant errors. Such errors that do occur are due to disturbances in the system. To determine the form of these errors the conveyor was operated at 50 Units per Minute (UPM) with the PID controller presented in Chapter 4. The position error at the completion of each unit was recorded and analysis shows that the error distribution approximates to a normal distribution (Figure 2.7). For a normal distribution the error in a single measurement is[28]:

$$x = \bar{x} \pm \frac{\sigma}{\sqrt{n}} \tag{2.1}$$

where n is the number of tests and σ is the standard deviation. For the majority of experiments conducted, the mean square error has been calculated based on twenty measurements. An approximation for the error can then be found by taking the square root of the mean square value and dividing by $\sqrt{20}$.

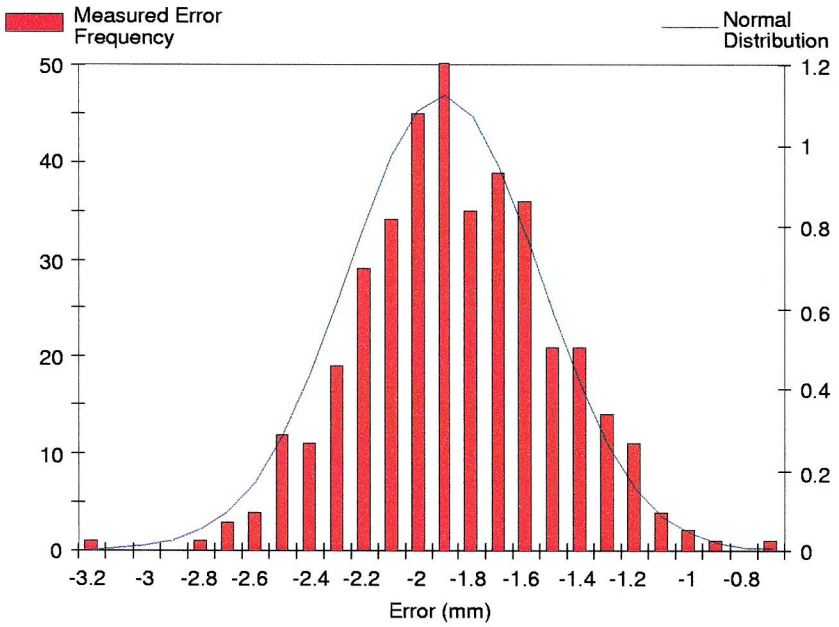

Figure 2.7: Measured error and normal distributions

2.7 Summary

The design of the system allows implementation and testing of various controllers to be undertaken. The two axes can be operated as a single unit or individually. Under the individual mode of operation a spare encoder counter is available for additional feedback or measurement. The design of the system also allows the recording of time, demand and feedback data for subsequent analysis.

Chapter 3

System Modelling

3.1 Introduction

Modelling of the system has been undertaken using two separate approaches, each leading to a different expression. The first method required a theoretical analysis of the individual system components, and produced a separate expression for each element. Each expression was then verified against step response data. The individual models were then combined, resulting in a time-based model of the complete system. Concurrently, a second representation was constructed by fitting a linear model to experimentally obtained frequency response data. Both models are necessary due to the use of a discrete time simulator. This introduces errors into the frequency response of the theoretical model, while the step response of the frequency based model does not account for system non-linearities. Controllers have been designed with the linear model and then validated against the theoretical model before experimental implementation.

Realisation of both models has been performed in a commercial, hierarchical, graphical dynamic simulation software package. Models are constructed from a number of simple components and then grouped into a single element. Dynamic models of the plant are then readily embedded into con-

trol system models for design and analysis.

3.2 Time Based Model

3.2.1 Overall Model Construction

The components of each axis have been identified as the I/O card, inverter, induction motor, encoder, drive belt and associated pulleys, and the conveyor and dispenser. Each of the identified components forms a self-contained model. Non-linear friction is inherent in a number of components and it was considered appropriate to also represent it as a separate model. Though each sub-system has been represented as an individual block, each is closely dependent on both proceeding and succeeding blocks (Figure 3.1).

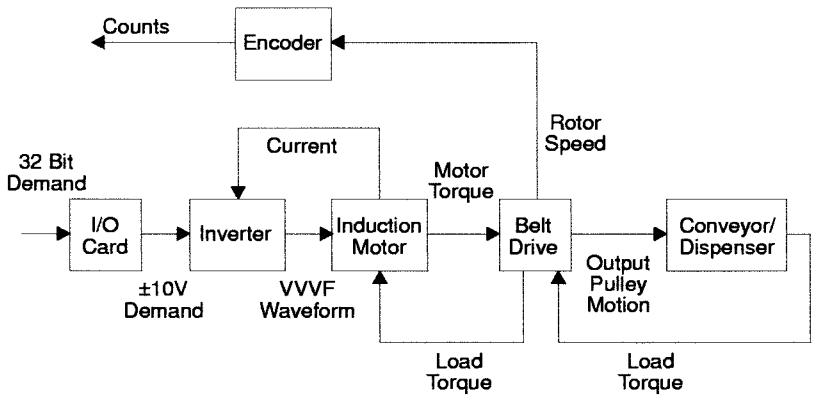

Figure 3.1: Time based model overview

3.2.2 Coulomb Friction

Friction can be described as consisting of three components[29, 30], a static friction component that only exists at zero speed, a constant torque component, referred to as Coulomb friction, and linear viscous friction (Figure 3.2). All three components are present in the experimental apparatus, with

Coulomb friction being dominant at normal operating speeds.

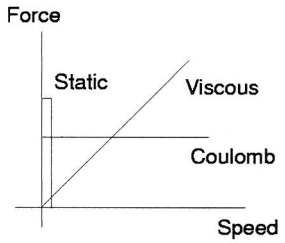

Figure 3.2: Components in a frictional torque

The static friction opposes the accelerating torque up to a set level, estimated to be a factor of 1.1 that of Coulomb friction. When the velocity is greater than zero, the non-linear friction component falls to the level of Coulomb friction, thereafter remaining constant until the velocity returns to zero. The effect is described by:

$$T_{nlf} = \begin{cases} T_{acc} & : \quad \dot{x} = 0;\ T_{nlf} < 1.1T_{col} \\ T_{col} & : \quad \dot{x} > 0 \end{cases} \tag{3.1}$$

where T_{nlf} is the non-linear frictional torque, T_{col} is the Coulomb torque value and T_{acc} is the accelerating torque.

Figure 3.3 shows the experimentally measured torque/speed characteristics from which it can be seen that the points in the dispenser characteristic are more scattered than for the conveyor. Due to the limited operational length of the dispenser measurements must be recorded rapidly whereas the conveyor can be operated for a considerable period of time allowing transients to decay and more time for recording readings. Estimates for the Coulomb and viscous friction values can be made from the y-axis intercept and the gradient respectively.

3.2.3 I/O Card

The input to the I/O card is a digital value representing the desired analogue output in millivolts. Measurement of the output voltage has shown

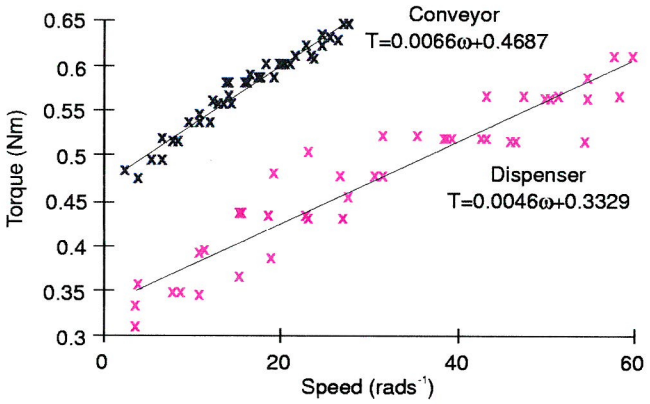

Figure 3.3: Friction characteristics for the experimental apparatus

an additional attenuation gives an overall gain of 0.98×10^{-3}. Quantisation caused by the digital to analogue conversion is also included.

3.2.4 Induction Motor

It is possible to represent a three phase induction motor directly using a phase variable model[31, 32]. However it is more common to represent a.c. motors using a two axis, d.q. approach[33, 34, 35, 36, 37, 38, 39, 40]. If the assumptions inherent in the d.q. method are acceptable then this provides a solution that requires less computational effort than other methods[41]. It is possible to compensate if the d.q. assumptions are unacceptable though this increases the complexity[42].

The equations for a two axis, d.q. representation of an N pole motor are:

$$\begin{bmatrix} v_{ds} \\ v_{qs} \\ v_{dr} \\ v_{qr} \end{bmatrix} = \begin{bmatrix} R_s + L_s p & 0 & M_{sr} p & 0 \\ 0 & R_s + L_s p & 0 & M_{sr} p \\ M_{sr} p & N\omega_r M_{sr} & R_r + L_r p & N\omega_r M_{sr} \\ -N\omega_r M_{sr} & M_{sr} p & -N\omega_r M_{sr} & R_r + L_r p \end{bmatrix} \begin{bmatrix} i_{ds} \\ i_{qs} \\ i_{dr} \\ i_{qr} \end{bmatrix} \tag{3.2}$$

$$T_r = J_{mot} p\omega_r + B_{mot}\omega_r - NM_{sr}(i_{ds}i_{dr} - i_{ds}i_{qr}) \tag{3.3}$$

where R_s, R_r and L_s, L_r are the stator/rotor resistance/self inductance, M_{sr} is the mutual inductance, w_r is the rotor mechanical angular velocity and p

is the differential operator. Parameter values were obtained by no-load and locked rotor tests. To allow simulation Equation 3.2 is expanded and pi_{dr} eliminated from the first equation and pi_{qr} eliminated from the second. The equations are then solved at each time step for the motor currents.

The motor model has been verified against torque/speed and speed step response data (Figure 3.4). The motor was assumed to be fed from a fixed voltage and frequency supply on an infinite busbar.

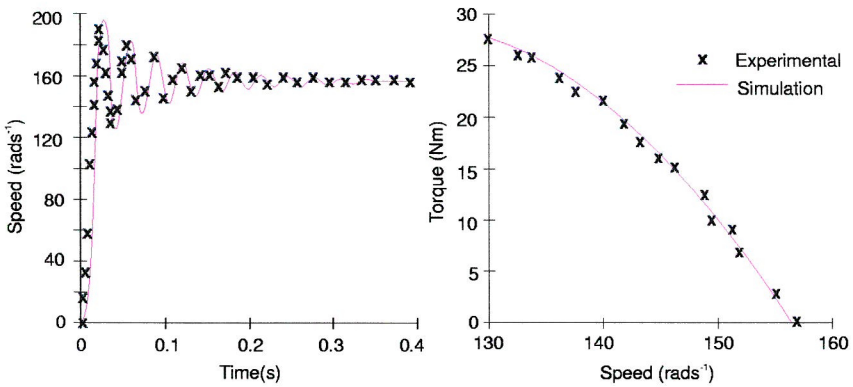

Figure 3.4: Characteristics of the induction motor on fixed supply

3.2.5 Inverter Model

For an inverter using sinusoidal pulse width modulation (SPWM) all harmonics less than $2h-1$ are eliminated[43], where h is the number of pulses per half cycle such that:

$$h = \frac{f_c}{2f_o} \tag{3.4}$$

if f_c and f_o are the carrier and output frequencies. For the inverters in the experimental apparatus the PWM switching frequency is 2.9kHz which implies that all harmonics below 1.45kHz are eliminated for a 1Hz output frequency. The electrical circuit of the induction motor can be considered to be a low pass filter with a bandwidth of a few tens of hertz[44], implying that the inverter can be represented as an ideal variable frequency, variable

amplitude sinusoid such that:

$$v_o(t) = \sqrt{2}V_{rms}\alpha \sin(\beta t) \tag{3.5}$$

where β is the demanded output frequency in rads^{-1} and α is an amplitude scalar to give the correct V/f characteristic.

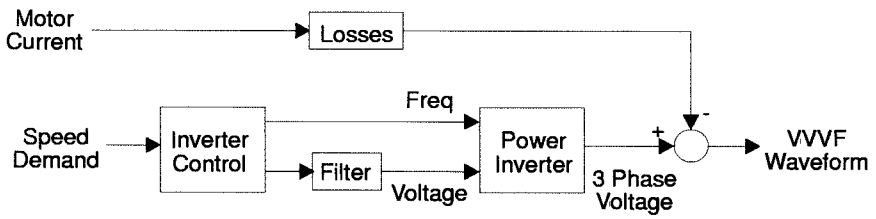

Figure 3.5: Overview of the inverter model

The inverter model consists of a control section to determine the required voltage and frequency, and an inverter to generate the VVVF waveform based on Equation 3.5 (Figure 3.5). The resistance and capacitance of the d.c. link are approximated as a low pass filter on the voltage demand signal. Load current is used to determine inverter losses which are then subtracted from the phase voltages.

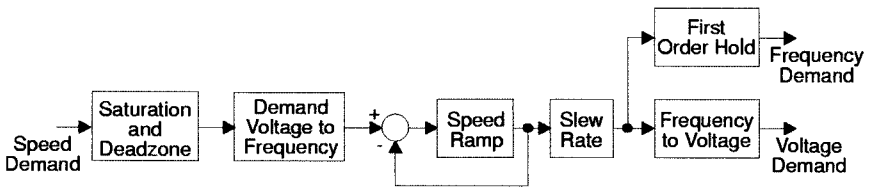

Figure 3.6: Block diagram of the inverter control block

Implementation of the inverter control block (Figure 3.6) is based upon information supplied by the manufacturer. Demand voltage to frequency and frequency to RMS voltage characteristics have been determined experimentally (Figure 3.7).

Figure 3.8 shows the experimental and theoretical step responses for the drive. Experimentally, a full speed step produces slightly underdamped os-

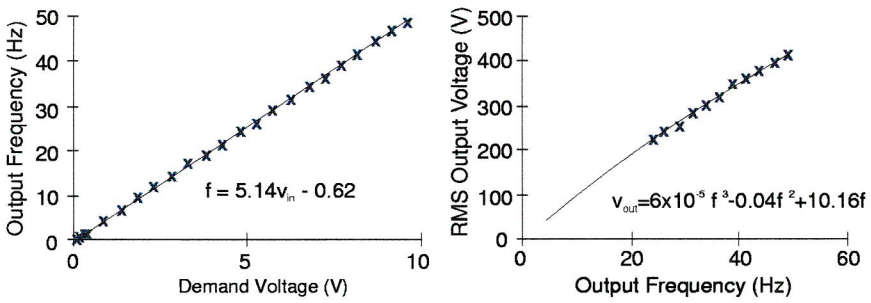

Figure 3.7: Frequency and voltage characteristics for the inverter

cillation compared to the theoretical simulation which exhibits considerable oscillation. The coupling of the motor to the conveyor/dispenser introduces a gear ratio that only reduces the motor speed by a factor of 2. Therefore, the drive will not be required to operate at high speeds and the proposed model is adequate for simulating the time response of the system.

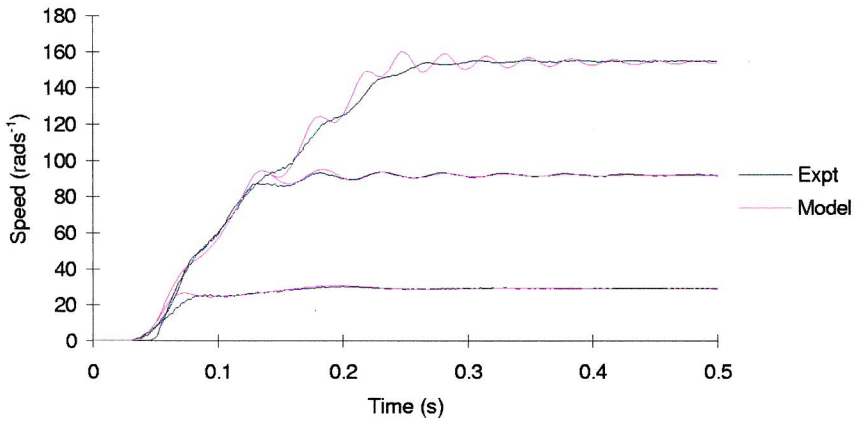

Figure 3.8: Step response of the induction motor on inverter supply

3.2.6 Belt Model

The belt model consists of a spring/mass/damper representation (Figure 3.9). Damping could be included in parallel with each spring element[24, 45], but this is negligible in comparison with other damping elements in the system. The input pulley contains inertia's and linear and non-linear

damping for both the motor and the input pulley.

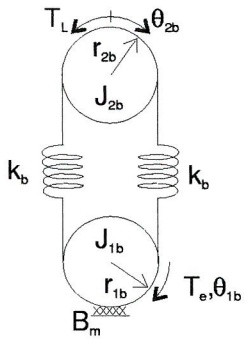

Figure 3.9: Belt model implementation

The equations for the linear model based on the above assumptions are:

$$\begin{bmatrix} \ddot{\theta}_{1b} \\ \ddot{\theta}_{2b} \end{bmatrix} = \begin{bmatrix} \frac{-2r_{1b}^2 k_b}{J_{1b}} & \frac{-B_m}{J_{1b}} & \frac{2r_{1b}r_{2b}k_b}{J_{1b}} & 0 \\ \frac{2r_{1b}r_{2b}k_b}{J_{2b}} & 0 & \frac{-2r_{2b}^2 k_b}{J_{2b}} & 0 \end{bmatrix} x(t) + \begin{bmatrix} \frac{1}{J_{1b}} & 0 \\ 0 & \frac{-1}{J_{2b}} \end{bmatrix} u(t) \quad (3.6)$$

$$y(t) = \begin{bmatrix} \theta_{2b},\ \dot{\theta}_{2b},\ \ddot{\theta}_{2b},\ J_{1b}\ddot{\theta}_{1b},\ T_1,\ \dot{\theta}_{1b} \end{bmatrix}^T \quad (3.7)$$

where:

$$x(t) = \begin{bmatrix} \theta_{1b} \\ \dot{\theta}_{1b} \\ \theta_{2b} \\ \dot{\theta}_{2b} \end{bmatrix}; u(t) = \begin{bmatrix} T_e \\ T_l \end{bmatrix} \quad (3.8)$$

where T_e is the motor electrical torque and T_l is the load torque from the mechanical system. The total load torque, $T_{lb} = T_1 + J_{1b}\ddot{\theta}_{1b}$, is an output from the belt model and is subtracted from the motor electrical torque to obtain the accelerating torque.

3.2.7 Conveyor Model

It is possible to analyse in detail the forces in chains in power transmission systems[26, 46, 47, 48, 49, 50], but these do not directly provide an indication of the motion of the chain along it's length. An analysis of the motion can be obtained by representing each link as a spring/mass system[51], but the

complexity of such a representation prevents simulation in a reasonable time. To reduce complexity, the chain and conveying surface have been combined and represented as two spring/mass/damper models of equal length (Figure 3.10). Additional inertia and damping terms represent the sprockets. Input to the system is the motion of the output belt pulley, with load torque forming the output.

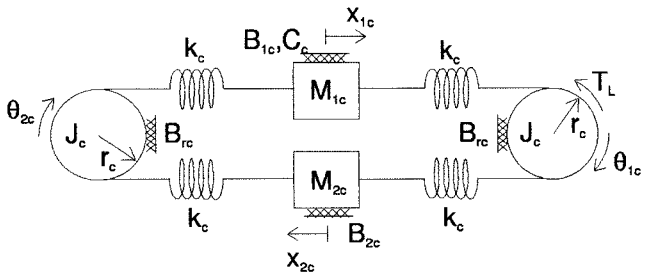

Figure 3.10: Conveyor model implementation

The conveyor model equations are:

$$\begin{bmatrix} \ddot{x}_{1c} \\ \ddot{\theta}_{2c} \\ \ddot{x}_{2c} \end{bmatrix} = \begin{bmatrix} \frac{-2k_c}{M_{1c}} & \frac{-B_{1c}}{M_{1c}} & \frac{r_c k_c}{M_{1c}} & 0 & 0 & 0 \\ \frac{r_c k_c}{J_c} & 0 & \frac{-r_c^2 2k_c}{J_c} & \frac{-B_{rc}}{J_c} & \frac{r_c k_c}{J_c} & 0 \\ 0 & 0 & \frac{r_c k_c}{M_{2c}} & 0 & \frac{-2k_c}{M_{2c}} & \frac{-B_{2c}}{M_{2c}} \end{bmatrix} x(t) + \begin{bmatrix} \frac{r_c k_c}{M_{1c}} \\ 0 \\ \frac{r_c k_c}{M_{2c}} \end{bmatrix} u(t) \tag{3.9}$$

$$T_l = J_c \ddot{\theta}_{1c} + B_{rc} \dot{\theta}_{1c} + 2r_c^2 k_c \theta_{1c} - r_c k_c (x_{1c} + x_{2c}) \tag{3.10}$$

where:

$$x(t) = \begin{bmatrix} x_{1c} \\ \dot{x}_{1c} \\ \theta_{2c} \\ \dot{\theta}_{2c} \\ x_{2c} \\ \dot{x}_{2c} \end{bmatrix} ; u(t) = [\theta_{1c}] \tag{3.11}$$

3.2.8 Dispenser Model

The dispenser model assumes that the belt can be represented by a linear massless spring. As a belt under compression exhibits negligible force, it is

possible to incorporate both drive and return sides of the belt in a single spring, two springs being used in the model to represent the belt either side of the trolley (Figure 3.11). The linear equations for such a system are given by:

$$\begin{bmatrix} \ddot{x}_{1d} \\ \ddot{\theta}_{2d} \end{bmatrix} = \begin{bmatrix} \frac{-2k_{dis}}{M_d} & \frac{-B_d}{M_d} & \frac{r_d k_{dis}}{M_d} & 0 \\ \frac{r_d k_{dis}}{J_d} & 0 & \frac{-r_d^2 k_{dis}}{J_d} & \frac{-B_{rd}}{J_d} \end{bmatrix} x(t) + \begin{bmatrix} \frac{r_d k_{dis}}{M_d} \\ 0 \end{bmatrix} u(t) \quad (3.12)$$

$$T_l = J_d \ddot{\theta}_{1d} + B_{rd} \dot{\theta}_{1d} + r_d k_{dis}(r_d \theta_{1d} - x_{1d}) \quad (3.13)$$

where:

$$x(t) = \begin{bmatrix} x_{1d} \\ \dot{x}_{1d} \\ \theta_{2d} \\ \dot{\theta}_{2d} \end{bmatrix}; u(t) = [\theta_{1d}] \quad (3.14)$$

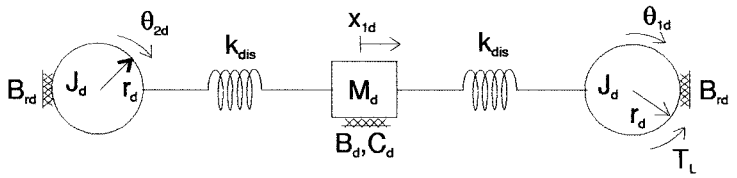

Figure 3.11: Dispenser model implementation

3.2.9 Model Validation

Validation of the time based model has been achieved by comparison with experimentally obtained step response (Figure 3.12). Model parameters have been obtained experimentally and from manufactures data (Appendix A). Some tuning was undertaken until a response to within 5% was obtained. From Figure 3.12 it can be seen that the model and experimental are not in perfect agreement. It would be possible to further improve the performance of the model by introducing nonlinearities such as nonlinear springing into the mechanical models and accurate modelling of the PWM generation in the inverter. However, the increase in complexity and computational time

that would be required for these outweigh the benefits that may be obtained in terms of improved accuracy. Further development may be required if the model proves to be inadequate in the controller design process.

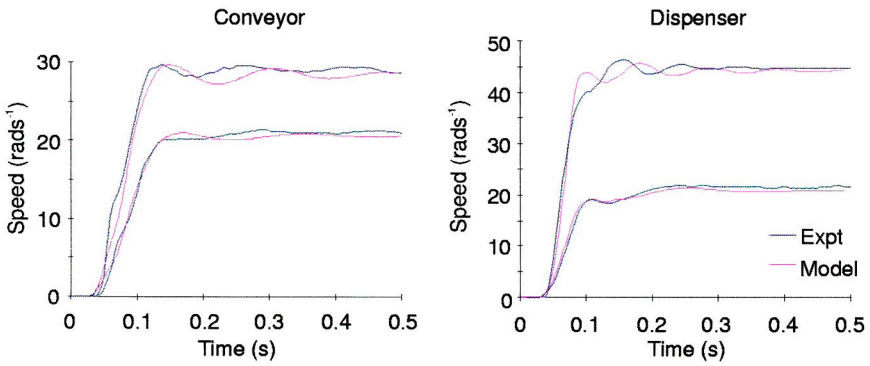

Figure 3.12: Conveyor and dispenser axes step response

3.3 Frequency Response Models

Frequency response analysis provides linear models and information on effects that are not readily apparent from time response methods, such as pure time delays. For the purposes of experimentation the system excitation is a sinusoidal voltage, with speed at the relevant shaft as the output.

3.3.1 Induction Motor and Drive Frequency Response

The frequency response of the drive system with a sinusoidal excitation of 1V amplitude and 5V d.c. offset is linear to around 30Hz. Above this, the motor shaft frequency falls as the demand frequency increases suggesting a sampler at around 60Hz on the inverter input (Figure 3.13). A sampler placed on the input of the theoretical model produces excessive oscillation of the motor shaft. Repositioning the sampler on the frequency demand output from the inverter control block restores a satisfactory step response.

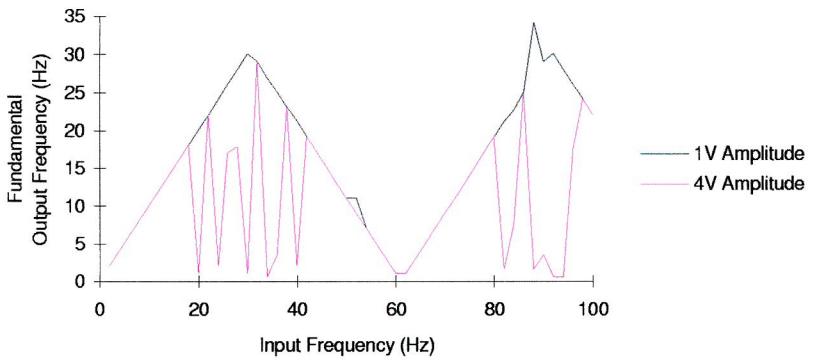

Figure 3.13: Variation of motor shaft frequency with demand frequency

Increasing the excitation amplitude introduces a frequency quenching effect into the response (Figure 3.13) particularly noticeable around 30Hz. To generate sinusoidal motion at the motor shaft the output of the inverter is required to be both a frequency modulated (FM) and an amplitude modulated (AM) signal. For an ideal VVVF inverter the output signal will have the form:

$$v(t) = \alpha x(t) \sin(\beta x(t) t) \tag{3.15}$$

where $x(t) = \sin(\omega t)$ is the inverter input and α and β are constants to provide the correct relationship between input voltage, output voltage and output frequency. Figure 3.14 shows the voltage spectrum for a sinusoidal input with amplitude 4V and frequency 30Hz, with a d.c. offset of 5V. A demand of 10V corresponds to full speed at 50Hz, the 5V offset therefore demands half speed which produces the component at 25Hz. The main peaks in the spectrum are separated by a frequency equal to that of the demand frequency. If the inverter is incapable of producing the higher frequencies, or producing a demand with the correct frequency spectrum, then the output will not be as required. If attenuation was the only effect then the output for 4V peak demand would be expected to be a sinusoid of the correct frequency and speed offset but of reduced amplitude. Instead the frequency at 25Hz is found to dominate causing the output to be at a constant speed. As the

demand amplitude is increased, the magnitude of the frequency components other than the offset increase and the inverter becomes less able to create the full range of required frequencies.

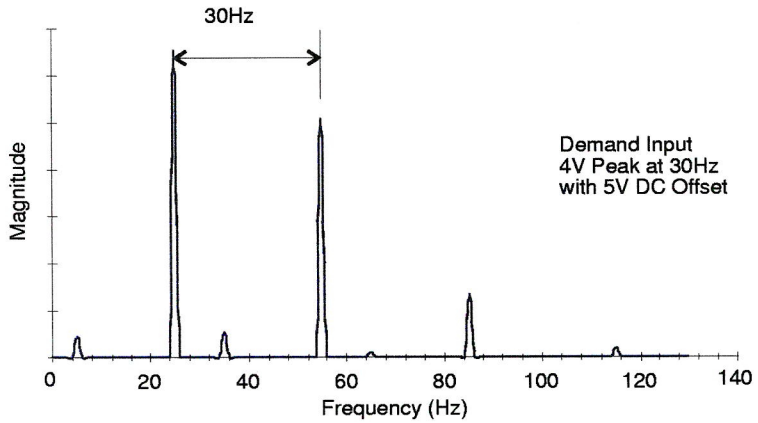

Figure 3.14: Ideal VVVF inverter voltage spectrum

3.3.2 Frequency Response of Mechanical Sub-Systems

A d.c. servo has been used to excite the mechanical components of each axis. By recording the speed of the motor and output shafts and subtracting the two frequency responses obtained, the frequency response of the mechanical components was derived.

Conveyor Frequency Response

To observe the effect of non-linear friction on the frequency response of the conveyor, the experiment was performed twice. First with zero d.c. offset in the input excitation and then again with sufficient d.c. offset to ensure that the conveyor velocity never fell to zero. As expected, the results indicate that when the velocity does not pass through zero the system exhibits less damping (Figure 3.15). The transfer function obtained from both sets of

experiments are:

$$G_{con}(s) = \frac{20.3(s+200)e^{-3.4\times10^{-3}s}}{s^2+100s+100^2} \tag{3.16}$$

$$G_{con(offset)}(s) = \frac{46(s+170)e^{-3.6\times10^{-3}s}}{s^2+84s+140^2} \tag{3.17}$$

The pure time delay apparent from the frequency response is caused by gaps in the chain links. As the system is excited the chain links start moving consecutively, creating a time delay between input and output shafts.

3.3.3 Dispenser Frequency Response

As for the conveyor, a time delay was observed from the frequency response for the dispenser. The dispenser is also represented by a second order model:

$$G_{dis}(s) = \frac{126\times10^3 e^{-2\times10^{-3}s}}{s^2+440s+550^2} \tag{3.18}$$

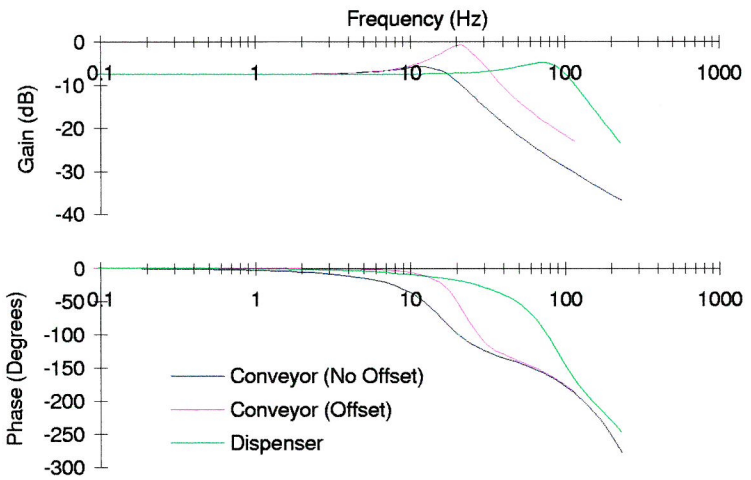

Figure 3.15: Conveyor and dispenser frequency responses

3.3.4 Linear Model

For controller design purposes a linear model of the complete system is required. This has been achieved by using the PC as a signal generator and

determining the frequency response of the complete system up to 30Hz (Figure 3.16). For the conveyor, a fourth order model was obtained (Equation 3.19) with a third order approximation being seen to be sufficient for the dispenser (Equation 3.20).

$$G_{conveyor}(s) = \frac{615.06 \times 10^6}{(s^2 + 49s + 35^2)(s^2 + 54s + 180^2)} \tag{3.19}$$

$$G_{dispenser}(s) = \frac{6.47 \times 10^6}{(s + 35)(s^2 + 99s + 110^2)} \tag{3.20}$$

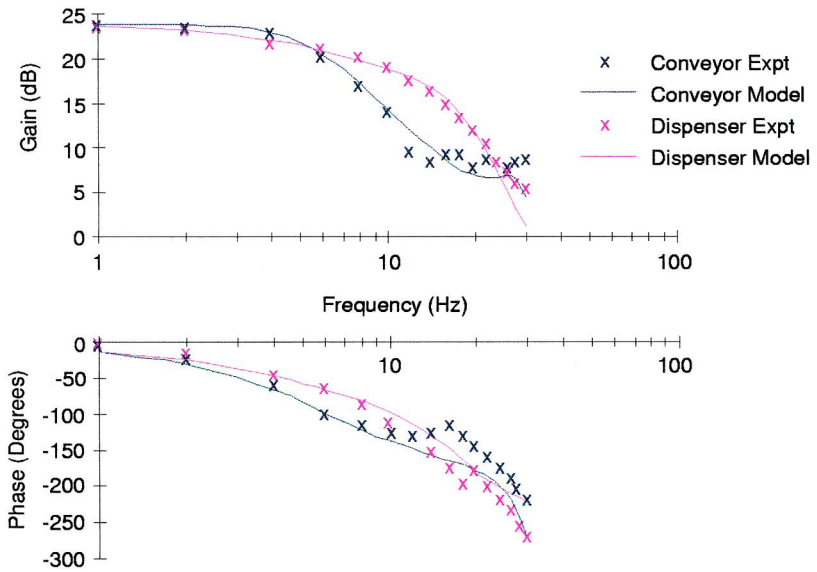

Figure 3.16: Complete frequency responses for both axes

When the step response of the linear models was compared with experimental data (Figure 3.17), the steady state gain was found to be incorrect and the response excessively fast. If the delay in the measured response was due to a pure time delay of the magnitude indicated in the step response, then the phase lag would rapidly increase as the frequency rose, while the gain would not be effected. As the experimental and model low frequency phase characteristics are in good agreement, a pure time delay does not explain the discrepancy. A discontinuous non-linearity such as a deadzone

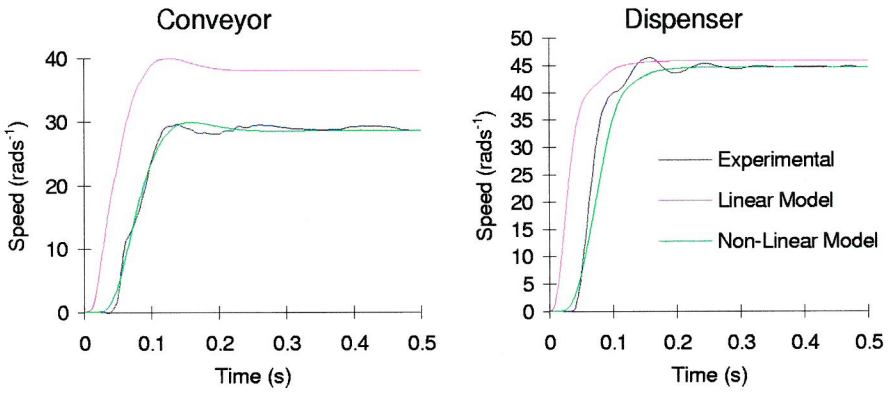

Figure 3.17: Linear and nonlinear step responses

would cause the observed steady state effect, but not the delay. Discontinuous non-linearities are difficult to analyse in the transient state. The most common method of evaluation is describing functions, using a method based on Fourier analysis. Tables of describing functions[29, 52] present a deadzone as:

$$N(j\omega) = \begin{cases} 0 & : \quad E_m \leq E_d/2 \\ \frac{2K}{\pi}\left[\frac{\pi}{2} - \sin^{-1}\frac{E_d}{2E_m} - \frac{E_d}{2E_m}\sqrt{1-\frac{E_d}{2E_m}^2}\right] & : \quad E_m > E_d/2 \end{cases} \tag{3.21}$$

where E_m is the magnitude of the input and the deadzone has the form of Figure 3.18. As can be seen from Equation 3.21 the deadzone does not effect the phase of the system, and is in effect, a gain whose value is dependent on the input.

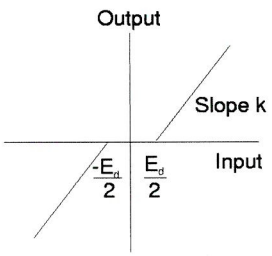

Figure 3.18: Deadzone characteristic

Though the deadzone may account for the gain error, it does not account

for the delay, which is caused by the combination of the deadzone with a slew rate limit. The effect of adding a slew rate limit and a deadzone to the input of the linear model can be seen in Figure 3.17, where the step response of what is now a nonlinear model approximates more closely to that of the experimental.

The physical existence of the non-linearities has been confirmed in previously described models, both occurring within the inverter. The slew rate is a user adjustable parameter, while the deadzone can be seen in the demand/voltage characteristic (Figure 3.7), where an input voltage is required to reach a threshold of 0.12V before obtaining a frequency output.

This analysis therefore provides three linear models, a model with accurate frequency response and a simple nonlinear model with accurate step response. The third model is acquired if the steady state gain of the linear model is reduced such that the model and experimental steady state gains have the same value.

3.4 Summary

A theoretical model of the system has been obtained and verified against step response data. This model suffers from a poor frequency response and excessive computational overheads making design of controllers difficult. To overcome this, linear models have been developed, the most important being that with a correct frequency response. This has a short simulation time and is therefore well suited to use in design. Once controllers have been designed they can be verified with the theoretical model before practical implementation.

Chapter 4

PID Based Control

4.1 Introduction

For the last 50 years the control method of choice for many industrial application has been proportional, integral and differential (PID) control. There are a number of reasons for this longevity. Firstly, it is extremely simple to implement in both continuous and discrete time. Secondly, selection of the gain values can be achieved by a trial and error approach, and there are several preferred methods of tuning the controllers. Research has been ongoing for the past fifty years to derive methods of accurately tuning PID controllers with the most well known being that derived empirically by Zeigler and Nichols[53]. Recent trends have been towards online automatic tuning[54, 55, 56, 57], though these seem to be mainly concerned with regulators rather than tracking controllers.

4.2 Performance Index

In order to be able to compare the performance of practical controllers, it has been necessary to define a series of experiments and evaluate the results using a performance index. For all controller experiments, each axis was

started from the same point and continuously run for 40 units, the position of the last 20 units being recorded at 100Hz and averaged. Each test was analysed to provide a measure of the mean square error (MSE) for the entire profile, MSE during the dispensing time (MSE(D_t)) and the settling time. Settling time is measured from the start of the dispensing time to the point where the system reaches and maintains the position specification.

To obtain a measure of the performance of the controller, each measured parameter is averaged across a range of unit rates, 30-60UPM. The values for the three average parameters are then combined using a cost function to obtain a single value indicating overall controller performance. The cost function, J, is defined as:

$$J = \alpha_1 \bar{e_t^2} + \alpha_2 T_{st} + \alpha_3 \bar{e_{dt}^2} \tag{4.1}$$

where $\bar{e_t^2}$, T_{st} and $\bar{e_{dt}^2}$ are the MSE for the total profile, the MSE during the dispensing time and the settling time respectively. The weights, α_1, α_2 and α_3 are 1, 2 and 3 respectively, though the MSE during the dispensing time had previously been multiplied by 1000 to bring it to the same order of magnitude as the other variables. It is considered essential for the controller to bring the system to within specification during the dispensing time, hence the highest weighting. A short settling time is also highly desirable, while the accuracy over the entire profile is more a consequence of good results for the previous two measures. Values for the weighting parameters that are greater then those indicated tended place too much emphasis on a single parameter, swamping the effect of other parameters. Smaller weighting did not adequately emphasise those parameters that are considered important. Results obtained were normalised such that the results for the PID controller were 1.

4.3 Classical PID Control

The general form of the PID controller is:

$$\begin{aligned} u_c(t) &= K_p e + \frac{K_i}{T_i} \int e\, dt + K_d T_d \frac{de}{dt} \\ &= u_p + u_i + u_d \end{aligned} \tag{4.2}$$

Converting to the z-domain and expanding provides the controller positional algorithm[27]:

$$\begin{aligned} u_p(i) &= K_p e(i) \\ u_i(i) &= u_i(i-1) + \frac{K_i T_s}{2T_i}\left[e(i) + e(i-1)\right] \\ u_d(i) &= \frac{K_d T_d}{T_s}\left[e(i) - e(i-1)\right] \end{aligned} \tag{4.3}$$

where K_p, K_i and K_d are the proportional, integral and derivative gains and T_s, T_i, and T_d are the sampling, integral and derivative times.

For the proportional and integral sections of the controller, the sample frequency is 2.5kHz, but this has been reduced to 500Hz for the differential. At high sample frequencies the position feedback signal may not change from one sample to the next, giving a derivative action that is often zero. This effect is minimised by reducing the differential sampling time. Frequency response analysis of the linear model indicates that the slower differential term has a slightly higher low frequency gain and smaller bandwidth. While the bandwidth is still sufficient for the demand profile, the increase in low frequency gain may improve the steady state response. Simulation with the full model provides a performance index of 1 and 3.9 for the 500Hz and 2.5kHz differential systems respectively. The degradation is caused by considerable overshoot of the conveyor suggesting that the 2.5kHz system is less damped than the 500Hz version.

4.3.1 PID Controller Tuning

To ensure that the controller is adequately tuned, the Zeigler-Nichols closed loop tuning algorithm has been employed[53]. This method entails operating the system with proportional control only and increasing the gain until the

output exhibits constant amplitude oscillations. The gain and period at this point are used to determine the PID controller gains. If:

$$K_I = \frac{K_i T_s}{T_i} \quad \text{and} \quad K_D = \frac{K_d T_d}{T_s} \tag{4.4}$$

then from Zeigler-Nichols the gain equations are:

$$k_p = 0.6K_u; \quad K_I = \frac{0.6K_u T_s}{T_u} \quad K_D = \frac{0.6K_u T_u}{8T_{sd}} \tag{4.5}$$

where K_u and T_u are the Ultimate Gain and Ultimate Period.

Gain values obtained in this manner produced a response that was not satisfactory. When indexing the conveyor failed to remain stationary during the dispensing time but dithered as the integral value ramped back and forth through the inverter deadzone. Reduction of the integral gain brought the conveyor to rest, but the steady state error increased with each unit due to integral windup. It is therefore necessary to reset the integral sum upon completion of each unit. However, under synchronisation, the original integral gain allows the conveyor to reach and maintain the desired position rapidly and accurately.

Gain values for the dispenser axis suggested by Zeigler-Nichols tuning were found to be satisfactory. However, the value of K_I was reduced at unit rates in excess of 50UPM to maintain a satisfactory response. So doing reduced the gain between 10 and 30rads^{-1} and increased the gain margin hence improving stability. Gain values for all controllers are reproduced in Appendix B. Numerical results for the performance of each controller are presented in Appendix C.

4.3.2 PID Results

The mean square error for the conveyor output shaft fluctuates more than the same measurement at the motor shaft (Figure 4.1). As position feedback is only available from the motor shaft, the conveyor is effectively operated

in open loop. Though the performance of the system under PID control is adequate, there is a definite decrease in performance as the unit rate is increased with the MSE during the dispensing time increasing by a factor of 30. The settling time also increases from below 0.01s to above 0.03s.

The settling time at 30UPM indicates that the conveyor never achieves the position specification of 53mrad at the motor shaft. The conveyor stops with a position error of 70mrad, while the motor stops at 10mrad. Therefore the motor is well within specification while the conveyor is considerably outside, and this is again a consequence of the conveyor being open loop.

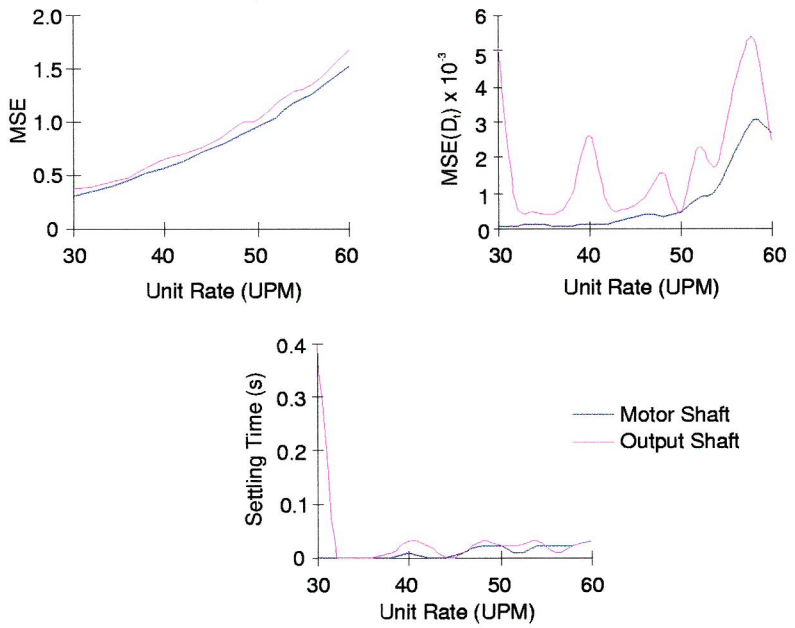

Figure 4.1: Conveyor results under PID control

The dispenser performs far less satisfactorily (Figure 4.2) than the conveyor, though due to the dispenser operating with a nett position demand of 0, the discrepancy between values recorded at the motor and dispenser shaft is reduced. The dispenser performance degrades much more noticeably with unit rate than the conveyor, as indicated by the increase in all measured variables above 50UPM . By 56UPM the settling time indicates

that the dispenser never obtains the performance specification during the dispensing time. Satisfactory performance would appear to be limited to less than 54UPM, though below this the settling time is on average 0.08s, so a value of 0.1s, or 25% of the dispensing time, would be the minimum to ensure correct product dispensing.

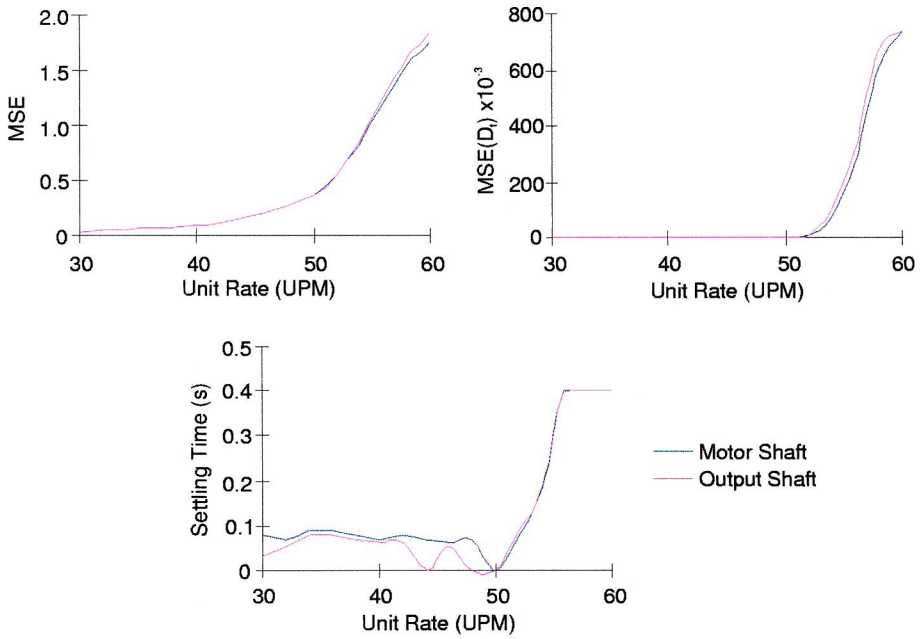

Figure 4.2: Dispenser results under PID control

For the conveyor synchronising, the motor MSE during the dispensing time worst case value, across all unit rates tested, is 1.7×10^{-2}. This is equivalent to a position error of 10mrad, implying that the conveyor is within specification for all unit rates. For tracking a ramp, the PID controller performs well, reducing the error to a minimum quickly and remaining well within limits at all times and for all unit rates.

4.4 Lag/Lead Compensation

The PID controller is a general version of the lag/lead network[27, 58] which has a superior phase response compared to the PID controller. The phase for the lag/lead network commences at 0^0 at zero frequency and returns to 0^0 as the frequency increases (Figure 4.3). A PID controller has a phase that starts at -90^0 and moves to $+90^0$ as the frequency is increased. This can lead to a degradation in stability margins that would adversely affect the performance.

A lag/lead network has been designed by considering the frequency response of the complete system and designing for a phase margin of approximately 30^0 and a gain margin of 7dB. The stability margins are rule of thumb values used in servo design for sometime[59]. Figure 4.3 shows that the frequency response of the compensator so designed is very similar to that of the PID controller. It has recently been shown that a PID controller tuned by Zeigler-Nichols does achieve stability margins of approximately these values[60].

4.4.1 Results

The cost function results obtained are 15 and 2 for the conveyer and dispenser respectively. From the frequency response, it can be seen that the low frequency gain is significantly reduced for the lag/lead compensator compared with the PID. This dramatically affects the steady state error, causing a significant increase in the MSE(D_t) and the settling time. Though through the dynamic section of the demand profiles the lag/lead compensator is only marginally worse than the dispenser, the settling times and MSE(D_t) are more heavily weighted in the cost function. Therefore, these are responsible for the very poor results in comparison with the performance of the PID based controller.

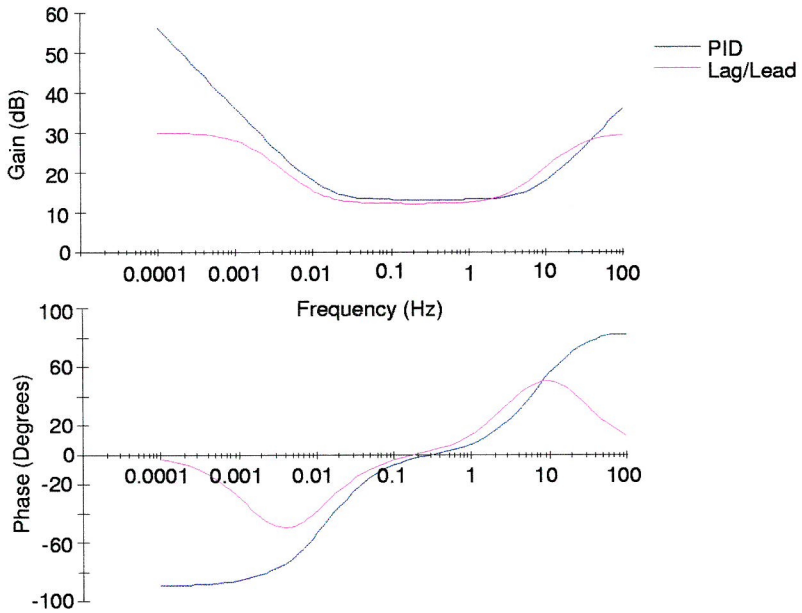

Figure 4.3: PID and lag/lead frequency response

4.5 PID with Velocity Feedforward

A common method of improving the performance of PID controllers in commercial implementations is to add a velocity feedforward term (Figure 4.4). In this method it is realised that the input to the drive is a velocity demand, and to improve the performance a proportion of the actual velocity demand is passed directly to the drive. For a d.c. servo system operating with a velocity control loop in the drive, K_v is equal to 1. The position loop then adjusts the velocity demand to remove the position error.

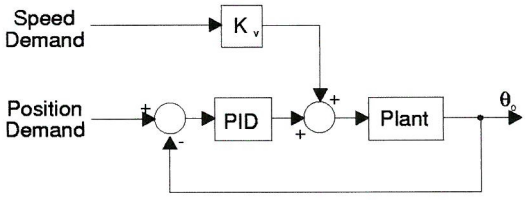

Figure 4.4: PID with velocity feedforward

4.5.1 Results

The conveyor cost function is 1.01 which is almost identical to the PID value, but Table 4.1 shows that the motor performs considerably better than the motor under PID control. Though the motor performs better, the improvement is cancelled by the conveyor mechanics performing considerably worse than for PID control. The velocity feedforward controller feeds the velocity demand straight into the drive, but the velocity demand contains a number of discontinuities that cause sudden changes at the motor. The conveyor is unable to react as quickly, therefore producing a significant error.

Addition of the velocity feedforward term improves the dispenser performance dramatically producing a cost function of 0.29. Only the settling time is slightly degraded, the addition of velocity feedforward introducing oscillation during the dispensing time. The feedforward introduces a zero into the closed loop system, hence causing the system to overshoot. Otherwise, the performance of the dispenser is superior to the PID and the mechanics are not as adversely affected as for the conveyor.

	Conveyor		Dispenser	
Parameter	Feedforward	PID	Feedforward	PID
MSE	0.38	0.80	0.21	0.47
MSE(D_t)	0.59	1.06	31.6	109
Settling Time	0.01	0.01	0.17	0.15

Table 4.1: Average results for the feedforward controller

The performance of both axes degrades at the same rate as for the PID system as the unit rate is increased. This is to be expected as the feedforward term does not address the system dynamics.

4.6 Velocity Estimation

A limitation of the encoder feedback is the poor velocity resolution, the resolution being given by[61]:

$$\omega_{res} = \frac{\theta_{res}}{T_s} \tag{4.6}$$

With a sample frequency of 2.5kHz, the velocity resolution is limited to only 7.9 rads^{-1}. If the velocity resolution could be improved then use could be made of the velocity feedback signal in the controller.

To obtain a measure of motor shaft velocity the signal from the encoder could be filtered, but this will introduce unacceptable lags. As a model of the system exists, it would be possible to use this for velocity estimation, though a more accurate estimation would be made if a state observer was constructed. For a purely deterministic system where the model parameters are well known, a full or reduced order observer would be suitable. As the model is a linear approximation of a non-linear system, a stochastic estimator, such as the Kalman estimator, would be more appropriate.

4.6.1 Kalman Estimator Design

The plant shown in Figure 4.5 has states and outputs corrupted by noise, and is represented by a state space model of the form:

$$\dot{x} = Ax + Bu + Lw \tag{4.7}$$

$$y = Cx + Du + v \tag{4.8}$$

where w and v are random noise signals that corrupt the states and the outputs respectively. The matrix L couples the state disturbances into the states. From Figure 4.5 the state space equation for the estimator is:

$$\dot{\hat{x}} = A\hat{x} + Bu + G(y - C\hat{x} - Du) \tag{4.9}$$

where $\hat{x}$ is the state estimate and G is the Kalman gain matrix that couples the plant outputs to the estimator states. By feeding back the plant output the sensitivity to modelling errors is reduced[62]. Motor torque often provides an additional feedback signal to the estimator[61, 63], but is unavailable for this system.

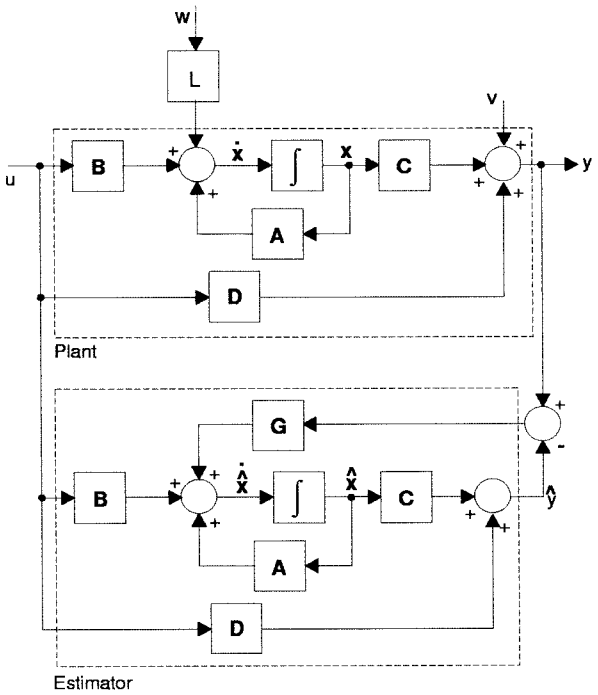

Figure 4.5: Block diagram of system and noise components

The performance of the Kalman estimator is determined by the selection of the values in G. These are selected so as to provide the estimator with dynamics that are considerably faster than the plant, and also minimise the covariance of the estimation error. To obtain G, the covariance values for the system and measurement noise signals must be determined. The analysis then provides four equations that are solved recursively[52]. This can be performed at each sample instant which produces a varying gain matrix but is computationally intensive and can not be achieved in the available programming bandwidth. For this reason a stationary Kalman

filter has been implemented. Here the value of G is determined offline using a computer aided control system design (CACSD) package to solve the gain equations such that the estimator error covariance is minimised.

The quantisation effect of the encoder was found to generate high frequency components in the velocity estimate. This was overcome by inserting a low pass filter between the encoder and the Kalman filter, the breakpoint tuned to minimise the oscillation and measurement lag.

4.6.2 Multiple Loop PID Control with Kalman Estimator

A common method of obtaining position control is to place a position loop around a speed loop (Figure 4.6) with the output of the position loop becoming the input to the speed loop. Both loops contain a PID controller with the inner speed loop designed to execute at 2.5kHz, while the outer position loop executes at 500Hz. Speed feedback for the speed loop is obtained by use of the Kalman estimator. The loops were tuned empirically, firstly the inner loop until a smooth speed response was obtained, and then the position loop was added.

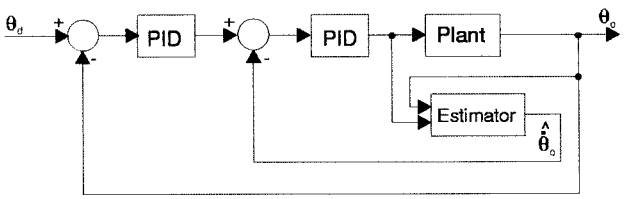

Figure 4.6: Position controller with both speed and position loops

The performance indices obtained for this controller are 1.39 and 0.73 for the conveyor and dispenser respectively, with the performance still rolling off as the unit rate is increased. Though the performance of the dispenser is improved, that for the conveyor is not.

Once tuned, the frequency response of the controllers were analysed with the linear model (Figure 4.7). The bandwidth of the conveyor controller

has been reduced from approximately 40rads^{-1} for the PID controller to 14rads^{-1} making the system slower to respond. The gain between 10 and 40rads^{-1} has also been dramatically reduced, degrading the low frequency response. However, for the dispenser, the bandwidth is only reduced by 3rads^{-1} and the low frequency gain is very slightly improved. Also the gain rolls off more slowly with less phase lag, improving the response at higher frequencies.

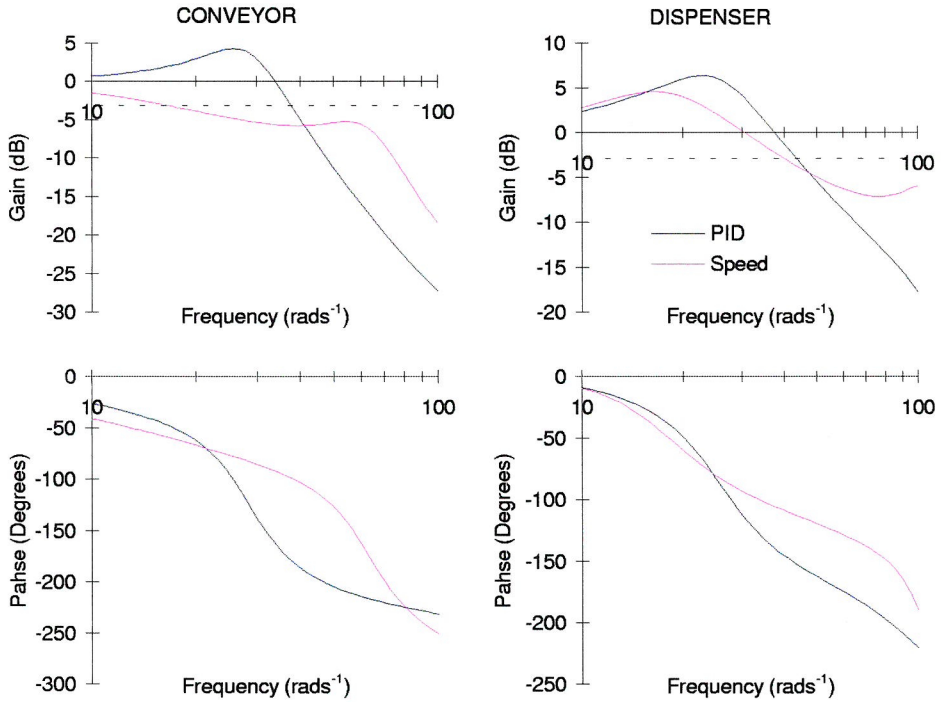

Figure 4.7: Frequency response of single and multi-loop PID controllers

Better tuning of the conveyor was found to be extremely difficult without inducing instability. Though part of the reason for the poor conveyor performance is the difficulty involved in tuning six gains, if the estimator does not provide an accurate velocity measurement then the performance will be compromised. The conveyor operates with a slower velocity than the dispenser therefore producing less encoder counts. This reduced position feedback will make speed estimation more difficult.

4.7 Flux Vector Inverters

The constant voltage to frequency inverters used in the experimental apparatus contain a number of elements that make accurate control difficult. At the time of their design, a.c. drives were intended for situations where some speed variation was desired but accurate position control was not necessary, such applications being the domain of d.c. servo systems. The constant V/f method of control provides a system with adequate steady state characteristics but the transient response is very poor[64]. Modern a.c. drives are intended to be direct competition for the d.c. drives market and it is expected that such a drive should give a performance advantage over the current system. A modern flux vector inverter has been obtained that has two modes of operation (Figure 4.8). In the open loop mode, the inverter is a direct replacement for the existing system, operating with an output that maintains a constant voltage to frequency ratio. If an encoder is connected to the drive, then a speed control loop and independent manipulation of the torque and magnetising components of the current waveform can be implemented. This creates a system that should theoretically be equivalent to a d.c. servo system as seen from the drive input.

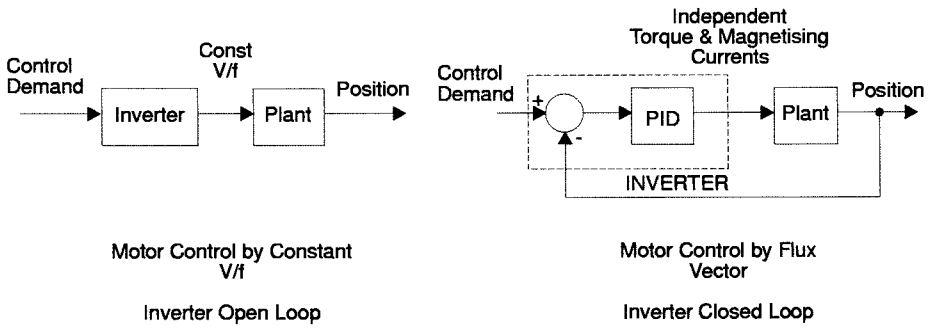

Figure 4.8: Flux vector inverter operational modes

4.7.1 Flux Vector - Inverter Open Loop

The flux vector inverter was tested with the PID controller, with no adjustment to the controller gains. The performance indices are 3.42 and 1.01 for the conveyor and the dispenser respectively.

	Conveyor		Dispenser	
Parameter	Flux Vector	Const V/f	Flux Vector	Const V/f
MSE	0.66	0.81	0.30	0.46
MSE(D_t)	3.22	0.75	107.8	106.9
Settling Time	0.10	0.01	0.11	0.14

Table 4.2: Average results for both inverters in constant V/f mode

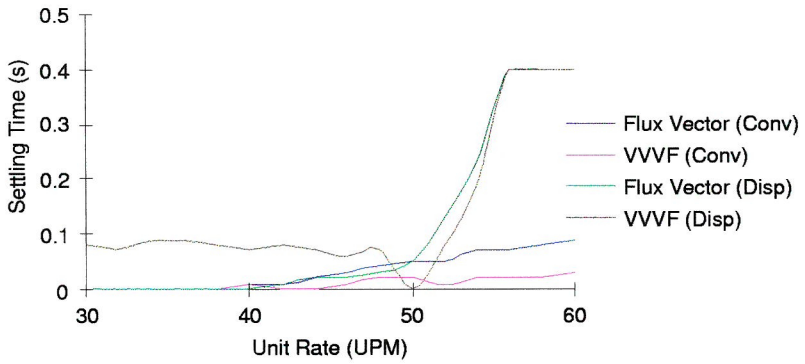

Figure 4.9: Settling times for both inverters in constant V/f mode

As seen from Table 4.2, the conveyor does not perform well during the MSE(D_t), even though the total MSE is better. As there is no deadzone in the new inverter, the forward path gain is effectively increased which causes the conveyor to overshoot at low unit rates. As the unit rate is increased, the conveyor takes longer to settle (Figure 4.9), but no longer overshoots. The output from the controller is effectively a velocity demand, which for open loop velocity control would have the shape shown in Figure 4.10. As the velocity demand falls towards zero, the constant V/f inverter enters the deadzone causing the inverter output to fall to zero, stopping the motor. For the flux vector inverter there is no deadzone, hence the motor is brought to

a controlled stop which, at higher unit rates, takes longer than just setting the inverter output to zero. The final position error may be superior to that of PID, but the lethargy with which it is achieved produces a very poor MSE(D_t) and settling time. In this instance, the presence of the deadzone is a benefit.

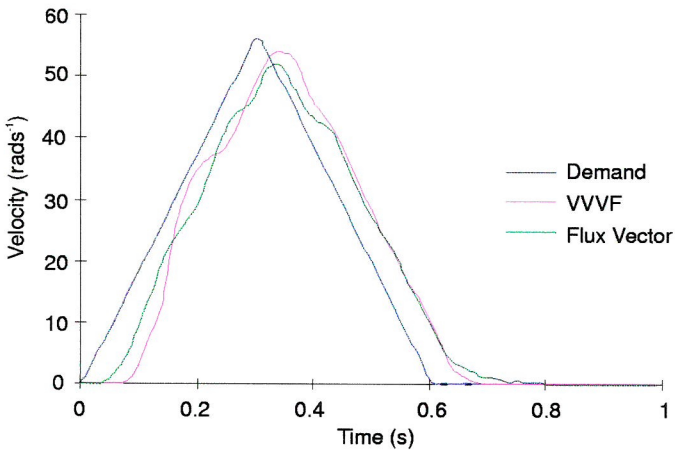

Figure 4.10: Velocity response for both inverters with constant V/f

4.7.2 Flux Vector - Inverter Closed Loop

To convert the modern drive from constant V/f operation to closed loop flux vector, the motor encoder was connected to the inverter. The inverter speed loop was then tuned by adjustment of the PID gains before the position loop in the PC was closed and the gains associated with that controller tuned.

In an induction motor, the flux and torque producing currents are coupled, and any change to the supply current will affect both magnetising and torque producing components. By maintaining the V/f ratio constant, the magnetic circuit is prevented from saturating and the torque/speed characteristic is linearised[64]. This provides adequate steady state but poor transient performance. Flux vector inverters aim to overcome this by decoupling the current components which is achieved by modelling the motor

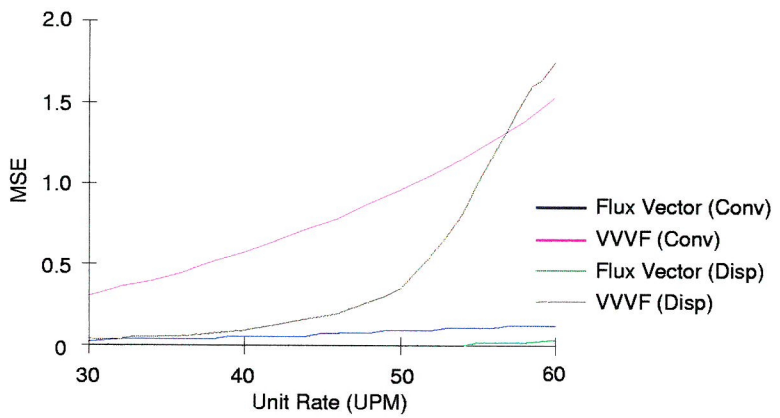

Figure 4.11: Flux vector and constant V/f inverter MSE results

as a two phase system, one phase for each current component. Such a motor model has been used in the time base model. The precise requirement for each current component can then be determined to achieve the control objective[35, 40, 65, 66]. The result is the required 2 phase current which is converted back to 3 phase and generated at the inverter terminals by altering the switching pattern of the power devices. Hence, this form of control is only available at component level. By decoupling the magnetising and torque producing components of the current, the motor and drive effectively form a separately excited d.c. system.

The result of the computation is a considerable improvement in performance which can be clearly seen in Figure 4.11, though the performance still degrades as the unit rate is increased. The closed loop flux vector inverter achieves performance indices of 0.11 and 0.004 for the conveyor and dispenser respectively. Analysis of the results show that the closed loop flux vector inverter performs well in excess of the VVVF inverter operated with a speed loop, implying that the main reason for the improvement is the use of the current decoupling rather than the speed controller.

4.8 Summary

The performance of the PID controller has been shown to require improvement, especially as the unit rate is increased. Different implementations of the basic PID structure have been investigated. The addition of a velocity feedforward term produces a significant performance improvement but the performance continues to degrade as the unit rate is increased. It is also apparent that the mechanical load is open loop, and though the motor may be within specification, the load may not be. The performance of the conveyor appears to be worse than that of the dispenser, even though the conveyor follows the simpler demand. Once the conveyor velocity has fallen to zero no further improvement is possible, whereas the dispenser error can be reduced throughout the profile.

Comparison of the original inverter with a modern constant V/f inverter has been shown to provide some improvement. Conversion of the modern drive to closed loop mode utilising flux vector current decoupling, dramatically improves performance but requires tuning of up to six gains. This method can not be retro-fitted to existing systems without replacement of components, and also has an increase in complexity for the commissioning engineer.

Chapter 5

Model Based Controllers

5.1 Introduction

A generic controller such as PID endeavours to control the response of a system without directly addressing the system's dynamics. This may lead to an adequate response, but the performance will be improved if the controller incorporated knowledge of the plant dynamics in the design. In Chapter 3, three models were described, a full non-linear model, a linear model and a hybrid. The full non-linear model can immediately be excluded for this application as it requires considerable simulation time, and will not operate in a real time environment. The hybrid model contains non-linearities in the form of slew rate limits and deadzones. While the acceleration limits can be easily implemented, modelling of the deadzone requires a discontinuous equation. This will then require a system that switches between controllers designed with a model based on each section of the discontinuous equation. Consequently the linear model is most suitable and will ensure that computation is possible within the available programming bandwidth.

Today, the majority of control research revolves around model based controllers. Though there are many variations they fall broadly into two categories, state feedback controllers and those designed in the digital domain.

State feedback realises that all but the simplest of systems contain more than one state. By placing a feedback gain on each of these states, summing their outputs and then subtracting from the demand, the desired closed loop response is obtained. Theoretically, the closed loop poles could be located anywhere in the s-plane by suitable selection of the gain values[67].

Direct digital controller design attempts to use the advantages of implementation in a digital environment. As the controller only exists as a mathematical entity within the processor, the design is not constrained by the need to be implemented on linear physical components. As such, the controller can be designed to give just the required form of compensation. More advanced tracking controllers can be designed that use feedforward elements[68] and knowledge of the shape of the desired output[69]. Though a controller designed in the digital domain is not limited by the need to be physically realisable in hardware, they must still satisfy the requirements of causality and stability. In addition, the limitations of the control actuator must be considered.

5.2 Optimal Control

State variable feedback (SVF) uses system states as additional feedback elements in the controller (Figure 5.1). SVF design is performed by specifying the desired closed loop pole locations and then solving the characteristic equation to position $m \times n$ gains, where n is the number of states and m the number of inputs.

If the system is completely state controllable, then it is possible to find suitable values for the gain matrix, K, so as to place the closed loop poles at any location in the left hand plane (LHP) of the s-domain[52, 62]. In practice, the location of the poles is limited by the necessity of avoiding excessive control effort. SVF control requires measurement of all the state variables. This may not be practically possible for engineering reasons, or

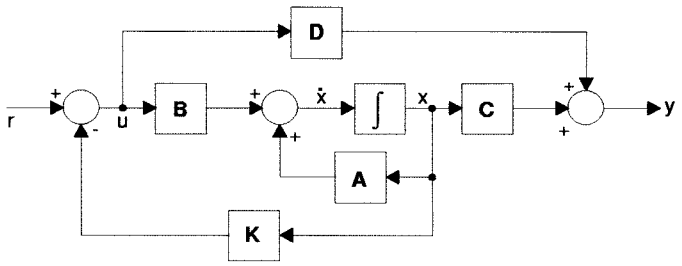

Figure 5.1: SVF regulator structure

the states may not represent any physical property. Under these conditions either a model of the system or a state observer must be implemented as part of the controller. The system must therefore be completely state observable as well as state controllable for SVF control to be suitable.

From Figure 5.1 it can be seen that the equation describing the SVF controller is:

$$\begin{aligned} \dot{x} &= [A - BK]\,x + Br \\ y &= [C - DK]\,x + Dr \end{aligned} \tag{5.1}$$

Selection of suitable closed loop pole locations for high order systems and systems with multiple inputs can be difficult. Optimal control theory isolates the designer from the necessity of selecting pole locations. The gain matrix is automatically calculated by determining values that minimise the cost function:

$$J = \int_0^\infty \left[x^T Q x + u^T R u\right] dt \tag{5.2}$$

By selecting appropriate values for the matrices Q and R, emphasis can be placed on either minimising the states or minimising the control effort. The values of Q and R are adjusted until a suitable response is obtained. Selection of values for the weighting matrices must ensure that Q is positive semidefinite and R is positive definite. In practice, Q and R are normally diagonal, with leading diagonal elements of Q being non-negative and those of R being positive.

5.2.1 SVF Design

Optimal control theory produces regulating controllers with the structure shown in Figure 5.1. Regulators are not suitable for this problem as it is necessary for the output to track the input. However, a tracking problem can be considered as regulation of the error. Conversion of the problem to the error space[62] would then require a regulating controller and allow use of standard optimal control theory.

Error Space Transformation

The model of a single axis is represented by:

$$\begin{aligned} \dot{z} &= Fz + Gu \\ y &= Hz \end{aligned} \tag{5.3}$$

and the input is of the form:

$$\dddot{r} = 0 \tag{5.4}$$

The input could be defined as a more complicated waveform, though this increases the complexity in the final controller structure. The error is defined to be:

$$e = y - r \tag{5.5}$$

Differentiating twice and substituting for $\dddot{r}$ and $\dddot{y}$ gives:

$$\ddot{e} = H\ddot{z} \tag{5.6}$$

Defining the error space vector to be:

$$\xi = \ddot{z} \tag{5.7}$$

and the control vector in the error space as:

$$\mu = \ddot{u} \tag{5.8}$$

then the state equation in error space is:

$$\dot{\xi} = F\xi + G\mu \tag{5.9}$$

This can be written in the normal state variable form of:

$$\begin{aligned} \dot{x} &= Ax + B\mu \\ y &= Cx \end{aligned} \tag{5.10}$$

if:

$$x = \begin{bmatrix} e \\ \dot{e} \\ \xi \end{bmatrix}; \; A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & H \\ 0 & 0 & F \end{bmatrix}; \; B = \begin{bmatrix} 0 \\ 0 \\ G \end{bmatrix}; \; C = H \tag{5.11}$$

The control effort, μ, is defined to be:

$$\mu = -\begin{bmatrix} K_1 & K_2 & K_0 \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \\ \xi \end{bmatrix} \tag{5.12}$$

The controller structure can be determined by substituting for ξ in Equation 5.12 and expanding to give:

$$u = -K_1 \int_0^t \int_0^t e\,dt\,dt - K_2 \int_0^t e\,dt - K_0 z \tag{5.13}$$

The desired controller therefore has an integral arrangement (Figure 5.2), commonly referred to as an internal model structure.

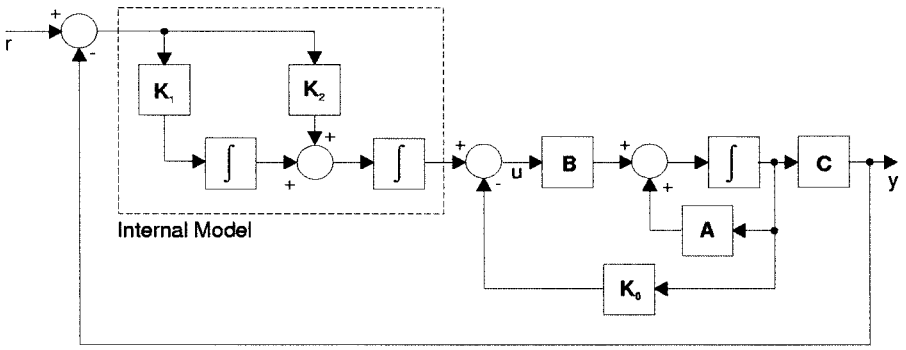

Figure 5.2: Tracking SVF block diagram

Simulation is then used to determine the most suitable values of Q and R in Equation 5.2. With R set to 1 a satisfactory response is obtained with Q as:

$$Q_{11} = 1 \times 10^9; \quad Q_{22} = 1 \times 10^7 \tag{5.14}$$

for the conveyor and

$$Q_{11} = 1 \times 10^5; \quad Q_{22} = 1 \times 10^8 \tag{5.15}$$

for the dispenser, all other elements in Q being zero. As the states involved are e and $\dot{e}$ then variation of Q_{11} and Q_{22} effectively allows control of proportional and damping gains.

5.2.2 Results

To implement the optimal controller, a method of measuring the system states is required. The Kalman filter designed in Chapter 4 was first assessed but the simulation suggested that it is unstable when used in a closed loop configuration. High frequency components introduced by the encoder are especially noticeable on some states, even with filtering. If this state is associated with a high gain in K_0 then the system can become unstable. For this reason a model of the plant rather than an observer was implemented (Figure 5.3). The two state space models of Figure 5.3 could be combined into a single regulating structure. The chosen method of implementation has the advantage that access to the gains K_1 and K_2 is possible. This allows for some tuning of the controller to compensate for discrepancies between the model and plant.

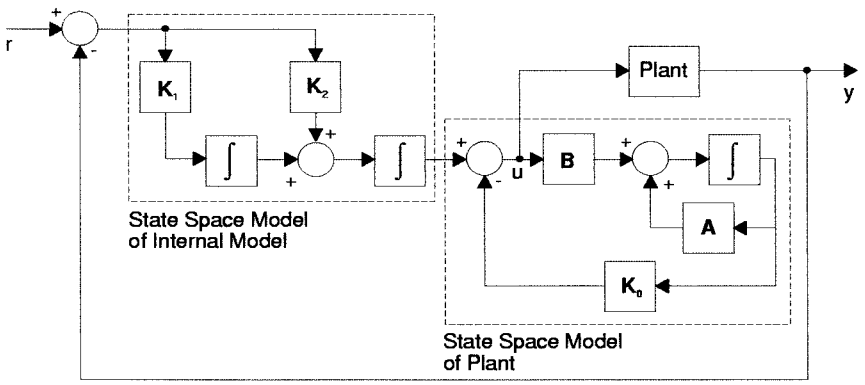

Figure 5.3: Implementation of tracking controller

Experimentation produced cost function results of 0.42 and 0.62 for the conveyor and the dispenser respectively, and the average results are displayed in Table 5.1. The optimal controller performs considerably better than PID except for the dispenser settling time. The controller is slightly worse at low unit rates, but a small increase is less significant here when the settling time is small compared with the time to complete one unit. At higher unit rates the settling times for PID and the optimal controller are the same (Figure 5.4), but the superior position performance of the optimal controller produces the excellent cost function. The performance of the controller on both axes degrades as the unit rate is increased, but not as quickly as for PID.

	Conveyor		Dispenser	
Parameter	Optimal	PID	Optimal	PID
MSE	0.45	0.81	0.34	0.46
MSE(D_t)	0.29	0.75	69	107
Settling Time	0.01	0.01	0.17	0.14

Table 5.1: Motor average results for Optimal and PID controllers

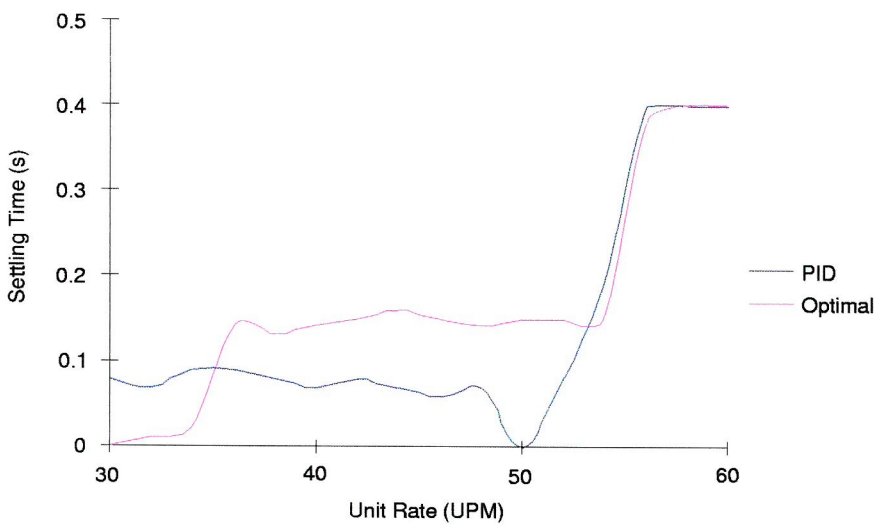

Figure 5.4: Dispenser optimal and PID settling times

For comparison the conveyor was also tested with an optimal controller

designed for a constant input, $\dot{r} = 0$ in Equation 5.4. This achieved a cost function of 0.69, again improving on PID but not matching the result for the ramp input optimal controller. The constant input controller is penalised for a poorer MSE(D_t) and settling time caused by slightly worse tracking. As the system is only designed to have zero error to a constant input, worse tracking of the indexing profile is to be expected.

5.3 Digital Controllers

There are a number of methods for designing controllers directly in the z-domain[27, 52, 70, 71]. Kalman's method attempts to force the plant to settle in the minimum number of samples, the theoretical minimum being equal to the order of the plant. Deadbeat response follows an input exactly at the sample instants with one sample delay, and Dahlin's method produces a closed loop response that is characterised by a first order lag and a pure time delay. None of the above methods are considered suitable for this application. The deadbeat and Kalman's method are too severe for this purpose and may lead to excessive controller effort due to the very short sample time. Dahlin's method does not deal well with ringing poles or zeros outside the unit circle. This can lead to a response that is far from first order. The method chosen is to use a direct digital design method where the required closed loop response is specified. This does have the disadvantage of producing high order controllers but allows more flexibility in design.

5.3.1 Direct Digital Design

If the plant is represented by the discrete time transfer function $G_p(z)$ then the direct digital design method produces a controller of the form:

$$G_c(z) = \frac{1}{G_p(z)} \frac{F(z)}{1 - F(z)} \tag{5.16}$$

where $F(z)$ is the desired closed loop transfer function. Design is performed by specifying the steady state accuracy, rise time and the velocity error constant.

For zero steady state error, the steady state gain of the closed loop system must have a value of 1. This implies that:

$$F(z)|_{z=1} = 1 \tag{5.17}$$

It can be shown[27] that the velocity error requirement can be fulfilled if:

$$\frac{1}{T_s K_v} = -\left.\frac{dF(Z)}{dz}\right|_{z=1} \tag{5.18}$$

Initially $F(z)$ is specified to be:

$$F(z) = \frac{b_0 + b_1 z}{z - \tau} \tag{5.19}$$

This provides two degrees of freedom which can be solved to satisfy the design criteria of Equations 5.17 and 5.18.

5.3.2 Selection of $F(z)$

To satisfy the causality constraint, $F(z)$ must have at least the same pole zero excess as the plant. In discrete time systems pole zero excess is a measure of the transport delay, so reduction of the poles zero excess would require the controller to anticipate changes in the demand signal. If a zero order hold is used to transform the continuous time linear model then the pole zero excess for both axes is 1. Therefore an extra pole must be added to $F(z)$. The controller must also satisfy the stability constraint. This requires that $F(z)$ must incorporate as zeros, any zeros of the plant which lie outside the unit circle. The required transfer functions for $F(z)$ are therefore:

$$\begin{aligned} F_{conv}(z) &= \frac{(b_0 + b_1 z)(z - c_1)(z - c_2)}{z^3(z - \tau)} \\ F_{disp}(z) &= \frac{(b_0 + b_1 z)(z + d_1)}{z^2(z - \tau)} \end{aligned} \tag{5.20}$$

where c_1, c_2, and d_1 are the zeros outside the unit circle. Applying the steady state and velocity error requirements results in a set of simultaneous equations that are then solved for b_0 and b_1.

Simulation showed that the control effort saturated on the supply rails at specific points (Figure 5.5). The design procedure inverts the plant and as the plant contains an integrator, the compensator contains a differentiator. The discontinuities caused by instantaneous changes in acceleration in the demand signal and quantisation caused by the encoder are responsible for the higher frequencies. These are differentiated and cause the sudden and dramatic increase in control effort. This could be minimised by smoothing out the acceleration profile which can not be achieved by filtering of the demand signal as this alters the location of critical points in the profile. Preliminary investigations suggest that to obtain a reasonable control effort, the differential of acceleration has to be specified as consisting of a series of ramps. It is then necessary to integrate repeatedly to obtain the required position profile. Implementation of such a demand would make direct comparison between this controller and PID difficult. Therefore, the large control effort excursions were removed by placing low pass filters on the encoder feedback signal and the output from the controller. The design then balances the performance of the controller with acceptable lags for the filters.

The encoder in the plant provides an integrator in the forward path suggesting that the controller does not require an integrator to remove steady state error. This is not so as the encoder is placed beyond the point where disturbances enter the system, hence the controller adds an integrator. Simulation shows that this produces an acceptable error for the conveyor during the dispensing time, but not for the dispenser. Both systems are effectively Type 1, but the dispenser requires zero error to a ramp. This is normally achieved by adding a second forward path integrator to create a Type 2

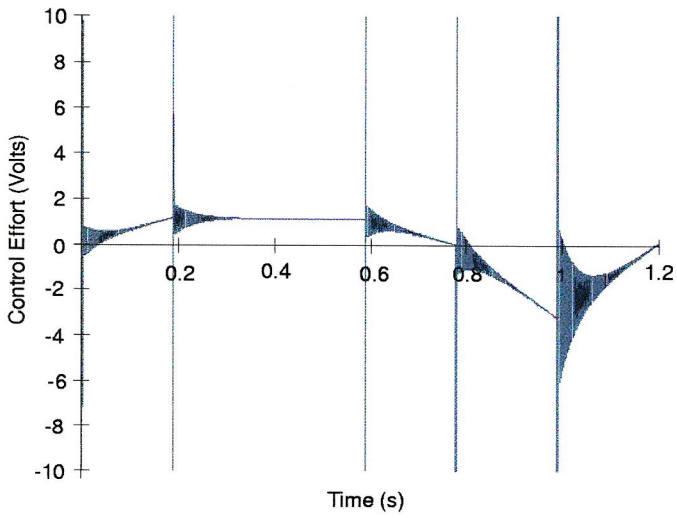

Figure 5.5: Dispenser control effort for the linear model.

system. The PID controller is only Type 1 but achieves a small error to a ramp by having a large value of integral gain. In this case the velocity error is not zero but small enough to satisfy the specification at most unit rates. Such an effect could be achieved by increasing the K_v term but this leads to an unacceptable control effort. Increasing K_v reduces the value of b_1, which moves the zero closer to the unit circle making the system more sensitive to high frequencies. Improvement of the direct digital controller is therefore achieved by adding a gain and integral term in parallel with the compensator (Figure 5.6). This reduces the error to a ramp, but also provides a degree of tuning by which modelling errors can be compensated for.

5.3.3 Results

The controller and associated filters have been implemented as state space models with the compensating integral gain tuned to produce the best response. Cost functions obtained for the direct digital controller are 0.57

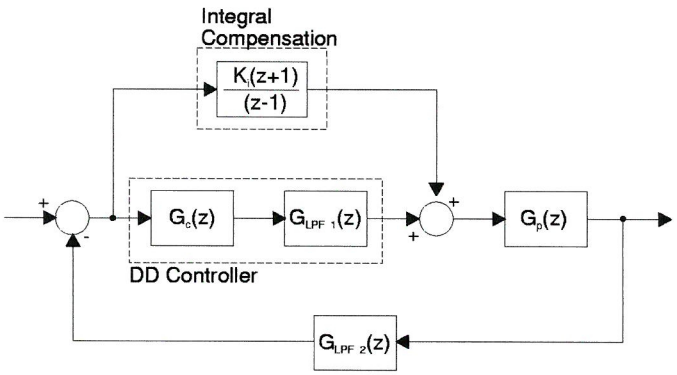

Figure 5.6: Direct digital plus integral controller block diagram.

and 0.02 for the conveyor and the dispenser respectively. The reason for the dramatic improvement in dispenser response is due to a significant improvement in the MSE(D_t). The average results for this parameter are 1.71 and 107 for the Direct Digital and PID controllers respectively. Though at low unit rates the values are similar, above 52 UPM the PID values rise by around a factor of 300 at 60UPM (Figure 5.7). The direct digital controller only increases by a factor of 3 between 52 and 60UPM, this controller being far more consistent across the unit range. This is also true of the conveyor, with very little increase in parameter values as the unit rate is increased. The use of knowledge of the dynamics in the controller design produce the considerably more consistent performance.

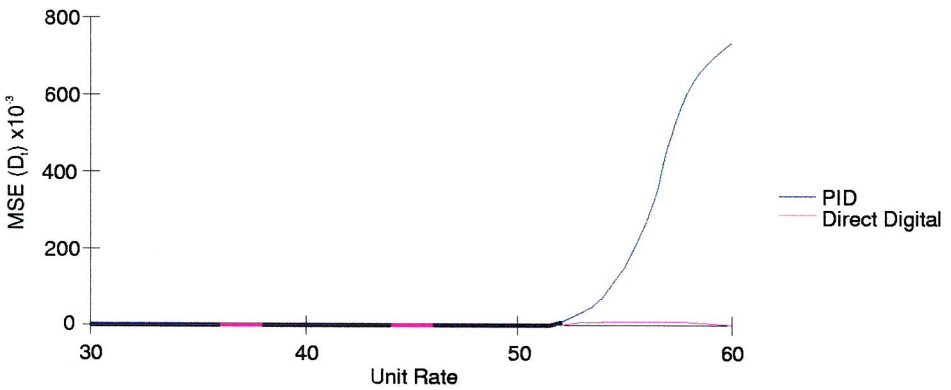

Figure 5.7: Dispenser MSE(D_t) results for direct digital and PID control

5.4 Summary

The performance of both model based controllers is superior to that of PID control. However, their very nature requires knowledge of the system dynamics. Due to non-linearities, design is not as straightforward as for a purely linear system and care must be exercised to ensure that the control effort is not excessive. The direct digital controller is particularly sensitive to high frequency components due to the presence of a differentiator. Both controllers contain a tuning parameter that can be used to overcome deficiencies in the model.

Chapter 6

Knowledge Based Controllers

6.1 Introduction

Knowledge based controllers fall broadly into two types; those that are constructed with knowledge inbuilt and those that obtain their knowledge during operation. Fuzzy controllers are an example of the earlier type with the knowledge being encoded into a rule base. The knowledge is obtained either from the operator of the plant, if the controller is to automate a system that is currently manual, or from the design engineer. Iterative learning control obtains knowledge of the plant during operation. In this type of controller the performance is improved based on knowledge recorded from previous trials.

Fuzzy control attempts to allow linguistic variables such as *more* and *less*, *faster* and *slower* to be used in the design of controllers. This is achieved by dividing the range of inputs into a series of overlapping sets[72, 73, 74]. An input variable is then fuzzified by determining by how much it belongs to each of the fuzzy sets. Each result is used in a series of *if-then* statements, which form the knowledge base, to determine a new fuzzy set. This new fuzzy set is then converted to a crisp value by either taking the centre of area or the centre of mass of the set. The performance of the fuzzy controller

is almost entirely dependent on the rule base, design of which could be considered almost as much of a "black art" as tuning of a PID controller. Consequently there are numerous papers describing the arrangement of fuzzy controllers, especially those based around PID type structures[75, 76, 77, 78, 79]. Fuzzy theory has also been used in the online tuning of conventional PID controllers[80, 81, 82].

Iterative learning control is a relatively new technique, first being proposed in 1984[83]. This method of control is only applicable to systems that continually repeat the same demand profile. The concept is to record the error for the current trial and use this to alter the shape of the demand profile such that the tracking error between output and a reference profile is reduced. As the number of iterations increase then, theoretically, the tracking error should fall to zero. As the system now operates in two dimensions, one of time and one of trial number, the analysis becomes considerably more complicated. Research in recent years has been focused on determining the stability of iterative learning controllers, and to predict the rate at which the error converges. This is commonly achieved by analysis of the system in a suitable function space, initially that of the H_∞ space, though other spaces are also used[84, 85, 86, 87, 88]. Recent work also looks at the design of optimal iterative learning controllers where the system learns such that a cost function is minimised[89].

Originally the iterative learning controller was developed for use in robotic manipulators[90, 91]. Other applications are uncommon, with reported applications in CNC[92] and servo systems[93, 94]. Whereas fuzzy control has been applied to almost every aspect of control engineering, and it is possible to buy commercial microprocessors optimised for fuzzy computation, it would appear that learning control has not found wide application. There is a considerable base of theoretical papers and papers that evaluate claims using simulation but very few practical applications. An overview of

learning control is provided in Moore[95] and Horowitz[96].

6.2 Fuzzy Control

A fuzzy controller is divided into three distinct elements (Figure 6.1). The fuzzifier transforms the crisp input to the fuzzy space. The inference engine uses these values and the knowledge base to determine the required fuzzy output. The defuzzifier then converts the fuzzy control effort into a crisp control signal.

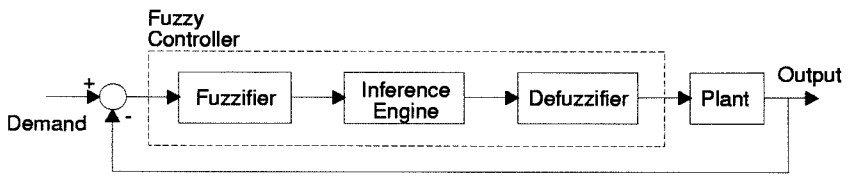

Figure 6.1: Fuzzy controller block diagram

6.2.1 Fuzzifier

The fuzzifier has three inputs; the error, the change in error and the acceleration. These are scaled such that they lie in a universe of discourse between -100:100. To simplify computation this range is also discretised into integer values. The universe is divided into a number of overlapping sets (Figure 6.2) ranging from Large Negative to Large Positive. For ease of computation the sets have been defined to have a triangular shape, but a shape based on an exponential is equally valid. For each of the inputs the membership value, μ, is determined such that:

$$\mu(x, x_0, b) = \begin{cases} 0 & : \quad x < x_0 - \frac{b}{2} \\ \frac{2(x-x_0)+b}{b} & : \quad x_0 - \frac{b}{2} \leq x < x_0 \\ \frac{-2(x-x_0)+b}{b} & : \quad x_0 \leq x \leq x_0 + \frac{b}{2} \\ 0 & : \quad x > x_0 + \frac{b}{2} \end{cases} \tag{6.1}$$

where x is the input value, x_0 is the location of the centre of the fuzzy set and b is the breadth at the base of the fuzzy set. The fuzzifier determines a value of μ for each input and each set. These values are then passed to the inference engine.

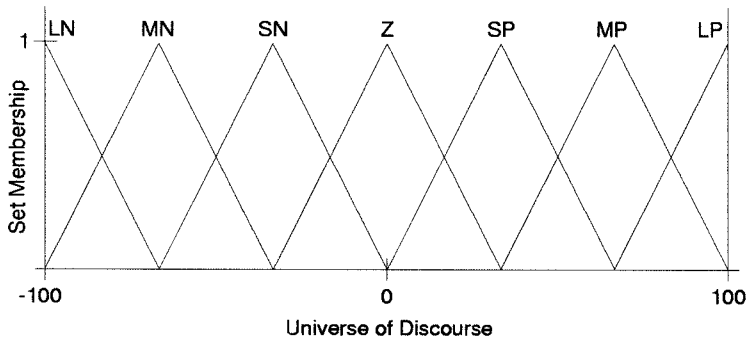

Figure 6.2: Fuzzy sets

6.2.2 Inference Engine

The inference engine contains the controller knowledge in a series of look-up tables (Figure 6.3) with the implemented controller consisting of two sections. The first has an output that is dependent on the error and the rate of change of error. The second uses the demand acceleration to determine a value for the integration limit. As there is only one forward path integrator, the system is only Type 1. Therefore, the error increases rapidly in the presence of acceleration causing the integral sum to wind-up. To avoid overshoot when the acceleration falls to zero the value of the integral sum is limited during acceleration. This is achieved by implementing a normal integral summation, and then using a fuzzy rule base and the acceleration as the input to determine the value to which the integral sum should be limited. In effect, the controller is a fuzzy PID controller with an additional integral limit to improve response.

There are a number of possible inference methods[74]. The star impli-

PD Rule Base

Output \ Error	NL	NM	NS	Z	PS	PL	PM
NL	NL	NL	NL	NL	NM	NS	Z
NM	NL	NL	NL	NM	NS	Z	PS
NS	NL	NL	NM	NS	Z	PS	PM
Z	NL	NM	NS	Z	PS	PM	PL
PS	NM	NS	Z	PS	PM	PL	PL
PM	NS	Z	PS	PM	PL	PL	PL
PL	Z	PS	PM	PL	PL	PL	PL

(Change in Error)

Integration Limit Rule Base

Limit \ Acceleration	NL	NM	NS	Z	PS	PL	PM
	Z	PS	PM	PL	PM	S	Z

Figure 6.3: Fuzzy rule bases

cation method is used here as this is believed to give a smoother change in output than other implication methods[73]. A new table with the same dimensions as the rule base, referred to as the table of weights, is calculated such that the weight at element l, j is:

$$w_{l,j} = A_l(e) \times A_j(ce) \tag{6.2}$$

where $A_l(e)$ is the membership for the error of set l and $A_j(ce)$ is the membership of the change in error for set j. Each fuzzy set indicated in the rule base is then scaled so:

$$B'_k = w_{l,j} B_{l,j} \quad 1 \leq k \leq Number\,of\,Rules \tag{6.3}$$

where $B_{l,j}$ is the fuzzy set indicated in the rulebase at row l, column j. This produces a series of k weighted overlapping fuzzy sets. These are then combined by moving through the discrete universe comparing the values of each weighted set at each point and taking the maximum value:

$$B(y) = max(B'_{i,k}) \tag{6.4}$$

where i is the point in the discrete universe of discourse. This creates a single fuzzy set that becomes the input to the defuzzifier.

6.2.3 Defuzzifier

The defuzzifier converts the fuzzy set that it is passed by the inference engine to a crisp single value that is suitable for output to the plant. This can be achieved in a number of ways[74] but the simplest is to take the centre of area. For a discrete universe this is achieved by:

$$y_0 = \frac{\sum_{i=1}^{n} B(y)y}{\sum_{i=1}^{n} B(y)} \tag{6.5}$$

where n is the number of discrete points in the universe of discourse and y is the value of the universe at the point n.

6.2.4 Results

When implemented, the maximum speed of execution for the controller was only 60Hz. Due to the requirement to compute values for each fuzzy set, each element in the rule base and each point in the universe of discourse, the number of multiplications required at each time step is immense. To improve the speed of execution the software has been modified to minimise floating point calculations. By detecting zeros it is possible to omit a considerable portion of the computation. In this manner the speed of computation has been increased to 1kHz, but as this is still 2.5 times less than that of the other controllers, the fuzzy controller immediately suffers a performance penalty.

The controller requires three gain values, one on each input so that they may be scaled to the range of the universe of discourse. With these gains suitably selected the cost function results for the fuzzy controller are 2.94 and 0.55 for the conveyor and dispenser respectively. Though the dispenser appears to have an improved response, this belies a very poor settling time

(Figure 6.4). The fuzzy controller only brings the dispenser to within specification below 44UPM compared with 54UPM for the PID controller. The reason for the superior cost function result is a less dramatic increase in MSE(D_t) above 50UPM than the PID controller, leading to a better average value for MSE(D_t) (Table 6.1).

	Conveyor		Dispenser	
Parameter	Fuzzy	PID	Fuzzy	PID
MSE	1.07	0.81	0.37	0.46
MSE(D_t)	3.22	0.75	61.3	107
Settling Time	0.07	0.01	0.37	0.14

Table 6.1: Motor average results for fuzzy and PID controllers

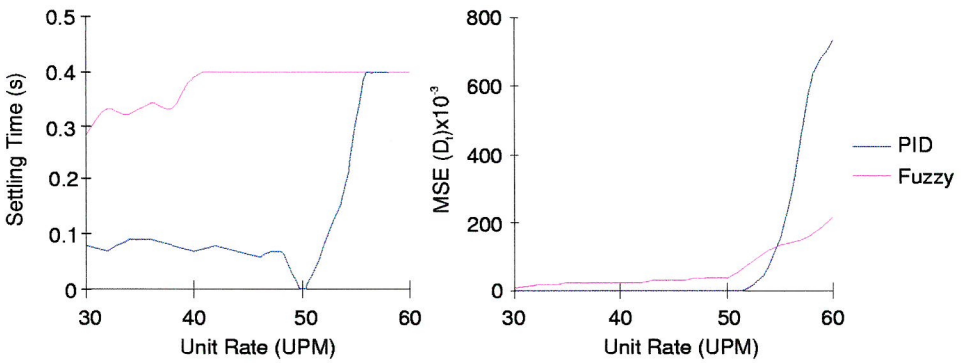

Figure 6.4: Comparison of fuzzy and PID performance for the dispenser

The conveyor performs considerably worse under this fuzzy controller than PID control. The MSE for the entire profile is very similar to the PID result, but increases more rapidly above 50UPM. The reason for the poor performance is a very poor MSE(D_t) (Table 6.1). This result is in part due to the reduction in sampling frequency and to the design of the integral limit. As the conveyor acceleration is always non-zero apart from during the dispensing time, the integral summation will always be limited. This may lead to an increase in the tracking error that the integral term is unable to remove between the demand acceleration falling to zero and the conveyor

coming to rest.

6.3 Iterative Learning Control

As the system is required to follow the same position profile for each unit, it is ideally suited to an iterative learning controller which has been constructed around the existing PID loop. The PID is required to provide disturbance rejection and the learning controller slowly removes position error as the number of completed iterations increases. A memory is used to store the cumulative effect of some proportion of the tracking error over the previous iterations, which is then used to update the demand for the current iteration. This can be represented by:

$$u_{k+1}(t) = u_k(t) + L(t)e_{k+1}(t) \tag{6.6}$$

where k is the k^{th} iteration, $k+1$ is the current iteration $L(t)$ is the learning controller function and t is the time during a single iteration. This approach is referred to as the current error method[97]. Substituting for $u_k(t)$ gives:

$$\begin{aligned} u_k(t) &= u_{k-1}(t) + L(t)e_k(t) \\ \Rightarrow \quad u_{k+1}(t) &= [u_{k-1}(t) + L(t)e_k(t)] + L(t)e_{k+1}(t) \end{aligned} \tag{6.7}$$

This can be generalised to:

$$u_{k+1}(t) = u_{k-m}(t) + L(t) \sum_{i=k-m+1}^{k+1} e_i(t) \tag{6.8}$$

If $m = k$ then this reduces to:

$$u_{k+1}(t) = u_0(t) + L(t) \sum_{i=1}^{k+1} e_i(t) \tag{6.9}$$

where:

$$e_{k+1}(t) = r(t) - y_{k+1}(t) \tag{6.10}$$

and

$$u_0(t) = 0;\ v(t) = u(t) + r(t) \tag{6.11}$$

This is implemented as indicated in Figure 6.5 where N is the number of samples in one iteration. It should be realised that this system is not true iterative learning as the initial conditions for each iteration are not identical. Resetting to identical initial conditions is not practical due to the time required to achieve. This would dramatically reduce the overall unit rate for the system.

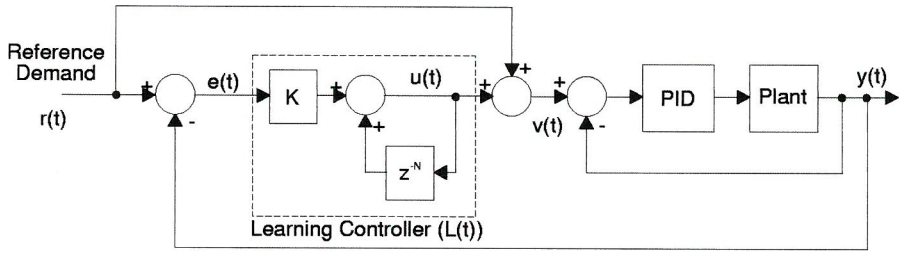

Figure 6.5: Learning controller block diagram

6.3.1 Initial Learning Controller Design

When the iterative learning controller is implemented as indicated in Figure 6.5 then the mean square error reduces by a small amount before growing, eventually leading to instability (Figure 6.6).

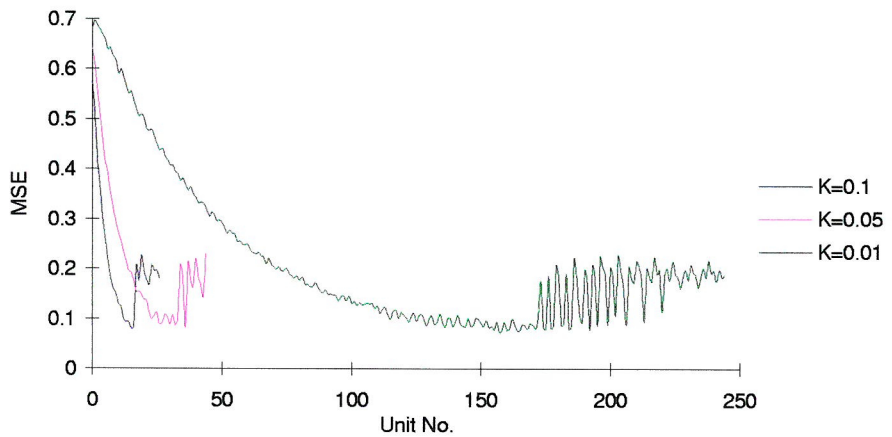

Figure 6.6: Error results for the initial learning controller

For a step input applied at time t, the output will not reach the desired value until some time later, $t+\tau$. If the response at t is required to be at the desired value then it will be necessary to apply the step t' seconds earlier. For the iterative learning controller this means that the learning demand is:

$$u_{k+1}(t) = r(t) + L(t) \sum_{i=1}^{k+1} e_i(t - t') \tag{6.12}$$

This is very simply represented in the z-domain, with the delay becoming:

$$z^{-(n+\tau)} \tag{6.13}$$

where:

$$\tau = t' \times f_s \tag{6.14}$$

6.3.2 Online computation of τ

The necessary time lag could be determined offline, but this would require knowledge of the plant. One of the main attractions of the iterative learning controller is that it does not require this knowledge. As both the reference position and the system output are known, then approximation to τ is straightforward. The demand reference and system output are recorded, and at the completion of the iteration the output position data is shifted relative to the demand position data until a minimum for the magnitude error sum is obtained. The number of repetitions required to achieve a minimum provides an approximation for τ. If N is the number of samples in the data then τ can be found from:

$$\tau = \tau' : \min \left\{ \frac{1}{N - \tau'} \sum_{n=1}^{N-\tau'} |u_0(n) - y_0(n + \tau')|, 0 \leq \tau' < N \right\} \tag{6.15}$$

Mean square error would normally be used but this introduces a large number of multiplications. Use of the mean of the error magnitude reduces the computational overhead. The data is recorded in $u_0(n)$ and $y_0(n)$ at a lower sample frequency than the main controller, also to reduce computation time.

Even so, it can require up to 2ms before the algorithm determines a minimum. For this reason the computation is only performed at the start of the run.

Figure 6.7 shows that with the time lag compensation added to the controller, the error has been found to converge to an acceptable value and then begin to diverge. This is due to discrepancies in the calculation of τ caused by quantisation effects. As the data from which τ is determined is sampled at 500Hz it is only possible to determine τ to an accuracy of $\pm$1ms. An error in the value of τ introduces a cumulative error in the updated demand profile. At large MSE values this will be insignificant, but will become more important as the controller attempts to reduce the MSE to zero.

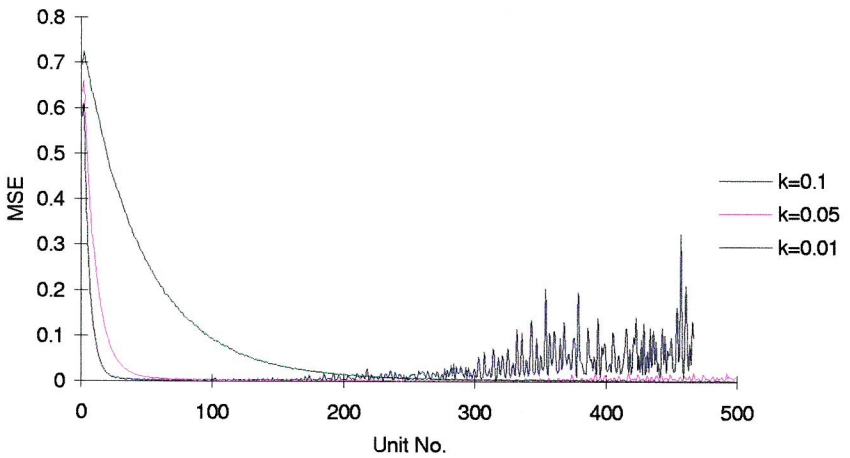

Figure 6.7: Error convergence for new controller

By inserting a MSE threshold a *sufficient* iterative learning controller is created. If the MSE falls below a predetermined point, the performance of the system is considered to be sufficient and the learning controller is then deactivated. The system is now a normal PID controller, and can therefore be analysed using classical control theory. As such, the system will be stable for all further iterations, as long as the dynamics of the plant remain

substationally unaltered. If the dynamics of the system should change, or a suitably large disturbance occur, then the sufficient learning controller is reactivated. Hysteresis between the switching points ensures that continual activation and deactivation of the learning controller is avoided.

Figure 6.8 demonstrates the disturbance performance of the controller. The proportional term of the PID controller is instantly halved at iteration 150 dramatically slowing the response. In the face of this large change in system dynamics, the iterative learning controller is reinitiated and returns the MSE to the sufficient MSE threshold.

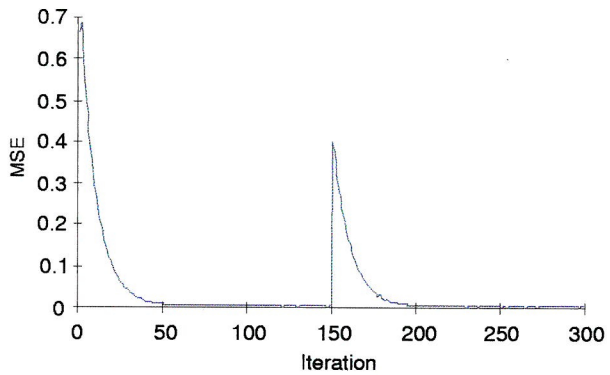

Figure 6.8: Response of learning system to a change in system dynamics

6.3.3 Results

A learning gain value of $k = 0.05$ was used for all tests. This produces a superior control effort profile than $k = 0.1$. There is little difference between control efforts for $k = 0.05$ and $k = 0.01$ but the higher value converges considerably quicker.

The sufficient iterative learning controller has cost function results of 0.26 and 0.01 for the conveyor and dispenser axes respectively. The dispenser is improved well beyond that of the conveyor, performance of the conveyor being limited by the requirement for it to come to rest. As the dispenser velocity never falls to zero continual improvement is possible. The

performance of the learning controller on both axes is significantly improved over that of a standard PID controller, and is achieved at very little computational expense or increase in controller complexity. The performance is also very consistent as the unit rate is increased. Only the flux vector system has performance indices that are similar and this requires new equipment and tuning of up to six gain terms. It is also incapable of reacting to changes in dynamics without manual intervention.

Figure 6.9 shows results obtained for the dispenser with a PID controller and for the sufficient iterative learning controller after it has learnt. The iterative learning system follows the demand with considerably greater accuracy than the equivalent PID system. The integral term in both the learning and the PID controllers has been detuned and this demonstrates the ability of the learning controller to improve poorly tuned systems.

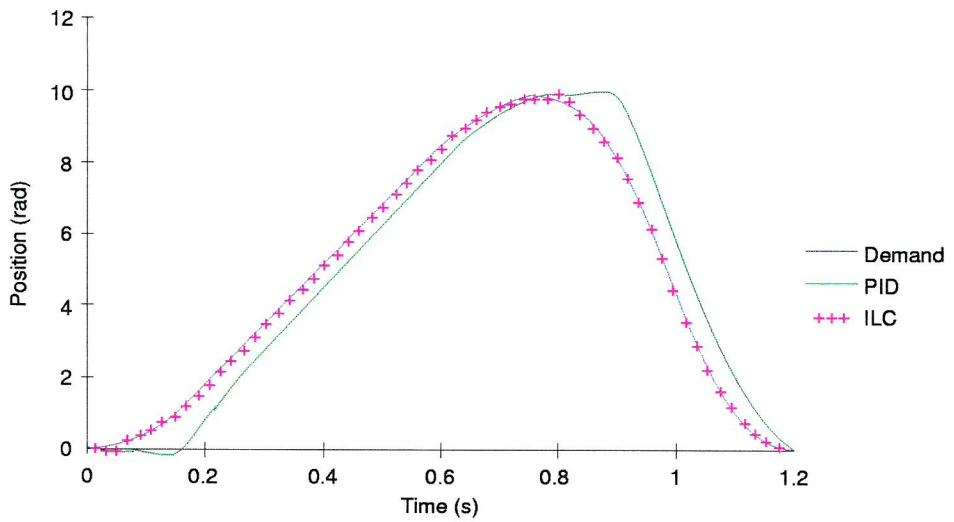

Figure 6.9: Dispenser position results for the PID and learning controllers

When the dispenser is in synchronising mode, the precise position of the dispenser is less important than the relative position between the two axes. Figure 6.10 shows the relative error between the conveyor and the dispenser. The PID system has a PID controller on both axes. The learning system

has a PID controller on the conveyor axis and a learning controller on the dispenser axis. The performance of the PID controller for the constant speed requirement of the conveyor in this mode of operation has been shown to be acceptable (Chapter 4). The learning system maintains the relative position within specification for the majority of the profile. The PID system is only just within specification by the start of the dispensing time. As the unit rate is increased, the PID controller would not be able to ensure satisfaction of the specification by the start of the dispensing time.

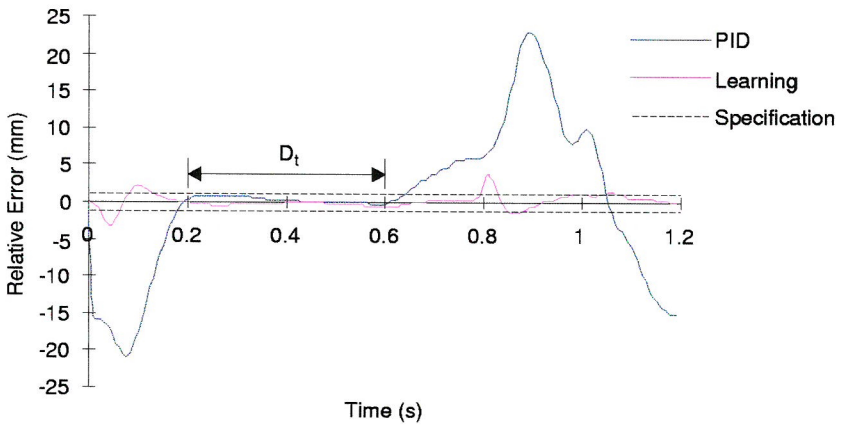

Figure 6.10: Relative error for the learning controller (50UPM)

The effect of the iterative learning controller on shaping the demand profile is indicated in Figure 6.11 where a dispenser demand signal is presented. The lag in the system causes the updated demand to appear in advance of the reference signal. However, closer inspection reveals that the learnt demand is not simply a time displaced copy of the reference. Firstly, the time difference between the two signals is not constant. Secondly, the effect of non-linearities in the system can be seen, especially where the velocity demand passes through zero. To overcome the effect of the inverter deadzone at this point the learnt signal differs considerably from that of the reference.

The repeatability of a system is commonly quoted as a range around the mean equivalent to three standard deviations[98]. If the error is con-

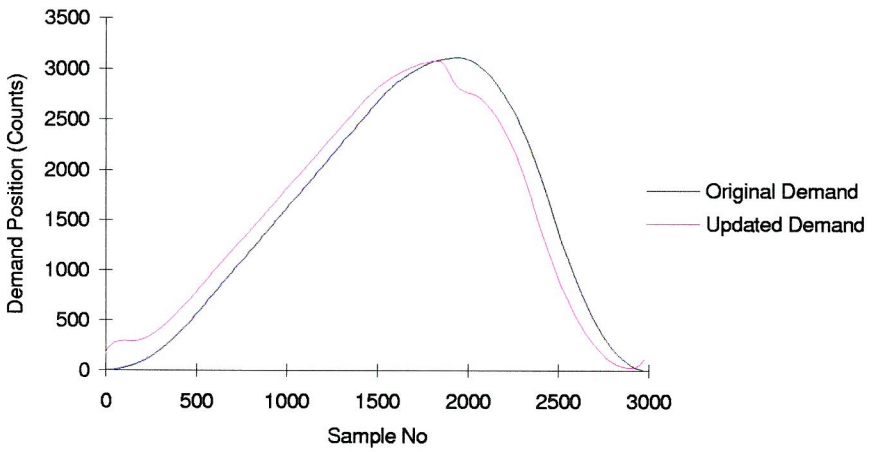

Figure 6.11: Demand signal shaping for the dispenser axis

sidered to have a normal distribution then three standard deviations from the mean will include 99.8% of all units. The results averaged across the unit rate range are reproduced in Table 6.2. Neither controller succeeds in operating the conveyor in such a way that the mechanical components are consistently within the position specification of 53mrad though the motor is. The poor mechanics make repetition very difficult compared to the motor, hence the increase in standard deviation. Also the control is hindered by the mechanical components being outside the control loop.

	Conveyor Error (mrad)		Dispenser Error (mrad)	
Location	Learning	PID	Learning	PID
Motor	-3±47	-13±44	19±24	127±33
Mechanical	-33±140	-39±140	20±21	137±28

Table 6.2: Average error and repeatability results during D_t

The dispenser PID controller achieves the repeatability requirement but the error causes the system to be outside the position specification. The learning controller has a considerably superior error value with a similar repeatability, ensuring that the motor is within specification at almost all unit rates (Figure 6.12).

For both axes the repeatability of the two controllers is very similar. This is to be expected as the disturbance rejection in the learning controller is provided by the PID loop.

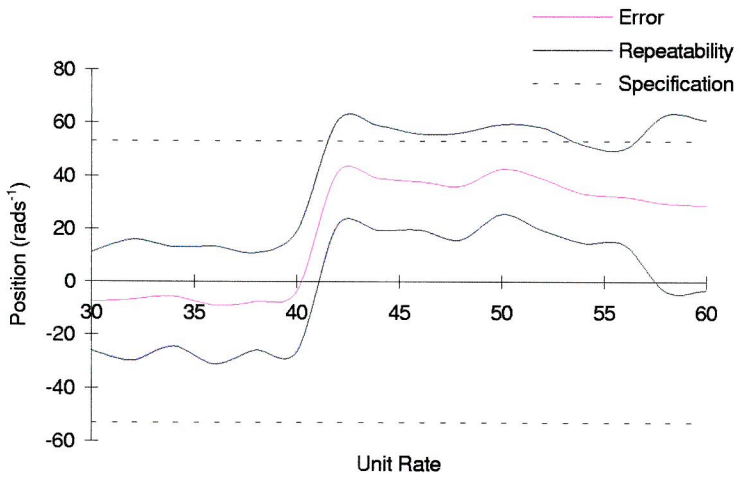

Figure 6.12: Dispenser repeatability under learning control

Figure 6.7 suggests that the error profile should decay. This is confirmed in Figure 6.13 where the improvement in the actual error as the unit number increases is clearly visible. From this plot it would appear that the learning controller is unable to compensate for all the dynamics. However, if the system could be left to operate for long enough without becoming unstable, the error would fall to zero in all places.

6.4 Summary

The fuzzy controller cost function appears to indicate that the controller, certainly for the dispenser, performs better than PID. Closer inspection indicates that the performance at low unit rates is considerably worse than PID but does not degrade to the same extent above 50UPM. The poor performance is partly caused by the exceptionally high computational overhead. Also the fuzzy controller is not well suited to this specific application.

The problem is well defined with an accurate mathematical model of the system. The required performance is also well defined, with a very specific goal. Therefore, there is no area for fuzziness, the application being better served by a more rigorous solution. Though fuzzy control is not suited to the error path controller, it would be suitable in a tuning role as indicated in the literature.

The learning controller is similar to the fuzzy controller in that it is a knowledge based system, but in this case the knowledge is obtained directly by the controller. The difference in performance is dramatic, with the learning controller performance equal to that of the flux vector drive. However, the learning controller requires minimal tuning, can be retrofitted to existing systems and is adaptive. The learning controller also benefits over model based systems in that it does not require knowledge of the system dynamics.

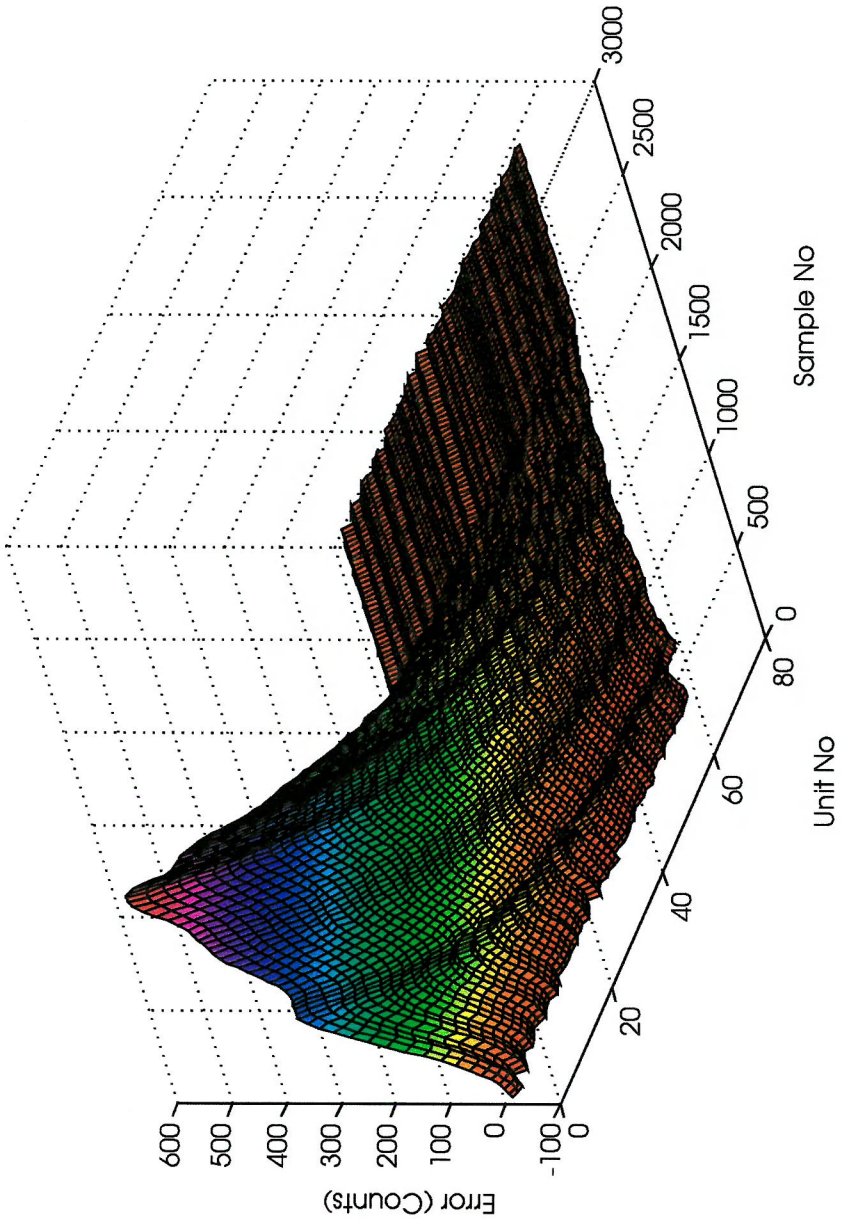

Figure 6.13: Learning error profile for the conveyor at 50UPM

Chapter 7

Drift and Synchronisation Compensation

7.1 Introduction

It is the intention of the system to control the linear position of the mechanical axis, but for reasons of expense and simplicity only motor rotational position is available for feedback. Consequently, the desired linear position has to be converted to angular position, with the radius of the drive sprocket forming the constant of proportionality. This value cannot be known exactly and will vary with operating conditions. Consequently, any error will result in a cumulative error in the linear position. The nett effect is to cause the linear position of the mechanical system to drift relative to the rotational position of the motor. This drift is not compensated for by the controller as the mechanical components are open loop. It is therefore necessary to consider ways of closing this loop to prevent position drift, but without introducing a significant increase in sensor requirements. A method has been proposed in the literature that uses a flag and a homing sensor to measure the average length of the chain[99]. The indexing distance is then varied accordingly. This will eliminate drift, but not errors incurred during the

measurement of the chain length.

Under synchronisation the position of the two axes are related, the correct relative position being required for proper synchronisation. The current controllers take no account of this, treating each axis as an independent entity. Consequently any position error in one axis is not reflected in the other, causing an error in the relative position between the two. To maintain the relative position within specification, it is therefore desirable to investigate methods of coupling the axes so that errors occurring in a single axis are compensated for in both. This can be accomplished in one of two ways[100]. The first, master-slave synchronisation, treats one of the axes as the master and the other is constrained to follow[101, 102]. This system performs best where the two axes follow the same trajectory and where the constant lagging of the slave is acceptable. The second method is to treat the two axes as equals and cross couple the error for each axis into the other[103, 104]. This has the disadvantage that it is possible that neither axis will have an acceptable error, but if the relative position is important this may be of little concern. Derivations of these techniques include the use of fuzzy control[105], preview systems[106], robust and optimal control[107].

Synchronisation and drift both require a linear position accuracy in excess of $\pm$1mm which converts to approximately 17 counts or 53mrad at the motor shaft.

7.2 Open Loop Disturbance Compensation

The linear distance, l, moved by a sprocket of radius r through an angle θ is given by:

$$l = r\theta \tag{7.1}$$

If dr and $d\theta$ are the errors in the measured values, then the error in the linear distance is given by:

$$dl = \theta dr + r d\theta \tag{7.2}$$

Errors in θ caused by rounding in the demand generation have been minimised by the use of double precision floating point numbers. The encoder has a finite resolution of ± 3.14mrad, but over a large number of measurements the average error will tend to zero, so will not contribute to drift. The controller ensures that the motor follows the demand θ with the minimal error so it is therefore valid to assume:

$$d\theta \rightarrow 0 \tag{7.3}$$

which implies that the error in the linear position is simply:

$$dl = \theta dr \tag{7.4}$$

Any error in the sprocket radius causes an error in the linear position that increases the further the axis moves.

Error in the linear position is defined as the difference between the desired position and the actual position:

$$E = l_d - l_a \tag{7.5}$$

where l_d and l_a are the demand and actual positions. Drift is the difference between actual linear positions for successive measurements, which can be considered as the rate of change of error:

$$\begin{array}{rrcl} & D & = & l_a(n) - l_a(n-1) \\ \text{but} & l_a(n) & = & E(n) - l_d(n) \qquad \text{and} \quad l_d(n-1) = l_d(n) \\ \Rightarrow & D & = & E(n) - E(n-1) \end{array} \tag{7.6}$$

7.2.1 Uncompensated Drift

To measure drift for the conveyor axis, a flag was attached to the conveyor such that it passed through a sensor once every conveyor cycle of 20 units.

Triggering of the sensor activates an interrupt that saves the values on the encoder counters. If there is no drift then the encoder count will be nl_d where n is the number of complete cycles. Comparison of the value recorded from the encoder with the demand position gives an indication of the drift. A similar mechanism was constructed for the dispenser.

Figure 7.1 shows that the conveyor exhibits an average drift of -83mrad per cycle while the average dispenser drift is only -0.3mrad. As the nett position demand for the dispenser is zero, Equation 7.4 implies that the error and hence the drift will also be zero.

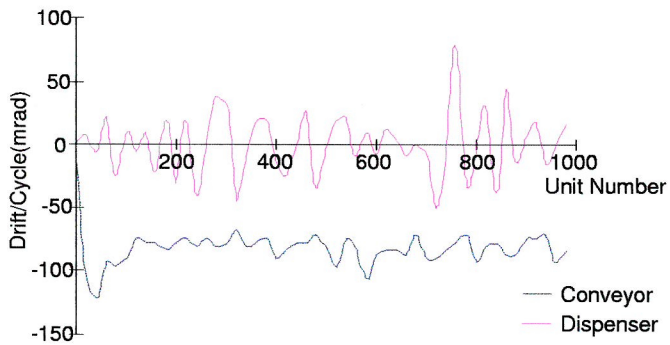

Figure 7.1: Conveyor and dispenser uncompensated drift at 50UPM

7.2.2 Drift Compensation

Equation 7.4 suggests that the relationship between position error and sprocket radius is linear. Hence, if the sprocket radius is constant the change in error over successive measurements, or the drift, will also be constant. Figure 7.2 confirms this experimentally and indicates that it is possible to eliminate drift by variation of the sprocket radius, an effect that is achieved by adjusting the radius value used in the calculation of the demand signal. By measuring the drift, an estimate for the straight line can be made and the correct sprocket radius determined.

In practice this is not possible as the drift value is corrupted with noise.

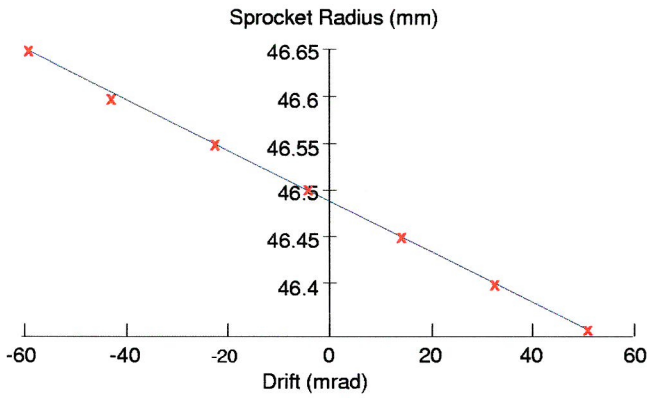

Figure 7.2: Relationship between drift and sprocket radius

To overcome this, a least squares approach is used to estimate the radius. Feedback data is used to generate an equation relating drift and radius of the form:

$$r = c_0 + c_1 d \tag{7.7}$$

where r and d are the radius and drift respectively and c_0 and c_1 are constants. Using least squares[108], n measurements can be used to generate an approximation for Equation 7.7 by solving the simultaneous equations:

$$nc_0 + \left(\sum_{i=1}^{n} d_i\right) c_1 = \sum_{i=1}^{n} r_i \tag{7.8}$$

$$\left(\sum_{i=1}^{n} d_i\right) c_0 + \left(\sum_{i=1}^{n} d_i^2\right) c_1 = \sum_{i=1}^{n} d_i r_i \tag{7.9}$$

where r_i and d_i are the radius and the drift at iteration i. The drift is required to be zero, so Equation 7.7 implies $r = c_0$. Equations 7.8 and 7.9 are then used to eliminate c_1 and the estimate for the new radius is:

$$\hat{r} = \frac{\sum_{i=1}^{n} d_i \sum_{i=1}^{n} d_i r_i - \sum_{i=1}^{n} d_i^2 \sum_{i=1}^{n} r_i}{\left(\sum_{i=1}^{n} d_i\right)^2 - n \sum_{i=1}^{n} d_i^2} \tag{7.10}$$

The results for such a system (Figure 7.3) show that the average drift for the conveyor is reduced from -83mrad to 4.5mrad per cycle, and this is

still sufficient to cause the error to exceed the specification within 200 units. Obviously it is insufficient to compensate for drift only, error and drift must be corrected simultaneously.

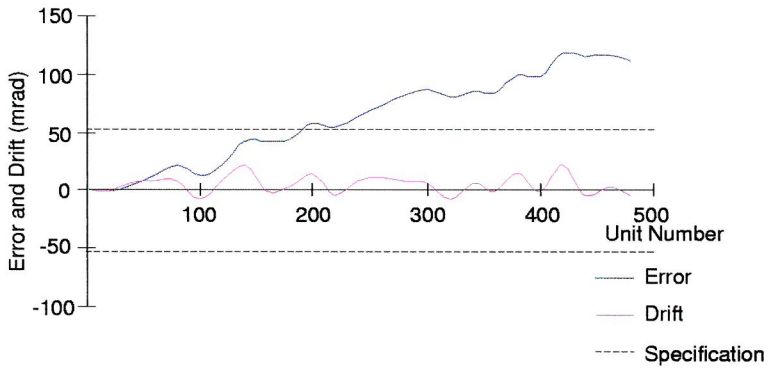

Figure 7.3: Conveyor error and drift with drift compensation only

7.2.3 Error and Drift Compensation

To compensate for the error, the least squares algorithm is used as before but the estimated value of radius is adjusted further so that the error is reduced to zero at completion of the next cycle. The demand angle of rotation for the sprocket is updated based on the estimate of sprocket radius. This and the error value are used to determine the required value of radius that will set the error to zero at completion of the next cycle, and is found from:

$$\begin{aligned} \theta &= \frac{d_m}{\hat{r}} \\ \Rightarrow \quad r' &= \frac{d_m}{\theta - E_{rad}} \end{aligned} \tag{7.11}$$

where d_m and E_{rad} are the linear distance between units and the linear position error referred to the motor shaft.

Simulation showed that if the radius of the drive sprocket changed, the least squares algorithm was very slow to recover (Figure 7.4). Limiting the number of samples in the least squares algorithm produced a system that responded much more quickly but makes the estimate more sensitive to noise on the measurement signal.

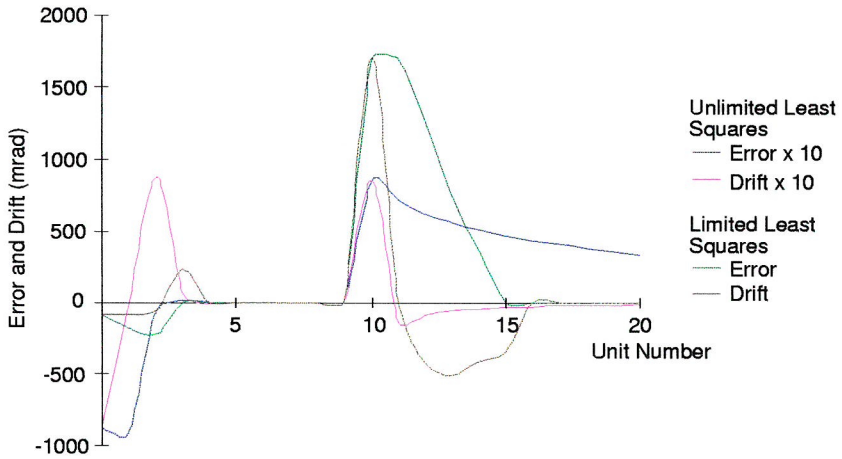

Figure 7.4: Simulated compensator responses to a step change in radius

The drift compensator described was implemented and found to bring drift and error to within specification in under 100 units (Figure 7.5) with the least squares algorithm limited to 5 samples.

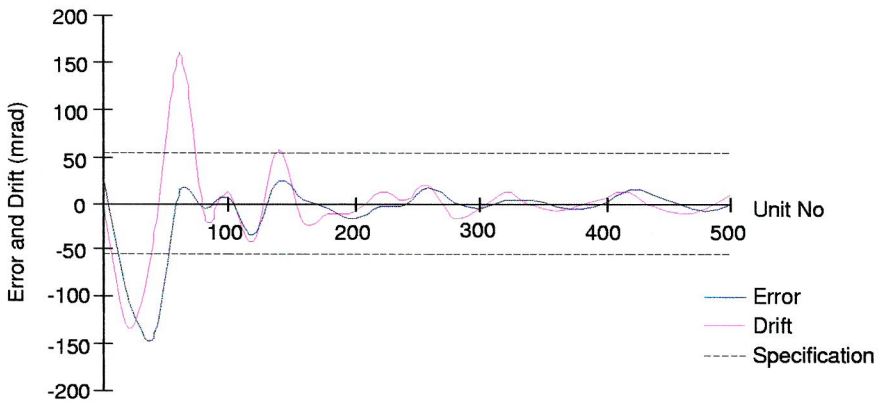

Figure 7.5: Experimental response of fully compensated system

7.3 Synchronisation Control

When the system is in the synchronisation mode of operation, the precise position of an individual axis is less important than the relative position between the two axes. Under normal PID control at 50UPM the relative

position error, the synchronising error, is within specification. The addition of synchronising elements has been investigated to ascertain whether they can reduce the synchronising error in the presence of disturbances applied to one axis.

7.3.1 Master-Slave Synchronisation Controller

The demand profile for the conveyor is altered so that it becomes the difference between the dispenser and the original conveyor position demands (Figure 7.6). To minimise the synchronising error a proportion of the dispenser error is coupled into the conveyor axis. The updated demand profile

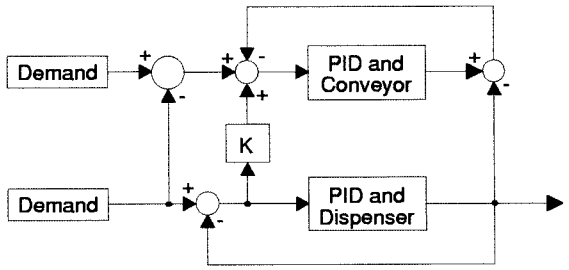

Figure 7.6: Master-slave controller block diagram

for the conveyor results in the conveyor accelerating and then rapidly decelerating, with the profile attempting to keep the axes in synchronisation throughout the entire unit. This more arduous conveyor profile will negate many advantages introduced by running the system under synchronisation, and will introduce larger errors onto the conveyor axis than would otherwise have been incurred.

7.3.2 Cross-Coupled Synchronisation Controller

A proportion of the error is coupled into each axis via a gain (Figure 7.7). In this manner each axis follows the demand trajectory, but is affected by the error in the second axis. Because the conveyor is not being driven by a profile derived from the conveyor and dispenser profiles, the conveyor moves

in a considerable smoother motion and the effect of the synchronisation controller is less apparent to the eye. The gains have been tuned by use of the simulation to provide the minimum error during the dispensing time.

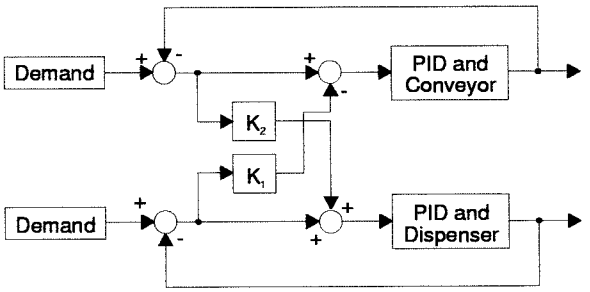

Figure 7.7: Cross-Coupled controller block diagram

7.3.3 Optimal Synchronisation

The optimal controller is designed using the same procedure outlined in Chapter 5. The synchronisation error is defined to be:

$$\begin{aligned} e_s(t) &= (u_1(t) - u_2(t)) - (y_1(t) - y_2(t)) \\ &= e_1(t) - e_2(t) \end{aligned} \tag{7.12}$$

The C matrix in the state space model is constructed such that the error from each axis forms the output, such that:

$$C = \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix} \tag{7.13}$$

and the synchronisation objective is then:

$$S = \begin{bmatrix} c_1 & -c_2 \end{bmatrix} \tag{7.14}$$

This is then formed into a quadratic cost function[107]:

$$J = \int_0^\infty \left[\alpha_1 x^T C^T C x + \alpha_2 x^T S^T S x + u^T R u\right] dt \tag{7.15}$$

which can be rearranged to the more familiar form of:

$$J = \int_0^\infty \left[x^T Q x + u^T R u\right] dt \tag{7.16}$$

if:

$$Q = \alpha_1 C^T C + \alpha_2 S^T S \tag{7.17}$$

The design now proceeds as for the single axis optimal controller with the parameters α_1 and α_2 used to control emphasis on either individual axis tracking or the synchronisation objective.

7.3.4 Synchronisation Results

Figure 7.8 shows the mean square synchronising error for the various controllers compared to a PID system when operated at 50UPM. Results are for a system with no disturbance, and for a simulated disturbance achieved by applying a gain of 0.7 to the dispenser control effort. Results obtained are then normalised around those of the PID controller. The performance of the cross-coupled and the master-slave controller without a disturbance is considerably poorer than that of the PID controller alone. However, both controllers reduce the effect of the disturbance applied to the system, the cross-coupled system noticeably so.

The degradation in performance is due to the new conveyor trajectory. Under normal PID control the conveyor has to follow a constant velocity, a requirement that PID control fulfils admirably. The synchronisation controllers effectively change the trajectory that the conveyor must follow. If the conveyor could track the new trajectory perfectly then errors in the dispenser axis would be compensated for and the synchronising MSE would improve. As the conveyor can not track the new trajectory perfectly, the errors on the dispenser axis will not be compensated for, errors in the conveyor axis increase and the overall MSE increases.

The only controller that shows an improvement over PID control is the optimal controller. This may be entirely due to improved tracking of the demand signals, but the synchronising MSE for the optimal controller with no cross-coupling shows that this is not so. The large synchronising error for

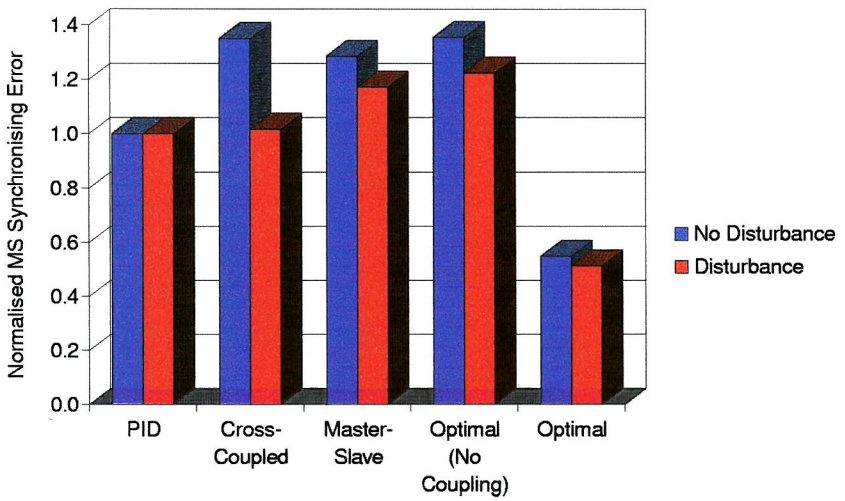

Figure 7.8: Normalised mean square synchronisation error comparisons

the non-cross-coupled optimal controller, $\alpha_2 = 0$, is due to poorer tracking of the conveyor signal than the PID controller, which could be improved to some degree by increasing the integral gain value. When the optimal controller is cross-coupled with $\alpha_2 = 100\alpha_1$ then the improvement in synchronisation error is significant.

Figure 7.9 shows that though the MSE for the entire profile is improved under optimal control, in the absence of a disturbance none of the controllers give a better performance during the dispensing time than the PID controller. For all the controllers adding extra degrees of freedom to regulate the synchronisation error has made the system more oscillatory.

Synchronisation control is obtained through adjusting the profiles of the two axes to minimise the synchronisation error. This results in the conveyor operating with violent changes in speed for which it is not designed. For unit rates in excess of 50UPM, attempting to control the synchronisation error in this way is not suitable. The synchronisation error is only required to be minimal during the dispensing time. To extend the application of synchronisation control to above 50UPM, a method of only applying syn-

chronisation control when it is required must be developed. In this manner, violent changes in conveyor speed caused by the large error in the dispenser axis when it returns to the origin can be avoided.

7.4 Summary

The causes of drift in the conveyor axis have been identified and shown to consist of random and constant components. A method has been suggested to overcome these effects and ensure that the conveyor position remains close to that of the motor.

Under dispensing, the precise position of each axis is less important than the relative position between the two. Controllers have been tested to determine whether it is possible to reduce the synchronisation error by considering the conveyor and dispenser as a coupled system. A performance improvement over PID is only achieved by use of an optimal controller, and in the absence of disturbances the performance of the PID controller during the dispensing time is superior. Due to the nature of the new demand on the conveyor, attempts to couple the two axis are limited to unit rates less then 50UPM.

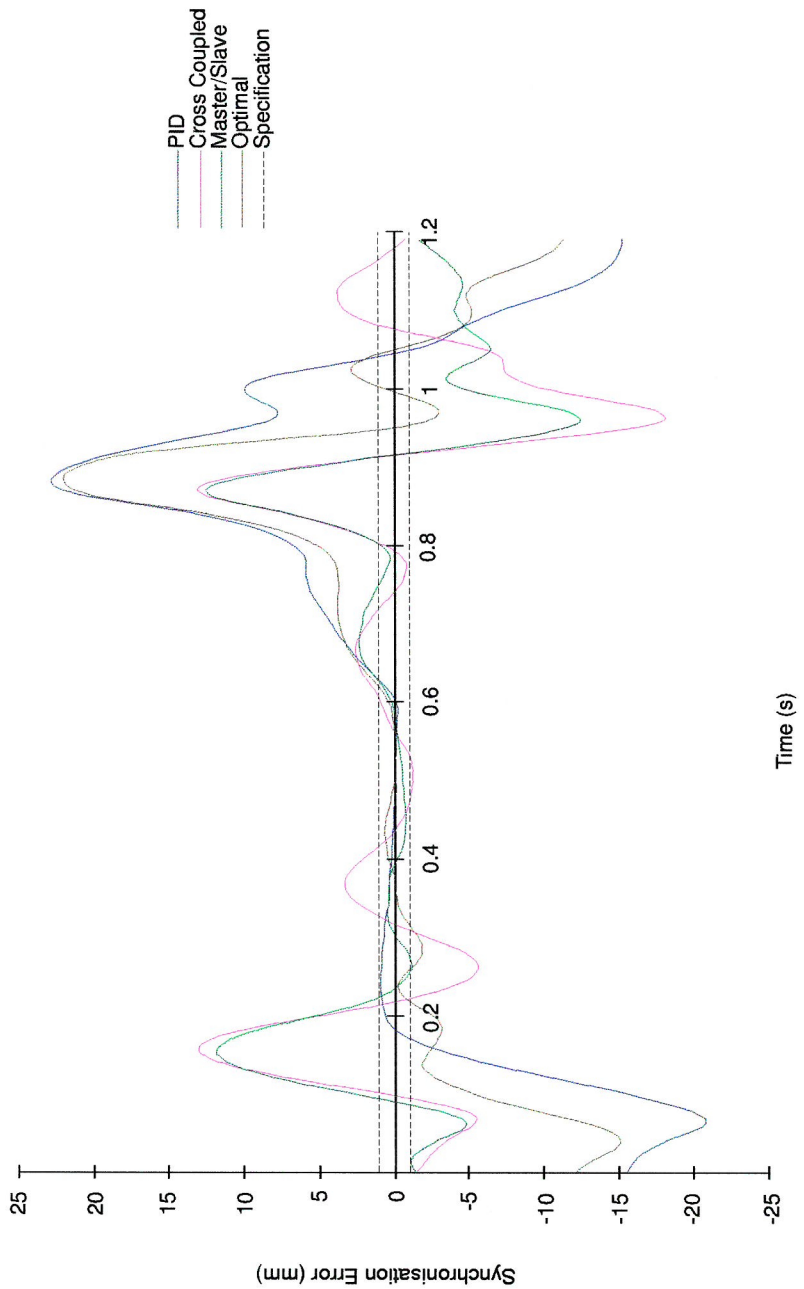

Figure 7.9: Comparison of synchronisation error for different controllers

Chapter 8

Conclusions and Further Work

8.1 Conclusions

The controller cost function for both axes has been combined and then normalised around the value obtained for the PID controller. This provides a single value that compares the performance of the various controllers and the results are shown in Figure 8.1.

From this, it is apparent that the fuzzy controller and the flux vector inverter in open loop mode are inferior to the original PID system. The fuzzy controller is difficult to design properly and executes slower than the other controllers. The flux vector system is penalised heavily for a poor settling time and MSE(D_t) for the conveyor even though the dispenser performs better than PID.

By adding information about the dynamics of the system in the form of a model, the performance can be improved to around 0.4 of the PID value. These systems are limited by the requirement to design the controller around detailed knowledge of the system dynamics, whereas a similar improvement can be obtained by the addition of a velocity feedforward term to the PID

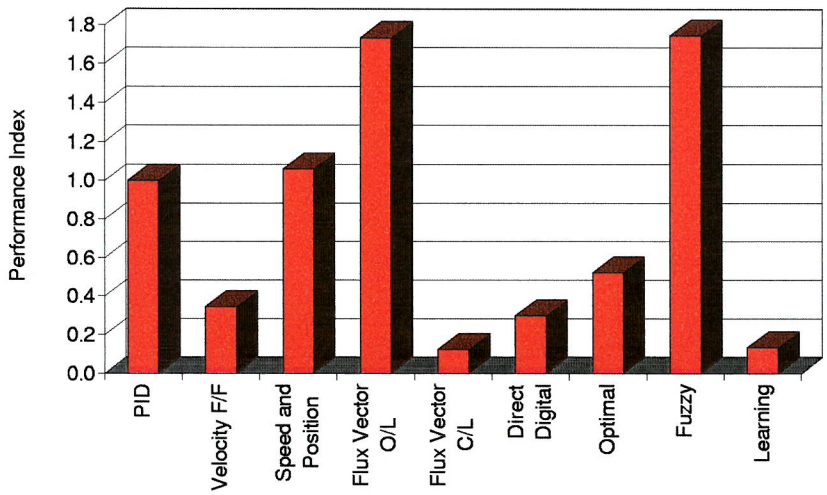

Figure 8.1: Controller performance indices

system. This has the advantage of not requiring knowledge of the dynamics but does provide another gain that requires tuning.

The most superior systems are the flux vector inverter in closed loop mode and the learning controller, both producing a performance index of approximately 0.1. Implementation of a flux vector system requires replacement of existing system components and also the tuning of another controller. Conversely, the learning controller only adds a single gain to the system and in addition, there is no requirement to replace existing hardware.

A method for compensating for drift has been proposed and has been shown to adequately remove drift and error. However, there are a number of limitations to the proposed method. Firstly, it requires a flag to be attached to the system which may be accidentally moved out of alignment. Secondly, it only operates once per conveyor cycle, resulting in a very poor resolution.

Methods for ensuring synchronisation have been investigated. Of these, the method based on optimal control was found to be the most suitable. As the unit rate was increased, difficulties were encountered with the synchronisation controllers as the demanded trajectory for the conveyor became more

severe. Further investigation in this area will need to consider methods of relaxing the synchronisation requirement when it is not expressly required.

From these results it is suggested that the most appropriate method of improving the control of chain conveyor systems is to use a sufficient iterative learning controller. This has been shown to improve the trajectory following and synchronisation error but also has the advantages of not eliminating the PID controller or requiring new equipment. This solution is therefore very attractive to an industry that is reluctant to replace a system that is well known. The learning controller is also adaptive and will ensure an adequate performance even if the dynamics of the system change, without the need for manual intervention. The ability to learn also provides compensation for poorly tuned controllers.

8.2 Further Work

This project has identified a possible solution for the control of chain conveyor systems that fulfils all the requirements. From this work a number of possible directions for further research have become apparent.

- Close the loop around the mechanical plant

  All the proposed controllers fail to improve the performance of the conveyor during the dispensing time much beyond that of PID. To overcome this difficulty the control loop must be closed around the mechanics. A sensor may be added to the conveyor but such a sensor must be mechanically robust and not require expensive cabling. It may be possible to use a magnetic system to monitor the passing of the chain. Electronics located at the computer end would then extrapolate an accurate position measurement from this data.

  Alternatively it may be possible to model the system more accurately. Further research may be able to determine a method of modelling the

variation in sprocket radius based on the system velocity and acceleration. This could be combined with a more accurate model of the system dynamics to provide an estimate of the conveyor position. Such an estimator could be combined with the new sensor.

Once the improved feedback data is obtained then some method would need to be found of integrating this into the controller. It may form the sole source of position feedback, but is more likely to supplement that from the motor encoder.

- Development of the learning controller

  This can progress in a number of directions:

  - Replacement of the PID loop

    The repeatability of the learning controller has been shown to be similar to that of the PID controller. This is due to the learning controller operating over a time period equivalent to one unit as opposed to one sample instant for the PID controller. The learning controller therefore relies on the PID loop for disturbance rejection. If the PID controller is replaced with a controller that exhibits a lower MSE(D_t) then the repeatability will be improved.

  - Calculation of the Learning Gain

    At present, tuning of the learning gain has been achieved manually. Due to the nature of the system this can be a very time consuming exercise. Firstly, the controller should be implemented on a number of systems with a wide range of dynamics. If from this it is found that a range of learning gain values is required then a method for determining the optimum value, preferably online, should be obtained. Such methods exist in the literature but require knowledge of the plant dynamics. A more desirable method may be a Zeigler-Nichols equivalent method for learning

control.

- Specification of the number of iterations for convergence

  The current system has no method of predicting when the system will reach the sufficient error condition. From an industrial perspective it would be most desirable to be able to specify the number of iterations that are to be performed to obtain the required performance.

- Online modelling

  The model based controllers discussed in this project were constructed around a simple model, offline. It would be interesting to investigate whether a model generated at run-time would produce a better controller. Also a number of the above suggestions may require system models, and if these were to be practicable then models of the system would need to be generated online.

To conclude, a significant improvement in the performance of chain conveyor system can be obtained by the use of a sufficient iterative learning controller. Given the computational power of PLC's and the high requirements of modern manufacturing, then there would appear to be little reason to remain solely with a controller design that is 60 years old.

Appendix A

Full Model Parameter Values

A.1 General Parameters

OPERATIONAL PARAMETERS			
UPM	=	50	
op_mode	=	-1	-1 for indexing, 1 for synchronisation
GEAR RATIOS			
conv_gear	=	2.5	
disp_gear	=	2.4	
CONTROLLER SAMPLING FREQUENCY			
fs	=	2500	
Ts	=	1/fs	
SPROCKET RADIUS			
rd_conv	=	0.046482	Conveyor sprocket radius (m)
rd_disp	=	0.038195	Dispenser pulley radius (m)
ENCODER PULSES			
pulses	=	2000	PPR from motor encoders
DEMAND PROFILE PARAMETERS			
pt	=	0.4	Dispensing time (s)
beta	=	0.75	Time for conveyor to reach v_c
delta	=	0.5	Ratio of forward travel to d_m
sigma_synch	=	0.4	Dispensing time
dm	=	0.3175	Distance between centres
dt	=	0.1	Fillable length of tray
da	=	0.25	Proportion of d_t to synch in
I/O CARD PARAMETERS			
DAC_limit	=	10000	Maximum DAC voltage
DAC_resolution	=	12	

ENCODER PULSES PER UNIT AT THE MOTOR

dm_conv_mot = dm x pulses x conv_gear/(2.0 x π x rd_conv)

dm_disp_mot = dm x pulses x disp_gear/(2.0 x π x rd_disp)

A.2 Drive Parameters

INVERTER			
deadzone	=	0.121	
r	=	2.0151	Inverter losses
slew_rate	=	211.8675	Inverter slew rate
kspeed	=	59.4519	Inverter speed integral
cr	=	134	Inverter filter
INDUCTION MOTOR			
Vs	=	415	Supply Voltage
f	=	50	Supply frequency (Hz)
Rs	=	11	Stator resistance (Ω)
Rr	=	12	Rotor resistance (Ω)
Lsl	=	0.03	Stator self inductance (H)
Lrl	=	0.0442	Rotor self inductance (H)
Lm	=	0.81	Mutual inductance (H)
Ls	=	Lsl + Lm	
Lr	=	Lrl + Lm	
Msr	=	Lm	
P	=	2	No of Poles
Jm	=	0.00427	Inertia (kgm^2)
Bm	=	0.005	Viscous friction (Ns/m)
Cm	=	0.1	Dry friction (Nm)

A.2.1 Conveyor Parameters

Kc	=	175052	Stiffness (N/m)
M1c	=	9.1	Part chain mass (kg)
M2c	=	9.2	Part chain mass (kg)
B1c	=	0.9	Viscous friction (Ns/m)
B2c	=	0.01	Viscous friction (Ns/m)
Brc	=	0.02	Bearing viscous friction
Cc	=	25.2	Dry friction (N)
J1c	=	0.002	Sprocket inertia (kgm^2)
J2c	=	J1c	Sprocket inertia (kgm^2)
r1c	=	rd_conv	Sprocket radius (m)
r2c	=	rd_conv	Sprocket radius (m)

A.2.2 Dispenser Parameters

K1d	=	50000	Stiffness (N/m)
K2d	=	K1d	Stiffness (N/m)
M1d	=	1	Dispenser mass (kg)
J1d	=	2e-4	Pulley inertia (kgm^2)
J2d	=	J1d	Pulley inertia (kgm^2)
B1d	=	1	Viscous friction (Ns/m)
Brd	=	0.0125	Bearing viscous friction
r1d	=	rd_disp	Pulley radius (m)
r2d	=	r1d	Pulley radius (m)
Cd	=	20.92	Dry friction (N)

Appendix B

Controller Gain Values

B.1 PID Controller Gains

Unit	Conveyor			Dispenser		
Rate	kp	ki	kd	kp	ki	kd
30	4.5	0.00015	51	5.6	0.011	69.8
32	4.5	0.00015	51	5.6	0.011	69.8
34	4.5	0.00015	51	5.6	0.011	69.8
36	4.5	0.00014	51	5.6	0.011	69.8
38	4.5	0.00013	51	5.6	0.011	69.8
40	4.5	0.00012	51	5.6	0.011	69.8
42	4.5	0.00012	51	5.6	0.011	69.8
44	4.5	0.00011	51	5.6	0.011	69.8
46	4.5	0.00011	51	5.6	0.011	69.8
48	4.5	0.00010	51	5.6	0.011	69.8
50	4.5	0.00010	51	5.6	0.011	69.8
52	4.5	0.00010	51	5.6	0.009	69.8
54	4.5	0.00009	51	5.6	0.007	69.8
56	4.5	0.00008	51	5.6	0.005	69.8
58	4.5	0.00007	51	5.6	0.003	69.8
60	4.5	0.00006	51	5.6	0.001	69.8

B.2 PID with Velocity Feedforward

Unit	Conveyor				Dispenser			
Rate	kp	ki	kd	kv	kp	ki	kd	kv
30	4.5	0	51	900	5.6	0.011	69.8	500
32	4.5	0	51	790	5.6	0.011	69.8	480
34	4.5	0	51	670	5.6	0.011	69.8	460
36	4.5	0	51	570	5.6	0.011	69.8	440
38	4.5	0	51	500	5.6	0.011	69.8	420
40	4.5	0	51	440	5.6	0.011	69.8	400
42	4.5	0	51	320	5.6	0.011	69.8	380
44	4.5	0	51	270	5.6	0.011	69.8	360
46	4.5	0	51	270	5.6	0.011	69.8	340
48	4.5	0	51	230	5.6	0.011	69.8	320
50	4.5	0	51	190	5.6	0.011	69.8	300
52	4.5	0	51	120	5.6	0.009	69.8	280
54	4.5	0	51	120	5.6	0.007	69.8	260
56	4.5	0	51	112	5.6	0.005	69.8	240
58	4.5	0	51	110	5.6	0.003	69.8	220
60	4.5	0	51	100	5.6	0.001	69.8	200

B.3 Speed and Position Loop Gains

Unit	Conveyor					
Rate	Kp(pos)	Ki(pos)	Kd(pos)	Kp(spd)	Ki(spd)	Kd(spd)
30	0.007	0	0.001	1200	0.030	100
32	0.007	0	0.001	1200	0.029	100
34	0.007	0	0.001	1200	0.028	100
36	0.007	0	0.001	1200	0.027	100
38	0.007	0	0.001	1200	0.026	100
40	0.007	0	0.001	1200	0.025	100
42	0.007	0	0.001	1200	0.024	100
44	0.007	0	0.001	1200	0.023	100
46	0.007	0	0.001	1200	0.022	100
48	0.007	0	0.001	1200	0.021	100
50	0.007	0	0.001	1200	0.020	100
52	0.007	0	0.001	1200	0.021	100
54	0.007	0	0.001	1200	0.023	100
56	0.007	0	0.001	1200	0.020	100
58	0.007	0	0.001	1200	0.020	200
60	0.007	0	0.001	1200	0.020	200

Unit	Dispenser					
Rate	Kp(pos)	Ki(pos)	Kd(pos)	Kp(spd)	Ki(spd)	Kd(spd)
30	0.0100	0.00003	0.001	1000	0.7	5000
32	0.0102	0.00003	0.001	1000	0.7	5000
34	0.0104	0.00003	0.001	1000	0.7	5000
36	0.0106	0.00003	0.001	1000	0.7	5000
38	0.0108	0.00003	0.001	1000	0.7	5000
40	0.0110	0.00003	0.001	1000	0.7	5000
42	0.0110	0.00003	0.001	1000	0.7	5000
44	0.0110	0.00003	0.001	1000	0.7	5000
46	0.0110	0.00003	0.001	1000	0.7	5000
48	0.0110	0.00003	0.001	1000	0.7	5000
50	0.0110	0.00003	0.001	1000	0.7	5000
52	0.0110	0.00003	0.001	1000	0.7	5000
54	0.0100	0.00003	0.003	1000	0.7	5000
56	0.0100	0.00001	0.005	1000	0.7	5000
58	0.0100	0.000001	0.010	1000	0.7	5000
60	0.0080	0.0000001	0.010	1000	0.7	5000

B.4 Closed Loop Flux Vector Inverter

Unit	Conveyor					
Rate	Kp(pos)	Ki(pos)	Kd(pos)	Kp(spd)	Ki(spd)	Kd(spd)
30	14	0	51	200	40	40
32	14	0	51	200	40	40
34	14	0	51	200	40	40
36	14	0	51	200	40	40
38	14	0	51	200	40	40
40	14	0	51	200	40	40
42	14	0	51	200	40	40
44	14	0	51	200	40	40
46	14	0	51	200	40	40
48	14	0	51	200	40	40
50	14	0	51	200	40	40
52	14	0	51	200	40	40
54	14	0	51	200	40	40
56	14	0	51	200	40	40
58	14	0	51	200	40	40
60	14	0	51	200	40	40

Unit	Dispenser					
Rate	Kp(pos)	Ki(pos)	Kd(pos)	Kp(spd)	Ki(spd)	Kd(spd)
30	20	0.08	100	200	40	40
32	20	0.08	100	200	40	40
34	20	0.08	100	200	40	40
36	20	0.08	100	200	40	40
38	20	0.08	100	200	40	40
40	20	0.08	100	200	40	40
42	20	0.08	100	200	40	40
44	20	0.08	100	200	40	40
46	20	0.08	100	200	40	40
48	20	0.08	100	200	40	40
50	20	0.08	100	200	40	40
52	20	0.08	100	200	40	40
54	20	0.08	100	200	40	40
56	20	0.08	100	200	40	40
58	20	0.08	100	200	40	40
60	20	0.04	100	200	40	40

B.5 Model Based Controller Gains

Direct Digital

Unit Rate	Conveyor ki
30	0.00030
32	0.00030
34	0.00029
36	0.00028
38	0.00027
40	0.00026
42	0.00025
44	0.00023
46	0.00021
48	0.00019
50	0.00017
52	0.00011
54	0.00005
56	0.00003
58	0.00001
60	0

SVF with Ramp Input

Conveyor	
k1	k2
0.000070	2.02
0.000065	2.02
0.000060	2.02
0.000055	2.02
0.000050	2.02
0.000040	2.02
0.000036	2.02
0.000032	2.02
0.000028	2.02
0.000024	2.02
0.000020	2.02
0.000013	2.02
0.000006	2.02
0.000002	2.02
0.000002	2.02
0.0000008	2.02

SVF

Conveyor
k1
1.00
1.00
1.00
1.00
1.00
1.00
1.00
0.94
0.94
0.86
0.80
0.60
0.55
0.58
0.59
0.60

Unit Rate	Dispenser ki
30	0.0070
32	0.0072
34	0.0074
36	0.0076
38	0.0078
40	0.0080
42	0.0082
44	0.0084
46	0.0086
48	0.0088
50	0.0090
52	0.0092
54	0.0094
56	0.0096
58	0.0098
60	0.0100

Dispenser	
k1	k2
0.00120	0.8561
0.00124	0.8561
0.00128	0.8561
0.00132	0.8561
0.00134	0.8561
0.00140	0.8561
0.00152	0.8561
0.00164	0.8561
0.00176	0.8561
0.00188	0.8561
0.00200	0.8561
0.00200	0.8561
0.00140	0.8561
0.00100	0.8561
0.00050	0.8561
0.00020	0.8561

B.6 Knowledge Based Controller Gains

Fuzzy Controller

Unit	Conveyor			
Rate	Error	CE	PD	Int
30	0.070	1	65	0.00030
32	0.070	1	65	0.00028
34	0.070	1	65	0.00026
36	0.070	1	65	0.00024
38	0.070	1	65	0.00022
40	0.070	1	65	0.00023
42	0.070	1	65	0.00020
44	0.070	1	65	0.00005
46	0.070	1	65	0.00005
48	0.070	1	65	0.00010
50	0.070	1	65	0.00500
52	0.065	1	65	0.00500
54	0.065	1	65	0.00500
56	0.060	1	65	0.00500
58	0.055	1	65	0.00500
60	0.050	1	65	0.00500

Learning Contorller

Conveyor			
kp	ki	kd	kl
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05
4.5	0.0001	51	0.05

Unit	Dispenser			
Rate	Error	CE	PD	Int
30	0.11	1.5	65	0.01
32	0.11	1.5	65	0.01
34	0.11	1.5	65	0.01
36	0.11	1.5	65	0.01
38	0.11	1.5	65	0.01
40	0.11	1.5	65	0.01
42	0.11	1.5	65	0.01
44	0.11	1.5	65	0.01
46	0.11	1.5	65	0.01
48	0.11	1.5	65	0.01
50	0.11	1.5	65	0.01
52	0.11	1.5	65	0.01
54	0.11	1.5	65	0.01
56	0.11	1.5	65	0.01
58	0.11	1.5	65	0.01
60	0.11	1.5	65	0.01

Dispenser			
kp	ki	kd	kl
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05
5.6	0.001	69.8	0.05

Appendix C

Tables of Results

The following tables contain the average results for each controller at each value in the unit rate range. Only one set of results for the PID controller has been included. In practice, a number of results were obtained over a period of time. The set of PID results nearest to a controller were then used in the comparison of that controller. This is to allow for subtle changes to the system dynamics.

C.1 Conveyor Motor Results

C.1.1 MSE Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	0.31	0.43	0.10	0.36	0.22	0.03	0.05	0.10	0.16	0.31	0.0005
24	0.35	0.48	0.03	0.42	0.26	0.03	0.06	0.11	0.18	0.35	0.0004
34	0.39	0.54	0.03	0.48	0.30	0.03	0.07	0.13	0.21	0.40	0.0004
36	0.45	0.61	0.02	0.55	0.35	0.04	0.08	0.14	0.24	0.45	0.0003
38	0.51	0.68	0.03	0.62	0.40	0.04	0.09	0.16	0.28	0.50	0.0004
40	0.58	0.75	0.05	0.70	0.46	0.05	0.10	0.18	0.31	0.56	0.0004
42	0.64	0.83	0.15	0.79	0.51	0.06	0.11	0.20	0.35	0.64	0.0006
44	0.71	0.91	0.23	0.87	0.58	0.06	0.12	0.24	0.39	0.75	0.0005
46	0.79	1.00	0.25	0.97	0.64	0.07	0.13	0.27	0.44	0.84	0.0006
48	0.88	1.10	0.34	1.08	0.72	0.08	0.15	0.35	0.49	0.93	0.0005
50	0.97	1.21	0.46	1.19	0.80	0.08	0.16	0.45	0.54	1.01	0.0005
52	1.05	1.31	0.69	1.31	0.87	0.09	0.18	0.86	0.59	1.36	0.0005
54	1.17	1.43	0.76	1.44	0.97	0.10	0.20	1.12	0.66	1.49	0.0006
56	1.27	1.54	0.85	1.56	1.05	0.11	0.22	1.09	0.72	1.88	0.0006
58	1.39	1.67	0.93	1.71	1.16	0.12	0.24	1.16	0.78	2.42	0.0006
60	1.53	1.81	1.06	1.88	1.27	0.13	0.26	1.23	0.86	3.15	0.0006
Ave	0.81	1.02	0.38	1.00	0.66	0.07	0.14	0.49	0.45	1.07	0.0005

C.1.2 MSE(D_t) Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	0.09	16.11	0.17	0.04	2.06	0.03	0.00	4.23	0.05	1.84	0.10
24	0.06	15.85	0.88	0.02	2.89	0.01	0.07	1.58	0.01	0.28	0.05
34	0.13	13.55	0.26	0.08	3.27	0.04	0.15	1.50	0.02	1.18	0.02
36	0.07	14.78	0.20	0.08	2.95	0.03	0.31	3.43	0.01	0.63	0.01
38	0.04	16.43	0.13	0.18	1.66	0.02	0.05	0.56	0.01	0.16	0.23
40	0.12	14.45	0.01	0.42	0.86	0.02	0.03	0.74	0.07	1.07	0.01
42	0.10	14.45	0.80	0.21	0.88	0.02	0.07	0.37	0.08	0.70	0.02
44	0.24	16.59	0.88	0.52	0.76	0.05	0.11	0.58	0.12	2.32	0.02
46	0.42	17.69	0.67	0.76	1.03	0.03	0.13	0.17	0.23	2.18	0.00
48	0.35	19.26	0.58	1.12	1.09	0.03	0.09	0.44	0.30	2.23	0.09
50	0.47	21.47	0.13	1.87	1.66	0.04	0.05	0.28	0.31	4.87	0.15
52	0.87	25.55	0.70	2.23	2.87	0.16	0.39	0.96	0.42	11.36	0.10
54	1.10	24.90	0.76	1.57	3.59	0.12	1.10	1.51	0.50	3.66	0.09
56	2.19	28.56	1.03	2.71	5.43	0.09	0.83	1.49	0.63	3.05	0.49
58	3.05	33.70	1.22	5.53	9.25	0.30	0.83	2.06	0.93	5.50	0.25
60	2.66	35.43	1.08	6.01	11.23	0.43	0.77	1.43	0.88	10.55	0.23
Ave	0.75	20.55	0.59	1.46	3.22	0.09	0.31	1.33	0.29	3.22	0.12

C.1.3 Settling Time Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	0.00	0.40	0.00	0.00	0.00	0.00	0.00	0.40	0.00	0.02	0.00
24	0.00	0.40	0.00	0.00	0.40	0.00	0.00	0.00	0.00	0.00	0.00
34	0.00	0.40	0.00	0.00	0.40	0.00	0.00	0.00	0.00	0.01	0.00
36	0.00	0.40	0.00	0.00	0.40	0.00	0.00	0.40	0.00	0.00	0.00
38	0.00	0.40	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00
40	0.01	0.40	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.02	0.00
42	0.00	0.40	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00
44	0.00	0.40	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.03	0.00
46	0.01	0.40	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.03	0.00
48	0.02	0.40	0.00	0.03	0.03	0.00	0.00	0.01	0.01	0.04	0.00
50	0.02	0.40	0.01	0.04	0.04	0.00	0.00	0.01	0.01	0.40	0.00
52	0.01	0.40	0.02	0.03	0.04	0.00	0.00	0.02	0.01	0.40	0.00
54	0.02	0.40	0.03	0.03	0.06	0.00	0.02	0.04	0.02	0.05	0.00
56	0.02	0.40	0.02	0.03	0.06	0.00	0.01	0.02	0.01	0.04	0.00
58	0.02	0.40	0.02	0.04	0.08	0.00	0.02	0.03	0.02	0.05	0.00
60	0.03	0.40	0.03	0.06	0.10	0.01	0.03	0.03	0.02	0.06	0.00
Ave	0.01	0.40	0.01	0.02	0.10	0.00	0.01	0.06	0.01	0.07	0.00

C.2 Conveyor Output Shaft Results

C.2.1 MSE Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	0.38	0.51	0.12	0.41	0.24	0.03	0.11	0.13	0.20	0.40	0.0009
24	0.38	0.54	0.05	0.45	0.29	0.03	0.08	0.11	0.17	0.39	0.0009
34	0.44	0.64	0.03	0.52	0.31	0.04	0.10	0.12	0.19	0.49	0.0008
36	0.48	0.66	0.02	0.61	0.39	0.05	0.09	0.15	0.23	0.50	0.0006
38	0.56	0.80	0.02	0.65	0.43	0.05	0.11	0.15	0.32	0.60	0.0004
40	0.66	0.81	0.04	0.71	0.48	0.06	0.12	0.17	0.30	0.61	0.0007
42	0.70	0.95	0.12	0.77	0.55	0.07	0.13	0.20	0.33	0.75	0.0007
44	0.75	0.98	0.20	0.92	0.61	0.08	0.14	0.27	0.38	0.82	0.0008
46	0.85	1.14	0.22	0.98	0.69	0.09	0.16	0.27	0.42	0.97	0.0010
48	0.98	1.18	0.31	1.11	0.78	0.10	0.17	0.35	0.49	0.98	0.0012
50	1.03	1.33	0.42	1.16	0.86	0.12	0.19	0.45	0.52	1.08	0.0016
52	1.18	1.37	0.66	1.32	0.97	0.11	0.20	0.89	0.61	1.44	0.0018
54	1.28	1.51	0.73	1.52	1.05	0.13	0.23	1.20	0.72	1.61	0.0026
56	1.35	1.64	0.83	1.57	1.14	0.12	0.24	1.12	0.75	2.01	0.0036
58	1.50	1.78	0.95	1.73	1.31	0.13	0.26	1.21	0.83	2.56	0.0041
60	1.68	1.94	1.08	1.94	1.35	0.19	0.29	1.37	0.91	3.37	0.0047
Ave	0.89	1.11	0.36	1.02	0.72	0.09	0.16	0.51	0.46	1.16	0.00

C.2.2 MSE(D_t) Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V C/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	5.06	35.48	2.39	0.62	0.64	0.03	12.55	9.48	0.88	14.92	0.82
24	0.68	24.15	4.52	0.20	0.46	0.00	0.94	0.33	1.77	1.97	1.08
34	0.43	30.71	2.65	0.44	3.03	0.06	1.16	0.28	2.65	9.93	1.31
36	0.40	20.73	6.90	0.87	0.40	0.04	0.04	2.40	1.77	2.66	0.97
38	0.93	37.91	5.33	0.20	0.39	0.02	0.23	0.01	0.82	5.18	0.18
40	2.62	19.61	7.87	0.26	0.38	0.04	0.37	0.08	1.17	2.98	1.02
42	0.72	82.75	3.20	1.54	0.21	0.53	0.45	0.04	1.81	8.63	0.59
44	0.52	21.58	1.89	1.12	0.84	0.82	0.13	0.42	2.26	5.37	0.36
46	0.88	35.83	2.83	1.01	1.41	0.43	0.11	0.72	1.60	10.80	1.30
48	1.57	20.63	1.74	1.70	1.13	0.47	0.11	0.72	0.51	1.39	1.14
50	0.43	27.26	5.62	4.90	2.11	3.17	0.30	1.34	2.16	2.75	1.44
52	2.29	20.22	3.52	2.99	6.53	0.40	0.39	1.58	1.61	7.41	1.04
54	1.78	18.94	6.06	2.35	6.36	0.10	1.02	2.33	0.77	1.75	1.65
56	4.18	26.97	3.81	5.81	10.13	1.78	0.18	3.53	1.42	1.86	1.88
58	5.36	31.73	3.13	8.98	21.62	1.92	0.60	3.38	1.81	3.68	2.07
60	2.46	27.58	3.87	3.64	11.38	0.76	0.35	1.28	1.59	5.04	2.37
Ave	1.89	26.99	4.08	2.29	4.19	0.63	1.18	1.74	1.54	5.40	1.20

C.2.3 Settling Time Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Const I/P	Optimal Ramp I/P	Fuzzy	Learning
30	0.40	0.40	0.00	0.01	0.00	0.00	0.40	0.40	0.00	0.40	0.00
24	0.00	0.40	0.40	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
34	0.00	0.40	0.00	0.00	0.40	0.00	0.00	0.00	0.00	0.40	0.00
36	0.00	0.40	0.40	0.02	0.00	0.00	0.00	0.01	0.00	0.02	0.00
38	0.01	0.40	0.40	0.01	0.00	0.00	0.00	0.00	0.00	0.40	0.00
40	0.03	0.40	0.40	0.01	0.00	0.00	0.00	0.00	0.00	0.05	0.00
42	0.02	0.40	0.40	0.00	0.01	0.00	0.00	0.00	0.00	0.40	0.00
44	0.00	0.40	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.40	0.00
46	0.01	0.40	0.40	0.01	0.02	0.00	0.00	0.00	0.00	0.40	0.00
48	0.03	0.40	0.00	0.02	0.04	0.00	0.00	0.00	0.00	0.03	0.00
50	0.02	0.40	0.40	0.40	0.06	0.20	0.00	0.00	0.00	0.04	0.00
52	0.02	0.40	0.40	0.01	0.08	0.00	0.00	0.00	0.00	0.40	0.00
54	0.03	0.40	0.40	0.03	0.08	0.01	0.02	0.03	0.02	0.03	0.01
56	0.01	0.40	0.40	0.40	0.09	0.04	0.00	0.01	0.00	0.02	0.00
58	0.02	0.40	0.01	0.40	0.40	0.00	0.00	0.02	0.01	0.02	0.00
60	0.03	0.40	0.40	0.04	0.12	0.05	0.01	0.03	0.01	0.04	0.00
Ave	0.04	0.40	0.28	0.09	0.08	0.02	0.03	0.03	0.00	0.19	0.00

C.3 Dispenser Motor Results

C.3.1 MSE Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	0.03	0.08	0.02	0.02	0.02	0.0002	0.03	0.03	0.05	0.0025
24	0.04	0.10	0.02	0.02	0.02	0.0003	0.03	0.04	0.07	0.0025
34	0.05	0.12	0.03	0.03	0.03	0.0004	0.04	0.05	0.08	0.0025
36	0.06	0.15	0.03	0.04	0.04	0.0005	0.04	0.06	0.10	0.0024
38	0.08	0.17	0.04	0.05	0.05	0.0007	0.05	0.07	0.12	0.0025
40	0.10	0.20	0.04	0.07	0.06	0.0010	0.06	0.08	0.15	0.0027
42	0.12	0.24	0.05	0.09	0.08	0.0014	0.08	0.09	0.18	0.0026
44	0.16	0.27	0.06	0.11	0.11	0.0018	0.09	0.11	0.23	0.0026
46	0.20	0.32	0.08	0.15	0.14	0.0024	0.11	0.13	0.28	0.0027
48	0.27	0.38	0.11	0.19	0.19	0.0032	0.13	0.15	0.32	0.0025
50	0.35	0.43	0.15	0.24	0.24	0.0042	0.16	0.19	0.36	0.0024
52	0.55	0.50	0.18	0.29	0.38	0.0055	0.20	0.26	0.43	0.0025
54	0.82	0.58	0.29	0.45	0.56	0.0072	0.24	0.41	0.52	0.0025
56	1.17	0.68	0.45	0.78	0.79	0.0093	0.30	0.76	0.66	0.0024
58	1.54	0.88	0.67	0.95	0.99	0.0119	0.45	1.30	0.88	0.0025
60	1.76	1.16	1.09	1.68	1.04	0.0353	0.89	1.75	1.45	0.0015
Ave	0.46	0.39	0.21	0.32	0.66	0.01	0.18	0.34	0.37	0.0025

C.3.2 MSE(D_t) Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	1.93	80.45	9.78	0.78	0.03	0.05	0.16	0.79	10.72	0.82
24	1.43	90.43	11.54	0.88	0.05	0.06	0.14	0.81	12.34	0.57
34	2.37	105.42	12.78	0.76	0.07	0.07	0.20	1.01	15.30	0.56
36	1.97	122.93	15.47	1.14	0.12	0.09	0.29	1.37	19.03	0.48
38	1.86	137.40	14.85	1.46	0.16	0.09	0.24	1.52	21.13	0.46
40	1.70	155.34	15.35	2.50	0.19	0.10	0.22	1.77	25.58	0.54
42	1.71	187.42	15.59	4.62	0.44	0.13	0.68	1.91	25.15	0.39
44	1.20	200.73	14.29	8.80	0.73	0.15	0.73	1.46	29.91	0.39
46	1.23	215.13	13.56	7.53	0.89	0.14	0.68	2.17	31.76	0.42
48	1.07	248.90	12.90	8.12	1.93	0.17	1.14	2.87	35.36	0.27
50	0.75	266.61	9.89	11.15	4.16	0.21	1.85	5.11	39.83	0.39
52	6.34	296.04	12.23	15.87	33.71	0.25	2.77	7.23	75.64	0.41
54	73.55	307.28	3.08	18.13	144.75	0.31	4.82	8.06	118.95	0.22
56	269.37	327.52	6.42	154.61	341.09	0.27	5.63	76.51	141.27	0.18
58	608.56	402.19	63.57	290.20	586.73	0.33	5.96	393.02	163.11	0.20
60	735.46	441.12	273.86	720.53	609.61	3.45	1.76	598.86	216.03	0.16
Ave	106.91	224.06	31.57	77.94	3.22	0.37	1.71	69.03	61.32	0.40

C.3.3 Settling Time Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	0.08	0.40	0.12	0.03	0.00	0.00	0.00	0.00	0.28	0.00
24	0.07	0.40	0.12	0.04	0.00	0.00	0.00	0.01	0.33	0.00
34	0.09	0.40	0.13	0.00	0.00	0.00	0.00	0.02	0.32	0.00
36	0.09	0.40	0.14	0.09	0.00	0.00	0.00	0.14	0.34	0.00
38	0.08	0.40	0.14	0.10	0.00	0.00	0.00	0.13	0.33	0.00
40	0.07	0.40	0.14	0.09	0.00	0.00	0.00	0.14	0.39	0.00
42	0.08	0.40	0.14	0.10	0.01	0.00	0.01	0.15	0.40	0.00
44	0.07	0.40	0.14	0.06	0.02	0.00	0.01	0.16	0.40	0.00
46	0.06	0.40	0.12	0.07	0.02	0.00	0.01	0.15	0.40	0.00
48	0.07	0.40	0.12	0.10	0.03	0.00	0.02	0.14	0.40	0.00
50	0.00	0.40	0.12	0.11	0.05	0.00	0.03	0.15	0.40	0.00
52	0.08	0.40	0.16	0.11	0.13	0.00	0.03	0.15	0.40	0.00
54	0.19	0.40	0.16	0.09	0.23	0.00	0.08	0.15	0.40	0.00
56	0.40	0.40	0.12	0.40	0.40	0.00	0.08	0.38	0.40	0.00
58	0.40	0.40	0.40	0.40	0.40	0.00	0.09	0.40	0.40	0.00
60	0.40	0.40	0.40	0.40	0.40	0.09	0.04	0.40	0.40	0.00
Ave	0.14	0.40	0.17	0.14	0.10	0.01	0.03	0.17	0.37	0.00

C.4 Dispenser Output Shaft Results

C.4.1 MSE Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	0.03	0.10	0.02	0.02	0.01	0.0003	0.03	0.03	0.07	0.0046
24	0.04	0.12	0.02	0.02	0.02	0.0003	0.03	0.04	0.08	0.0044
34	0.05	0.15	0.03	0.03	0.02	0.0004	0.04	0.04	0.10	0.0043
36	0.06	0.17	0.03	0.04	0.03	0.0007	0.05	0.05	0.12	0.0044
38	0.08	0.23	0.03	0.05	0.04	0.0009	0.05	0.07	0.15	0.0046
40	0.10	0.23	0.04	0.07	0.06	0.0014	0.06	0.08	0.18	0.0048
42	0.12	0.27	0.05	0.09	0.08	0.0017	0.08	0.09	0.22	0.0034
44	0.16	0.31	0.06	0.12	0.11	0.0021	0.09	0.11	0.28	0.0036
46	0.20	0.36	0.08	0.15	0.14	0.0029	0.11	0.13	0.33	0.0039
48	0.27	0.42	0.11	0.20	0.19	0.0040	0.14	0.15	0.37	0.0035
50	0.36	0.47	0.15	0.25	0.25	0.0053	0.17	0.19	0.42	0.0034
52	0.55	0.55	0.18	0.30	0.39	0.0069	0.21	0.26	0.50	0.0038
54	0.84	0.63	0.29	0.46	0.59	0.0091	0.25	0.41	0.58	0.0036
56	1.22	0.74	0.45	0.81	0.83	0.0122	0.31	0.77	0.73	0.0035
58	1.62	0.94	0.68	1.00	1.04	0.0174	0.46	1.33	0.96	0.0037
60	1.84	1.23	1.13	1.74	1.09	0.0459	0.92	1.80	1.54	0.0024
Ave	0.47	0.43	0.21	0.33	0.31	0.01	0.19	0.36	0.41	0.00

C.4.2 MSE(D_t) Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	1.29	89.75	10.32	1.32	0.07	0.28	0.48	0.26	9.44	0.23
24	0.99	96.46	11.85	1.39	0.08	0.37	0.52	0.45	10.77	0.24
34	1.70	115.77	13.89	1.59	0.04	0.58	0.43	0.61	14.42	0.30
36	1.47	126.90	15.39	2.05	0.05	1.14	0.42	0.96	17.01	0.31
38	1.41	143.95	15.06	2.37	0.04	0.13	0.45	0.90	18.83	0.29
40	1.16	160.99	15.59	3.78	0.13	0.03	0.33	1.13	22.83	0.29
42	1.06	191.77	15.96	6.48	0.32	0.16	0.84	1.48	21.17	1.81
44	0.66	215.90	15.26	11.84	0.90	0.65	1.57	1.32	30.44	1.69
46	0.92	227.88	14.84	11.63	1.21	0.79	1.57	1.78	31.03	1.66
43	0.67	256.58	13.00	10.12	1.79	0.86	1.41	2.42	32.38	1.43
50	0.44	275.59	9.96	13.76	3.87	0.94	1.95	4.77	35.16	1.93
52	10.42	306.26	13.03	22.61	42.92	0.85	4.96	7.06	75.39	1.63
54	94.89	328.12	2.29	22.28	171.07	1.25	8.47	7.64	116.63	1.24
56	313.53	341.45	8.60	173.95	383.30	0.96	8.04	84.27	138.11	1.21
58	666.59	424.51	72.43	318.48	627.75	0.25	8.25	412.05	161.06	1.08
60	740.64	456.80	282.53	745.65	605.70	6.18	2.38	608.37	217.67	1.03

Ave	114.86	234.92	33.12	84.33	114.95	0.96	2.63	70.97	59.50	1.02

C.4.3 Settling Time Results

Unit Rate	PID	Lag Lead	PID + F/F	Speed + Position	Flux V O/L	Flux V C/L	Direct Digital	Optimal Ramp I/P	Fuzzy	Learning
30	0.03	0.40	0.12	0.06	0.00	0.00	0.00	0.00	0.27	0.00
24	0.05	0.40	0.12	0.06	0.00	0.00	0.00	0.00	0.31	0.00
34	0.08	0.40	0.13	0.08	0.00	0.00	0.00	0.01	0.28	0.00
38	0.08	0.40	0.14	0.10	0.00	0.00	0.00	0.02	0.31	0.00
38	0.07	0.40	0.14	0.11	0.00	0.00	0.01	0.02	0.30	0.00
40	0.06	0.40	0.13	0.10	0.00	0.00	0.00	0.02	0.29	0.00
42	0.06	0.40	0.14	0.11	0.00	0.00	0.01	0.15	0.38	0.00
44	0.00	0.40	0.14	0.07	0.01	0.00	0.01	0.15	0.38	0.00
46	0.05	0.40	0.12	0.10	0.02	0.00	0.01	0.15	0.39	0.00
48	0.00	0.40	0.12	0.11	0.03	0.00	0.02	0.13	0.40	0.00
50	0.00	0.40	0.11	0.12	0.05	0.00	0.03	0.15	0.40	0.00
52	0.09	0.40	0.15	0.12	0.13	0.00	0.03	0.15	0.40	0.00
54	0.18	0.40	0.15	0.10	0.24	0.00	0.08	0.15	0.40	0.00
56	0.40	0.40	0.14	0.40	0.40	0.00	0.08	0.39	0.40	0.00
58	0.40	0.40	0.40	0.40	0.40	0.00	0.08	0.40	0.40	0.00
60	0.40	0.40	0.40	0.40	0.40	0.15	0.05	0.40	0.40	0.00

Ave	0.12	0.40	0.17	0.15	0.11	0.01	0.03	0.14	0.36	0.00

C.5 Cost Function Results

Controller	Conveyor			Dispenser		
	Motor	Output	Total	Motor	Output	Total
PID	1.00	2.16	1.00	1.00	1.07	1.00
PID 2	1.00	2.78	1.00	1.00	1.08	1.00
Lag/Lead	20.63	26.94	15.05	2.09	2.20	2.07
Velocity F/F	0.54	3.29	1.01	0.29	0.31	0.29
Speed + Position	1.76	2.62	1.39	0.73	0.79	0.73
Flux Vector O/L	3.42	4.37	2.46	1.01	1.07	1.00
Flux Vector C/L	0.11	0.65	0.24	0.00	0.01	0.01
Optimal	0.33	1.27	0.42	0.63	0.65	0.62
Direct Digital	0.35	1.46	0.57	0.02	0.03	0.02
Fuzzy	3.54	5.76	2.94	0.58	0.56	0.55
Learning	0.09	0.90	0.26	0.00	0.01	0.01

Appendix D

Motion Control Card Specifications

The motion control card is a two axis interface card that resides in a single ISA slot within the PC. The hardware consists of a programmable interrupt controller that is capable of interrupting the host PC at periods of 0.1ms increments. Two 32 bit counters provide an interface to the encoder feedback signal. Software allows configuration of the counters so that receipt of a marker pulse will generate an interrupt to latch the value currently in the counter. Two 12 bit DAC's provide a $\pm$10V analogue demand signal. Digital I/O on the card allows the implementation of enabling signals. The pin designations are provided in the following table.

The software consists of a TSR function that is loaded at boot up. A C function, *open_motion*, opens the card and provides access from within a C program. The card is closed by use of function *close_motion*.

Communication with the card at run time is through two functions:

long *read_motion* (long *command*, long *channel*)

and

void *write_motion* (long *command*, long *channel*)

Pin No.	Encoder		Output	
	Signal	Function	Signal	Function
1	Ai	A phase input	+15V	+15V supply
2	Bi	B phase input	0V	0V common
3		Reserved	-15V	-15V supply
4	Mi	Marker input	enC1	Motor enable 1(coll)
5	Prb	Probe input	enE1	Motor enable 1(emit)
6	nAi	Inverse A input	An1	Analogue output 1
7	nBi	Inverse B input	An2	Analogue output 2
8		Reserved	enC2	Motor enable 2 (coll)
9	nMi	Inverse marker i/p	enE2	Motor enable 2 (emit)
10	nPrb	Inverse probe i/p		
11	+12V	+12V supply		
12	+5V	+5V supply		
13	0V	0V common		
14	-5V	-5V supply		
15	-12V	-12V supply		

Table D.1: Interface card connections

Channel is an integer that selects the channel upon which the command is to be performed. *Write_motion* adjusts the card as required, whereas *read_motion* returns a long representing the desired parameter. *Command* is one of a number of defined strings that allow access to the various parts of the card such as the interrupt controller, the encoder counters and the DAC's. Using this structure the command to read the 32 bit value on the encoder counter of axis 0 is:

count = read_motion(AXIS_32,0L)

and to write 2000mV to the DAC for axis 1 is:

write_motion(DAC_MV,1L,2000L)

References

[1] A.D. Barton, P.L. Lewin, and D.J. Brown. High speed indexing of a chain based conveyor. *Fifth International Conference on Factory 2000*, pages 491–495, 1997.

[2] A.D. Barton, P.L. Lewin, and D.J. Brown. Analysis and control of a high speed indexing chain conveyor. *Eighth International Conference on Electrical Machines and Drives*, pages 314–318, 1997.

[3] A. Harrison, J. W. Hayes, and A. W. Roberts. The feasibility of high speed narrow belt conveyors for bulk solids handling. *The Institution of Engineers, Australia Mechanical Engineering Transactions*, pages 190–200, 1982.

[4] D.J. Hind and R.C. Thurlow. Monitoring and control of conveyors and haulages. *International Conference on Mines Transport*, 1(11):1–12, 1983.

[5] A. Harrison. Non-linear processes and chaos in belt conveyor systems. *Bulk Solids Handling*, 11(1):147–155, 1991.

[6] A.W. Roberts. Advances in the design of mechanical conveyors. *Bulk Solids Handling*, 14(2):255–281, 1994.

[7] Editorial. Conveyor systems. *Modern Materials Handling*, 38(11):50–57, August 1983. Part of a series on the Automated Factory.

[8] J. Cooper. Non-rusting plastic ball bearings find favour for food conveyors. *Anti-Corrosion Methods and Materials*, 33(3):11, 1986.

[9] P. Dvorak. Update on chains and sprockets. *Machine Design*, pages 46–54, August 1989.

[10] H. Jacobson. Proportional controls boost performance of walking beam conveyor. *Hydraulics and Pneumatics*, pages 29,30,54, November 1993.

[11] P.I.F. Thomas. Conveying and control for automated manufacture. *Proceedings of the 3rd International Conference on Automated Materials Handling*, pages 163–173, 1986.

[12] R. Holzhauer. Unit/pallet handling conveyors. *Plant Engineering*, 50(6):44–48, 1996.

[13] R.A. Kulwiec, editor. *Materials Handling Handbook*. Wiley, second edition, 1985.

[14] M. Ince. Designer delivery systems. *The Engineer*, pages 34–35, February 1988.

[15] E.J. Muth and J.A. White. Conveyor theory - a survey. *AIIE Transactions*, 11(4):270–277, 1979.

[16] A.K. Gunal, S. Sadakane, and E.J. Williams. Modelling of chain conveyors and their equipment interfaces. *Proceedings of the Winter Simulation Conference*, pages 1107–1114, 1996.

[17] R. Greenhill. Cracking the wip. *Manufacturing Engineer*, pages 55–57, April/May 1991.

[18] Gore Consulting Group. Opportunities for advanced sensors in the food and drink industry - a survey. *Measurement and Control*, 21:197–201, 1988.

[19] H.A. Slight. Instrumentation in the food industry. *Measurement and Control*, 24:197–199, 1991.

[20] B. Dobson. No crumbs. *Drives and Controls*, 14(10):60, November/December 1998.

[21] W. H. Schwartz. Move it right - assembly positioning techniques. *Assembly Engineering*, 33(5):30–33, 1990.

[22] A.D. Barton, P.L. Lewin, and D.J. Brown. Practical implementation of a real-time iterative learning position controller. Submitted to the International Journal of Control, 1999.

[23] A.D. Barton and P.L. Lewin. Improving the control of chain conveyor manufacturing systems. To be published in the proceedings of the Second International Conference on the Control of Industrial Process, 1999.

[24] S. Abrate. Vibrations of belts and belt drives. *Mechanisms and Machine Theory*, 27(6):645–659, 1992.

[25] R.A. Morrison. Polygonal action in chain drives. *Machine Design*, 24(9):155–159, 1952.

[26] S. Mahalingan. Polygonal action in drive chains. *Journal of the Franklin Institute*, pages 23–28, January 1958.

[27] J. Golten and A. Verwer. *Control System Design and Simulation.* McGraw-Hill Book Company, 1991.

[28] N.C. Barford. *Experimental Measurements: Precision, Error and Truth.* Addison-Wesley, first edition, 1967.

[29] D.P. Atherton. *Nonlinear Control Engineering.* Van Nostand Reinhold, 1982.

[30] G. Younkin. Modelling machine tool feed servo drives using simulation techniques to predict performance. *IEEE Conference Record - IAS Annual Meeting*, 2:1699–1706, 1989.

[31] I. Colak, S.D. Garvey, and M.T. Wright. Simulation of induction machines using phase variables and inverse inductance matrix. *International Journal of Electrical Engineering Education*, 32:354–365, 1995.

[32] S.D.T. Robertson and K.M. Hebbar. A digital model for three-phase induction machines. *IEEE Transactions on Power Apparatus and Systems*, PAS-88(11):1624–1634, 1969.

[33] H.E. Jordan. Analysis of induction machines in dynamic systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-84(11):1080–1088, 1965.

[34] P.C. Krause. Simulation of symmetrical induction machinery. *IEEE Transactions on Power Apparatus and Systems*, PAS-84(11):1038–1053, 1965.

[35] C.V. Jones. *Unified Theory of Electrical Machines.* Butterworth and Co., 1967.

[36] P.C. Krause. The method of multiple reference frames applied to the analysis of symmetrical induction machinery. *IEEE Transactions on Power Apparatus and Systems*, PAS-87(1):218–227, 1968.

[37] A.K. DeSarkar and G.J. Berg. Digital simulation of three phase induction motors. *IEEE Transactions on Power Apparatus and Systems*, PAS-89(6):1031–1037, 1970.

[38] N.N. Hancock. *Matrix Analysis of Electrical Machinery.* Pergamon Press, second edition, 1974.

[39] B.K. Bose. *Power Electronics and AC Drives.* Prentice Hall, 1987.

[40] P. Vas. *Vector Control of AC Machines.* Oxford University Press, 1990.

[41] M. L. Macdonald and P. C. Sen. Control loop study of induction motor drives using dq model. *IEEE Transactions on Industrial Electronics and Control Instrumentation*, 26(4):237–243, Nov 1979.

[42] G.R. Slemon. Modelling of induction machines for electric drives. *IEEE Transaction on Industry Applications*, 25(6):1126–1131, 1989.

[43] M.H. Rashid. *Power Electronics - Circuits, Devices and Applications.* Prentice Hall, second edition, 1993.

[44] V.R. Stefanovic and T.H. Barton. The speed-torque transfer function of electric drives. *IEEE Transactions on Industry Applications*, IA-12(5):428–436, 1977.

[45] J. Reinicke-Murmann and P. Kreuter. Analysis and development of a camshaft drive system using a computer simulation. *SAE Special Publications*, pages 13–21, 1990.

[46] C.D. Mote. Dynamic stability of an axially moving band. *Journal of the Franklin Institute*, 238(5):329–346, 1968.

[47] S.R. Turnbull and J.N. Fawcett. Dynamic behaviour of roller chain drives. *Institute of Mechanical Engineering - Mechanisms*, pages 29–35, 1972.

[48] N.E. Hollingworth and D.A. Hills. Forces in a heavy-duty chain drive during articulation. *Proceedings of Institute of Mechanical Engineers*, 200(5):367–374, 1986.

[49] S.T. Ariaratram and S.F. Asokanthan. Dynamic stability of chain drives. *Transactions of the ASME*, 109:412–418, 1987.

[50] D. Jacobson and J. Maddocks. On the dynamics of chains. *Society for Industrial and Applied Mathematics*, 52(6):1563–1583, 1992.

[51] E.N. Martinez. Effective mass of a classical linear chain. *American Association of Physics Teachers*, 61(12):1102–1110, 1993.

[52] K. Dutten, S. Thompson, and B. Barraclough. *The Art of Control.* Addison-Wesley Longman, 1997.

[53] J.G. Zeigler and N.B. Nichols. Optimum settings for automatic controllers. *Transactions of the ASME*, pages 759–768, November 1942.

[54] K.J. Astrom and T. Hagglund. Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica*, 20(5):645–651, 1984.

[55] C.C. Hang, K.J. Aström, and W.K. Ho. Refinements of the Zeigler-Nichols tuning formula. *IEE Proceedings - D*, 138(2):111–118, 1991.

[56] K.J. Aström, C.C. Hang, P. Persson, and W.K. Ho. Towards intelligent PID control. *Automatica*, 28:1–9, 1992.

[57] T.S. Schei. A method for closed loop automatic tuning of PID controllers. *Automatica*, 28(3):587–591, 1992.

[58] R.C. Dorf. *Modern Control Systems.* Addison-Wesley, sixth edition, 1992.

[59] H. Lauer, R.N. Resnik, and L.E. Matson. *Servomechanism Fundamentals.* McGraw-Hill, second edition, 1960.

[60] A. DePaor. A fiftieth anniversary celebration of the Zeigler-Nichols PID controller. *International Journal of Electrical Engineering Education*, 30:303–316, 1993.

[61] R. D. Lorenz and K. W. Van Patten. High resolution velocity estimation for all digital ac servo drives. *IEEE Transactions on Industry Applications*, 24(4):701–705, 1991.

[62] G.F. Franklin, J.D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems.* Addison Wesley, third edition, 1994.

[63] H. Kim, J. Choi, and S. Sul. Accurate position control for ac servo motor using novel speed estimator. *IECON Proceedings (Industrial Electronics Conference)*, 1:627–632, 1995.

[64] A. Hughes. *Electric Motors and Drives.* Butterworth-Heinemann, second edition, 1993.

[65] R.B. Sepe and J. H. Lang. Implementation of discrete time field oriented current control. *IEEE 8th Applied Power Electronics Conference*, pages 135–140, 1993.

[66] U. Dirker. A.C. flux vector control improves induction motor performance. *Power Conversion and Intelligent Motion*, pages 52–59, June 1995.

[67] P. Mota, J. P. Rognon, and H. Le-Huy. Digital position servo system: A state variable feedback system. *IEEE Transaction on Industry Applications*, IA-20(6):1473–1481, 1984.

[68] M. Tomizuka. Zero phase error tracking algorithm for digital control. *ASME Journal of Dynamic Systems, Measurement and Control*, 109:65–68, 1987.

[69] M. Tomizuka. On the design of digital tracking controllers. *ASME Journal of Dynamic Systems, Measurement, and Control*, 115:412–418, 1993.

[70] G.S. Virk. *Digital Computer Control Systems.* Macmillan Education, first edition, 1991.

[71] G.H. Hostetter. *Digital Control System Design.* Holt, Reinhart and Winston Inc., first edition, 1988.

[72] P.J. King and E.H. Mamdani. The application of fuzzy control systems to industrial processes. *Automatica*, 13:235–242, 1977.

[73] M. Sugeno. An introductory survey of fuzzy control. *Information Sciences*, 36:59–83, 1985.

[74] J. Jantzen. Fuzzy control. Lecture notes, Electric Power Engineering Department, Technical University of Denamrk, DK-2800, Lyngby, 1994. Revision 4.

[75] S. Tzafestas and N.P. Papinikolopoulos. Incremental fuzzy expert PID control. *IEEE Transactions on Industrial Electronics*, 37(5):365–371, 1990.

[76] S.J. Qin. Auto tuned fuzzy logic control. *Proceedings of the American Control Conference*, pages 2465–2469, 1994.

[77] W. Z. Qiao and M. Mizumoto. PID type fuzzy controller and parameters adaptive method. *Fuzzy Sets and Systems*, 78:23–35, 1996.

[78] M. Santos, S. Dormido, and J.M. Cruz. Derivative action in PID fuzzy controllers. *Cybernetics and Systems*, 27:413–424, 1996.

[79] G.K.I. Mann, B.G. Hu, and R.G. Gosine. Fuzzy PID controller structures. *Canadian Conference on Electronics and Computer Engineering*, 2:788–791, 1997.

[80] C.W. de Silva. Simulation studies of an analytical fuzzy tuner for a PID servo. *Proceedings of the American Control Conference*, 2:2100–2105, 1991.

[81] S. He, S. Tan, and F. Xu. Fuzzy self-tuning of pid controllers. *Fuzzy Sets and Systems*, 56:37–46, 1993.

[82] G. Zhou and J.D. Birdwell. Fuzzy logic based PID autotuner design using simulated annealing. *Proceedings of the IEEE/IFAC Joint Symposium on Computer Aided Manufacturing*, pages 67–72, 1994.

[83] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronic systems. *Proceedings of the 23rd Conference on Decision and Control*, pages 1064–1069, 1984.

[84] K. Furuta and M. Yamakita. The design of a learning control system for multivariable systems. *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 371–376, 1987.

[85] F. Padieu and R. Su. An H-infinity approach to learning control system. *International Journal of Adaptive Control and Signal Processing*, 4:465–474, 1990.

[86] Z. Geng, R. Carroll, and J. Xie. Two-dimensional model and algorithm analysis for a class of iterative learning control systems. *International Journal of Control*, 52(4):833–862, 1990.

[87] D.H. Owens. Iterative learning control - convergence using high gain feedback. *Proceedings of the 31st Conference on Decision and Control*, pages 2545–2546, 1992.

[88] N. Amann, D.H. Owens, and E.Rogers. Iterative learning control for discrete-time systems with exponential rate of convergence. *IEE Proceedings on Control Theory Applications*, 143(2):217–224, 1996.

[89] N. Amann, D.H. Owens, and E. Rogers. Iterative learning control using optimal feedback and feedforward actions. *International Journal of Control*, 65(2):277–293, 1996.

[90] S. Kawamura, F. Miyazaki, and S. Arimoto. Realization of robot motion based on a learning method. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):126–134, 1988.

[91] D. Wang. A simple iterative learning controller for manipulators with flexible joints. *Automatica*, 31(9):1341–1344, 1995.

[92] D. Kim and S.Kim. An iterative learning control method with application for CNC machine tools. *IEEE Transactions of Industry Applications*, 32(1):66–72, 1996.

[93] H.J. Park, H.S. Cho, and S.R. Oh. A discrete iterative learning control method with application to electric servo motor control. *Proceedings of the American Control Conference*, 3:2640–2645, 1991.

[94] J.J. Lee and J.W. Lee. Design of iterative learning controllers with VCR servo system. *IEEE Trans on Consumer Electronics*, 39(1):13–24, 1993.

[95] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Iterative learning control: A survey and new results. *Journal of Robotic Systems*, 9(5):563–594, 1992.

[96] R. Horowitz. Learning control of robot manipulators. *ASME Journal of Dynamic Systems, Measurement, and Control*, 115:402–411, 1993.

[97] D.H. Owens, N. Amann, and A. Wahl. Studies in optimization-based iterative learning control. Internal report, Centre for Systems and Control Engineering, Exeter University, North Park Road, Exeter EX4 4QF, 1993.

[98] M. Harley. Understanding repeatability and accuracy. *Drives and Controls*, page 24, Nov 1998.

[99] S. Ganti. The secrets of precision indexing. *American Machinist*, pages 66–68, September 1995.

[100] M. Tomizuka, J. Hu, T. Chiu, and T. Kamano. Synchronisation of two motion control axes under adaptive feedforward control. *Journal of Dynamic Systems, Measurement and Control*, 114:196–203, 1992.

[101] D.M. Tome and J. Tal. Master/slave motion synchronization for industrial processes. *Proceedings of the Annual Symposium on Incremental Motion Control*, pages 61–68, 1986.

[102] R. Scott. Axis synchronisation by encoder following. *Drives and Controls*, pages 70–72, September 1991.

[103] Y. Koren. Cross coupled biaxial computer control for manufacturing systems. *ASME Journal of Dynamic Systems, Measurement and Control*, 102:265–272, 1980.

[104] P.K. Kulkarni and K. Srinivasan. Cross coupled compensators for contouring control of mult-axial machine tools. *Manufacturing Engineering Trans. NAMRC XIII Proceedings*, pages 558–566, 1985.

[105] P. Moore and C.M. Chen. The synchronisation of servo drives using fuzzy logic control. *IEE Colloquium (Digest)*, 176(5):1–9, 1994.

[106] M. Tomizuka, D. Dornfeld, X.Q. Bian, and H.G. Cai. Experimental evaluation of the preview servo scheme for a two axis positioning system. *ASME Journal of Dynamic Systems, Measurement, and Control*, 106:1–5, 1984.

[107] J. Hu, T. Chiu, and M. Tomizuka. On motion synchronisation of two axes systems. *Monitoring and Control for Manufacturing Processes*, pages 267–282, 1990.

[108] A. Jeffrey. *Mathematics for Engineers and Scientists.* Van Nostrand Reinhold, fourth edition, 1989.