

**Simulation and Adaptive Control for Valve Friction  
Problems in a Paper Mill**

by

Jan A. Bergstrom

B.A.Sc., University of Waterloo, 1993

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF  
**Master of Applied Science**

in

THE FACULTY OF GRADUATE STUDIES  
(Department of Electrical and Computer Engineering)

we accept this thesis as conforming  
to the required standard

**The University of British Columbia**

December 1998

© Jan A. Bergstrom, 1998

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical & Computer Engineering

The University of British Columbia  
Vancouver, Canada

Date Dec 22 / 98

# Abstract

This thesis attempts to address issues of poor performance in a paper mill. It looks at two specific issues; the need to perform offline performance evaluations using simulation tools, and the specific problems associated with friction in control valves.

The first issue is addressed through the development of an object oriented component library for the dynamic simulation of complex systems in the mill. Modelling of a paper machine approach system is used as the basis for developing a structure that could later be expanded to include blocks for simulation of all parts of the paper mill. The simulation is used to identify potential problems, and to evaluate some possible solutions. It is also used to evaluate the effects on the final product of valve friction in a particular control loop.

The second issue involves the oscillatory behaviour generated when a valve with friction is placed under conventional automatic control. The nature of friction and its effects on a typical pneumatic valve are analysed. Various macroscopic friction characteristics are used to create an adaptive control algorithm that can be used on the valve without a requirement for high-bandwidth control. The strategy is tested on a simulation of a pneumatic valve with friction and compared to conventional PI control.

# Contents

<b>Abstract</b>	<b>ii</b>
<b>Contents</b>	<b>iii</b>
<b>List of Tables</b>	<b>v</b>
<b>List of Figures</b>	<b>vi</b>
<b>Acknowledgements</b>	<b>viii</b>
<b>Dedication</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Paper Making and the Paper Machine . . . . .	2
1.2 Designing Blocks for Simulation of a Paper Machine . . . . .	4
1.3 Designing a Controller for Valves with Friction . . . . .	5
1.4 Industrial Collaboration . . . . .	6
<b>2 Simulation Blocks for the Modelling of a Paper Machine</b>	<b>8</b>
2.1 OMOLA and OmSim . . . . .	9
2.2 The Pulp Library . . . . .	12
2.2.1 Terminal Definition . . . . .	12
2.2.2 Acausality . . . . .	13

2.2.3	Modularity . . . . .	15
2.2.4	Component Structure . . . . .	17
2.2.5	Complex Objects . . . . .	19
2.3	Approach System Simulation . . . . .	21
2.3.1	General Consistency Control . . . . .	22
2.3.2	Effects of Valve Friction on Consistency Control . . . . .	24
2.4	Summary . . . . .	27
<b>3</b>	<b>Control of Pneumatic Valves with Friction</b>	<b>28</b>
3.1	Friction Models . . . . .	29
3.1.1	Adverse Effects of Friction . . . . .	36
3.2	The Pneumatic Valve . . . . .	38
3.3	Problem Definition . . . . .	42
3.4	Current Friction Compensation Techniques . . . . .	45
3.5	Modelling . . . . .	56
3.6	Control Design . . . . .	60
3.6.1	A Careful Look at the Motion of a Valve . . . . .	60
3.6.2	How This Affects the Design . . . . .	63
3.6.3	The Control Algorithm . . . . .	64
3.7	Controller Performance . . . . .	79
3.8	Summary . . . . .	84
<b>4</b>	<b>Conclusions</b>	<b>94</b>
	<b>Bibliography</b>	<b>97</b>

# List of Tables

3.1	The nominal model parameters . . . . .	59
-----	--	----

# List of Figures

1.1	Main parts of a paper mill . . . . .	2
1.2	The approach system . . . . .	3
2.1	A simple OMOLA code sample . . . . .	11
2.2	Structure of fluid dynamics terminal. . . . .	13
2.3	Structure of mass flux terminal. . . . .	14
2.4	Full terminal structure. . . . .	15
2.5	The endpoints of a flow section . . . . .	16
2.6	Internal subcomponents for a two port flow device . . . . .	16
2.7	Class hierarchy for the fluid dynamic modules . . . . .	18
2.8	OMOLA code for implementing plug flow equations. . . . .	19
2.9	Modular construction of the deculator . . . . .	20
2.10	OmSim model of approach system . . . . .	21
2.11	Step response of the approach system model . . . . .	23
2.12	The dynamic response for consistency disturbances . . . . .	24
2.13	The dynamic response for valve oscillations . . . . .	26
3.1	Friction in a free-body diagram . . . . .	29
3.2	Simple Coulomb friction vs. velocity . . . . .	30
3.3	Static and Coulomb friction . . . . .	31
3.4	Complex friction model . . . . .	32

3.5	Valve and actuator schematic . . . . .	39
3.6	Block diagram of valve and controller . . . . .	41
3.7	Typical friction induced oscillations . . . . .	42
3.8	The performance of the Knocker . . . . .	52
3.9	A block diagram of the valve model. . . . .	57
3.10	Measured and simulated chamber pressure response . . . . .	58
3.11	Simulated step response of a pneumatic valve . . . . .	61
3.12	Response of the valve to different pulse sizes . . . . .	64
3.13	Graph of pulse magnitude vs. slide distance . . . . .	65
3.14	Graph of pulse duration vs. slide distance . . . . .	66
3.15	Characterized control sequence and system response . . . . .	68
3.16	A typical consistency regulation loop . . . . .	79
3.17	A block diagram of the full consistency control loop . . . . .	80
3.18	A block diagram of the simplified loop used in the simulations . . . . .	80
3.19	Evaluation of on-line parameter estimates . . . . .	84
3.20	Test 1, long step disturbances, adaptive control . . . . .	86
3.21	Test 1, long step disturbances, PI control . . . . .	87
3.22	Test 2, short step disturbances, adaptive control . . . . .	88
3.23	Test 2, short step disturbances, PI control . . . . .	89
3.24	Test 3, sinusoidal disturbance, adaptive control . . . . .	90
3.25	Test 3, sinusoidal disturbance, PI control . . . . .	91
3.26	Test 4, increased friction, adaptive control . . . . .	92
3.27	Test 4, increased friction, PI control . . . . .	93

# Acknowledgements

I would like to thank my supervisor, Dr. Guy Dumont, for all his support and guidance in this thesis, especially in directing me towards a topic that proved to be both challenging and rewarding. I appreciate the great opportunity he has provided to work with wonderful people from all over the world, including sending me abroad for my own work. I would also like to thank Dr. Michael Davies and Dr. Ezra Kwok, my de facto co-supervisors from the Control Group, for their suggestions and guidance.

I would like to express my gratitude to Professor Karl Johan Åström and the rest of the Department of Automatic Control, for hosting me for two very informative weeks in Lund, Sweden and for all their help and suggestions with regard to my work. A special thanks to Jonas, Tomas, and Sven Erik for providing very helpful support on OmSim while I was in Lund, and from afar, when I was back in Canada. Thank you also to everyone who so enthusiastically helped me meet friends and colleagues of my late grandfather, an unexpected treat on my trip.

Many thanks go to all my colleagues at the Pulp and Paper Centre for all their help and support. Specifically I would like to thank Greg, Kayvan, Ali, and Lahoucine for listening to and providing suggestions for all my problems along the way. I would also like to thank Fredrik, Casper, and Denis for providing me with a small, local users group of support during my days of working with OmSim, a very necessary requirement to keeping ones sanity when working with any software product. Many thanks also to Eddy Yap for all the data he provided for my simulation,

and for all the expertise he was able to provide to help me tune the model.

Thanks to Carl Sheehan, and to NorPac Controls for all their time and equipment, a great asset in understanding, characterizing and modelling the pneumatic valve.

I thank the Natural Sciences and Engineering Research Council of Canada for the scholarship which funded this work. I also thank the BC Advanced Systems Institute, and Dr. Guy Dumont for their additional scholarship and funding.

Finally I would like to thank my family for all their encouragement through the years, without which I wouldn't be where I am today. Most importantly I want to thank my wife, Sheriann, for all her patience, understanding and her sincere effort to be interested in a topic that is somewhat removed from most peoples fascination.

JAN A. BERGSTROM

*The University of British Columbia*

*December 1998*

To my farfar, Arne Beurling.

# Chapter 1

## Introduction

In today's age of electronics and the so called *Paperless Office*, the use of paper has not disappeared. Instead of decreasing the amount of paper being used, advances in technology have simply increased our expectations of the ease of access to printed material. From high speed printing presses to laser printers, the demand for fast, continuous operation has become tantamount and the cost of a stoppage due to flaws in the paper unacceptable. This has driven the need for higher quality in the paper product. At the same time, increased global competition has put downward pressure on the prices of the product.

As a result, the paper industry has been forced into using more precision machinery, running at ever increasing speeds. To support this, the paper mill has become increasingly computerized, with more and more control systems in place to ensure that the final product is of the desired quality. Additionally, most mills are under 24 hour operation in order to maximize their capital expenditure, leaving little time to stop the process in order to fix problems with existing equipment, much less to test new control strategies.

This thesis focuses on aspects of these two points. Namely it attempts to provide an environment within which to simulate the process, to help identify problems and to test the effectiveness of new control strategies. It also attempts to address

one of the problems that lead to machine stoppage, faulty valve performance caused by friction.

## 1.1 Paper Making and the Paper Machine

A paper mill is a complex system with hundreds of subsystems, and thousands of control loops. It can roughly be broken down into four sections (see Figure 1.1). First there is the *stock preparation system* where the wood fibre is prepared and mixed in the proper proportions. The *approach system* is where this pulp and water slurry is diluted (with more water) and prepared for the paper machine. The *paper machine* itself starts with the *headbox*, which distributes the slurry onto a moving mesh (called the *wire*). The excess water is drained, the paper is pressed, and dried and wound onto a large reel. Finally there is a system to recirculate and clean various flows in the mill.

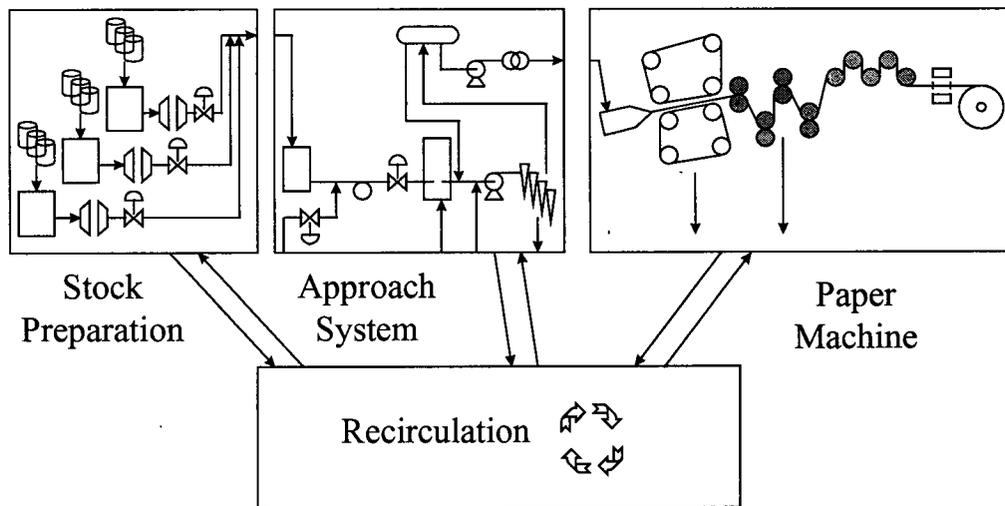


Figure 1.1: A rough schematic of the main areas of a paper mill.

The quality of the paper is usually gauged in terms of the steady-state value and variability of several measurements. These include the *basis weight* (the weight

of the paper in grams per square metre) and moisture content. The component makeup of the sheet is also important, but is often not measurable. The paper characteristics are complex functions of the consistency of the pulp slurry, the volume of pulp hitting the wire, the component makeup of the pulp, as well as the drainage and drying in the paper machine.

These characteristics are usually measured just before the paper is wound onto the reel. Some closed-loop control is done between this measurement and the start of the approach system. Because of a long time delay however, basis weight control is only effective at low frequencies (in the order of tens of minutes per cycle). Anything faster than this is essentially running in open loop once it has left the stock preparation area.

In order to produce the low variability demanded by modern printing presses, it is imperative that the quality of the flow entering the headbox be uniform and consistent. This stresses the importance of the control loops in the approach system.

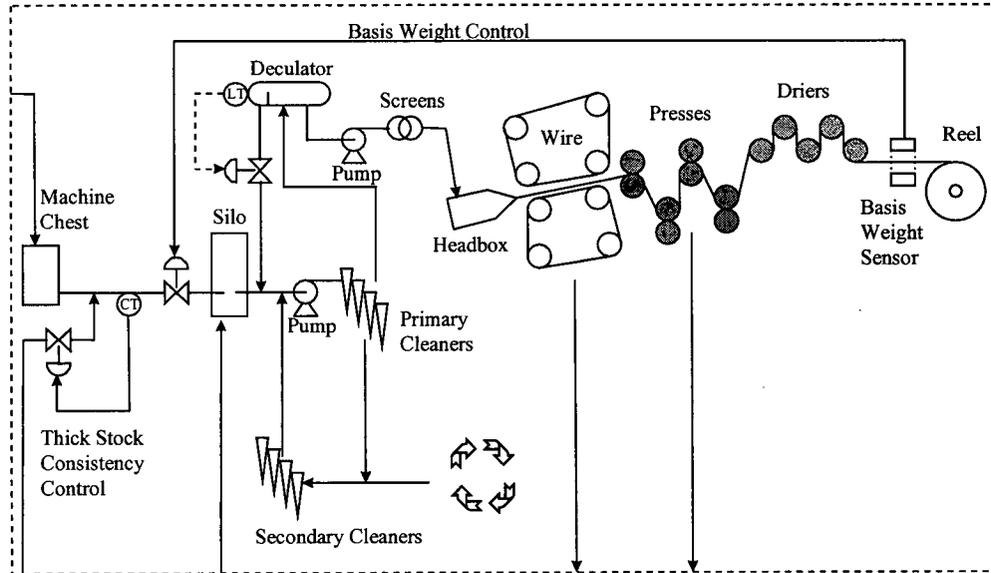


Figure 1.2: A schematic representation of the approach system (and paper machine).

The focus in this thesis is the approach system of the paper machine. The

layout of this part of the mill is shown in Figure 1.2. In general, the approach system begins with the machine chest/mixing chest. This is where the various component stock flows are mixed together. The flow from this tank is diluted slightly. This dilution flow is used to regulate the consistency of the flow leaving the machine chest. The resultant flow usually has a consistency around 3 or 4%, and is referred to as *thick stock*. The thick stock undergoes its major dilution through the silo, where the flow picks up white water (recirculated runoff from the paper machine), resulting in a consistency around 1%. This is now referred to as *thin stock*. Before being sent to the paper machine the thin stock is cleaned of heavy contaminants through a bank of hydrocyclones called the cleaners, cleared of large pulp flocks in the screens, and usually allowed to settle in a tank, in order to remove air and provide a steady pressure head (typically in a device called a deculator). Before reaching the headbox, it must pass through a pump to gain the additional head needed to force it onto the wire. The headbox may or may not include a damping device (called an attenuator) to lessen the effect of pressure pulsations. In order to maintain steady flows and even pressure at various operating points, portions of the flow are also recirculated. In this case a portion of the headbox flow returns to the deculator, and overflow from the deculator, returns to the suction of the primary fan pump. The order of the devices just mentioned may change slightly from machine to machine.

## 1.2 Designing Blocks for Simulation of a Paper Machine

As just mentioned, plant operators are faced with a set of conflicting objectives when it comes to managing the paper mill. The need to maintain operation around the clock makes it increasingly difficult to improve performance, by leaving little time to deal with problems, much less to test new ideas, without losing valuable production hours. A potential solution to improve this situation involves the use of realistic simulations of the system on which problems can be identified and new configura-

tions and control strategies can be tested and validated, before being implemented on the real system.

The first part of this thesis attempts to address this issue and is part of a joint project between the Pulp and Paper Centre at UBC and the Lund Institute of Technology, Lund, Sweden to develop a library of basic blocks to facilitate the simulation of complex paper machine systems. The work is being done in a modelling language called OMOLA for use with the simulation package OmSim, both of which were developed at the Lund Institute of Technology <sup>1</sup> [2]. Intrinsic to the joint work for this part of the thesis, various design decisions were made in consultation to satisfy this thesis as well as that of Tornhagen and Persson [34].

The focus in this part of the thesis was to develop an object framework and a set of basic components setting the foundation of a class library that could later be expanded to allow easy simulation of different parts of the paper mill. The structure and blocks were tested with a simple simulation of the approach system of a newsprint machine. This was used to show the effect of faulty valve performance on the eventual quality of the paper.

### 1.3 Designing a Controller for Valves with Friction

According to Bialkowski [11], two thirds of all control valves in a mill oscillate when in automatic mode and of these over half were oscillating due to characteristics of the pneumatic valve. One of the main problems with pneumatic valves is related to the presence of friction in the moving parts of the valve.

Friction, a force that opposes motion, is present in almost all mechanical systems. In many cases we take advantage of, and count on the effects of friction, like in the brake system of a car, or the simple tying of knots. In control systems however, friction is often a hindrance rather than a benefit. Its nonlinear nature

---

<sup>1</sup>OmSim is available at no cost, via anonymous FTP at <ftp://ftp.control.lth.se/pub/cace> with versions for Sun and HP workstations as well as PC's running Linux. Further information can be obtained at <http://www.control.lth.se/cace>.

tends to cause problems for conventional control strategies, yet the complexity of the problem typically means that it is ignored all together in control design. This often leads to degradation in performance with the potential for limit cycles in the system.

A better understanding of friction, and some of the techniques that can be used to address its effects can go a long way toward improving the situation. The second part of this thesis examines the nature of friction in a pneumatic valve, and proposes an adaptive control strategy that can be applied to a troublesome valve, without stopping the process. The objective is to eliminate oscillations in troublesome frequency ranges as identified in the first part of the thesis.

## 1.4 Industrial Collaboration

The work in the first part of this thesis was originally intended to be done in conjunction with the research centre of a paper company. They were sponsoring the project, providing the initiative that was to result in access to a mill and their data, in order to build, validate and test the model. The closure of this centre, and subsequent sale of the paper division led to a loss of this support.

The modelling work in the thesis was then forced to rely on the data collected, and the model built by Yap [48] for specifications, measurements and validation. The work of Yap [48] was completed before the closure of the research centre, and was done on the same paper machine that had been intended for this thesis.

The second part of the thesis was focused on a particular valve in the aforementioned mill. This valve was provided and supported by NorPac Controls of North Vancouver. From the beginning, NorPac was very supportive and enthusiastic about this work. After the closure of research centre, NorPac was able to provide access to similar valves in their warehouse. A combination of documentation, measurements of valve dimensions, and tests performed on stock valves, provided insight into the characteristics of the valve. Unfortunately, the nature of the testing equipment, the

lack of a real process environment, and the new (low friction) quality of the valve, precluded testing the control design on the real system.

## Chapter 2

# Simulation Blocks for the Modelling of a Paper Machine

The need to identify problems, and to test changes in a paper mill without affecting the operation of the process has been the driving force behind the use of dynamic simulation. This joint project with Lund intends to facilitate the use of dynamic simulation by establishing code library for the OMOLA/OmSim environment. The long term objective is to be able to draw on the library to quickly build a dynamic simulation of a paper mill, in order to test new control strategies, or system configurations. This requires that the library be built with a flexible architecture, with a common data interface between components, and in such a way that it can easily be expanded to include new equipment models.

This part of the thesis will describe the library that has been built so far, and will use this library for a simulation of the approach system of a real paper machine. Some tests will be performed using the simulation to identify potential areas of concern. Having done work in the second part of the thesis to show the oscillatory effects of friction in valves, the simulation here will also be used to identify the frequency ranges that are most likely to show up on the final paper product, and thus identify the most critical problems.

Section 2.1 describes some of the basics of OMOLA and OmSim, giving an idea how the environment works, and how the models are represented. Section 2.2 outlines the design objectives for the class library that has been designed, as well as some of its key features. Section 2.3 shows the use of the library components and presents the simulation results.

## 2.1 OMOLA and OmSim

OMOLA is an object-oriented dynamic modelling language, that can be used to construct continuous, discrete or hybrid models. OmSim is a set of graphical tools to help manipulate and simulate the models built in OMOLA. It is composed of various parts; a library browser to find models, a graphical user interface (GUI) to view, manipulate and edit OMOLA models, a class browser, a model interpreter and compiler, and a simulator. The simulator itself has many components including; initial condition solvers, various integration routines, discrete event handlers, a GUI interface to adjust parameters, change model values, and view simulation results, a file based interface to import and export simulation data to and from flat-files, and finally a scripting language (OCL) to automate simulation tasks, including setup. For more detailed information about OMOLA and OmSim see [2, 30].

The OMOLA modelling language consists of three basic constructs; variables, equations and classes.

*Variables*, like in most languages, are used to store the data in OmSim. They can be continuous or discrete, and of different types (Real, Integer, String, Symbol, Matrix).

*Equations* represent the logical relationships between variables. Some equations are also implicit in the way a model is put together (for example with connections between blocks). Equations are either equalities, or assignments. Assignments imply an input/output relation between variables. Equations do not carry this implication.

*Classes*, as in most object-oriented environments, are a type of template for code. When a class is used, it is called an instance of the class. In OMOLA, classes and instances of classes are used to structure, organize and assemble the various variables and equations. Everything in OMOLA is done within the context of some class object. New classes can be derived from existing classes. They can contain variables, equations and instances of existing classes, or can even define and use new classes. OMOLA defines a number of important base classes used for all designs.

- Model is the base class for all concrete components in OmSim.
- Terminal is an object used in joining model components together. A terminal defines the interface to the *outside world*. There are three main types; *SimpleTerminal*, *ZeroSumTerminal* and *RecordTerminal*.
- Connection is used to specify a linking of two component models. It joins the terminals of the components. A connection of terminals implies new equations for the system. The form of the equation is determined by the types of terminals being connected.
- Event is a means used by OmSim to fire and propagate discrete state changes in the model.
- Parameter as the name implies, is a parameter of the model which can be set when the model is compiled.
- Layout is used to hold graphical information about the model (bitmap and location) and is ignored by the compiler.

One of the most important features of the OMOLA, OmSim environment is the ability to create models that are non-causal (in the input-output sense). Systems are modelled with true differential equations or differential algebraic equations (DAEs), that do not require the *a priori* selection of input or output variables.

Once a model is defined, the system can be simulated with any combination of input/output configurations, as long as the proper number of boundary conditions are satisfied for the specifications of the model. This allows for the creation of very realistic components, that can be used to answer many different types of questions without requiring reformulation.

As an example, a simple pipe can be modelled with the following conservation equation and dynamic energy equation

$$\begin{aligned} 0 &= Q_1 + Q_2 \\ P_1 + \rho g h_1 &= P_2 + \rho g h_2 + k_{loss} \frac{1}{2} \rho \frac{Q|Q|}{A^2} + \frac{l\rho}{A} \dot{Q} \end{aligned} \quad (2.1)$$

These equations can easily be implemented as an OmSim model with the simplified code shown in Figure 2.1. Note that there is no functional (input/output) relationship defined between variables. Given the parameters and any four of  $(P_1, h_1, Q, P_2, h_2)$ , the remaining variable can be solved for. Also note that this sample does not include any graphical code, and does not make use of any class hierarchy other than to include instances of the flow terminals (defined later in Figure 2.2).

---

```
FlowModel ISA Model WITH
  T1 ISA FlowCutTerm;
  T2 ISA FlowCutTerm;

  rho, k, A, l ISA Parameter;

  T1.Q + T2.Q = 0;
  T1.P + rho*::g*T1.h = T2.P + rho*::g*T2.h
    + k * rho * T1.Q * ABS(T1.Q) / 2 / A^2
    + l * rho * Q' / A;
END;
```

---

Figure 2.1: Code for a basic OMOLA simulation block (excluding graphical information).

OmSim has several numerical integration routines, including *Euler*, *Runge-*

*Kutta*, *DASRT*, and *Radau5*, the last two of which can handle DAE equation systems. In order for OmSim to use these standard techniques, the model must be interpreted and compiled. Unlike systems such as *Simulink*, which simulate the model in a *sequential modular* way, OmSim uses a *simultaneous* approach. It manipulates the entire set of equations and solves the system as a large matrix problem. This approach is similar to that used by Aspentech's *SPEEDUP<sup>TM</sup>* [30].

## 2.2 The Pulp Library

The main objective of the project was to generate a set of simulation blocks that could easily be used for many problems. The blocks had to be flexible, extensible, and easily reusable. The following features were considered important:

- the interface to the blocks had to contain enough information to make them generally useful
- components should be non-causal in their equations
- wherever justified, components should be non-causal with respect to flow direction
- the library should take advantage of object-oriented features of inheritance, encapsulation, and reuse

### 2.2.1 Terminal Definition

The variables in the terminals determine what can and can't be tracked by the model and also dictate which components are compatible. Any flexibility built into this structure facilitates future expansion. The slurry was treated as a homogeneous fluid, with a single set of fluid properties for a given condition. Volumetric flow rate, pressure, and height of the fluid are needed in order to calculate flow rates. This

basic fluid information was grouped into one RecordTerminal for easy manipulation (see Figure 2.2).

---

```
FlowCutTerm ISA RecordTerminal WITH
  Q ISA FlowTerminal;
  P ISA PressureTerminal;
  h ISA HeightTerminal;
END;
```

---

Figure 2.2: Structure of fluid dynamics terminal.

Fibre mass flow was used to track consistency. To allow for further expansion, this data was set up as arrays, rather than single elements, allowing for consistent manipulation of any amount of data using vector equations. Since there are three main classes of material in the system that are likely to behave differently from each other, the mass flow was set up as three component arrays; one for Fibres, one for Fines and one for Additives (see Figure 2.3).

Finally, temperature was considered an important variable in order to help calculate other fluid properties. The complete slurry information terminal is a combination of the structures just mentioned (see Figure 2.4).

### 2.2.2 Acausality

In order to ensure noncausal relationships the equations in the blocks should be created as equalities rather than assignments and interfaces to the blocks should be defined without predetermined causality. Pipes, and tanks should not have predefined inputs or outputs, the directions of fluid flow should not be predetermined, and the dependent and independent variables associated with components and models should not have to be selected *a priori*.

A common way to track (fibre) mass flow along a pipe is with a *plug-flow* model using delay. The *Delay* operation in OMOLA is not a bi-directional function, and thus imposes causality on the model. This problem can be overcome in most

---

```

MassFluxTerminal ISA RecordTerminal WITH
% Mass Flux (Flow Rate) for consistency tracking
Mf ISA ZeroSumTerminal WITH
    unit := "kg/s";
    quantity := "mass.flow.rate";
    n TYPE Integer;
    value TYPE column[n];
    default TYPE STATIC column[n];
END;
% Dummy Data to allow for consistent equations
% - also contains Consist Data at endpoints
Co ISA SimpleTerminal WITH
    unit := "1";
    quantity := "number";
    n TYPE Integer;
    value TYPE column[n];
    default TYPE STATIC column[n];
END;
nComp TYPE Integer;
value TYPE column[nComp];
value = Mf;
Mf.n := nComp;
Co.n := nComp;
END;

```

---

Figure 2.3: Structure of mass flux terminal.

blocks using conditional logic (see example in Section 2.2.4). Endpoints in the flow require extra logic, because only the input end defines the flow. Since the OmSim compiler checks the model by comparing the number of equations and unknowns, it is not possible to have an equation at each endpoint. Because OMOLA requires that conditional equations be defined for all conditions, it is not possible to have a pair of half (mutually exclusive) equations like (2.2) and (2.3), (where  $Q$  is the volumetric fluid flow and (2.2) resides in a block at the left end of the model and (2.3) in a block at the right).

---

```

SlurryCutTerm ISA RecordTerminal WITH
  F ISA FlowCutTerm; % Fluid Flow
  Fib ISA MassFluxTerminal; % Fibre Mass
  Fin ISA MassFluxTerminal; % Fines Mass
  Add ISA MassFluxTerminal; % Additive Mass
  T ISA TempTerminal; % Flow Temperature
  nFib TYPE Integer; % num Fibre-Flows
  nFin TYPE Integer; % num Fines-Flows
  nAdd TYPE Integer; % num Additives
  Fib.nComp := nFib;
  Fin.nComp := nFin;
  Add.nComp := nAdd;
END;

```

---

Figure 2.4: Full terminal structure.

$$IF Q \geq 0 THEN FibreFlow = MassFlowFromLeft \quad (2.2)$$

$$IF Q < 0 THEN FibreFlow = MassFlowFromRight \quad (2.3)$$

Instead, it is necessary to introduce a dummy variable into the flow path that can be used to complete the ELSE portion of (2.2) or (2.3) when the endpoint is the output. This resulted in the terminal definition of Figure 2.3 and code in the endpoint blocks as seen in Figure 2.5. Finally, to ensure that the equation system is consistent when the fluid flow is zero an extra parameter was added to each endpoint, to specify if that end “calculates the mass flow when fluid flow is zero”. This is not shown.

### 2.2.3 Modularity

In order to ensure flexibility and extensibility, the blocks were designed in a modular way, along similar lines to those outlined in [31]. The main blocks in the library represent physical objects in the plant. Similar objects are tied together through inheritance from abstract classes that contain the common structural components (like terminals). The physical objects themselves have also been broken down internally

---

```

zeros(nComp,1) = IF Connect == 'TankAt2 AND Q > 0
                OR Connect == 'TankAt1 AND Q < 0
THEN T1.Co - T2.Co
ELSE IF Connect == 'TankAt1
  THEN T1.Mf - Q * T1.Co * TP.rho
  ELSE T1.Mf - Q * T2.Co * TP.rho;
T1.Mf + T2.Mf = zeros(nComp,1); % net mass flow is 0

```

---

Figure 2.5: Equations for the endpoints of the flow sections. It is assumed that at the tank end, the dummy terminal provides the consistency in the tank.

into logical components (or modules) which typically represent different physical behaviour of the component. The flexibility of the architecture lies in these internal modules. They have been designed as self-contained units with a common interface (for all modules that perform a similar function). An example of the internal modules is shown in Figure 2.6 and is explained in the next section.

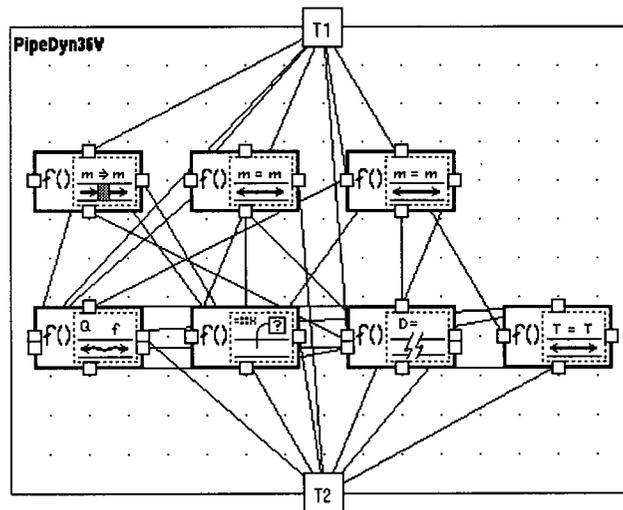


Figure 2.6: Example of the internal subcomponents for a two port flow device. This particular example is a pipe with dynamic fluid calculations, and plug flow for one of the mass transport blocks.

This structure takes advantage of OMOLA's ability to override class be-

haviour anywhere in the program structure. As a result, it is possible to build the complete model only once, using specific components, with specific levels of complexity, and later simply swap out an internal module in a component (for a more or less complex one). This feature can be used to quickly test the effect of a more complex subcomponent equation on the overall system, to upgrade the model with more detailed equations, or to analyze different points of interest (without slowing down the simulation with equations that are not needed for the current analysis).

#### 2.2.4 Component Structure

Several basic component blocks have been built. They include pipes, tanks, junctions, pumps, and valves. Pipes, pumps and valves all share the same two node flow structure, and were thus developed from the same common base class, *TwoFlow*. As mentioned earlier, the internal structures of the components were broken into logical subcomponents, representing various properties of the object. The *TwoFlow* component was broken down into 7 blocks as follows:

- one to track the fluid dynamics
- one to calculate average fluid properties of the flow
- three for the constituent arrays to calculate their mass flows (one each for Fibres, Fines and Additives)
- one to track heat flow
- one to calculate transportation delay (to potentially be used with the above)

The OmSim graphical representation of the subcomponents in a pipe object are shown in Figure 2.6. This basic breakdown is common to all pipes, pumps and valves, with the only mandatory difference being in the fluid dynamics module (lower left).

As an example of the varying sets of subcomponent models that were designed, the class hierarchy for the fluid dynamics blocks for different pipe sections is shown in Figure 2.7. It contains models for dynamic and steady-state flow energy equations, and incorporates varying degrees of complexity in calculating pipe friction. The rightmost blocks would contain, through inheritance, the same type of information as the simplified example of Figure 2.1. The leftmost blocks would only contain the common information such as terminal definitions, and conservation of mass equations.

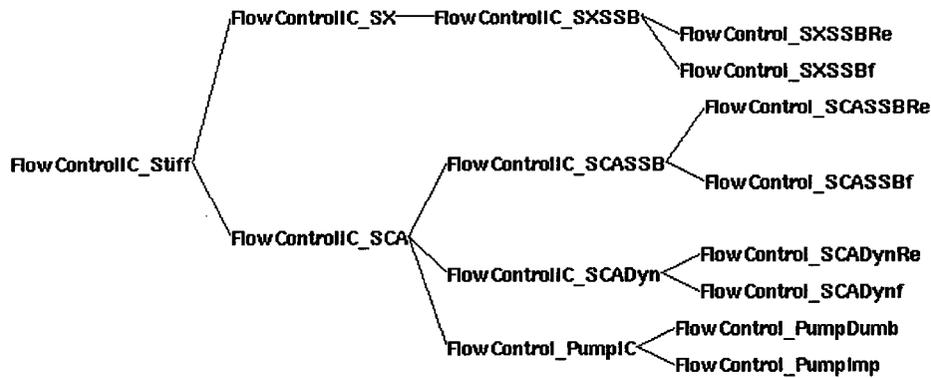


Figure 2.7: A portion of the class hierarchy for the fluid dynamic modules for pipe flows. The lower tree is for constant diameter pipes and devices while the upper tree is for pipes where the diameter changes.

As a further example of the varying degrees of complexity, mass flow could either be considered instantaneous (2.4), plug flow (2.5), or a lagged mixing process.

$$0 = \dot{m}_{Fib1} + \dot{m}_{Fib2} \quad (2.4)$$

$$0 = \begin{cases} \frac{\dot{m}_{Fib2}}{|Q_2|+\epsilon} - \frac{\dot{m}_{Fib1}(t-\tau)}{|Q_1(t-\tau)|+\epsilon}, & Q_1 > 0 \\ \frac{\dot{m}_{Fib1}}{|Q_1|+\epsilon} - \frac{\dot{m}_{Fib2}(t-\tau)}{|Q_2(t-\tau)|+\epsilon}, & Q_1 \leq 0 \end{cases} \quad (2.5)$$

Figure 2.8 shows how the first two alternatives would be implemented in OMOLA. In both the equations and the code, the flow through the object is considered positive from 1 to 2.  $T1$  is the terminal at point 1,  $T1.F.Q$  is the value of the flow,  $T1.Fib$  is the mass flux, and  $td$  is a delay determined in another block.

---

```

T1.Fib + T2.Fib = 0

0 = IF T1.F.Q > 0
  THEN T2.Fib/(ABS(T2.F.Q) + eps)
    - DELAY( T1.Fib/(ABS(T1.F.Q) + eps), td )
  ELSE T1.Fib/(ABS(T1.F.Q) + eps)
    - DELAY( T2.Fib/(ABS(T2.F.Q) + eps), td )

```

---

Figure 2.8: OMOLA code for implementing plug flow equations.

Tanks were built in a modular fashion as well. The internal behaviour of the tank (with only one input/output) was grouped into an inner tank model. This was further modularized, in a similar fashion to the flow components above, with different blocks for different physical properties (fluid accumulation based on tank shape, constituent mass and heat accumulation, fluid properties, and tank pressure). The inner tank model represents the common features between all tanks, regardless of the number of ports (or connections). When building actual usable tank models, it would incorporate an appropriate number of port components (with endpoint equations as described in Section 2.2.2). It would include one inner tank model that would have all ports connected to it (using the features of Simple and ZeroSum terminals to generate the appropriate equations automatically)<sup>1</sup>. This optimizes the amount of code reuse while maintaining flexibility and maintainability in the final components that are built. Parts of a tank model can be seen in the deculator model of the next section.

### 2.2.5 Complex Objects

Wherever possible, more complex objects were built as compound structures from simpler components. An example of this is the deculator (see Figure 2.9). It consists

---

<sup>1</sup>In the current implementation of OmSim it was necessary to manually code the equations for the multiple connections to the inner tank, rather than use the standard visual connections. This was due to a bug in version 3.5 that did not properly interpret connections between terminals that contained arrays.

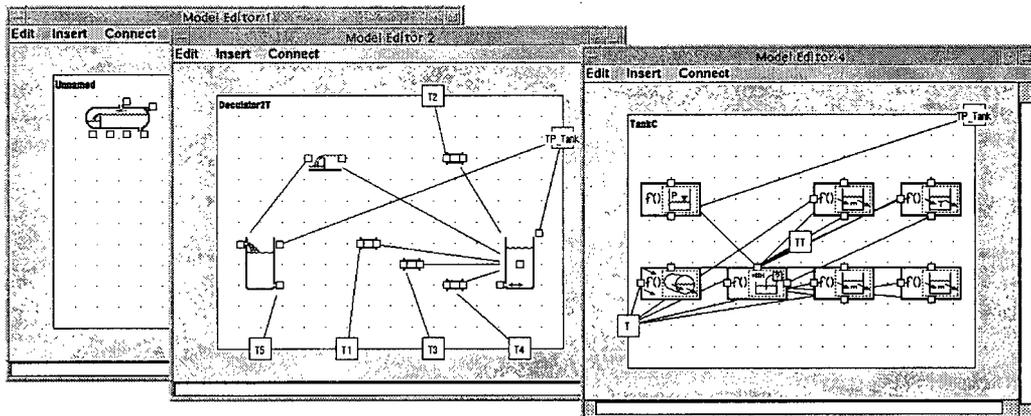


Figure 2.9: OmSim representation of the modular construction of a complex object (deculator). In Model Editor 2 the left tank device is a complete tank model while the right one is the primary inner tank model associated with the deculator. Model Editor 4 shows the module breakdown of the inner tank block.

of two tanks; a primary and an overflow, with a separating wall in between. The tank is a cylinder on its side, with an ambient pressure below atmospheric, controlled by an external compressor.

The deculator was built from a standard four port tank. The inner tank model was modified to add a terminal to transmit the ambient pressure from the external compressor. The standard volume calculation block was also replaced with that for a cylinder on its side. An extra port fitting was added as the interface to the overflow tank. This port used a block with a rectangular weir equation for the fluid dynamics sub-component, rather than the standard pipe equation block. A second, completely functional, two port tank model was added as an overflow tank. The flow from the weir was directed to the second tank. Since the overflow tank is complete, there was no need for a fitting device between its port and the external terminal of the deculator.

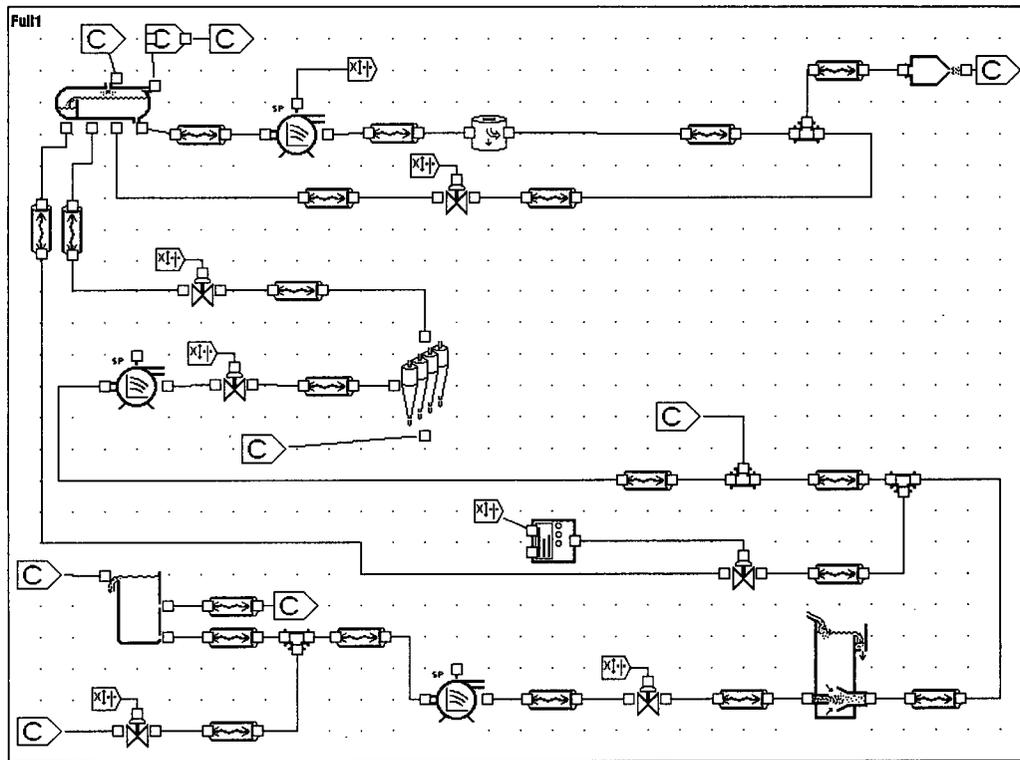


Figure 2.10: OmSim graphical representation of the approach system model. The system starts at the bottom left with the machine chest and flows right and up through the machine pump, basis weight valve, silo, deculator recirc, secondary cleaners recirc, primary fan pump, primary cleaners, deculator, secondary fan pump, screens, and headbox.

## 2.3 Approach System Simulation

A simulation was made of the approach system of a real paper machine. The layout of this system is shown in Figure 2.10. For the current simulation, dimensions of components and layout of the plant were obtained from plant drawings and specifications. Information about steady-state conditions was obtained from DCS measurements where possible, or from sampling performed at the mill where necessary. Most of the data gathering work was done by Yap [48].

The model of the approach system was built using the blocks described in this paper. Only one constituent flow was tracked (fibre consistency). Most pipes

used dynamic energy equations and a plug flow model (with the delay calculated instantaneously from the flow rate and pipe volume). Tanks used equations reflecting the actual tank shape to calculate fluid level and an ideal mixing model for fibre accumulation. Pump models incorporated equations representing the pump curves. Valves used steady-state flow equations based on specified valve  $Cv$ , and assumed a linear characteristic with no transport delay. In order to minimize the number of parameters to be entered every time the model was started, a library of components was built for this particular paper machine (derived from the components in the Pulp Library) which set defaults for all parameters in accordance with the data obtained from the mill. Other components such as controllers, were obtained from libraries developed in Lund. The final model can be seen in Figure 2.10.

The model was validated against steady-state data from the mill, as well as against the simulation developed in [48]. Some simple step responses of the system were generated to demonstrate the dynamic response of the model (Figure 2.11). In these tests, the secondary fan pump speed was increased to increase the flow through the headbox. Subsequently the basis weight valve was opened further to increase the amount of thick stock entering the system.

### 2.3.1 General Consistency Control

The model has been used to compare the effects of consistency disturbances entering the machine chest versus ones entering with secondary cleaners accepts. Disturbances of  $\pm 10\%$  of the steady-state consistency were added to the flow at different frequencies. The resulting headbox consistency was recorded, and the results plotted in Figure 2.12 (normalized by the steady-state gain between disturbance and output). The results show that variations in the recycle consistency have a comparable effect on headbox consistency to variations in thick stock. More importantly, these variations are present through higher frequencies as well. Thick stock consistency is controlled with a consistency loop that was not included in this simulation, however,

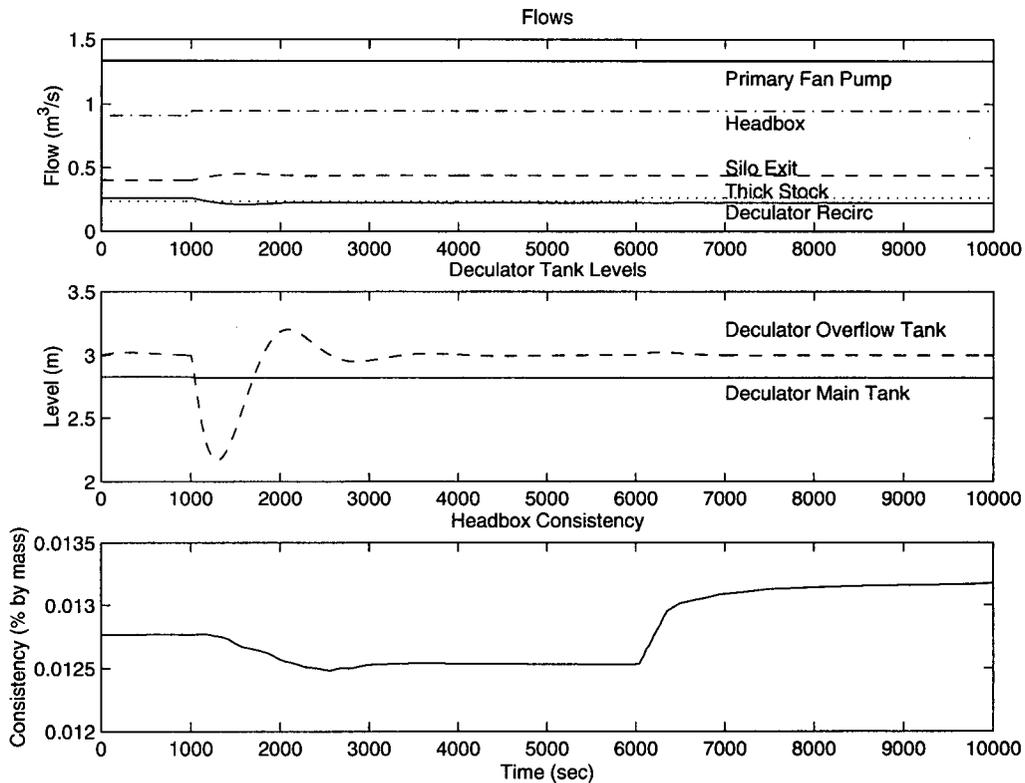


Figure 2.11: Step response of the approach system. The secondary fan pump speed was increased at 1000s, and the thick stock flow was increased at 6000s.

the existence of such a loop, and the similarity of the sensitivity to variations would indicate that consistency control is also warranted on the recycle flow.

A simple consistency control loop, as used on the thick stock flow, was tested on the recycle flow. The simulation quickly showed that this was not a feasible solution. Two flaws became immediately apparent. The changes in dilution flow, that were required to regulate the consistency oscillations, added flow and pressure variations to the system that could be seen throughout the process, right to the headbox. This effect was minimized somewhat by the behaviour of the silo. It acted as a natural damper for these flow fluctuations, maintaining a more stable flow at its output. This damping resulted from supplemental silo whitewater, being sucked (by the fan pump) into the flow, to make up for any decreases in the supply flow.

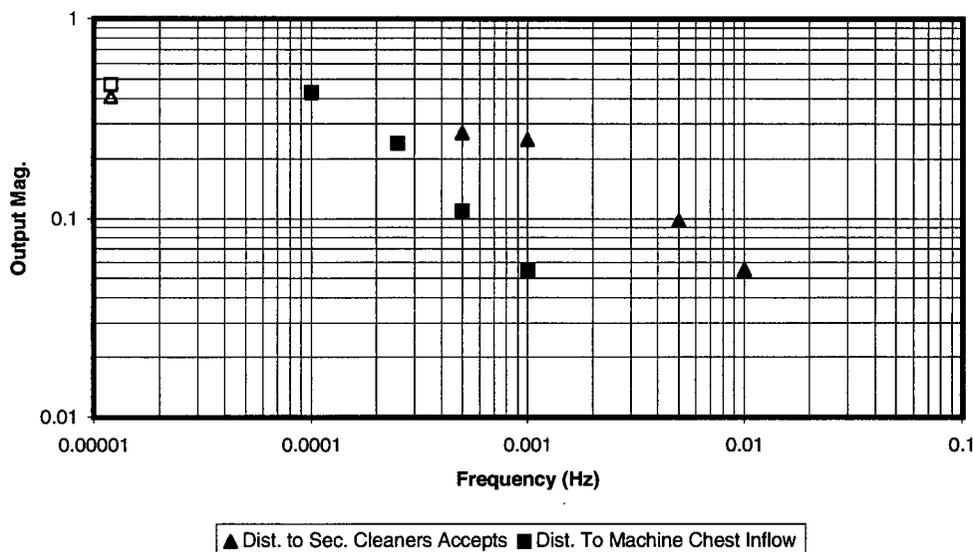


Figure 2.12: The dynamic response for consistency disturbances injected at the machine chest and at the secondary cleaners accepts. The two hollow points represent steady-state values.

This had the unfortunate effect of negating the consistency control action since, as dilution flow decreased, silo flow increased to make up the difference. Since both the dilution flow and silo water are usually whitewater of similar consistency, little net gain was achieved by the controller.

### 2.3.2 Effects of Valve Friction on Consistency Control

The simulation was also used to test the effects of valve-induced oscillations on the final paper quality.

In this test, the characteristics of the deculator model were modified slightly. Instead of using the ideal mixing model of the previous section, the main tank in the deculator was modelled by a combination of plug flow and ideal mixing model. This is commonly accepted as a more realistic approach. An arbitrary division would model half the tank as plug flow and the other half as ideal mixing. Since the

deculator is a complex tank with multiple connections (inflows and/or outflows), such a simple division was not possible. Instead, the tank's total volume ( $V_t$ ) was divided into effective sub-volumes ( $V_{eff-i}$ ) for each of the  $n$  connections  $i$ , based on the amount of flow associated with the connection ( $Q_i$ ):

$$V_{eff-i} = V_t \frac{|Q_i|}{\sum_{i=1}^n |Q_i|} \quad (2.6)$$

Inflow volumes were assumed to be plug flow with the delay calculated based on the flow at that connection

$$delay_i = \frac{V_{eff-i}}{|Q_i|} \quad (2.7)$$

while the outflow volumes were all combined and treated as one perfect mixing tank with effective volume calculated as follows (assuming negative flows are outflows)

$$V_{eff-ideal} = V_t \frac{\sum_{i=1}^n |Q_i| (SIGN(-Q_i) + 1)/2}{\sum_{i=1}^n |Q_i|} \quad (2.8)$$

This formulation ignores complex effects like an inflow and outflow located next to each other, and not taking part in overall dynamics. For a simple tank with one input and one output of equal flow rate, the model simplifies to the basic half/half split described at the onset. It should also be noted that (2.7) simplifies to:

$$delay_i = \frac{V_t}{\sum_{i=1}^n |Q_i|} \quad (2.9)$$

for all cases.

To simulate the potential disturbances caused by valve friction, sinusoidal signals of various frequency were added to the steady-state position of the consistency control loop dilution flow valve. The steady-state position of this valve in the simulation (given the sizing of the valve, and an average desired dilution flow of 10% of the thick stock flow) was 20% open. From the second part of the thesis, the largest oscillations observed in the tests of the valve under PI control were  $\pm 10\%$

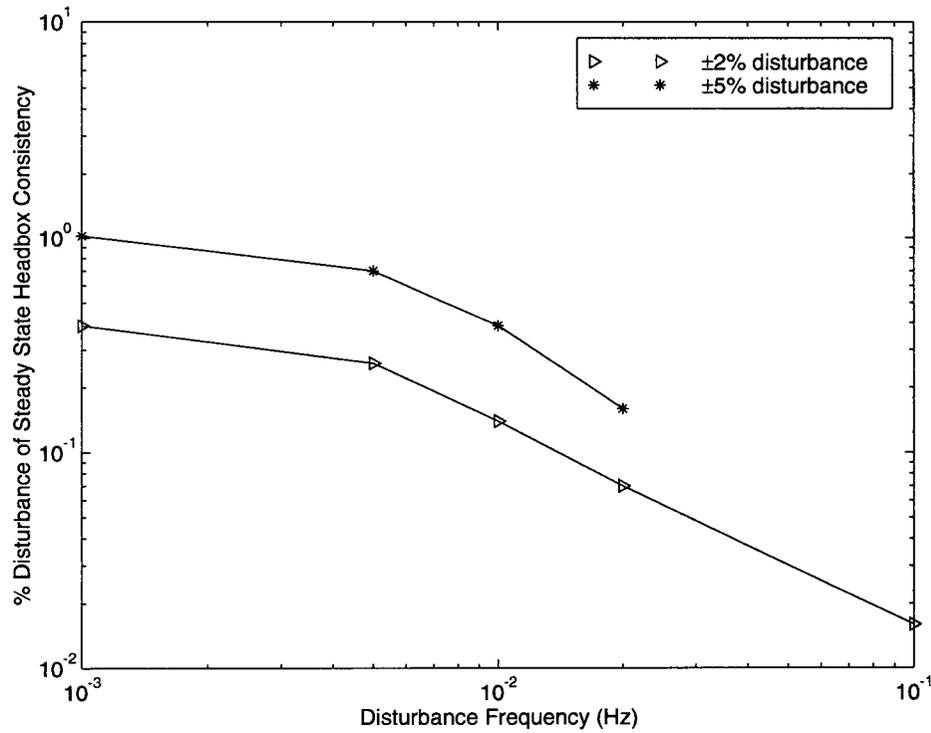


Figure 2.13: The response in consistency at the headbox to oscillations in valve position. The output is in percent of steady-state headbox consistency.

of the desired flow. Oscillations of 10% of the steady-state were therefore added to the valve position (ie  $\pm 2\%$  opening). These results are shown in Figure 2.13. Since the jumps in valve position determine the size of the oscillations regardless of the steady-state value, it is more reasonable to quantify the oscillation magnitude in terms of the full travel of the valve. Using this approach, oscillations of  $\pm 5\%$  total valve travel should be added to the steady-state signal. The results are also shown in Figure 2.13.

The most troublesome and prevalent frequencies (identified in the literature [10, 35, 23, 38] and in the simulations of Chapter 3) are in the range from 40s to 200s. From Figure 2.13 it can be seen that oscillations of  $\pm 2\%$  of valve travel appear as disturbances to the final consistency of  $\pm 0.26\%$  of its steady-state value. Even

when the disturbances are increased to  $\pm 5\%$  of valve travel, the final consistency is only disturbed by  $\pm 0.7\%$  of its steady-state value. Although significant to the final quality of the paper, this is small when considering the size of the offending oscillations. This is due to both the filtering effect of the deculator, as well as the small contribution of dilution flow to the final product. Piipponen [35] shows how the relative gain for a typical consistency loop is about  $-0.1$ , reducing the effect of friction oscillations on this loop. However, without this effect to mask the friction oscillations, disturbances 10 times larger would appear in the final product. This would be the case in any additive control loop (such as bleaching agents or dies), where the relative gain would be close to 1.

## 2.4 Summary

The foundation for a library of components for the simulation of paper machine systems was outlined in this chapter. The objective of building a modular and flexible structure was validated. The benefits of generic, flexible components was shown in the simple construction of a complex object like the deculator. The advantage of the modularized components was apparent in the ease with which new dynamics could be inserted (as with the substitution of a combined plug-flow and perfect-mixing model).

The simulation was successfully used to show the adverse effects of various disturbances on the final paper product. It was also used to test a potential control strategy. The simulation identified a flaw in this strategy without having to implement it on the real system. The next chapter will outline a control algorithm that could reduce the valve oscillations tested here.

## Chapter 3

# Control of Pneumatic Valves with Friction

As mentioned in the introduction, Bialkowski [11] states that two thirds of all control valves oscillate when in automatic mode and of these over half were oscillating due to characteristics of the pneumatic valve. Oscillations created in one loop can cascade their way through the whole process and eventually show up in the quality of the final product. This situation is particularly bad in situations where the valves control critical flows in the mill. The consistency control loop on the thick stock flow, described in the first part of this thesis, is an example of a loop, typically controlled by a pneumatic valve, and whose performance cannot be allowed to degrade.

Besides poor tuning, one of the major factors contributing to poor valve performance is the presence of friction in the valve. A survey of one particular pulp and paper mill identified 35% of valves having friction problems [20]. Excess friction can be due to improper adjustment in the packing of the valve, misalignments due to wear in the valve, or simply due to the nature of the application and design of the valve. Problems of the first two types are likely to appear in a valve like the consistency controller. Problems of the last type often occur in the pulp bleaching process where the strict environmental regulations require tight (and as a result,

high friction) seals in the valves controlling the flow of bleaching agent.

Much work has been done to investigate and identify these problems. Bialkowski [11] discusses the adverse effects of poorly tuned loops. Forsman [21] and Hägglund [23] discuss oscillation identification techniques, making specific mention of friction. Eborn [18] and Piipponen [35] use simulation to investigate the effects of friction in industrial control loops. Fitzgerald [19] gives a good overview of valve design.

The easiest solutions to the problem, involve servicing the valve or upgrading the valve to eliminate the friction problem. This type of solution involves stopping the process and taking the valve out of service. This part of the thesis proposes an adaptive controller that could be applied to the valve without stopping the process. It would be intended to help compensate for the effects of friction until such time that a scheduled shutdown occurs and the valve could be serviced or replaced.

Section 3.1 reviews basic principles and some mathematical models of friction. Section 3.2 introduces the pneumatic valve, compares it to other valves, and describes its operation. This is followed by an analysis of the control problem in Section 3.3, and a review of current work in the area of friction compensation in Section 3.4. A model to test the proposed controller is presented in Section 3.5, the controller is described in Section 3.6, and results are presented in Section 3.7.

### 3.1 Friction Models

Before proceeding with a discussion of the valve design and friction compensation techniques, some background will be provided on the nature of friction itself. Some of the formulations currently in use in the control community to model and study friction will be presented.

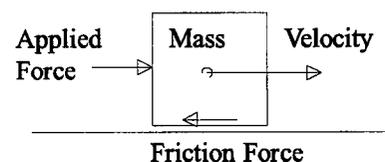


Figure 3.1: A simple mass balance showing the action of friction.

Friction is a complex behaviour that has been recognized and studied for

a long time, but never fully quantified. In recent years, increasing focus has been placed on developing friction models for the purposes of control system design [3, 4, 32, 33]. This introduction is mostly based on these references, except where cited otherwise. It is by no means a complete coverage of friction models, but it attempts to provide the background for the compensation techniques discussed later. For a more in depth coverage of these models and references to further reading, see the references cited above.

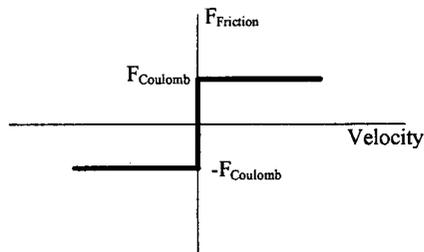


Figure 3.2: Simple Coulomb friction vs. velocity

of the object. It grows with the applied force, until a maximum value, after which it remains constant, acting in a direction opposite to the motion. This form of friction is referred to as *Coulomb friction*. When the object is stationary, the force is only as large as the applied force. Once the applied force exceeds the Coulomb friction, motion will begin, and the friction will be equal to the Coulomb friction in a direction opposing the motion (irrespective of the applied force). See Figure 3.2. The relationship can also be expressed as a discontinuous equation:

$$F_f = \begin{cases} -F & \text{if } v = 0 \text{ and } |F| < F_C \\ -\text{sgn}(v) F_C & \text{otherwise} \end{cases} \quad (3.1)$$

where  $F$  is the applied force,  $F_f$  is the total friction force,  $F_C$  is the Coulomb friction level and  $v$  is the object velocity. This type of a model shows simple hysteresis or dead-zone. When the object comes to rest and the applied force is reversed, it must first drop below the Coulomb friction level to zero, and then increase beyond the

Friction is due to the contact dynamics between an object (in motion, or being forced into motion) and the surface on which it moves. In general, friction acts to keep the object at rest. A simple schematic is shown in Figure 3.1.

In its simplest approximation, friction is a force that opposes the motion of

Coulomb friction level in the opposite direction before an effect is seen on the object.

Coulomb friction is often represented by a function of only velocity:

$$F_f = -\text{sgn}(v) F_C \quad (3.2)$$

This function may be easier to handle and simulate but ignores very important characteristics at zero velocity. If the system spends any finite time at zero velocity, this model is inappropriate.

The Coulomb model as a whole ignores many aspects of friction. One of the most important characteristics that is missing is the fact that friction is typically higher before motion begins. Once the initial friction is overcome, and the object begins to move, the friction drops down to a lower, constant level. The lower level is equivalent to the Coulomb friction of the previous

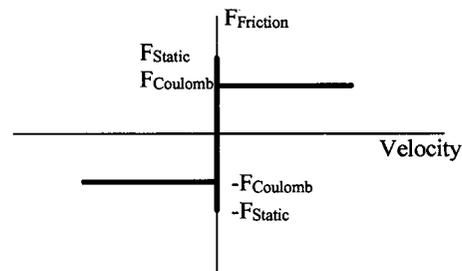


Figure 3.3: Static and Coulomb friction

model. The higher initial friction is termed *static friction*. It has been explained as the formation of small bonds between an object and the surface when there is no relative motion (see [4]). These bonds oppose any applied force until the force is so strong that they break. At this point the static friction has been overcome, and the opposing force reverts to the lower, Coulomb friction. In its complete form, this model would have the following form

$$F_f = \begin{cases} -F & \text{if } v = 0 \text{ and } |F| < F_S \\ -\text{sgn}(F) F_S & \text{if } v = 0 \text{ and } |F| \geq F_S \\ -\text{sgn}(v) F_C & \text{otherwise} \end{cases} \quad (3.3)$$

where  $F_S$  is the static friction level (greater than Coulomb friction). Figure 3.3 shows the relationship between friction and velocity in this model. This model can also be simplified as in (3.2) to make it only a function of velocity, and rendering it

undefined for zero velocity.

In order to overcome the numerical difficulties associated with (3.2) and (3.3), a formulation called the *Karnop* model is used. It defines a small region of velocity  $v_0$  beneath which the object is assumed to stick, and static friction applies:

$$F_f = \begin{cases} -F & \text{if } |v| < v_0 \text{ and } |F| < F_S \\ -\text{sgn}(F) F_S & \text{if } |v| < v_0 \text{ and } |F| \geq F_S \\ -\text{sgn}(v) F_C & \text{otherwise} \end{cases} \quad (3.4)$$

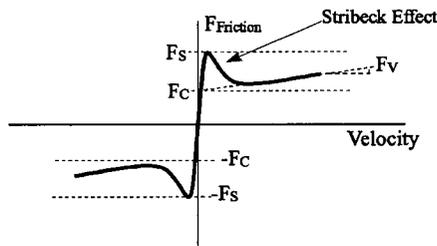


Figure 3.4: Complex friction model with static, Coulomb and viscous friction

Although these models provide insight into the behaviour of friction, they do not represent the more continuous characteristics of the real world. They also lack other important features. Figure 3.4 shows a representation of a more complete friction model. This includes aspects of Coulomb and static friction, but also many others.

The model is continuous through zero velocity. It features rapidly increasing friction at low velocities. This allows for small amounts of presliding, spring-like motion, a characteristic that has been observed in experimentation. It incorporates increasing dynamic viscous friction with velocity (characterized by slope  $Fv$ ). Finally, it includes falling friction levels, also called the *Stribeck effect* (between the region of static friction and Coulomb friction) a very important aspect in explaining observed behaviour. One example of a model that incorporates everything but the presliding motion is the exponential function:

$$F_f = (\alpha_0 + \alpha_1 e^{-|v/v_0|^\beta} + \alpha_2 |v|) \text{sgn}(v) \quad (3.5)$$

Here,  $\alpha_0$  represents the Coulomb friction level,  $\alpha_0 + \alpha_1$  the static friction level, and  $\alpha_3$  the viscous friction rate.  $v_0$  represents a velocity where the model switches from static to Coulomb friction and  $\beta$  is usually a number between 1/2 and 2. The

formulation is modified in minor ways by different authors, including; fixing  $\beta$  at 1, moving the viscous friction outside of the  $sgn(v)$ , or removing it altogether.

A model proposed in [28] includes all the features described above and continuity through zero velocity:

$$F_f = (\alpha_0 + \alpha_{1*} e^{-|v/v_0|} - (\alpha_0 + \alpha_{1*}) e^{-|n v/v_0|} + \alpha_2 |v|) sgn(v) \quad (3.6)$$

where  $\alpha_{1*}$  is an implicit function of the static friction level and  $n$  gives a representation of the slope of the friction-velocity curve near zero velocity.

The models thus far still lack one major characteristic. From tests performed with (3.6) in [9] it was found that this type of *static* model does not reflect the behaviour of the true system. Static friction models show the same friction characteristics on acceleration and deceleration. True systems do not. One possible explanation could be that while systems need an extra push to start from rest (due to the static friction), once in motion, they are not as quick to come to rest again due to the fact that the bonds that caused the static friction need to form while at rest. Thus the systems is not pulled across the same *Stribeck hump* on deceleration, as during acceleration. This would imply that there must not only be the hysteresis associated with Coulomb friction, but there must also be some hysteresis in the friction model itself (i.e. in the curve of Figure 3.4). This is verified in the literature where in general the phenomenon is referred to as *frictional memory* and results in a new class of *dynamic* friction models.

A simple approach to adding memory to a friction model would be to include a lag between the velocity, and the resultant friction. If we use  $F_f(v, F)$  as the friction model of (3.5), and assume that a lagged friction  $F_{f-L}$  is to be used as the actual friction force at time ( $t$ ), then we can have one of two possible formulations:

$$\begin{aligned} F_{f-L} &= F_f(z, F) \\ \frac{Z(s)}{V(s)} &= \frac{1}{\tau s + 1} \end{aligned} \quad (3.7)$$

or

$$\frac{F_{f-L}(s)}{F_f(s)} = \frac{1}{\tau s + 1} \quad (3.8)$$

where  $\tau$  is the amount of frictional lag and  $z$  is an internal, unmeasurable state of the friction model. A frictional lag of 3ms was suggested in [4] for a more complex formulation. It is not known what effect either of these realizations of frictional lag has on the value of 3ms.

Many different dynamic friction models have been proposed in the literature. The following traces the development of one set of models important to the friction compensation techniques discussed later. In [15], the paper cited by Olsson [32] in his Ph.D. as being the first presentation of this model, the following is used:

$$\begin{aligned}\frac{dz}{dt}\epsilon &= v g(v) - |v| z \\ g(v) &= \alpha_0 + \alpha_1 e^{-(v/v_0)^2} \\ F &= z + \alpha_2 v\end{aligned}\tag{3.9}$$

This results in a transfer function for  $z$  (again, an internal, unmeasurable state of the model), of

$$Z(s) = \frac{g(v) v}{\epsilon s + |v|} = \frac{v}{s \epsilon/g(v) + |v|/g(v)}\tag{3.10}$$

with a steady-state value for  $z$  and friction of

$$\begin{aligned}z_{ss} &= \operatorname{sgn}(v) g(v) \\ F_{ss} &= g(v) \operatorname{sgn}(v) + \alpha_2 v \\ F_{ss} &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) + \alpha_2 v\end{aligned}\tag{3.11}$$

However, in [32] and in [16] (referenced by Olsson as the second appearance of the model). The form is:

$$\begin{aligned}\frac{dz}{dt} &= v - \frac{|v|}{g^*(v)} z \\ g^*(v) &= \frac{1}{\sigma_0} (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \\ F &= \sigma_0 z + \sigma_1 \dot{z} + \alpha_2 v\end{aligned}\tag{3.12}$$

with  $\alpha_0 = F_C$  and  $\alpha_1 = F_S - F_C$ . The transfer function becomes

$$Z(s) = \frac{v}{s + |v|/g^*(v)}\tag{3.13}$$

and the steady-state value for  $z$  and friction become

$$\begin{aligned} z_{ss} &= \operatorname{sgn}(v) g^*(v) \\ F_{ss} &= \sigma_0 g^*(v) \operatorname{sgn}(v) + \alpha_2 v \\ F_{ss} &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) + \alpha_2 v \end{aligned} \quad (3.14)$$

Finally, in [13] and in [33] they use a model which is of the form (and name it as the LuGre model):

$$\begin{aligned} \frac{dz}{dt} &= v - \sigma_0 \frac{|v|}{g(v)} z \\ g(v) &= \alpha_0 + \alpha_1 e^{-(v/v_0)^2} \\ F &= \sigma_0 z + \sigma_1 \dot{z} + \alpha_2 v \end{aligned} \quad (3.15)$$

This results in a transfer function for  $z$ :

$$Z(s) = \frac{v}{s + (\sigma_0 |v|)/g(v)} \quad (3.16)$$

with a steady-state value for  $z$  and friction of

$$\begin{aligned} z_{ss} &= \operatorname{sgn}(v) g(v)/\sigma_0 \\ F_{ss} &= g(v) \operatorname{sgn}(v) + \alpha_2 v \\ F_{ss} &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) + \alpha_2 v \end{aligned} \quad (3.17)$$

If the simple lag from (3.8) is applied to the Tustin steady-state friction model in (3.5), we get:

$$\begin{aligned} f_{ss}(v) &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) + \alpha_2 v \\ F(s) &= \frac{f_{ss}(v)}{\tau s + 1} \end{aligned} \quad (3.18)$$

or

$$\dot{F} \tau = f_{ss}(v) - F$$

which can be rewritten as

$$\begin{aligned} f_{ss}(v) &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) + \alpha_2 v \\ \frac{dz}{dt} \tau &= f_{ss}(v) - z \\ F &= z \end{aligned} \quad (3.19)$$

or if the viscous friction is not to be lagged, then

$$\begin{aligned} g_{\pm}(v) &= (\alpha_0 + \alpha_1 e^{-(v/v_0)^2}) \operatorname{sgn}(v) \\ \frac{dz}{dt} \tau &= g_{\pm}(v) - z \\ F &= z + \alpha_2 v \end{aligned} \quad (3.20)$$

In all cases,  $z$  (or  $\sigma_0 z$ ) more or less represents the non-linear portion of the friction. To compare, the dynamics of  $z$  in (3.20), (3.9), and (3.15) are:

$$Z_{(3.20)}(s) = \frac{g_{\pm}(v)}{\tau s + 1}, \quad Z_{(3.9)}(s) = \frac{g(v) v}{\epsilon s + |v|}, \quad Z_{(3.15)}(s) = \frac{g_*(v) v}{g_*(v) s + |v|} \quad (3.21)$$

or

$$Z_{(3.20)}(s) = \frac{\operatorname{sgn}(v) g(v)}{\tau s + 1}, \quad Z_{(3.9)}(s) = \frac{\operatorname{sgn}(v) g(v)}{\frac{\epsilon}{|v|} s + 1}, \quad Z_{(3.15)}(s) = \frac{\frac{1}{\sigma} \operatorname{sgn}(v) g(v)}{\frac{g(v)}{\sigma |v|} s + 1} \quad (3.22)$$

This shows that in (3.20) the frictional lag is a separate parameter,  $\tau$ , with the steady-state friction level defined by  $g_{\pm}(v)$ . In (3.9) this was expanded to adjust the lag by velocity,  $\epsilon/|v|$ , while leaving the steady-state friction the same. This meant that the lag went down as the velocity increased. The expression was simplified by pulling  $\operatorname{sgn}(v)$  out of  $g_{\pm}(v)$  and multiplying by  $|v|$  to give  $v$ . Finally, in (3.15), the lag was changed from a parameter to a scaled version of  $g(v)$ . No justification was made for this change, however it produces the effect of increasing the lag during sticking and in the Stribeck region. To adjust the magnitude of the lag,  $\sigma_0$  is adjusted. With this modification,  $z_{ss}$  is also modified, and no longer represents the non-linear part of friction, but it is rescaled by  $\sigma_0$  when used to get  $F$ . Olsson [32] gives physical meaning to the modified configuration by referring to  $\sigma_0$  as the spring constant of bristles (causing the friction), and  $z$  as the average deflection of the bristles. Finally, according to [32] the addition of  $\sigma_1 \dot{z}$  was done to fix problems in the hysteresis curve.

### 3.1.1 Adverse Effects of Friction

The various characteristics of this complex nonlinear function show up in different ways depending on the motion of the object in question. The simplest characteristic

is caused by Coulomb friction. When the object is at rest it effectively appears as a *deadband* when comparing the applied force to the effective force acting to move the object. Once in motion however, the Coulomb friction continues to act in the direction opposing motion, and is now independent of the applied force, thus eliminating the deadband-like behaviour.

In a control scenario without integral action, the presence of Coulomb friction will also create a deadband, within which error signals will not be able to generate sufficient control action to overcome Coulomb friction. With integral action and delay, Coulomb friction can cause limit cycles.

The fact that static friction is greater than Coulomb friction can create interesting, and troublesome effects. The negative slope in the friction velocity curve between static and Coulomb friction results in an unstable region within the system dynamics. If a system is controlled (with PD velocity control or PI position control) such that the objective is a steady velocity close to the unstable region, then the mass will not move smoothly, but rather will jump in a slide-stop-slide-stop motion called *stick-slip*. This occurs because the force must build high enough to overcome the static friction. Once overcome and moving, the friction level drops (along the Stribeck curve), causing increasing excess force, and increasing velocity until the object accelerates past the desired velocity. This finally starts generating decreasing control force. As the mass slows down, the instability of the Stribeck curve is reversed, decelerating the object to a halt. If the system is under position control (PID), with a fixed position reference, the resultant motion is usually a limit cycle or *hunting* around the reference. This occurs since the integral action builds up to overcome the static friction, then overshoots the reference with the excess acceleration, as above, before coming to rest again. The cycle then reverses in the opposite direction. If no integral action is used, the result is the same as with Coulomb friction. The deadband simply becomes larger.

The more complex dynamic characteristics of friction cause it to exhibit

different magnitudes upon acceleration and deceleration. This could help to stabilize the effects of stick-slip motion near a critical velocity. The effect would appear as a damping of the cycles and would result because on deceleration, the frictional force would not have time to build to the full level of static friction before the cycle is reversed.

Other complex behaviour includes *presliding displacement*. This occurs in the region of sticking. It has been found that rather than exhibiting absolute restraining force in the stick region as a Karnop model would suggest, the friction tends to behave like a very stiff (but damped) spring in this region. This means that forces applied from rest that are lower than the static friction level tend to cause small (elastic type) motion returning to the original position once the force is removed.

## 3.2 The Pneumatic Valve

There are many different valve and actuator designs on the market [19]. All work slightly differently from each other. Some valves control the flow by moving a plug in a linear motion, while others use a rotational motion. Some valves are used primarily for on/off control, while others are used in regulation tasks. Of interest in this report are valves that perform a regulatory function. The motion of the valve is relatively unimportant, since friction, and linkage dynamics can exist in linear and rotational motion, and in general, mass, force and position are simply replaced by inertia, torque and angle in the dynamic equations.

Actuators are the devices which are used to position the valve plug, in order to control the actual flow. There are likely even more actuator designs than actual valve designs. They can be broadly grouped into three categories; pneumatic, hydraulic, and electric-servo. Pneumatic actuators tend to provide the worst performance of the three. The other two can be used to produce higher, faster torques, and more accurate positioning than pneumatic systems. Nonetheless many plants use predominantly pneumatic systems. There are several reasons for this. First, pneu-



today. In the paper mill, pneumatic valves are used in the consistency control loops described in the previous section, and also in the brightness control loops where addition of bleaching agents requires tight packing and results in high friction. In both these cases, tight regulation of the control variable is necessary to produce uniform product. For this reason, pneumatic systems are the focus of this chapter.

Figure 3.5 shows the design of a typical double acting pneumatic actuator and valve. The valve position is controlled by allowing high pressure supply air to flow into one chamber of the actuator, while evacuating air from the other. The relationship between airflow and position is unstable, thus an inner loop controller (the positioner) is always included to ensure stable valve position response to the control pressure signal. Valves using a spring and diaphragm actuator, where air pressure generates force in one direction and a spring in the other direction are naturally stable, and can be driven without an inner loop controller. The double acting actuators can be smaller, lighter and cheaper for the same application, and therefore tend to be in wider use [19]. This thesis focuses on issues associated with this type of actuator.

The positioner can be a self contained unit or an integral part of the actuator. Mechanical feedback through levers and springs is usually the method used to control this loop. A limited adjustment of the feedback gains may be possible, by either changing the preload on one of the springs, or the action point of one of the levers. The system receives its control signal as a pressure reading (either directly in air pressure or from an *I/P transducer* which converts a current signal to a pressure signal). The pressure signal is balanced, mechanically against the position of the actuator stem. An imbalance controls the flow of air to the two chambers of the actuator, through a small, fast pilot valve.

Figure 3.6 shows a block diagram of this type of valve within a process and under the control of the *Distributed Control System* or *DCS* which controls whole sections of the mill (the actual control algorithm is usually implemented in

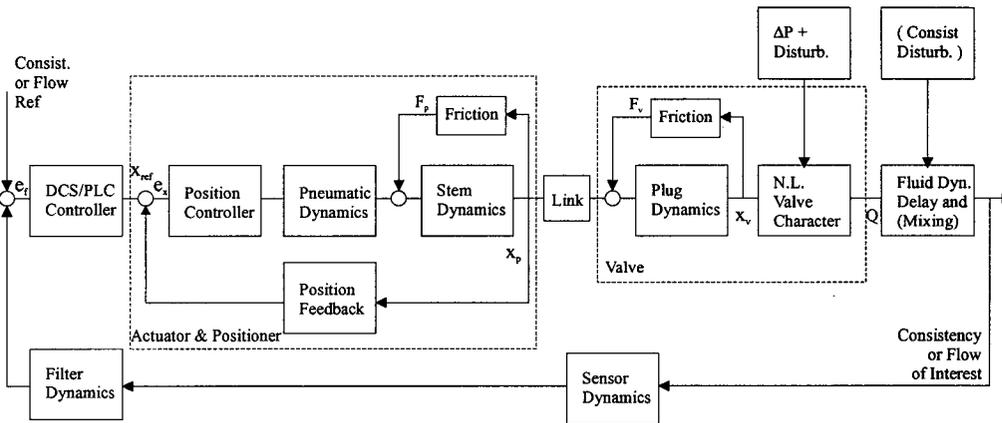


Figure 3.6: A block diagram representation of the main dynamics in a valve system under automatic control.

a dedicated unit called a *Programmable Logic Controller* or *PLC*). As can be seen from the diagram, there are many sources of nonlinear behaviour. Degradation in performance can occur with increasing friction in either of the two friction blocks (also marked  $F_p$  and  $F_v$  in Figure 3.5). Sensor readings are likely noisy and nonlinear. When controlling a flow consistency, delay between an injection point (where the control is performed) and a measurement point, can be substantial. Finally, a loose linkage between actuator and valve would be a source of backlash in the system. In such a case, the dynamics and friction characteristics of the valve and actuator can act independently making analysis difficult. Also, a loose linkage means that a positioner is no longer providing useful feedback of the valve position, and thus even the most advanced positioner (with built in friction compensation) would still be subject to the adverse effects of friction and backlash in the valve. For this analysis, it will be assumed that the linkage is fixed, with no backlash. This can allow the stem and plug dynamics to be modelled as one, with the effects of friction combined as well. It is assumed that the positioner is a simple mechanical feedback system.

For most valves in the mill, the controller at the PLC level will be some form of PID. There may still be more complex algorithms running at a higher level,

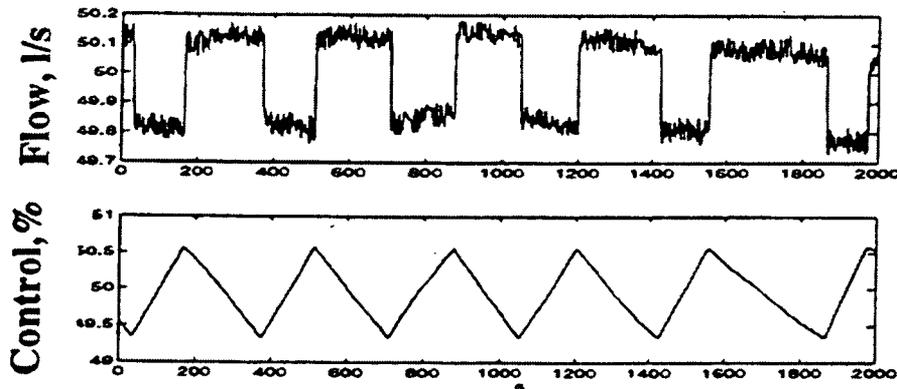


Figure 3.7: The characteristic self-induced oscillations caused by friction. This represents the signals from a flow control loop [35]. The upper graph shows the typical square wave of the measured flow while the lower graph shows the typical triangular shape of the control signal. Time scale is in seconds.

coordinating interaction between variables, but these are likely to feed reference signals to the PID. The typical output that results when there is friction (including static friction in the valve) is the hunting described in the previous section. This is illustrated in Figure 3.7. Note the square-like oscillations in the output and the triangular oscillations in the control signal. These oscillations can vary in magnitude and period. The most detrimental are those with long periods (50-200s) since they are not damped out by any tank dynamics, as seen in the first part of the thesis. These are also the frequency ranges cited in references from industry as being the main friction related problems [10, 23, 35, 38].

### 3.3 Problem Definition

Given the prevalence of pneumatic systems, and the characteristics they entail, the issue becomes one of pushing their performance to the limit, while not over taxing them so as to induce undesired behaviour. Valve manufacturers have already begun to address this issue, by designing valves that use low friction packing, and

positioners designed to reduce the ill effects of friction (by taking advantage of the tighter feedback available in the inner loop). Process and installation design can also help to address the problem in critical applications by including devices that help overcome the limitations of pneumatic system (the use of boosters for example [19]). The fact still remains that there are many old valves in use that do not have these features, where upgrading cost is prohibitive, but performance is affecting the end product. These valves could potentially benefit from application of alternate control techniques.

A second issue involved in any positioning system is the effect of wear. As the valve is used, its components will wear down resulting in undesirable behaviour. Although routine maintenance would help alleviate this problem, it may not always catch the problem. In less cost conscious times, a common insurance for this type of problem would have involved redundant valves with bypass lines that would enable the process to continue running while one of the two valves was serviced. In today's environment of tight budgets and slim margins, such luxuries are rarely available [7]. Thus, if the poorly behaving valve is producing critical oscillations in the process, the whole system may have to be shut down immediately upon discovery of the problem, a costly response to a single failure. In the more common scenario the failure is not critical, and the decision is to continue running with the poorly performing valve, for durations as long as a month [7]. This may also produce disparities in the final product until it is fixed, and is therefore also a costly problem for quality conscious plants. Short of complete failure of the valve however, if increasing friction is the culprit in the poor performance, modified control strategies that allow the process to continue to operate at near optimum levels until the next scheduled shut down (at which point the problem can be properly addressed) are attractive.

In order to effectively consider possible control strategies for the system outlined above, the following constraints must be taken into account:

- The inner loop of the position controller is fixed

- For a short term solution, the control should be done from the DCS
- The current configuration provides no access to the pressure in the actuator chambers (and thus the force being applied). The same applies to the stem position measurement (assumed to be high precision). There is thus no possibility of position or velocity feedback.
- The measured control variable will be related to valve position in some complex nonlinear fashion. It is however subject to high noise, sensor dynamics, and most importantly, a substantial delay and also potentially lag.
- In general, valves in processes are used for regulation purposes. They are required to respond to setpoint changes, but in general, the control loop's disturbance rejection characteristics are much more important than tracking of a continuously changing reference. Depending on the disturbance characteristics and control objectives, a valve could spend much of its time at rest, or close to a specific point, changing directions continuously. Position control will predominate, as will *hunting* if the static friction is significant.
- If the valve is stuck at rest, the integral of the control signal can be viewed as roughly proportional to the force being applied. Once in motion, the relationship switches to the control signal being roughly proportional to velocity, although any desire to control force remains through an integral action.
- The integral relationship to force implies lag, meaning that instant control of force is impossible.
- The fact that the inner loop is inaccessible, means there is no way to know the position error, and there is no way to zero it exactly.

These issues all need to be considered when evaluating the control strategies available. The emphasis in this thesis is on systems where the inner loop is not

designed to handle friction, and the situation where wear has deteriorated the performance of the loop. The control must be applied through the slower outer loop, due to the requirement not to halt the process. The strategy is generally intended as an interim solution, as specified above.

### 3.4 Current Friction Compensation Techniques

Much effort has been focused by the control community on friction compensation in recent years. Armstrong-Helouvry, Dupont and Canudas De Wit [4] do an extensive survey of all aspects relating to systems with friction. In terms of control, they break the area into four categories:

1. Regulation: This involves control where the object spends time at rest. Friction problems are characterized by limit cycling or *hunting*.
2. Tracking with Velocity Reversals: This involves high or low speed motion. Friction problems typically result in unsmooth motion through zero velocity, or potentially undesired stand-still upon crossing zero velocity. When occurring in 2 dimensional circular motion for example, one axis of motion may stop while the other keeps moving resulting in a feature called quadrature *glitch*.
3. Tracking at Low Velocities: This type of motion is differentiated from the previous one in the fact that the desired motion is in one direction, without velocity reversal. Friction problems occur in this case due to Stribeck behaviour, and an attempt to move at the steady velocity in the unstable region of the friction/velocity curve. This usually results in characteristic *stick-slip* motion.
4. Tracking at High Velocities: This type of motion typically does not reside in the unstable region of the Friction/velocity curve but rather in the stable, positively sloped region associated with viscous friction. The primary friction

related problem is non-linearity, and increasing tracking error with increasing speed.

A great deal of the focus in the literature has however been placed on rate control (the last three points). Unfortunately, the problems encountered in the pneumatic valve are primarily related to the first point. There is however still insight to be learned from these other problems.

A number of non-model based techniques exist, and are summarized in [4]. The most obvious is to try to minimize the friction itself. This can be done with better lubrication (or better stem packing in the case of a valve). It can also be minimized with proper maintenance of the valve. Frictional effects could also be minimized in the design of the system itself, by changing system dynamics, or minimizing forces on surfaces with high friction. In the long run, these are likely the best solutions to the problem, and are slowly being applied by valve manufacturers, but at the moment, non-ideal valves still exist, where maintenance has not been done, or not solved the problem.

Dither is a common approach for overcoming the effects of friction. This involves imposing a high frequency motion on the joint. It can either be done normal to the direction of friction (which would tend to modify the magnitude of the friction) or it could be applied parallel to the direction of friction (in which case it has an effect of averaging the nonlinearity). Unfortunately, both of these methods require relatively direct access to the point of sticking, in order that the high frequency signal can be applied there. This is limited by the integrating characteristics of the pneumatic actuator associated with most control valves.

High-bandwidth, high gain control can usually be used to provide good reference positioning (see [13]), but this has the obvious disadvantages of requiring high speed and high gain control elements, and making the system sensitive to noise. A related approach, called *Joint Torque Control* attempts to enclose the friction in a high gain high-bandwidth inner force loop. This can then be controlled with a

slower outer loop that need not be concerned about friction.

*Dual Mode Control* is a name given to the use of two controllers on the same process but for different resolutions. This can be done by switching to a second control algorithm, or even a second actuator, in order to perform high precision control after the initial controller has moved the system within range of the second one. Some systems also use two controllers in parallel with a midranging algorithm that ensures the high precision controller does not reach the limits of its range [1]. This approach might be an ideal solution in situations where high friction is unavoidable and tight positioning is still required. This method would however require process redesign.

When possible, the easiest way to perform model based friction compensation is to estimate the parameters of a friction model off-line, and then use these parameters to perform nonlinear feedback control. This approach is extensively covered in [3], and is also used in [26] and [36]. With the given problem, this is not considered an option, due to the desire to maintain operation of the process.

Some authors have taken existing, conventional control algorithms, and made various modifications to them based on their knowledge of the dynamics of friction. Dupont [17] studies the minimum requirements for stiff PD control in order to eliminate stick-slip motion in velocity control. This is important because it tends to indicate that frictional effects can be minimized with high rates of force application. Shen and Wang [39] also attempt to calculate limits on possible PD and PID controller parameters in order to eliminate undesirable friction behaviour. In order to minimize adverse performance imposed by these designs, gain scheduling was proposed. Limit cycles at zero velocity were eliminated with deadbands on the integral action. Since integral action is required to eliminate steady-state error caused by friction, it cannot be eliminated altogether. The deadband is a compromise that still results in steady-state error but is quoted to improve control over situations without integral action. Southward, Radcliffe and MacCluer [41] use knowledge

of the maximum static friction level, in combination with the proportional control parameter and adjust the error signal such that the control signal is always large enough to overcome friction. This results in a *bang-bang* controller that ideally results in zero error for a fixed reference position. Schäfer and Brandenburg [37] apply a similar idea by using a switching controller that increases the control action when in a stuck position. Piipponen [35] looks at tuning strategies on PID controllers to minimize the adverse effects of friction in control valves. Most of these methods require a priori knowledge of the maximum level of static friction. Hägglund [24] proposes a technique specifically aimed at pneumatic valves where little knowledge of the system is required. It also makes modifications to a PID control signal by adding small pulses to it. This method is described in more detail later.

To avoid having to measure the friction parameters beforehand, Friedland and Park [22], Tafazoli, de Silva and Lawrence [43], and Tafazoli [42] use a nonlinear observer to estimate the Coulomb friction level. They produce very good estimates of the friction, including changes (towards the static friction level) near zero velocity. They do however require accurate high-bandwidth measurements of applied force and resultant velocity. Canudas de Wit, Olsson, Åström and Lischinsky [16] also use a nonlinear observer to estimate friction, and need only an accurate measurement of velocity. They however do need a priori knowledge of a large parameter set for the LuGre model that they use.

A number of authors have also created adaptive control strategies for systems with friction. The first group have used conventional model based, RLS or MRAC methods. The work in [14, 13, 6, 46] all rely on relatively complex nonlinear models of friction. In order to make these models more amiable to estimation, they are typically simplified to models that are linear in one or two parameters. In order for these methods to work, they require high-bandwidth sensing and feedback. Canudas de Wit and Lischinsky [13] demonstrate good results in both rate control and positioning tasks, and show quick adaptation to changing levels of friction.

They do comment however, that the positioning task could as easily be solved with simple control methods, given the high-bandwidth loop used in the experiment.

A second approach to adaptation loosens the requirements on the bandwidth of the control loop. Instead of trying to estimate actual parameters of a friction model, these methods try to estimate characteristics of the resultant friction behaviour. The advantage of these techniques over those that do estimate friction parameters is that the sensing and control bandwidth need not be nearly as high. This has been quite successful in repetitive tasks where velocity control is the goal. Maqueira and Masten [29] use the technique to remove the glitch caused by Coulomb friction in velocity reversals. A friction compensation control sequence is added to the output of a conventional controller during velocity zero crossings. The sequence is characterized by a pulse magnitude and duration. These two parameters are adapted based on absolute velocity error near the zero crossing, and integrated velocity error during a window around the zero crossing respectively. Velocity zero crossings are detected with a fast sensor, whose signal is passed through various relay circuits and filters (to avoid multiple detections of a single crossing). Tung, Anwar and Tomizuka [45] use a similar approach, but on a repetitive task. Instead of attempting to capture the zero crossing event, as done in [29] they use the cyclic nature of their process to add the friction compensation at a time one period after the last addition. The compensation is adapted independently across the full cycle based on the error at that point in the cycle on previous passes. Although these methods don't require the same high-bandwidth loop of the friction model based methods, they still require a bandwidth with an accuracy in the range of the errors they are trying to eliminate. These last two methods would unfortunately not be directly applicable to a positioning task.

A related concept, the use of precomputed, pulses to move the system in discrete steps, has also been studied by several authors. This approach uses open loop control actions to deal with the high-bandwidth friction problem, and then

closes a slower loop around this system. As described earlier, Hägglund [24] used fixed open loop pulses to speed a conventional PID controller in finding its objective. Armstrong-Hélouvry [3] did in depth investigations of the nature of friction models, and developed a strategy to control the force in a robot manipulator to within  $1/60^{th}$  of the level of static friction. The method was based on careful investigation of the system in order to select a friction model and appropriate parameters, followed by the generation of a lookup table that was used to compute the control pulses for the desired force output. Hojjat and Higuchi [25] used an electro-magnetic system to generate short, intense force pulses to investigate the possibility of precise position control. Popović, Gorinevsky and Goldenberg [36] did careful tests on a positioning system to develop a thorough lookup table relating pulse height, and durations to corresponding system slide lengths. This table was subsequently used with a fuzzy logic controller to accurately position the system. Yang and Tomizuka [47] also use pulse control to position their system. Instead of testing the system in advance, they use adaptation to deduce a relationship between pulse duration and slide length.

Other approaches have also been taken to controlling systems with friction. Lee, Misawa and Lucca [27] and Slotine and Li [40] show that high-bandwidth sliding mode control can be used to successfully overcome friction. Baril and Gutman [8] use a nonlinear search algorithm to adapt parameters for friction compensation. Tao and Kokotović [44] develop a unified approach to dealing with systems with backlash, deadzone, and hysteresis nonlinearities. Their approach builds into the estimation the ability to switch between parameters depending on the direction of motion.

## **Evaluation**

Of the control techniques presented in the previous section, many cannot be applied to the current problem due to their dependence on accurate high speed sensing of system position and force. The ones which require extensive system testing in order

to establish friction parameters must also be discarded due to the desire to maintain undisturbed process operation, and again due to a lack of accurate, high-bandwidth measurements of valve position.

The methods that stand out for this application are those which use open loop control to address the friction characteristic. Specifically, the work of Hägglund [24] and Yang and Tomizuka [47] is appropriate.

In line with the objectives of this thesis, the *Knocker* developed by Hägglund [24] is meant to be used as a temporary measure on a control valve after stick-slip limit cycles are detected and until such time that the process is shut down for routine maintenance when the problem can be fixed with proper servicing of the valve. The method involves the addition of short pulses to the regular (PI) control signal. The purpose of the pulses is to give the valve just enough energy to overcome static friction, and nudge it to the next closest stick position. The resultant control is thus:

$$u(t) = u_c(t) + u_k(t) \quad (3.23)$$

$$u_k(t) = \begin{cases} a \operatorname{sign}(u_c(t) - u_c(t_p)) & t \leq t_p + h_k + \tau \\ 0 & t > t_p + h_k + \tau \end{cases} \quad (3.24)$$

where  $u_c(t)$  is the conventional control signal, and  $u_k(t)$  is the added Knocker signal and  $t_p$  is the start time of the last knock. The parameters of the Knocker are:  $a$  the amplitude of the knock control signal,  $\tau$  the duration of the knock, and  $h_k$ , the length of time between knocks (start to start). The size of the physical disturbance that reaches the valve is proportional to  $a\tau$ , and therefore this product can be viewed as determining the energy in each knock.

The author places some limitations on usable values for these parameters. To avoid uncontrolled evacuation from the low pressure side of the valve,  $a$  should be kept within the bounds  $1\% < a < 4\%$  of the maximum control signal. It was recommended that  $a$  be kept fixed for the design. The energy would then be adjusted with  $\tau$  which was placed at 2 or 3 times the sampling interval (of 0.2s in their

application on a 150mm valve). It was suggested that  $\tau$  may need to be larger for larger valves and actuators. Finally they suggest setting  $h_k$  to between 2 and 5 times the length of  $\tau$ .

In experiments on a real valve (150mm water flow control valve with bad stick characteristics) they compare performance of conventional PI control, to the same control augmented by the Knocker. They report a reduction in integrated absolute error to between 55% and 75% of the original error, and reductions in integrated squared error to between 31% and 54% of the original error. They also suggest that performance may be improved by incorporating a deadzone on the integral portion of the controller, to reduce the amount of limit cycling once a point is reached close to the desired setpoint. Figure 3.8 shows the improvements for one test of this controller over conventional PID control.

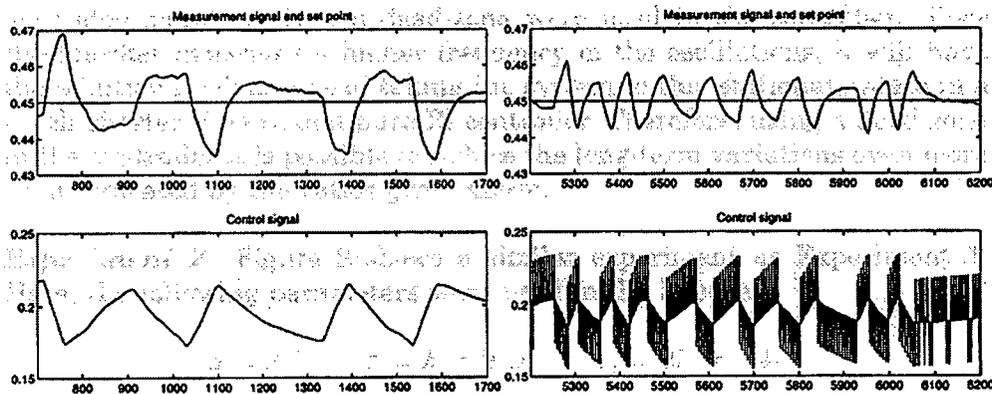


Figure 3.8: The time domain response of a valve system with (right) and without (left) the Knocker active. Output signal is on top and control signal on the bottom [24].

It is stated that the objective of the Knocker is to quickly overcome static friction with a second advantage of quickly crossing the Coulomb deadzone. This can be seen in Figure 3.8 in that the period of oscillation is greatly reduced. A significant amount of cycling is still present however, and even with an integral deadzone implemented as suggested by the author, the system must still rely on a

knock to reach the reference position by chance.

As described in Section 3.1, the effect of static friction in a PID control loop with slow dynamics is to create an overshoot whenever the integral action builds up enough to overcome static friction. This is because the control signal must be large enough to overcome static friction, but once motion starts, much less force is required to maintain motion. The excess force, which had to be generated by integrator windup, is no longer required but cannot be (and is not) undone fast enough, due to the slow system dynamics and sensing.

If the stem is not moving (which is the assumption here, since we want to "knock" the valve to the correct position) and we are not at the desired position (ie there is an error signal being fed to the actuator), then the dynamics of the inner loop imply some form of integral control. The Knocker applies a pulse to the control signal, meaning it increases the position error during the time  $\tau$ . The result, after integration, of each "knock" is to create a force step rather than a force pulse. This means that force is never reduced, thus resulting in a very similar sized excess slide once static friction is overcome.

The work of Yang and Tomizuka [47] attempts instead to make calculated jumps to the reference position using a method called *Pulse Width Control* (PWC). They propose using conventional control without integral action (unless necessary to generate non-zero offset) to bring the system to within close range of the reference position, at which point they switch to PWC. PWC involves short, open loop control bursts of finite calculated duration, designed to overcome static friction, and result in a sliding distance that ends at rest at the desired reference position. The distance slid due to an impulse is a nonlinear function of the friction, the system dynamics, and the applied force (magnitude and duration). Using a simplified model of these sliding dynamics, linear in a single parameter, they use an adaptive scheme to estimate the sliding distance, and control the system with small incremental slides.

The system used for the study is one axis of a two axis positioning table. The

model of the system is a sliding free mass, acted on by a control force, friction, as well as viscous damping. The conventional control used is linear state feedback with Coulomb friction compensation. In order to avoid stability problems, the Coulomb friction is only compensated to a fraction of its estimated value (0.9). With this configuration they calculate a maximum steady-state error:

$$|e_{ss}|_{max} = \frac{1}{k_P}(f_s - r f_c) \quad (3.25)$$

where  $k_P$  is the proportional position feedback gain,  $f_s$  and  $f_c$  are the static and Coulomb friction, and  $r$  is the Coulomb friction compensation ratio just mentioned. This maximum error is important to the control algorithm, because the threshold for the switch to PWC must be greater than this limit.

In order to generate the appropriate pulse widths for a specified position error, they needed a model that would predict the displacement as a function of pulse duration. Several theoretical models were created, and compared to experimental data. The experimental data had some amount of variation, which was attributed to variations in Coulomb friction. A simplified model that could easily be used for adaptation, and which captured the primary nature of the displacement was selected:

$$d = b t_p^2 \operatorname{sgn}(f_p) \quad (3.26)$$

where  $d$  is the displacement due to the pulse,  $f_p$  is the applied pulse magnitude,  $t_p$  is the pulse duration, and  $b$  is a function of the system mass, friction levels, pulse magnitude, and other system characteristics.  $b$  is left to be estimated by an adaptation algorithm.

To model the dynamic behaviour of this system, they use a discrete system where the input is the nonlinear function of the pulse duration (3.26) and the sampling time is determined by the time it takes the system to come to rest after a pulse. The system is thus represented as follows:

$$x(k+1) = x(k) + b u(k) \quad (3.27)$$

$$b u(k) = b t_p^2 \operatorname{sgn}(f_p) \quad (3.28)$$

where  $x(k)$  is the measured position at stopping time  $k$  and  $u(k)$  is the control input, which must be applied using an appropriate pulse duration, back-calculated using (3.28). The pulse amplitude,  $f_p$ , is proposed to be kept constant but at a level exceeding the stick level  $f_s$  and less than the actuator saturation level. The controller uses simple proportional feedback of the form:

$$u(k) = \frac{K_c}{\hat{b}} e(k) \quad (3.29)$$

$$e(k) = x_{ref} - x(k) \quad (3.30)$$

As  $\hat{b} \rightarrow b$ , the error dynamics become:

$$e(k+1) = (1 - K_c)e(k) \quad (3.31)$$

producing an asymptotically stable system for  $0 < K_c < 2$ .

The unknown parameter  $\hat{b}$  can be calculated using a standard recursive estimation using one of two error formulations:

$$e_o(k) = d(k) - \hat{b}(k-1) u(k-1) \quad (3.32)$$

$$e_i(k) = u(k-1) - d(k)/\hat{b}(k-1) \quad (3.33)$$

where  $d(k)$  is the slide distance  $x(k) - x(k-1)$  due to the pulse at  $(k-1)$ . The authors prefer the first method due to calculations performed in the estimation. The appearance of  $d^2(k)$  in the second method is replaced by  $u^2(k)$  in the first, reducing sensitivity to noise. A model reference adaptive controller was also proposed. Both methods were tested on a simulation and on the real system. For the tests,  $K_c$  was set less than one to 0.8,  $f_p = 2.7 f_s$  (at 371N) and the smallest pulse duration on the system was 0.25ms. The authors demonstrated good results, which approached and stayed at the reference position without limit cycles.

A number of characteristics of their system are different from those in the control valve, making it impossible to apply the method directly:

- They were able to produce an instant force impulse
- Their force impulse could be made larger than static friction
- They had a pure integrating dynamic between pulse and position
- The applied force could be zeroed after each pulse
- No natural position feedback (spring action)

These items will be addressed later in the control design.

### 3.5 Modelling

In order to design and test a control strategy for a valve with friction, it was first necessary to have a simulation of the system. Such a simulation consists of two important parts; the valve and positioner itself, including compressible fluid equations for the air in the actuator chambers, and secondly a friction model.

The valve was modelled after equipment available from Fisher Valves at NorPac Controls. Detailed physical measurements were made of a *Fisher 6"-V-100* v-ball valve and a *Fischer 1061-30* double acting pneumatic actuator. A set of dynamic tests were performed using a Fisher Flow Scanner. The system tested consisted of a *Fisher 8"-V-150* valve, the *1061-30* actuator and a *Fisher 3610* positioner (components recommended by Fisher to be used together). These tests were done *on the bench* (no flow process connected), and all components were new (showing minimal effects of friction).

The equations for the valve model were obtained from work done by Eborn [18] in his Master's thesis. Detailed equations were presented for all aspects of a *150mm* pneumatic valve (a 6 inch valve size). The thesis included intricate details of the mechanical feedback and other parts of the valve. For simplicity, this thesis only uses the ideal gas flow equations associated with the air flows (see [18] for details on the equations and associated assumptions), the ideal gas law for air pressure in the

chambers, and assumes linear relationships in much of the remainder of the model. The mechanical feedback is modelled by a simple PD algorithm. A block diagram of the valve model is shown in Figure 3.9.

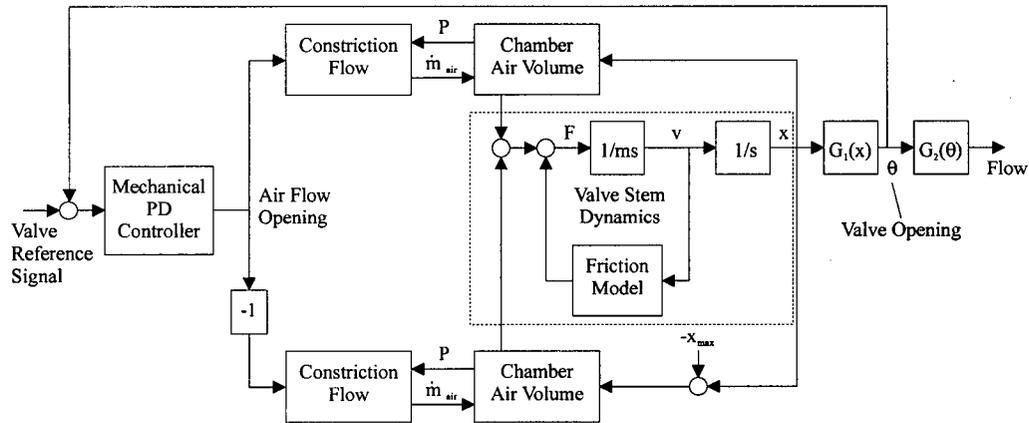


Figure 3.9: A block diagram of the valve model.

A significant problem when attempting to simulate systems with friction is the fast dynamics of the friction behaviour and relatively slow dynamics of the remainder of the system. In order to avoid long simulation times and potential simulation instability, a Karnop style friction model (3.4) was first used. A potentially important characteristic of friction, its change in response to different rates of force application, was not observed with this model. Detailed simulations relating to this problem were conducted in [9]. It was therefore necessary to select a more complex, dynamic model. The LuGre model (3.15) was used instead. This produced the desired effect.

Parameters for the model were obtained, where possible, from the measurements and tests described above. Since the valve being tested was new the level of friction was chosen substantially higher than the 200N identified during testing. Coulomb friction of 800N with maximum static friction 50% higher at 1200N was chosen in rough accordance with the values suggested in [18] (1000N and 1500N

respectively). The dynamic friction parameters ( $\sigma_0, \sigma_1, v_0$ ) can only be estimated with careful measurements and complex nonlinear parameter estimation (see [13]). These parameters were instead chosen using [18] and [32] for guidance, and when scaled appropriately, were similar to those measured in [13] on a real system.

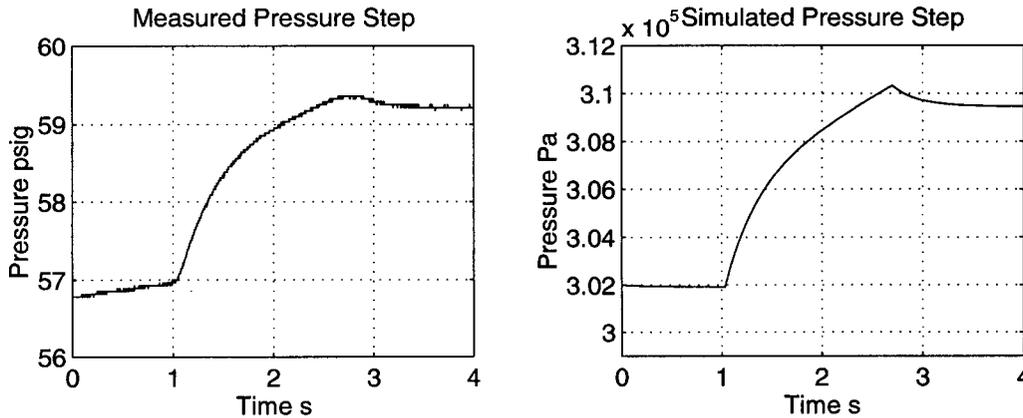


Figure 3.10: Measured and simulated chamber pressure change in response to a small reference step (causing no valve motion).

The parameters for the inner loop PD controller were selected to produce similar chamber pressure curves during small control actions (where no valve motion was observed). The simulated valve was configured with  $300N$  of Coulomb friction and no static friction. Figure 3.10 shows graphs of the simulated and measured pressure changes to a 0.5% control pulse. The magnitude of the step was adjusted so that 5, 1.6s pulses cause the net force to register a change of about  $500N$  (likely  $2 \times 205N$  plus some static friction), as observed in the test. The actual magnitude of the pressure was not matched since both chambers would have to be matched simultaneously to produce the desired net force. Again, to simulate a deteriorated valve, the nominal values were reduced slightly. This will be discussed in further detail later.

The nominal values for the parameters in the model have been summarized in Table 3.1.

Parameter	Value	Comment
Actuator:		
Inner Diameter	12cm	
Height	14cm	
Plate Thickness	2cm	
Plate Travel	9cm	
Shaft & Plate Mass	2.2kg	
Positioner:		
Max Air Flow Orifice	10mm <sup>2</sup>	as in [18], adjusted with PD-
Supply Pressure	653kPa(abs)	80psig (recommended 80-100) -
V-100 Valve:		
Mass	4.3kg	
V-150 Valve:		
Mass	6kg	estimated
Coulomb Friction	210N	from Flow-Scanner tests
Friction Model		
$F_c (\alpha_0)$	800N	In the range suggested in [18] & [24]
$F_s (\alpha_0 + \alpha_1)$	1200N	as above
$\alpha_2$ (viscous fric.)	100Ns/m	estimated
$\sigma_0$	1.0e8N/m	chosen as described above
$\sigma_1$	1.5e4Ns/m	chosen as described above
$v_0$	2.0e - 4m/s	chosen as described above
Inner PD Controller:		
P	0.062	reduced to 0.05 for nominal case
D	0.7	reduced to 0.4 for nominal case
N	2.3	for a derivative term: $P * D / (s/N + 1)$

Table 3.1: The nominal model parameters

## 3.6 Control Design

Using ideas presented by Yang and Tomizuka [47] and Hägglund [24] as well as others discussed in Section 3.4, a controller will now be proposed to better position pneumatic valves with excess static friction. Given the limitations imposed by the configuration of the valve, and the current problem definition, the objective of the control algorithm will be to use discrete open loop control events that nudge the valve from one position to the next, allowing it to come to rest in between. An outer, adaptive loop will be applied to this configuration to determine the control actions and learn some characteristics of the friction.

### 3.6.1 A Careful Look at the Motion of a Valve

Before beginning with control design, it is worth explaining the dynamics that occur in a valve with static friction when a control signal is applied. Figure 3.11 shows the results of a step reference change to a pneumatic valve.

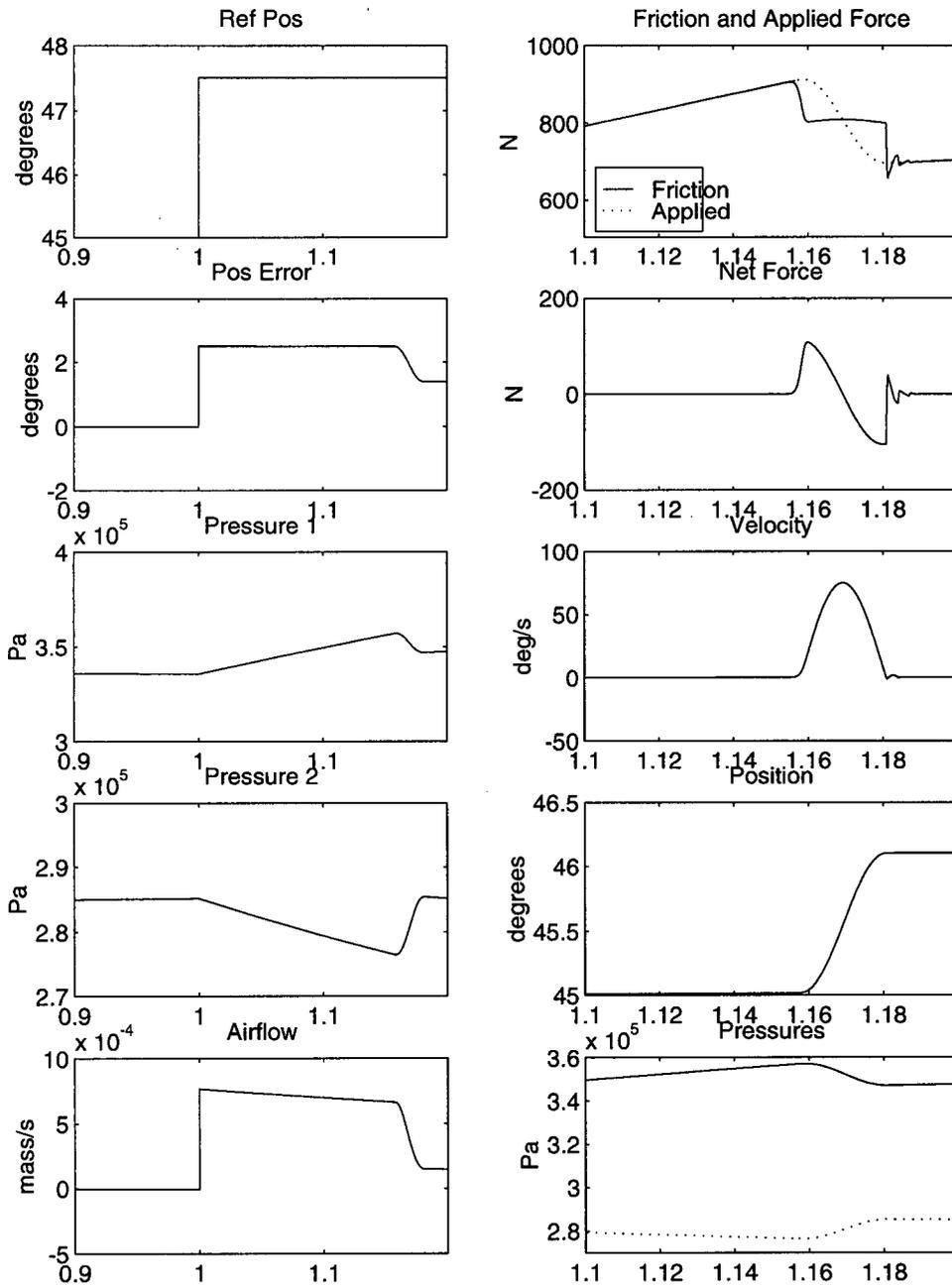


Figure 3.11: Simulated dynamics in a pneumatic valve as a reference signal change is made. The valve is controlled by its internal mechanical PD position controller.

If the control change is reasonably small, a small error is applied to the inner PD controller. This causes a small flow of air into one of the air chambers and out of the other (resulting in a change of pressures and increase in force on the valve). With the applied force less than the static friction level, the friction force builds up to match the applied force, resulting in no net force, no valve motion and a continued error signal. The flow continues while this error is applied to the inner PD controller. As soon as the applied force surpasses the static friction level, the valve begins to move. This increased velocity quickly causes the friction to decrease along the Stribeck curve. As friction decreases (and applied force remaining relatively constant), the net force increases rapidly, causing the valve to accelerate. The motion in the valve changes the volume in the air chambers, causing a rapid re-alignment of pressures eventually resulting in the applied force dropping below the friction level (which is now more or less constant at the Coulomb friction level - changing slightly with viscous friction). This drop in applied force (and now net negative force) quickly brings the valve to rest again. Note that the flow of air into and out of the chambers maintains the same direction during the whole process (set by the PD controller which is still detecting an error). Also note that for this small reference error, the rate of increase in force due to air flow is minor compared to the rapid decrease in force caused by valve movement (and resultant chamber volume change).

This illustrates why the valve will tend to make quantum jumps of position when there is high static friction. Even if the jump were to pass the reference position, the airflow regulated by the inner PD controller could never adjust the applied force fast enough to stop the valve before it passed the reference position. In other words, once the valve is in motion, the changes in force are dominated by the volumetric changes in the chambers (due to motion of the valve) and the fast changes in friction force. The flow rates of air that would be needed to keep up with (much less control) these dynamics would have to be several orders of magnitude

higher than current levels.

### 3.6.2 How This Affects the Design

In [47] the authors specify that their control pulse must be significantly higher than the level of static friction. By applying pulses of such magnitude, the change in net force due to changes in friction across the Stribeck curve are minor. It is therefore reasonable to assume that there will be a fairly consistent relationship between the duration of the pulse and length of slide. The method takes advantage of their knowledge of the friction characteristic, by choosing large forces, which dwarf the friction. It must also rely on the fact that the rates of application and removal of force are faster than the dynamics of the friction.

In the case of the pneumatic valve, the rate of application of force cannot approach that of the friction dynamics. For this reason, it is impossible to generate force pulses that dwarf the friction force. Also, as illustrated by the preceding example, the stopping of the system is dominated by the changes in chamber size. It is therefore not possible to control the stopping of the valve motion with the control signal. This second point, coupled with the inability to produce arbitrarily fast increases in force has the effect of limiting the system to a minimum slide distance.

The approach in [47] used force control. After each slide, the force could be reset to zero, with knowledge that the system was in the centre of the friction deadband. This meant that every slide was started from a consistent point, no matter the desired direction of motion. In the case of the pneumatic system, force is not known, and the slide is stopped by a rebalancing of forces in the two air chambers. There is no certain knowledge of where the system comes to rest in the friction deadzone. If the deadzone is large (ie large Coulomb friction) and the applied energy small however, it could be assumed that at the end of the slide the system comes to rest near the edge of the deadzone. This can be seen in Figure 3.11

where the friction force (in the upper right plot) settles to about  $700N$  (near the  $800N$  Coulomb friction level) as the system comes to rest just after  $1.18sec$ . This implies that the energy needed for the first pulse after a direction reversal would be larger than any subsequent pulses (in order to cross the Coulomb deadband).

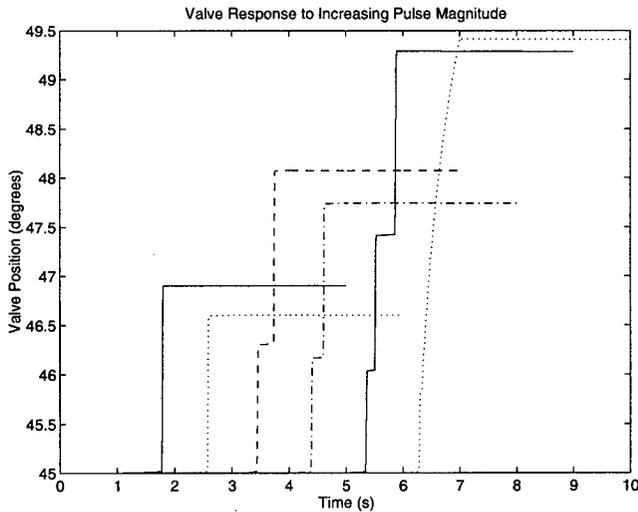


Figure 3.12: Response of the valve to different pulse sizes. Each line is a different test started one second later than the previous, with increasing magnitude pulses from left to right.

order that it moves the valve to the appropriate position. Secondly, even if the system reaches a stuck position where the process output is at the desired level, any discrepancy between the steady-state reference, and the current valve position (as measured by the inner loop mechanical position feedback) would cause continued airflow, and an eventual jump to a new valve position.

### 3.6.3 The Control Algorithm

The solutions to the issues of the previous section will now be addressed the design of the controller is explained.

In [47], the system was a pure integrator between pulse and position. This meant that no steady-state control signal was needed to maintain the position between pulses. Once the reference position was reached, they simply stopped producing control pulses. The configuration of the valve with the inner loop PD position controller, has two negative effects. A steady-state control reference must be generated for the inner loop in

### The Pulse-Slide Relationship

Because of the inability to create large force impulses it was necessary to exploit more subtle features of the friction dynamics in order to make a PWC technique work. The original hope had been to manipulate a pulse height applied to the inner loop. This would have the effect of adjusting the rate of air flow to the chambers. It was hoped that a larger error signal would produce an airflow rate that could rival the pressure change due to chamber volume change, at least for a short period. It was hoped that the effect on slide length would be monotonically increasing with pulse height.

This turned out not to be the case. Instead, over a given region, the slide length decreased with pulse height (see Figure 3.12 which shows the different steps produced by different pulse magnitudes). Further consideration revealed that the effect was due to the frictional lag. A higher force application rate results in less time for the friction to build to the full static friction level, thus causing the

applied force to exceed the friction at a lower level. The cycle in Figure 3.11 completes sooner, resulting in a shorter slide. The slide length does not increase until the pulse height is high enough (and the resultant airflow great enough) to produce an additional slide during the pulse duration. Figure 3.13 shows the resultant relationship between pulse height and slide distance, obtained from a simulation of the valve.

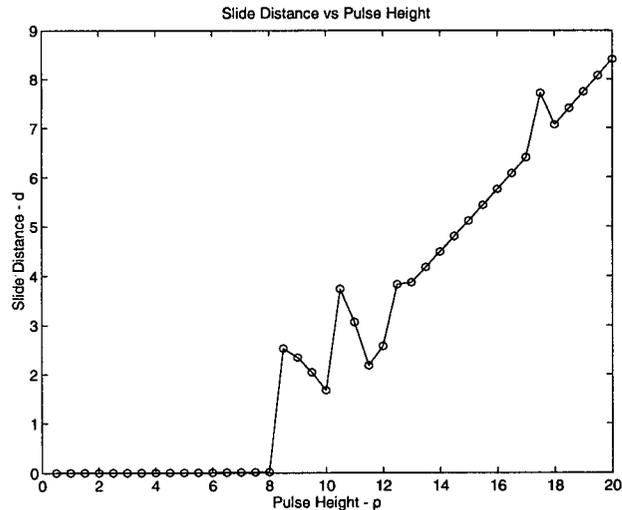


Figure 3.13: Relationship between pulse magnitude and slide distance for pulses with 1 second duration

Although an adaptive algorithm with one parameter would be able to pinpoint a single location on such a curve if repeated attempts were made to achieve a single slide length, the requirement (especially during learning) to jump back and forth across the reference to within various distances from the reference implies that an accurate representation of the whole curve (over a given region) is required.

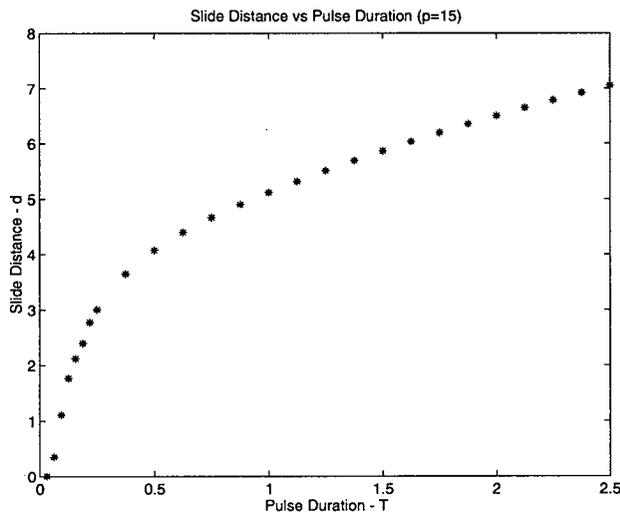


Figure 3.14: Relationship between pulse duration and slide distance for pulses with 15% magnitude. This test was conducted setting the valve near the edge of the deadzone before initiating each pulse.

Instead of adjusting pulse height, the duration of the pulse was adjusted, having selected a pulse height large enough to generate a smooth slide. This produced a pulse duration, slide distance characteristic as shown in Figure 3.14. As can be seen, the continuous nature of this curve would make it much more amiable to estimation.

In order to generate the required steady-state signal, it was necessary to include integral action in the basic controller. The gain of the system was also required in order to couple slide distances to changes in the steady-state control signal. This value would also have to be estimated online.

In Figure 3.12 it was noticed that at very high reference pulses, the slide went from multiple steps to a single step which was quickly quenched, and resulted in a smooth slide until the valve was closer to the reference (or the reference was reduced). This characteristic is supported by previous work [17, 39, 4] which showed that higher rates of force application would quench the stick-slip motion associated with unidirectional velocity control near zero.

## The Basic System Dynamics and Estimation

Beginning with the assumption that the system without friction has first-order dynamics, the equations are:

$$y_{k+1} = a y_k + (1 - a) G u_k \quad (3.34)$$

The eventual steady-state equilibrium of this system is  $G u_k$ . A linear system will tend to approach this point asymptotically. Given the integrating nature of the inner loop controller, the introduction of friction, although producing step-like motion with periods of stand-still, will still result in a system wanting to move towards  $G u_k$  (even if it eventually jumps over it). Therefore, introduce a formulation for a command-pulse generated step, and let the tendency to move towards the steady-state continue to be modelled by the first-order dynamics (even though such asymptotic motion does not occur). This means that the command-pulse will be used to generate changes in the valve position, while  $u$  will be used as a steady-state reference to hold the valve in its new position.

Using either of the two pulse-slide relationships to generate a control signal, combined with the steady-state reference, the following equations can be proposed for the system behaviour from one pulse to the next (refer to Figure 3.15):

$$y_{k+1} = a ( y_k + f(T_k, \omega_k, b) ) + (1 - a) G u_k \quad (3.35)$$

or in derivative form as:

$$\begin{aligned} \Delta y_{k+1} &= y_{k+1} - y_k \\ &= (y_{k+\epsilon} - y_k) + (1 - a)(G u_k - y_{k+\epsilon}) \\ &= a f(T_k, \omega_k, b) + (1 - a) (G u_k - y_k) \end{aligned} \quad (3.36)$$

where the slide distance for a given pulse is represented by the nonlinear function  $f(T, \omega, b)$ , with the relationship between the slide distance, and the independent variables ( $T$  and/or  $\omega$ ) is parameterized in some parameter vector  $b$ . The following

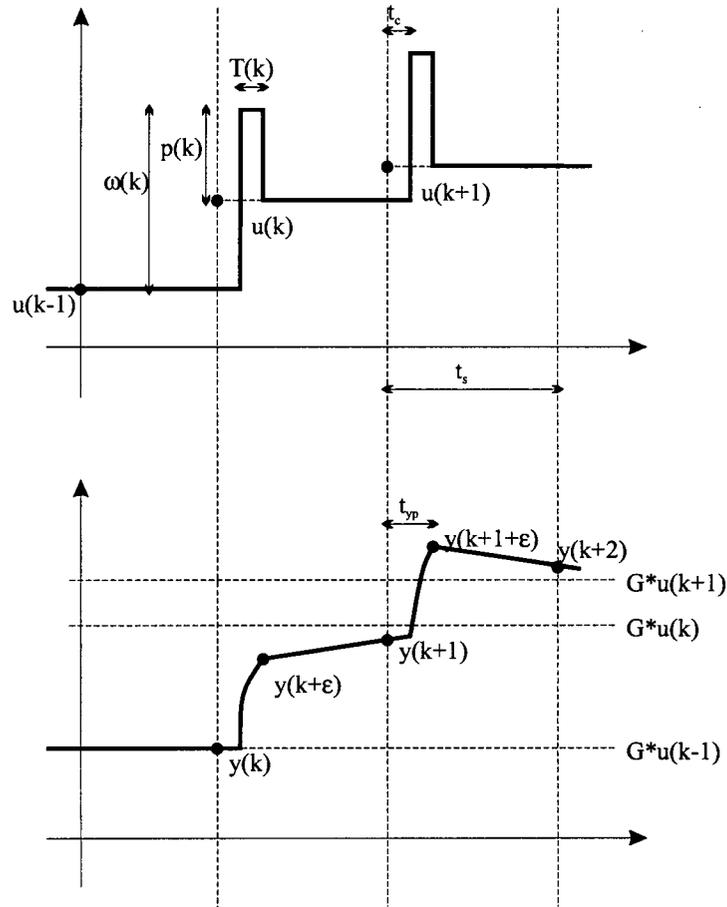


Figure 3.15: Characterized control sequence and system response

parameterizations were proposed:

$$f(\omega_k, b) = b \omega_k \quad (3.37)$$

$$f(T_k, b) = \frac{b_1 T_k + b_2}{T_k + 0.6} \quad (3.38)$$

The option of varying both  $T$  and  $\omega$  was not considered.

Evaluating the above formulation, if the system is fast ( $a \rightarrow 0$ )

$$y_{k+1} = G u_k \quad (3.39)$$

and if the system is slow ( $a \rightarrow 1$ )

$$\begin{aligned} y_{k+1} &= y_k + f(T_k, \omega_k, b) \\ \Delta y_{k+1} &= f(T_k, \omega_k, b) \end{aligned} \quad (3.40)$$

If a control signal (with  $T_k$  or  $\omega_k$  selected to give the desired performance) is generated as follows:

$$\begin{aligned} u_k &= u_{k-1} + f(T_k, \omega_k, \hat{b})/\hat{G} \\ u(t) &= \begin{cases} u_{k-1} + \omega_k, & k t_s \leq t < k t_s + T_k \\ u_{k-1} + f(T_k, \omega_k, \hat{b})/\hat{G}, & k t_s + T_k \leq t < (k+1) t_s \end{cases} \end{aligned} \quad (3.41)$$

then as  $\hat{b} \rightarrow b$  and  $\hat{G} \rightarrow G$

$$\Delta y_{k+1} \rightarrow f(T_k, \omega_k, b) \quad (3.42)$$

meaning that the control set  $(T_k, \omega_k)$  is expected to produce a step equal in size to the change requested with the change in  $u$  (ie  $G \Delta u_k$ ). Thus, if at the previous sample,  $y_k \approx G u_{k-1}$  (regardless of  $a$ ) or if the system is fast ( $a \rightarrow 0$ ) compared to  $t_s$  then

$$y_{k+1} \rightarrow G u_k \quad (3.43)$$

These last two expressions give separate formulations involving  $b$  and  $G$  where  $b$  represents the relationship between pulse size, and output change, and  $G$  represents the steady-state gain of the system (needed to couple pulse commands into the steady-state reference signal  $u$ ). We can use these formulations to estimate  $\hat{G}$  and  $\hat{b}$  using two RLS estimators (with exponential forgetting) with

$$\begin{aligned} e_G &= y_k - \hat{y}_k \\ &= y_k - \hat{G} u_{k-1} \end{aligned} \quad (3.44)$$

and

$$\begin{aligned} e_b &= \Delta y_k - \Delta \hat{y}_k \\ &= (y_k - y_{k-1}) - f(T_{k-1}, \omega_{k-1}, \hat{b}) \end{aligned} \quad (3.45)$$

where  $f$  is evaluated using (3.37) and (3.38). The regressor in the first case (3.37) is simply taken as  $\omega$ , while in the second case (3.38), a two variable regression vector is formulated:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \frac{T_k}{T_k+0.6} \\ \frac{1}{T_k+0.6} \end{bmatrix} \quad (3.46)$$

which is linear in the parameter set  $[\hat{b}_1, \hat{b}_2]$ , and allows a reasonable match of the curve shape in Figure 3.14.

All estimation was done on positive numbers only (meaning negative control actions had their resulting slides reflected to the opposite sign). At this point, any control actions producing negative slides had the slide distance set to zero (on the assumption that the data was invalid due to noise or disturbance), and were thus ignored in the estimation of  $\hat{b}$ .

In order for the estimator to work properly, it is important that motion of the valve is due to the pulse that was applied. This means that between samples there should be no motion of the valve due to other effects than the pulse. The most important adverse effect is the integration that occurs in the inner loop due to discrepancies between the steady-state reference and the actual position of the valve. To avoid this it is recommended that the inner loop P be detuned.

Until this point, all work has been discussed without reference to a specific pulse-slide relationship. In the controller that was implemented, the pulse duration - slide relationship of equation (3.38) was used. Although some of the remaining comments might be applicable to both techniques, much of the remainder of this chapter was written with specific reference to the relationship that was implemented.

### **Non-Zero Initial Force**

If the inner loop is detuned, it is less likely that the force balance in the valve chambers will change significantly between pulses. If this is true, it could be expected that after any slide, the system would consistently come to rest in the same region of

the Coulomb friction deadband. This means, as explained earlier, that less energy would be required for pulses to move the valve in the same direction as the last slide, as opposed to a pulse that tries to reverse the direction. For this reason it was decided to use two sets of functions,  $f$ , with independent parameters;

- $f_{first}(\hat{b}_{first})$
- $f_{subsequent}(\hat{b}_{subsequent})$

The appropriate function would be used in calculating the current control action, based on the direction of the previous control action and the desired current control action. Estimation of the parameters would also be done in parallel depending on the direction of the previous control action.

### Minimum Slide

The fact that the system is limited by a minimum slide introduces a number of issues.

1. It will not be possible to use regular control which approaches the reference asymptotically, since this could eventually place the valve at a distance from the reference that is less than the minimum slide distance. It is therefore necessary, at least when close to the reference, to use dead beat control. Thus we pick  $T_k$  (or  $\omega_k$ ) such that:

$$f(T_k, \omega_k, \hat{b}) = e(k) = y_{ref} - y_k \quad (3.47)$$

2. If the system ever ends up in a situation where the error is less than the minimum slide, it will be necessary to knock the system far enough away from the reference that the next control action can successfully hit the reference.
3. It will be necessary to estimate what the minimum slide distance is. This is done by increasing the estimate every time a pulse is applied that produces no slide.

4. To ensure that this number does not grow unreasonably due to noise, the value must also be reduced when valid slides occur.
5. If the estimate of the minimum slide is too small, and the controller attempts to produce a jump that is smaller than this distance, it will have added energy to the controller that has not yet been expended in a slide. The next attempt to move the valve will then have the benefit of the energy supplied by the previous attempt as well as the current attempt. Use of a data pair consisting only of a single control action, and resultant slide distance, to update the parameter estimates, would produce false results. For this reason, any control action not producing a slide turns off the estimation until after the next slide is recorded.
6. In order to decouple the minimum slide estimate from poor estimates of  $\hat{b}$ , the minimum pulse duration  $T_{min}$  or the minimum pulse height  $\omega_{min}$ , (depending on the algorithm) is recorded instead.

The following algorithm outlines the skeleton of the resultant estimator:

```

if  $\Delta y_k > d_{threshold}$ 
    if wait_for_slide = FALSE
        do regular estimation
            if  $T_{k-1} < 0.3 T_{min}$ 
                 $T_{min} = \lambda_{min} T_{min}$ 
            endif
        else % (wait_for_slide = TRUE)
            wait_for_slide = FALSE
        endif
    else % ( $\Delta y_k < threshold$ )
        if wait_for_slide = FALSE
            wait_for_slide = TRUE
        endif
    endif

```

```

    if  $T_{k-1} > T_{min}$ 
         $T_{min} = \lambda_{min} T_{min} + (1 - \lambda_{min}) T_{k-1}$ 
    endif
else
    %just ignore the data - wait for slide
endif
end

```

### The Controller

The various features outlined above were incorporated into the design of the estimator. The parameters from the estimator were used in a switching control algorithm as outlined below. The values for  $\hat{b}$  used in  $f(T, \hat{b})$  and  $f_{inv}(d, \hat{b})$  were switched depending on a desired direction reversal or not (between  $\hat{b}_{first}$  and  $\hat{b}_{subsequent}$  respectively). The controller uses a deadband around the reference to ensure that oscillations do not result from noise, or small errors in the control action. The controller also changes strategy if the error is greater than some maximum step size  $e_{big}$  (which could not be reasonably produced with the current pulse height). In the example that follows, this is done by requesting steps of  $0.8 \times e_{big}$  in this region. This was further augmented in the implemented controller, to use proportional control (of  $u$  only, no pulses) when outside a region of  $3 \times e_{big}$ . A more elegant approach would switch to PID in this region. It is important however to have a reasonably large region around the reference where several large steps would be required to reach the reference, since these help the estimator learn its parameters better.

```

if abs( $e$ ) <  $e_{min}$                                 %Within Deadzone - No Control
     $T_k = 0$ ;
     $u_k = u_{k-1}$ ;
elseif abs( $e$ ) >  $e_{big}$                             %Outside Max Step - Go Slow
     $T_k = f_{inv}(\text{sign}(e) * 0.8 * e_{big}, \hat{b})$ ;

```

```

     $u_k = u_{k-1} + (\text{sign}(e) * 0.8 * e_{big}) / \hat{G};$ 
elseif  $\text{abs}(e) < f(T_{min}, \hat{b})$     %Inside Min Step - Get out
     $T_k = \text{sign}(e) * (\text{abs}(f_{inv}(e, \hat{b})) + K_{min} * T_{min});$ 
     $u_k = u_{k-1} + (e + \text{sign}(e) * f(K_{min} * T_{min}, \hat{b})) / \hat{G};$ 
else
    %Within Range - Normal pulse
     $T_k = f_{inv}(e, \hat{b});$ 
     $u_k = u_{k-1} + e / \hat{G};$ 
end

```

### Sampling Interval and Noise

As mentioned earlier, the existence of a minimum slide distance forces the control algorithm to be deadbeat. Since such a controller is very sensitive to noise, and again because the control actions are limited by the minimum slide, any attempt to use deadbeat control would have a tendency to amplify the sensor noise. This would be aggravated by any delay in the system. It is therefore necessary to try to filter as much measurement noise as possible. Because the parameter estimates are used for deadbeat control, it is important that they too are accurate. Also, since the system is likely nonlinear from one reference point to another, it must have fast convergence to capture the parameter changes. The combination of these factors resulted in a decision to use a long control interval in order to gain the benefit of additional noise filtering.

By selecting a sampling time sufficiently larger than the process lag and delay, the system would be allowed to reach steady-state between a control action and the subsequent measurement. This meant that no dynamics had to be modelled in the estimation. This is important when considering the switching estimator. If a first-order system with delay were considered, more than one sample worth of data would have to be remembered, and switching between parameter estimates could not easily be decoupled.

The resulting recommendation is to use a sampling time  $t_s$  which is about 3 times the length of the delay and/or lag in the system. Because of the long control interval, it becomes necessary to introduce an anti-aliasing filter for frequencies above  $1/2$  (or more conservatively,  $1/4$ ) the sampling frequency. Since there is no access to the continuous sensor signal, this filter must be implemented in discrete time, at the system sampling rate (usually 1 second). In order to ensure unbiased control action, the measured error signal is passed through an inverse filter (with an equivalent time constant, but implemented at the control sampling frequency). This provides a prediction of the actual error based on the filtered signal.

As mentioned earlier, it is desirable to take advantage of the difference between the system sampling time and the control sampling time to do filtering on the measurements that feed the estimator as well. This would allow a more aggressive forgetting factor, without a fear of noise sensitivity. Because of the desire to maintain a zero-order model of the system, the time constant on this filter must be chosen to provide close to 100% settling during one control interval.

### **Additional Modifications**

In using the controller, several additional modifications were made to improve performance, and limit instability.

It was found that if the estimate of  $\hat{b}$  were to evolve such that  $f(T, \hat{b}) \approx 0$  for all  $T$ , then no further control could remove it from there. This occurred because the steady-state control signal  $u_k$  is incremented by  $f(T, \hat{b})$ . If  $u_k$  does not change, the valve may step due to the pulse, but eventually returns back to the original position (referenced by  $u_k$ ). To avoid this, a protection algorithm was added to help kick the estimator out of this situation. When  $f(T_{max}, \hat{b})$  estimates a value close to zero, it is replaced by a large value ( $e_{big}$ ). Likewise, if the estimator receives a data pair consisting of  $T \approx T_{max}$  and  $d \approx 0$ , then  $d$  is replaced by  $e_{big}$ .

If a disturbance occurs halfway through a control interval, the predictor for

the anti-aliasing filter is not able to produce a proper estimate of the actual system output. If the disturbance is a sudden change that remains stationary at its new level (like a valve slip or any smoothed step disturbance faster than the control interval) then an attempt by the controller to reach the reference on this first signal will not be successful (since there is more error that has yet to pass the filter). This would result in less than optimal response to such errors, and would likely also mean oscillatory response since the remaining error could be less than the minimum slide distance (meaning a total of three or more control actions to overcome the disturbance).

By waiting an additional control interval instead, the predictor would be allowed to come much closer to the actual output. This would ensure a one step response to the error and persistence of the error for less than two control steps. Such a technique would obviously not be advantageous in the general case with persistent disturbances or reference changes, since it would slow the control action to every second control interval. If however, the controller was maintaining steady-state operation in the deadband (ie no control actions) such a technique would be helpful. A test was added for the number of inactive control intervals. If this was greater than or equal to some parameter  $n_{nocontrol}$ , then the sensing of an error, would not cause a control action until the next interval. This has the additional benefit of leaving the valve where it is, even if a single measurement is outside the error band, due to noise in the sensor.

Generating similar problems, the error could have been due to a valve jump (from the desired location to one side or the other) caused by slow integration in the inner loop, of a misalignment between the control reference and the actual valve position. If this were the case, then  $u_{k-1}$  was requesting a position somewhere between the previous output and the current output. Therefore, if the controller was inactive, as before, the corrective action should request a new steady-state reference  $u_k$ , somewhere between  $u_{k-1}$  and  $u_{k-1} + e_k/\hat{G}$  (where  $e_k = y_{ref} - y_k$ ). The best

guess would be to choose

$$\begin{aligned} u_k &= u_{k-1} + 0.5 e_k / \hat{G} \\ &= u_{k-1} + 0.5 f(T_k, \hat{b}) / \hat{G} \end{aligned} \quad (3.48)$$

### Summary of Parameters

The following parameters are needed for the controller:

- Forgetting Factors:  $\lambda_{bf}, \lambda_{bs}, \lambda_{min}, \lambda_G$
- Estimation initialization
- Control sampling time:  $t_s$
- Pulse Height:  $\omega$
- Dead zone:  $e_{min}$
- Max Step:  $e_{big} \approx 0.6 G \omega$
- Max Time:  $T_{max} < 0.4 t_s$  or  $2sec$
- Protection: force  $f(T_{max}, \omega, \hat{b}) \approx e_{big}$
- Replacement Control for  $e > 3 * e_{big}$
- Min Slide Escape:  $1 < K_{min} < 2$  or  $K_{min} = rand[1, 2]$
- Wait an extra interval if  $n_{nocontrol}$  steps without control
- Anti-aliasing filter, and associated predictor
- Filter on output measurements for estimator
- No-Slide threshold:  $0 < d_{threshold} < e_{min}$

Some suggestions were given for these parameters as they were introduced, some suggestions are given above, and some need to be chosen based on reasonable guesses of the system gain and characteristics. A few final suggestions are provided here:

The most critical parameter is the pulse height  $\omega$ . This needs to be large enough to overcome the stick-slip behaviour of friction. A suggestion based on the simulations would be to make it 3 to 5 times the average stick slip jump, divided by the estimated gain of the system.

The deadzone  $e_{min}$  should be chosen with respect to the sensor noise in the system. Assuming noise variance of  $\sigma$ , a Gaussian distribution would place 95% of the readings within  $2\sigma$  of the actual measurement, a safe margin to allow the algorithm with  $n_{nocontrol}$  to handle the rest. The anti-aliasing filter improves this somewhat. For a simple first-order filter with pole  $a$ :

$$f_{ilt} = \frac{1-a}{z-a} \quad (3.49)$$

the expected output variance of a noisy signal with variance  $\sigma$  is:

$$\sigma_{filt} = \sqrt{\frac{1-a}{1+a}} \sigma \quad (3.50)$$

allowing the error band to be reduced significantly.

However, in view of the inaccuracy of the estimation and control, one should not expect zero error. The recommendation is to choose a bound of  $2\sigma_{filt}$  plus about 0.5% of the average output level for control and estimation error. This could be increased if oscillations are too persistent.

### Outstanding Issues

A few issues were not addressed in this thesis regarding the controller. They are listed here for completeness, but no further comment is made.

- No input lag on the inner loop was tested. This could have an effect on pulse shape, minimum slide distance, or the maximum force application rate.

- How to perform careful online tuning.
- The effect of disturbances to valve forces (due to flow forces in the liquid)
- What if one cannot produce the smooth slide?

### 3.7 Controller Performance

The control algorithm was tested in an environment that was meant to represent a consistency control loop (see Figure 3.16). The consistency of a primary flow is regulated by controlling the addition of a dilution or additive flow. The consistency of the resultant flow is measured down stream of the

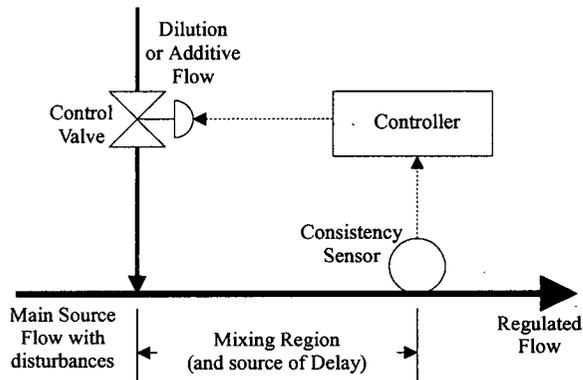


Figure 3.16: A typical consistency regulation loop

addition point, usually some finite distance along the pipe (in order to ensure sufficient mixing of the flows). This mixing distance between the point of addition and point of measurement results in a delay between control actions on the regulatory flow, and measurable changes in the regulated flow.

The main features of the above configuration can therefore be summarized by valve dynamics, valve flow relationship, a mixing dynamic, a gain (between regulatory flow and resultant consistency), a time delay, and disturbances to the system. (see Figure 3.17). Since the valve flow dynamics with pressure disturbances essentially amounts to a nonlinear gain disturbance, and the mixing dynamic will be fast compared to the delay, the basic model was simplified to that shown in Figure 3.18.

The nominal system was tested assuming a desired output of 450 on a range from 0 to 900 on the full range of the valve (given a preselected upstream and

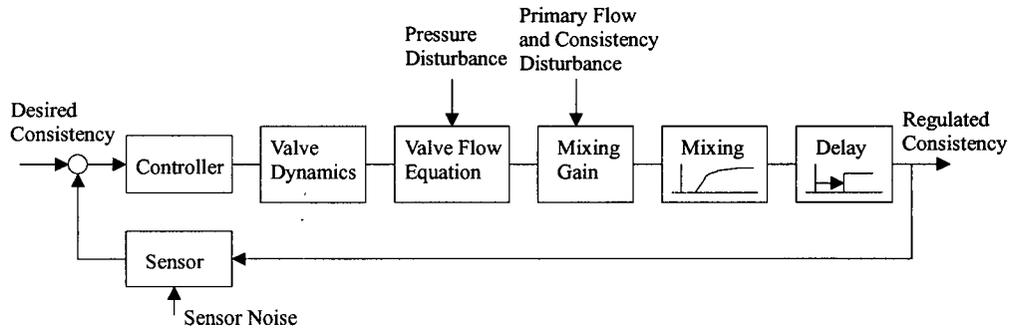


Figure 3.17: A block diagram of the full consistency control loop

downstream pressure on the valve). The valve was designed to take control signals in the range from 0 to 90, resulting in a gain of 10 for the nominal pressure conditions. Sensor noise was added at a standard deviation of 10. Various disturbances were added to the upstream pressure signal causing the unregulated output to fluctuate between 300 and 600 (depending on the disturbance). Delay in the system was taken as 3 seconds. The existing sampling time for the loop was assumed to be 1 second.

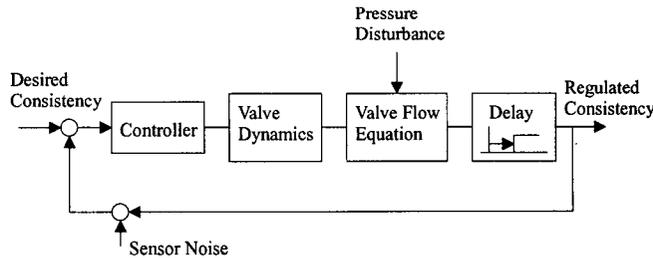


Figure 3.18: A block diagram of the simplified loop used in the simulations

In order to compare the performance of the adaptive controller, tests were also performed with a discrete PI controller. Using a simple Dahlin tuning algorithm, assuming a process time constant of 1 second, gain of 10, delay of 3 seconds and desired closed loop time constant of 7 seconds, the resultant continuous controller is:

$$U(s)/E(s) = 0.01(1 + 1/s) \quad (3.51)$$

and the equivalent discrete controller, for a 1 second sampling time is:

$$U(z)/E(z) = 0.01(1 + 1/(1 - z^{-1})) \quad (3.52)$$

A deadband was also implemented on the outer loop integrator in order to allow a fair comparison between the two strategies. This was set to  $\pm 8$  to match that used with the adaptive algorithm.

When the PI controller was tested on the system, it proved to produce little oscillation of the type identified in industrial applications (even with high friction as specified in Table 3.1). Although oscillations existed, and the magnitude was reasonably high, the period was quite short. Changing the outer loop PI parameters had little effect on increasing the oscillations. The good performance must therefore be primarily due to the high gain in the inner loop, a conclusion justified by the many high gain friction compensation techniques discussed in the literature. Given that the parameters were matched to a new actuator and positioner, it is reasonable to assume that they represent a system that had not undergone any wear. An old positioner from the plant could be expected to exhibit, backlash, higher friction, and less stiff spring action in all its internal components, resulting in a lower gain inner loop with worse response. With this justification, the nominal inner loop control parameters were reduced as specified in Table 3.1. In order to test further deterioration of the valve, the inner loop gain was also tested at a proportional gain of 0.01 (the same as that used for the adaptive controller). This slow inner loop gain tended to produce the types of oscillations shown in the literature and can therefore be considered representative of the inner loop deterioration, typical of a misbehaving valve in industry.

The adaptive controller was run with the following nominal parameters:

- $\lambda_{bf}, \lambda_{bs}, \lambda_{min} = 0.95, \lambda_G = 0.92$
- Estimation initialization  $\hat{b}_s = (10, -5), \hat{b}_s = (60, -2), \hat{G} = 7$
- Control sampling time:  $t_s = 10$
- Pulse Height:  $\omega = 17$
- Dead zone:  $e_{min} = 8$

- Max Step:  $e_{big} = 50$
- Max Time:  $T_{max} = 2sec$
- Protection: force  $f(T_{max}, \omega, \hat{b}) \approx e_{big}$
- Replacement Control for  $e > 3 * e_{big}$ :  $u_k = 0.5 e, T_k = 0$
- Min Slide Escape:  $K_{min} = rand[1, 2]$
- Wait an extra:  $n_{nocontrol} = 2$
- Anti-aliasing filter: First order,  $a = 0.855$  at  $t_{sample} = 1s$
- Filter for estimator: First order,  $a = 0.6$  at  $t_{sample} = 1s$
- No-Slide threshold:  $d_{threshold} = 2$
- Inner Loop Gain: reduced to 0.01

The following figures show the results of simulations with the adaptive pulse controller and the PI controller. Four sets of tests are shown. The first three use the nominal friction parameters in Table 3.1. The last set tests the system under increased friction ( $F_c = 1000N$  and  $F_s = 1700N$ ). In the last test the step pulse height was increased to 20 from 17 in the adaptive controller. Each set of tests used a different disturbance.

Each test is presented over two pages. The first page shows the uncontrolled disturbance in the top graph. The subsequent graphs on the first page show the performance of the adaptive controller. The control signal is presented, followed by the measured output, then a filtered output (to highlight slow oscillations generated by the controller), and finally the parameter estimates. The second page shows the performance of the PI controller. The top three graphs show the performance when the inner loop has its nominal parameters ( $K = 0.05, D = 0.4$ ). The lower three graphs show the PI controller with the misbehaving valve positioner (inner loop

parameters;  $K = 0.01$ ,  $D = 0.4$  - note that these lower inner loop gains are those required by the adaptive algorithm). As with the adaptive controller, the graphs represent the control signal, the measured output, and the filtered output.

In the graphs of the parameter estimates, the dashed line is  $\hat{G}$ , the dot-dashed line is  $\hat{T}_{min}$ , the dotted lines represent the  $\hat{b}_{first}$  estimates, the solid lines the  $\hat{b}_{subsequent}$ . In both these later two cases, the larger number is  $b_1$  and the smaller is  $b_2$ , as per equation (3.38).

In evaluating the performance of the controllers, two things are important. First, the general amount of deviation from the setpoint needs to be evaluated. Second, the extent of any persistent oscillatory behaviour needs to be identified. To do the former, an integrated error measure is used. Instead of the standard integrated squared error, an integrated absolute error was used. Since no attempt was made in the adaptive controller to optimize the response far from the reference, this method tends to have larger amounts of error just after a step disturbance. To equalize this, all errors larger than a threshold were discarded in calculating the measure. Finally, the first 1000s were discarded to allow the adaptive controller time to stabilize. The standard integrated absolute error is then:

$$IAE = \frac{1}{3000} \sum_{k=1000}^{4000} |e(k)| \quad (3.53)$$

where the limited version,  $IAE_{lim}$ , is divided only by the number of points used.

It can be seen from the graphs, that the adaptive algorithm performs best on the infrequent step disturbances. The  $IAE_{lim}$  is improved in all cases except when compared to the nominal PI for the sinusoidal disturbance. In all cases the adaptive algorithm has none of the large low frequency oscillations of the detuned PI controller. It also improves slightly over the nominal PI controller, tending not to have the persistent oscillations in the 40 second range.

Finally, the online parameter estimates are evaluated for the first test (Figure 3.20). The final parameters at 4000sec are;  $\hat{b}_{first} = [53.28, -5.303]$ ,  $\hat{b}_{subsequent} = [83.99, 1.347]$ ,  $\hat{T}_{min} = 0.0733sec$ ,  $\hat{G} = 9.09$ .  $\hat{G}$  is lower than the nominal gain of 10

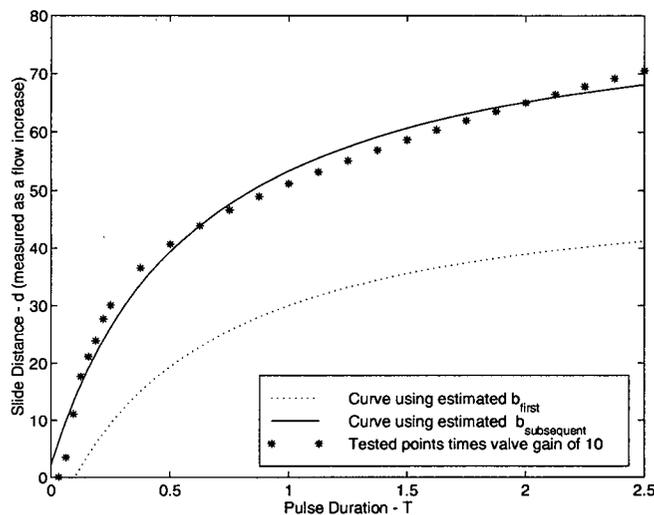


Figure 3.19: The  $\hat{b}$  parameter estimates compared to the nominal test results (scaled by the nominal valve gain of 10)

because the pressure disturbance has reduced the system gain at this point (see the top graph in Figure 3.20). The  $\hat{b}$  parameter estimates are graphed together with the ideal parameter curve (from Figure 3.14) in Figure 3.19. The ideal curve was scaled by the nominal valve gain of 10 and should be close to  $b_{subsequent}$  because it was generated by setting the initial force near the edge of the deadzone before each test. A good match is seen between  $b_{subsequent}$  and  $\hat{b}_{subsequent}$ . The shorter steps seen in the graph of  $\hat{b}_{first}$  is exactly as expected since some of the applied energy is required to cross the Coulomb deadzone. It can also be seen from this curve that short pulses produce no slide at all. The smallest pulse length to produce a slide for  $\hat{b}_{first}$  is very close to the value estimated separately by  $\hat{T}_{min}$ .

### 3.8 Summary

This chapter has analysed the problems associated with friction in a pneumatic control valve. The nature of friction and some existing control techniques were evaluated. A new approach was presented making use of a knowledge of friction behaviour

but without requiring a detailed internal model of friction nor high-bandwidth sensing or actuation. The control algorithm was tested on a detailed model of the valve and was compared to conventional PI control. The adaptive algorithm performed well, converging to valid estimates of the system parameters, and tracking changes in system gain. The controller performed substantially better than PI control when applied to a system with slow abrupt disturbances. In all other cases it still outperformed PI control if the valve positioner (inner loop controller) was misbehaving. It performed as well or only slightly worse (in the case of constantly changing disturbances) than PI when the valve positioner was optimal.

The fact that the valve positioner (inner loop controller) had to be detuned (under PI control) in order to observe the types of oscillations discussed in the literature tends to indicate that friction may not be the sole cause of serious valve oscillations. Friction in combination with a faulty or undersized positioner, undersized actuator, or insufficient air-supply flow may be necessary to produce these oscillations.

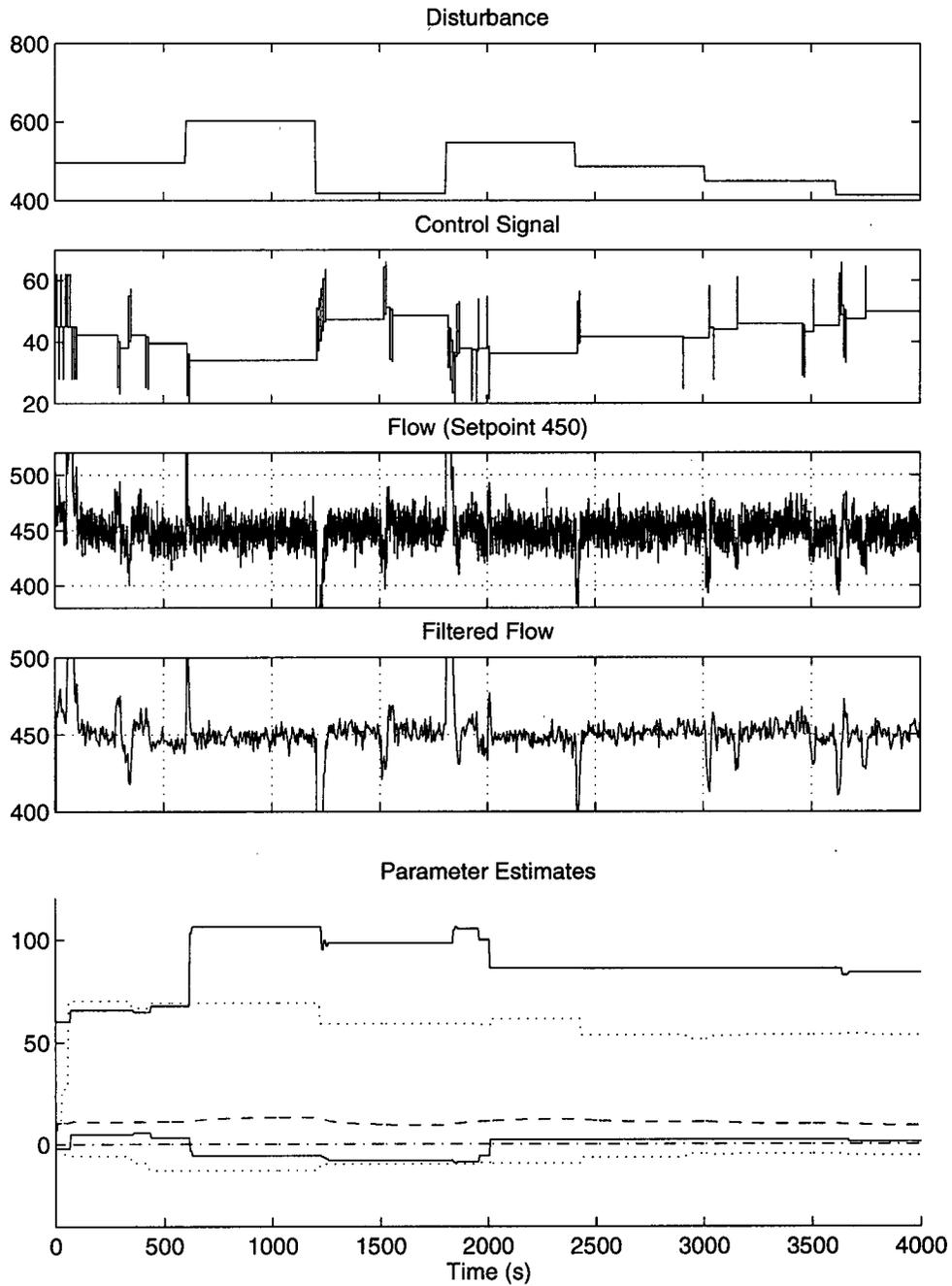


Figure 3.20: Test 1, long step disturbances (602s period). Results for adaptive control.  $IAE = 11.49$ ,  $IAE_{lim}(45) = 9.37$ . In the bottom graph, the dashed line is  $\hat{G}$ , the dot-dashed line is  $\hat{T}_{min}$ , the 2 dotted lines are  $\hat{b}_{first}$ , the 2 solid lines  $\hat{b}_{subsequent}$ . For each  $\hat{b}$  set, the upper line is  $\hat{b}_1$  and the lower  $\hat{b}_2$ .

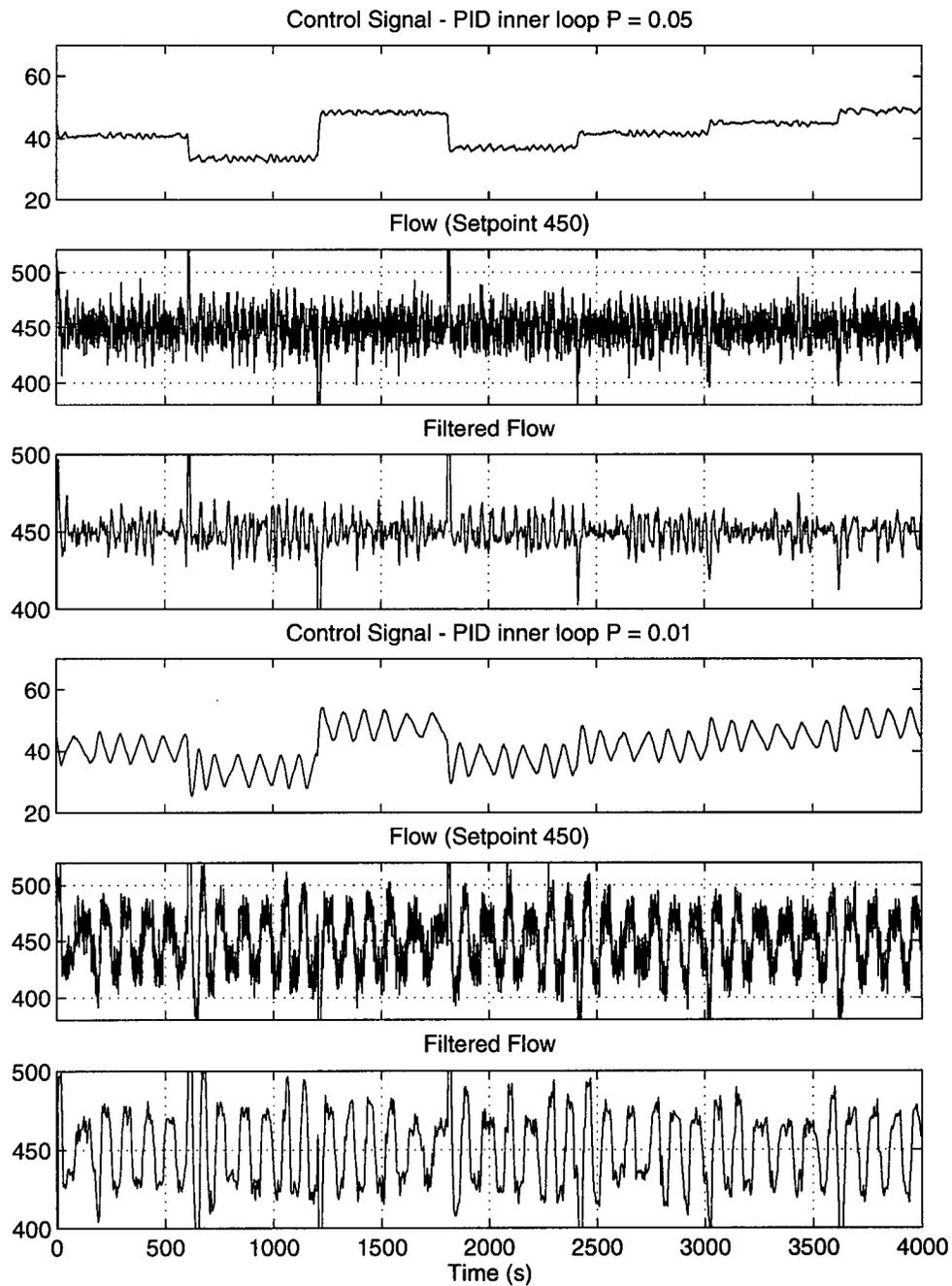


Figure 3.21: Test 1, long step disturbances (602s period). Results for PI control. Nominal positioner:  $IAE = 11.25$ ,  $IAE_{lim}(45) = 10.45$ . Misbehaving positioner:  $IAE = 22.32$ ,  $IAE_{lim}(45) = 20.70$ .

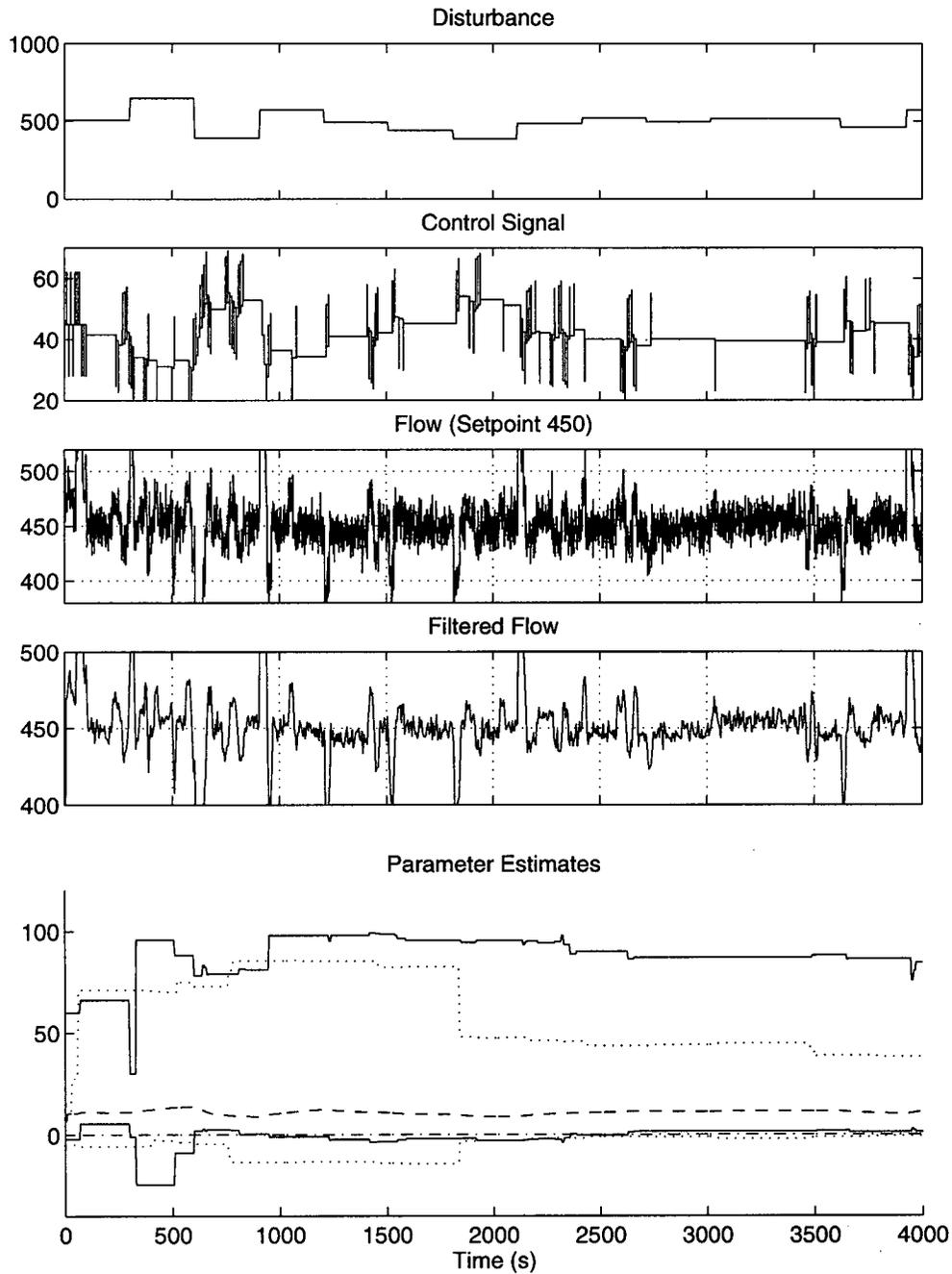


Figure 3.22: Test 2, short step disturbances (302s period). Results for adaptive control.  $IAE = 13.19$ ,  $IAE_{lim}(45) = 10.74$ . In the bottom graph, the dashed line is  $\hat{G}$ , the dot-dashed line is  $\hat{T}_{min}$ , the 2 dotted lines are  $\hat{b}_{first}$ , the 2 solid lines  $\hat{b}_{subsequent}$ . For each  $\hat{b}$  set, the upper line is  $\hat{b}_1$  and the lower  $\hat{b}_2$ .

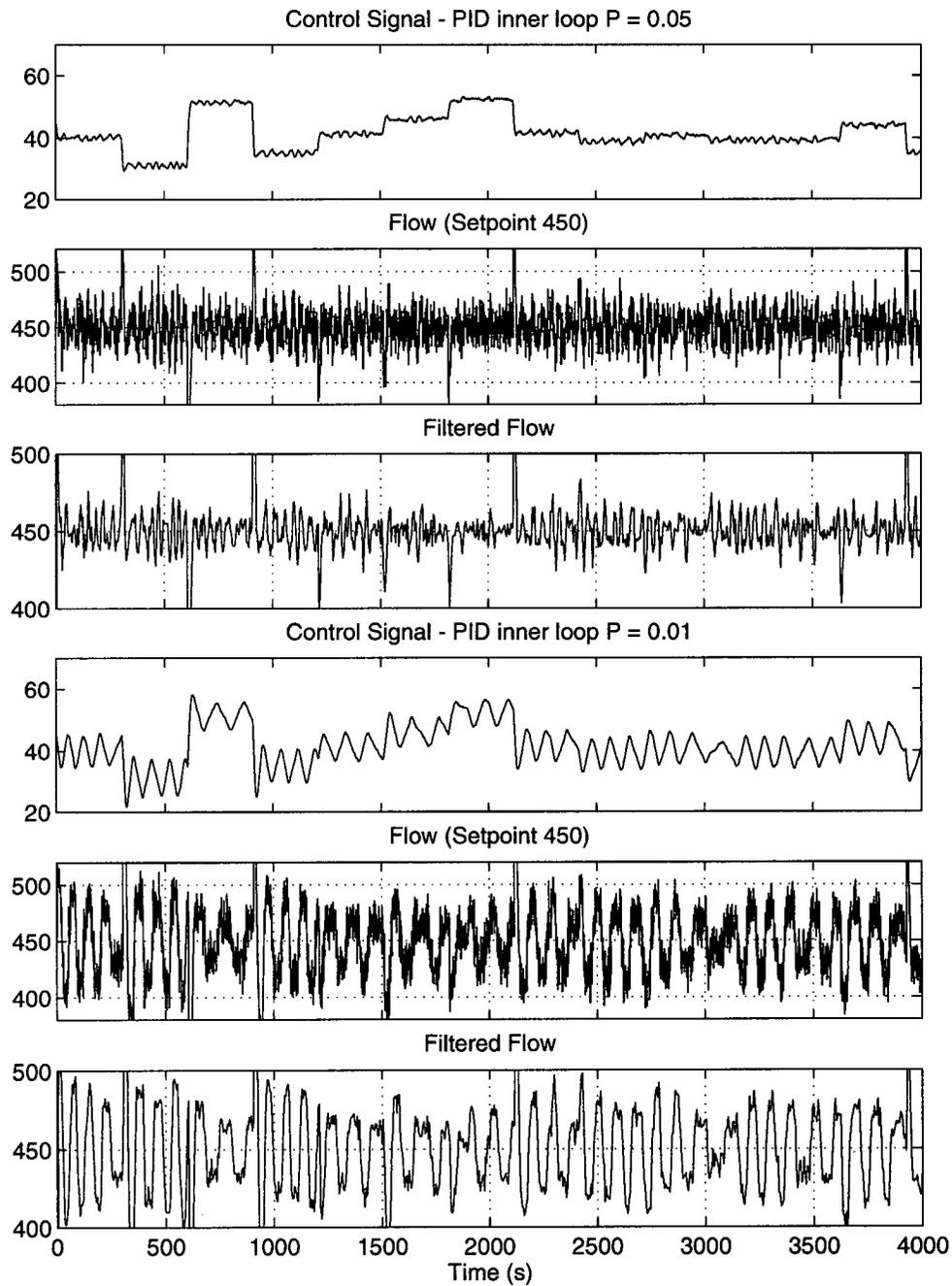


Figure 3.23: Test 2, short step disturbances (302s period). Results for PI control. Nominal positioner:  $IAE = 10.74$ ,  $IAE_{lim}(45) = 10.93$ . Misbehaving positioner:  $IAE = 23.18$ ,  $IAE_{lim}(45) = 20.61$ .

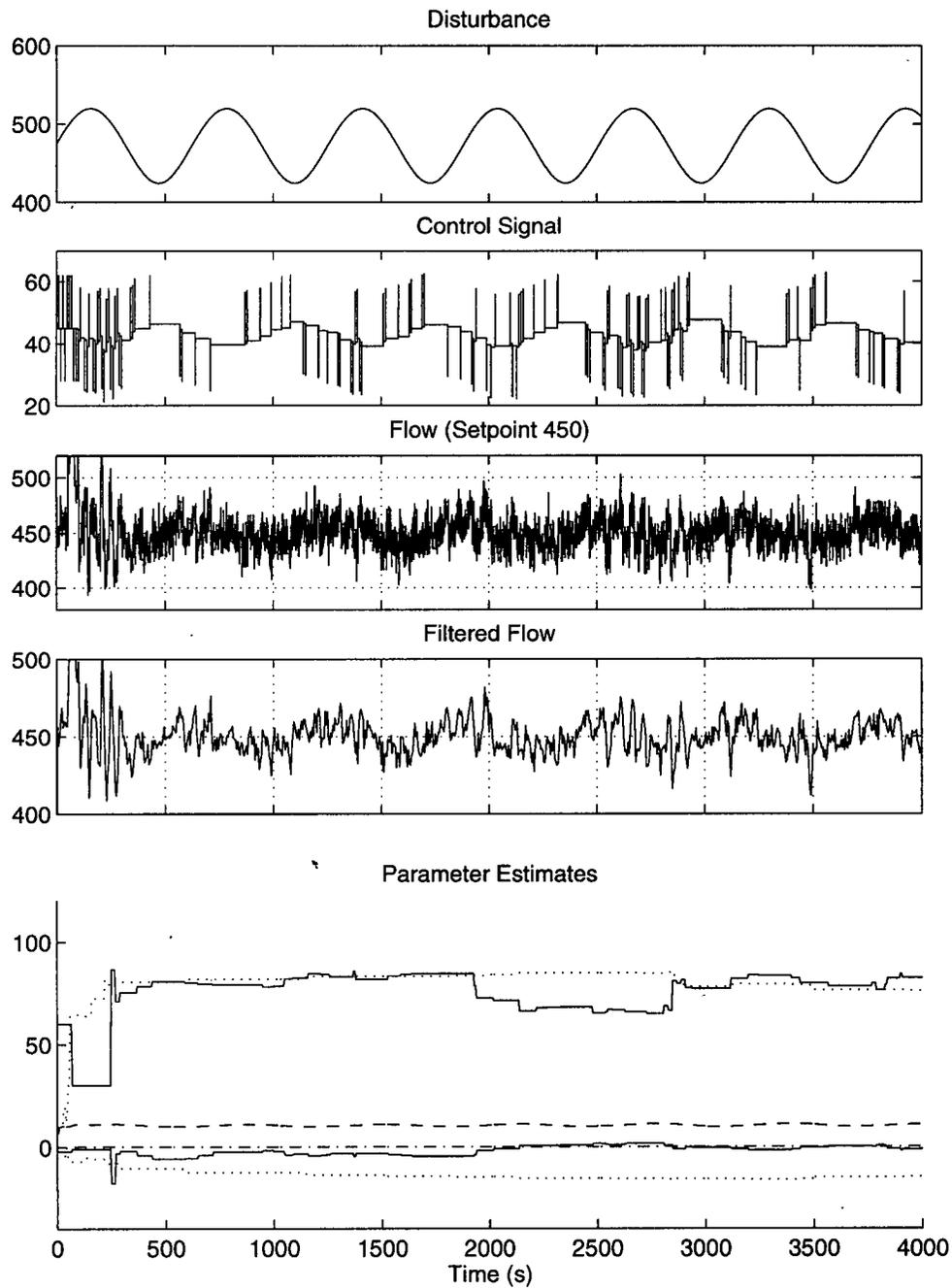


Figure 3.24: Test 3, sinusoidal disturbance. Results for adaptive control.  $IAE = 11.14$ ,  $IAE_{lim}(45) = 11.05$ . In the bottom graph, the dashed line is  $\hat{G}$ , the dot-dashed line is  $\hat{T}_{min}$ , the 2 dotted lines are  $\hat{b}_{first}$ , the 2 solid lines  $\hat{b}_{subsequent}$ . For each  $\hat{b}$  set, the upper line is  $\hat{b}_1$  and the lower  $\hat{b}_2$ .

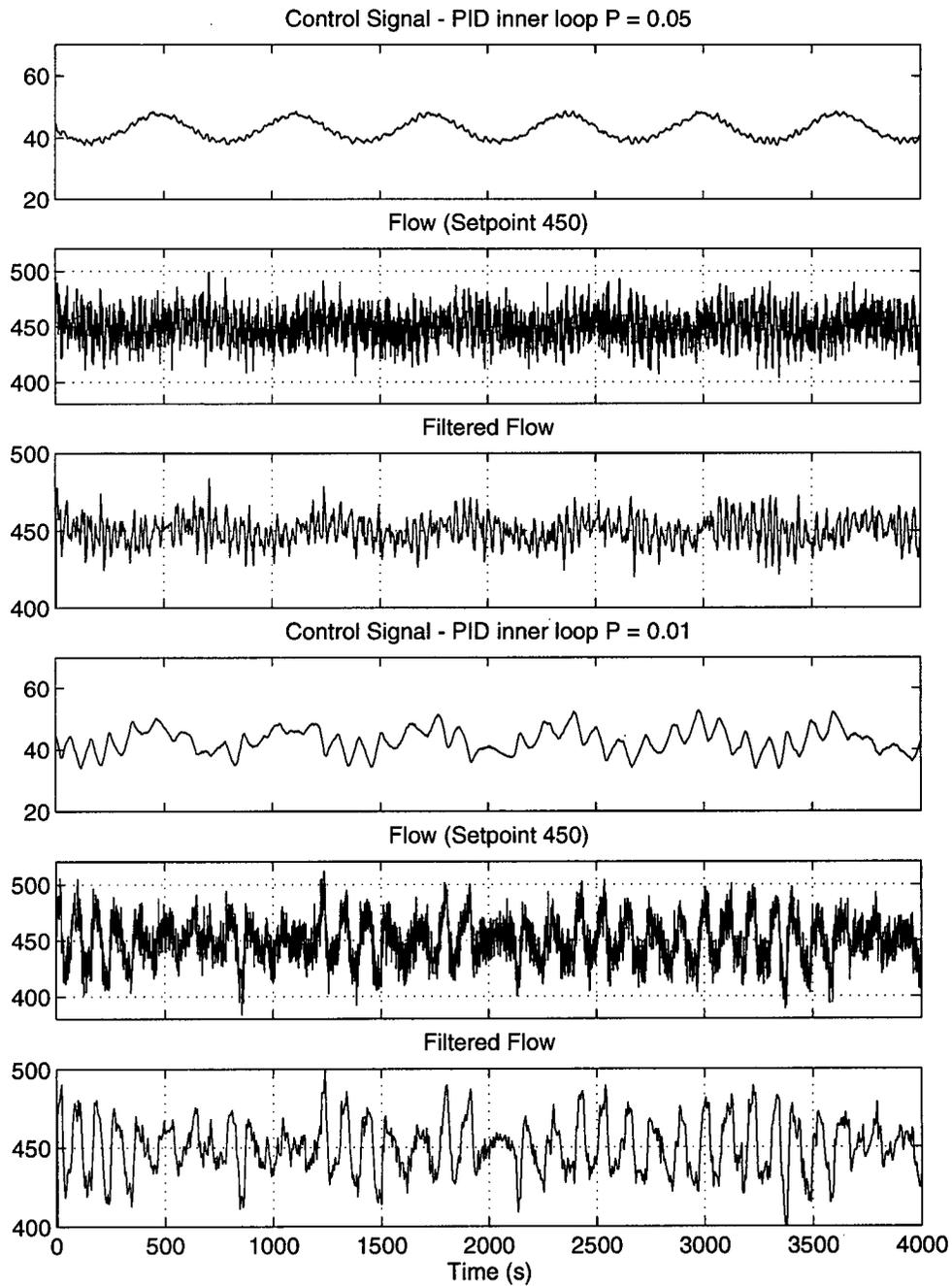


Figure 3.25: Test 3, sinusoidal disturbance. Results for PI control. Nominal positioner:  $IAE = 10.78$ ,  $IAE_{lim}(45) = 10.77$ . Misbehaving positioner:  $IAE = 15.72$ ,  $IAE_{lim}(45) = 15.26$ .

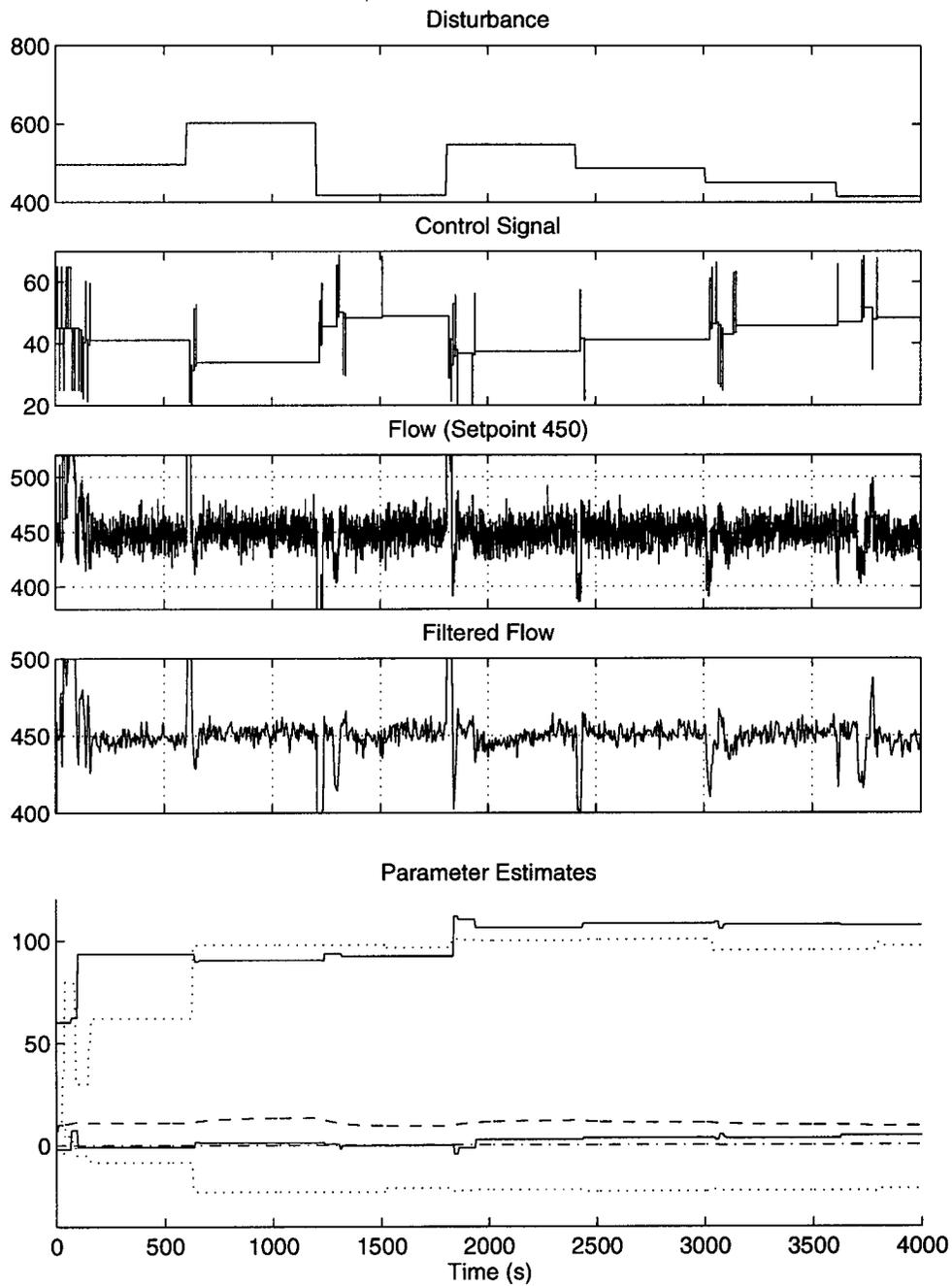


Figure 3.26: Test 4, long step disturbances (602s period), increased friction. Results for adaptive control.  $IAE = 11.56$ ,  $IAE_{lim}(45) = 9.42$ . In the bottom graph, the dashed line is  $\hat{G}$ , the dot-dashed line is  $\hat{T}_{min}$ , the 2 dotted lines are  $\hat{b}_{first}$ , the 2 solid lines  $\hat{b}_{subsequent}$ . For each  $\hat{b}$  set, the upper line is  $\hat{b}_1$  and the lower  $\hat{b}_2$ .

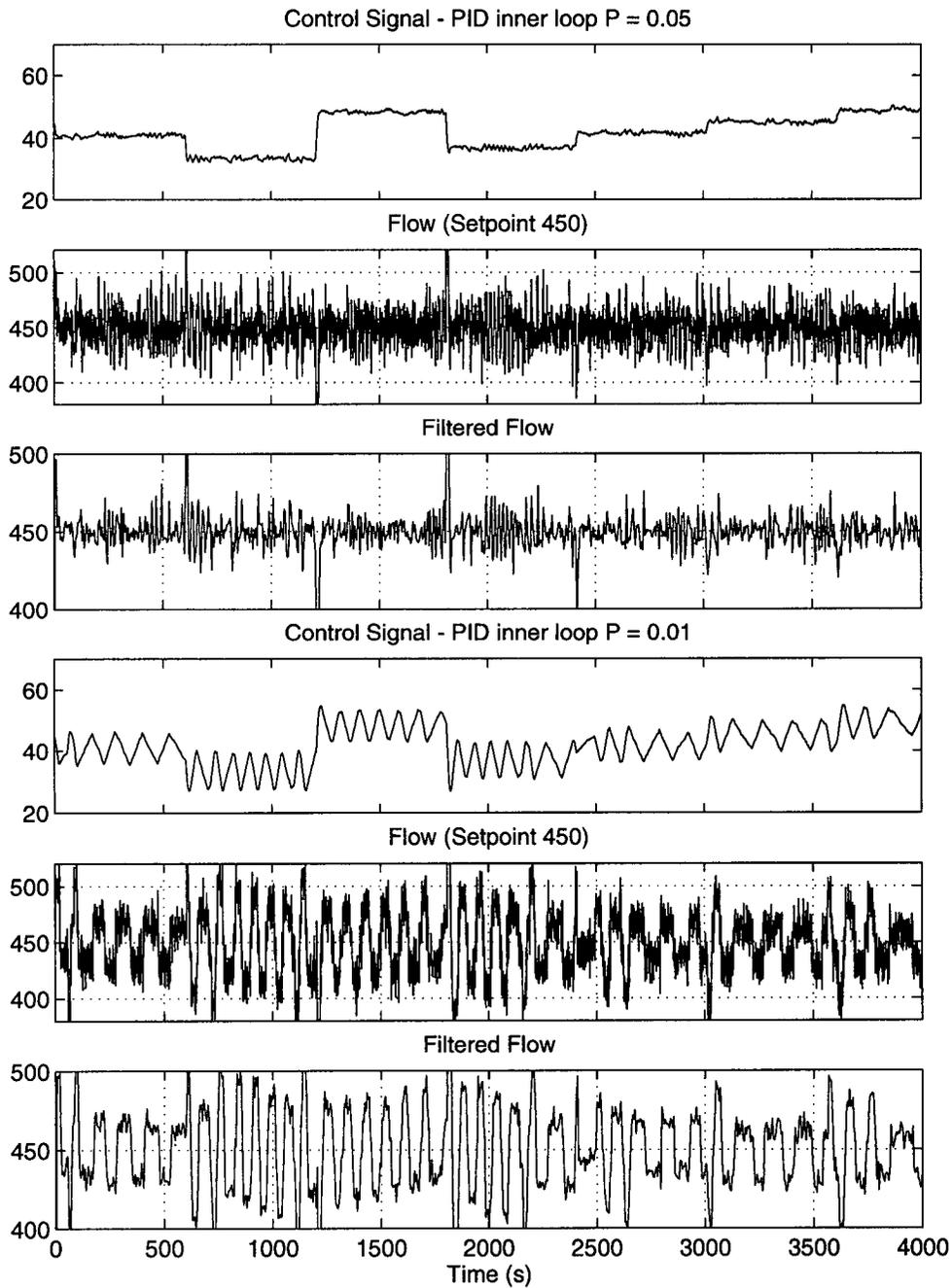


Figure 3.27: Test 4, long step disturbances (602s period), increased friction. Results for PI control. Nominal positioner:  $IAE = 11.80$ ,  $IAE_{lim}(45) = 10.88$ . Misbehaving positioner:  $IAE = 23.2$ ,  $IAE_{lim}(45) = 19.36$ .

## Chapter 4

# Conclusions

The model and the simulation blocks have been built as part of ongoing work to test and evaluate new strategies to improve the output in paper mills. It has been shown that a model can be built, and used to analyse and evaluate the performance in the mill. The basic model was quickly modified with different components, and sub component complexities. The same model was easily used to test different issues relating to the operation of the system. The work has shown that OMOLA and OmSim can be used for the desired purpose. Related work has been done by Persson and Tornhagen [34] and Bonnewijn [12] using OMOLA and OmSim to analyze the filtering and dynamic filtering capabilities of cleaners.

The simulations of friction-induced oscillations on the paper machine show that the greatest effect occurs for oscillations with periods above 100s. It is the combination of large friction problems and the need for slow PI tuning in delay dominated systems (to avoid instability) that combines to produce such large slow oscillations. Although the oscillations produced in the second part of the thesis didn't increase beyond 100s, a larger delay than that assumed, would require a slower outer loop controller and would quickly increase the length of these oscillations. This is especially true in the worst cases shown in Chapter 3 where, once the inner loop was unable to cope with the friction, the dynamics of the outer loop became the

dominant factor in determining the oscillation frequency.

Although the simulations of the paper machine show relatively small effects on headbox consistency for consistency loop valve oscillations, such disturbances are still unacceptable. The fact that the effects are present on the final output also highlights the fact that friction in loops with higher relative gain could cause serious problems in the final product.

The adaptive control algorithm has proved to work as well as a well-tuned PI controller (with fast inner loop dynamics), and significantly better on a valve with a misbehaving positioner (high inner loop deterioration). Its performance was only slightly worse, in the case of gradual disturbances, than the well-tuned PI controller with good positioner. The true benefit of the proposed controller is to reach a point close to the reference position, and maintain the valve in the specified position for an extended period of time. Due to the nature of friction in valves (causing quantum movements of the valve), any fine, gradual control is rendered impossible. Gradual disturbances then become increasingly difficult to handle, because any attempt at control immediately imposes large steps on the system, and a minimum error of half the step size. The well-tuned PI was able to perform better in this case as it made maximum use of the high gain inner loop, a feature that may not always be available.

The low inner loop gain that was needed for the PI controller to produce the types of oscillations shown in the literature, indicates that poor performance is perhaps not simply due to friction alone, but also to an undersized, underpowered, or deteriorated positioner. If this is the case, the obvious question to be asked is if the adaptive algorithm still works in this case. It is difficult to state conclusively, since no tests were done on a misbehaving valve. For this reason it is impossible to know exactly what parameter combination might be found in such a valve. The current adaptive design is however designed to work with a low inner loop gain. It does however also require that large control signals can be generated against

the valve, and that these control signals result in increased airflow. If this is not possible with the faulty positioner, then the friction control for the algorithm will likely not be effective. There still remains the possibility, however, of using the second relationship (slide distance to pulse height). Even with a limited range in the air flow rate, it should still be possible to generate different step lengths. The more complex relationship, as seen in Figure 3.13, would however require a modified estimation technique from the linear one that was originally suggested (3.37). The *Adaptive Nonlinear Modeler* proposed by Åström [5] might be appropriate, as it models a static nonlinearity with a piecewise linear relationship through  $n$  points (defining  $n$  parameters).

A useful extension to the control algorithm presented would be to switch to PID control when the position error deviates substantially (beyond  $3 e_{big}$ ), and switch back as the reference is approached. In cases of large disturbances or reference changes, this would produce better performance than the simple proportional control used currently.

The work in this thesis has shown that simulation can be used as an effective tool to identify problems in the paper mill, and also to design and test alternative control strategies. The adaptive controller has been shown to perform quite well in cases where conventional control induces large oscillations on the system, and as well as conventional control when disturbances are more active. Further testing of the controller is recommended. Tests on a real system would be ideal, but simple characteristic offline tests of valves exhibiting large friction could also be useful.

# Bibliography

- [1] Allison, B. J. and Isaksson, A. J.: 1996, Comparison of Mid-Ranging Controllers, *Control Systems '96*, Halifax, Nova Scotia, pp. 49–52.
- [2] Andersson, M.: 1994, *Object-Oriented Modeling and Simulation of Hybrid Systems*, PhD thesis, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [3] Armstrong-Hélouvry, B.: 1991, *Control of Machines with Friction*, Kluwer Academic Publishers, Norwell, Massachusetts.
- [4] Armstrong-Hélouvry, B., Dupont, P. and Canudas De Wit, C.: 1994, A Survey of Models, Analysis Tools and compensation methods for the control of machines with friction, *Automatica* **30**(7), 1083–1138.
- [5] Åström, K. J.: 1985, The adaptive nonlinear modeler, *Internal report*, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [6] Bai, E.-W.: 1997, Parametrization and Adaptive Compensation of Friction Forces, *The International Journal of Adaptive Control and Signal Processing* **11**(1), 21–31.
- [7] Ball, J.: 1998, Discussion on valve problems in a paper mill, *private communication*, Prince George Pulp and Paper, Prince George, BC.

- [8] Baril, C. G. and Gutman, P.-O.: 1997, Performance Enhancing Adaptive Friction Compensation for Uncertain Systems, *IEEE Transactions on Control Systems Technology* **5**(5), 466–79.
- [9] Bergstrom, J.: 1997, Non-linear friction dynamics in a valve under pi control, *Term report, elec 557*, Department of Electrical Engineering, University of British Columbia, 2356 Main Mall, Vancouver, BC, Canada.
- [10] Bialkowski, W.: 1992, Dreams vs. Reality: A View From Both Sides of the Gap, *Control Systems '92*, Whistler, BC, pp. 283–95.
- [11] Bialkowski, W. L.: 1994, Control valve dynamic specification, v. 2.1, *Internal report*, EnTech Control Engineering Inc., Toronto, Canada.
- [12] Bonnewijn, D.: 1997, *Study of Disturbance Propagation and Damping in a Paper Machine Approach System via OmSim Dynamic Modeling*, Master's thesis, Département d'Ingénierie Mathématique, Faculté de Sciences Appliquées, Université Catholique de Louvain and Pulp and Paper Centre Control Group, University of British Columbia.
- [13] Canudas de Wit, C. and Lischinsky, P.: 1997, Adaptive Friction Compensation with Partially Known Dynamic Friction Model, *The International Journal of Adaptive Control and Signal Processing* **11**(1), 65–80.
- [14] Canudas de Wit, C., Noël, P., Aubin, A. and Brogliato, B.: 1991, Adaptive Friction Compensation in Robot Manipulators: Low Velocities, *The International Journal of Robotics Research* **10**(3), 189–199.
- [15] Canudas de Wit, C., Olsson, H., Åström, K. J. and Lischinsky, P.: 1993, Dynamic Friction Models and Control Design, *Proceedings of the American Control Conference*, San Francisco, CA, pp. 1920–26.

- [16] Canudas de Wit, C., Olsson, H., Åström, K. J. and Lischinsky, P.: 1995, A New Model for Control of Systems with Friction, *IEEE Transactions on Automatic Control* **40**(3), 419–425.
- [17] Dupont, P. E.: 1994, Avoiding Stick-Slip Through PD Control, *IEEE Transactions on Automatic Control* **39**(5), 1094–97.
- [18] Eborn, J.: 1994, *Modelling and simulation of an industrial control loop with friction*, Master's thesis, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [19] Fitzgerald, B.: 1995, *Control Valves for the Chemical Process Industries*, McGraw Hill Inc., New York.
- [20] Fitzgerald, W. V.: 1993, Control Valve Performance and it's Effect on Process Control, *TAPPI/ISA PUPID Process Control Conference*, Nashville, TN, pp. 173–83.
- [21] Forsman, K.: 1998, Performance monitoring of sensors, controllers and actuators with applications to paper making, *Control Systems '98*, Porvoo, Finland, pp. 275–281.
- [22] Friedland, B. and Park, Y.-J.: 1992, On Adaptive Friction Compensation, *IEEE Transactions on Automatic Control* **37**(10), 1609–12.
- [23] Hägglund, T.: 1992, Disturbance supervision in feedback loops, *Internal report*, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [24] Hägglund, T.: 1995, The knocker - a compensator for stiction in control valves, *Internal report*, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.

- [25] Hojjat, Y. and Higuchi, T.: 1991, Application of Electromagnetic Impulsive Force to Precise Positioning, *Int. J. Japan Soc. Precision Engineering* **25**(1), 39–44.
- [26] Johnson, C. T. and Lorenz, R. D.: 1991, Experimental Identification of Friction and Its Compensation in Precise, Position Controlled Mechanisms, *Proceedings of the IEEE Industry Applications Society Annual Meeting*, Dearborn, Michigan, pp. 1400–5.
- [27] Lee, S.-B., Misawa, E. A. and Lucca, D. A.: 1996, Sliding Mode Compensation of Dry Friction, *IEEE International Conference on Control Applications*, Dearborn, Michigan, pp. 809–13.
- [28] Majd, V. J. and Simaan, M. A.: 1995, A continuous friction model for servo systems with stiction, *IEEE Conference on Control Applications*, Albany, NY, pp. 296–301.
- [29] Maqueira, B. and Masten, M. K.: 1993, Adaptive Friction Compensation for Line-Of-Sight Pointing and Stabilization, *Proceedings of the American Control Conference*, San Francisco, CA, pp. 1942–46.
- [30] Mattsson, S. E., Andersson, M. and Åström, K. J.: 1993, Object-oriented modeling and simulation, in D. A. Linkens (ed.), *CAD for Control Systems*, Marcel Dekker, Inc., pp. 31–69.
- [31] Nilsson, B.: 1993, *Object-Oriented Modeling of Chemical Processes*, PhD thesis, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [32] Olsson, H.: 1996, *Control Systems with Friction*, PhD thesis, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.

- [33] Olsson, H., Åström, K. J., Canudas de Wit, C., Gäfvert, M. and Lischinsky, P.: 1997, Friction models and friction compensation, *Internal report*, Department of Automatic Control, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
- [34] Persson, F. and Tornhagen, C.: 1998, *Object oriented simulation of a paper machine*, Master's thesis, Department of Automatic Control, Lund Institute of Technology and Pulp and Paper Centre, University of British Columbia, P.O. Box 118, S-221 00 Lund, Sweden.
- [35] Piipponen, J.: 1996, Controlling Processes With Nonideal Valves: Tuning of Loops and Selection of Valves, *Control Systems '96*, Halifax, Nova Scotia, pp. 179–86.
- [36] Popović, M., Gorinevsky, D. and Goldenberg, A.: 1995, A Study of Response to Short Torque Pulses and Fuzzy Control of Positioning for Devices with Stick-slip Friction, *IEEE Conference on Control Applications*, Albany, NY, pp. 302–7.
- [37] Schäfer, U. and Brandenburg, G.: 1993, Model Reference Position Control of an Elastic Two-Mass System with Compensation of Coulomb Friction, *Proceedings of the American Control Conference*, San Francisco, CA, pp. 1937–41.
- [38] Sell, N. J. (ed.): 1995, *Process Control Fundamentals for the Pulp & Paper Industry*, TAPPI Press, Atlanta, GA.
- [39] Shen, C. N. and Wang, H.: 1964, Nonlinear compensation of a second- and third-order system with dry friction, *IEEE Transactions on Applications and Industry* **83**(71), 128–36.
- [40] Slotine, J.-J. E. and Li, W.: 1991, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, New Jersey.

- [41] Southward, S. C., Radcliffe, C. J. and MacCluer, C. R.: 1991, Robust Non-linear Stick-Slip Friction Compensation, *ASME Journal of Dynamic Systems, Measurement, and Control* **113**(4), 639–45.
- [42] Tafazoli, S.: 1997, *Identification of Frictional Effects and Structural Dynamics for Improved Control of Hydraulic Manipulators*, PhD thesis, Department of Electrical Engineering, University of British Columbia, 2356 Main Mall, Vancouver, BC, Canada.
- [43] Tafazoli, S., de Silva, C. and Lawrence, P.: 1995, Friction Estimation in a Planar Electrohydraulic Manipulator, *Proceedings of the American Control Conference*, Seattle, WA, pp. 3294–98.
- [44] Tao, G. and Kokotović, P. V.: 1996, *Adaptive Control of Systems with Actuator and Sensor Nonlinearities*, John Wiley & Sons, Inc., New York.
- [45] Tung, E., Anwar, G. and Tomizuka, M.: 1991, Low velocity friction compensation and feedforward solution based on repetitive control, *Proceedings of the American Control Conference*, Boston, MA, pp. 1076–81.
- [46] Vedagarbha, P., Dawson, D. and Feemster, M.: 1997, Tracking Control of Mechanical Systems in the Presence of Nonlinear Dynamic Friction Effects, *Proceedings of the American Control Conference*, Albuquerque, NM, pp. 2284–88.
- [47] Yang, S. and Tomizuka, M.: 1988, Adaptive Pulse Width Control for Precise Positioning Under the Influence of Stiction and Coulomb Friction, *ASME Journal of Dynamic Systems, Measurement, and Control* **110**(3), 221–27.
- [48] Yap, E.: 1998, *Dynamic modelling of a paper machine wet end*, Master's thesis, University of British Columbia.