MODEL PREDICTIVE CONTROL DESIGN FOR LINEAR MULTIVARIABLE SYSTEMS UNDER CONTROL VALVE STICTION

by

SEAN SURAJ JEREMIAH

14286

A Dissertation
Submitted to the Department of Chemical Engineering
in Partial Fulfilment of the Requirements for the
Bachelor of Engineering (Hons.) Chemical Engineering

Department of Chemical Engineering,
Universiti Teknologi PETRONAS.

December 2014

CERTIFICATION OF APPROVAL

MODEL PREDICTIVE CONTROL DESIGN FOR LINEAR MULTIVARIABLE SYSTEMS UNDER CONTROL VALVE STICTION

by

SEAN SURAJ JEREMIAH

14286

A Dissertation
Submitted to the Department of Chemical Engineering
in Partial Fulfilment of the Requirements for the
Bachelor of Engineering (Hons.) Chemical Engineering

Approved By,

…………………………………………………………….
(DR. MARAPPA GOUNDER RAMASAMY)
on behalf of DR. HASLINDA ZABIRI

Department of Chemical Engineering,
Universiti Teknologi PETRONAS.

December 2014

DECLARATION OF ORIGINALITY

This is to declare that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein has not been undertaken or done by unspecified sources or persons.

…………………………………….
(SEAN SURAJ JEREMIAH)
Date: 25 December 2014

ABSTRACT

Product quality and production costs are dependent on optimal control of the process. Product variability and oscillation of the process would indicate poor control. Such is the consequence of nonlinearity in the response of the control valves towards controller instructions. The most common cause of nonlinearities is static friction (stiction) in the mechanical assembly of the valve. The ability of a hybrid model predictive control (MPC) formulation to compensate for stiction was to be tested. The formulation was previously shown to be able to compensate for backlash by solving a mixed integer quadratic programming (MIQP) problem.

Simulation studies were conducted using a model of a paper machine headbox model in Simulink. The hybrid MPC formulation was updated to run on current software versions and the Choudhury stiction model was integrated into the system. Due to errors in the simulation, the ability of hybrid MPC to compensate for stiction finally could not be determined. The errors encountered are documented as well as recommendations to overcome the shortcomings of the simulation.

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to the following people

- Dr. Haslinda Zabiri, my patient supervisor whose guidance and insights proved to be invaluable and motivating
- Bashariah Kamaruddin, my cheerful colleague who was ever-ready to assist
- Dr. Marappa Gounder Ramasamy, who recommended that this project be conducted
- My housemates and coursemates who were around to cheer me on
- My dear parents for their consistent support and loving kindness.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This section provides an overview of the area of study as well as the research gap addressed and the objectives of the study.

1.1 BACKGROUND

1.1.1 Model Predictive Control

Model Predictive Control (MPC) is a process control method that takes into account the future state of the process, over a finite time horizon, to compute an optimal control strategy.

The future controller input (i.e. independent variables) and state of the process (i.e. the dependent variables) are modelled using an explicit dynamic model which is fed with the current measurements from the process and process variable targets. The term explicit dynamic model implies that model represents the process over a period of time and can be used to calculate the future state of the process at the current time.

At each time interval, the controller computes a number of control steps across the prediction horizon that would shift the process towards the control target but only implements the first step. In contrast, a proportional-integral-derivative (PID) controller is not aware of the effect of subsequent control steps on the process as the response of the PID controller is based on present and past errors (i.e. deviation from the control target).

1.1.2 Hybrid MPC

In order to calculate optimal control moves, MPC controllers conventionally utilise a plant model which does not account for process nonlinearities. This is a cause for sub-optimal control performance. A hybrid MPC design framework developed by Zabiri and Samyudia is able to compensate for control valve backlash thus improving control performance.

1.2 PROBLEM STATEMENT

Control valve nonlinearities such as backlash and static friction (stiction) will result in control performance degradation and process variability due to oscillation of the control valve. This can have significant impact on product quality and on energy consumption. Oscillation also inherently induces wear of the internal components of the valve thus decreasing the lifespan of the valves and increasing maintenance costs.

The possibility exists for the hybrid MPC design developed by Zabiri and Samyudia which is able to compensate for backlash to be further optimized and used to compensate for stiction.

Stiction in part exhibits characteristics similar to backlash. Backlash results when the controller output changes but the valves position does not change. When stiction occurs, there is initially a change in controller output and no change in control valve position before an abrupt change in the control valve position. It is hoped that by compensating for this so-called, deadband, the zone where the control valve position does not change, stiction can be eliminated.

1.3 SCOPE AND OBJECTIVES OF STUDY

This ultimate goal of this study is to extend a pre-existent MPC design to not only compensate for backlash but for stiction as well.

Therefore, in brief, the objectives of this study are as follows:

- Update hybrid MPC solver instructions to enable it to run on current numerical simulation software
- Simulate control valve stiction and integrate it into a plant model
- Investigate the ability of hybrid MPC to compensate for stiction

No modifications will be made to the core algorithms used to compute optimal control moves in the hybrid MPC framework as the objective of the study is the determine if the current formulation is able to compensate for stiction.

1.3.1 Update hybrid MPC solver instructions to enable it to run on current numerical simulation software.

Certain function calls and referencing methods in the original hybrid MPC program have been phased out over the years and are now obsolete. In order to use the formulation, it is essential that the program be made to run with current software versions without altering the core formulations.

1.3.2 Simulate control valve stiction and integrate it into a plant model

Several methods of defining control valve stiction are in existence. A suitable method of simulating stiction accurately while preserving compatibility with the MPC program must be found. Once the stiction model is complete, it has to be integrated into a plant model to simulate effect stiction on the process.

1.3.3 Investigate the ability of hybrid MPC to compensate for stiction

The ability of the hybrid MPC to compensate for stiction can be investigated once the plant model is functioning as expected. It is hoped that by compensating for the backlash component of stiction, the amount of stiction present can be reduced.

CHAPTER 2

LITERATURE REVIEW

2.1 DEVELOPMENT OF MODEL PREDICTIVE CONTROL

The advent of model predictive control was in the mid-70s when more powerful processors made computer based control a reality [2]. A landmark paper by Richalet and co-workers in 1978 reported numerous successful MPC implementations in large-scale industries. [3] The technology continued to mature matured in the 1990s as computers became cheaper and processors became more powerful. Today, it is part of a broad range of technologies that fall under the umbrella of Advanced Process Control (APC).

The precursor to MPC was the linear quadratic Gaussian (LQG) controller [4] which employed an infinite prediction horizon. This allowed the optimization to be solved explicitly while guaranteeing closed loop stability. Model Predictive Heuristic Control (MPHC), Dynamic Matrix Control (DMC) and Generalised Predictive Control (GPC) appeared about the same time around the mid-70s to the mid-80s. MPHC employed a finite impulse response (FIR) model and a reference trajectory along with coincidence points whereas DMC employed a truncated step response (TSR) model and least squares minimization of error with respect to a constant setpoint. GPC on the other hand, employed a transfer function model and made use of stochastic effects.

In the late-80s, MPC practitioners began to shift to a state space model and quadratic programming was used to solve the open-loop optimization problem. The 90s saw MPC gaining a reputation as one of the best optimal control techniques. In the late-90s, Bemoporad and Morari introduced Hybrid MPC for systems with continuous dynamics and logical rules. In the more recent years, improvements have been made to computation techniques leading to fast optimization. This allowed MPC technology to gain traction outside the process industry and find its way into the regulation of mechanical and electronic systems.

2.2 OPERATING PRINCIPLES OF MPC

MPC takes into account the future state of the process, over a finite time horizon, to compute an optimal control strategy. The future controller input (i.e. independent variables) and state of the process (i.e. the dependent variables) are modelled using an explicit dynamic model which is fed with the current measurements from the process and process variable targets. [5, 6] The term explicit dynamic model implies that model represents the process over a period of time and can be used to calculate the future state of the process at the current time.

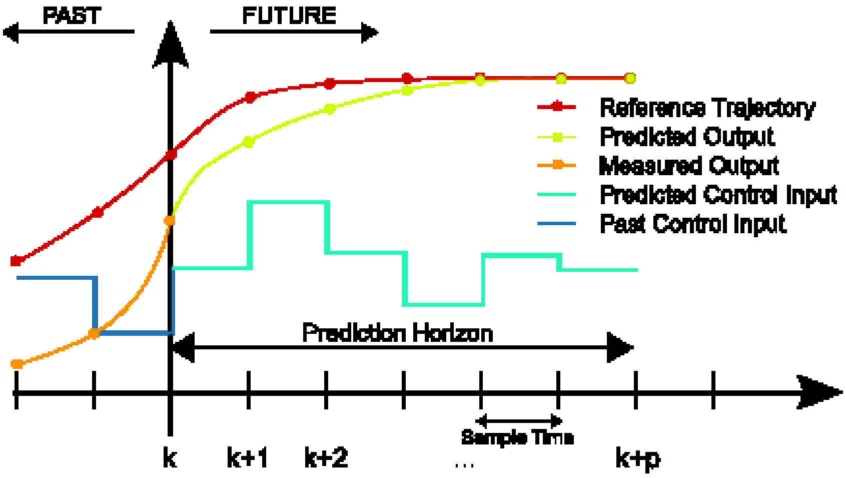

Figure 1 : MPC input – output graph adapted from CCForum.com

At each time interval, the controller computes a number of control steps across the prediction horizon that would shift the process towards the control target but only implements the first step.

In contrast, a proportional-integral-derivative (PID) controller is not aware of the effect of subsequent control steps on the process as the response of the PID controller is based on present and past errors (i.e. deviation from the control target). Also, if a change in the operating conditions of the process alter the dynamics of the system, a PID controller would have to be retuned to achieve optimal performance. An MPC controller however would be already aware of the changes as the dynamics would already be incorporated into the model. [7]

That said, feedback control often complements MPC. It is used to counter model inaccuracies. However, this practice is not favourable as feedback control action may come in too late. [8]

Further advantages are possessed by MPC. The first is that disturbances can be anticipated and compensated for allowing the process to be driven closer to the optimal operating conditions. Secondly, with the availability of the process model, the controller has the capability to deal with process dynamics. Also, because the controller is now able to predict future control steps, it can take the necessary measures to minimise the likelihood of constraint violations that exist to due to equipment characteristics and safety concerns. [9]

Although a seemingly powerful technology, the inherent flaw of MPC is that it is inefficient in handling model uncertainties thus making its robustness questionable. [10] This takes effect when the mathematical model programmed into the controller is unable to reproduce the process behaviour accurately. Among the factors that contribute to model uncertainties are parametric uncertainties, nonlinearities and dynamics. [11] In the first case, certain parameters that affect the process may be unknown or not modelled. In the case of nonlinearities, certain aspects of the process that are actually nonlinear may have been assumed to be linear in the model. And in the case of dynamics, the dynamics of the process may not have been properly understood and modelled. All the contributing factors mentioned above may arise from the actuator, the sensor or the plant and will have an impact on the performance of the controller.

2.3 ACTUATOR NONLINEARITIES

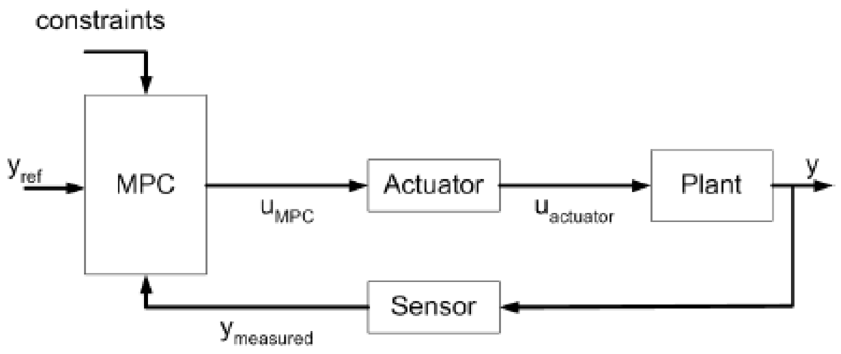

Figure 2: A closed control loop adapted from Zabiri & Samyudia [11]

In most cases, the actuator is a control valve. It is the final control element by which a process is brought under control. Control performance degradation can result from control valves not being able to execute control measures as directed by the controller. Such performance degradation can be said to be a result of nonlinearity in the response of the control valve. Desborough and Miller found that 20 – 30 % of all control loops perform poorly due to control valve problems. [12] Poor control performance will result in oscillations which lead to poor product quality and energy wastage. Furthermore, oscillations from one control loop may propagate to different plant units. [13] The most common cause for control valve nonlinearity is static friction (stiction). [1] A valve is said to be experiencing stiction when the valve cannot overcome the static friction in the mechanical assembly of the valve.

Since a process model and not merely the process state is used as a basis for control, MPC can be used to compensate for nonlinearities that result from control valves. Taking into account process and valve nonlinearities would allow better control performance for processes that undergo frequent changes of setpoints i.e. servo problems. Both linear and nonlinear MPC exist as do linear and nonlinear process models and are distinct from one another. The advantages of using nonlinear process models and thus nonlinear MPC for processes that require maintaining the setpoint, i.e. regulator problems, are lesser than for servo problems which involve changing in the setpoint. [8] Furthermore, the additional computational complexity of non-linear

MPC models, which produce better representations of the process than linear models do, limits the widespread use of non-linear MPC. [2]

Bemporad and Morari [14] proposed a framework for modelling mixed logical dynamical systems which brought together logical rules and continuous dynamics to beget mixed integer quadratic programming (MIQP). Fast, on-line optimization of the MIQP problem was a challenge that led to the development of table lookup based methods. Jones and Morari introduced a method to compute approximate explicit control laws that is less complex but has a tolerable error. [2, 15]

2.4 CONTROL VALVE STICTION SIMULATION

According to Choudhury, Thornhill and Shah [16], the most common cause for control valve nonlinearity is stiction. A valve is said to be experiencing stiction when the valve cannot overcome the static friction in the mechanical assembly of the valve. Even though the controller output changes, the valve position will not change until a certain pressure is applied to the actuator. [17] This friction is a result of the tight packing surrounds the valve stem to prevent process fluid losses. Besides this, friction is also caused by the corrosion of the valve stem and deposits on the valve seat which cause the valve plug to stick. The best solution to overcome stiction is removing the valve for maintenance but this is not always possible. [18]

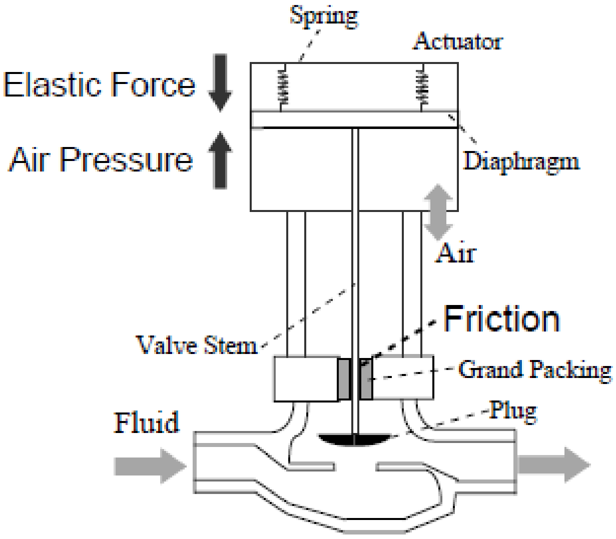

Figure 3 : Cross-sectional view of a pneumatic control valve. [19]

A formal definition provided by Choudhury, Thornhill and Shah [1] describes stiction to have four components: deadband, stickband, slip-jump and moving phase.

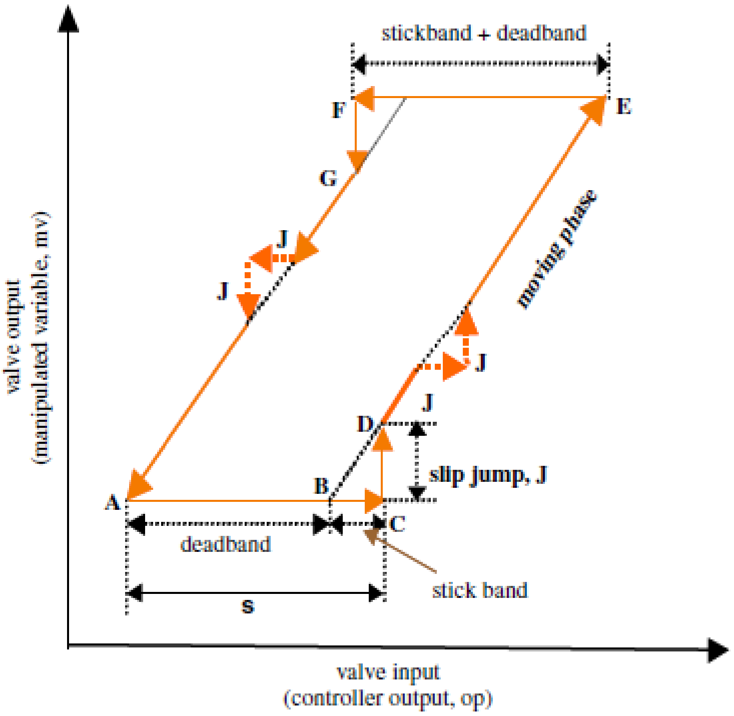

Figure 4 : Typical input-output behaviour of a sticky valve adapted from Choudhury et al [1]

Deadband and stickband occur when the valve does not move although the input of the valve changes. Deadband may be caused by backlash which is a result of loose mechanical parts. Stickband occurs due to static friction. Slip-jump occurs when there is a sudden release of the potential energy accumulated due to the static friction.

The figure above illustrates stiction. S represents the amplitude of the input signal when the valve is sticking and J represents the amplitude of the slip-jump. When a valve comes to rest at Position A, it sticks. The controller output causes the valve to overcome the deadband (AB) and subsequently the stickband (BC). The valve jumps to Position D then continues to move. Although uncommon, the valve may stick again in between Positions D and E due to low or zero velocity. This pattern may continue as the valve returns back to its original position.

Stiction is measured as a percentage of the valve travel or span of the control signal. [20] For example, if the control signal spans from 4 to 20 mA, one percent stiction would imply that the valve would only start to move if the control signal is greater than or equal to 4.16 mA.

2.4.1 Stiction Models

Both physics based and data driven (empirical) models exist for stiction. Physical model stem from Newton's Second Law, which states that the net force acting on an object is the product of its mass and acceleration. Friction can be modelled as a static function of velocity or by using time-varying parameters.

Using the classical model, Newton's Second Law rewritten for a sliding stem valve is,

$$M\frac{d^2x}{dt^2} = \Sigma \text{ Forces} = F_a + F_r + F_f + F_p + F_i \qquad \text{[Eqn. 1]}$$

Where, F_a is the force applied by the pneumatic actuator, F_r is the spring force, F_f is the friction force, F_p is the force due to the fluid pressure drop and F_i is the additional force required to force the valve into the seat. For the stiction model, F_p and F_i can be assumed to be zero.

As studied by Olsson and colleagues, [21] the classical friction model can be used to model deadband and stick-slip effects.

$$F_f = \begin{cases} -F_c sgn(v) - vF_v & if\ v \neq 0 \\ -(F_a + F_r) & if\ v = 0\ and\ |F_a + F_r| \leq F_s \\ -F_s sgn(F_a + F_r) & if\ v = 0\ and\ |F_a + F_r| > F_s \end{cases} \qquad \text{[Eqn. 2]}$$

The first line represents the moving friction. The second line is the case when the valve is stuck and the third line represents the friction at the instant of breakaway.

Karnopp [22] devised an improved static classical model to overcome the limitations of the classical model at null speed and to avoid switching between model equations.

Armstrong-Helouvery et al. [23] proposed a devised a dynamic model in which allow the static friction coefficient to vary at the time the body is stuck. This dynamic model known as the Seven Parameter model was used inside the Karnopp model. It was subsequently upgraded to model friction more accurately at low velocities and during velocity reversal. [21]

In circumstances where the parameters needed to model friction accurately cannot be found, empirical models come into play. Such models are data-driven and therefore do not require parameters such as static, coulomb and viscous friction to be estimated and valve mass, spring constant, etc. to be known. Although requiring lesser parameters to predict stiction, results are not always consistent. [19]

The goal of data-driven stiction models is to simulate the jump i.e. the conversion of potential energy to kinetic energy, which occurs when stiction is overcome.

Stenman et al. introduced a one parameter model to simulate the stickband. [24] Choudhury and co-workers described an algorithm for a two-parameter data driven model which consists of the size of the deadband plus stickband (specified in the input axis) and slip-jump (specified in the output axis). [1] The illustration of this model is given above in the definition of stiction. Kano and colleagues [25] as well as He and colleagues [26] have proposed improvements to the Choudhury model which allow the model to deal with both deterministic and stochastic signals.

2.4.2 The Chowdhury Model

The academic community has commended the Choudhry model for its ability to reproduce stiction accurately and without complexity. The model is even capable of replicating the results of physical models. [27] Much of the definition of stiction provided earlier in this section originates from the Chowdury model.

Stiction is said to occur in four conditions which are in reality not mutually exclusive.

- Pure Deadband (J=0)
- Undershoot (J < S)
  The valve output can never reach the valve output since there is always some offset present.
- Pure Slipstick (J = S)

Once stiction is overcome, the output matches the input accordingly.

- Overshoot (J > S)

  The output exceeds the setpoint.

2.5 CONTROL VALVE STICTION COMPENSATION

The pre-existing non-MPC based stiction compensation methods which have been studied have their limitations. The first is dithering and impulsive control proposed by Armstrong-H`elouvry and colleagues. [23] The control valve input is to be applied in a series of pulses instead of a steady signal. The problem with this technique is that 90% of control valves are pneumatically operated and therefore filter high frequency dither. The effectiveness of the input-output linearization technique developed by Kayihan and Doyle [28] is limited since certain valve properties and parameters such as stem mass, stem velocity, etc. may not be known. Detuning the integral effect in the controller as proposed by Gerry and Ruel [20] can cause steady state offset. The knocker signal approach proposed by Hägglund [29] requires additional parameters that characterize the knocker signal to be tuned. This method and the two-move method proposed by Srinivasan and Rengaswamy [30] result in aggressive movement of the valve stem which may wear the valve quickly.

There is currently no literature on MPC based stiction compensation methods. Zabiri and colleagues suggest that there is an absence of effective stiction quantification methods although detection methods do exist. [31] Daneshwar and Noh recommend that research be directed towards stiction compensation as much study on stiction detection has been done. [13]

2.6 MIQP MPC

With the introduction of binary variables to classical MPC quadratic programming formulation, an MPC design by Zabiri and Samyudia [11] utilises the so called mixed integer quadratic programming (MIQP) approach to translate an approximation of a nonlinear inverse backlash strategy to a set of linear inequalities as the system approaches steady-state. The MPC framework involves a quadratic objective function and mixed linear inequalities in the form of binary variables denoting the presence or absence of backlash. Using this approach, the nonlinearity itself is modelled then the inverse of the nonlinearity is applied in series with the controller. This approach has been shown to be able to effectively eliminate backlash by supressing input movement to avoid the backlash zone without needing to retune the controller. The activation time and suppression time are also determined automatically. However, by design this approach is only effective if the inputs have not been saturated.

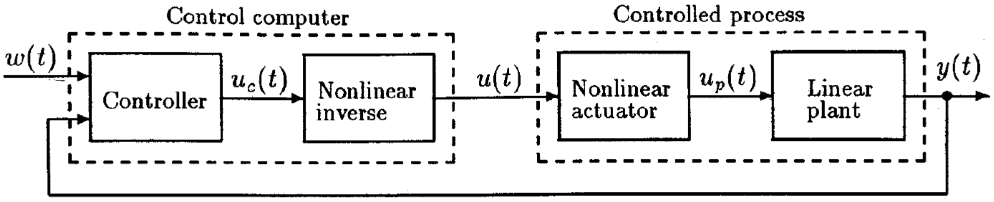

Figure 5: Inverse backlash strategy implemented by Zabiri and Samyudia [9]

CHAPTER 3

METHODOLOGY

Simulation and analysis is conducted using the Simulink component of MATLAB. In order to solve the mixed quadratic programming (MIQP) problem, the General Algebraic Modelling System (GAMS) which is a high-level modelling system for mathematical programming and optimization is used.

In the first phase of this project, the MPC design and interface between MATLAB and GAMS was upgraded to enable the program to run with the latest simulation software versions. This was followed by search for the most suitable stiction model. The model was then integrated into the existing MPC design. Simulation studies were then conducted using this new model.

3.1 PAPER MACHINE HEADBOX MODEL

The above mentioned process control model is used in this project is based on the control study of the pulp fibres percentage in aqueous suspension, N_2 and the liquid level, H_2 in a paper machine headbox by Ying et. al. [32] The primary control objectives here are to hold N_2 and H_2 at their respective setpoints.

The schematic below illustrates the process.

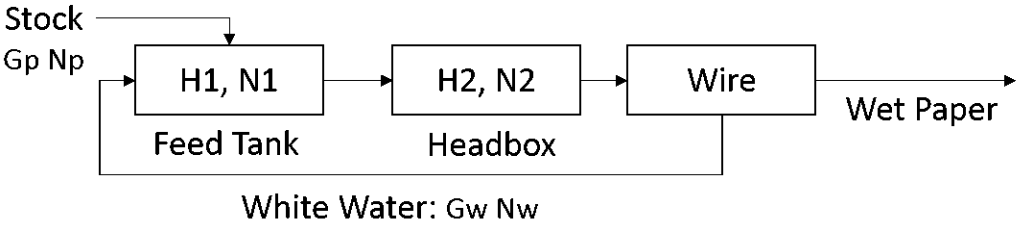

Figure 6: Schematic of the Paper Machine Headbox Elements. [32]

Manipulated Variables

G_p	flow rate of stock entering the feed tank
G_w	recycled white water flow rate

Measured Output

H_2	liquid level in the headbox
N_1	feed tank consistency
N_2	headbox consistency

Measured Disturbance

N_p	consistency of stock entering the feed tank

Unmeasured Disturbance

N_w	consistency of white water

The process above can be linearized analytically to form the following state-space matrices.

$$A = \begin{bmatrix} -1.930 & 0 & 0 & 0 \\ 0.394 & -0.426 & 0 & 0 \\ 0 & 0 & -0.630 & 0 \\ 0.820 & -0.784 & 0.413 & -0.426 \end{bmatrix}$$

$$B = \begin{bmatrix} 1.274 & 1.274 & 0 \\ 0 & 0 & 0 \\ 1.340 & -0.650 & 0.203 \\ 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \text{[Eqns. 3]}$$

3.2 MODEL PREDICTIVE CONTROLLER CONFIGURATION

The MATLAB function for a constrained MPC, CMPC is used compute the controller output. The function makes use of the discretized linear model which has been saved in a MPC mod format. The configuration parameters specified below are consistent with the recommended parameters supplied by the publishers. [33]

The prediction horizon is set to 20 period and the input horizon is set to 5 moves.

An equal weightage of 2 is set for both manipulated variables, G_p and G_w. The output variables H_2 and N_2 are assigned weights of 1 and 5 respectively whereas no weightage is assigned to N_1.

The constraints on the manipulated variables, G_p and G_w are set at a minimum of -10 and a maximum of 10. The maximum up and down rate are configured as 2 and -2 respectively. No constraints are imposed on the output variables.

All initial values for the input, output and disturbance are set to 0 while the setpoint i.e. the reference trajectory for H_2, N_1 and N_2 is set to 1, 0 and 0 respectively.

3.3 STICTION MODEL

The model used in this study is the Chowdhury model as it has been shown to reproduce stiction behaviour accurately and without complexity. The model is based on a series of “if-else” statements as illustrated in the flowchart below.

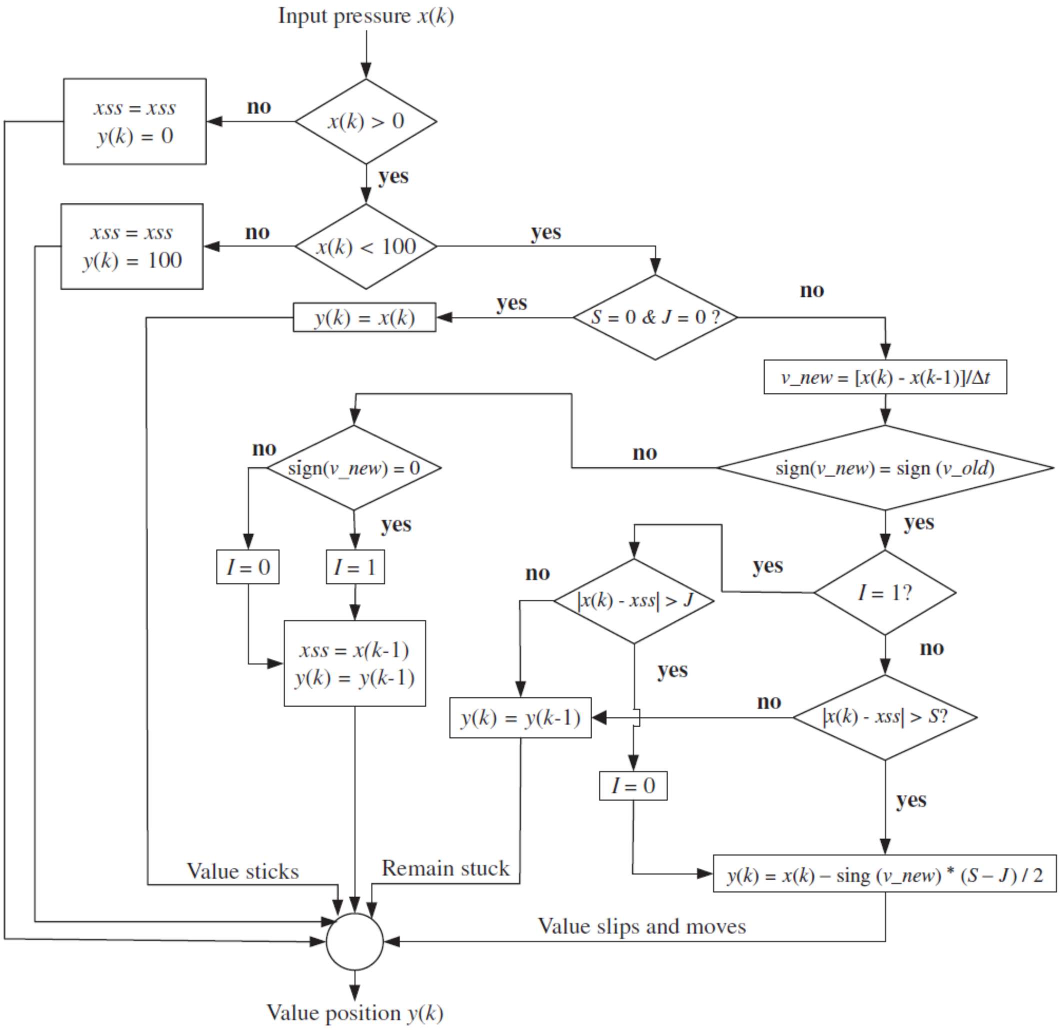

Figure 7: Flowchart for the Chowdhury model algorithm [17]

3.4 SIMULINK CONFIGURATION

The block diagram interface used in the simulation study done by Zabiri and Samyudia is shown below. The Paper Machine Model accepts four inputs and returns three outputs. The three outputs in addition to the measured disturbance is fed to the MPC controller. Subsequently, the controller computes the optimal control moves and that information is fed the actuator which manipulates the two controlled variables.

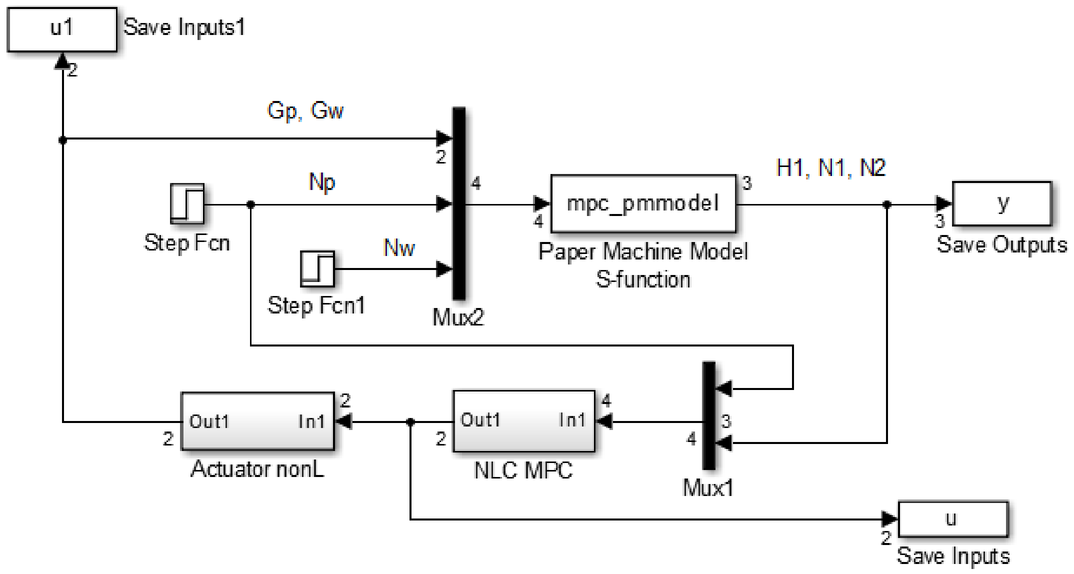

Figure 8: The updated Simulink model which is used simulate the process.

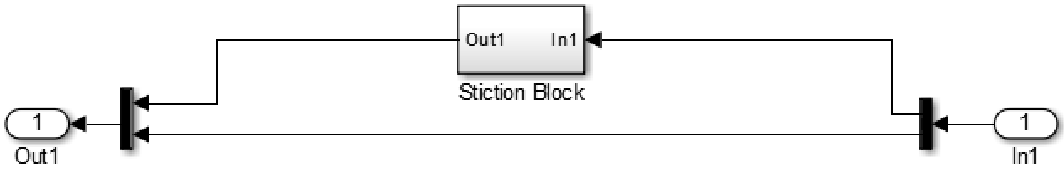

Figure 9: Simulink block for actuator stiction based on the Chowdhury model [34] contained within the 'Actuator nonL' subsystem block shown in Figure 8

The algorithm behind the stiction block is illustrated in the section above.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 UPDATING HYBRID MPC SOLVER INSTRUCTIONS

The optimization problem in the hybrid MPC formulation is a mixed integer quadratic programming (MIQP) problem which cannot be solved within MATLAB. After scanning through the code segment by segment, it was found that there were problems in the interface between MATLAB and GAMS. Certain function calls and referencing methods in the original hybrid MPC program were phased out over the years and are now obsolete.

After looking at alternative solvers, it was decided to that the most effective means to solving the problem was to fix the interface. Using other solvers would have required additional costs for licences and also time as more recoding would have been needed.

GAMS had developed the GAMS Data Exchange (GDX) protocol to allow quicker transfer of data between the GAMS software and MATLAB and also to hasten debugging. First, the variable definitions and syntax in MATLAB were modified in accordance with the GDX protocol. Once variables could successfully transferred to and read GAMS, the coding within GAMS pertaining to the handling of data had to be modified so that GAMS could send the solution to the optimization problem back to MATLAB. Finally, the segment of coding in MATLAB which was used to read values from GAMS was updated.

Once MATLAB could run the coding without terminating prematurely, this objective was considered to be accomplished.

4.2 SIMULATING CONTROL VALVE STICTION

As explained in the earlier section, several forms of physical and empirical stiction models exist. The advantages and limitations of the models were studied and are outlined in the previous section.

The Choudhury Model was selected due to it being able to accurately represent stiction behaviour without requiring valve parameters to be known and also due to its ease of implementation. A SIMULINK function block had already been developed for the Choudhury Model and could be easily integrated into the SIMULINK model.

This objective was considered completed once stiction behaviour could be observed in the control valve output.

For simplicity, G_p and G_w will thenceforth be referred to by Input 1 and Input 2 respectively whereas H_2, N_1 and N_2 will be known as Output 1, Output 2 and Output 3 respectively.

Manipulated Variables

G_p	flow rate of stock entering the feed tank
G_w	recycled white water flow rate

Measured Disturbance

N_p	consistency of stock entering the feed tank

Measured Output

H_2	liquid level in the headbox
N_1	feed tank consistency
N_2	headbox consistency

Unmeasured Disturbance

N_w	consistency of white water

Time is shown in seconds.

The following input and output charts demonstrate the response of the process to a setpoint change of H_2 from 0 to 1. Standard MPC is being used in this test and no actuator nonlinearity is present.

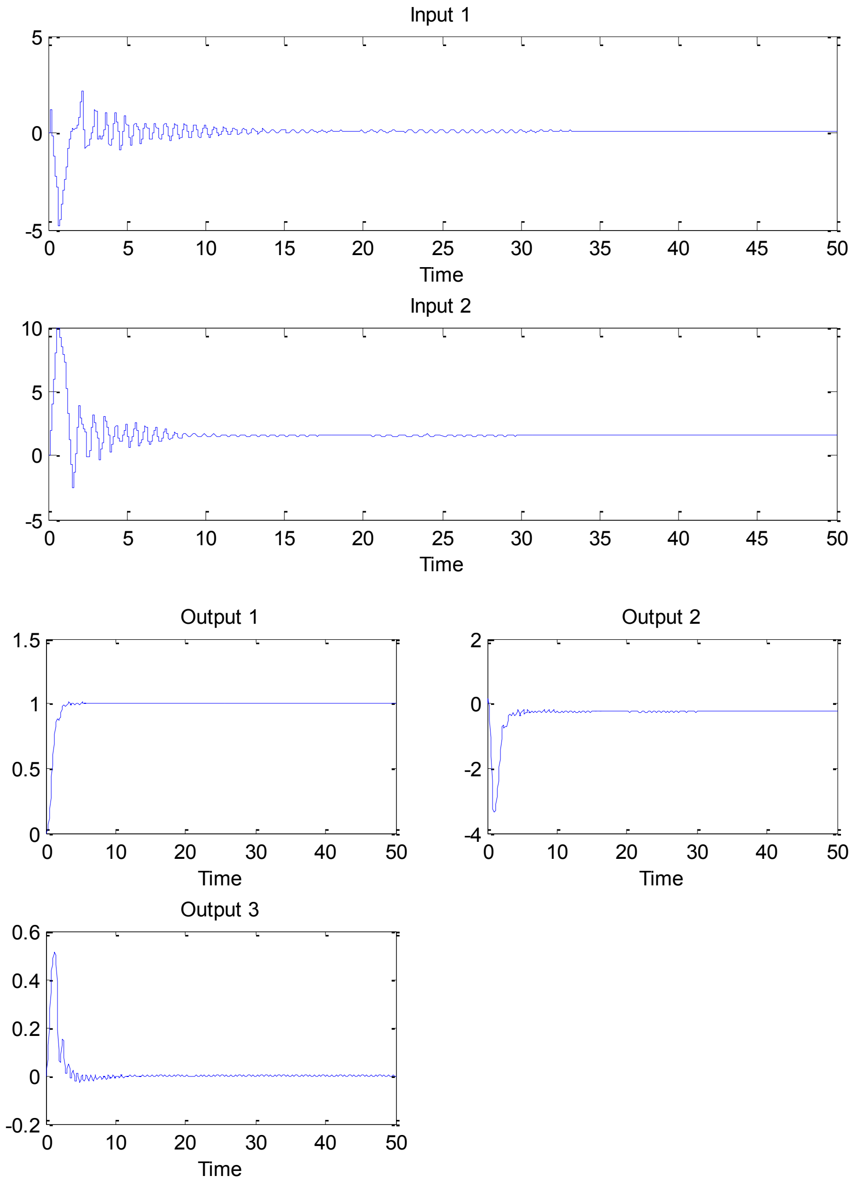

Figure 10: Baseline input-output response

Mild oscillation is observed initially at the input. The output appears to settle quickly and not deviate far from steady-state conditions. The response produced here is similar to that published by the developers of the paper machine headbox model.

With stiction (S = 1, J = 0) is incorporated into Input 1 of the process, both the input and output begin to oscillate and do not settle completely. Input 1 saturates for a few seconds and Input 2 reaches its maximum briefly. The outputs take longer to settle and continue to oscillate about their setpoint.

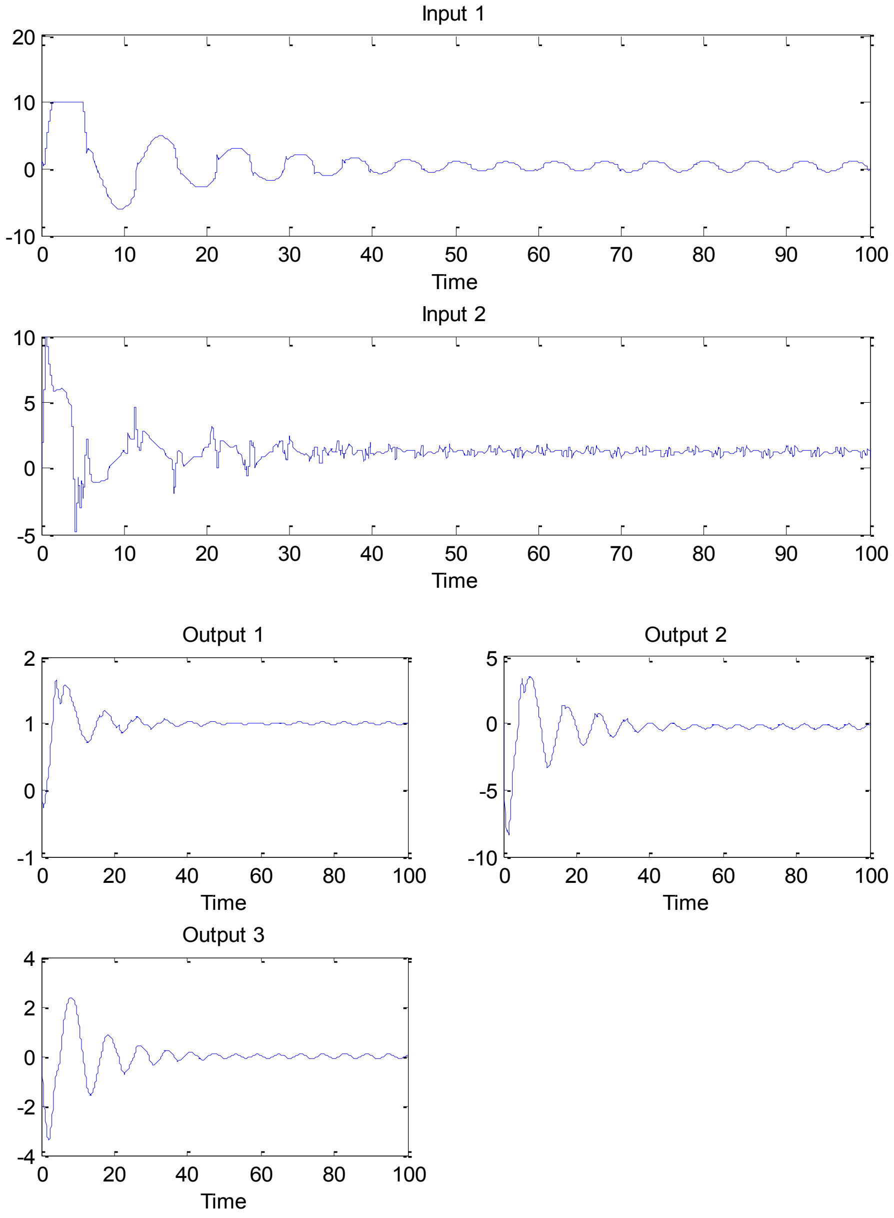

Figure 11: Response with mild stiction on Input 1

As shown below, the same pattern with larger oscillation is observed when S is set to 5.

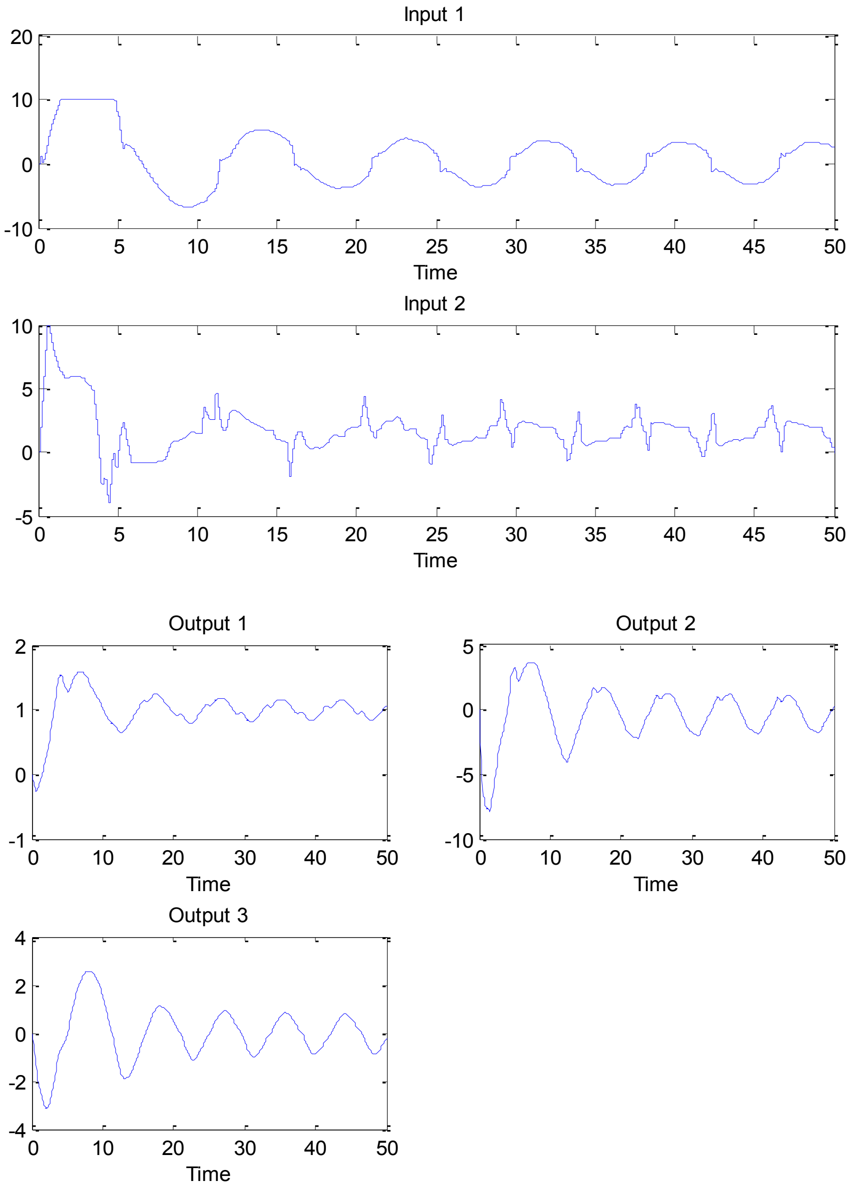

Figure 12: Response with strong stiction on Input 1

When stiction is applied on Input 2. MATLAB is no longer able to simulate stiction beyond the simulation time of 6.1 seconds probably because Outputs 2 and 3 rise to unreasonably high values such that MATLAB is unable to find a solution to the optimization problem. Furthermore, oscillation similar to that observed with stiction on Input 1 is not observed. For these reasons, it was decided that simulation on Input 2 was not to be conducted.

The response up to 6.1 seconds is shown below.

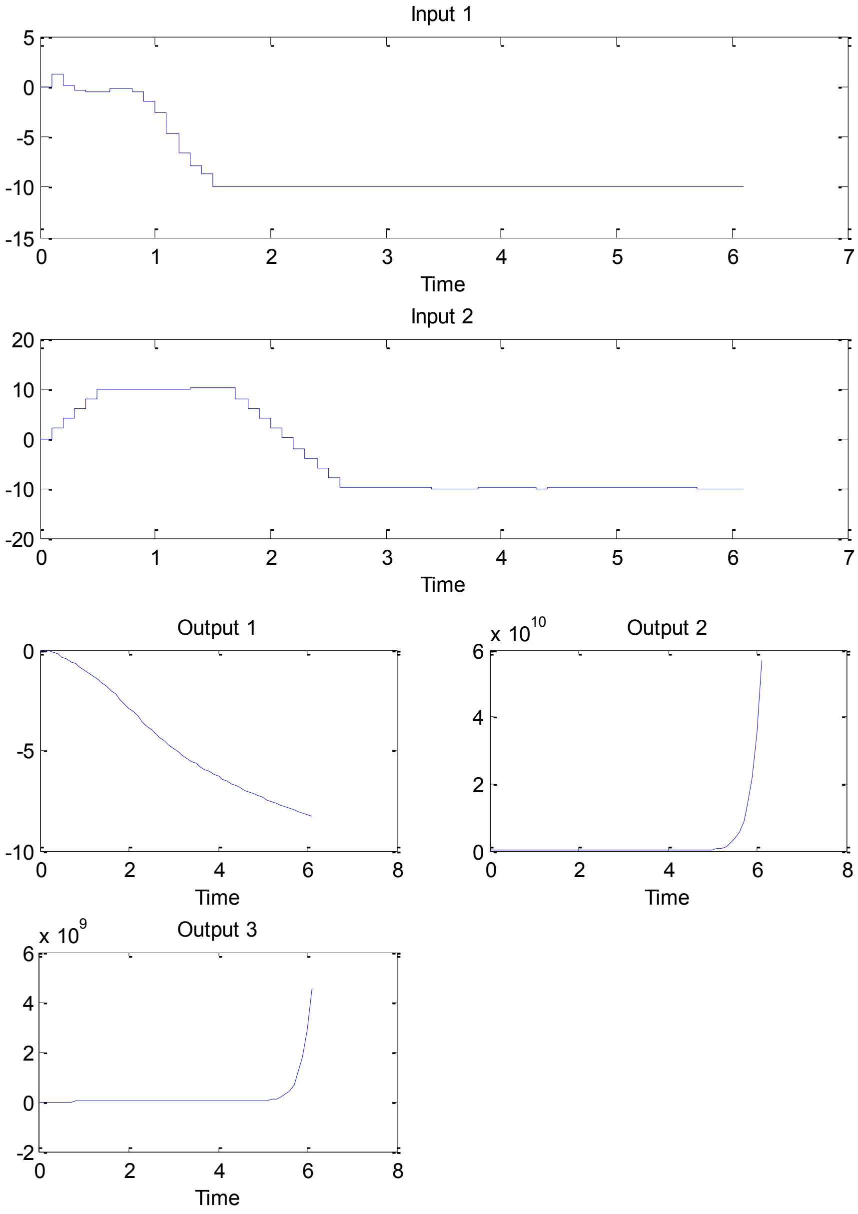

Figure 13: Response with stiction on Input 2

4.3 COMPENSATING FOR STICTION

The final part of this study was to investigate the ability of Hybrid MPC to compensate for stiction as it could for backlash. This objective is not fulfilled as the simulation did not produce reasonable results.

Two types of MPC algorithms were used throughout this study, standard MPC and hybrid MPC. The optimization problem in the standard MPC formulation could be solved within the MATLAB environment alone whereas Hybrid MPC required GAMS to solve the optimization. Hybrid MPC (optimized by GAMS) is expected to perform similar to standard MPC (optimized by MATLAB) if the compensation function is turned off.

However, as shown in Figure 14, this was not observed here. The standard MPC produced expected results while the hybrid MPC produced vastly different results even without the presence of any nonlinearity. Severe chattering in the inputs is observed and the outputs oscillate about their setpoint and the oscillation continues to increase in amplitude as time passes.

When stiction is introduced in Input 1, it violates its constraints and saturates. Input 2 responds in a manner inconsistent with the stiction behaviour observed previously. The outputs also do not reach their setpoint. This can be seen in Figure 15.

This problem was initially overlooked as much attention was being focused on reproducing stiction behaviour. As shown earlier, this was successfully done within the MATLAB environment. Preliminary tests on the formulation did not reveal this issue and even if it had shown up earlier, source of the inconsistency was not pinpointed.

Once discovered, efforts were taken to search for possible solutions to the error online and well as to search for MPC parameters that could give acceptable results by trial and error. Varying weights, setpoints, constraints, number of input moves and length of prediction horizon were not successful. This could be an indication that the objective function used in GAMS has not been specified correctly and that the optimization problem is weak.

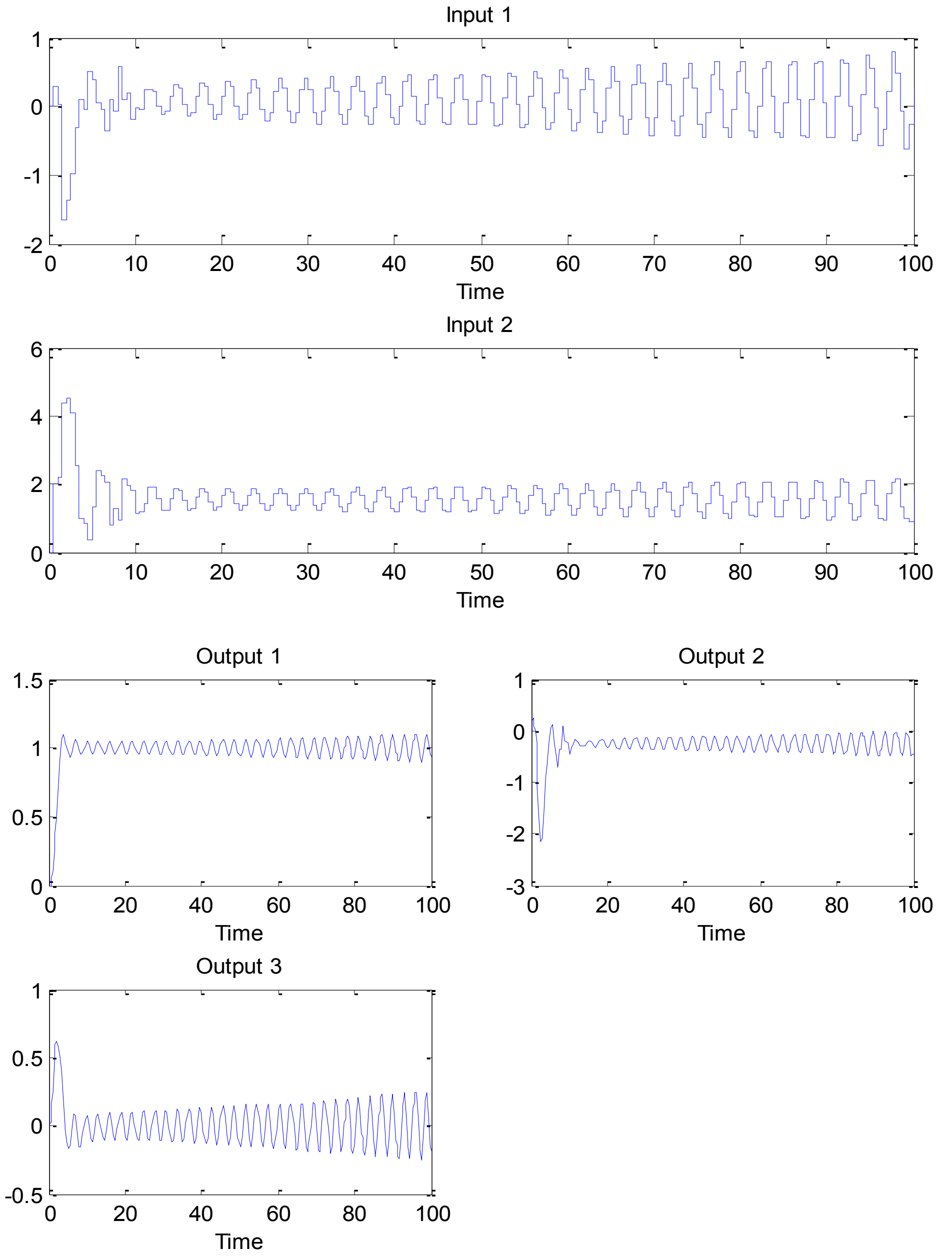

Figure 14: Response when optimization is solved using the GAMS with no stiction present

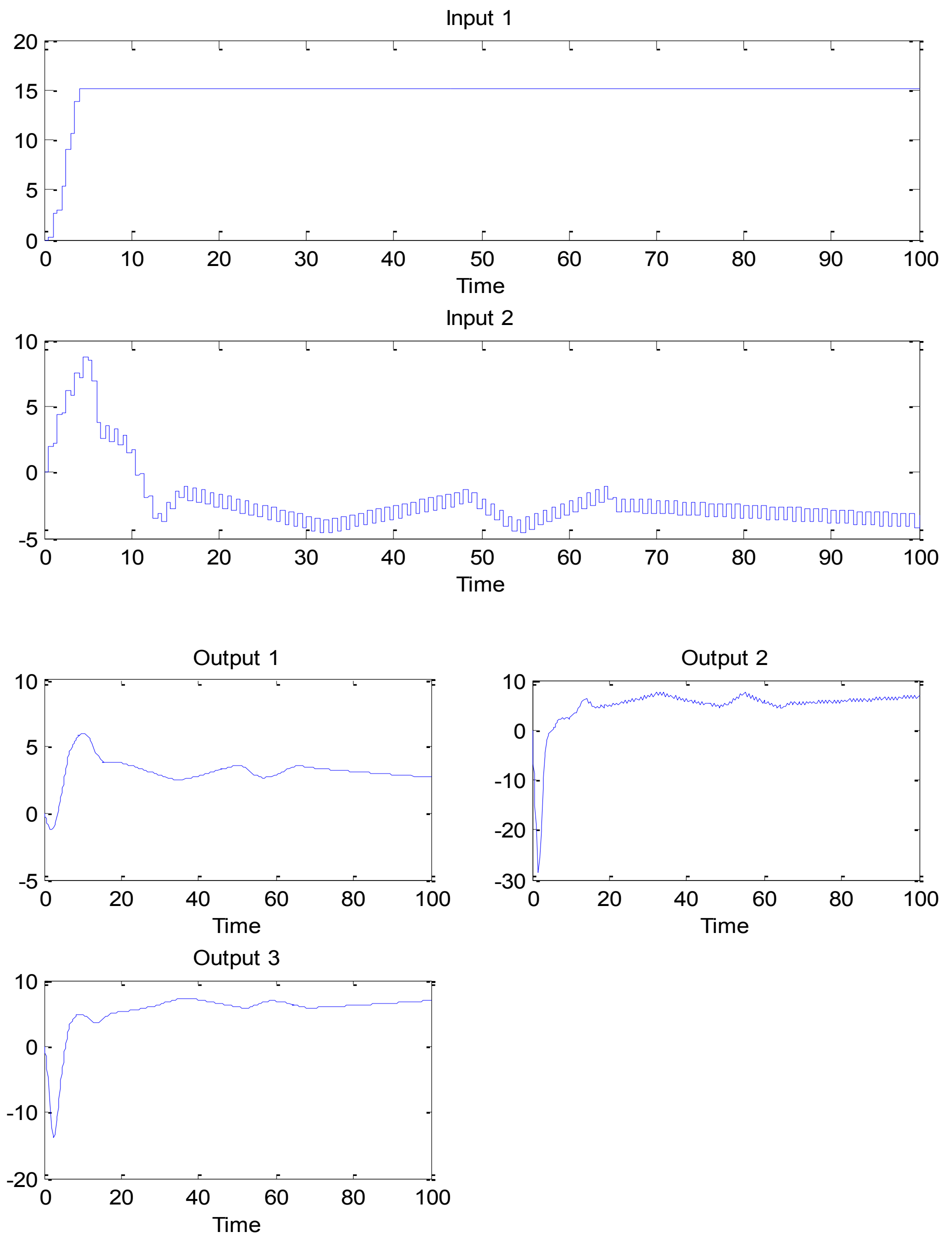

Figure 15: Response when optimization is solved using the GAMS with stiction present

CHAPTER 5

CONCLUSION

The hybrid MPC formulation developed by Zabiri and Sumyudia [11] was updated to run on current software versions of MATLAB and GAMS. The software interfacing between MATLAB and GAMS was tested and was found to run without runtime errors.

Several stiction models were studied and the Chowdhury Model was finally integrated into the system. Stiction behaviour was reproduced within the MATLAB environment.

The ability of hybrid MPC to compensate for stiction remains unknown. Problems were encountered with the simulation and the response produced was found to be unreasonable. When stiction was not present, excessive oscillation was observed and when it was present, little oscillation was observed and values reached high extremes. Efforts to find a solution to the simulation problem by varying MPC parameters did not succeed. Several recommendations for future work are provided.

CHAPTER 6

RECOMMENDATIONS

The source of the erroneous simulation results has not been ascertained. It is the recommendation of the author and Dr. Marappa Gounder Ramasamy that future researchers relook the GAMS algorithm and the solver configuration. Compare the current coding with the original then investigate if it could have caused a significant impact on the simulation. Also note that the GAMS algorithm used with the paper machine headbox model in this study was not exactly the same algorithm used by Zabiri in the original study. It was also developed by Zabiri but for another purpose and was expected to produce similar results.

If at all possible, solving the MIQP optimization problem within the MATLAB environment could be simpler and make the validity of the simulation less questionable. This recommendation is supported by Dr. Asna Mohd Zain who evaluated this project during the qualification screening for entry to the Science, Engineering and Design Exhibition (SEDEX).

Future researchers should also note that the latest paper machine headbox model specifies four outputs instead of the three shown in the study. The fourth output is the consistency of white water, N_w is an unmeasured disturbance. Several runtime errors which carried along to separate parts of the simulation as changes were made when trying to integrate the fourth output were encountered. Therefore, it was left out as it does not affect the dynamics of the system.

As suggested by Dr. Timothy Ganesan Andrew during the viva voce for this project, future researchers could look at system characterization. With the introduction of stiction, the system may no longer be able to be represented by a linear quadratic model. Therefore, linear MPC would not be appropriate for this problem. However, this could open up a new dimension of study in the lesser researched area of nonlinear MPC.

CHAPTER 7

REFERENCES

[1] M. Choudhury, S. L. Shah, and N. F. Thornhill, *Diagnosis of Process Nonlinearities and Valves Stiction: Data Driven Approaches*. Berlin: Springer, 2008.

[2] J. H. Lee, "Model predictive control: review of the three decades of development," *International Journal of Control, Automation and Systems,* vol. 9, pp. 415-424, 2011.

[3] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control: Applications to industrial processes," *Automatica,* vol. 14, pp. 413-428, 1978.

[4] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering,* vol. 82, pp. 35-45, 1960.

[5] E. F. Camacho and C. B. Alba, *Model predictive control*: Springer, 2013.

[6] T. C. S. Wibowo, N. Saad, and M. N. Karsiti, "The simulation of MISO MPC for gaseous pilot plant control with presence of measurement noise," in *Intelligent and Advanced Systems, 2007. ICIAS 2007. International Conference on*, 2007, pp. 1107-1110.

[7] A. Li, "COMPARISON BETWEEN MODEL PREDICTIVE CONTROL AND PID CONTROL FOR WATER-LEVEL MAINTENANCE IN A TWO-TANK SYSTEM," Master of Science, The Swanson School of Engineering, University of Pittsburgh, Beijing, 2010.

[8] J. B. Rawlings, "Tutorial overview of model predictive control," *Control Systems, IEEE,* vol. 20, pp. 38-52, 2000.

[9] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica,* vol. 36, pp. 789-814, 6// 2000.

[10] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice,* vol. 11, pp. 733-764, 7// 2003.

[11] H. Zabiri and Y. Samyudia, "A hybrid formulation and design of model predictive control for systems under actuator saturation and backlash," *Journal of Process Control,* vol. 16, pp. 693-709, 2006.

[12] L. Desborough and R. Miller, "Increasing customer value of industrial control performance monitoring-Honeywell's experience," in *AIChE symposium series*, 2002, pp. 169-189.

[13] M. Daneshwar and N. M. Noh, "Valve stiction in control loops—A survey on effective methods of detection and compensation," in *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*, 2012, pp. 155-159.

[14] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica,* vol. 35, pp. 407-427, 1999.

[15] C. N. Jones and M. Morari, "Polytopic approximation of explicit model predictive controllers," *Automatic Control, IEEE Transactions on,* vol. 55, pp. 2542-2553, 2010.

[16] M. Choudhury, S. L. Shah, N. F. Thornhill, and D. S. Shook, "Automatic detection and quantification of stiction in control valves," *Control Engineering Practice,* vol. 14, pp. 1395-1412, 2006.

[17] C. Garcia, "Comparison of friction models applied to a control valve," *Control Engineering Practice,* vol. 16, pp. 1231-1243, 10// 2008.

[18] B. C. Silva and C. Garcia, "Comparison of Stiction Compensation Methods Applied to Control Valves," *Industrial & Engineering Chemistry Research,* vol. 53, pp. 3974-3984, 2014/03/12 2014.

[19] S. Sivagamasundari and D. Sivakumar, "A Practical Modelling Approach for Stiction in Control Valves," *Procedia Engineering,* vol. 38, pp. 3308-3317, 2012.

[20] J. Gerry and M. Ruel, "How to measure and combat valve stiction online," in *ISA International Fall Conference, Houston, TX*, 2001.

[21] H. Olsson, K. J. Åström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky, "Friction Models and Friction Compensation," *European Journal of Control,* vol. 4, pp. 176-195, // 1998.

[22] D. Karnopp, "Computer simulation of stick-slip friction in mechanical dynamic systems," *Journal of dynamic systems, measurement, and control,* vol. 107, pp. 100-103, 1985.

[23] B. Armstrong-Hélouvry, P. Dupont, and C. C. De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica,* vol. 30, pp. 1083-1138, 1994.

[24] A. Stenman, F. Gustafsson, and K. Forsman, "A segmentation ‐ based method for detection of stiction in control valves," *International Journal of Adaptive control and signal processing,* vol. 17, pp. 625-634, 2003.

[25] M. Kano, H. Maruta, H. Kugemoto, and K. Shimizu, "Practical model and detection algorithm for valve stiction," in *IFAC symposium on dynamics and control of process systems*, 2004, pp. 5-7.

[26] Q. P. He, J. Wang, M. Pottmann, and S. J. Qin, "A curve fitting method for detecting valve stiction in oscillating control loops," *Industrial & engineering chemistry research,* vol. 46, pp. 4549-4560, 2007.

[27] M. A. A. S. Choudhury, Thornhill, N. F., Shah, S. L., "A Data-Driven Model for Valve Stiction," in *ADCHEM 2003*, Hong Kong, 2004.

[28] A. Kayihan and F. J. Doyle III, "Friction compensation for a process control valve," *Control engineering practice,* vol. 8, pp. 799-812, 2000.

[29] T. Hägglund, "A friction compensator for pneumatic control valves," *Journal of Process Control,* vol. 12, pp. 897-904, 2002.

[30] R. Srinivasan and R. Rengaswamy, "Approaches for efficient stiction compensation in process control valves," *Computers & Chemical Engineering,* vol. 32, pp. 218-229, 2008.

[31] H. Zabiri, A. Maulud, and N. Omar, "NN-based algorithm for control valve stiction quantification," *WSEAS Transaction on Systems and Control,* vol. 4, pp. 88-97, 2009.

[32] Y. Ying, M. Rao, and Y. Sun, "BILINEAR CONTROL STRATEGY FOR PAPER-MAKING PROCESS," *Chemical Engineering Communications,* vol. 111, pp. 13-28, 1992/01/01 1992.

[33] MATHWORKS. (22 September 2014). *Paper Machine Process Control*. Available: http://www.mathworks.com/help/mpc/ug/paper-machine-process-control.html

[34] M. A. A. Shoukat Choudhury, N. F. Thornhill, and S. L. Shah, "Modelling valve stiction," *Control Engineering Practice,* vol. 13, pp. 641-658, 5// 2005.