Robust Model Predictive control of Cement Mill circuits

A THESIS

submitted by

M GURUPRASATH

(Roll Number: clk 0603)

for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF CHEMICAL ENGINEERING.

NATIONAL INSTITUTE OF TECHNOLOGY,

TIRUCHIRAPPALLI-620015

August2011

THESIS CERTIFICATE

This is to certify that the thesis titled **Robust Model Predictive control of Cement Mill circuits**, submitted by **GuruPrasath**, to the National Institute of Technology, Tiruchirappalli, for the award of the degree of **Doctor of Philosophy**, is a bonafide record of the research work carried out by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

M. Chidambaram
Research Guide
Department of Chemical Engineering
National Institute of Technology
Tiruchirappalli- 620 015. India

Tiruchirappalli- 620 015.
Date:

ACKNOWLEDGEMENTS

I would like to thank Prof. M. Chidambaram for his invaluable guidance and suggestions. The timely completion of this thesis is due to his constant support, boundless work and planning. Even though he was busy as Director he spent much time with me for discussions. In the last few months of this thesis, even though, he is in IIT, Madras, he guided me in completing the thesis in time. Organizing things in a suitable time frame inspired me to finish many things well before time. It's my privilege to work with him.

I would like to thank M/s. FLSmidth Private Limited for allowing me to use the commercial simulation package CEMulator for conducting various experiments to compare various controllers. Also I would like to thank M/s. UltraTech Cement, Arakkonam for trusting me and allowing to test the controller online.

I would also like to thank my co-supervisor Dr. John Bagterp Jørgensen without whom this Ph.D would not have been possible in its current form. Even though he is busy in DTU, he is providing continuous support and guidance through online discussions.

I would like to thank Prof. T. K. Radhakrishnan for his constant support during initial phase of this PhD work and during the last few months of the thesis work. His initiatives and suggestions inspired me in completing things well before time.

I would like to express my gratitude to Mr. T. R. Hari, Mr. Sudeep Sar and Mr. Rajendra Bhargava of FLSmidth Private Limited, for their encouragement and support during the complete research work. It has been privilege to work with them as they not only provided financial support but also motivated me in all possible ways.

My great thanks to Bodil Recke, Jørgen Knudsen, Hassan Yazdi and Bo Frederiksen of FLSmidth A/S. who initiated the research contract and kept always open to the ideas that we suggested.

I am extremely thankful to my doctoral committee members, Prof. B. Venkatramani, Dr. SankaraNarayanan, and Prof. T. K. Radhakrishnan for their critical comments and valuable suggestions in due course of time. Also I would like to thank Prof. N. Anantharaman for his timely support in completing this research work.

I thank all the staff members in the Chemical Engineering Department for their help and Co-operation.

I have no words to thank my colleague and friends Kavitha Stanley, Sridhar. P, Jose Pinto and Ajit Balaji the way encouraged and motivated me in all possible ways particularly in doing experimental works in office and plant.

Finally this work might not have been completed without support of my family. It is impossible to put in words the way my mother and wife supported me in completing my thesis in due course of time.

ABSTRACT

KEYWORDS: MPC ; Cement Mill; Moving Horizon; constraints.

The present work considers the control of ball mill grinding circuits which are characterized by non-linearities and disturbances. The disturbances are due to large variations and heterogeneities in the feed material. Thus the models obtained by simple tests on these mills are subject to large uncertainties which may result in poor performance of conventional control solutions.

A regularized ℓ_2-norm based finite impulse response (FIR) predictive controller with input and input-rate constraints is developed. The estimator used is based on a simple constant output disturbance filter. The FIR based regulator problem is solved by convex quadratic program (QP) by converting the objective function into a standard form. The QP is solved using an algorithm based on interior point method. The model used here is single input- single output SOPDT with a zero transfer function model. The performance of the predictive controller in the face of plant-model mismatch is investigated by simulations.

In order to improve the performance of MPC, a moving horizon constrained regularized ℓ_2 estimator based on impulse response models is developed. Here the estimator is used to estimate the unknown disturbance by solving the optimization problem. By using a SOPDT with a zero transfer function model, the performance of the estimator with measurement noise is provided. Also the closed loop performance of a MPC with moving horizon estimator with SISO system is investigated.

The predictive controller is equipped with soft output constraints that are used to have robustness against plant-model mismatch. Soft output constraints are the limits around the set point, where the errors are penalized minimum within a dead band called soft limits and penalized heavily as soon as the error exceeds the

band. By simulation first with SISO system and then with 2×2 MIMO system, the performance of the proposed controller and conventional predictive controller is investigated. For MIMO system, the input vectors are Elevator Load, Fineness and the output vectors are Feed rate, Separator speed. In case of plant-model mismatch more than 50 %, the conventional MPC response become oscillatory, whereas the soft MPC provides lesser variations in output resulting in much stable response.

The Model Predictive Controller (MPC) with soft constraints is used for regulation of a cement mill circuit. The uncertainties in the cement mill model are due to heterogeneities in the feed material as well as operational variations. The uncertainties are characterized by the gains, time constants, and time delays in a transfer function model. The controllers are compared using a rigorous cement mill circuit simulator. The simulations reveal that compared to conventional MPC, soft MPC regulates cement mill circuits better by reducing the variations in manipulated variables by 50%.

The performance of MPC with soft constraints is also compared with existing Fuzzy Logic controller implemented in a real time plant with closed circuit cement ball mill. The real time results show that the standard deviation of manipulated variables and the controlled variables are reduced with the soft MPC. The reduction in standard deviation in quality is 23%.

The performance of the same controller (MPC) applied to a cement ball milling circuit with large measurement sample delay is investigated. Usually fineness is measured hourly by sample analysis in the laboratory. The predictive controller designed with the model from fast sample data (1 min sample) when applied to control with hourly sampled measurements, the controller is to be re-tuned. The parameters needed to be re-tuned are weight for penalties on measurement error (Q_z), weights of penalties on manipulated variables (S) and rate of change of manipulated variables (R). Also the hard constraints on input- rate movements are to be re-tuned to reduce the variations in the controller. In this work, the controller model is added with half the sample delay of measurement to improve the performance of the controller with one time tuning.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

NITT	National Institute of Technology, Tiruchchirappalli
MPC	Model Predictive Control
MHE	Moving Horizon Estimation
MV	Manipulated Variable
CV	Controlled Variable
OPC	Ordinary Portland Cement
PPC	Puzzalona Portland Cement
SISO	Single Input- Single Output
MIMO	Multi Input - Multi Output
FOPDT	First Order Plus Dead Time
SOPDT	Second Order Plus Dead Time

NOTATION

A	System matrix
B	Input matrix
B_d	Input disturbance matrix
d_k	Disturbance vector
y_k	Output
w_k	Process noise
v_k	Measurement noise
u_k	Input
b_k	Bias Estimate
P_0, Q	Variance of Process noise
R	Variance of measurement noise
H_i	Impulse response coefficients
H	Hessian matrix
g	gradient
Q_z	Weight on measurement error
S	Weight on Actuator movements
N	Prediction Horizon
n	Control Horizon
ϕ	Objective Function
η	Soft constraint
S_η	Weight on Soft constraint
β	Zero of the system
τ_1, τ_2	Time constants
τ_d	Time Delay

CHAPTER 1

Introduction

Model predictive control (MPC) has become a standard technology in the high level control of chemical processes. MPC or receding horizon control is a form of control in which the control action is obtained by solving on-line, at each sampling instant, a finite open-loop optimal control problem, using the current state of the plant as the initial state; the optimization yields an optimal control sequence in which the first control move is applied to the plant.

However, only very little guidelines are available regarding tuning methodologies of such controllers in the face of the inevitable plant-model mismatch. The closed-loop performance of nominal linear model predictive control can be quite poor when the models are uncertain. Consequently, some years after commissioning, many high-level control systems are turned off due to poor closed-loop performance. This is often due to changes in the plant dynamics caused by wear and tear combined with lack of the necessary human resources at the plant to re-tune and maintain the MPC.

Using soft output constraints along with hard constraints in a novel way, the poor performance of predictive control in the case of plant-model mismatch can be improved significantly. Constraints are physical limitations the control system must take into consideration when implementing control actions. Usually constraints to the controllers are of two types, hard constraints and soft constraints. Hard constraints are those which are need to be necessarily satisfied, whereas soft constraints can be violated but penalized heavily whenever violated. A constrained optimization problem is one in which there are inequality or equality constraints that are imposed while seeking to maximize an objective function.

Alternatively, a constrained optimization problem can be defined as a regular constraint satisfaction problem augmented with a number of "local" cost functions.

The aim of constrained optimization is to find a solution to the problem whose cost, evaluated as the sum of the cost functions, is minimized. Excellent review on Model predictive control and optimization methods are available on Maciejowski (2002), Camacho and Bordons (2004) and Rossiter (2003).

The cement mill circuit requires many soft output constraints to be considered in a MPC formulation of the control problem. It is one of the best examples of highly non-linear system to be controlled by a linear model. The uncertainties in the system are large enough to cause the plant-model mismatch quite often.

Further, Comminution is a major unit operation in a cement plant, accounting for about 50 - 75 % of the total plant energy consumption. Comminution can be of two types, ball mill grinding and vertical roller mill grinding. Nevertheless when grinding is required the ball mill is the most accepted element in the cement grinding. The reasons are high reliability, the good possibility of gypsum dehydration, simple operation (does not necessarily mean efficient) and the easy to maintain construction. Finish grinding based on ball mill operation in general is extremely inefficient. Just 4 % of energy available is efficiently used for grinding. Loading the cement mill too little results in early wear of the steel balls and a very high energy consumption per tonnes cement produced. Conversely, loading the mill too much results in inefficient grinding such that the product quality cannot be met. Cement quality is measured by its chemical composition and its particle size distribution. Blaine is an aggregate number for the particle size distribution measuring the specific surface area of the cement powder.

Loading the cement mill too much, may even result in a phenomena called plugging such that the plant must be stopped and plugged material removed from the mill. Consequently, optimization and control of their operation are very important for running the cement plant efficiently, i.e. minimizing the specific power consumption and delivering consistent product quality meeting specifications. New control methodologies are proposed for improving the performance of such process. The improved operation resulting from these controllers can potentially lead to large energy savings and at the same time provide a more consistent product quality.

1.1 Motivation

Extensive research is being conducted for improving the performance of different modules in the cement process. The control of ball mill grinding circuit is considered as the most important and difficult control problem. Efficient control is required in order to reduce the specific production costs while maintaining the product quality, at an acceptable level. The control philosophy for cement mill thus remains challenging.

Conventionally, the grinding circuits are controlled by multi-loop PID controllers, but these controllers generally have drawbacks, such as input/output pairing problems and hard tuning work. For grinding circuits characterized by large time delays, a predictive control is more suitable in this case (Chen *et al.*, 2009).

Rajamani and Herbst (1991*a*) have proposed feedback and optimal control methods for optimizing ball mill grinding circuits. But the controller does not consider the constraints in the real time system thus the controller may attain unstable operating ranges quite often.

Van Breusegem *et al.* (1994) and de Haas *et al.* (1995) have developed an LQ controller for the cement mill circuit. This controller was based on a first order 2×2 transfer function model identified from step response experiments. Ramasamy *et al.* (2005) have developed constrained MPC using MATLAB toolbox based on input/output models for the control of cement mill circuit.

From the above studies, it is evident that the model predictive controller are commonly used because of its robustness and handling of constraints. But the performance of the controller mainly depends on model developed in each of the methods. Normally cement mills have large uncertainties and there will always be plant mode mismatch. The above works on cement mill control use linear models and do not take care of uncertainties in the system. Also considering the hard constraints in the controller the solution becomes infeasible and also the controller reaches saturation quite often.

When non-linear control algorithms are considered, Magni *et al.* (1999) and Grognard *et al.* (2001) have developed a Nonlinear Model Predictive Control algorithm based on a lumped nonlinear model of the cement mill circuit. But these works are based on neural network modeling and cannot extrapolate the conditions when operating ranges shift. Also they are computationally complex and the models have to be reduced to be used in control algorithm which results in infeasible control actions.

Scokaert and Rawlings (1999) have proposed a state based soft constraint approach for handling the infeasibility with respect to conventional predictive control approach. They illustrated by using a non- minimum phase system that state constraints will be included when the solution becomes infeasible. The main drawback of such controllers are they require larger QP solutions resulting in slower response which may not be feasible in real time.

Another method on range control, Roubal and Havlena (2005) have provided a soft limit band for the output where the controller does not react for change in output within the region, such methods always leave offset like a normal dead band controller. A hard constrained MPC is converted into soft constrained by introducing a slack variable (Kerrigan and Maciejowski, 2000), here the slack variable will be included in the objective function when the controller solutions become infeasible during the hard constraint approach.

The estimator design for such constrained MPC has been one of the most area of research as it helps in determining the unknown disturbances for providing efficient control solutions. Based on linear state space models, Muske and Rawlings (1993a) have presented a moving horizon estimator and used input or output disturbances to have steady state offset free control. Here the estimator is used to determine the unknown disturbances in the system based on state space method.

Boyd and Vandenberghe (2004) have used Finite Impulse Response models for robust linear programming. The main advantage of using FIR models is that they are in a form that can be easily applied to robust linear programming like second order cone programming and also can be useful for parametrization of the system with missing observations.

The research works referred above have been proven theoretically and no steps have been taken to use the methods in real time. Based on the research works cited above and by considering the issues involved in the cement mill control because of uncertainties present, a robust model predictive controller is necessary which can handle the variations in cement mill control because of such uncertainties and also computationally simple.

1.2 Objectives

The main objectives of the investigations in this thesis are:

1. (a) To develop a predictive controller based on FIR models with 'ℓ_2' regression norm along with input and input-rate constraints with a simple estimator and to evaluate the performance of the controller related to the uncertainty of impulse response co-efficients.

   (b) To develop a regularized l_2 moving horizon estimator based on finite impulse response (FIR)models with input and input-rate constraints and to evaluate the closed loop performance of the above estimator with a predictive regulator.

2. (a) To develop a robust soft constraints based predictive controller with simple estimator for linear systems.

   (b) To compare the performance of the constrained controller with nominal predictive controller by simulation.

3. (a) To implement the soft MPC in a real time cement mill circuit and compare the performance with that of the other controllers

   (b) To evaluate by simulation the performance of soft MPC handling the large sample delay measurements

The organization of the thesis is as follows:

Chapter 2 provides the basic motivation of the work with a detailed literature survey on model predictive controllers with hard and soft constraints and control strategies for cement mill circuit.

Chapter 3 provides the discussion on Model Predictive Control Based on Finite Impulse Response Models with simple estimator. This chapter also presents the details on deriving a Moving Horizon Estimation. Also this chapter presents the details on Model Predictive Control with Soft Output Constraints is provided.

Chapter 4 gives Comparison of Soft MPC with conventional MPC using simulation. First the controllers are compared with simple SISO system. Then a model of cement mill is considered for comparing the closed loop performance of the controllers using Matlab.

Chapter 5 gives the basics on Cement Manufacturing Process and Cement Milling circuit. Also the basic control strategy of cement mill circuit is discussed.

In Chapter 6 applications of Soft MPC to Cement Mill Circuit are discussed. A transfer function model of cement mill obtained and the controller is implemented in the simulator. The performance of the controller is then compared with conventional MPC in simulator. The soft MPC is then implemented in real plant and the performance of the controller is compared with already existing Fuzzy Logic controller.

Chapter 7 provides the detailed study on Implementation of MPC to a Large Sample Delay System. Here the controller performance is investigated using the cement mill simulator first with every minute sample and then with model including the sample delay.

Summary and Conclusions are given in Chapter 8.

Appendix A gives the formulation of Quadratic program for FIR based MPC, MHE and Soft Constraints based MPC.

Appendix B gives the flow chart for the MPC execution in MATLAB.

Appendix C gives the generalized form of deriving Quadratic program

Appendix D provides the basic algorithm of Interior point methods.

Appendix E provides the Matlab codes for design of soft MPC and simulation in closed loop.

Appendix F gives a brief description of ECS/CEMulator system where the performance of the controllers are compared.

CHAPTER 2

Literature Survey

In this chapter, the published literature is reviewed on the model predictive controllers generally used in industries and the MPC with hard constraints. A brief review of soft constraint based MPC and controller for cement industries is also presented.

Excellent review on model predictive control is available. Reviews on stability of model predictive control are given by Mayne *et al.* (2000), Zheng (1998), Zheng and Morari (1995) and Limon *et al.* (2006) and a review on tuning methods have been provided by Garriga and Soroush (2010). Detailed survey reports on industrial applications of model predictive control are given by Bemporad and Morari (1999) and Morari and Lee (1999) and Qin and Badgwell (2003) and Bemporad and Morari (1999). Garcia *et al.* (1989) have discussed the basic theory on model predictive control .

2.1 Model Predictive Control in Industries

There are many control strategies in use today like intelligent control, adaptive control, stochastic control, optimal control etc. Optimal control is such a control technique in which we minimize certain cost index to achieve desired performance. The two types of optimal control techniques are

- Linear Quadratic Gaussian (LQG)
- Model Predictive Control (MPC)

Model Predictive Control technique is the most widely used technique in industry as opposed to LQG based controllers. The LQG controllers were termed as failure and the reasons for this failure are given by Garcia *et al.* (1989) and Richalet *et al.* (1976). They have provided the reasons that the LQG controllers are not successful because they cannot handle the following:

- constraints
- process nonlinearities
- model uncertainty (robustness)
- unique performance criteria

Further, MPC is classified into Linear MPC and Non-Linear MPC depending on the specific problem statement. Both linear and nonlinear systems have specific problem statements and utilize different optimization methods. Non- Linear MPC uses non-linear models for prediction and it requires iterative solution of optimal control problems on a finite prediction horizon. But non-linear MPC cannot be solved as convex optimization problem. Some of the work on non-linear MPCs are given by Miller *et al.* (2000) and Santos *et al.* (2008). They have provided a tool to analyze the stability of constrained non-linear model predictive control.

Linear MPCs are most commonly used techniques in industry because of computational simplicity and faster solutions in solving real time optimization problems. Further, linear MPC used in real time applications can be classified into following types

- Dynamic Matrix Control (DMC)
- IDCOM (Identification- Command)
- General Predictive Control (GPC)
- Moving Horizon Control (MHC)

These major classification of MPC is based on the type of algorithm used for solving optimization problem. While the MPC paradigm encompasses several different variants, each one with its own special features, all MPC systems rely on the idea of generating values for process inputs as solutions of an on-line (real-time) optimization problem.

2.2 Model Predictive Control

Model Predictive Control, or MPC, is an advanced method of process control that has been in use in the process industries such as chemical plants and oil refineries since the 1980s. MPC as the name suggests use explicit models of the plant to predict the future behavior of the controlled variables. Based on the prediction, the controller calculates the future moves on manipulated variables by solving the optimization problem online. Here the controller tries to minimize the error between predicted and the actual value over a control horizon and the first control action is being implemented. Model predictive controllers rely on dynamic models of the process, most often linear empirical models obtained by system identification. MPC is also referred to as receding horizon control or moving horizon control (Qin and Badgwell, 2003).

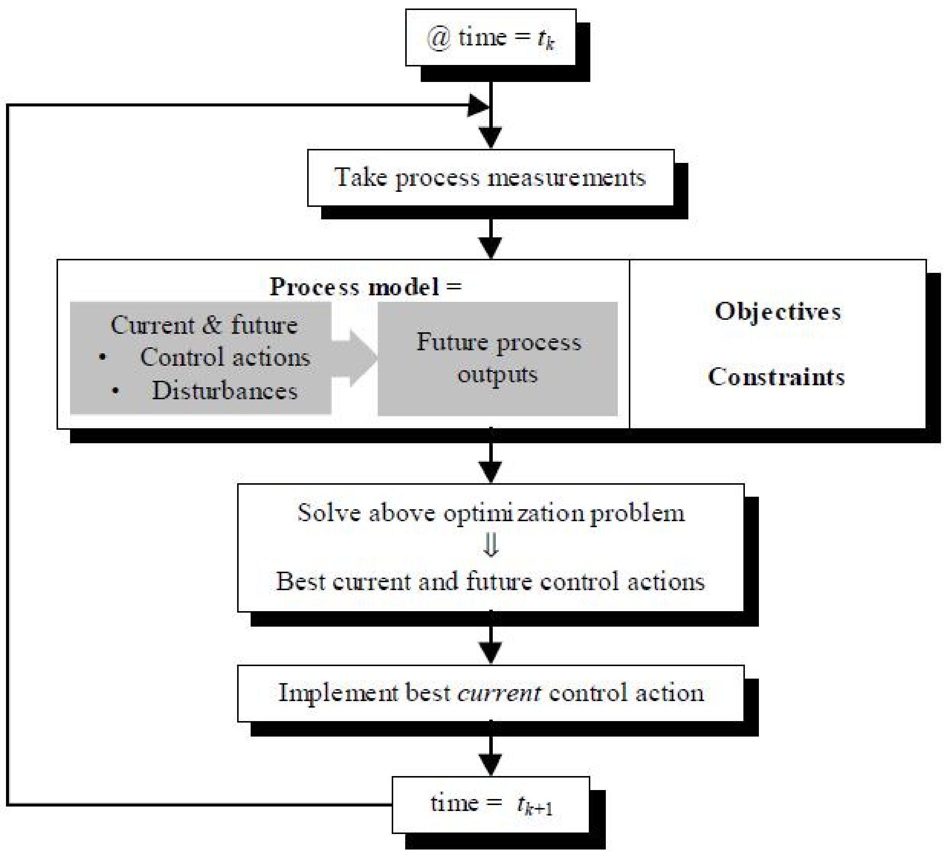


Figure 2.1: Model Predictive Control Scheme.

Figure 2.1 also makes it clear, that the behavior of an MPC system can be quite complicated, because the control action is determined as the result of the on-line optimization problem. The problem is constructed on the basis of a process model and process measurements. Process measurements provide the feedback (and, optionally, feed-forward) element in the MPC structure. Figure 2.1 shows the structure of a typical MPC system. Normally different types of MPCs provide different approaches in handling the following.

- Input-output model,
- disturbance prediction,
- objective function,
- measurement,
- constraints, and
- sampling period (how frequently the on-line optimization problem is solved).

Regardless of the particular choice made for the above elements, on-line optimization is the common thread tying them together.

2.2.1 Elements of MPC

All the MPC algorithms possess common elements and different options can be chosen for each element giving rise to different algorithms. These elements are

- Prediction Model
- Objective Function and
- Control Law

Prediction Model

The model is the cornerstone of MPC; a complete design should include the necessary mechanisms for obtaining the best possible model, which should be complete enough to fully capture the process dynamics and allow the predictions to be calculated, and at the same time to be intuitive and permit theoretic analysis. The use of the process model is determined by the necessity to calculate the predicted output at future instants. The different strategies of MPC can use various models to represent the relationship between the outputs and the measurable inputs, some of which are manipulated variables and others are measurable disturbances which can be compensated by feed forward actions. Some of the available types of models are

- Finite impulse response model
- Step response model
- State space model
- Transfer function descriptions like AR(MA)X models
- Auto- Regression with external input (ARX) model

Various types of models are used with MPC, with the FIR (Finite Impulse Response) or Step response models and ARX (Auto-Regressive with eXternal inputs) models being the most common in industrial practice. Step or impulse response models are non- parametric models that are widely used in industries. The advantage of such models are, they reveal plant time constant, gain and delay directly from the process graphs. Also FIR models requires less prior information than transfer function models.Also FIR models need the information of only settling time which can be easily attained. These are the main advantages of using FIR models where the plant has many input- output variables and has complicated dynamic responses due to interactions.

But the disadvantages of FIR models are that they can be used for only stable systems and is difficult to be used in identifying processes with slow dynamics. In such cases, transfer functions models are used where the dynamics are slow and can be converted into any form like ARX for linear systems and ARMAX model for non-linear applications. But the model mismatch could cause bias in the estimated parameters.

State space model formulation can be used to augment the model easily with additional states to represent the effect of disturbances. Also it can be provided in both linear and non-linear form. These are easy to determine the system both in continuous form and discrete form but it is quite difficult to determine the state space models in real time.

Objective Function

The various MPC algorithm propose different cost functions for obtaining the control law. The general aim is that the future output on the considered horizon should follow a determined reference signal and at the particular constraint. The objective functions are either minimization or maximization problems depending on the application. Normally cost functions used in process controls are minimization functions with some inequality constraints.

Obtaining the Control Law

In order to obtain values it is necessary to minimize the functional part of the objective function. To do this, the values of the predicted outputs are calculated as a function of past values of inputs and outputs and future control signals making use of the model chosen and substituted in the cost function, obtaining an expression whose minimization leads to the looked for values. An analytical solution can be obtained for the quadratic criterion if the model is linear and there are no constraints, otherwise an iterative method of optimization is used.

2.2.2 Dynamic Matrix Control

Dynamic Matrix Control (DMC) was the first Model Predictive Control (MPC) algorithm introduced in early 1980s. Nowadays, DMC is available in almost all commercial industrial distributed control systems and process simulation software packages. The original work on DMC have been proposed by Cutler and Ramakar (1980). A detailed review on DMC control techniques have been provided by Camacho and Bordons (1999, 2004). DMC control is based on a discrete time step response model that calculates a desired value of the manipulated value that remains unchanged during the next time step. The new value of the manipulated variable is calculated to give the smallest sum of squares error between the set point and the predicted value of the controlled variable. The number of time steps the DMC uses for its prediction is called the "Prediction Horizon".

Prediction:

A brief overview of Dynamic Matrix Control has been given by Chidambaram (2003). The dynamic model used to predict the future values of the controlled variable is represented by a vector, A, whose elements are defined as

$a_i = \frac{\Delta y(t_i)}{\Delta u(t_0)}$

where $\Delta y(t_i) = y(t_i) - y(t_0)$,
$y(t)$ is the value of the controlled variable at time t
$\Delta u(t_0)$ is the change in manipulated variable at t_0. The prediction values along the horizon will be

$$y_k = \sum_{1}^{N}[a_i \Delta u(k-i)] + a_N u(k-N-1) + d(k) \tag{2.1}$$

The present value of disturbance is estimated by the difference between present measurement output and the effects of past inputs is calculated as

$$d(k) = y_{meas}(k) - \sum_{1}^{N}[a_i \Delta u(k-i)] - a_N u(k-N-1) \tag{2.2}$$

Thus the linear estimate of the future output can be written in a matrix notation

$y_{lin} = y_{past} + A\Delta u + d$

where $y_{lin} = [y(k+1), y(k+2), \ldots y(k+p)]^T$ and
$d = [d(k+1), d(k+2), \ldots d(k+p)]^T$

Since future values of d(k+i) are not available, the above estimate is used and it is assumed to the same over the future sampling instants. A more accurate estimate of the d(k+i) is possible, provided the load disturbance is measured and a reliable load disturbance to measured output model is available.

The effects of the known past inputs on the future output is defined by the vector y_{past}. A is the dynamic matrix composed of step response coefficients as explained above. P denotes the length of prediction horizon and M is the moving horizon of the number of future moves $\Delta u(k), \ldots, \Delta u(k+m-1)$ calculated by the DMC algorithm. With these definitions, the future output is predicted for any given vector of future control moves Δu.

For calculating the control inputs the following control objective is used

$$\min_{\Delta u} E \sum_{i=1}^{P} \gamma^2(i)[y_{sp}(k+i) - y_{lin}(k+i)]^2 + \sum_{j=1}^{M} \lambda^2[\Delta u(K+M-j)]^2 \tag{2.3}$$

where γ and λ are time varying weights in the output error and on change in input, respectively. The least square solution for the above problem is given by

$$\Delta u = [A^T \Gamma^T \Gamma A + \Lambda^T \Lambda]^{-1} A^T \Gamma^T \Gamma (y_{sp} - y_{past} - d) \tag{2.4}$$

usually the first calculated Δu is implemented and the calculations are repeated at the next sampling instant.

2.2.3 DMC tuning strategy

Since most of the process are represented by FOPDT models. The tuning method (Shridhar and Cooper, 1997) suggested as below.

1. It is assumed the system is of the form

$$\frac{y(s)}{u(s)} = \frac{K_p}{\tau_p s + 1} e^{-\theta_p s} \tag{2.5}$$

2. With the above transfer function model, first the sampling time is decided by satisfying $T \leq 0.1\tau_p$ and $T \leq 0.5\theta_p$
3. Then the discrete dead time is calculated as $k = \frac{\theta_p}{T} + 1$
4. The prediction horizon and the model horizon as the process settling time in samples is calculated as $P = N = \frac{5\tau_p}{T} + k$
5. The control horizon M is an integer in the range of 1 to 6
6. The move suppression coefficient is given by
$f = 0 \qquad M = 1$
$f = \frac{M}{500}(\frac{3.5\tau_p}{T}) + 2 - \frac{(M-1)}{2} \qquad M > 1$

7. Implement DMC using the traditional step response matrix of the actual process and the following parameters computed in steps 1-5:
   - sample time, T
   - model horizon (process settling time in samples), N
   - prediction horizon (optimization horizon), P
   - control horizon (number of moves), M
   - move suppression coefficient, λ

Tuning of unconstrained SISO DMC is challenging because of the number of adjustable parameters that affect closed-loop performance. Practical limitations often restrict the availability of sample time, T, as a tuning parameter.

Nevertheless moving horizon principle is the widely used technique in real time control.

2.2.4 Principle of moving horizon MPC

An excellent overview of the state of the art on moving horizon based MPC is given by Garcia *et al.* (1989), Camacho and Bordons (2004) and Goodwin *et al.* (2004). Model predictive control systems consists of an estimator and a regulator as illustrated in Figure 2.2. The inputs to the MPC are the target values, r, for the process outputs, z, and the measured process outputs, y. The output from the MPC is the manipulated variables, u.

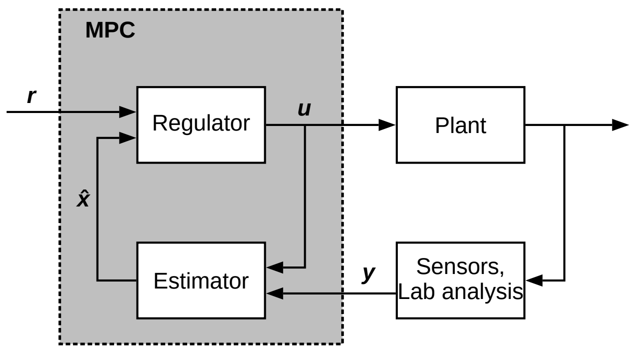


Figure 2.2: Generic model predictive control system.

The principle of moving horizon is given in Figure 2.3. MPC is based on iterative, finite horizon optimization of a plant model. At time t the current plant state is sampled and a cost minimizing control strategy is computed via a numerical minimization algorithm as given in Equation (2.6) for a relatively short time horizon in the future which is called as control horizon N_r. Specifically, an online calculation is used to estimate the projected trajectory over period of prediction horizon N_e and find a cost-minimizing control strategy until the length of control horizon. Only the first step of the control strategy is implemented, then the plant state is sampled again and the calculations are repeated starting from the now current state, yielding a new control and new predicted state path. The prediction horizon keeps shifting forward and for this reason this is called as receding or moving horizon control.

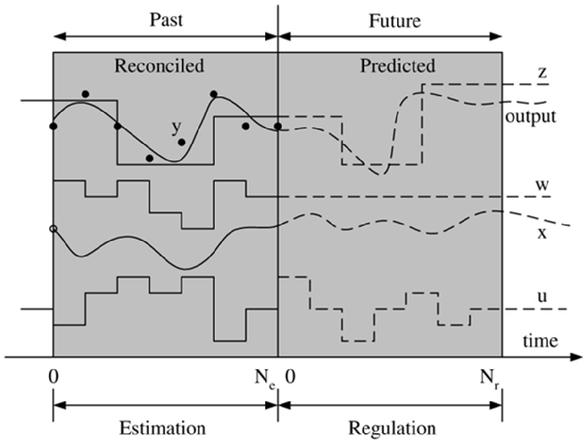


Figure 2.3: Principle of Moving Horizon MPC.

Normally MPCs are equipped with constraints on the manipulated inputs and outputs. Constraints can be of two types: Hard constraints and soft constraints. Hard constraints represent absolute limitations imposed on the system. These names illustrate that hard constraints are to be necessarily satisfied and cannot be violated. Soft constraints only express a preference of some solutions that can be violated and is normally penalized heavily once they are violated. The optimization methods for solving predictive control algorithms are described in Maciejowski (2002).

2.3 Review of MPC in industries

More than 15 years after model predictive control (MPC) appeared in industry as an effective means to deal with multivariable constrained control problems, the approach has been considered as a better way of solving industrial solutions. Bemporad and Morari (1999) have reported a survey on robust predictive control techniques used in industries. The first MPC was based on IDCOM and Dynamic Matrix Control(DMC) way back in 1980's. Then a concept on Generalized Predictive Control(GPC) is introduced. Identification- Command (IDCOM) is based on Model Predictive Heuristic Control (MPHC) commonly known as Model Algorithmic Control (MAC). This method makes use of the truncated step response of the process and provides a simple explicit solution in the absence of constraints.

DMC is much similar to IDCOM where the dynamic matrix is generated from the plant step tests. The identification process begins with understanding the unit objectives and selection of Manipulated variables, controlled variables and disturbance variables. The step tests are conducted to capture data (both numerical and graphical) providing the relationship between controlled variable and the manipulated variable. The unit step response is then used for prediction model.

The basic idea of GPC is to calculate a sequence of future control signals in such a way that it minimizes a multistage cost functions defined over a prediction horizon. The index to be optimized is the expectation of a quadratic function measuring the distance between the predicted system output and some predicted reference sequence over the horizon plus a quadratic function measuring the control effect.

Complete review of methods for solving unconstrained and constrained problems are dealt in Maciejowski (2002). Generally optimization problems are solved numerically assuming a minimization problem until one reaches a minimum. The unconstrained optimization problems can be solved as a least squares problem. The method of least squares is a standard approach to the approximate solution of over-determined systems, i.e. sets of equations in which there are more equations than unknowns. "Least squares" means that the overall solution minimizes the sum of the squares of the errors made in solving every single equation.

The big problem with considering minima is that normally there may be many local minima and the algorithm may stuck in one of the local minimum, unaware of where the global minimum lies. Hence most of the optimization problems are solved as convex problem. A convex optimization problem is one in which because of convexity of the objective function, there is only one minimum or connected to a set of equally good minima. Thus by solving convex problems global minimum is always guaranteed.

Good general books and literatures on optimization are available. Fletcher (1987) and Gill *et al.* (1981) have provided relevant material on optimization algorithms especially for LP and QP methods. A whole book on using convex optimization for control design is given in Boyd and Baratt (1991), however this book only talks about solutions for convex optimization problems but does not deal with predictive control. Convex problems are generally solved using quadratic program(QP). The quadratic program is of the form

$$\min_{\theta} \quad \frac{1}{2}\theta^T \Phi \theta + \phi^T \theta \tag{2.6a}$$

$$s.t. \quad \Omega\theta \leq \omega \tag{2.6b}$$

Here the Φ is Hessian and normally θ is referred as error between the actual and predicted value. If there are no constraints this is clearly convex if Hessian of the objective function has to be positive semi-definite. Since the constraints are linear inequalities, the objective function is a convex quadratic function.

A Linear Program (LP) is the special case of QP where the Hessian $\Phi = 0$ in Equation (2.6), so that the objective function is linear rather than quadratic. It is also convex when Ω and ϕ are such that a minimum exists. In this case the minimum always occurs at a vertex (or possibly an edge). Thus the constrained objective function can be expressed as a convex object with flat surface. This is also known as simplex method. There are large number of standard algorithms available for solving LP, one of the methods is simplex method.

Morari and Lee (1999) have investigated the evolution of controllers in the industries with PIDs being the most commonly used as it is well proven and then the unconstrained control problems. When these controllers unable to handle the complete industrial requirements, knowledge based controls and DMCs are used. But the DMC formulation is completely deterministic and did not include any explicit disturbance model. Then GPC is intended to offer a new adaptive control alternative. But such controllers are not much suitable for multi-variable constrained systems which are more common in industries. Thus the work reviews use of constrained MPCs based on linear MPCs used in industrial applications. Real time applications of constrained MPC become feasible once the formulations are solved either by LP or QP resulting in much faster controller response. In order to improve the stability of such constrained systems works on 'contraction constraints' are suggested.

Miller *et al.* (2000) have presented a case study on control of nonlinear systems subject to constraints. A detailed approach on implementing Lyapunov functions based control applications on nonlinear systems and the applications with laboratory experiments are defined. The effectiveness of the resulting actions are also demonstrated.

Brosilow and Joseph (2002) have proposed economic objectives of using constrained MPC in the industrial applications. In many of the applications it is proposed that when the number of manipulated variables are more when compared with the control variables it is desirable from the economic point of view to have setpoints to the manipulated variables itself. Then a real time optimizer is used to compute the economic target values for both the output and input variables, by including a cost term for inputs in the objective function. But this may cause a steady state error in the system. To avoid such a situation it is proposed to use multi loop control which provides a extra degree of freedom for moving the inputs. In this chapter, in order to analyze the performance of conventional MPC and soft MPC,each of the plant model parameters (gain, time delay, time constant and zero) considered are varied one by one and the controllers are simulated with the perturbed model.

The closed-loop performance of nominal linear model predictive control can be quite poor when the models are uncertain. Consequently, some years after commissioning, many high-level control systems are turned off due to bad closed-loop performance. This is often due to changes in the plant dynamics caused by wear and tear combined with lack of the necessary human resources at the plant to re-tune and maintain the MPC. Model predictive controllers with robust performance against model plant mismatch is therefore crucial in long-term maintenance and success of MPC system. Using soft output constraints in a novel way, the poor performance of predictive control in the case of plant-model mismatch can be improved significantly.

A survey of reviews on model predictive controllers available in the industries is given in Table 2.1

Table 2.1 Review of Model Predictive Controllers in Industries

S.No	Author	Problem	Comments
1	Bemporad and Morari (1999)	Reported survey of robust predictive control techniques in industries. Advantages and disadvantages of difference controllers(DMC, IDCOM, GPC) discussed	Complete review of different MPCs available in industries are discussed.
2	Morari and Lee (1999)	Reviews use of constrained MPCs based on linear MPCs used in industrial applications. Difference between unconstrained MPC and constrained MPC provided. Contraction constraints to improve stability.	Become feasible only if solved through LP or QP.
3.	Miller et al., (2000)	Study of non linear systems subject to constraints. Detailed approach on Lyapunov functions implemented on non-linear systems. Laboratory experiments conducted to study the effectiveness in the control.	Complexity in calculation makes it least preferred in industrial applications.

2.3.1 Review on Tuning of MPC

Garriga and Soroush (2010) have provided a review of tuning guidelines for model predictive control from theoretical and practical perspectives. A detailed review of available methods to tune on DMC, GPC and state space representations and other formulations such as MPL-MPC. General steps involved in tuning for increasing the controller performances are discussed. Based on the formulation of control law the tuning parameters have been discussed. Off-line tuning methods are suggested in which each parameters are individually tuned as given below.

- prediction horizon,
- control horizon,
- model horizon
- Weights on Outputs
- Weights on Rate of Change of Inputs
- Weights on the Magnitude of the Inputs
- Reference Trajectory Parameters
- Constraint Parameters
- Covariance Matrix and Kalman Filter Gain

Also review on auto tuning methods have been provided. The advantage of using an auto tuning is that the control engineer is not required to have a great amount of system knowledge to initialize the tuning procedure. Also tuning parameters are update along with the optimization algorithm and thus they are set to optimal values. Advances in covariance least- squares technology are expected to make Kalman filtering much more accessible by automatically identifying the main tuning parameters.

The challenges in process control and choosing an appropriate control strategy for the respective applications have been discussed by Rhinehart *et al.* (2011). This work is an editorial based on a presentation "Advanced classical or Model predictive control?". Some of the important factors that attribute difficulty in industrial control like constraints, individuality of process, sensors, cause and effect relations, initial capital const are discussed in detail. Also economic benefits of each controllers and usage of different applications like PID, PFC, ADRC, GMC or PMBC for each processes depending on the operating conditions are provided.

The first step involved in tuning is to develop an accurate process model. In all the MPC frameworks development of perfect model will make the tuning procedure much easier as it can be straight forward. If the controller performance is poor then it must be considered the model is poor until proven otherwise.

2.3.2 MPC with Hard constraints

One of the advantages of using MPC over other controllers is it allows operation closer to constraints compared with conventional controls, which leads to more profitable operation. Often these constraints are associated with direct costs, frequently energy costs. For instance, in a manufacturing unit the power consumption must be kept as minimum as possible with same level of production, this is a constraint on manufacturing process. Constraints can be present in both input as well as output. Most commonly the input constraints on the control signals, that is to the process or manipulated variables and rate constraints are hard constraints. This may be because of various reasons like saturation, physical limitations etc., These constraints can never be violated.

For example in Equation (2.6), the inequality constraints

"$\Omega\theta \leq \omega$"

is called hard constraints as the condition needs to be strictly satisfied when calculating the optimization solution. The best example of hard constraints in real time is the high and low limit of the manipulated variables which cannot be varied beyond the limits because of the physical restrictions like vibrations etc.,

Clarke (1988) has proposed a generalized model predictive controller with hard input constraints. The controller is based on the minimization of long-range cost function. The model used for controller is CARIMA model. The closed loop performance of the controller is investigated using simulation.

Iino *et al.* (1993) have proposed a new method by modifying the Generalized predictive control. Firstly, a Kalman filter based predictor is introduced in order to improve the robustness of the predictor against noises. Secondly, a time-dependent weighting factor is introduced into the MPC's quadratic type cost function, in order to improve the transient response characteristics. Thirdly, a parameter tuning method is proposed that adjusts the weighting factors in the cost function considering robust stability of the control system. Finally, the proposed MPC method with and without constraint conditions that are the upper/lower limits and rate limits for both manipulation variables and process control variables, is formulated. The controller is tested in an ethylene plant's dynamic simulator. The models are obtained by simple step tests in the plant. ARMA types of models are used for prediction.

A design method of LQ optimal control law is considered for constrained continuous-time systems by Kojima and Morari (2004). Here the control laws are obtained based on quadratic programming. The control law converges to exact solutions by introducing singular value decomposition for finite-time horizon linear systems. By employing the control problem to a double integrator with constraints it is clarified that the receding horizon control is equivalent to that of the state feedback control where the gain is calculated by a piecewise affine state functions.

Guzman *et al.* (2009) have provided a solution for output tracking problem for uncertain systems subject to input saturation. In order to tackle constraints and modeling errors an external supervisory control method is proposed. Thus a cascade loop with any type of inner control and a GPC for outer loop is considered. A robust constrained Linear Matrix Inequality (LMI) based approach is developed as a solution to control such systems. The existing control loop is first converted into state space representation and LMI is used to provide state space feed back for the inner loop controller. The controller is then tested in an integrator plant with delay with a inner loop PI controller. The inner control loop is studied with PI controller in the presence of uncertainties and it is found that stability problems occur. Then the controller is included with the GPC for controlling the inner loop considering input saturation and it is found that the performance results also ensuring constrained robust stability.

A survey of some of the reported work on the model predictive controllers with hard constraints relevant to the present work is given in Table 2.2

Table 2.2 Reported work on design and implementation of MPC with hard constraints

S.No	Author	Problem	Comments
1	Clarke and Tsang (1988)	Generalised Predictive Control with hard input constraints. Minimize long range cost function. Modeling the GPC - CARIMA model.	Hard constrained MPCs may sometime lead to input saturation that is not desirable.
2	Iino et al., (1993)	A new input/output MPC with frequency domain technique and its application to ethylene plant. MPC with hard input and output constraints Techniques - DMC and MAC. ARMA type of model for prediction.	Hard constrained MPCs may sometime lead to input saturation that is not desirable.
3.	Kojima and Morari (2004)	Design method of LQ optimal control law for constrained continuous time domain Systems. Control laws based on QP. Control problem a double integrator with constraints.	Receding horizon principle used. Results verify the control equivalent to state feed back with gain calculated by piecewise affine state functions
4	Guzman et al., (2009)	A robust constrained reference governor approach using linear matrix inequalities . Two different loop controllers. An outer Loop MPC control with Linear Matrix Inequalities and local control loop for maintaining the variables.	Solve a set of constraints described by LMI and BMI complex to be extended for constraints other than input constraints.

2.3.3 Soft Constrained MPC

In many of the constraint control problems the controller solutions become infeasible because of the hard constraint violation problems which may be result of various factors in real time. State and output constraints may lead to infeasibility of the optimization problem. For example, an output disturbance may push the output out the feasible region such that no feasible input trajectory is able to bring in back in the constraint region. In that case the hard output constraints can no longer be maintained. The soft constrained MPC explained above mostly take care of the effects due to the disturbances of the system.

One systematic strategy for dealing with infeasibility is to soften the constraints. That is, rather than regard the constraints as hard boundaries which can never be crossed, to allow them to be crossed occasionally but only if necessary. Usually input constraints are hard constraints and there is no way in which they can be softened, like actuator limits. Maciejowski (2002) has provided a detailed explanation of how to use soft constraints in the optimization problem. A possible way to proceed is to discard the output constraints that cause the infeasibility.

A more subtle way that tries to bring the output back in the feasible region is the use of a slack variable ϵ.

The slack variable is introduced to relax the constraints that cause the infeasiblity according to the normal hard constraint formulation $Ax.b + \epsilon$. Additionally, a term $\epsilon^T Q_3 \epsilon$ is added to the cost function where $Q_3 > 0$ is some positive definite weighting matrix. The slack variables are treated as free variables and are optimized such that on the one hand the infeasible constraints are relaxed and on the other hand the constraint violation is minimized. The optimization problem remains a quadratic programming problem because the new variables ϵ are introduced quadratically in the cost function and linearly in the constraints.

Scokaert and Rawlings (1999) have proposed a state based soft constraint approach for handling the infeasibility with respect to conventional predictive control approach. Two types of solutions are provided for such systems, a minimal time approach and a soft constraint approach. It is illustrated by using a non- minimum phase system that state constraints will be included when the solution becomes infeasible. In the first approach the control algorithm identifies the smaller time in which the state constraint can be satisfied and in the second method a soft constraint approach where the controller penalizes for state constraints violation. The closed loop and open loop responses of both the approaches are investigated. It is found that in both the cases the controller performance are close to Pareto optimal. Also soft-constraint MPC formulations are nominally exponentially stabilizing and asymptotically stabilizing under decaying perturbations. The main drawback of such controllers are they require larger QP solutions resulting in slower response which may not be feasible in real time.

Kerrigan and Maciejowski (2000) have proposed a MPC method, in which the hard constrained MPC is converted into soft constrained by introducing a slack variable on the states. A slack variable is included in the objective function with moving horizon principle when the controller solutions become infeasible during the hard constraint approach. The MPC problem is treated as a multi-parametric quadratic program (mp-QP) and exact penalty functions are introduced in order to find a condition on the lower bound for the violation weight. By introducing slack variables the non-smooth, exact penalty function can be converted into a smooth, soft-constrained QP problem. It is shown from examples that if the constraint violation weight that is used in the soft-constrained cost function is larger than the maximum norm, the solution is guaranteed to be equal to the hard-constrained solution for all feasible conditions that were considered.

Bemporad *et al.* (2002) have proposed a feedback control law to minimize a quadratic performance criterion. Here the control law is piecewise linear and continuous for both the finite horizon problem with model predictive control and infinite time measure for constrained linear regulation. A LQ regulator algorithm is developed with hard constraints. But in order to avoid feasibility problem output constraints are softened /relaxed. The stability of the controller is analyzed and found satisfactory as MPC. One advantage of such technique is that it can be implemented without any online computations, but cannot be applied for large scale applications.

Another method on range control explained by Roubal and Havlena (2005) and Havlena and Lu (2005) provide a soft limit band for the output where the controller does not react for change in output within the region. The basic idea of range control approach is to replace the set point of controlled variable Y by a set range which is defined by the sequence of low and high limits $\hat{y}_L$ and $\hat{y}_H$ as given in Figure 2.4. Then the optimality criterion can be expressed as a quadratic programming

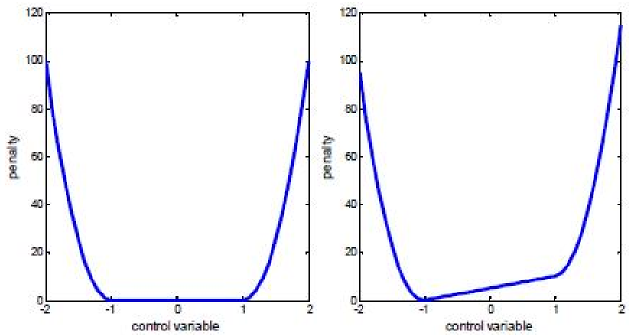


Figure 2.4: Penalty function and modified penalty function of Range control, $\hat{y}_L = -1$ $\hat{y}_H = 1$.

problem as given by Equation 2.7.

$$\min_{\theta} \quad \frac{1}{2} \|\hat{y}_k - S\hat{u}_k - \hat{w}_k\|^2_{Q_y} + \|\Delta\hat{u}\|^2_{Q_u} \tag{2.7a}$$

$$s.t. \quad \hat{y}_L \leq \hat{w}_k \leq \hat{y}_H \tag{2.7b}$$

Thus the controller acts only when the controlled variables are beyond the range of constraints. In order to avoid steady state offset because of no action within the band the controller is solved by least squares solution whenever the controlled variables are within the band. The controller is implemented in distributed parameters system which is described by a linear two-dimensional (dependent on two spatial directions) parabolic partial differential equation.This partial differential equation is transformed to the discrete state space description using the finite difference approximation. A model with a large dimension is obtained and has to be reduced for an advanced control design. The balanced truncation method is used for the model dimension reduction. The challenges faced when using range control methods are discussed and such methods always leave offset like a normal dead band controller. So a modified penalty function is included. Most of the soft constraints are for state constraints where the internal state of the system gets disturbed because of the various disturbances A detailed study on the real time implementation of the control method in industries is made.

A survey of important work on the model predictive controllers with soft constraints is given in Table 2.3

Table 2.3 Reported work on design of soft constraint based Model Predictive Controllers

S.No	Author	Problem	Comments
1	Scokaert and Rawlings (1999)	State soft constraint based MPC. For achieving feasibility two approaches are provided. 1. Minimal time approach 2. Soft constraints approach. A minimal time approach and soft-constraints approach are provided. A penalty term for compensating the violations.	Only if the model of the system reflects the state perfectly. Require larger QP solutions resulting in slower response which may not be feasible in real time.
2	Kerrigan and Maciejowski (2000)	Soft constraints based on the range of penalty function. Behaves as soft constrained MPC and hard constrained MPC on constrained violations. MPC to discrete-time LTI state space model, soft constraint as slack variable	QP dependent on both current state, previous control input and reference trajectory. This may not be feasible all solutions.
3	Bemporad, Morari et al., (2002)	A LQ regulator algorithm with hard constraints. To avoid feasibility problem hard output constraints are softened or relaxed. A piecewise linear solution is provided for the linear QP. The feasibility of the control mainly depends on initial state.	Cannot replace the large scale MPC as it cannot handle all the situations satisfactorily.

Table 2.3 (contd..)

4	Roubal and Havlena (2005)	The control objective include simple range control objective along with other hard constraints. The controlled variable setpoint replaced by sequence of low and high limits.	A dead band controller based on QP. Can leave offset as there is no action for outputs within the band
5	Havlena and Lu (2005)	Range control for disturbance adsorption problem. A penalty function for the outputs within a specified range. Implemented in distributed parameter system described by two dimensional parabolic partial differential equation.	Power and chemical/ refining industries need such integration for providing long term production and energy efficiencies.

2.3.4 Robust Model Predictive controllers

A controller is said to be robust which are designed for a particular set of parameters if it would also work well with different sets of parameters. Thus a robust controller is one which can handle systems with uncertainties or disturbances effectively. Robust methods aim to achieve robust performance and/or stability in the presence of bounded modeling errors.

Warren and Marlin (2004) have proposed controller solves a second-order cone program (SOCP) at each execution in order to determine the set of control moves that will optimize the expected performance of the closed-loop system while maintaining the uncertain process outputs and inputs within their allowable bounds. The proposed formulation uses a probabilistic, closed-loop description of system uncertainty that is calculated off-line. On-line, the MPC requires the solution of a convex second-order cone program that can be efficiently solved with existing interior-point algorithms. The controller discussed avoids this limiting assumption while maintaining robust output constraint handling. The process is assumed to be linear time-invariant (LTI) within the prediction horizon. The controller deals with plant-model uncertainty by replacing deterministic constraints. Here a method for effectively handling probabilistic input constraints is proposed. Also the system uncertainty is split into several uncertainty regions. The uncertainty associated with each region is smaller than the total system uncertainty, allowing each subset to approach the input constraint more quickly. This reduces the " back-off" caused by the probabilistic input constraints. Case studies illustrate the improved dynamic performance of the multi-region method. The experimental results are verified using a 1st order, isothermal CSTR along with modeling error. From experimental methods it is proven that the robust controller can handle uncertainties quite effectively.

Kassmann *et al.* (2000) have presented a new formulation of the steady-state target calculation that explicitly accounts for model uncertainty. When model uncertainty is incorporated, the linear program associated with the steady-state target calculation can be recast as a second order cone program. Also the advantages of primal-dual interior-point methods in the resulting structure is discussed. The mathematical parameterizations of possible plants is known as the uncertainty description. It can take many different forms and can be given parametrically or statistically. A heavy oil fractionator with 2×2 model is taken for simulation comparison. The closed loop comparison is made between nominal LP and robust LP. This SOCP can greatly improve control by rigorously accounting for modeling uncertainty. The resulting SOCP structure can be exploited to develop efficient numerical solutions based on primal-dual interior-point methods.

A survey of some of the important work on robust model predictive controllers available for handling uncertainties is given in Table 2.4

Table 2.4 Reported work on Robust MPC based on Second Order Cone Programming Technique

S.No	Author	Problem	Comments
1	Warren and Marlin (2004)	SOCP based MPC for handling uncertainty has been proposed. Deals with probabilistic output constraints solved in offline. SOCP solved using efficient interior-point algorithms. A 1^{st} order , isothermal CSTR considered along with modeling error. Case studies illustrate the improved dynamic performance of the multi-region method	A robust MPC using SOCP implemented by simulation and need to be modified to match the industrial approach.
2	Kassmann and Badgwell (2004)	The MPC algorithm is steady-state target for accounting model uncertainty using SOCP calculation followed by dynamic optimization. A heavy oil fractionator with 2 x 2 model taken for simulation comparison.	SOCP can greatly improve control by rigorously accounting for modeling uncertainty.

2.4 Ball Mill Control

Researchers have been working in optimizing the ball mill grinding process along with the separation process. Various studies have been done for analyzing the dynamics of the ball milling circuit. The electrical energy consumed in the cement production is approximately 90 kWh/tonne. 30% of the electrical energy is used for raw material crushing and grinding while around 40% of this energy is consumed for grinding clinker to cement powder as given by Fujimoto (1993); Jankovic *et al.* (2004). Global cement production use approximately 2% of the worlds primary energy consumption and 5% of the total industrial energy consumption as reviewed by Concil (1995) and Austin *et al.* (1984). Also ball mill is one of the difficult

Rajamani and Herbst (1991*a*) have proposed feedback and optimal control methods for optimizing ball mill grinding circuits. Here the ball mill considered was an semi-autogenous mill which is a wet method of grinding compared to normal dry grinding circuits. Rajamani and Herbst (1991*b*); Herbst *et al.* (1992) have developed dynamic and simplified models of the cement mill circuit. Based on these modeling, two PI controllers were tuned and an optimal control was designed. The model was developed by introducing step changes in the real time system. The parameters considered are online particle size distribution with respect to fresh feed rate and sump level. First a feedback control based on maximization principle is designed for increasing the feed rate and an optimal control based on minimizing the integral square error was developed. Both the controllers were tested first in simulator and then in real plant. From the real time results it was shown that the settling time of optimal controller was around 6 min when compared with feed back controller (26 min). Also the Feed back control exhibited oscillations with an ISE value of particle size is 16.6 whereas optimal control had an ISE value of 8.5 which is almost half that of the Feed back control.

Van Breusegem *et al.* (1994) and Van Breusegem *et al.* (1996*a*) have proposed a model based algorithm for the regulation of cement mill circuit. The controller was based on Linear Quadratic algorithm. Here a black-box model was developed based on the experimental data collected from the simulation. The dynamic model in the form of four first order differential equations were converted into state space to be used in LQ control. Integral actions were added for making the control offset free. A dynamic simulator was developed based on the models and the controller performance was compared with PI controller and from the results it was shown that the standard deviation of quality had been reduced and also the variation in tailings were also less.

Boulvin *et al.* (2003) have developed a grey box model with algebraic and partial differential equation with unknown parameters for cement grinding process . Through experimental data the unknown parameters were estimated and a dynamic simulator was developed to analyze the control and process behavior in real time operation. Two controllers PI and LQ control developed by Van Breusegem *et al.* (1994) were compared with the simulated model. Then a cascaded control for regulating mill flow rate and simple PI control for fineness control was applied. Incase of online measurement of recirculation flow a Feed forward controller was proposed. Here the feed-forward provide best results in terms of mill flow regulation. But since the difficulty in online measurement cascade control was considered as best solution.

Ramasamy *et al.* (2005) have developed input/output models through step response tests in simulator. Multi loop PI was designed and decoupled to account for interaction between the control loops. The de-tuning factor was based on IAE of output variables. The model predictive control from MATLAB was compared with PI and from the results it was shown that the MPC achieves better decoupling. Then constraints were included in the model using MATLAB toolbox. From the results it was confirmed that the constrained MPC provides sluggish operation than unconstrained but it provides a much better decoupling. Also the MPC was compared with decoupled PI controllers and the results showed that PI controller results in more oscillations and longer settling time.

Conventionally, the grinding circuits are controlled by multi-loop PID controllers, but these controllers generally have drawbacks, such as input/output pairing problems and hard tuning work. Chen *et al.* (2009) have provided an adaptive DMC for handling ball mill grinding circuit with multiple model developed based on three different SOPDT transfer function models. The 3×3 MIMO transfer function system was developed using step response test. A simple DMC objective function was considered and the models were switched depending on the input variation ranges. The performance of DMC compared with PI and it was found that the quality variations were reduced by 3%

Chen *et al.* (2008) have developed constrained DMC for controlling ball mill process. The real time implementation of 3×3 MIMO transfer function model based on step response tests. In addition to normal DMC objective function, hard constraints on manipulated variable, rate of change of manipulated variable and controlled variable were included. The prediction horizon was selected as P = 20 and control Horizon as M = 5. From the results it was shown that the online measurement of quality is improved from 88% to 95%.

Martin and McGarel (2001) have proposed Nonlinear MPC for controlling the real time cement mill circuit. Here the NMPC process gains were calculated based on model from neural networks. Data from plant was obtained to train the neural network model and this provide non-linear process gain for NMPC. Also maximum and minimum gains were also calculated. Hard constraints on manipulated variable and rate of change in manipulated variables were also included. From the results it was reported with such type of control the product quality consistency was close to 95%.

A summary of reported work on Survey on model predictive controllers available for control of ball mill circuits is given in Table 2.5

Table 2.5 Reported work on Model Predictive Controllers for ball milling processes

S.No	Author	Problem	Comments
1	Rajamani and Herbst (1991a)	Feedback and optimal control methods for optimizing ball mill grinding circuits.Semi-autogenous mill considered for control. Models developed by Rajamani and Herbst (1991a) used. Optimal control based on minimization of ISE tested with Feedback control based on maximization principle.	Optimal control perform better than Feedback control. Constraints not considered in the controllers.
2	Van Breusegem et al. (1994) and Van Breusegem et al. (1996a)	Model based algorithm for the regulation of cement mill circuit. Controller based on LQ algorithm. Black-box model based on simulation data reduced to first order differential equations.	Controller compared with PI control and standard deviation in quality is reduced.
3	Boulvin et al. (2003)	Agrey box model with algebraic and partial differential equation with unknown parameters for cement grinding process. Unknown parameters estimated through simulation data. PI and LQ controllers compared with simulated model.	feed-forward provide best results in terms of mill flow regulation. Diffculty in online measurement, so cascade control considered as best solution.

Table 2.5 (contd..)

4	Ramasamy et al. (2005)	Developed input/output models through step response tests in simulator. Multi loop PI designed and decoupled. MPC compared with PI controller in MATLAB toolbox .	Constrained MPC sluggish in operation but provides much better decoupling. PI controller results in more oscillations.
5	Chen et al. (2009)	An adaptive DMC for handling ball mill grinding circuit with multiple model developed based on three different SOPDT transfer function models. This avoid input/output pairing problem.	Unconstrained DMC is better than PI control but cannot be used interms of constraints and uncertainties.
6.	Chen et al. (2008)	Constrained DMC for controlling ball mill process. real time implementation of 3×3 MIMO transfer function model. Hard constraints on MVs, rate of change in MVs and CVs along with objective function	Results compared with unconstrained DMC and found to increase the standard deviation in quality variations. Can be implemented only with online measurement of quality.
7.	Martin and McGarel (2001)	Nonlinear MPC for controlling the real time cement mill circuit. NMPC process gains calculated based on model from neural networks. Data from plant obtained to train the neural network model and this provide non-linear process gain for NMPC. . Hard constraints on MVs, rate of change in MVs included.	Quality consistency improved with such controllers. But difficult to be used in real time because of computational complexity and modeling

2.4.1 Cement Mill modeling

The main problem with such control techniques are that the controller performance degrades with plant-model mismatch. Researches have been working in the area of modeling to improve the performance of such controls.

The closed loop ball mill-classifier grinding circuit has been described by lumped model (Benzer *et al.*, 2001). Here the lumped model specifically includes the mill load (amount of material inside the mill) as a state variable. Another way of modeling the cement milling circuit is based on the input/output characteristics of the milling process which can be considered as a black-box model as presented by Lepore *et al.* (2002, 2003, 2004, 2007*a*,*b*) in their various works.

Discrete-element model techniques have been developed by Cleary (2006), Powell and McBride (2006) and Jayasundara *et al.* (2008) which can be used to provide information on the radial dynamics of the materials and also the axial distribution of the different particle sizes within the discharging ball mill. This facilitates the clear understanding on the estimation of power draw and of liner wear inside the ball mill.

Neural network is one of the methodology used for determining the black-box model, the cement mill modeling based on neural networks has been developed by Martin and McGarel (2001) and Topalov and Kaynak (2004). The main advantage of using black-box model is the occurrence of non-linear behavior with the input/output parameters considered for modeling. The non-linear behavior can be made just complex enough for the description of the main process dynamics while remaining tractable for control design as given by Huusom *et al.* (2005), Efe and Kaynak (2002) and Grognard *et al.* (2001). But these models are quite difficult to be implemented in the real time systems, because of the complexity in design and difficulty to be coupled with control algorithm. Thus it is required to reduce the models to lower order to be used in control applications which will result in loss of important dynamics in the system.

2.5 Fuzzy Logic Controller in Cement Industries

In general, the most widely used technique in cement industries for optimization is based on Fuzzy Logic. The control rules of the fuzzy controller are obtained from the knowledge of the operators. These types of controllers are used in cement plant widely because they do not require mathematical/ empirical model of the plant and can be easily configured even for non-linear systems.

ZimmerMann (1996) has given an overview of current applications of Fuzzy control to real world problems. A brief discussion of Fuzzy control systems for rotary cement kiln has been provided. The main problem in mathematical modeling control strategy is that the relationships between input and output variables are complex, nonlinear. The control variables chosen are exhaust gas temperature, burning zone temperature, oxygen percentage and liter weight. The manipulated variables are coal feed rate, kiln fuel and induced draught fan speed. The aim of the kiln control is to automate the routine control strategy of an experienced operator. After discussions with operators, 75 operating conditions as fuzzy are defined as drastically low, low, slightly low, ok, slightly high, high and drastically high. Fuzzy rules are written to change the drive load, fan speed and fuel for maintaining the above control variables.

Cao *et al.* (2008) have proposed a high precision sampling fuzzy logic controller with self-optimizing to the cement ball mill circuit. This controller, based on fuzzy control strategy, improves the control precision by a fuzzy interpolation algorithm. The fuzzy logic controller and the self optimizing algorithm along with sampling and interpolation algorithm are implemented in a MATLAB simulator with a second order plus dead-time model which demonstrates the ball mill circuit. Fuzzy rules are altered to make the control parameter reach steady state without steady state error. In addition, the high precision sampling fuzzy logic controller has been put into practice in the clinker cement production workshop of a cement mill. The rules are altered as per real time and it is found that the power consumption is decreased and the particle size distribution becomes standard.

Wardana (2004) has proposed a Fuzzy-PID controller for maintaining the under grate pressure by varying the grate speed. Here the classic representation of Mamdani logic operations are applied, a 7×7 rule of fuzzy algorithm and centre of area for defuzzification are implemented. The performance of the FLC is compared with conventional PID controller. It has been found that the standard deviation of under grate pressure has been reduced from 50 mmH_2O to 5 mmH_2O and also the temperature of clinker output is reduce from around 150^{o}C to around 90^{o}C.

Lin and Chin (1996) have proposed an application of fuzzy logic inference technique on a cement grinding roller control. The control of cement grinding roller is that the oil-pressure is commanded to follow a desired setting pressure. A neural network scheme is applied to identify the system model for establishing the simulation program for evaluating the derived control algorithm and the fuzzy rules are proposed to infer the desired setting pressure to replace the original PI-like method. The controller is applied in real time cement plant and it is found that the stability of operation has been improved significantly with the proposed control technique.

The Fuzzy Logic Controller used for comparing the performance of soft MPC is a commercial package available in FLSmidth's Process EXpert system(Automation (2008)). It has been widely used in industries for controlling various cement plant applications over the years. The controller is already available in the cement plant and has been used directly for comparing the soft MPC.

From the above literatures it can be viewed that the control strategies used for cement mill circuit are based on the models obtained from the plant and there is no detailed study on plant-model parameter mismatch. Normally, cement circuits are characterized by large uncertainties because of input material variations and mechanical wear and tear over the period of time. Also the controllers consider only online measurements and infer the quality parameters for control, which is quite uncommon in real plant. This will affect the success of such controllers in real time.The work on robust MPCs provided in the literatures have not been tried in controlling real time cement mill circuit. Thus there is a need for design of robust MPC and develop MPC based on soft constraints to address the uncertainties present in the cement mill.

CHAPTER 3

An evaluation of existing MPC tools

In this chapter, the effect of uncertainties on an unconstrained Dynamic matrix Controller similar to the one proposed by Cutler and Ramakar (1980) is investigated. Prasath and Jorgensen (2008) have proposed a predictive controller based on FIR models with 'ℓ_2' regression norm, including the regularization weights (R and S)(denoted as regularized l_2 finite impulse response (FIR) predictive controller) along with input and input-rate constraints. Feedback from estimator is based on a simple constant output disturbance filter. Prasath and Jørgensen (2009) have proposed a moving horizon constrained regularized ℓ_2 estimator (MHE) based on finite impulse response models (FIR). The performance of the controller with both the simple estimator and Moving horizon estimator in the face of plant-model mismatch is simulated using a SOPDT with a zero transfer function model. Then the work by GuruPrasath and Jorgensen (2009) on the controller equipped with soft output constraints that are used to have robustness against model plant mismatch is investigated. The simulations can be used to benchmark ℓ_2 MPC against FIR based robust MPC as well as to estimate the maximum performance improvements by robust MPC. The bench mark study includes

- The performance of the controllers in the presence of uncertainties in the system and the disturbances in the system
- Handling the infeasibility involved in solving the constraint problem.

3.1 Introduction

Dynamic Matrix Control (DMC) was the first Model Predictive Control (MPC) algorithm introduced in early 1980s. Nowadays, DMC is available in almost all commercial industrial distributed control systems and process simulation software packages.

DMC based on step response coefficients for blast furnace has been dealt by Cutler and Ramakar (1980). The DMC is represented with a set of numerical co-efficients based on step test. The technique is conjunction with minimizing the integral of error/time curve. The controller developed is compared with existing PID controller in the plant for handling dead time. Future moves on actuator and the corresponding future measurements are predicted using a method similar to moving horizon principle. Here the step response co-efficients are used to update the dynamic matrix and the feed back is taken from real data, but not based on estimation.

In Prasath and Jorgensen (2008), FIR based MPC is developed for industrial control purpose (especially with cement plant application) where the system includes both process and measurement noise, the analysis on performance of the controllers in a stochastic system is made. Marafloti *et al.* (2010) have proposed a recursive least squares based persistently exciting MPC by referring this work on FIR based MPC.

The effect of uncertain models on the performance of a regularized ℓ_2 model predictive controller with input constraints, input-rate constraints and soft output constraints have been investigated by Maciejowski (2002), Goodwin *et al.* (2005) and Qin and Badgwell (2003).

In contrast to state space parameterizations, the FIR model is in a form that can easily be applied in robust predictive control, i.e. predictive control based on robust linear programming or second-order cone programming, the work on such programming have been explained by Hansson (2000) and Vandenberghe *et al.* (2002). The predictive control based on second-order cone programming have been investigated by Ben-Tal and Nemirovski (2001) and Boyd and Vandenberghe (2004).

The controller performance can be improved by using a FIR based moving horizon estimator where the unmeasured disturbances are estimated based on the measured outputs and fed back into the controller.

The advantage of FIR parametrization is that it is both linear in parameters (impulse response coefficients) and the decision variables. In the regulation problem, decision variables are the manipulated variables, while the decision variables in the estimation problem are the known process disturbances. In addition, the finite impulse response parametrization also yields a band structured Hessian in the resulting quadratic program that can be used for its efficient solution.

Based on linear models Muske and Rawlings (1993*b*) have used a moving horizon estimator and used input or output disturbances to have a steady state offset free control. Robertson and Lee (1995) and Robertson *et al.* (1996) have presented moving horizon estimators for nonlinear systems and relate the optimization based approach of moving horizon estimation to a probabilistic state estimation approach. Rao *et al.* (2001) have proposed sufficient conditions for the stability of moving horizon state estimation with linear models subject to constraints on the estimate. Rao and Rawlings (2002) have illustrated using a series of example monitoring problems, the practical advantages of MHE by demonstrating how the addition of constraints can improve and simplify the process monitoring problem.

All these methods cited above are based on state space approach. In Prasath and Jørgensen (2009), a method based on convolution approach is adopted for the estimation of unknown disturbances from the measurement outputs. The estimator produces non-smooth disturbance estimates upon which the regulator reacts and introduces real output variations by its manipulation of the process inputs. The simulations are used to critically address the performance limitations in case of measurement noise.

In most of the literatures reported,the soft constraints are included in the system when the hard constrained optimization problem has no solution because of either process conditions or disturbances in the system. The soft constraints are introduced as a slack variable and they become active only if the original hard constrained solution becomes infeasible. In GuruPrasath and Jorgensen (2009), the slack variables on soft constraints are included in the regular objective function and the optimization is solved in real time. Here the slack variables are used as dead band across the reference variable, which improves the robustness of the controller by providing two different solutions one within the dead-band and the other outside the band. This technique is similar to the technique that have been proposed by Honeywell technologies in RMPC (Qin and Badgwell, 2003; Havlena and Lu, 2005; Havlena and Findejs, 2005) where a funnel type objective function is solved. Compared to classical control, the use of soft constraints has some similarities to PID control with dead zone as given by Shinskey (1988).

In this chapter, initially the performance of the controllers are simulated using a generic SOPDT with zero transfer function model of the system for SISO case. MIMO simulation system the model is obtained from the available cement mill circuit simulation for which the details are given in later chapters.

3.2 Dynamic Matrix Control(DMC)

The major elements of DMC are (i) the model, (ii) estimation of the disturbance and projection into the future and (iii) computation of control inputs. The model used is a discrete truncated step response model. The basic formulation of DMC has been given by Camacho and Bordons (1999). The DMC reported in section 2.2.2 is implemented on a FOPDT transfer function.

Based on the tuning methods suggested, the control parameters formulated for the DMC are Sampling Time - 0.1, Prediction Horizon - 20, Moving Horizon - 10, Simulation Length - 200, the smoothing factor is varied between 0.0 and 0.7 to smoothen the prediction values based on moving average and to analyze the performance of the controller for different ranges of smoothing factors. The smoothing factors are used to smoothen the prediction, based on the moving average principle and two values of smoothing factor $\alpha = 0$ and $\alpha = 0.7$ are used to provide variations in the prediction.

First the performance of the DMC is analyzed by tuning the control weights $\lambda = 0.1, 1, 10, 100, 1000$ as given in Figure 3.2. This is done to find the optimum

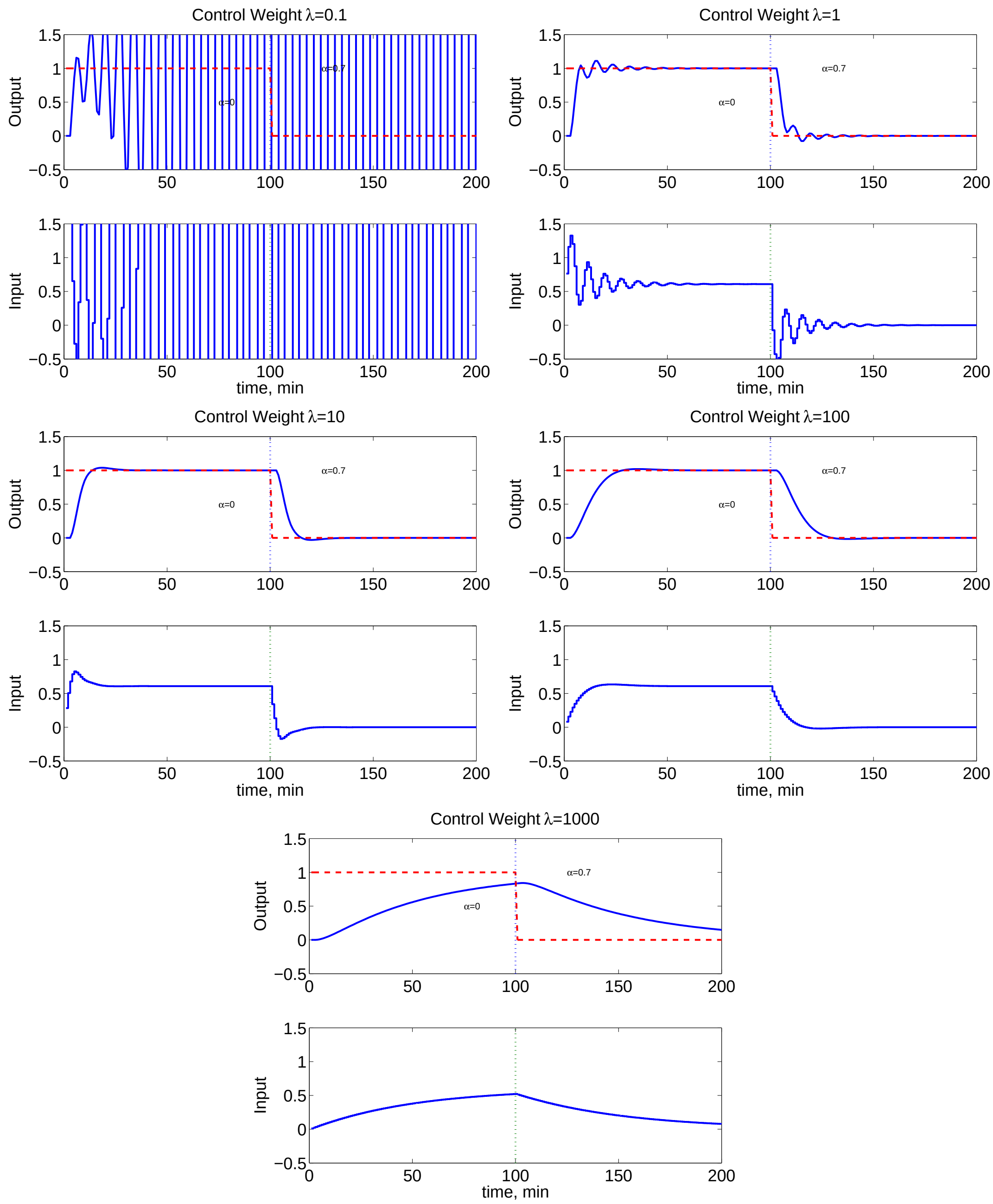


Figure 3.1: Performance of Dynamic Matrix controller for step responses with different control weights $\lambda = 0.1, 1, 10, 100, 1000$.

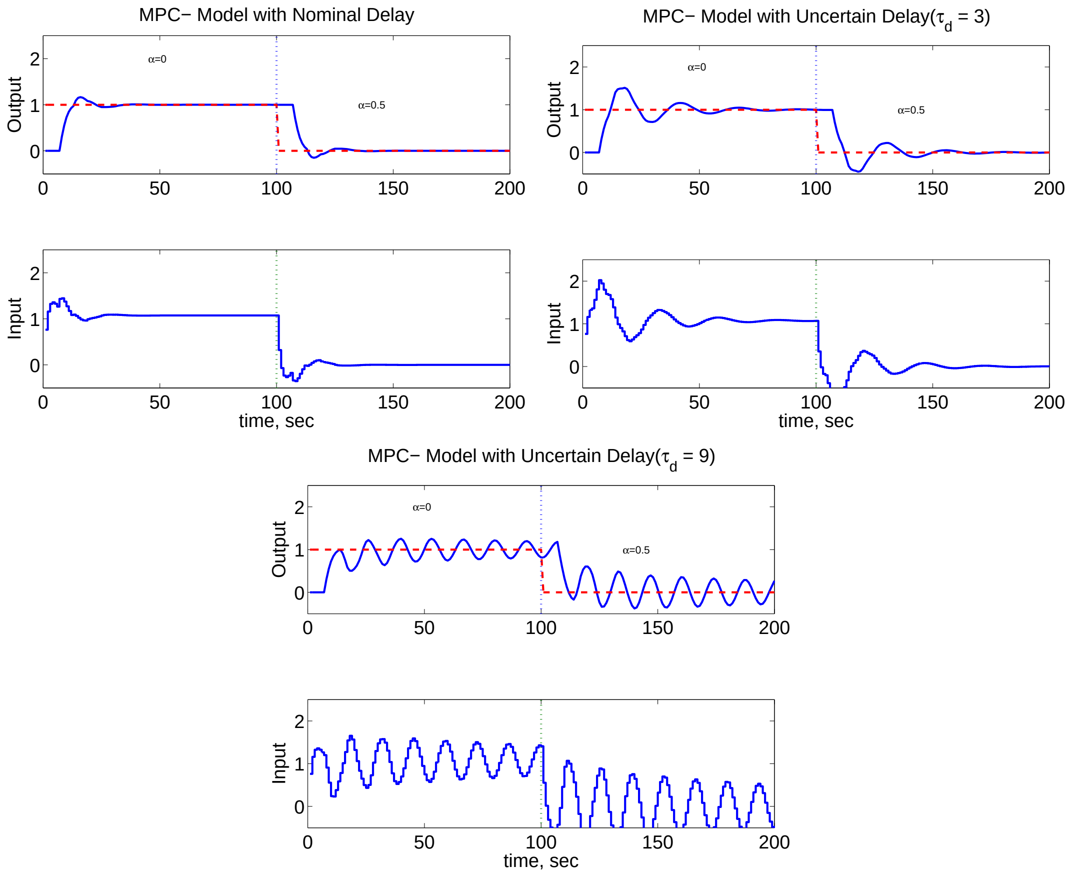


Figure 3.2: Performance of the DMC with uncertainty in delay with control weight $\lambda = 10$.

weight for nominal model and the same weight is used for analysis of performance of the controller during plant-model mismatch. Here the DMC is made online in a closed loop based on the state space model obtained from the FOPDT transfer function as in Equation 3.1.

$$G(s) = \frac{1}{(5s+1)}e^{-5s} \tag{3.1}$$

The step response coefficients for the controller are also extracted from the same transfer function (Equation 3.1).

From Figure 3.1, it can be observed that when the control weight is 0.1 the controller is quite unstable as there is no restriction in input moves resulting in huge variations in the output, and when the control weight is increased the input moves of the controller are restricted and thus reduces the oscillations in the output. But as the control weights are extremely high the control moves become sluggish and the output never settles when the control weight is 1000. Thus from the Figure 3.1 it is clear that when the optimum control weight for smooth performance of DMC with the FOPDT model is $\lambda = 10$.

The performance of the DMC is then studied for the system with model uncertainty. As a case study, the DMC is tested with models with time delay uncertainty. Figure 3.2 shows the performance of the controller when the time delay of plant model is same as nominal model for optimum tuning conditions. When the time delay of plant model is different from the controller model the behavior of the MPC is seen in the Figure 3.2. For nominal model the settling time is around $60s$. When the time delay of plant model is higher than the nominal model the performance of controller completely deteriorates and the control actions become oscillatory and the MPC becomes unstable when the plant delay becomes almost double the time delay in nominal model. Although DMCs are quite simpler in approach and less complex, since the extension of DMC cannot be used for robust programming techniques and does not have constraints included in the controller, the Model predictive control based on moving horizon principle using FIR based models is proposed.

3.3 FIR Model Based MPC

In this section, model predictive control based on Finite Impulse Response (FIR) models is derived and converted into standard quadratic form. The advantage of using FIR based model is that it can be applied for getting models from applications with missing observations or lab measurements and also can be easily used with higher order models. FIR models can only be applied to stable system. Here the MPC with FIR based model is formulated assuming the system is linear. The cost function used is a quadratic function on measurement error and the input-rate variations with hard constraints on input and input-rate in the form of linear inequalities. Also it is considered that the controller and the system are time-invariant. FIR models normally have the advantage that they can be easily used for parameterizing system with missing observations. Also FIR models are directly estimated from plant test data. In addition, FIR model structure requires lesser prior information than the transfer function models, like FIR models need only information of settling time in contrast to the other models where the information about time delay, model order are required.

3.3.1 Plant and Sensors

The plant considered here is assumed to be a linear state space system

$$\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + Bu_k + B_d d_k + G\boldsymbol{w}_k \tag{3.2a}$$

$$\boldsymbol{z}_k = C\boldsymbol{x}_k \tag{3.2b}$$

with x being the states, u being the manipulated variables (MVs), d being unmeasured disturbances, and w being stochastic process noise. z denotes the controlled variables (CVs). The measured outputs, y, are the controlled outputs, z, corrupted by measurement noise, v. Consequently

$$\boldsymbol{y}_k = \boldsymbol{z}_k + \boldsymbol{v}_k \tag{3.3}$$

The initial state, the process noise, and the measurement noise are assumed to be normally distributed stochastic vectors

$$\boldsymbol{x}_0 \sim N(\bar{x}_0, P_0) \tag{3.4a}$$

$$\boldsymbol{w}_k \sim N_{iid}(0, Q) \tag{3.4b}$$

$$\boldsymbol{v}_k \sim N_{iid}(0, R) \tag{3.4c}$$

The measured output, y, is the signal available for feedback and used by the estimator. u is the signal generated by the control system and implemented on the plant. The subscript 'iid' stands for independent and identically distributed (Shao, 2003), it is used to represent the variance of process and measurement noise.

3.3.2 Regulator

Stable processes can be represented by the finite impulse response (FIR) model

$$z_k = b_k + \sum_{i=1}^{n} H_i u_{k-i} \tag{3.5}$$

in which $\{H_i\}_{i=1}^n$ are the impulse response coefficients (Markov parameters). b_k is a bias term generated by the estimator. b_k accounts for discrepancies between the predicted output and the actual output. The output predictions used by the regulator are based on the FIR model (Equation 3.5). Consequently, using the FIR model in Equation 3.5, the regularized l_2 output tracking problem with input constraints may be formulated as

$$\min_{\{z,u\}} \quad \phi = \frac{1}{2} \sum_{k=0}^{N-1} \|z_{k+1} - r_{k+1}\|_{Q_z}^2 + \|\Delta u_k\|_S^2 \tag{3.6a}$$

$$s.t. \quad z_k = b_k + \sum_{i=1}^{n} H_i u_{k-i} \qquad k = 1, \dots, N \tag{3.6b}$$

$$u_{\min} \leq u_k \leq u_{\max} \qquad k = 0, \dots, N-1 \tag{3.6c}$$

$$\Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max} \qquad k = 0, \dots, N-1 \tag{3.6d}$$

in which $\Delta u_k = u_k - u_{k-1}$ and Q_z, S and R are the regularization weights and $\|z_{k+1} - r_{k+1}\|_{Q_z}^2$ is the general representation of least square norms with weight.

This can be understood as

$$\|z_{k+1} - r_{k+1}\|_{Q_z}^2 = \|z_{k+1} - r_{k+1}\| \, Q_z \, \|z_{k+1} - r_{k+1}\|^T.$$

In this formulation, the control and the prediction horizon are identical. If desired, a prediction horizon longer than the control horizon could be included in the formulation. However, it is preferred instead to select the control horizon sufficiently long such that any boundary effects at the end of the horizon has no influence on the solution in the beginning of the horizon. Equation 3.6 can be converted to a constrained linear-quadratic optimal control problem. Efficient algorithms exists for the solution of such problems with long prediction horizons, N as explained by Jørgensen *et al.* (2004). In this work a dense quadratic program is formulated in standard form that is equivalent with Equation 3.6. The derivation for formulating QP for FIR based regulator is given in Appendix A.1. Various optimization techniques are available for solving such quadratic programs and here an optimization algorithm based on interior point method is used. The flow chart for the MATLAB execution program and the closed loop simulation of the controller is given in Appendix B and the interior point method algorithm for solving the QP is given in Appendix D

The principle of moving horizon estimation and control process is illustrated in Figure 3.3.

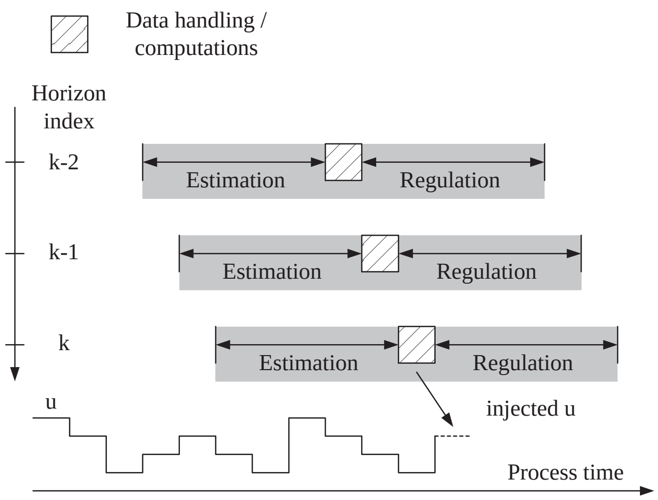


Figure 3.3: The principle of moving horizon estimation and control.

3.3.3 Simple Estimator

To have offset free steady state control when unknown step responses occur, there must be integrators in the feedback loop. This may be achieved using a FIR model in difference variables. Assume that the relation between the inputs and outputs may be represented as

$$\Delta y_k = \Delta z_k = e_k + \sum_{i=1}^{n} H_i \Delta u_{k-i} \tag{3.7}$$

in which Δ is the backward difference operator, i.e. $\Delta y_k = y_k - y_{k-1}$, $\Delta z_k = z_k - z_{k-1}$, and $\Delta u_k = u_k - u_{k-1}$. This representation is identical with the FIR model (Equation 3.5)

$$y_k = z_k = \hat{b}_k + \sum_{i=1}^{n} H_i u_{k-i} \tag{3.8}$$

if $\hat{b}_k$ is computed by

$$e_k = \Delta y_k - \sum_{i=1}^{n} H_i \Delta u_{k-i} \tag{3.9a}$$

$$\hat{b}_k = \hat{b}_{k-1} + e_k \tag{3.9b}$$

Note that in the regulator optimization problem $b_1 = b_2 = \ldots = b_N = \hat{b}_k$ at each time instant. This is based on the assumption that the disturbances enter the process as constant output disturbances. Of course this may not be how the disturbances enter the process in practice, and significant performance deterioration may result as a consequence of this representation.

3.4 Effect of Parameter Uncertainty in the System

In this Section the plant is considered of the form

$$Z(s) = Y(s) = G(s)U(s) + G_d(s)D(s) \tag{3.10}$$

with the transfer functions

$$G(s) = \frac{K(\beta s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)}\, e^{-\tau s} \tag{3.11a}$$

(3.11b)

$$G_d(s) = \frac{K_d(\beta_d s + 1)}{(\tau_{d1} s + 1)(\tau_{d2} s + 1)}\, e^{-\tau_d s} \tag{3.11c}$$

The disturbance model, $G_d(s)$, is kept fixed at its nominal value, while the transfer function, $G(s)$, from $U(s)$ varies around its nominal value. This is used to illustrate the consequence of model uncertainty on the MPC closed-loop performance. The nominal system considered here is $K = K_d = 1$, $\tau_1 = \tau_2 = \tau_{d1} = \tau_{d2} = 5$, $\beta = \beta_d = 2$, and $\tau = \tau_d = 5$. The system is converted to discrete time using a sample time of $T_s = 1$ and a zero-order-hold assumption on the inputs.

The predictive controller is based on the impulse response coefficients of the following system

$$\hat{Z}(s) = \hat{Y}(s) = \hat{G}(s)U(s) \tag{3.12}$$

in which $\hat{G}(s)$ is equal to the nominal system of $G(s)$. The simple estimator described in Section 3.3.3 is used for bias estimation. The input limits are $u_{\text{min}} = -1$, $u_{\text{max}} = 1$, $\Delta u_{\text{min}} = -0.2$, and $\Delta u_{\text{max}} = 0.2$. The horizon of the impulse response model is $n = 40$ and the control horizon is $N = 120$. The MPC is tuned with $Q_z = 1$ and $S = 10^{-3}$.

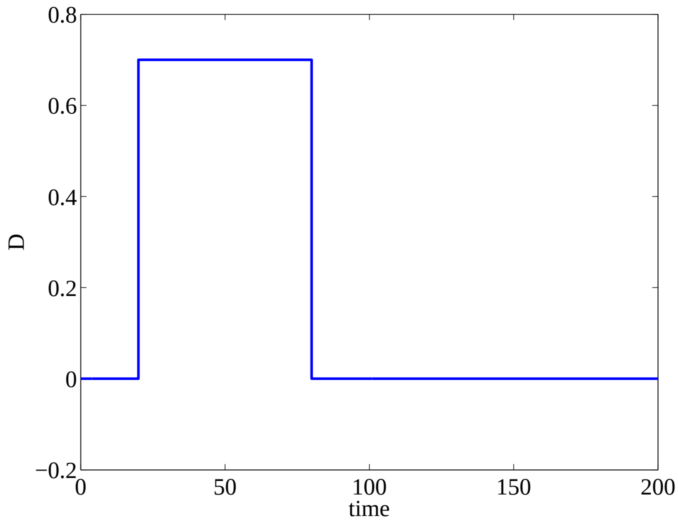


Figure 3.4: Disturbance used for the simulations.

The performance of the controller on the different plants, $G(s)$, is evaluated using the disturbance function in Figure 3.4. This is an unmeasured disturbance and it is unknown to the controller.

3.4.1 Effect of Uncertainty in Gain

The effect of uncertainty in the gain, K is considered. The uncertainty in gain is provided by varying the plant gain as $K = 0.5$ and $K = 1.5$ to the nominal gain $K = 1$. The impulse responses of the system with the with different gains are illustrated in Figure 3.5. The corresponding closed-loop performance of the MPC is illustrated in Figure 3.6.

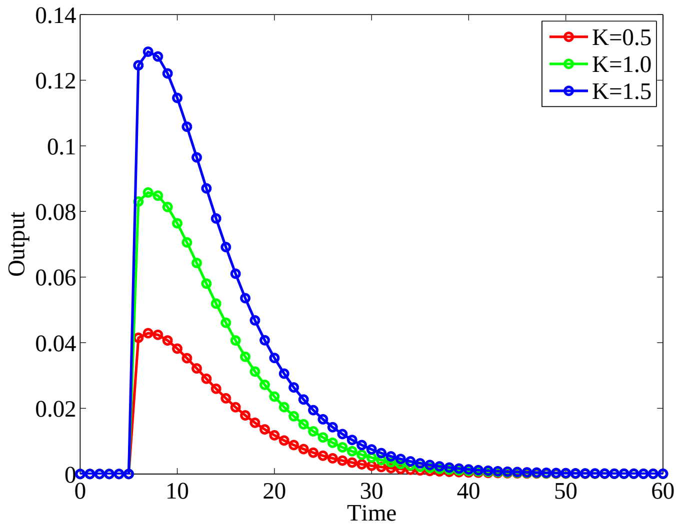


Figure 3.5: Impulse responses for different gains, K, in (3.19a).

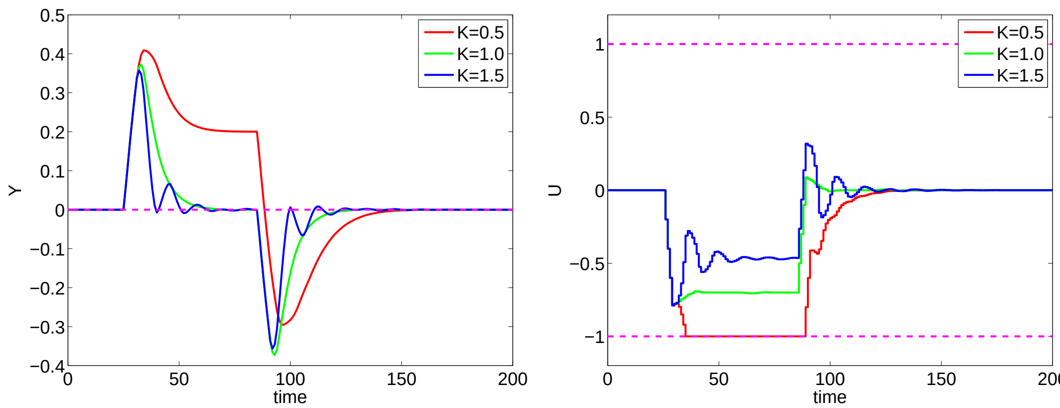


Figure 3.6: Closed-loop MPC performance with gain uncertainty.

In Figure 3.5, it is evident that the gain uncertainty affect the impulse responses significantly. Consequently, there is a performance degradation of the closed-loop MPC response. This is evident in Figure 3.6. The steady state offset for the case $K = 0.5$ in the period $50 - 100$ is due to an infeasible set point. The MV rides its lower limit and is too small (in an absolute sense) to reject the disturbance. For the case $K = 1.5$, the peak overshoot value is 0.32 when compared with nominal model where the peak overshoot is 0.35. The settling time for the gains $K = 1.0$ and $K = 1.5$ are same i.e., 40 s, but the manipulated variable oscillates more in case of $K = 1.5$, as it is more aggressive which is not desirable in real time control.

3.4.2 Effect of Uncertainty in Time Constant

The uncertainty in the time constant, τ_1, of the system in the Equation 3.19a is considered. The impulse responses for different time constants, $\tau_1 = 3.5$, $\tau_1 = 5$ and $\tau_1 = 6.5$ are illustrated in Figure 3.7. The corresponding closed-loop performance of the MPC is illustrated in Figure 3.8.

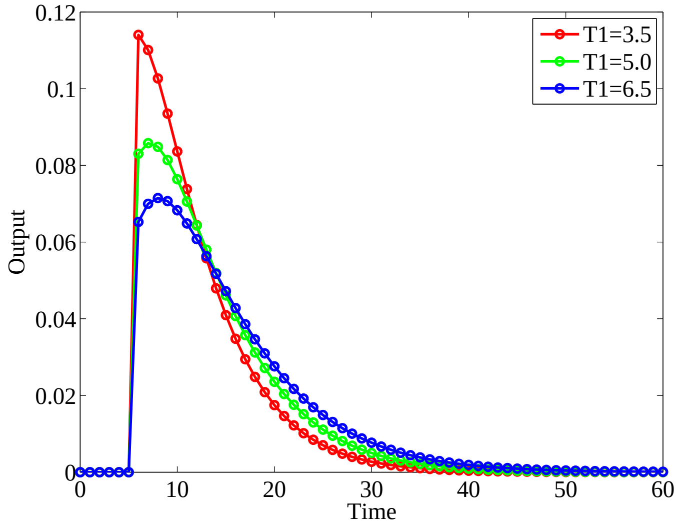


Figure 3.7: Impulse responses for different time constants, τ_1, in Equation 3.19a.

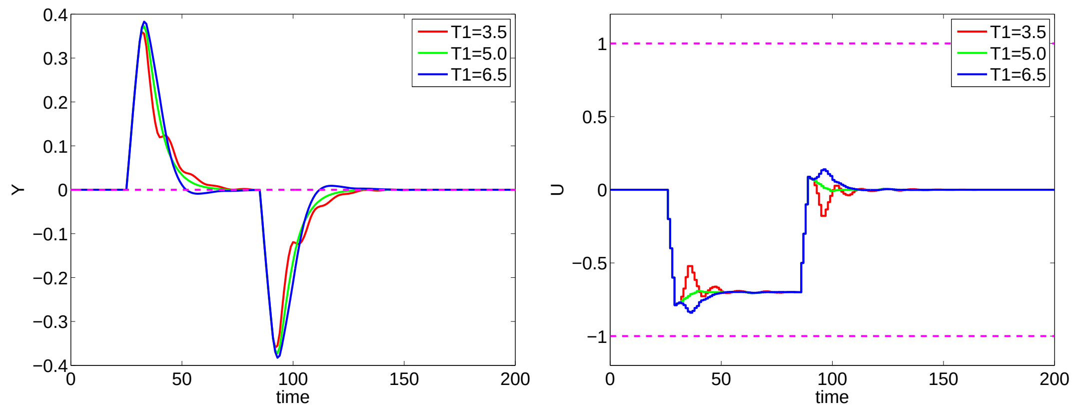


Figure 3.8: Closed-Loop performance with uncertainty in one of the time constants, τ_1.

The peak overshoot of the impulse response with time constant, $\tau_1 = 3.5$ is 0.18 and $\tau_1 = 6$ is 0.07, when compared to the impulse response with nominal value 0.08. However, the degradation in the closed-loop MPC performance due to variations in the time constant, τ_1, is modest, i.e the settling time of the controller for the different values of time constants is $50 - 60$s approximately. Thus it is evident that the closed loop performance of the controller with varying time constants are almost equal.

3.4.3 Effect of Uncertainty in zero β

The impulse responses for different values of β are illustrated in Figure 3.9. This is related to an uncertainty in the zero of the system. It is evident that the impulse response with $\beta = 4$ is 0.15 which is much more different from the nominal impulse response ($\beta = 2$) where amplitude of the response is 0.08. The impulse response with $\beta = 0$ is 0.07 which is much closer to nominal response. From Figure 3.10, it is seen that $\beta = 0$ has higher overshoot of 0.35 when compared with $\beta = 4$ where amplitude is 0.25, but has lesser oscillations and settles with a settling time of 75. It is evident from Figure 3.10, the closed loop performance for the case $\beta = 4$ becomes oscillatory and variations in U are quite high when compared with the performance of the controller having $\beta = 0$.

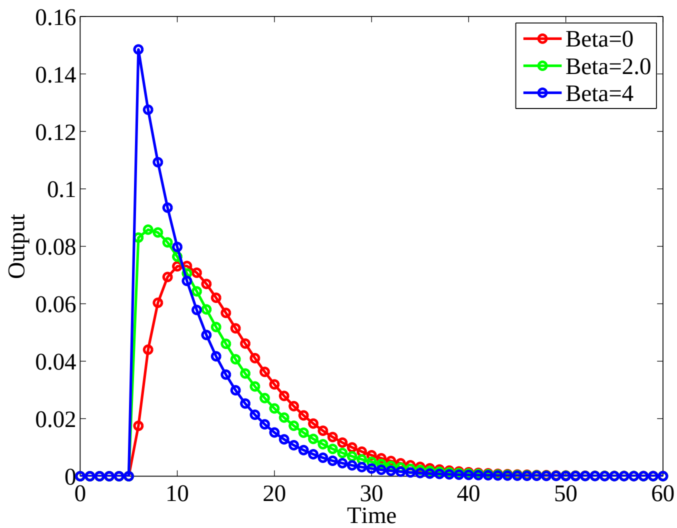


Figure 3.9: Impulse responses for different values of β in (3.19a).

3.4.4 Effect of Uncertainty in Time Delay

The effect of variations in the time delay, τ, of Equation 3.19a on the impulse responses is illustrated in Figure 3.11. The nominal value of time delay is $\tau = 5.0$ and the time delay considered in plant model are $\tau = 3.0$ and $\tau = 7.0$. The

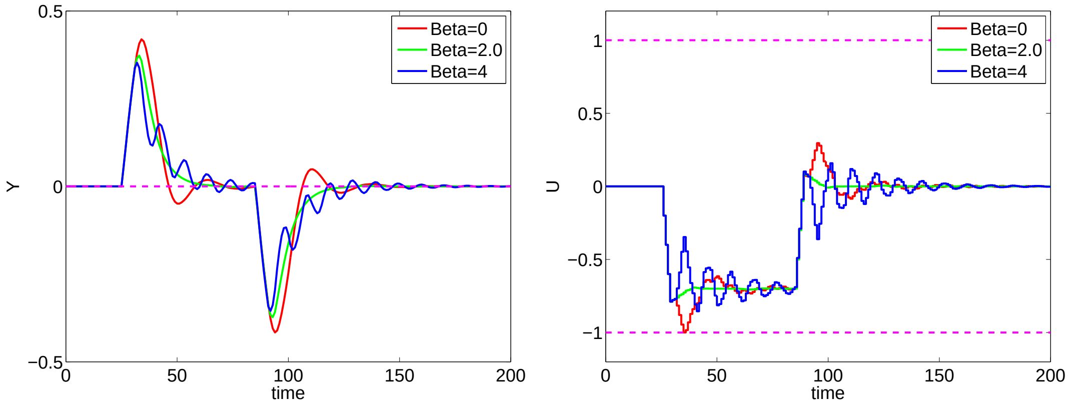


Figure 3.10: Closed-loop MPC response for uncertain values of zero β in the plant (3.19a).

peak amplitude of the impulse response are shifted horizontally which occurs ar different time instants. However, the closed loop response degrades considerably with $\tau = 3.0$ and $\tau = 7.0$, as both the responses reach sustained oscillations when compared with nominal delay. Accordingly, this situation with variations in the time delay corresponds to significant impulse response uncertainty.

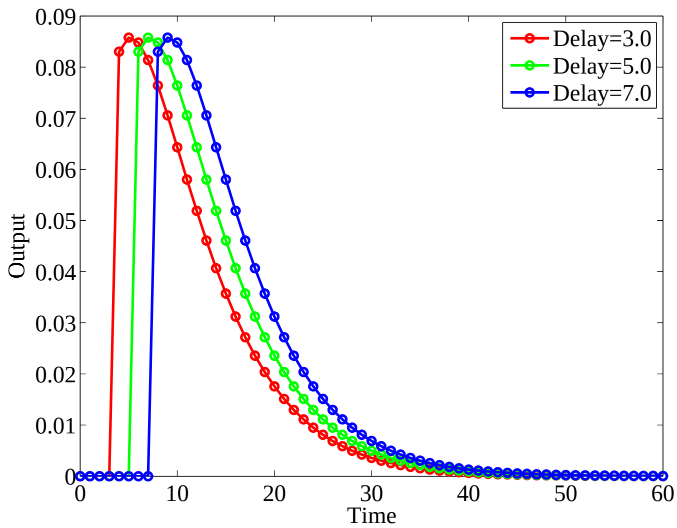


Figure 3.11: Impulse responses for different time delays, τ, in Equation 3.19a.

From the variations in impulse response coefficients it is seen that the degradation of the closed-loop MPC performance is significant as shown in Figure 3.12. The controller response becomes oscillatory when the uncertainty in the time delay of the system becomes more than 40 % of the nominal value and the actuator reaches sustained oscillations and even with $\tau = 3.0$, it reaches the saturation limit.

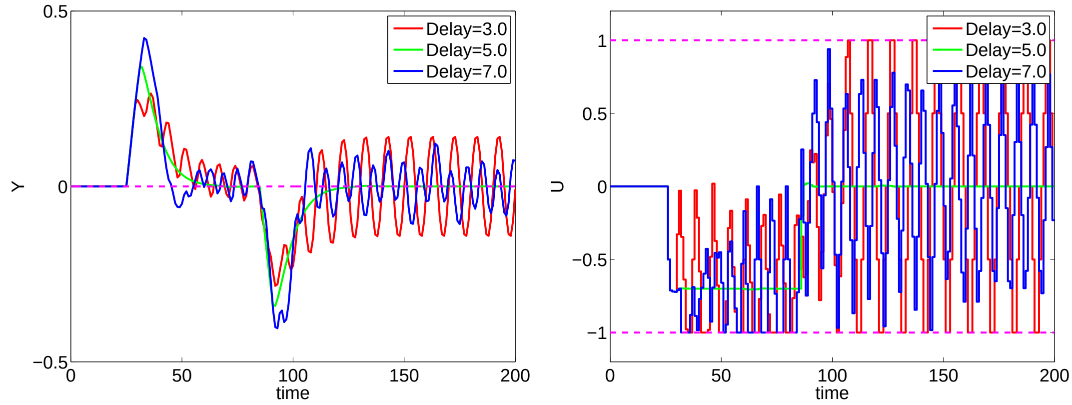


Figure 3.12: Closed-loop MPC performance for uncertainties in plant time delays.

Thus it can be seen that the uncertainty of the impulse response coefficients are well suited to measure the resulting closed-loop MPC performance degradation.

3.4.5 Effect of Measurement Noise and Process Noise

The effect of process noise and measurement noise as well as model uncertainty on the closed-loop MPC performance is investigated in this section.

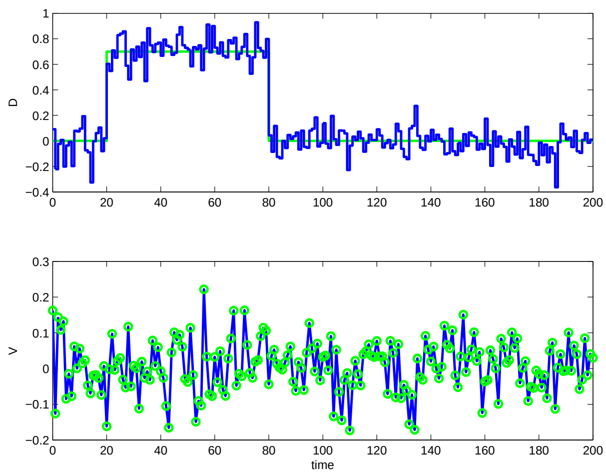


Figure 3.13: Top: Deterministic disturbance function(D) with added process noise. Bottom: Measurement noise(v).

It is assumed that the process noise enters the system in the same way as the unmeasured disturbance, i.e. $G = B_d$. The simulations in this section are based on the process noise and measurement noise illustrated in Figure 3.13. The signals are generated using a process noise variance of $Q = 0.01^2$ and a measurement noise

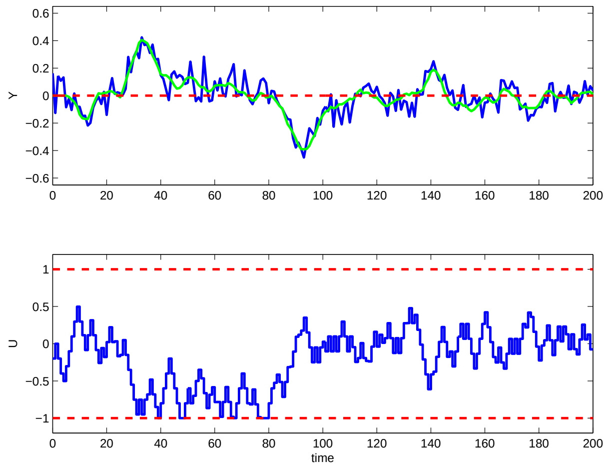


Figure 3.14: Closed-loop MPC performance for the nominal system - Stochastic Case, Top: Output with noise (blue) and Filtered value of output(green).

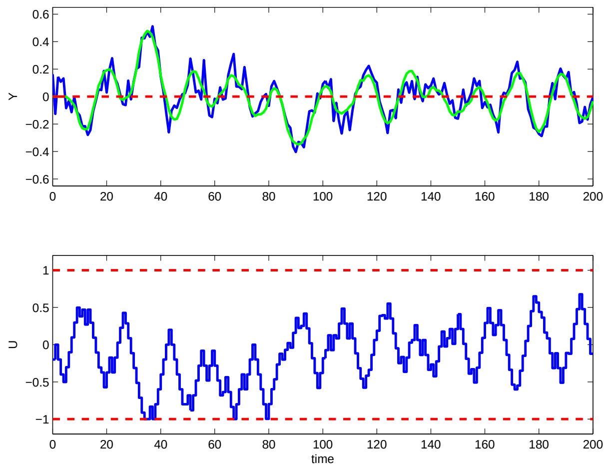


Figure 3.15: Closed-loop MPC performance with plant gain $K = 1.5$ compared to nominal gain $K = 1.0$ with noise included in the process, Top: Output with noise (blue) and Filtered value of output (green).

variance of $R = 0.005^2$. The steady state ($x_0 = 0$) is used as the initial state for the simulations.

The closed-loop MPC performance in the case, when the process model equals the nominal model used for controller design is illustrated in Figure 3.14. Obviously, the performance is degraded compared to the deterministic case. As is evident in Figure 3.15, the closed-loop performance degrades and becomes quite oscillatory in the case when there is a gain mismatch ($K = 1.5$). This is because of the actuator reactions for smaller variations in the output which is actually because of the noise present in the system and when the plant gain becomes 1.5 times

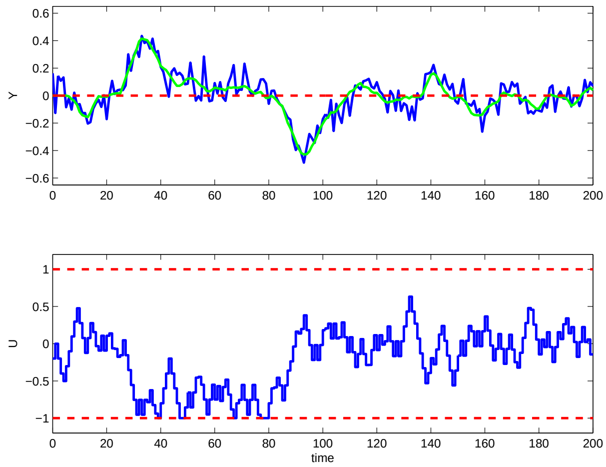


Figure 3.16: Closed-loop MPC performance for $\tau_1 = 6.5$ with noise included in the process (nominal value $\tau_1 = 5$), Top: Output with noise (blue) and Filtered value of output (green).

greater than the controller gain, the performance of the controller still degrades and becomes unstable.

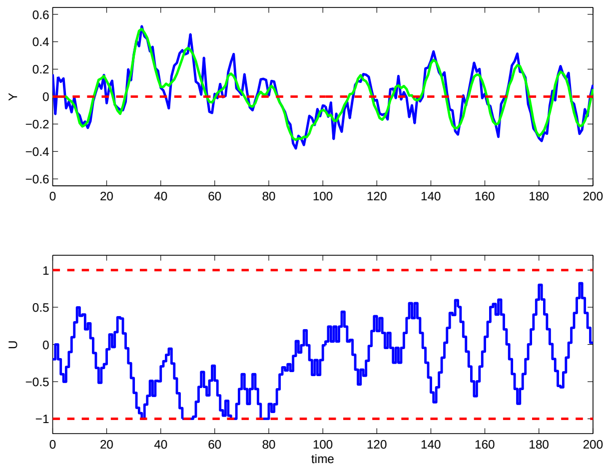


Figure 3.17: Closed-loop MPC performance for $\beta = 4$ with noise included in the process (nominal value $\beta = 2$), Top: Output with noise (blue) and Filtered value of output (green).

The effect of a time constant mismatch ($\tau_1 = 6.5$) is illustrated in Figure 3.16. In this case the performance degradation compared to the nominal case is less pronounced. Thus the closed loop performance does not degrade significantly with uncertainty in time constant as in the deterministic case but the controller provides variation in the actuator because of the noise present in the process.

As provided in Figure 3.17, the closed-loop response degrades significantly and

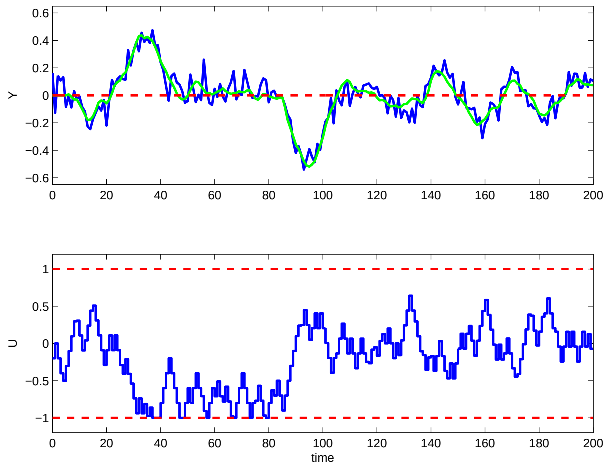


Figure 3.18: Closed-loop MPC performance for uncertainty in delay $\tau_d = 7$ with noise included in the process.

becomes quite oscillatory, when the plant has a zero ($\beta = 4$) when compared to the nominal value, i.e., $\beta = 2$. For this realization of the stochastic signals the effect of a zero mismatch (when it is double the nominal value), results in the largest performance degradation.

The controller performance in case of time- delay mismatch ($\tau_d = 7$) of the system, with the nominal value of delay $\tau = 5.0$, the closed loop performance of the controller is similar to deterministic case and the controller variations are really because of the noise present in the system. This is clearly illustrated in Figure 3.18.

It is seen that the closed-loop MPC performance degradation due to plant-model mismatch is tightly related to the uncertainty of the impulse response coefficients. The affine nature of the FIR model implies that it can be directly applied in predictive controllers based on robust linear programming as well as predictive controllers based on second-order cone programming.

In order to improve the performance of the predictive controller, the constant output disturbance filter is replaced with a moving horizon estimator. The advantage of using moving horizon estimator is that the unmeasured disturbance will be estimated over the prediction horizon based on the measured outputs instead of assuming it to be a constant value.

3.5 Moving Horizon Estimation

Figure 3.19 illustrates the estimation and regulation problem.The estimator must select a disturbance sequence such that historical measurements are reconciled, while regulator must select manipulated variables such that the distance of the predicted outputs to the set-points is minimized. This control law is implemented in a moving horizon such that only the first MV in this sequence is implemented on the plant. At the next sample time the procedure is repeated.

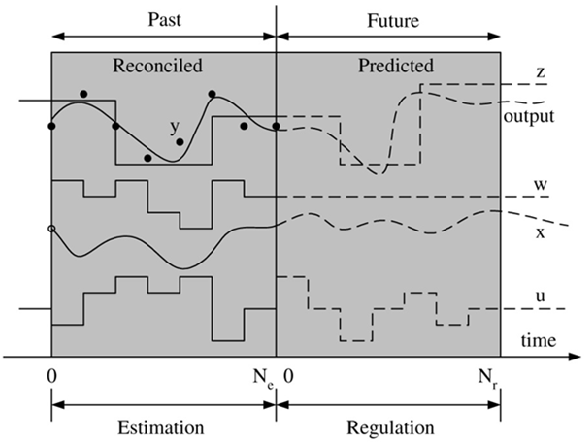


Figure 3.19: Moving horizon estimation and regulation.

3.5.1 Model used by Regulator and Estimator

The plant and sensors are assumed as in section 3.3 to be a linear state space system. The model used in the estimation problem is a finite impulse response (FIR) model

$$z_k = b_k + \sum_{i=1}^{n} H_{u,i} u_{k-i} + \sum_{i=0}^{n} H_{d,i} d_{k-i} \tag{3.13}$$

in which $\{H_{u,i}\}_{i=1}^{n}$ are the impulse response coefficients from the manipulated variables(or known process inputs) to the outputs, while $\{H_{d,i}\}_{i=0}^{n}$ are the impulse response coefficients from the unknown process disturbances denoted by d_k. The process disturbances, d_k, are assumed to consist of a deterministic and a stochastic component, i.e. $d_k = \begin{bmatrix} \hat{d}'_k & \hat{w}'_k \end{bmatrix}'$. It is assumed that there is no immediate effect from the control inputs to the outputs, i.e. $H_{u,0} = 0$. In contrast, the unknown disturbances may have a direct influence on the outputs i.e. in general $H_{d,0} \neq 0$.

This parametrization allows unknown input as well as output disturbances. b_k is the bias term that may be used to account for the mean of z_k.

In this work, the moving horizon estimator is based on the FIR model as in Equation 3.13. The simple estimator in section (3.3.3) used by the model predictive controller in the previous chapter, is replaced by the following set of equations, the controller part of the calculation is similar to Equation 3.6a.

$$\min_{\{z,d\}} \quad \phi = \frac{1}{2}\sum_{k=0}^{N_e} \|z_k - y_k\|_{Q_z}^2 + \|\Delta d_k\|_S^2 + \|d_k\|_R^2 \tag{3.14a}$$

$$s.t. \qquad z_k = \bar{b_k} + \sum_{i=0}^{n} H_{d,i} d_{k-i} \tag{3.14b}$$

$$d_{\min} \leq d_k \leq d_{\max} \qquad k = 0, \ldots, N_e \tag{3.14c}$$

$$\Delta d_{\min} \leq \Delta d_k \leq \Delta d_{\max} \qquad k = 0, \ldots, N_e \tag{3.14d}$$

$$where \qquad \bar{b_k} = b_k + \sum_{i=1}^{n} H_{u,i} u_{k-i} \qquad k = 1, \ldots, N_e \tag{3.14e}$$

It is assumed that $\{d_i\}_{i=-1}^{-n}$ are disturbance values fixed at their previous estimates such that the real decision variables in Equation (3.14e), are $\{d_k\}_{k=0}^{N_e}$. The decision variables are normally split into a slow varying component (level), d_k and a rapid varying component (process noise), w_k, thus the unknown disturbance can be given as $d_k = \bar{d_k} + w_k$.

3.5.2 Regulator

For the regulator the current time is indexed as $k = 0$ as in Figure 3.19. The output predictions used by the regulator are based on the FIR model with the past unknown disturbances fixed at their last estimated values, $\left\{ d_k = \hat{d_k} = \begin{bmatrix} \hat{\bar{d}}_k' & \hat{w_k}' \end{bmatrix}' \right\}_{k=-N_e}^{0}$, and the predicted future disturbances are equal to the estimated current mean disturbance value, i.e. $\left\{ d_k = \begin{bmatrix} \hat{\bar{d}}_0' & 0' \end{bmatrix}' \right\}_{k=1}^{N_r}$. This implies that the output prediction in the regulator becomes

$$z_k = \bar{b_k} + \sum_{i=1}^{n} H_{i,u} u_{k-i} \tag{3.15}$$

with

$$\bar{b_k} = b_k + \sum_{i=0}^{n} H_{d,i} d_{k-i} \tag{3.16}$$

in which $\{H_i\}_{i=1}^n$ are the impulse response coefficients (Markov parameters). b_k is a bias term generated by the estimator. b_k accounts for discrepancies between the predicted output and the actual output. In this paper, the output predictions used by the regulator are based on the FIR model as in Equation (3.15). Consequently, using the FIR model in Equation (3.15), the regularized l_2 output tracking problem with input constraints may be formulated as

$$\min_{\{z,u\}} \quad \phi = \frac{1}{2} \sum_{k=0}^{N_r-1} \|z_{k+1} - r_{k+1}\|_{Q_z}^2 + \|\Delta u_k\|_S^2 \tag{3.17a}$$

$$s.t. \quad z_k = b_k + \sum_{i=1}^{n} H_i u_{k-i} \qquad k = 1, \ldots, N \tag{3.17b}$$

$$u_{\min} \leq u_k \leq u_{\max} \qquad k = 0, \ldots, N-1 \tag{3.17c}$$

$$\Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max} \qquad k = 0, \ldots, N-1 \tag{3.17d}$$

in which $\Delta u_k = u_k - u_{k-1}$. Here we are more concerned with the design of estimator rather than a regulator, so we adopt a standard approach for formulation of moving horizon estimation as in Appendix A.2.

3.6 Simulation

In this Section the plants are considered of the form

$$Z(s) = Y(s) = G(s)U(s) + G_d(s)D(s) \tag{3.18}$$

with the transfer functions

$$G(s) = \frac{K(\beta s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-\tau s} \tag{3.19a}$$

$$\tag{3.19b}$$

$$G_d(s) = \frac{K_d(\beta_d s + 1)}{(\tau_{d1} s + 1)(\tau_{d2} s + 1)} e^{-\tau_d s} \tag{3.19c}$$

The disturbance model, $G_d(s)$, is kept fixed at its nominal value, while the transfer function, $G(s)$, from $U(s)$ varies around its nominal value. This is used to illustrate the consequence of model uncertainty on the MPC closed-loop performance. The nominal system considered here is $K = K_d = 1$, $\tau_1 = \tau_2 = \tau_{d1} = \tau_{d2} = 5$, $\beta = \beta_d = 2$, and $\tau = \tau_d = 5$. The system is converted to discrete time using a sample time of $T_s = 1$ and a zero-order-hold assumption on the inputs and corrupted by normally distributed measurement noise.

$$y(t_k) = z(t_k) + v(t_k) v(t_k) \sim N_{iid}(0, r^2) \tag{3.20}$$

The nominal model used by estimator and regulator is

$$\hat{Z}(s) = \hat{G}(s)U(s) + \hat{G}_d(s)(\bar{D}(s) + W(s)) \tag{3.21}$$

with $\hat{G}(s) = G(s)$ and $\hat{G}_d(s) = G_d(s)$ for the nominal parameters.

Here the performance of regulator is evaluated with MHE by introducing an unknown step disturbance and try to estimate the disturbance and use it as a feed back to the MPC. The output are measured at discrete times (Sample Interval $T_s = 1$) and corrupted by normally distributed measurement noise.

The constraint limits are $\bar{d}_{min} = -10$, $\bar{d}_{max} = 10$, $\Delta\bar{d}_{min} = -1$, $\Delta\bar{d}_{max} = 1$, $w_{min} = -0.5$, $w_{max} = 0.5$, $\Delta w_{min} = -2$ and $\Delta w_{max} = 2$. The horizon of the FIR model is $n = 50$ and the estimation window has size $N_e = 200$.

A. Batch Estimation :

In batch estimation, the outputs are generated by disturbance scenario. The manipulated inputs are zero in this scenario. The disturbance is given as step up and step down, this disturbance is unknown to the estimator. The measured outputs are corrupted by measurement noise.

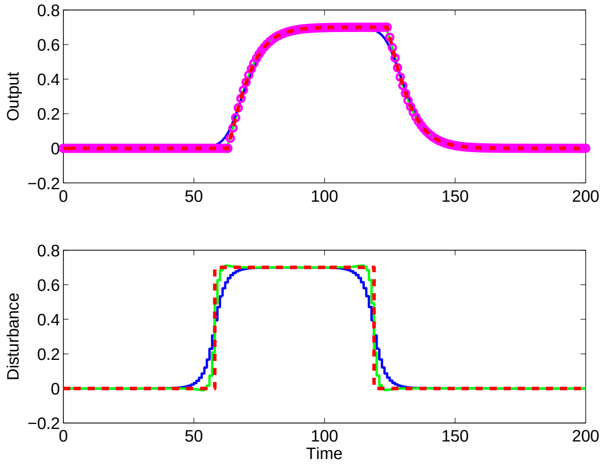


Figure 3.20: Batch estimation with no measurement noise. Top: Measured Output Bottom: Actual and Estimated disturbance Low regularization weights ($S_d = 0.1$ and $R_w = 0.01$).

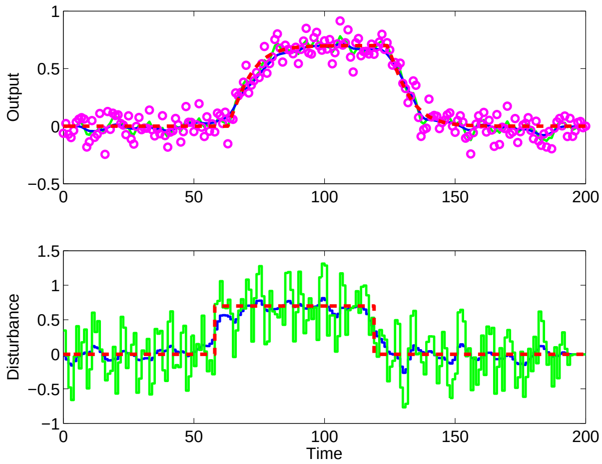


Figure 3.21: Batch estimation with measurement noise and ($S_d = 0.1$, $R_w = 0.01$). Top: Measured Output Z (solid line)and Output with noise Y (Dots). Bottom: Actual Disturbance (Dotted lines), the estimated deterministic disturbance(blue) and the total disturbance(with the stochastic component added)(green).

In Figure 3.20, the upper plot is the measured output Z (solid line) and output with measurement noise Y(dots), generated by the unknown disturbance. The lower plot is the actual disturbance(red dotted line) and the estimated disturbance(blue solid line). In the simulation of stochastic disturbance, it is considered that the measured output is corrupted by normally distributed measurement noise that has a standard deviation, $r = 0.2$. The most important tuning parameters in the estimator are the regularization weights S_d and R_w.

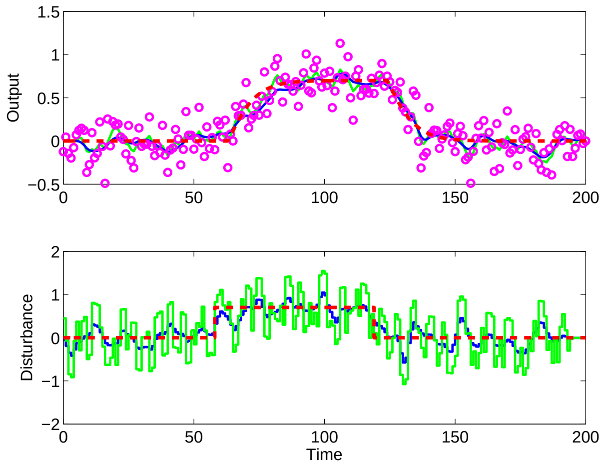


Figure 3.22: Batch estimation with measurement noise. Medium regularization weights ($S_d = 1$ and $R_w = 0.1$), (Legends : As in Figure 3.21).

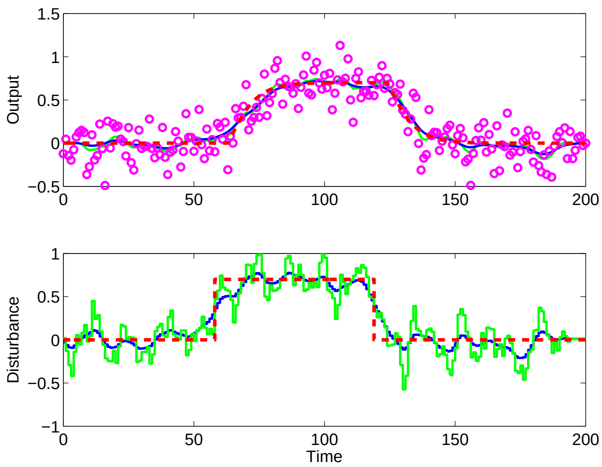


Figure 3.23: Batch estimation with measurement noise. High regularization weights ($S_d = 5$ and $R_w = 0.5$), (Legends : As in Figure 3.21).

The task of the estimator is to reconstruct the unknown disturbance, D, from the measured outputs. It can also be called as the smoother. In case of the system with no measurement noise, the estimated disturbance can be reconstructed arbitrarily accurately by lowering the regularization weights. This is illustrated in Figure 3.20. Figure 3.21 provides the details of both deterministic component of the estimated disturbance and the sum of the estimated deterministic and stochastic component. The blue curve in the lower part of the figure is the deterministic component of the estimated disturbance and the green curve is the sum of the estimated and stochastic disturbance.

From Figures 3.21, 3.22 and 3.23, it can be noted that the inclusion of measurement noise disturbs the estimation to a large extent. It can be seen that with lower regularization the variations of the estimated unknown disturbance exactly follows the measurement noise. In the Figure it can be seen that the measurement noise has a standard deviation $r = 0.2$ and with low regularization the standard deviation of estimated disturbance is 0.18. The frequency variations in the estimated disturbance components can be attenuated only by increasing the regularization weights in the estimation as shown in Figures 3.21- 3.23. But the increase in regularization will affect the estimator performance.

B. Moving Horizon Estimation:

When the estimator in Equation A.13, is implemented in a moving horizon manner for the unknown disturbance with the inclusion of measurement noise that has a standard deviation $r = 0.2$, the result illustrated in Figure 3.24 is obtained.

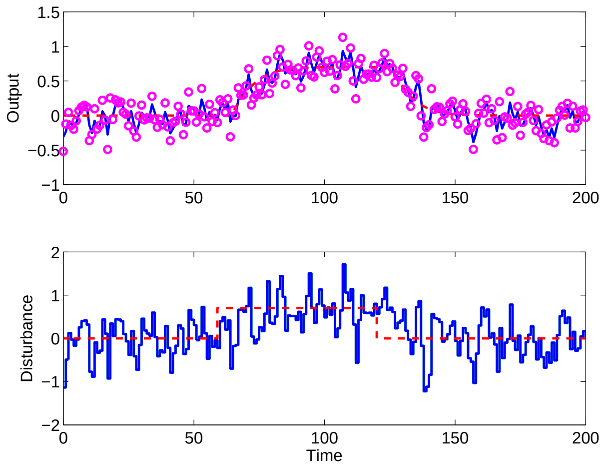


Figure 3.24: Moving Horizon Estimation with measurement noise ($r = 0.2$) and ($S_d = 1$ and $R_w = 0.1$), (Legends : As in Figure 3.21)

By increasing the regularization weights, high frequency variations in the estimated disturbance is attenuated. But this will result in the slower adjustment in estimating the deterministic change in the real disturbance. The estimated disturbance lags the real disturbance due to the time delay in the model. Thus it is seen from the figures that by determining the proper regularization weights on the parameters a better disturbance estimate can be provided. This estimated disturbance can then be used as a feed back to the regulator.

C. Closed Loop Simulation:

Let $Q_z = 1$, $S = 0.1$, $u_{min} = -1$, $u_{max} = 1$, $\Delta u_{min} = -0.2$, $\Delta u_{max} = 0.2$ and $N_r = 200$ in the predictive regulator. The closed loop performance of MPC with moving horizon estimator is shown in Figures 3.26 - 3.28. Figure 3.25 shows the response obtained in the case of no measurement noise. This response illustrates the ideal behavior of the system. In particular, the achievable estimated disturbance. The actuator variations in the controller are also quite smoother but still provides delayed settling time because of the time delay in the model.

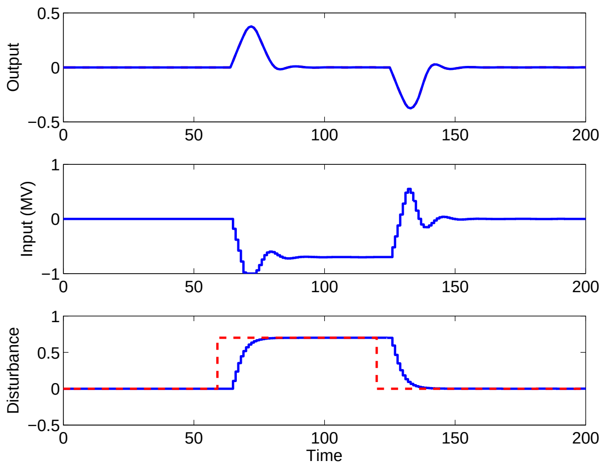


Figure 3.25: Closed-loop MPC simulation without measurement noise. Bottom: Actual disturbance (red dotted line) and the estimated disturbance (blue). Regularization ($S_d = 0.1$ and $R_w = 0.01$).

As a case study, the closed loop performance of the estimator is analyzed when MHE is made in closed loop with the MPC with measurement noise. The measurement noise is included with the standard deviation $r = 0.2$. The performance of the estimator is investigated with different regularization from the objective function as in Equation A.13. When the regularization weight of the MHE has been made very low, the estimation just follows the measurement noise and the actuator reacts following the measurement noise providing real output variations by manipulating the process input as shown in Figure 3.26. From Figure 3.28, it is observed that as the regularization weight is increased the estimation does not follow the measurement noise but the estimation becomes sluggish. Thus the

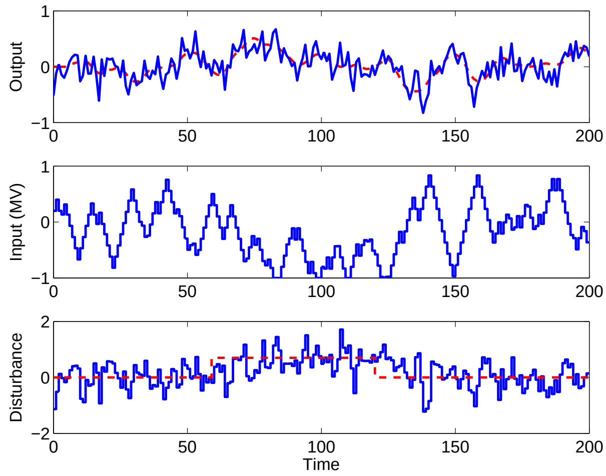


Figure 3.26: Closed-loop MPC simulation with measurement noise ($r = 0.2$). Low regularization ($S_d = 0.1$ and $R_w = 0.01$), Legends as in Figure 3.25.

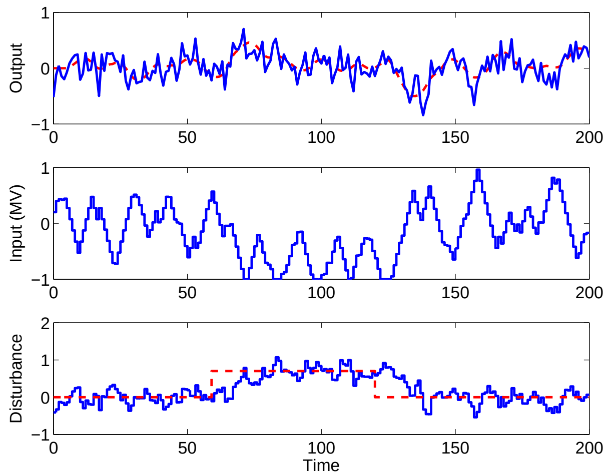


Figure 3.27: Closed-loop MPC simulation with measurement noise ($r = 0.2$). Medium regularization ($S_d = 1$ and $R_w = 0.1$), Legends as in Figure 3.25.

controller manipulates the process input with regards to slower variations in the disturbances. But the output variations become sluggish, along with variations because of measurement noise and hence the closed loop performance becomes poor.

The results obtained from closed loop performance of MPC with MHE show that the MHE produces non-smooth disturbance estimates upon which the regulator reacts and introduces real output variations by its manipulation of the process input. By increasing the regularization weights in the estimator, the introduced process variations are slightly reduced at the expense of slower disturbance rejec-

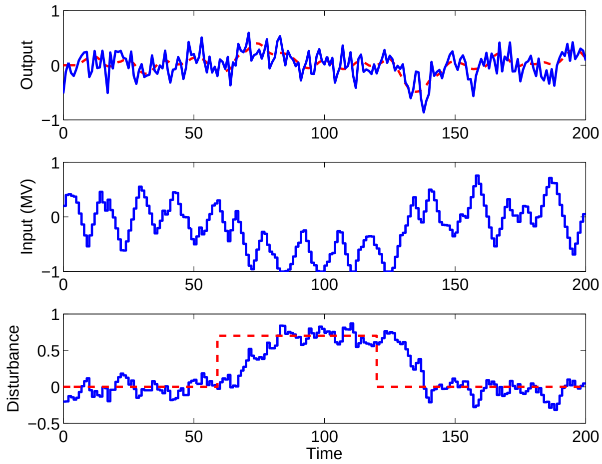


Figure 3.28: Closed-loop MPC simulation with measurement noise ($r = 0.2$). High regularization ($S_d = 5$ and $R_w = 0.5$), Legends as in Figure 3.25.

tion. These results indicate that MHE is no panacea to MPC. The improvements can be made by considering other norms, convex formulations (e.g. l_1 or l_2 with dead zone). In an unconstrained case a finite horizon LQG controller can be represented with an unstructured disturbance model in the present work. For the case $N_r = N_e \to \infty$ it approaches a stationary LQG controller with an unstructured disturbance model for offset free control. In the next chapter, in order to improve the robustness incase of plant-model mismatch, a soft constraint is included in the predictive controller with a simple estimator. It is assumed that the same results would be obtained using a state- space model based predictive control system, but the advantage of FIR based model is that future predictions are governed by past disturbances, which results in better closed loop performance.

3.7 Soft Constrained MPC

The soft constraints can be used to tune and improve the performance of linear model predictive control. Specifically, the effect of uncertain models on the performance of a regularized l_2 model predictive controller with input constraints, input-rate constraints and soft output constraints is investigated in this chapter.

The estimator used here is simple estimator. The reason for using simple estimation instead of MHE is that, the moving horizon estimator are computationally complex and the algorithm needs to solve both regulation and estimation problem simultaneously.

Here the objective function of the MPC is similar to that of FIR based MPC, but it includes quadratic and linear weights on soft output constraints with respect to the reference parameter along with normal deviation and actuator objective functions as given below

$$\begin{aligned} \min_{\{z,u,\eta\}} \phi = \frac{1}{2} \sum_{k=0}^{N-1} \|z_{k+1} - r_{k+1}\|^2_{Q_z} + \|\Delta u_k\|^2_S \\ + \sum_{k=1}^{N} \frac{1}{2} \|\eta_k\|^2_{S_\eta} + s'_\eta \eta_k \end{aligned} \tag{3.22a}$$

subject to the constraints

$$z_k = b_k + \sum_{i=1}^{n} H_i u_{k-i} \qquad k = 1, \dots N \tag{3.22b}$$

$$u_{\min} \leq u_k \leq u_{\max} \qquad k = 0, \dots N-1 \tag{3.22c}$$

$$\Delta u_{\min} \leq \Delta u_k \leq \Delta u_{\max} \qquad k = 0, \dots N-1 \tag{3.22d}$$

$$z_k \leq z_{\max,k} + \eta_k \qquad k = 1, \dots N \tag{3.22e}$$

$$z_k \geq z_{\min,k} - \eta_k \qquad k = 1, \dots N \tag{3.22f}$$

$$\eta_k \geq 0 \qquad k = 1, \dots N \tag{3.22g}$$

in which $\Delta u_k = u_k - u_{k-1}$ and η_k is the slack for soft limits around the reference variable. S_η and s'_η are weights on the slack variables. The derivation for formulating QP from the Equation 3.22 is given in Appendix A.3. The general form of the derivation for all the Quadratic program equations is given in Appendix C.

3.7.1 Soft Constraint Principle

The basic principle of soft constraint based MPC is shown in Figure 3.29. This illustrates the basic stage cost function for ℓ_2 model predictive control (conven-

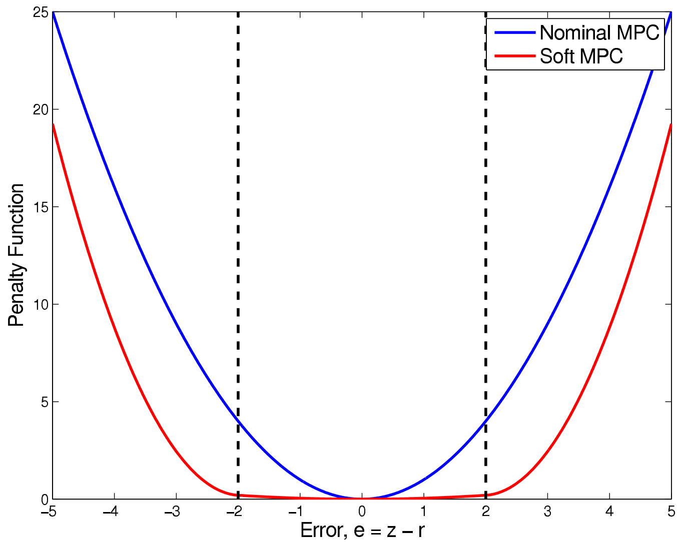


Figure 3.29: The principle of soft constraint MPC and conventional MPC(Error,e Vs Penalty Function, ρ).

tional MPC) and ℓ_2 predictive control with a dead zone (soft MPC). The stage cost function, or the penalty function, is plotted as function of the set point error, $e = z - r$.

The penalty function of the conventional MPC is a quadratic function. The plot shows the variation of penalty function against reference output error. For MPC without soft output constraints, the penalty function is quadratic with respect to the output as given by the solid parabolic line in Figure 3.29. The penalty function shown in Figure 3.29 is denoted by ρ for conventional controller and the controller with soft constraints. The penalty function of the soft constrained MPC is constructed in such a way that it is zero or almost zero within the dead zone between the soft limits and grows quadratically when the set point error exceeds the soft limits.

In order to make the controller offset free, a small quadratic penalization is included when the error is inside the limits shown as solid parabolic red line in Figure 3.29. The small penalty within the soft limits ensures that the controller produces a steady state offset free response. By having the penalty small within the soft constraints, the controller does not react much to small errors. In this way the significant real disturbances to the process is avoided by the controller because it does not react to measurement noise or plant-model mismatch. Outside the soft limits, it is assumed that the deviation from target is due to a real process disturbance, and the soft MPC may be designed to react in the same way as the conventional MPC. The standard form of Quadratic algorithm for solving the soft constraint formulation is given in Appendix A.3

The matlab codes for designing a soft MPC and closed loop simulation of the system in given in Appendix E.

3.7.2 Selection of Soft Limits

The soft output limits considered in the controller is a limit or band around the reference target of the controlled variable. The performance of the MPC is determined based on the selection of limits around the reference target. Too wide limit can cause the controller performance sluggish and even it may lead to steady state offset. Selecting the soft limits very close to the reference can cause the controller behave like a conventional MPC and will reduce the robustness of the controller. Figure 3.30 provides the method for choosing the soft limits. The two curves in the figure are the linear and quadratic term of the soft constraint. The upper plot in the figure is the measurement noise over a time horizon and the lower plot is the response of the linear and quadratic term of the soft constraints for the system with measurement noise. It is seen from the figure, that the controller almost rejects the noise and acts only with respect to the process variations.

The soft limits must be selected in a way, such that the controller does not react to the unnecessary parameter variations because of the measurement noise, process noise and also the real process variation is considered. In Figure 3.30 it is seen that for the measurement noise with variance $\sigma^2 = 2.5$, the soft limits chosen here is $z_{max} = 2$, $z_{min} = -2$, so that the controller reacts very less on noise and control aggressively for real disturbance in the process parameter. Thus the soft limits must be chosen by considering the variance of the noise (measurement noise as well as process noise) in the system. In this way the controller can be made to react only to the real process disturbance and avoid unnecessary actions for the noise or plant-model mismatch. Thus as a thumb rule it can be defined that when

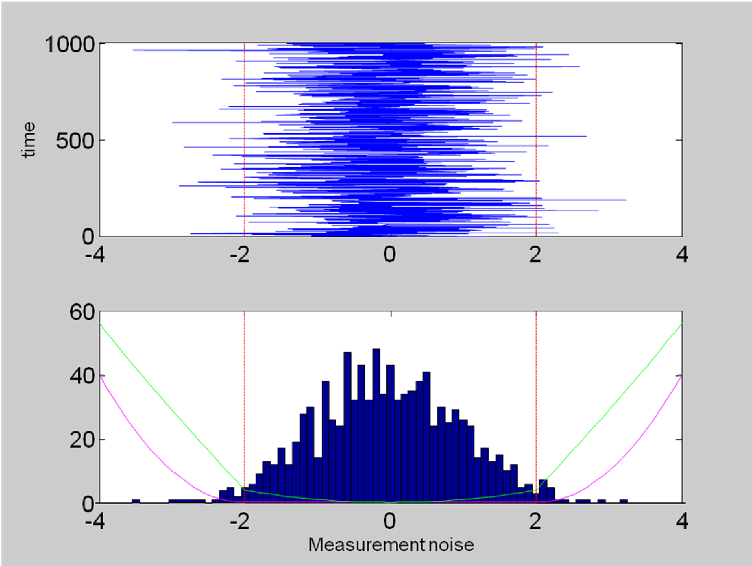


Figure 3.30: Linear (green) and Quadratic term (pink) of the soft constraint.

the variance of the measurement noise in the system $\sigma^2 = N$. Then the soft limits to be chosen for the process z_{max}, z_{min} must be $0.8 - 0.9N$. In the next chapter, the performance of the developed soft MPC is compared with the conventional MPC to analyze the robustness of the controllers during plant-model mismatch.

3.8 Conclusion

Based on finite impulse response predictions, a regularized l_2 predictive controller with input and input-rate constraints is analyzed. It is verified by simulations that the closed-loop MPC performance degradation due to plant-model mismatch is tightly related to the uncertainty of impulse response coefficients. The closed loop performance of the controller is good when the uncertainty in each of the parameters (separately in gain, zero and time delay) are smaller(upto 40 % incase of time delay, 50 % in gain and twice incase zero) with the nominal model. The controller performance reaches sustained oscillations when the uncertainty in each of these parameters exceeds the above these values. Also presence of noise in the system further degrades the closed loop performance of the controller. By MATLAB simulations, the closed loop behavior of moving horizon estimation and model predictive control for a system with measurement noise is provided. It is evaluated that the closed loop response of the model predictive controller has a strong correlation to the magnitude of the measurement noise i.e., the quality of the sensor. However, significant process variations results from the control systems due to non-smooth disturbance estimates. These results are not due to the fact that the regulator and estimator are based on a FIR model.

Then the model predictive controller equipped with soft output constraints along with input and input - rate constraints is developed. The prediction is based on the Finite impulse response model, and the feedback is based on simple constant output disturbance filter.

CHAPTER 4

Comparison of Soft MPC with Nominal MPC Using Simulation

In this chapter, the controllers with and without soft constraints are compared both for SISO and MIMO cases. By MATLAB simulation, the performance of the new robust constrained predictive controller is compared with a conventional predictive controller both in nominal case and in uncertainty conditions. The model considered here is a SOPDT transfer function model with a zero. It is assumed that the disturbance in the model enters in the same way as the plant model. The performance of the controllers with process noise and measurement noise is also studied.

4.1 Introduction

Model predictive controllers with robust performance against plant-model mismatch is crucial in long-term maintenance and success of MPC system. Using soft output constraints in a novel way, it is demonstrated that the poor performance of predictive control in the case of plant-model mismatch can be improved significantly. The soft constraints create a dead zone around the set point and by simulation it is demonstrated, that the performance of such an MPC does not degrade much in the nominal case but improves significantly in the case of plant-model mismatch. This technique is similar but not identical to the funnels used by Honeywell in RMPC (Qin and Badgwell, 2003; Havlena and Lu, 2005; Havlena and Findejs, 2005). Normally in classical process control, PID control with dead zones (Shinskey, 1988) have been used in a similar way like soft output constraints based MPC. The soft output constraint included in the MPC acts as a dead zone to the controller to reduce its sensitivity to noise and uncertainty when the process output is close to its target. This use of soft constraints for robustness is new, simple, and gives good performance. Bemporad and Morari (1999) have provided an excellent survey of methodologies for robust model predictive control.

4.2 Simulation

Here the plant is considered of the form

$$Z(s) = Y(s) = G(s)U(s)y(t_k) = z(t_k) + v(t_k) \tag{4.1a}$$

with the transfer functions

$$G(s) = \frac{K(\beta s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-\tau s} \tag{4.2a}$$

$$\tag{4.2b}$$

$$G_d(s) = \frac{K_d(\beta_d s + 1)}{(\tau_{d1} s + 1)(\tau_{d2} s + 1)} e^{-\tau_d s} \tag{4.2c}$$

The nominal system is $K = K_d = 1$, $\tau_1 = \tau_2 = \tau_{d1} = \tau_{d2} = 5$, $\beta = \beta_d = 2$, and $\tau = \tau_d = 5$. The system is converted to discrete time using a sample time of $T_s = 1$ and a zero-order-hold assumption on the inputs.

The predictive controller is based on the impulse response coefficients of the following system

$$\hat{Z}(s) = \hat{Y}(s) = \hat{G}(s)U(s) \tag{4.3}$$

in which $\hat{G}(s)$ is equal to the nominal system of $G(s)$.

The simple estimator in section 3.3.3 is used for estimation. The input limits are $u_{min} = -1, u_{max} = 1, \Delta u_{min} = -0.2$, and $\Delta u_{max} = 0.2$. The horizon of the impulse response model, n = 40 and the control horizon, $N = 120$. The MPC is tuned with $Q_z = 1$ and $S = 10^{-3}$. The unknown deterministic process disturbance, D(s), the stochastic process disturbances, W(s) or w_k, and the measurement noise, $v(t_k) = v_k$, are used in the simulations. The stochastic process disturbances is $w_k \sim N(0, 0.01)$, and the stochastic measurement noise is $v_k \sim N(0, 0.01)$.

4.3 Soft constraints in MPC

A comparison on the performance of the MPC with soft constraints and the conventional MPC is made in deterministic case as well as when the process is perturbed by stochastic noises. In cases where noise variance is large, the MPC without soft constraints reacts to the noise variations resulting in oscillatory performance of the system. So the soft limit is included to penalize heavily only when it violates the limit and a very little action to drive the output to target with zero offset within the limit. This way of control relatively gives much stable operation avoiding unnecessary fluctuations in the input.

4.3.1 Effect of including noise in the system

To study the effectiveness of soft constraints, a simple case of second order system with a zero and input delay is considered as given in Equation (4.2a). The model used by the controller is identical to the deterministic part of the plant model. However, the plant has in addition to the deterministic part stochastic process disturbances and stochastic measurement noise as illustrated in Figure 4.1.

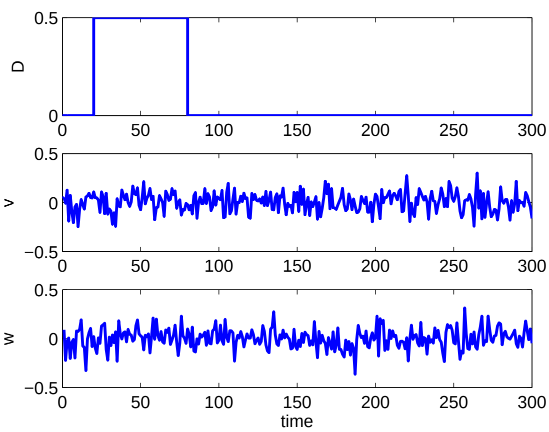


Figure 4.1: External signals used in simulation: Disturbance (d_k) measurement noise (v) and process noise(w).

First, a case is considered, in which the system is perturbed by measurement noise with variance $Q_v = 0.01$ and process noise with variance $Q_w = 0.01$ and it is assumed that there is no external disturbance $D(s) = 0$. The normal weight on measurement error $Q_z = 0.001$ incase of soft MPC and $Q_z = 1.0$ in case of conventional MPC. The comparison on the open loop performance of MPC for stochastic system with and without soft constraints is given in Figure 4.2. In this Figure, the variations of controller input when there is noise in the output is studied, but the values are not given back to the plant for control actions.

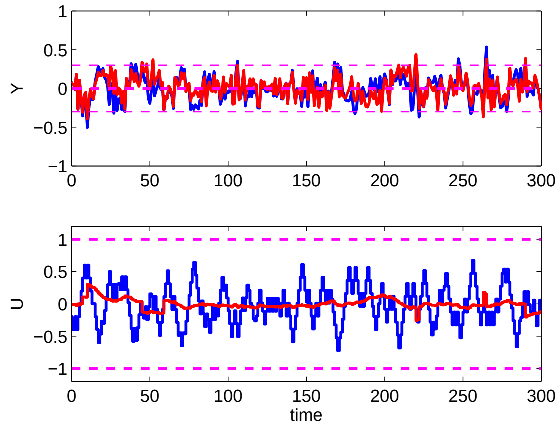


Figure 4.2: Comparison of Open loop performance of conventional and soft MPC,with nominal models applied to a stochastic system with no deterministic disturbance (Conventional MPC = blue, Soft MPC = red).

Though the output variations in both the controller are similar, the input variations shows the MPC without soft constraints acts even for noise variation. Thus the input variance of the soft MPC is much smaller than the input variance of the nominal MPC. Due to the low penalties within the soft limits, the soft MPC does not react to measurement noise and do not need to compensate such previous erroneous measurement noise induced input moves. The closed loop performance further degrades when there is a plant-model mismatch. Further, this can be illustrated clearly the system is simulated with plant-model mismatch to be discussed in the following sections.

4.4 Effect of uncertainties in the model

A deterministic system without process noise or measurement noise is considered. However, the model used by the controllers is different from the plant model. The process is perturbed by an unknown deterministic disturbance, D(s), as illustrated in Figure 4.1. The performance of the conventional MPC and the soft MPC for model-plant mismatches defined by the time delay, τ_d, the gain K, the time constant τ_1, and the zero β are compared.

4.4.1 Effect of uncertainty in Delay

The effect of uncertainty in Delay τ_d is considered for comparing the performances of conventional MPC and soft MPC. The performance of both the MPCs and their corresponding input variations for the deterministic case are illustrated in Figure 4.3 and Figure 4.4. When the delay τ_d of the plant model varies with the controller model there is a reasonable performance degradation in the close loop control. Here when the plant delay is either higher or lower than the model delay, it affects the performance considerably.

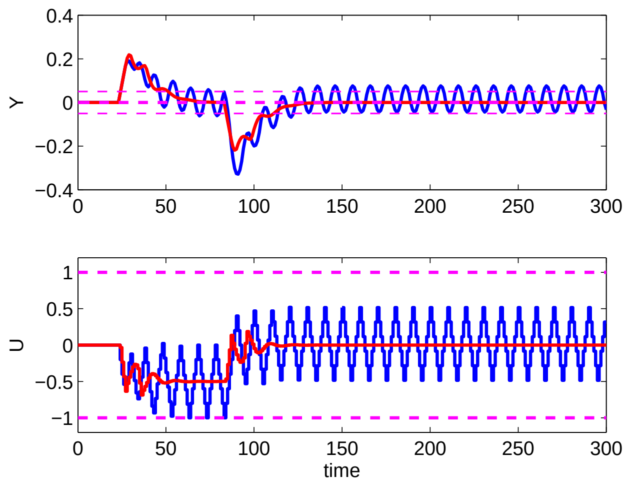


Figure 4.3: Closed-loop MPC performance with uncertainty in time delay, $\tau_d = 3$, in Equation (4.2a)(Conventional MPC = blue, Soft MPC = red).

The uncertainty in time delay, $\tau_d = 3.0$ and $\tau_d = 7.0$ is considered here. It can be seen that when compared to the controller variations with nominal delay $\tau_d = 5.0$ the MPC without soft constraints manipulates the control input for small variations in output, resulting in instability, where as soft MPC does not react to the small output variations resulting in faster stabilization. It can be seen that in case of soft MPC, the controller settles within a period of 30s where as the conventional MPC results in sustained oscillations.

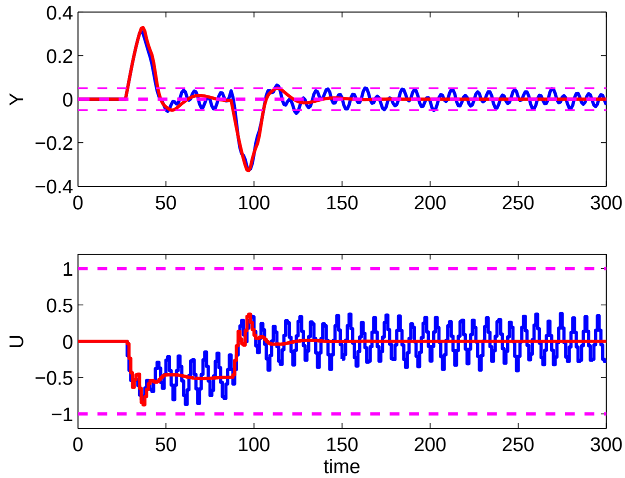


Figure 4.4: Closed-loop MPC performance with higher plant time delay , $\tau_d = 7$, in Equation (4.2a)(Conventional MPC = blue, Soft MPC = red).

4.4.2 Effect of uncertainty in gain

The effect of uncertainty in the gain, K is considered. The comparison on the performance of both the MPCs and their corresponding input variations for the deterministic case is illustrated in Figure 4.5.

When the gain of the process is on the higher side, i.e., $K = 2$ or more when compared with the controller model gain $K = 1$ the closed loop performance of MPC without soft constraints degrades as it starts acting to variations in output aggressively, resulting in sustained oscillations. This again makes the output oscillatory as shown in the Figure 4.5. But in case of soft MPC the controller does not react to small variations within the soft limits providing better stability. Here even the soft MPC has output variations getting oscillatory but settles faster (in $60s$) , whereas MPC without soft constraints oscillates the manipulated variable continuously resulting in continuous oscillations. Incase of plant gain on the lower side or slightly higher than the controller model gain, the controller will normally behave sluggish both in conventional MPC and soft MPC, so both the controllers behave in similar way resulting in stable operation.

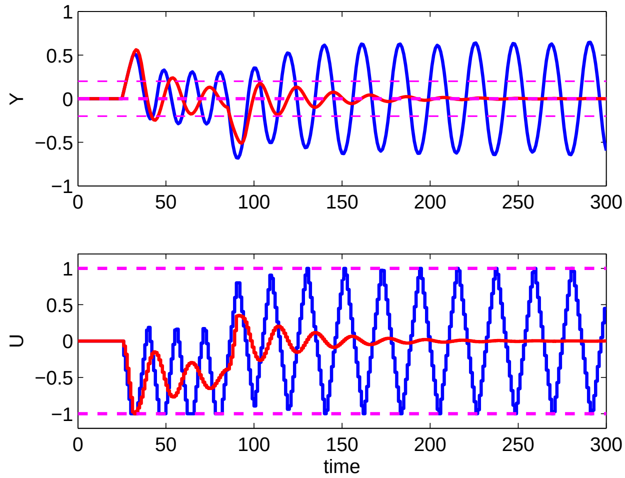


Figure 4.5: Closed-loop MPC performance with gain uncertainty, $K = 2$, in Equation (4.2a)(Conventional MPC = blue, Soft MPC = red).

4.4.3 Effect of uncertainty in Time Constant

In case of uncertainty in time constant, τ_1 of the system.

When one of the time constants of the system, $\tau = 2.0$ or less when compared to the nominal time constant $\tau = 5$, the output performance degrades as the input starts reacting to the variations resulting in sustained oscillations. As referred in chapter 4, the effect of time constant in all the other cases will remain same as they do not have significant impact on the stability.

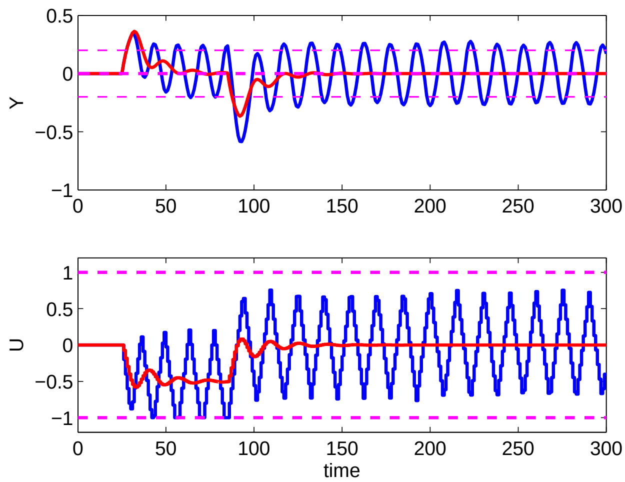


Figure 4.6: Closed-loop MPC performance with uncertain time constant, τ_1, in Equation (4.2a)(Conventional MPC = blue, Soft MPC = red).

The comparison on performance of both the MPCs with the uncertainty in time constant is illustrated in Figure 4.6. It can be seen that when the time constant of the plant is close to the sampling time i.e., $2s$ or less the controller performance degrades in case of conventional MPC as it provides continuous oscillations, but soft MPC provides stable variation even with time constants closer to the sampling time.

4.4.4 Effect of uncertainty in Zero

The uncertainty in the zero β of the system is considered as given in the Figure 4.7.

The plant having the value of zero $\beta = 4.5$ or more when considered with nominal model $\beta = 2$ the performance of the conventional MPC degrades with increased oscillation in the output, where as the soft MPC performs in a much stabilized way as shown. This behavior is almost similar to that of the uncertain gain model, but soft MPC settles much faster in this case (settling time of $40s$).

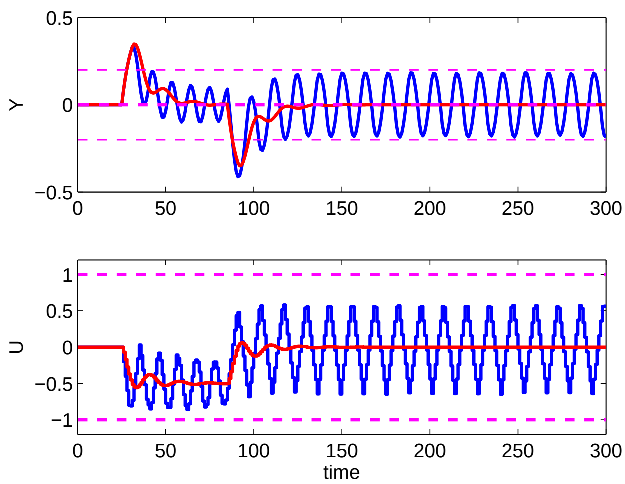


Figure 4.7: Closed-loop MPC performance with uncertain zero, β, in Equation (4.2a)(Conventional MPC = blue, Soft MPC = red).

In all of these cases only the deterministic part of the process is considered, the closed loop performance can be affected considerably when process noise and measurement noise are included into the system.

4.5 Effect of Process noise and Measurement noise

From the previous section, on comparing the performance of MPC with and without soft output constraints for a deterministic model, it is observed that the uncertainty in any of the parameters of the model degrades the performance of conventional MPC considerably than the MPC with soft constraints.

The effect of process noise and the measurement noise on the closed loop performance is investigated. It is assumed that the process noise enters the same way as the unmeasured disturbance, i.e. $G = B_d$.

For simulation purpose, we generate the signals with the process noise variance of $Q_w = 0.01^2$ and the measurement noise variance of $Q_v = 0.01^2$. The steady state ($x_0 = 0$) is used as the initial state for the simulations.

The closed loop performance of the conventional MPC and the soft MPC is compared with the nominal model and inclusion of the measurement noise . The closed loop performance is as shown in Figure 4.8. It can be seen that the conventional MPC reacts for small variations in output which is because of the noise perturbations in the system. This makes the inputs from the controller oscillatory resulting in unstable operation. In this case, the controlled variable, Y (or Z), of the two controllers are similar while the manipulated variable, U, of the soft MPC has significantly less variance than the manipulated variable, U, of the nominal MPC.

The closed loop performance of a conventional MPC and a soft MPC applied to the system with the external signals in Figure 4.1 and a plant-model mismatch in the gain is compared. The plant gain is $K = 2$ and the controller gain is $K = 1$. As evident in Figure 4.9, the closed-loop performance of the conventional MPC degrades and becomes quite oscillatory in the case when there is a gain mismatch ($K = 2.0$). It is seen that the performance of the soft MPC is significantly better than the performance of the conventional MPC. The superior performance is achieved by having a small set point deviation penalty within the soft constraints such that the controller does not react aggressively when close to the set point.

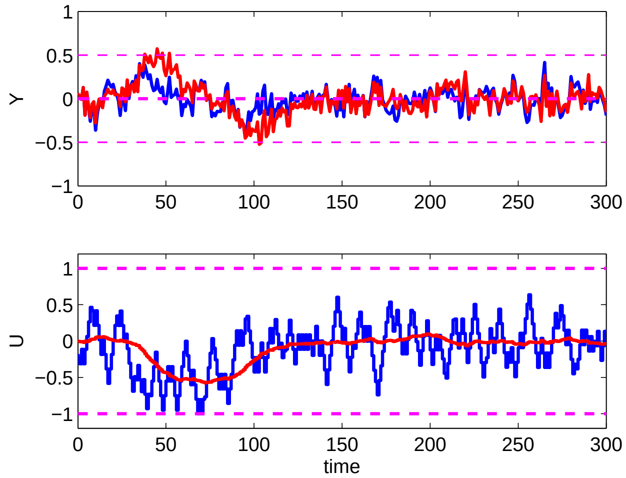


Figure 4.8: Comparison of normal and soft MPC with nominal models applied to a stochastic system with an unknown deterministic disturbance (Conventional MPC = blue, Soft MPC = red).

In this way it avoids perturbing the system due to stochastic measurement noise and plant-model mismatch.

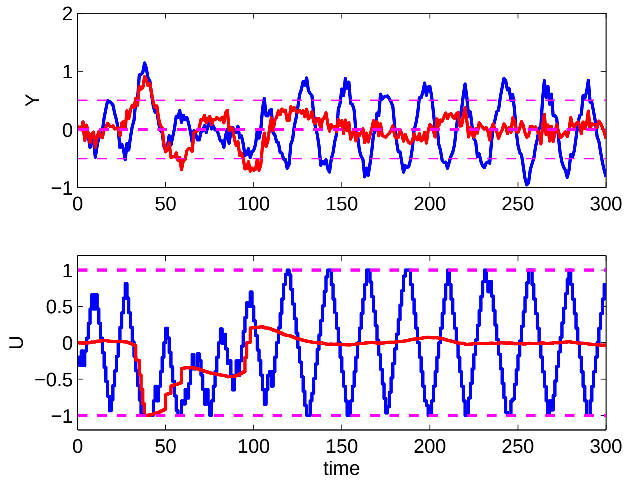


Figure 4.9: Closed-loop MPC performance with noise and gain uncertainty, The plant gain is $K = 2$ and the model gain is $K = 1$(Conventional MPC = blue, Soft MPC = red).

4.6 Simulation of MIMO System

The performance of the controllers for MIMO case is simulated using Matlab. The model considered here is taken from the cement mill circuit from the chapter 5.4.

The controlled variables are $\begin{bmatrix} Elevator Load \\ Fineness \end{bmatrix}$ and the manipulated variables are $U(s) = \begin{bmatrix} Feed \\ Separator speed \end{bmatrix}$. The fineness measurement to be obtained from online analyzers (continuous sample data). The details of getting the models from the cement mill circuit is explained in Chapter 5. The main control strategy of cement mill involves controlling of product fineness(blaine) and load on the mill (elevator power) by manipulating feed and separator speed. Grindability is taken as the disturbance parameter.

The transfer function model for cement mill relating controlled and manipulated variables based on the step test in the simulator can be given by $G(s) = \frac{Y(s)}{U(s)}$

$$G(s) = \begin{bmatrix} \frac{0.62}{(45s+1)(8s+1)}e^{-5s} & \frac{0.29(8s+1)}{(2s+1)(38s+1)}e^{-1.5s} \\ \frac{(-15)}{(60s+1)}e^{-5s} & \frac{5}{(14s+1)(s+1)}e^{-0.1s} \end{bmatrix} \tag{4.4}$$

where

$$Y(s) = \begin{bmatrix} Elevator Load \\ Fineness \end{bmatrix} \text{ and } U(s) = \begin{bmatrix} Feed \\ Separator speed \end{bmatrix}$$

The transfer function model in Equation 4.4 is fed to the controller.

The hardness of the clinker(grindability) entering the mill is considered as a disturbance model and an approximate model is derived for the transfer function between the hardness and the output, $Y(s)$ as given in equation 4.5.

$$G_d(s) = \begin{bmatrix} \frac{-1}{(32s+1)(21s+1)}e^{-3s} \\ \frac{(60)}{(30s+1)(20s+1)} \end{bmatrix} \tag{4.5}$$

The performance of the controllers is studied by introducing a step change in the disturbance model as shown in Figure 4.10.

The input limits considered for normal actions are based on trial and error method and the limits are

$$u_{min} = \begin{bmatrix} -7 \\ -4 \end{bmatrix},$$

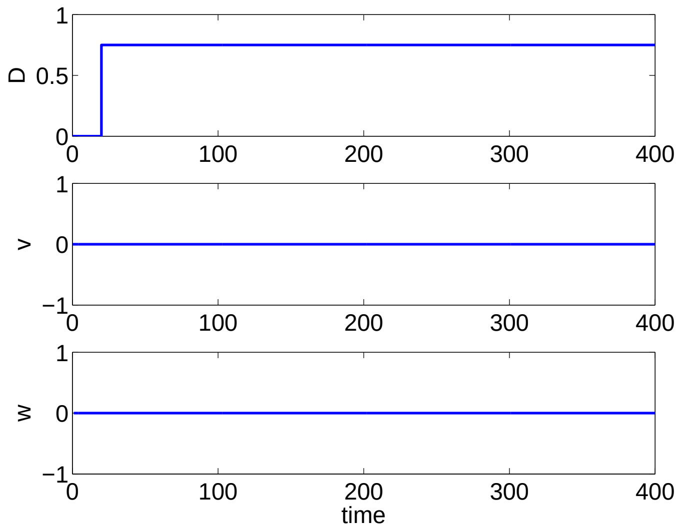


Figure 4.10: Disturbance Model as in Equation (4.5) with no process noise and measurement noise.

$$u_{max} = \begin{bmatrix} 7 \\ 4 \end{bmatrix},$$

$$\Delta u_{min} = \begin{bmatrix} -1.5 \\ -0.3 \end{bmatrix}, \text{ and}$$

$$\Delta u_{max} = \begin{bmatrix} 1.5 \\ 0.3 \end{bmatrix}$$. The horizon of the impulse response model is n $=$ 75 and the control horizon is N $=$ 225. The weightage parameters for conventional MPC are fixed considering the controller to be aggressive for variations in output thus the values

$$Q_z = \begin{bmatrix} 8 & 0 \\ 0 & 0.2 \end{bmatrix} \text{ and}$$

$$S = \begin{bmatrix} 10^{-8} & 0 \\ 0 & 1 \end{bmatrix}$$ are fixed. Similarly soft MPC is tuned with

$$Q_{zs} = \begin{bmatrix} 0.08 & 0 \\ 0 & 0.01 \end{bmatrix} \text{ and}$$

$$S_s = \begin{bmatrix} 10^{-2} & 0 \\ 0 & 1 \end{bmatrix}$$ for providing the same effectiveness in control actions during normal conditions.

The performance of the conventional MPC is slightly better than the MPC with soft constraints for nominal cases as shown in Figure 4.11. This is because the soft MPC does not react aggressively when the outputs are within the soft limits.

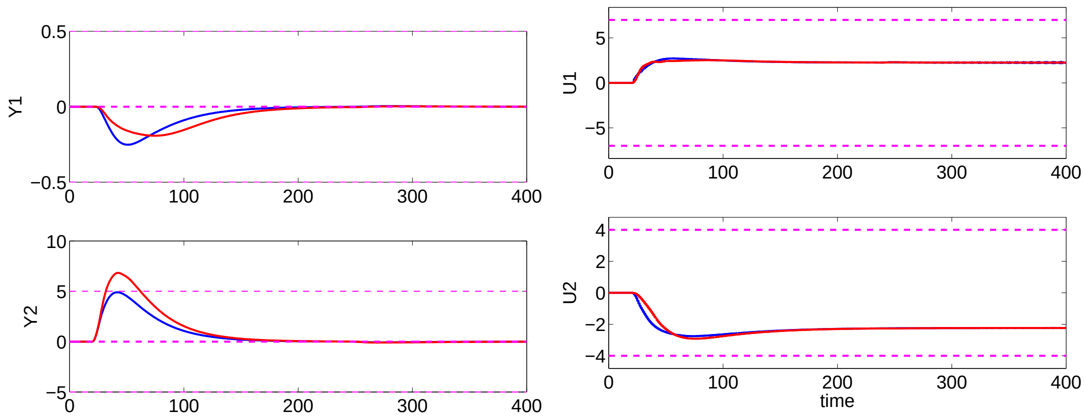


Figure 4.11: Variation of controllers Nominal Case, as in Equation (4.4) (Conventional MPC = blue, Soft MPC = red).

4.6.1 Effect of uncertainty in gain

The cement mill model is simulated using MATLAB by introducing uncertainty in the model gain using single value decomposition (SVD) method. The original gain of the system are decomposed using SVD as Gain,

$$\mathrm{K} = \begin{bmatrix} 1.6866 & 1.4240 \\ -14.9812 & 5.0200 \end{bmatrix} \text{ and then fed as a new model to the controller.}$$

The comparison of the controllers with the new gain is given in Figure 4.12. As the uncertainty in gain increases, the normal controller becomes aggressive resulting in oscillatory behavior. Since the weight on deviation of Y1 is more than Y2 the controller reacts more for maintaining the parameter(Y1) resulting in sustained oscillations, when compared to Y2. In case of soft MPC it can be seen that both the control parameters settle before 75% of the control horizon resulting in stable operation. These features are expected to contribute to better closed loop performance, easier maintenance, easier tuning, and longer lifetime of model predictive controllers for chemical processes.

In the next chapter, the performance of both the controllers when applied to a cement mill circuit which is subjected to large number of uncertainties is studied.

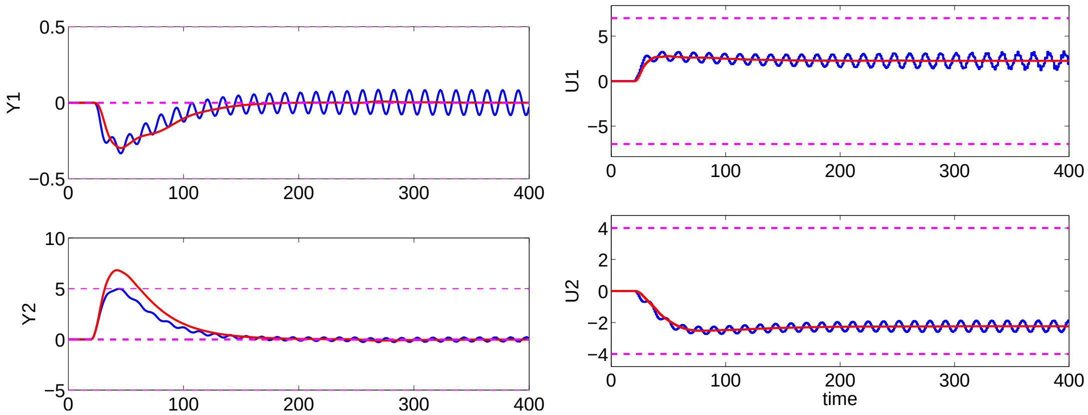


Figure 4.12: Comparison of Conventional MPC and Soft MPC in case of Uncertainty in Gain, K (MIMO), in Equation (4.4)(Conventional MPC = blue, Soft MPC = red).

4.7 Conclusion

It is verified by simulations that the soft MPC provides much robust performance in the face of plant-model mismatch than conventional MPC. In nominal case, it can be seen that the performance of conventional MPC is slightly better than the soft MPC. This is because in soft MPC, the controller actions are very small whenever the error is closer to target(within the soft band) resulting in sluggish performance. In SISO case with uncertain system, it can be seen that soft MPC stabilizes the system faster than conventional MPC. The performance of both the controllers are similar, with the uncertainty condition when the plant gain and zero is lesser than the controller gain and zero, also with plant having higher time constant values than the controller model. When there is uncertainty in time delay (both higher and lower values) of the system the soft MPC settles in $50s$. But in case of uncertainty in gain it can be seen that the soft MPC is little oscillatory and settles much slower. For the MIMO case, with uncertainty in the model gain, the conventional MPC results in sustained oscillations where as in soft MPC the oscillations are suppressed completely and the system settles before 75% of the control horizon.

CHAPTER 5

Cement Manufacturing Process

In this chapter, a brief overview of the cement plant architecture is given, in which the developed robust MPC is to be implemented. Also the importance of cement mill operation in the cement plant a and the need for control of cement mill circuit is discussed. A transfer function model for the cement mill circuit is developed from the data obtained from the cement mill simulation system through step response tests. The transfer function of the system is a 2×2 system with Feed rate and separator speed as manipulated variables and Elevator load and Fineness as controlled variables.

5.1 Introduction

The annual world consumption of cement is around 1.7 billion tonnes and is increasing at about 1% a year. The electrical energy consumed in the cement production is approximately 110 kWh/tonne. 30% of the electrical energy is used for raw material crushing and grinding while around 40% of this energy is consumed for grinding clinker to cement powder (Fujimoto, 1993; Jankovic *et al.*, 2004). Hence, global cement production uses 18.7 TWh which is approximately 2% of the worlds primary energy consumption and 5% of the total industrial energy consumption.

The cement manufacturing process is illustrated in Figure 5.1.

PROCESS DESCRIPTION

Cement industries typically produce portland cement, although they also produce masonry cement (which is also manufactured at portland cement plants). Portland cement is a fine, typically gray powder comprised of dicalcium silicate, tricalcium silicate, tricalcium aluminate, and tetracalcium aluminoferrite, with the addition of forms of calcium sulfate. Different types of portland cements are created based

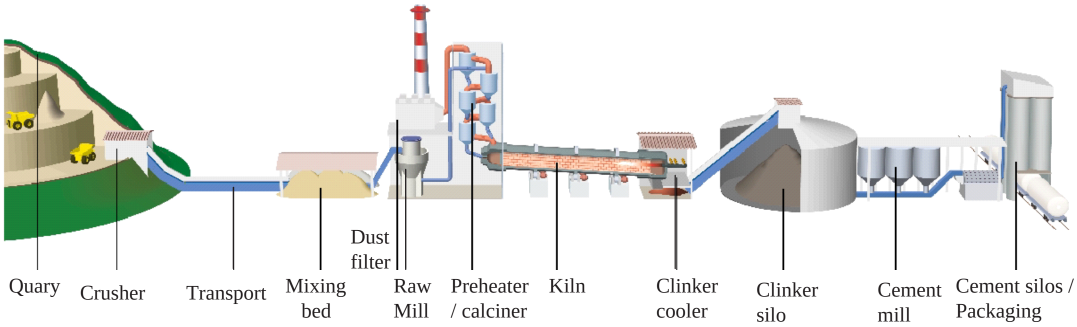


Figure 5.1: Cement plant (FLSmidthA/S (2004)).

on the use and chemical and physical properties desired. Portland cement types I - V are the most common. Portland cement plants can operate continuously for long time periods (i.e., 6 months) with minimal shut down time for maintenance. The air pollution problems related to the production, handling, and transportation of portland cement are caused by the very fine particles in the product. The stages of cement production at a portland cement plant are:

a. Procurement of raw materials

b. Raw Milling - preparation of raw materials for the pyroprocessing system

c. Pyroprocessing - pyroprocessing raw materials to form portland cement clinker

d. Cooling of portland cement clinker

e. Storage of portland cement clinker

f. Finish Milling

g. Packing and loading

a. Raw Material Acquisition

Most of the raw materials used are extracted from the earth through mining and quarrying and can be divided into the following groups: lime (calcareous), silica (siliceous), alumina (argillaceous), and iron (ferriferous). Since a form of calcium carbonate, usually limestone, is the predominant raw material, most plants are situated near a limestone quarry or receive this material from a source via inexpensive transportation. The plant must minimize the transportation cost since one third of the limestone is converted to CO_2 during the pyroprocessing and is subsequently lost. Quarry operations consist of drilling, blasting, excavating, handling, loading, hauling, crushing, screening, stockpiling, and storing.

b. Raw Milling

Raw milling involves mixing the extracted raw materials to obtain the correct chemical configuration, and grinding them to achieve the proper particle-size to ensure optimal fuel efficiency in the cement kiln and strength in the final concrete product. Three types of processes may be used: the dry process, the wet process, or the semidry process. If the dry process is used, the raw materials are dried using impact dryers, drum dryers, paddle-equipped rapid dryers, air separators, or autogenous mills, before grinding, or in the grinding process itself. In the wet process, water is added during grinding. In the semidry process the materials are formed into pellets with the addition of water in a pelletizing device.

c. Pyroprocessing

In pyroprocessing, the raw mix is heated to produce portland cement clinkers. Clinkers are hard, gray, spherical nodules with diameters ranging from 0.32 - 5.0 cm (1/8 - 2") created from the chemical reactions between the raw materials. The pyroprocessing system involves three steps: drying or preheating, calcining (a heating process in which calcium oxide is formed), and burning (sintering). The pyroprocessing takes place in the burning/kiln department. The raw mix is supplied to the system as a slurry (wet process), a powder (dry process), or as moist pellets (semidry process). All systems use a rotary kiln and contain the burning stage and all or part of the calcining stage. For the wet and dry processes, all pyroprocessing operations take place in the rotary kiln, while drying and preheating and some of the calcination are performed outside the kiln on moving grates supplied with hot kiln gases.

d. Clinker Cooling

The clinker cooling operation recovers up to 30% of kiln system heat, preserves the ideal product qualities, and enables the cooled clinker to be maneuvered by conveyors. The most common types of clinker coolers are reciprocating grate, planetary, and rotary. Air sent through the clinker to cool it is directed to the rotary kiln where it nourishes fuel combustion. The fairly coarse dust collected from clinker coolers is comprised of cement minerals and is restored to the operation. Based on the cooling efficiency and desired cooled temperature, the amount of air used in this cooling process is approximately 1-2 kg/kg of clinker. The amount of gas to be cleaned following the cooling process is decreased when a portion of the gas is used for other processes such as coal drying.

e. Clinker Storage

Although clinker storage capacity is based on the state of the market, a plant can normally store 5 - 25% of its annual clinker production capacity. Equipment such as conveyors and bucket elevators is used to transfer the clinkers from coolers to storage areas and to the finish mill. Gravity drops and transfer points typically are vented to dust collectors.

f. Finish Milling

During the final stage of portland cement production known as finish milling, the clinker is ground with other materials (which impart special characteristics to the finished product) into a fine powder. Up to 5% gypsum and/or natural anhydrite is added to regulate the setting time of the cement. Other chemicals, such as those which regulate flow or air entrainment, may also be added. Many plants use a roll crusher to achieve a preliminary size reduction of the clinker and gypsum. These materials are then sent through ball or tube mills (rotating, horizontal steel cylinders containing steel alloy balls) which perform the remaining grinding. The grinding process occurs in a closed system with an air separator that divides the cement particles according to size. Material that has not been completely ground is sent through the system again.

g. Packing and Loading

Once the production of portland cement is complete, the finished product is transferred using bucket elevators and conveyors to large, storage silos in the shipping department. Most of the portland cement is transported in bulk by railway, truck, or barge, or in 43 kg (94 pound) multiwalled paper bags. Bags are used primarily to package masonry cement. Once the cement leaves the plant, distribution ter-

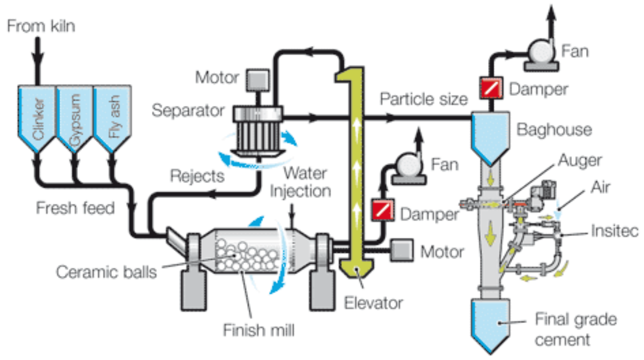


Figure 5.2: Cement mill grinding circuit (FLSmidthA/S (2004)).

minals are sometimes used as an intermediary holding location prior to customer distribution. The same types of conveyor systems used at the plant are used to load cement at distribution terminals.

5.2 Cement Ball Mill Process

The finish milling circuit consumes significant amounts of energy: throughput is high (generally in the range 120 t/h to 150 t/h) and the target particle size is relatively fine. Typically, finished cement is produced using ceramic ball mills, although roller mills are becoming increasingly popular. With either mill it is common for a dynamic separator to take a fine product cut, coarse material being recycled to the mill for further grinding.

Figure 5.2 illustrates a rotating cement ball mill in a closed loop with a separator. The ball mills used for grinding cement have usually two chambers separated by a metallic diaphragm as shown in Figure 5.3. The first compartment is filled with bigger balls and lifting liners and is supposed to do coarse grinding. The second compartment is the fine grinding chamber and is equipped with a smaller ball charge and classifying liners. Classifying liners ensure, that the ball charge is segregated and bigger balls accumulate at the beginning of the compartment and smaller balls toward the end. The feed materials (Clinker,Gypsum and Fly Ash) are fed to the first chamber of the mill where they are broken into smaller powder by the impact of falling balls. Then the particles pass through metallic diaphragm

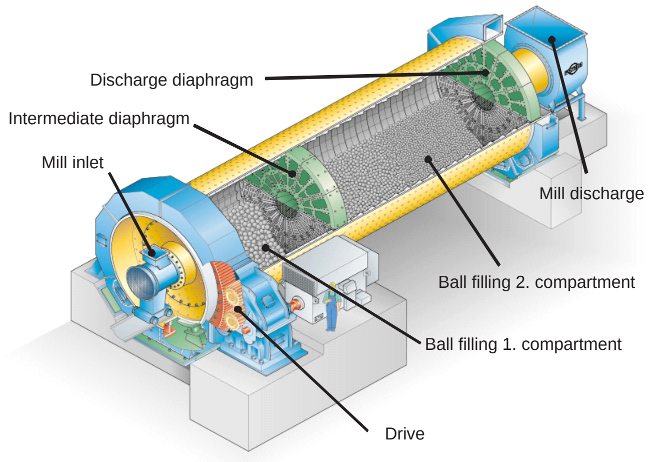


Figure 5.3: Ball mill(FLSmidthA/S (2004)).

into the second chamber. The second chamber consists of much smaller balls for making the particles finer.

The fine ground particles are then lifted by a bucket elevator and sent to a air classifier shown in Figure 5.4. A classifier is essential to the clicker grinding process in order to remove coarse particles, which require further grinding, from the fine particles, which meet specification.

In the classifier, the milled powder is suspended in the air stream. Air is sucked from the bottom of the classifier forcing coarse particles impact on the walls of the classifier. These coarse particles can then be collected as they drop down due to gravity through cone shaped cyclones; whereas fines particles are transported away in the air stream towards the center of the classifier. The rejects are then recirculated to the mill for further grinding. The finer particles then are collected as final product in cement silos. The classifier and the ball mill efficiency can be improved by maintaining a constant recirculation ratio (rejects to main feed ratio), which in-turn provide a consistent product quality.

5.2.1 Tromp Curves

The efficiency of a classifier must be determined to make sure that no oversized particles are present in the final product and that fine particles are not processed

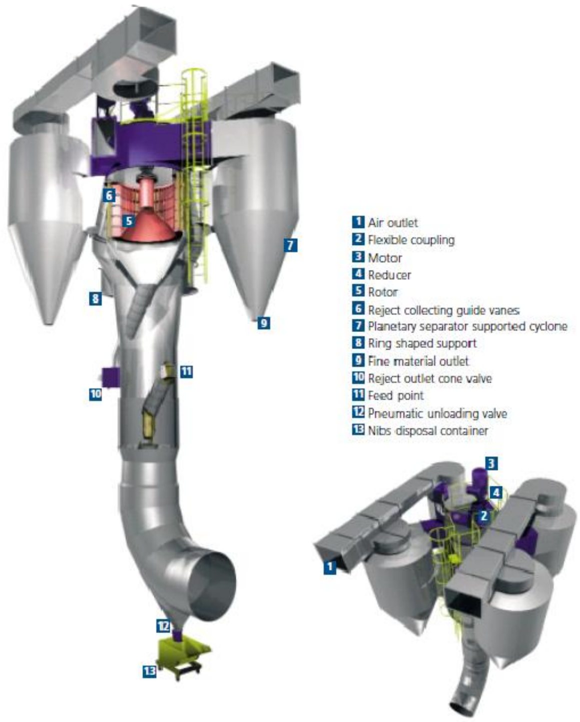


Figure 5.4: Classifier(FLSmidthA/S (2004)).

any further. This can be measured by plotting Tromp curves as shown in Figure 5.5. These plots show the probability that a particle of a particular size is separated into coarse (overflow) fraction. Exact critical examination of the classifier efficiency (sharpness of separation) takes place on the basis of the Tromp curve. This curve is calculated from the percentages by weight of the individual grain size fractions in the three classifier material flows: feed, tailings and fines. The result of the calculation is a selectivity factor (weight by %) for a specific grain fraction. From the entered quantity of points, the function shown in the diagram: "selectivity factor (%) as a function of the grain size (m)" is obtained.

5.3 Cement Mill Control Strategy

The ball mill control is considered to be a very difficult control problem. The main reason for this is the presence of non-linearities and uncertainties in the system. The model predictive control without soft constraints is more effective when the plant model matches with the controller model. Physical and chemical properties

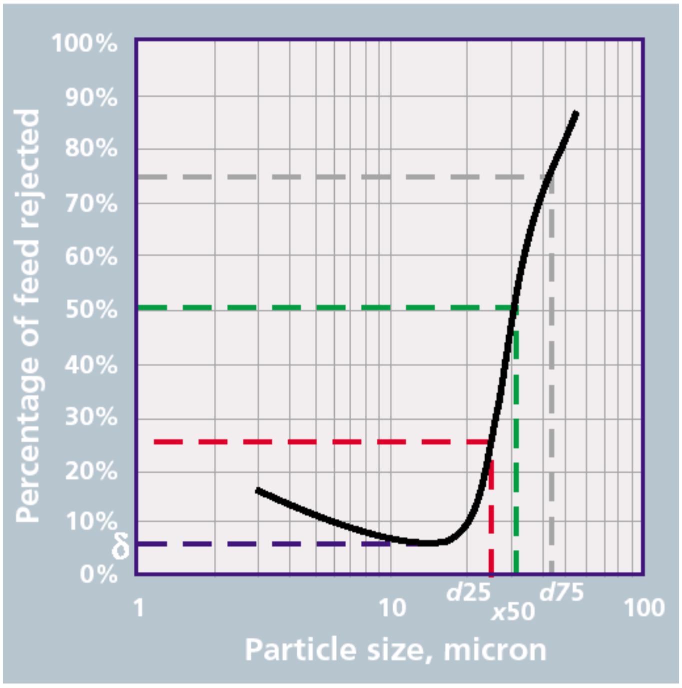


Figure 5.5: Tromp Curve(FLSmidthA/S (2004)).

of the raw material, mechanical wear and tear and various other factors deteriorates the model in long run. This model mismatch is considered for designing a controller which neglects the small variations and acts only to the large process disturbances. Stabilizing cement ball mill involves maintaining load on the mill and the fineness of the product. The mill loading can be represented by different parameters like Sound inside the mill created by collision of metal balls, mill main motor drive power and bucket elevator power. Normally elevator power is considered as the most reliable parameter as it indicates exact amount of ground material coming out of the mill. The amount of feed entering the mill and classifier speed for separating the fine and coarse particles are varied for controlling the above parameters.

5.4 Cement Mill Model

The transfer function model for cement mill relating controlled and manipulated variables is obtained using a simulation software called ECS/CEMulator. The ECS/CEMulator, reflects most of the real time conditions and hence the model can be considered to match a real plant. A detailed description of CEMulator is given in Appendix F. The models are obtained by varying the manipulated variables in step and observe the variations of controlled variables with respect to the step change. Different step sizes in each of the manipulated variables are given in order to get the exact impact on the controlled variables on the actuators. Also the variations in the controlled variables are obtained for different step sizes of Feed and Separator speed in the CEMulator as given in Figures 5.6 and 5.7. In the figure it can be seen that the step disturbances on Feed rate and Elevator load are given in both the directions with 5 % and 10 % variations and also for different operating conditions (Grindability of 28 and 36).

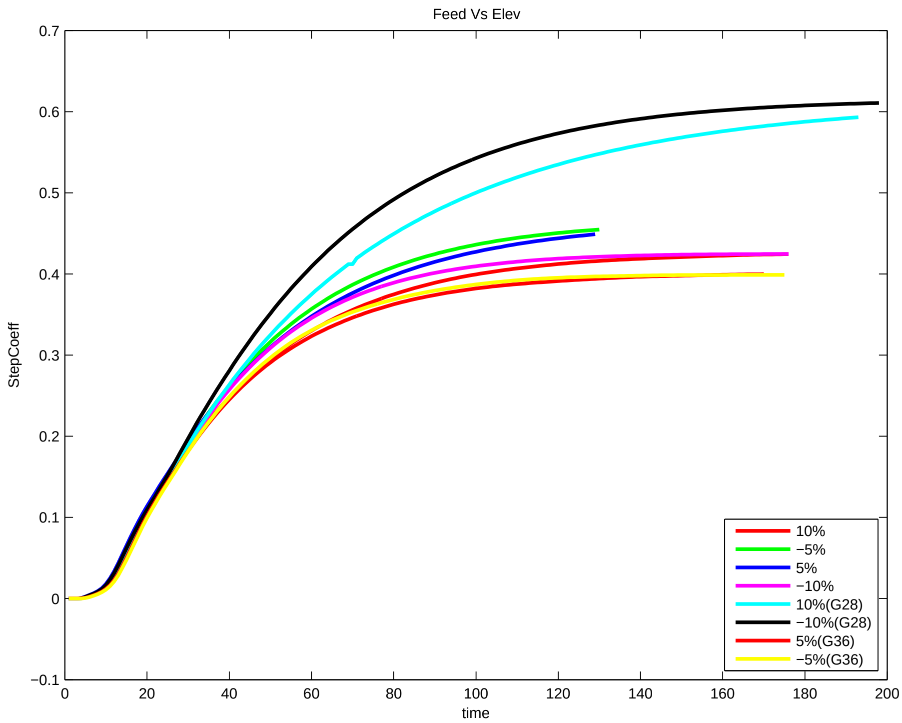
5.4.1 Step Test Procedure

The test procedures specified in the following sections describes the experiments needed obtain the data used for construction of a finite impulse response (FIR) model. Ideally, a model should be established for each operating point

The general principle in the model construction phase is to stabilize the system around its desired operating point and then perturb the system with a step change of each of the process inputs. The responses of the output (measurement) variables are recorded. The process inputs must be perturbed individually. By the data obtained by this procedure a model describing the influence of the process inputs on the process outputs can be constructed. The controller uses this model for computation of the control actions

The cement mill has to process inputs that must be perturbed. The fresh feed flow rate and the separator speed. In principle all possible variables should be recorded. Of particular importance are the Blaine (or residue, on-line as well as lab measurements), Folaphone, Elevator Power (or current), Mill Power and Reject Flow Rate as given in Table 5.1.

Based on these step response results an approximate model is identified, by plotting a step response curve with the transfer function models in the MATLAB as shown in Figure 5.8.



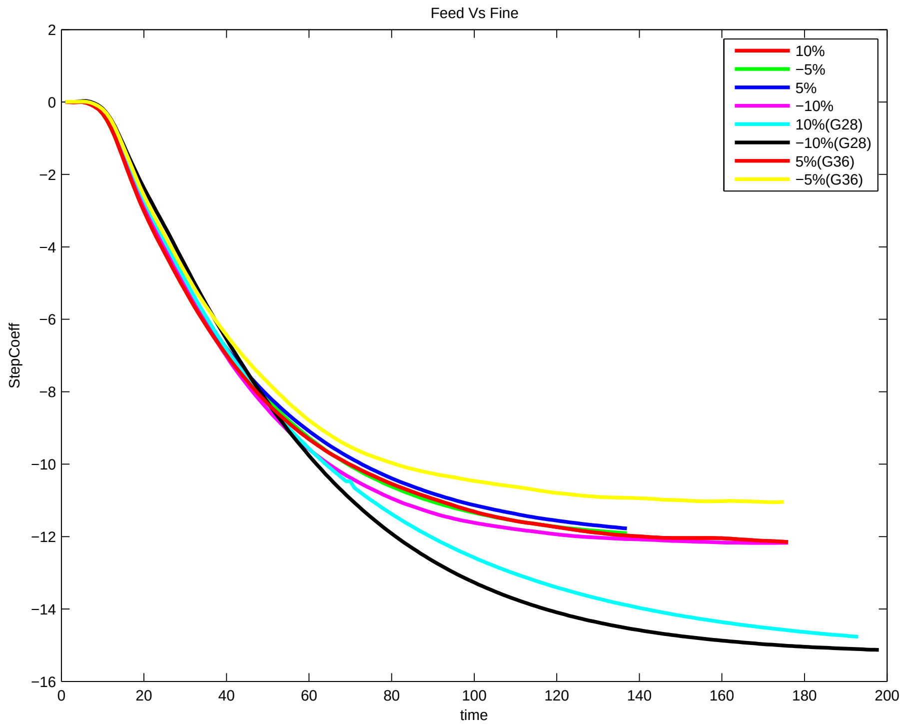


Figure 5.6: Step Response results for different operating conditions(different Grindability factors) and with different step sizes of Feed(+5 %, -5 % ..).

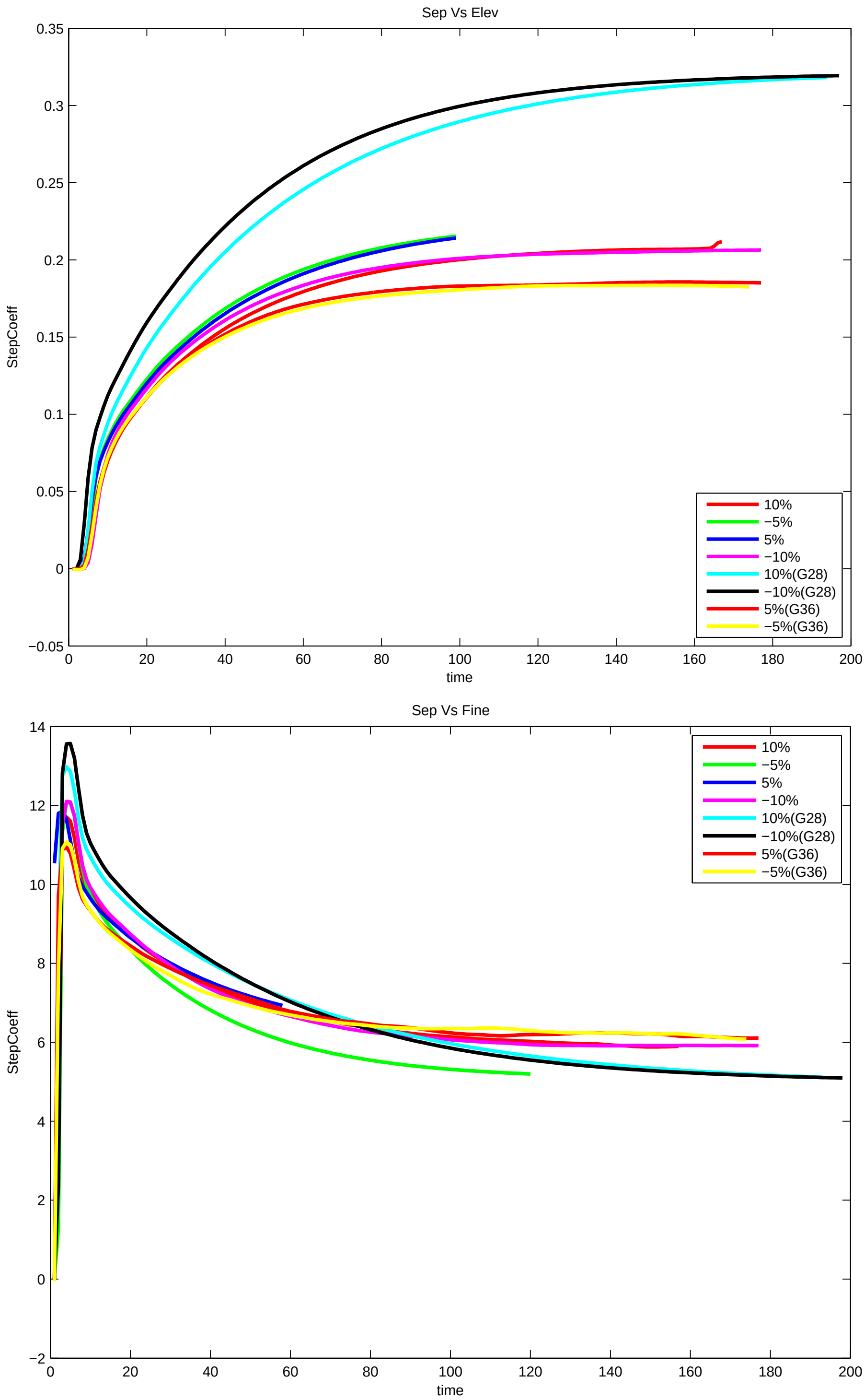


Figure 5.7: Step Response results for different operating conditions(different Grindability factors) and with different step sizes of Separator(+5 %, -5 % ..).

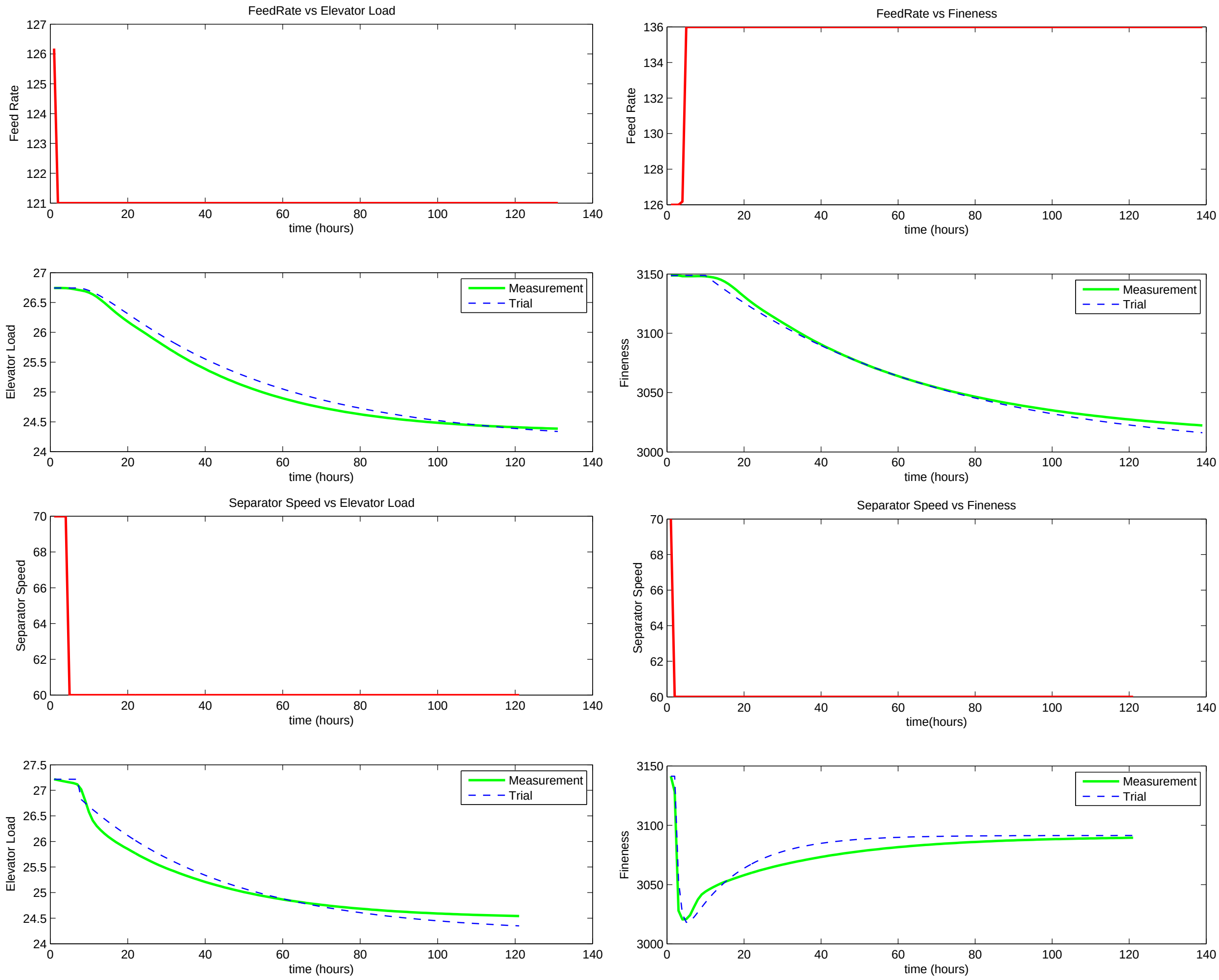


Figure 5.8: Model identification based on the plots obtained from the step response co-efficient(CEMulator data).

Table 5.1: Cement Plant Variables

Process Inputs	Process Outputs
Fresh Feed Rate	Blaine (lab or online analyzer)
Separator Speed	Elevator Power
-	Folaphone
-	Mill Power
-	Reject Flow Rate

The transfer function model for cement mill relating controlled and manipulated variables based on the step test in the simulator can be given by $G(s) = \frac{Y(s)}{U(s)}$

$$G(s) = \begin{bmatrix} \frac{0.62}{(45s+1)(8s+1)}e^{-5s} & \frac{0.29(8s+1)}{(2s+1)(38s+1)}e^{-1.5s} \\ \frac{(-15)}{(60s+1)}e^{-5s} & \frac{5}{(14s+1)(s+1)}e^{-0.1s} \end{bmatrix} \tag{5.1}$$

where

$$Y(s) = \begin{bmatrix} ElevatorLoad \\ Fineness \end{bmatrix} \text{ and } U(s) = \begin{bmatrix} Feed \\ Separatorspeed \end{bmatrix}$$

The transfer function model in Equation 5.1 is fed to the controller. The time constants and time delay values are in minutes. The model developed is used as the case study for controller performances using first the Matlab simulation and by running the controller online in the CEMulator. The main control strategy of cement mill involves controlling of product fineness(blaine) and load on the mill (elevator power) by manipulating feed and separator speed.

5.5 Conclusion

The need for control of cement ball mill grinding circuit and the difficulties involved in control are discussed. Also the cement mill transfer function model is obtained using ECS/CEMulator tool by conducting step tests on the manipulated variables and get the response of the controlled variables. A 2×2 transfer function model is developed in which Feed rate and Separator speed are the manipulated variables and Elevator load and Fineness are the controlled variables.

CHAPTER 6

Applications of Soft MPC to Cement Mill Circuit

In this chapter, the robust MPC developed in chapter 3.7 is used for regulation of a cement mill circuit. The MPC uses soft constraints (soft MPC) to robustly address the large uncertainties present in models that can be identified for cement mill circuits. First the robust MPC is compared with the conventional MPC in a CEMulator system. The soft MPC is also implemented in a real time cement ball mill grinding circuit. The model is obtained from the real plant by conducting step tests in the plant in the same way as done in simulation system. The controller performance is compared with that of the Fuzzy logic control already running in the plant.

6.1 Introduction

In the cement industry, the clinker grinding step consumes about one-third of the power required to produce 1 ton of cement (Touil *et al.*, 2008). Grinding is a high-cost operation consuming approximately 60% of the total electrical energy expenditure in a typical cement plant. 30% of the electrical energy is used for raw material crushing and grinding while around 40% of this energy is consumed for grinding clinker to cement powder (Fujimoto, 1993; Jankovic *et al.*, 2004). Hence, global cement production uses 18.7 TWh which is approximately 2% of the worlds primary energy consumption and 5% of the total industrial energy consumption.

Consequently, optimization and control of their operation are very important for running the cement plant efficiently, i.e. minimizing the specific power consumption and delivering consistent product quality meeting specifications.

The MPC algorithm uses soft constraints to create a piecewise quadratic penalty function in such a way that the closed loop system is less sensitive to model uncertainties than conventional MPC with a quadratic penalty function. Predictive models for cement mills are very uncertain. Variations and heterogeneities of the cement clinker feed affects the gains, time constants, and time delays of the cement mill circuit. To avoid the MPC being turned off shortly after commissioning due to bad closed-loop behavior, it is important that these unmeasurable uncertainties are accounted for. The developed soft MPC controls elevator power and blaine of the cement (product quality) by manipulating the fresh feed flow rate and the separator speed. The elevator power can be measured in almost all cement mill circuits. Blaine can either be measured by online particle size analyzers or estimated using soft sensors. The models relating the manipulated variables to the controlled variables are identified from step response experiments on the cement mill circuit.

Van Breusegem *et al.* (1994, 1996b,a) and de Haas *et al.* (1995) have developed an LQG controller for the cement mill circuit. This controller was based on a first order 2×2 transfer function model identified from step response experiments. Magni *et al.* (1999), Wertz *et al.* (2000), and Grognard *et al.* (2001) have developed a Nonlinear Model Predictive Control algorithm based on a lumped nonlinear model of the cement mill circuit. All these controllers controlled the product and recycle flow rate by manipulating the fresh feed flow rate and the separator speed. Efe and Kaynak (2002) have used the same lumped nonlinear model for nonlinear model reference control. Lepore *et al.* (2002, 2003, 2004, 2007a) as well as Boulvin *et al.* (1998, 1999, 2003) have applied a distributed reduced order model for Nonlinear Model Predictive Control of a cement mill circuit. They controlled the particle size distribution of the cement product by manipulating the fresh feed flow rate and the separator speed. Martin and McGarel (2001) have used a neural network model for Nonlinear Model Predictive Control of the cement mill circuit.

6.2 Cement Mill control Simulation results

The controllers are being compared by simulating using a tool called CEMulator, which represents the real time cement plant. FLSmidth Automation has developed the product which is an absolute realistic simulator of cement plant processes. The models are obtained by conducting step tests on the manipulated variables in the CEMulator as given in chapter 5.

The controllers are simulated in the CEMulator in the same way as MATLAB simulation as given in chapter 4.6. The major difference in CEMulator simulation is that the process reaction time in the CEMulator are similar to the real plant and hence the simulation has to be carried out for a longer period (in hours). Also the CEMulator model is a non-linear model in which the linear controller is tested. In case of MATLAB simulations, the controller is simulated at much faster rate (in seconds) and the plant model assumed is a linear state space model. Also the disturbances in the MATLAB simulations are given as simple step disturbances. These disturbances are deterministic, simpler and is assumed to have the same transfer function as the plant model. In case of CEMulator, the disturbances are uncertain as in the real plant. Thus the simulation environment for comparing both the controllers are entirely different in CEMulator with respect to MATLAB simulation.

Figure 6.1: Performance of MPC with grindability factor of 36 without measurement noise(Sepax Power Changed from 330-360 Kw - Green line).

Initially the performance of the MPC is tested by varying the grindability of the material fed into the cement mill. Grindability is the measure of hardness of the clinker used as the raw material for grinding. Increasing the grindability will make the clinker soft and can be easily ground. Also this will change the operating conditions of the cement mill as the efficiency of the grinding will be increased with increase in grindability and also the reverse case. First as a study of robustness the model obtained from step response test from the simulator as explained in the previous chapter is taken as the controller model. The grindability factor of clinker for the model in the previous chapter is 33, which is considered to be the nominal value. In Figure 6.1, the controller is made online by increasing the grindability factor from 33 to 36. Also the measurement noise is removed from the simulation. It can be seen that even with different operating conditions the controller performance remains almost constant stabilizing the system within 30 min from the disturbance.

In order to analyze the performance of the controller with lower grindability or hard clinker and also with measurement noise, the controller is made online with grindabilty factor of 28. Along with such a disturbance the sepax separator fan power is varied from 390 Kw to 300 Kw and again brought back to 300 Kw. This will in-turn directly affect the fineness of the final product as the fan power is used to lift the material from the separator. It can be seen from Figure 6.2 the controller stabilizes the system without much variation in the output for different operating conditions.

The simulation is carried for a minimum of 8 hours to observe the exact variations in the system when the controllers are running. This provides a very good comparison. The disturbances and the measurement noise are included in the simulation in the same way as it occurs in the real time system.

As discussed earlier, the performance of the controller in real time situations provide significant comparison. Here a case of uncertainty is considered for comparing the performance of the controller. The uncertainties of the system can be represented in terms of gain and time delay of the system. Thus based on experiments

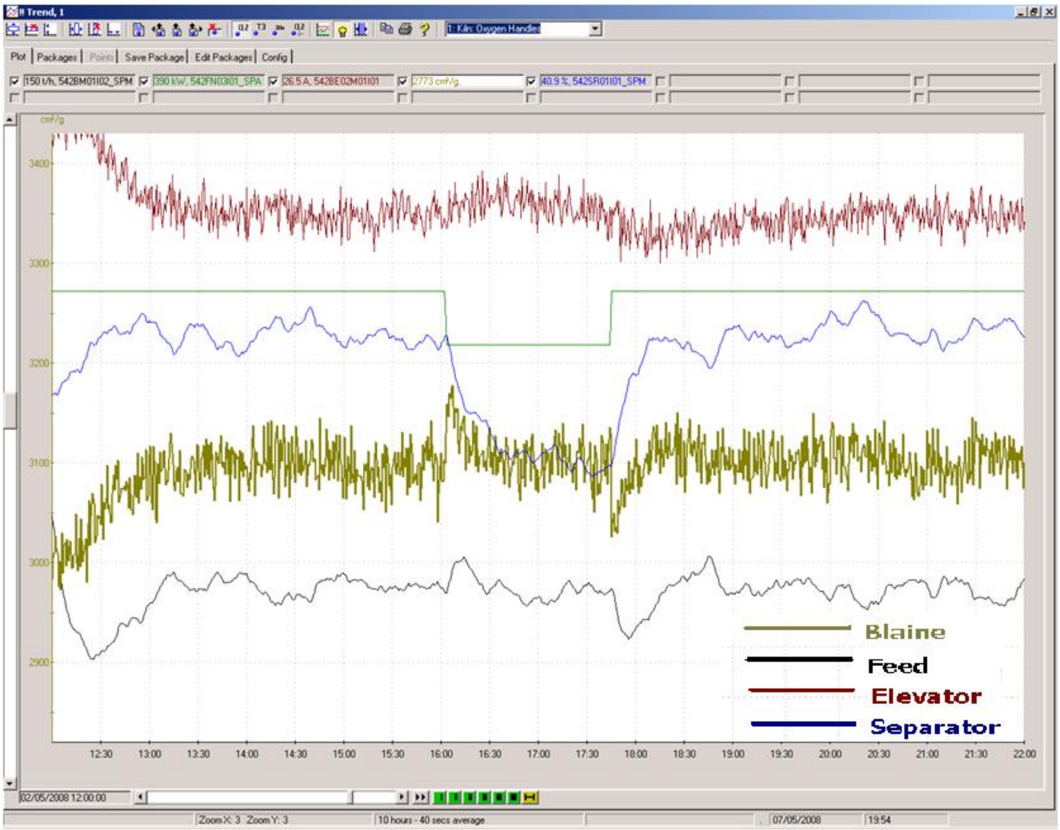


Figure 6.2: Performance of MPC with grindability factor of 28 and sepax power changed from 390- 300 KW(Green Line).

the system transfer function in Equation 4.4 is modified as

$$G(s) = \begin{bmatrix} \frac{0.2}{(45s+1)(8s+1)}e^{-5s} & \frac{0.12(8s+1)}{(2s+1)(38s+1)}e^{-1.5s} \\ \frac{(-8)}{(60s+1)}e^{-5s} & \frac{2}{(14s+1)(s+1)}e^{-0.1s} \end{bmatrix} \quad (6.1)$$

The time constants and time delay values are in minutes. A normal MPC with an ℓ_2 penalty function and a Soft MPC using an ℓ_2 penalty function with an almost dead zone as illustrated in Figure 3.29 are designed. The control targets are same for both the controllers. The target for Elevator Load is 30 and Fineness is 3100. The soft limits for Elevator Load are $Z_{min} = 28$ and $Z_{max} = 32$, and for Fineness the soft limits are $Z_{min} = 3000$ and $Z_{max} = 3200$. Also both the controllers start from the same steady state operating point as seen in Figure 6.3. The steady state values for the manipulated variables are Feed = 126 tonnes/hour, Separator speed = 70 % and for the controlled variables are Elevator Load = 26 and Fineness = 3100. Thus both the controllers are kept in same operating conditions to have a fair comparison.

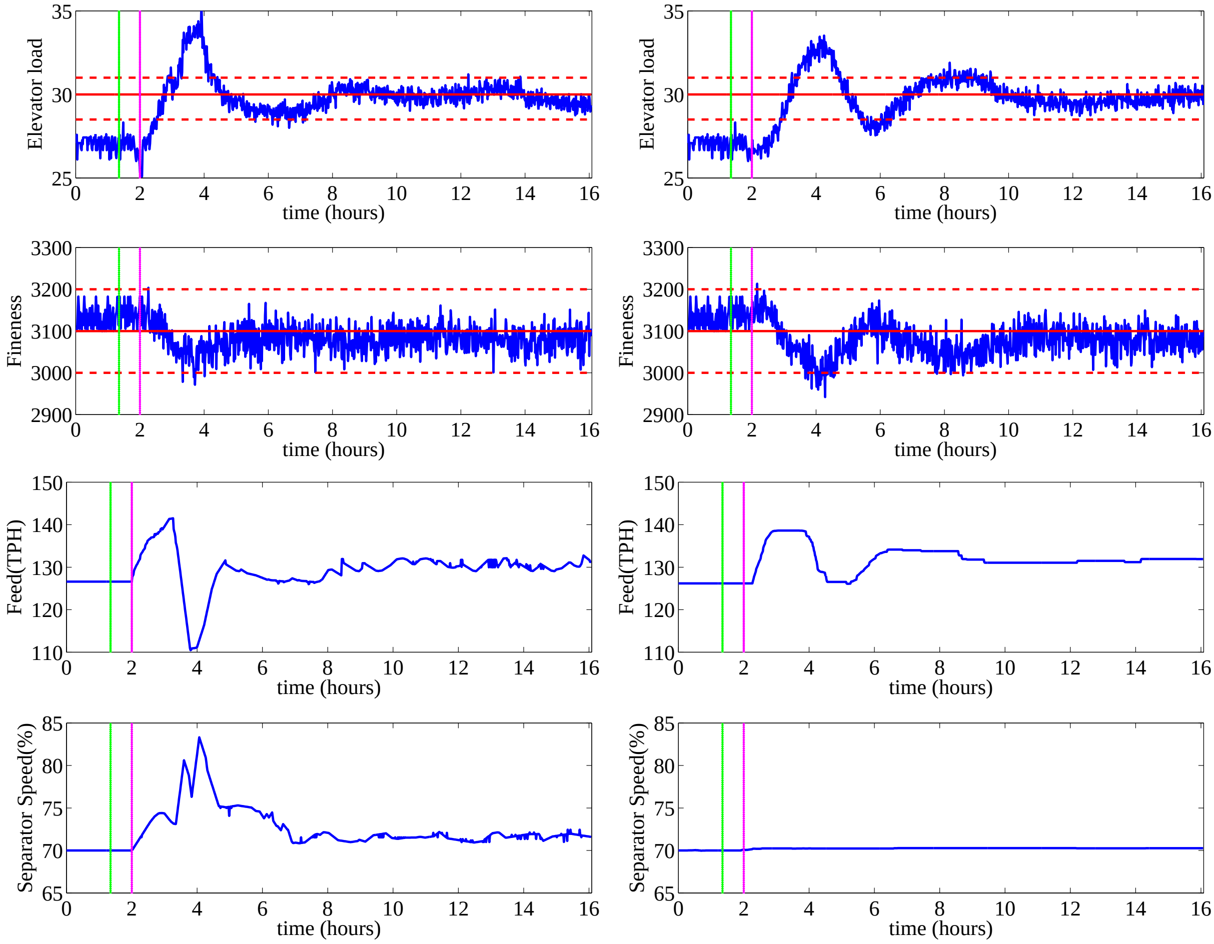


Figure 6.3: Conventional MPC(left) and Soft MPC(right) applied to a rigorous nonlinear cement mill simulator. The disturbances (change in hardness of the cement clinker) are introduced at time 1.35 hour (green line) and the controllers are switched on at time 2 hour (purple line) The soft constraints are indicated by the dashed lines.

The plant is considered to be stable, except the mismatch in the model parameters with reference to the controller model. This enables to compare the performance of the controllers with reference to only the model uncertainty. Using ECS/CEMulator from a steady state, a significant change in hardness of the cement clinker (as disturbance) is introduced at time 1.35 hr, and the controllers are switched on at time 2.0 hr. The resulting closed loop profiles for the Normal MPC and Soft MPC are illustrated in Figure 6.3. It is evident by the simulations that the variation of the output variables are more or less comparable for the two MPCs, but the Soft MPC manipulates the MVs in a more plant friendly manner than the

Normal MPC. The unnecessary actuator variations are reduced to large extent in case of MPC with soft constraints resulting in better stability of the system. Also the variance of feed rate and separator speed seems to be high incase of MPC without soft constraints when compared with soft MPC. It can be seen from Figure 6.3, the variation of Feed rate is from 140 tonnes/hour to 110 tonnes/hour in case of conventional MPC , whereas in case of soft MPC the maximum variation in Feed rate is 138 tonnes/hour and minimum 125 tonnes/hour which is almost 15 tonnes/hour (approximately 10%) lesser than conventional MPC. In case of separator speed it is up to 84% in conventional MPC whereas the separator speed varies only a little around the steady state value in soft MPC. Thus, these variations in manipulated variables will disturb the process significantly in real time. The optimization of the cement mill circuit with minimum separator variation is considered as efficient. This is because frequent variation in separator will disturb the recycle load, which will in-turn affect the efficiency of the ball mills. Also frequent variation in separator motor can cause mechanical/ electrical wear reducing the rate of separation. This will result in huge variation in Final product residue. (based on 22 micron sieve). Consequently, most practitioners would prefer the Soft MPC to the Normal MPC as it gives rise to less plant wear.

6.3 Real Time implementation

The controller is tested in a real time plant and compared with that of the already running high level controller based on Fuzzy logic principle.

The cement mill present in the plant is a closed circuit ball mill with two chambers. The cement ball mill has a design capacity of 150 tonnes/hour with sepax separator. The recirculation ratio of the circuit is 1.5 %. The final product recipes are Ordinary Portland Cement (OPC) and Puzzalona Portland cement (PPC). Since the plant where the controller is implemented in an independent grinding unit, the clinker produced from various other plants are transported through wagon for grinding. Thus the quality variations in the clinker are very large depending on the supplier. This results in continuous operational variations in the cement mill circuit. Also depending on the dispatch requirements the cement mill grinding recipes of final products need to be changed resulting in frequent shifting of operating points of the controller.

All the signals coming from the sensors of the grinding process are collected in a ECS SCADA (Supervisory Control And Data Acquisition of FLSmidth) system. The measurement data is obtained from PLC and logged every 10 seconds in the system. The quality measurement data (Fineness/Blaine) is sampled every hour using the samples collected through Auto- Sampling system.

The real time implementation of soft MPC application is done using a high level expert system tool developed by FLSmidth. The execution interval of MPC will be 1 min and the data update in the expert tool will be 30 seconds.

To have a common platform for comparison, the FLC and soft MPC are made online in similar operating conditions and the adaptive feature was made common for both the controllers. Here the adaptive controller changes the target on Elevator load which is considered to be the primary controlled variable, based on the quality variations in the output. Here Fuzzy calculation engine also executes every 30 seconds. The measurement data obtained from the PLC through input/output modules in the field is filtered, scaled and validated before used in the controller. The output from the controller is also scaled and configured for bumpless transfer when it is made online, this is important in order to have smoother transition of set points when the controller is shifted from manual to auto control loop. Interlocks are included incase of emergency shut down during abnormal conditions like power failure etc.

The models of the system are taken by doing step response tests in the real time system open loop. The main difference between the models obtained in real time with that of the ECS/CEMulator system as in chapter 5 is that the model from the real plant are uncertain and it is quite difficult to obtain a proper response on input and output. So a number of step tests are conducted with each of the manipulated variables and the responses are plotted to obtain an approximate model.

The general principle in the model construction phase is to stabilize the system around its desired operating point and then perturb the system with a step change of each of the process inputs. The responses of the output (measurement) variables are recorded. The process inputs must be perturbed individually. By the data obtained by this procedure a model describing the influence of the process inputs on the process outputs can be constructed. The controller uses this model for computation of the control actions.

Here, the feed, separator speed are perturbed and all the possible values of measurements are obtained. The lab measurement (Fineness) is modeled by increasing the frequency of sample collection i.e., collecting samples for every 15 min and then generating a model based on the data. Based on these step response results an approximate model is identified, by plotting a step response curve with the transfer function models in the MATLAB as shown in Figure 5.8

We consider 2×2 MPC controllers based on the models Y (s) = G(s)U(s) with Y (s) = [Elevator Load; Fineness] and U(s) = [Feed Rate; Separator Speed]. Based on the tests we identify the system transfer function as in (Equation 6.2)

$$G(s) = \begin{bmatrix} \frac{(0.47)(2s+1)}{(17s+1)(15s+1)}e^{-4s} & \frac{1}{12s+1}e^{-3s} \\ \frac{(-0.9)}{(10s+1)(12s+1)}e^{-5s} & \frac{2.5}{(4s+1)} \end{bmatrix} \tag{6.2}$$

Since the quality of the feed material vary frequently, we include target adaptation for improving the grinding efficiency of the ball mill. Thus the target of elevator load considered for controlling will be changing depending on the quality of the fine material. This is mainly to achieve optimum production while achieving the desired fineness.

The controller is made online with Feed and Separator Speed controlling Elevator Load and Fineness. First the system is made online with Fuzzy controller and then with soft MPC when the cement mill was running continuously in a single recipe, here it was producing PPC where Gypsum, Clinker and Fly Ash are the feed materials. This is just to have a basic comparison to be in same operating conditions. The following tuning and weighting factors are used while applying

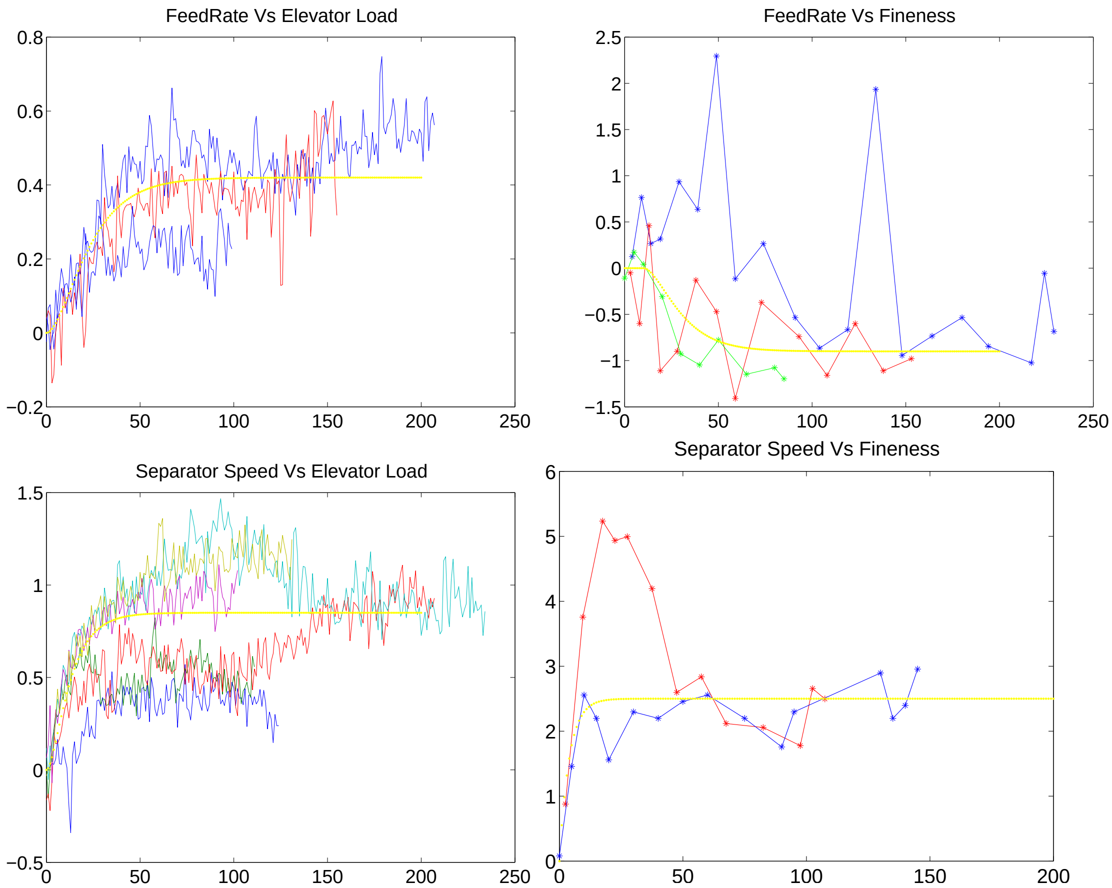


Figure 6.4: Model identification from the plant data from step response tests. The identified model is yellow solid line. The other lines indicate plot of real time data with various step tests conducted. The yellow line is based on the identification of the model with the other data plotted.

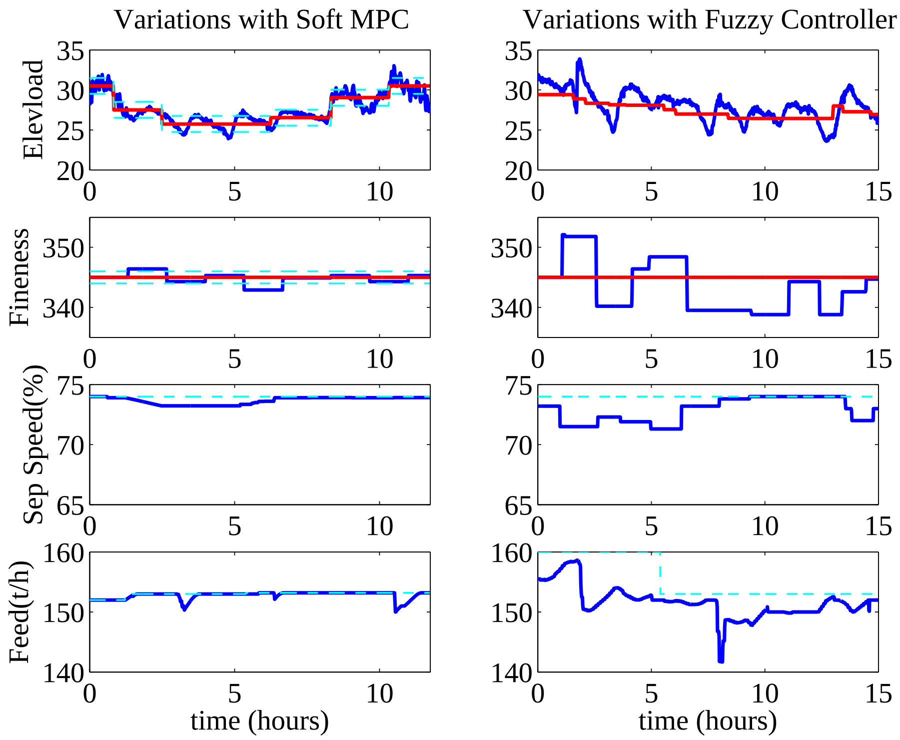


Figure 6.5: Comparison of Soft MPC (left) with the High Level controller (right) implemented in a closed loop cement mill Target (Red), Actual value (Blue), High and Low Limits(dashed line).

the soft MPC scheme to control the grinding circuit: Prediction horizon $N_p = 300$, control horizon $N = 100$, The weights on the errors are

$Q_z = \begin{bmatrix} 0.005 & 0 \\ 0 & 0.000025 \end{bmatrix}$, the weights on the manipulated variables

$S = \begin{bmatrix} 500 & 0 \\ 0 & 250000 \end{bmatrix}$ and the quadratic soft constraint weights are

$S_\eta = \begin{bmatrix} 9000 & 0 \\ 0 & 500 \end{bmatrix}$. Here we consider the linear soft constraint weight to be zero.

The weight on Fineness is small when compared to elevator load, as the fineness is a hourly sampled data and less reliable when compared with elevator load. Also the weight on separator speed is set to a large value because it is not permitted to move too freely in order to maintain a relatively stable operation.

From Figure 6.5, it is seen that the above soft MPC has the abilities of prediction and coordination, which could lead to much smoother responses of manipulated variables while maintaining the controlled variables stable. However in Figure 6.5, the controller runs at its high limits and hence the actuator movements are restricted to move on the higher side. These conditions occur because of the conditions in the plant where there is inconsistency in quality variations of the raw material fed into the mill . Even then, because of the smaller variations in the separator speed the standard deviation of the fineness has improved significantly incase of soft MPC when compared with the Fuzzy logic control. This is achieved because of the smoother variations in both the actuators.

Separator speed has been given higher priority in the soft MPC, the main reason is, that the separator cannot be varied frequently because it's a LT motor device, frequent variations in separator will result in reduced life time, also it results higher variation in residue of the final product which is not measured frequently but has to be maintained within the band as per the quality norms. The variations in separator speed also affect the recycle load which has a very big impact on the performance of the ball mill. Thus such variations in separator will reduce the efficiency of the ball mill in longer time.

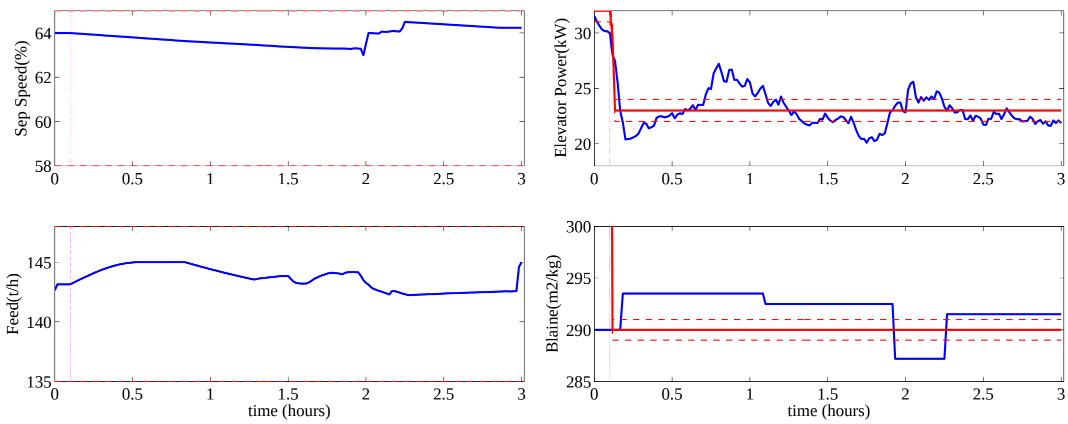


Figure 6.6: Actuator (left) and measurement (right) variations in soft MPC when the control parameters are inside and outside the soft constraint limits (dotted lines) when made online with the cement mill recipe changed to OPC (pink line)(Real Time Results).

Figure 6.6 gives the performance of soft MPC, where there is large margin for the controller to adjust its actuator to maintain the desired target of controlled variables. Here the soft MPC controller moves the actuator very little within the soft constraints and takes aggressive actions outside the soft limit band resulting in better stability and optimization of the cement mill circuit. The performance of the soft MPC has been significantly better for the changes in operating regions and also the quality variations.

The main difference in controlling the cement mill through a CEMulator and real plant is, the CEMulator is a non-linear simulation tool that includes all the dynamics of the system but the disturbances can be fixed and recorded. Also there is no uncertainty considered in the simulation and the system behavior is similar in all the situations. In real plant, the uncertainty in the process is common. This is because of the changes in operating conditions due to variations in the raw materials, wear and tear of the mechanical devices and various other reasons. In the next chapter, the performances of such controllers when applied to a cement mill with large sample delay in the fineness measurement is studied.

6.4 Conclusion

By simulation, using a rigorous process simulator and also implementing the controller in a real time plant, it is demonstrated how the Soft MPC is used to implement this control strategy despite the significant plant-model mismatch which is unavoidable in the control of cement mill circuits. When the controllers are compared with the cement mill simulation system it is seen that variations in output variables are similar in both the controllers, but the soft MPC achieves the variation with smaller actuator moves. The standard deviation of feed rate in conventional MPC is 4.4258 and separator speed is 2.4149.The standard deviation of feed rate in soft MPC is 3.269 and separator speed is 0.0878. Also the initial variation in feed rate and separator speed are high in case of conventional MPC when compared with soft MPC. It is observed that when both the controllers are made online in a real time cement mill circuit with similar process conditions and in the same recipe, the standard deviation in the quality parameter (i.e Fineness of the product) is reduced by around 23% with soft MPC when compared with that of FLC in the plant.

CHAPTER 7

Implementation of Soft MPC to a Large Sample Delay System

In this chapter, the performance of a model predictive controller (MPC) applied to a cement ball milling circuit with large measurement sample delay is investigated GuruPrasath *et al.* (2011). Maintaining fineness quality of the cement produced is important in the design of such controllers. Usually fineness is measured hourly by sample analysis in the laboratory. The predictive controller designed with the model from fast sample data (1 min sample) when applied to control with hourly sampled measurements, the controller is to be re-tuned a number of times (at least 3 - 4 times) for improving the performance of the controller. The parameters needed to be re-tuned are weight for penalties on measurement error (Q_z), weights of penalties on manipulated variables (S) and rate of change of manipulated variables (R). Also the hard constraints on input- rate movements are to be re-tuned to reduce the variations in the control actions. In this work, the controller model is added with half the sample delay of measurement to improve the performance of the controller with one time tuning.

7.1 Introduction

In all the discussions in previous chapters, the process is assumed to have all the measurements continuously available at each sampling instant. But most of the real time systems do not have continuous measurement for certain important parameters like quality. Usually quality parameters are results of analysis being done in the lab and will be available only when the lab analysis is completed. This may take several minutes to hours depending on the type of analysis.

For grinding circuits characterized by large time delays, a predictive control is more suitable in this case (Chen *et al.*, 2009). Conventionally, predictive controllers are more accepted in real time control of cement mill circuits because of their proven control methodology and robustness. Usually, the models used in the controller are obtained from continuous measurements obtained by step tests on the cement mill system. In the real plant, the elevator load is a continuous measurement where as the quality measurement Fineness or Blaine is collected every 1 hour using auto sampler, analyzed and entered as off-line measurement. This may result in getting a measurement every hour after the sample is analyzed and this will be available for control. Thus, when conventional controller applied to such measurement the controller parameters need to be re-tuned quite often to improve the controller performance.

A finite impulse response (FIR) model based MPC is considered for controlling the cement mill circuit. The cement mill control involves the control of mill loading and fineness. The main measure of fineness is specific surface. Because cement particles react with water at their surface, the specific surface area is directly related to the cement's initial reactivity. By adjusting the fineness of grind, a range of products can be produced from a single clinker. Tight control of fineness is necessary in order to obtain cement with the desired consistent day-to-day performance. Thus fineness control is one of the most important parameter in cement mill control circuit. Fineness values can be obtained approximately once in an hour as it is measured in lab by sample analysis. But the models on elevator load representing the loading on the mill and fineness for the controller are obtained at each sampling instant from step response experiments conducted on the cement mill simulation circuit. When such controllers are taken online to control hourly sampled measurements the control performance becomes poor and the control parameters need to be re-tuned as discussed above. These kind of problems can be solved as an multi-rate control problem. But here the problem is considered as a sample measurement delay problem and a method is proposed to effectively address the large sample delay measurements which significantly improve the performance of the controllers without much effort on tuning the system.

A method is proposed, in which the controller model is added with half the sample delay of the measurement and the performance of the controller is analyzed.

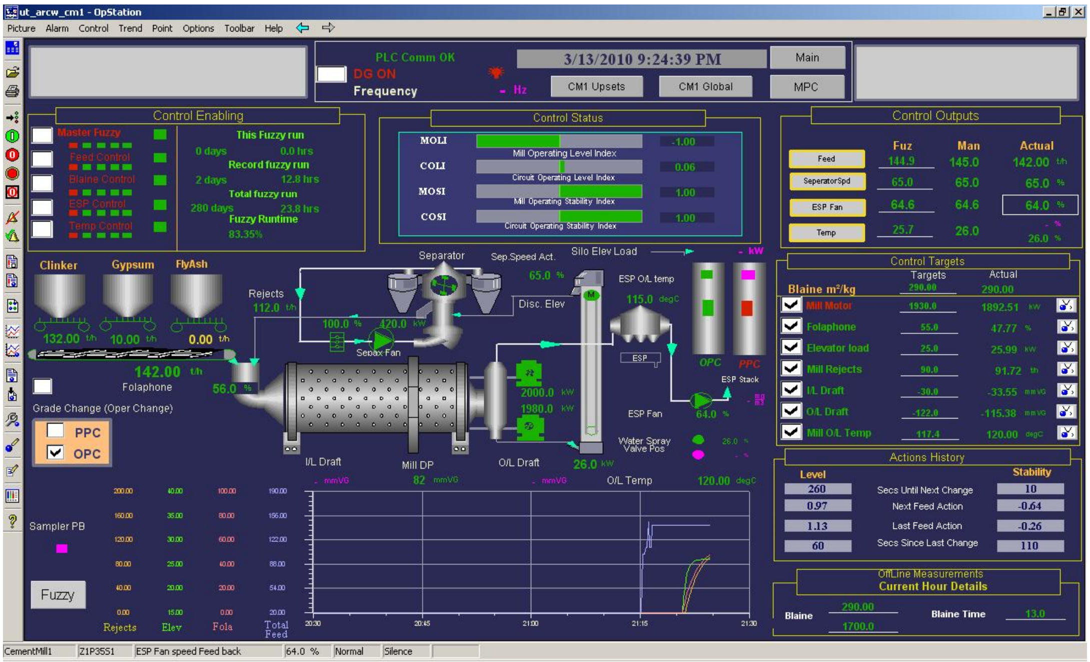


Figure 7.1: Operator station of High Level Control for closed loop cement mill FLSmidthA/S (2004).

7.2 System Implementation

The schematic diagram of ball mill control is shown in Figure 7.1.

The controllers are designed for handling different recipes where the operating parameters/ targets will be changed depending on the type of recipe. The main control loops considered are Feed rate and Fineness control. In addition for improving the performance of cement mill operation and to achieve better stability in the system, some of the control variables like differential pressure across the diaphragm, outlet temperature of the material and load on the separator fan are to be controlled. These variables even though do not contribute in cement mill performance significantly, the fluctuations in these variables cause the mill to be unstable. Such control variables can be handled by simple SISO control/ PID loops.

As discussed, the most important parameters that determine the performance of the mill is the amount of material loading inside the mill and the quality of cement being produced. Quality of final product in the cement ball mill remains one of the challenging and important parameter to be controlled. The quality is measured in terms of fineness of the cement by collecting samples from the final product and entered manually every hour in the supervisory system. A 2×2 MPC system is considered based on the models as given in section 5.4. The controller actions can be tuned from the tool available in the SCADA, called as Process Expert tool. This is a visual basic programming environment interfaced with other tools like Matlab, Excel etc., In this tool the tuning parameters can be directly entered from the available ECS/SCADA and the values can be taken to the PLC for overwriting the manual actions. Most important tuning parameters in the MPC are penalty on the actuator movements to restrict the amount of control actions and weight on the measurements to make the controlled variables to reach the desired targets.

The controller with model obtained from continuous sample data is made online in the ECS/CEMulator system with Feed and Separator Speed controlling Elevator Load and Fineness. Fineness of the cement controlled here is assumed to be hourly sample collected from the final product belt through an auto- sampler or manual spot collection and the samples are analyzed and entered from the quality lab. Thus the fineness value is updated in the system with a delay of one hour.

The following tuning and weighting factors are used, while applying the MPC to control the grinding circuit when the fineness measured using an online analyzer: Prediction horizon $N_p = 250$, control horizon $N = 50$, The penalties on the errors are

$Q_z = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.015 \end{bmatrix}$, the penalties on the manipulated variables are

$S = \begin{bmatrix} 100 & 0 \\ 0 & 1000 \end{bmatrix}$.

When the above tuning parameters are used for controlling, the hourly fineness data the performance of the controller degrades, thus for improving the performance of the MPC the control weights are needed to be re-tuned by trial and error after a number of times (here the stable operating parameters are obtained after 4 trials) as

$$Q_z = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.001 \end{bmatrix}, \text{ the weights on the manipulated variables}$$

$$S = \begin{bmatrix} 100 & 0 \\ 0 & 10000 \end{bmatrix}.$$

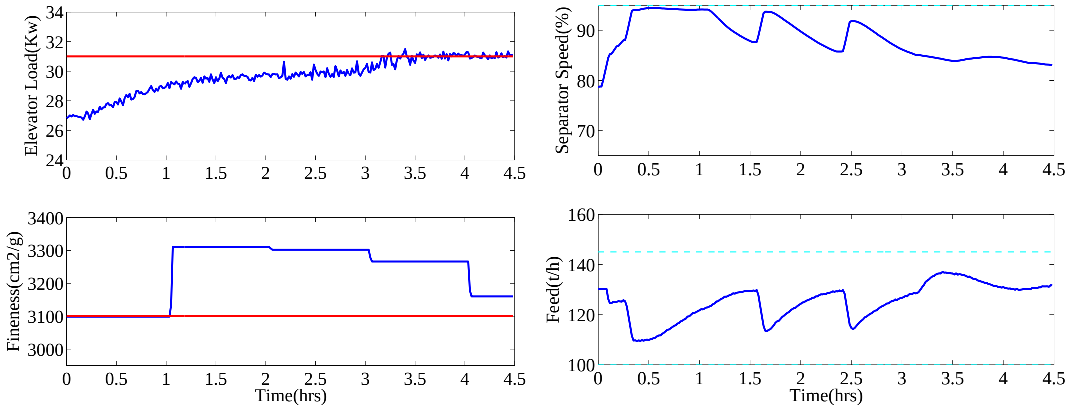


Figure 7.2: Variations in MPC with nominal models when applied to measurements having large sample delays, Target(Red), Actual Value(Blue).

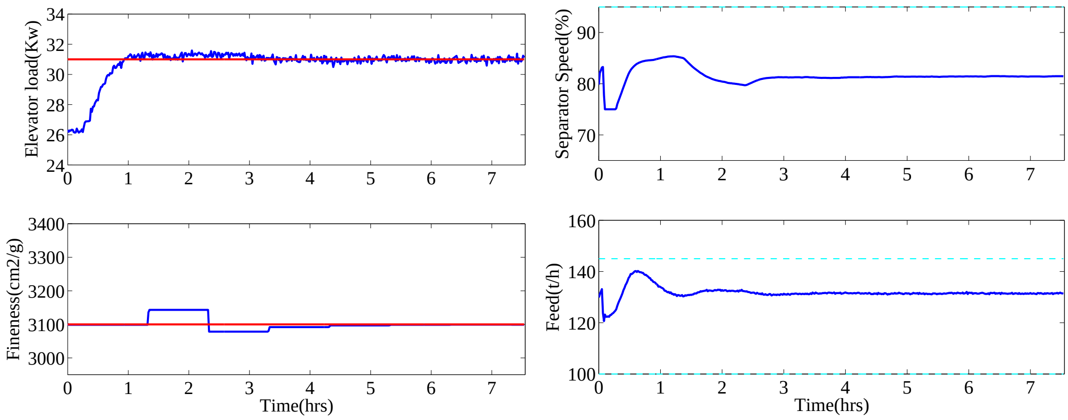


Figure 7.3: Variations in MPC with models including sample delay added to continuous model and taken online for controlling the large sample delay parameters(Fineness), Target(Red), Actual Value(Blue).

These tuning values are obtained by trial and error method, depending on the control actions of the controller. It can be seen that the penalty on Fineness is very small, this is because the fineness is a hourly sampled data and less reliable when compared with elevator load. Also the penalty on separator speed moves is set to a large value because varying the separator continuously causes wear in the separator blades thus reducing the separation efficiency in the long time. Thus the control actions become less aggressive as the penalty on the actuator movement (especially separator speed) is increased. This results in slower control actions on both fineness and elevator load making the controller sluggish. From Figure 7.2, it is seen that the controlled variable Elevator load settles at approximately 3.5 hours and there is an steady state offset in case of Fineness measurement. The actuator variations are also undesirable as it is quite oscillatory.

In order to avoid such a situation, the controller model for Fineness versus Feed and separator speed are added with time delay which is considered to be approximately half the sample delay of Fineness measurement. For example, with an hourly sample measurement of fineness we include a delay of 30 min in the fineness model of the controller as in Equation 7.1

$$G(s) = \begin{bmatrix} \frac{0.62}{(45s+1)(8s+1)}e^{-5s} & \frac{0.29(8s+1)}{(2s+1)(38s+1)}e^{-31.5s} \\ \frac{(-15)}{(60s+1)}e^{-35s} & \frac{5}{(14s+1)(s+1)}e^{-0.1s} \end{bmatrix} \tag{7.1}$$

When the predictive controller uses the model as in Equation 7.1, the performance of the controller significantly improve. The comparison of the controller with inclusion of half the sample delay in the model with that of the conventional controller is shown in Table 7.1. Also the proposed controller stabilizes the process much faster when compared to the conventional predictive controller. The penalties on the actuator and the control weights are kept the same that is used for nominal model. Thus the tuning of the control weights is done only once in this case. From Figure 7.3 it is seen that the Elevator load settles approximately in an hour and the standard deviation of Fineness is close to zero. Thus it is observed that the inclusion of half the sample delay in the controller model significantly reduces the tuning of system and also improves the stabilization of the process.

7.3 Conclusion

A method to use the MPC in an efficient way, for handling the large sample delay cement mill systems is provided. When both the controllers are applied for the control on cement mill using CEMulator with large delay measurements, the

Conventional Control(MPC)	**MPC with Sample Delay**
1. Model from CEMulator used for real-time control	1. Model is added with approximately half the sample delay
2. Controller need to be retuned for parameters like actuator movement weight(S), weight on measurement error (Q_z) etc., for each of the controlled variables and the parameters are to be re-tuned based on trial and error method	2. Only the transfer function to be changed and no further tuning required. This alters the bias term in the estimator used as controller feedback
3. Provide sluggish operation or act every hour	3. Changes in model affect the FIR co-efficients and prediction calculations GuruPrasath and Jorgensen (2009), by altering the hessian matrix values in the optimization solution.

Table 7.1: Comparison of Conventional control and Controller with sample Delay.

controller with continuous sample model needs to be re-tuned a number of times ($3-4$ times in this case) in order to improve the controller performance on different conditions. The settling time is approximately 3.5 hours in case of Elevator load and results in steady state offset of Fineness. By applying the proposed method in which half the sample delay is included with the existing delay in the model, a significant performance improvement is obtained in terms of stability and settling time of the process(a settling time of around 1 hour for Elevator load and almost zero standard deviation for Fineness) with little tuning (the tuning is done only once in this case).

CHAPTER 8

Summary and Conclusions

This work focusses on design of Finite Impulse Response based MPC for uncertain systems. Further, a simple moving horizon estimator is designed to improve the performance of the controller. Finally a soft constraint based MPC is developed and experimentally implemented on a cement mill circuit. Also an improved method of MPC design for large sample delay systems is proposed and implemented in a simulator. The results obtained in each section are summarized as below

8.1 An evaluation of existing MPC tools

Based on finite impulse response predictions, a regularized l_2 predictive controller with input and input-rate constraints is analyzed. An estimator based on a simple constant output disturbance filter is developed for the feed back. The FIR based regulator problem is solved by convex quadratic program (QP) by converting the objective function into a standard form. The QP is solved using an algorithm based on interior point method. Illustrative examples are given to understand the effect of uncertainties in the model based controllers. The results are verified by testing the controller with scalar SOPDT with a zero transfer function model.

A moving horizon estimator based on Finite impulse response models is developed. The estimator is used to estimate the unknown disturbance by solving the objective function. The estimated disturbance is then used as a feedback to the controller. The objective function solved for estimation is convex QP similar to the regulator problem. To improve the performance of the MPC, the estimator is then made closed loop with the regulator by replacing the simple integrator developed in the previous section. By using a SOPDT with a zero transfer function model the performance of the estimator with measurement noise is provided. Also the closed loop performance of a model predictive controller consisting of a predictive regulator and estimator with SISO system is investigated.

A ℓ_2 regularized predictive controller with soft output constraints along with hard input and input-rate constraints is developed and the efficient application of this controller to uncertain stochastic systems is demonstrated. The predictive controller is equipped with soft output constraints that are used in a novel way to have robustness against model plant mismatch. The QP is solved similar to the standard form. The estimator used here is the simple output disturbance filter. The control algorithm designed is then solved using interior point algorithm.

8.2 Comparison of soft MPC with conventional MPC

By simulation, first with SISO system and then with $2X2$ MIMO system, the performance of the new robust constrained predictive controller to a conventional predictive controller is investigated. Finite impulse response coefficients for the models are obtained with step tests conducted on a simulated cement mill circuit. Improved performances are obtained with the soft MPC developed than that of the conventional MPC when the controllers are taken in closed loop simulation.

8.3 Application of Soft MPC in cement mill circuit

The proposed model predictive controller which includes the soft output constraints is made online with the cement mill circuit using a simulator. The controller is mainly used to control elevator power and fineness of the cement mill as they contribute to production and quality. Step response tests are conducted in the cement mill circuit and a SOPDT with a zero transfer function model is developed. Grindability is taken as the disturbance parameter. Uncertainties in the simulator are generated by changing the grindability of the input feed material.

The controller performance is compared in the simulator by running the controllers online for 16 hours each. The conditions for the simulations are kept similar both in conventional MPC and soft MPC to have a perfect comparison. This is done by creating simulation tutorials where the disturbances are given exactly at the same time for both the controllers after taken online. The performance of the soft MPC is better than the conventional MPC with uncertainties present in the system.

Real time implementation

The proposed controller is applied to a real time cement plant. The controller is compared with Fuzzy logic controller available in plant. The controller is interfaced with the real time ECS/SCADA so that the set points generated by the controller are directly entered into the field PLC. The controllers are designed for handling different recipes where the operating parameters will be changed depending on the type of recipe. The main control loops considered are Feed and Fineness control. The simple loops in the circuit are controlled with the PIDs available in the PLC. This is to improve the stability of the circuit and have same conditions for comparing both the controllers. Quantitative performance improvements of soft MPC over the fuzzy logic controller are obtained.

8.4 Implementation of Soft MPC to a Large Sample Delay System

A method to design the MPC more robust in handling the large sample delay cement mill systems is provided. By simulation for MIMO systems the performance of soft MPC and conventional MPC with models obtained from continuous sampling of data are compared. The controllers are tested in cement mill simulator circuit as in the previous section. Elevator power and Fineness are the controlled variables. Feed rate and separator speed are the manipulated variables. The predictive controller designed with the model from fast sample data (1 min sample) when applied to control with hourly sampled measurements, the controller is to be re-tuned. The parameters needed to be re-tuned are weight for penalties on measurement error (Q_z), weights of penalties on manipulated variables (S) and rate of change of manipulated variables (R). Also the hard constraints on input- rate movements are to be re-tuned to reduce the variations in the controller. In this work, the controller model is added with half the sample delay of measurement to improve the performance of the controller with one time tuning.

8.5 Conclusions

The following conclusions are drawn based on the above investigations:

1. It is verified by simulations that the closed-loop MPC performance degradation due to plant-model mismatch is tightly related to the uncertainty of impulse response coefficients. The closed loop performance of the controller is good when the uncertainty in each of the parameters (separately in gain, zero and time delay) are smaller(upto 40 % incase of time delay, 50 % in gain and twice incase zero) with that of the nominal model. The controller performance reaches sustained oscillations when the uncertainty in each of these parameters exceeds the above these values. Also presence of noise in the system further degrades the closed loop performance of the controller. The simulations in the present work provide as the potential as well as expected limits on the performance improvement that can be achieved by robust MPC, i.e. an upper limit on the potential performance is the performance of the nominal model.

2. The performance of soft MPC is compared with conventional MPC through simulations. From the simulations it is observed that with the uncertainties of the model parameters, the conventional MPC provides sustained oscillations when the model mismatch becomes twice(in-terms of gain, time delay and time constants at each instant) that of the nominal model. In case of soft MPC the oscillations are suppressed completely and the system settles before 75% of the control horizon. Thus the soft MPC provides much better performance improvement in the face of plant-model mismatch than the conventional MPC.

3. The performance of the proposed controllers are evaluated by simulation using

the transfer functions on MIMO models. It is observed that the soft MPC varies the actuator very little when the controlled variables are within the soft limits. In case of conventional MPC the controller makes aggressive moves on the actuator for settling the output variables which is an undesirable action especially in real time plants. The variations in the output variables are similar in both the controllers, but soft MPC achieves the variation with smaller actuator moves. The standard deviation of feed rate in conventional MPC is 4.4258 and separator speed is 2.4149.The standard deviation of feed rate in soft MPC is 3.269 and separator speed is 0.0878. Also the initial variation in feed and separator speed are high in case of conventional MPC when compared with soft MPC. Thus soft MPC provides lesser actuator variations which is mostly preferred in plant when compared with the conventional MPC.

4. The proposed controller with soft output constraints for handling the uncertainties of the cement mill circuit is implemented in a real plant. The predictive controller is compared with the high level Fuzzy Logic controller existing in the plant. It is observed that when both the controllers are made online with similar process conditions and in the same recipe, the standard deviation in the quality parameter (i.e Fineness of the product) is reduced by around 23% with soft MPC when compared with that of FLC in the plant. Also the stability of the system is improved with much simpler and smaller actuator variations in soft MPC when compared with the FLC.

5. The proposed controller is tested with cement mill simulator with large sample delay. It is observed that incase of plant-model mismatch, the soft MPC suppresses the sustained oscillations by smaller actuator moves when compared to the conventional MPC. The same MPC when applied for cement mill control with large sample delay measurements the regularization weights (R and S)need to be re-tuned a number of times (at least 3 to 4 times in this case) to improve the performance of the controller. In this case the regularization weights are increased for making the controller actions sluggish. This results in longer settling time (approximately 3.5 hours in case of Elevator load and steady state offset of Fineness) of the controlled variables. By simulations, it is demonstrated that the proposed

method gives a significant improvement in the stability of the process, (a settling time of around 1 hour for Elevator load and almost zero standard deviation for Fineness).

Future Scope

The estimation in the MPC proposed can be further improved by using standard time series models like ARMA, ARMAX etc., which can be obtained using standard identification techniques and the controller performance can be investigated with the inclusion of such estimators.

In real time comparison of the soft MPC with other controllers, the performance of both the controllers can be done by keeping certain operating conditions like comparison with respect to same settling time of output etc., This will provide a platform in which the best tuned controllers are compared for a given condition.

The research can be further extended in developing a robust controller based on Second Order Cone Programming (SOCP) technique which can inherently handle uncertainties and disturbances in the system.

APPENDIX A

Quadratic Program Formulation

A.1 Quadratic program for FIR based MPC

Define the vectors Z, R, and U as

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \quad R = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \quad U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \tag{A.1}$$

Then the predictions by the impulse response model in Equation 3.6b may be expressed as

$$Z = c + \Gamma U \tag{A.2}$$

For the case $N = 6$ and $n = 3$, Γ is assembled as

$$\Gamma = \begin{bmatrix} H_1 & 0 & 0 & 0 & 0 & 0 \\ H_2 & H_1 & 0 & 0 & 0 & 0 \\ H_3 & H_2 & H_1 & 0 & 0 & 0 \\ 0 & H_3 & H_2 & H_1 & 0 & 0 \\ 0 & 0 & H_3 & H_2 & H_1 & 0 \\ 0 & 0 & 0 & H_3 & H_2 & H_1 \end{bmatrix} \tag{A.3}$$

and c is

$$c = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} b_1 + (H_2 u_{-1} + H_3 u_{-2}) \\ b_2 + (H_3 u_{-1}) \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} \tag{A.4}$$

Similarly, for the case $N = 6$, define the matrices Λ and I_0 by

$$\Lambda = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ -I & I & 0 & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 & 0 \\ 0 & 0 & -I & I & 0 & 0 \\ 0 & 0 & 0 & -I & I & 0 \\ 0 & 0 & 0 & 0 & -I & I \end{bmatrix} \quad I_0 = \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{A.5}$$

and

$$\mathcal{Q}_z = \begin{bmatrix} Q_z & & & \\ & Q_z & & \\ & & \ddots & \\ & & & Q_z \end{bmatrix} \quad \mathcal{S} = \begin{bmatrix} S & & & \\ & S & & \\ & & \ddots & \\ & & & S \end{bmatrix} \tag{A.6}$$

The objective function in Equation 3.6a may be expressed as

$$\begin{aligned} \phi &= \frac{1}{2} \sum_{k=0}^{N-1} \|z_{k+1} - r_{k+1}\|_{Q_z}^2 + \|\Delta u_k\|_S^2 \\ &= \frac{1}{2} \|Z - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2} \|\Lambda U - I_0 u_{-1}\|_{\mathcal{S}}^2 \\ &= \frac{1}{2} \|c + \Gamma U - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2} \|\Lambda U - I_0 u_{-1}\|_{\mathcal{S}}^2 \\ &= \frac{1}{2} U' \left(\Gamma' \mathcal{Q}_z \Gamma + \Lambda' \mathcal{S} \Lambda\right) U \\ &\qquad + \left(\Gamma' \mathcal{Q}_z (c - R) - \Lambda' \mathcal{S} I_0 u_{-1}\right)' U \\ &\qquad + \left(\frac{1}{2} \|c - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2} \|I_0 u_{-1}\|_{\mathcal{S}}^2\right) \\ &= \frac{1}{2} U' H U + g' U + \rho \end{aligned} \tag{A.7}$$

in which

$$H = \Gamma' \mathcal{Q}_z \Gamma + \Lambda' \mathcal{S} \Lambda \tag{A.8a}$$

$$g = \Gamma' \mathcal{Q}_z (c - R) - \Lambda' \mathcal{S} I_0 u_{-1} \tag{A.8b}$$

$$\rho = \frac{1}{2} \|c - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2} \|u_{-1}\|_S^2 \tag{A.8c}$$

Consequently, the FIR based MPC regulator problem in Equation 3.6 can be solved by finding the solution of the following convex quadratic program

$$\min_{U} \quad \psi = \frac{1}{2}U'HU + g'U \tag{A.9a}$$

$$s.t. \quad U_{\min} \leq U \leq U_{\max} \tag{A.9b}$$

$$b_l \leq \Lambda U \leq b_u \tag{A.9c}$$

in which

$$U_{\min} = \begin{bmatrix} u_{\min} \\ u_{\min} \\ \vdots \\ u_{\min} \end{bmatrix} \qquad U_{\max} = \begin{bmatrix} u_{\max} \\ u_{\max} \\ \vdots \\ u_{\max} \end{bmatrix} \tag{A.10}$$

and

$$b_l = \begin{bmatrix} \Delta u_{\min} + u_{-1} \\ \Delta u_{\min} \\ \vdots \\ \Delta u_{\min} \end{bmatrix} \qquad b_u = \begin{bmatrix} \Delta u_{\max} + u_{-1} \\ \Delta u_{\max} \\ \vdots \\ \Delta u_{\max} \end{bmatrix} \tag{A.11}$$

In a model predictive controller only the first vector, u_0^*, of $U^* = \begin{bmatrix} (u_0^*)' & (u_1^*)' & \ldots & (u_{N-1}^*)' \end{bmatrix}'$, is implemented on the process. At the next sample time the open-loop optimization is repeated with new information due to a new measurement.

A.2 Quadratic Program formulation for Estimator

Estimation problem can be formulated by considering a finite impulse response (FIR) model

$$z_k = b_k + \sum_{i=0}^{n} H_i u_{k-i} + \sum_{i=1}^{n} H_{id} d_{k-i} \tag{A.12}$$

in which $\{H_i\}_{i=0}^{n}$ are the impulse response coefficients. Here the moving horizon estimator are based on the FIR model.

Using the transfer function model, the estimation problem can be formulated as

$$\min_{\{z,d\}} \quad \phi = \frac{1}{2}\sum_{k=0}^{N_e} \|z_k - y_k\|_{Q_z}^2 + \|\Delta d_k\|_S^2 + \|d_k\|_R^2 \tag{A.13a}$$

$$s.t. \qquad z_k = \bar{b_k} + \sum_{i=0}^{n} H_{d,i} d_{k-i} \tag{A.13b}$$

$$d_{\min} \leq d_k \leq d_{\max} \qquad k = 0, \ldots, N_e \tag{A.13c}$$

$$\Delta d_{\min} \leq \Delta d_k \leq \Delta d_{\max} \qquad k = 0, \ldots, N_e \tag{A.13d}$$

$$where \qquad \bar{b_k} = b_k + \sum_{i=1}^{n} H_{u,i} u_{k-i} \qquad k = 1, \ldots, N_e \tag{A.13e}$$

It is assumed that $\{d_i\}_{i=-1}^{-n}$ are disturbance values fixed at their previous estimates such that the real decision variables in Equation A.13e are $\{d_k\}_{k=0}^{N_e}$. The decision variables are normally split into a slow varying component (level), d_k and a rapid varying component (process noise), w_k, the constraints will have the following interpretation,

$$d_k = \begin{bmatrix} \bar{d}_k \\ w_k \end{bmatrix}, \; d_{min} = \begin{bmatrix} \bar{d}_{min} \\ w_{min} \end{bmatrix}, \; d_{max} = \begin{bmatrix} \bar{d}_{max} \\ w_{max} \end{bmatrix}$$

$$\Delta d_{min} = \begin{bmatrix} \Delta \bar{d}_{min} \\ \Delta w_{min} \end{bmatrix} \text{ and } \Delta d_{max} = \begin{bmatrix} \Delta \bar{d}_{max} \\ \Delta w_{max} \end{bmatrix}$$

Typically $\Delta w_{min} = -\infty$, $\Delta w_{max} = \infty$ and Δw_k is not penalized in the objective function in order not to restrict the movements of rapid varying component of the unknown disturbance. However the size of w_k may be constrained and penalized in the objective function. Oppositely only the rate of movement of the slowly varying component is penalized. Consequently the weight matrices S and R have the structure

$$S = \begin{bmatrix} S_d & 0 \\ 0 & 0 \end{bmatrix}, R = \begin{bmatrix} 0 & 0 \\ 0 & R_w \end{bmatrix} \tag{A.14}$$

The typical case with $H_{d,i} = \begin{bmatrix} H_{\bar{d},i} & H_{w,i} \end{bmatrix}$ and $H_{\bar{d},i} = H_{w,i}$ corresponds to an unknown disturbance of the form $d_k = \bar{d}_k + w_k$.

Here an approach to formulate a dense quadratic program in standard form equivalent with Equation (A.13a) is provided. This is done by elimination of the outputs, $\{z_k\}_{k=0}^{N_e}$ from the optimization problem (Equation A.13a).

Define the vectors Z, Y, and D as

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \quad D = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{N-1} \end{bmatrix} \tag{A.15}$$

Then the predictions by the impulse response model (Equation A.13b) may be expressed as

$$Z = c + \Gamma D \tag{A.16}$$

For the case $N = 6$ and $n = 3$, Γ is assembled as

$$\Gamma = \begin{bmatrix} H_{d,0} & 0 & 0 & 0 & 0 & 0 & 0 \\ H_{d,1} & H_{d,0} & 0 & 0 & 0 & 0 & 0 \\ H_{d,2} & H_{d,1} & H_{d,0} & 0 & 0 & 0 & 0 \\ H_{d,3} & H_{d,2} & H_{d,1} & H_{d,0} & 0 & 0 & 0 \\ 0 & H_{d,3} & H_{d,2} & H_{d,1} & H_{d,0} & 0 & 0 \\ 0 & 0 & H_{d,3} & H_{d,2} & H_{d,1} & H_{d,0} & 0 \\ 0 & 0 & 0 & H_{d,3} & H_{d,2} & H_{d,1} & H_{d,0} \end{bmatrix} \tag{A.17}$$

and c is

$$c = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} \bar{b}_0 + \sum_{i=1}^{3} H_{u,i} u_{0-i} + \sum_{i=1}^{3} H_{d,i} d_{0-i} \\ \bar{b}_1 + \sum_{i=1}^{3} H_{u,i} u_{1-i} + \sum_{i=2}^{3} H_{d,i} d_{1-i} \\ \bar{b}_2 + \sum_{i=1}^{3} H_{u,i} u_{2-i} + \sum_{i=3}^{3} H_{d,i} d_{2-i} \\ \bar{b}_3 + \sum_{i=1}^{3} H_{u,i} u_{3-i} \\ \bar{b}_4 + \sum_{i=1}^{3} H_{u,i} u_{4-i} \\ \bar{b}_5 + \sum_{i=1}^{3} H_{u,i} u_{5-i} \\ \bar{b}_5 + \sum_{i=1}^{3} H_{u,i} u_{6-i} \end{bmatrix} \tag{A.18}$$

Similarly, for the case $N = 6$, define the matrices Λ and I_0 by

$$\Lambda = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ -I & I & 0 & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 & 0 \\ 0 & 0 & -I & I & 0 & 0 \\ 0 & 0 & 0 & -I & I & 0 \\ 0 & 0 & 0 & 0 & -I & I \end{bmatrix} \quad I_0 = \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{A.19}$$

and

$$\mathcal{Q}_z = \begin{bmatrix} Q_z & & & \\ & Q_z & & \\ & & \ddots & \\ & & & Q_z \end{bmatrix} \quad \mathcal{S} = \begin{bmatrix} S & & & \\ & S & & \\ & & \ddots & \\ & & & S \end{bmatrix} \tag{A.20}$$

Then the objective function in Equation (A.13a) may be expressed as

$$\begin{aligned} \phi &= \frac{1}{2}\sum_{k=0}^{N-1} \|z_{k+1} - y_{k+1}\|_{Q_z}^2 + \|\Delta d_k\|_S^2 + \|d_k\|_R^2 \\ &= \frac{1}{2}\|Z - Y\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|\Lambda d - I_0 d_{-1}\|_{\mathcal{S}}^2 + \frac{1}{2}\|D\|_{\mathcal{R}}^2 \\ &= \frac{1}{2}\|c + \Gamma D - Y\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|\Lambda D - I_0 d_{-1}\|_{\mathcal{S}}^2 + \frac{1}{2}\|D\|_{\mathcal{R}}^2 \\ &= \frac{1}{2} D' \left(\Gamma' \mathcal{Q}_z \Gamma + \Lambda' \mathcal{S} \Lambda + R\right) D \\ &\qquad + \left(\Gamma' \mathcal{Q}_z (c - Y) - \Lambda' \mathcal{S} I_0 d_{-1}\right)' D \\ &\qquad + \left(\frac{1}{2}\|c - Y\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|I_0 d_{-1}\|_{\mathcal{S}}^2\right) \\ &= \frac{1}{2} D' H D + g' D + \rho \end{aligned} \tag{A.21}$$

in which

$$H = \Gamma' \mathcal{Q}_z \Gamma + \Lambda' \mathcal{S} \Lambda + R \tag{A.22a}$$

$$g = \Gamma' \mathcal{Q}_z (c - Y) - \Lambda' \mathcal{S} I_0 d_{-1} \tag{A.22b}$$

$$\rho = \frac{1}{2}\|c - Y\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|d_{-1}\|_S^2 \tag{A.22c}$$

Consequently, the FIR based Estimator problem can be solved by the solution of the following convex quadratic program

$$\min_{D} \quad \psi = \frac{1}{2} D' H D + g' D \tag{A.23a}$$

$$s.t. \quad D_{\min} \leq D \leq D_{\max} \tag{A.23b}$$

$$b_l \leq \Lambda D \leq b_u \tag{A.23c}$$

in which

$$D_{\min} = \begin{bmatrix} d_{\min} \\ d_{\min} \\ \vdots \\ d_{\min} \end{bmatrix} \quad D_{\max} = \begin{bmatrix} d_{\max} \\ d_{\max} \\ \vdots \\ d_{\max} \end{bmatrix} \tag{A.24}$$

and

$$b_l = \begin{bmatrix} \Delta d_{\min} + d_{-1} \\ \Delta d_{\min} \\ \vdots \\ \Delta d_{\min} \end{bmatrix} \quad b_u = \begin{bmatrix} \Delta d_{\max} + d_{-1} \\ \Delta d_{\max} \\ \vdots \\ \Delta d_{\max} \end{bmatrix} \tag{A.25}$$

A.3 Quadratic Program formulation for Soft MPC

Define the vectors Z, R, and U as

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \quad R = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \quad U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \tag{A.26}$$

Then the predictions by the impulse response model in Equation (3.22) may be expressed as

$$Z = c + \Gamma U \tag{A.27}$$

For the case $N = 6$ and $n = 3$, Γ is assembled as

$$\Gamma = \begin{bmatrix} H_1 & 0 & 0 & 0 & 0 & 0 \\ H_2 & H_1 & 0 & 0 & 0 & 0 \\ H_3 & H_2 & H_1 & 0 & 0 & 0 \\ 0 & H_3 & H_2 & H_1 & 0 & 0 \\ 0 & 0 & H_3 & H_2 & H_1 & 0 \\ 0 & 0 & 0 & H_3 & H_2 & H_1 \end{bmatrix} \tag{A.28}$$

and c is

$$c = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} b_1 + (H_2 u_{-1} + H_3 u_{-2}) \\ b_2 + (H_3 u_{-1}) \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} \tag{A.29}$$

Similarly, for the case $N = 6$, define the matrices Λ and I_0 by

$$\Lambda = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ -I & I & 0 & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 & 0 \\ 0 & 0 & -I & I & 0 & 0 \\ 0 & 0 & 0 & -I & I & 0 \\ 0 & 0 & 0 & 0 & -I & I \end{bmatrix} \quad I_0 = \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{A.30}$$

Define $s_\eta = \begin{bmatrix} {s_{\eta 1}}' & {s_{\eta 1}}' & \dots & {s_{\eta N}}' \end{bmatrix}'$ and

$$\mathcal{Q}_z = \begin{bmatrix} Q_z & & & \\ & Q_z & & \\ & & \ddots & \\ & & & Q_z \end{bmatrix} \mathcal{S}_\rangle = \begin{bmatrix} S_i & & & \\ & S_i & & \\ & & \ddots & \\ & & & S_i \end{bmatrix} \tag{A.31}$$

with $i = \{u, \eta\}$. Then the objective function in Equation (3.22) may be expressed as

$$
\begin{aligned}
\phi &= \frac{1}{2}\sum_{k=0}^{N-1} \|z_{k+1} - r_{k+1}\|_{Q_z}^2 + \|\Delta u_k\|_S^2 + \frac{1}{2}\|\eta_{k+1}\|_{S_\eta}^2 \\
&\quad + s'_{\eta_{k+1}}\eta_{k+1} \\
&= \frac{1}{2}\|Z - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|\Lambda U - I_0 u_{-1}\|_{\mathcal{S}}^2 + \frac{1}{2}\|\eta\|_{S_\eta}^2 \\
&\quad + s'_\eta \eta \\
&= \frac{1}{2}\|c + \Gamma U - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|\Lambda U - I_0 u_{-1}\|_{\mathcal{S}}^2 \\
&\quad + \frac{1}{2}\|\eta\|_{S_\eta}^2 + s'_\eta \eta \\
&= \frac{1}{2}U'\left(\Gamma'\mathcal{Q}_z\Gamma + \Lambda'\mathcal{S}\Lambda\right)U \\
&\quad + \left(\Gamma'\mathcal{Q}_z(c - R) - \Lambda'\mathcal{S}I_0 u_{-1}\right)' U \\
&\quad + \left(\frac{1}{2}\|c - R\|_{\mathcal{Q}_z}^2 + \frac{1}{2}\|I_0 u_{-1}\|_{\mathcal{S}}^2\right) \\
&\quad + \frac{1}{2}\eta' S_\eta \eta + s'_\eta \eta \\
&= \frac{1}{2}U'HU + g'U + \rho + \frac{1}{2}\eta' S_\eta \eta + s'_\eta \eta \\
&= \frac{1}{2}x'\bar{H}x + \bar{g}'x + \rho
\end{aligned}
\tag{A.32}
$$

in which

$$H = \Gamma' \mathcal{Q}_z \Gamma + \Lambda' \mathcal{S} \Lambda \tag{A.33a}$$

$$g = \Gamma' \mathcal{Q}_z (c - R) - \Lambda' \mathcal{S} I_0 u_{-1} \tag{A.33b}$$

$$\rho = \frac{1}{2} \|c - R\|^2_{\mathcal{Q}_z} + \frac{1}{2} \|u_{-1}\|^2_{S} \tag{A.33c}$$

$$x = \begin{bmatrix} U \\ \eta \end{bmatrix} \tag{A.33d}$$

$$\bar{H} = \begin{bmatrix} H & 0 \\ 0 & S_\eta \end{bmatrix} \tag{A.33e}$$

$$\bar{g} = \begin{bmatrix} g \\ s_\eta \end{bmatrix} \tag{A.33f}$$

Consequently, we may solve the FIR based MPC regulator problem in Equation (3.22) by solution of the following convex quadratic program

$$\min_x \quad \psi = \frac{1}{2} x' \bar{H} x + \bar{g}' x \tag{A.34a}$$

$$s.t. \quad x_{\min} \leq x \leq x_{\max} \tag{A.34b}$$

$$b_l \leq \bar{A} x \leq b_u \tag{A.34c}$$

in which

$$A = \begin{bmatrix} \Lambda & 0 \\ \Gamma & -I \\ \Gamma & I \end{bmatrix} \tag{A.35a}$$

$$x_{\min} = \begin{bmatrix} U_{\min} \\ 0 \end{bmatrix} \quad x_{\max} = \begin{bmatrix} U_{\max} \\ \infty \end{bmatrix} \tag{A.35b}$$

where

$$U_{\min} = \begin{bmatrix} u_{\min} \\ u_{\min} \\ \vdots \\ u_{\min} \end{bmatrix} \quad U_{\max} = \begin{bmatrix} u_{\max} \\ u_{\max} \\ \vdots \\ u_{\max} \end{bmatrix} \tag{A.36}$$

and

$$b_l = \begin{bmatrix} \Delta U_{\min} \\ -\infty \\ Z_{\min} - c \end{bmatrix} \quad b_u = \begin{bmatrix} \Delta U_{\max} \\ Z_{\max} - c \\ \infty \end{bmatrix} \tag{A.37}$$

In a model predictive controller only the first vector,

u_0^*, of $U^* = \begin{bmatrix} (u_0^*)' & (u_1^*)' & \ldots & (u_{N-1}^*)' \end{bmatrix}'$, is implemented on the process. At the next sample time the open-loop optimization is repeated with new information due to a new measurement.

APPENDIX B

Flow Diagram for MPC Design

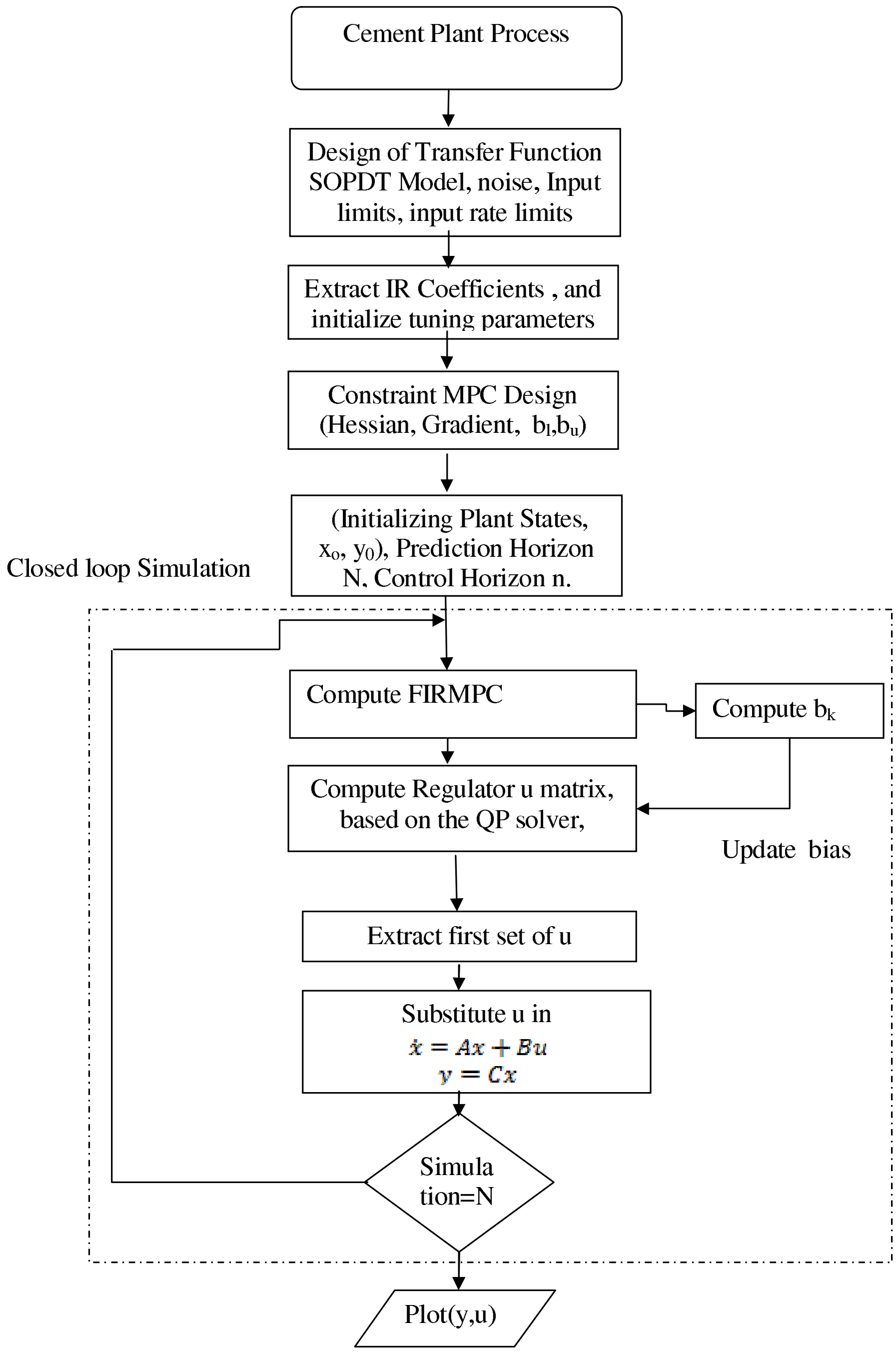


Figure B.1: Program Flow for MPC Design

APPENDIX C

General Form of Quadratic Program

C.1 Quadratic Program Formulation

Define the vectors Z, R, and U as

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \quad R = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \quad U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \tag{C.1}$$

Then the predictions by the impulse response model in Equation (3.22) may be expressed as

$$Z = c + \Gamma U \tag{C.2}$$

In the general form Γ is assembled as

$$\Gamma = \begin{bmatrix} H_1 & 0 & 0 & 0 & 0 & 0 \\ H_2 & H_1 & 0 & 0 & 0 & 0 \\ \vdots & H_2 & H_1 & 0 & 0 & 0 \\ H_(N-n) & \dots & H_2 & H_1 & 0 & 0 \\ 0 & H_(N-n) & \dots & H_2 & H_1 & 0 \\ 0 & 0 & H_(N-n) & \dots & H_2 & H_1 \end{bmatrix} \tag{C.3}$$

and c is

$$c = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} b_1 + (H_2 u_{-1} + H_3 u_{-2}) \\ b_2 + (H_3 u_{-1}) \\ b_3 \\ b_4 \\ \vdots \\ b_5 \\ b_N \end{bmatrix} \tag{C.4}$$

Similarly, define the matrices Λ and I_0 by

$$\Lambda = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ -I & I & 0 & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 & 0 \\ 0 & 0 & -I & I & 0 & 0 \\ 0 & 0 & 0 & -I & I & 0 \\ 0 & 0 & 0 & 0 & -I & I \end{bmatrix}_{N \times N} \qquad I_0 = \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{N \times 1} \tag{C.5}$$

Define $s_\eta = \begin{bmatrix} s_{\eta 1}' & s_{\eta 1}' & \dots & s_{\eta N}' \end{bmatrix}'$ and

$$\mathcal{Q}_z = \begin{bmatrix} Q_z & & & \\ & Q_z & & \\ & & \ddots & \\ & & & Q_z \end{bmatrix} \mathcal{S}_\rangle = \begin{bmatrix} S_i & & & \\ & S_i & & \\ & & \ddots & \\ & & & S_i \end{bmatrix} \tag{C.6}$$

with $i = \{u, \eta\}$. Then the objective function in Equation (3.22) may be derived as in Appendix A.3, then converted to the general convex quadratic program

$$\min_x \quad \psi = \frac{1}{2} x' \bar{H} x + \bar{g}' x \tag{C.7a}$$

$$s.t. \quad x_{\min} \leq x \leq x_{\max} \tag{C.7b}$$

$$b_l \leq \bar{A} x \leq b_u \tag{C.7c}$$

in which

$$A = \begin{bmatrix} \Lambda & 0 \\ \Gamma & -I \\ \Gamma & I \end{bmatrix} \tag{C.8a}$$

$$x_{\min} = \begin{bmatrix} U_{\min} \\ 0 \end{bmatrix} \quad x_{\max} = \begin{bmatrix} U_{\max} \\ \infty \end{bmatrix} \tag{C.8b}$$

where

$$U_{\min} = \begin{bmatrix} u_{\min} \\ u_{\min} \\ \vdots \\ u_{\min} \end{bmatrix} \quad U_{\max} = \begin{bmatrix} u_{\max} \\ u_{\max} \\ \vdots \\ u_{\max} \end{bmatrix} \tag{C.9}$$

and

$$b_l = \begin{bmatrix} \Delta U_{\min} \\ -\infty \\ Z_{\min} - c \end{bmatrix} \quad b_u = \begin{bmatrix} \Delta U_{\max} \\ Z_{\max} - c \\ \infty \end{bmatrix} \tag{C.10}$$

In a model predictive controller only the first vector,

u_0^*, of $U^* = \begin{bmatrix} (u_0^*)' & (u_1^*)' & \dots & (u_{N-1}^*)' \end{bmatrix}'$, is implemented on the process. At the next sample time the open-loop optimization is repeated with new information due to a new measurement.

APPENDIX D

Interior Point Method Algorithm

D.1 Interior Point Method Algorithm

Interior point methods are used to solve linear and nonlinear convex optimization problem. These algorithms have been inspired by Karmarkar's algorithm, developed by Karmarkar (1984) for linear programming. The basic elements of the method consists of a self-concordant barrier function used to encode the convex set. Contrary to the simplex method, it reaches an optimal solution by traversing the interior of the feasible region. Mehrotra (1992) has provided a primal- dual path- following technique which is the basis for most of the implementations. The main advantage of interior point methods is that their computational complexity (number of iterations, time to complete,...) is no worse than some polynomial function of parameters such as the number of constraints or the number of variables, where as complexity of other well known approaches, including Active Set methods, can be exponential in these parameters in the worst case.

The algorithm for interior point method for solving linear quadratic program is given below

The linear program can be formulated as

$$\min_{\boldsymbol{x}\in R^n} \quad \boldsymbol{g}'\boldsymbol{x} \tag{D.1}$$

$$\text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{D.2}$$

$$\boldsymbol{x} \geq \boldsymbol{0} \tag{D.3}$$

The Lagrangian function can be given as

$$L(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \boldsymbol{g}'\boldsymbol{x} - \boldsymbol{\mu}'(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) - \boldsymbol{\lambda}'\boldsymbol{x} \tag{D.4}$$

The optimality conditions are

$$\nabla_x L(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \boldsymbol{g} - \boldsymbol{A}'\boldsymbol{\mu} - \boldsymbol{\lambda} = 0 \tag{D.5}$$

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{D.6}$$

$$\boldsymbol{x} \geq 0 \tag{D.7}$$

$$\boldsymbol{\lambda} \geq 0 \tag{D.8}$$

$$\boldsymbol{x}_i \boldsymbol{\lambda}_i = 0, \qquad i = 1, 2, \ldots, m_C \tag{D.9}$$

These can be expressed as

$$\boldsymbol{r}_L = \nabla_x L = \boldsymbol{g} - \boldsymbol{A}'\boldsymbol{\mu} - \boldsymbol{\lambda} = 0 \tag{D.10a}$$

$$\boldsymbol{r}_A = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} = 0 \tag{D.10b}$$

$$\boldsymbol{X}\boldsymbol{\Lambda}\boldsymbol{e} = 0 \tag{D.10c}$$

$$\boldsymbol{x} \geq 0 \tag{D.10d}$$

$$\boldsymbol{\lambda} \geq 0 \tag{D.10e}$$

Thus the optimality conditions in matrix form is

$$F(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \begin{bmatrix} \boldsymbol{g} - \boldsymbol{A}'\boldsymbol{\mu} - \boldsymbol{\lambda} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{X}\boldsymbol{\Lambda}\boldsymbol{e} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0 \tag{D.11a}$$

These conditions can thus be solved using Newton like method after forming the matrix as shown below

$$\mathbf{F}(x,\mu,\lambda) = \begin{bmatrix} 0 & -A' & -I \\ A & 0 & 0 \\ \Lambda & 0 & X \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mu \\ \Delta\lambda \end{bmatrix} = - \begin{bmatrix} r_L \\ r_A \\ r_C \end{bmatrix},$$
$$\mathbf{x} \geq 0, \qquad \lambda \geq 0.$$

where:

$$\begin{bmatrix} \mathbf{x} \\ \mu \\ \lambda \end{bmatrix} \leftarrow \begin{bmatrix} \mathbf{x} \\ \mu \\ \lambda \end{bmatrix} + \alpha \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mu \\ \Delta\lambda \end{bmatrix}$$

For solving the above matrix first we have to define the affine direction $(\Delta\boldsymbol{x}^{\text{aff}}, \Delta\boldsymbol{\mu}^{\text{aff}}, \Delta\boldsymbol{\lambda}^{\text{aff}})^T$ by solving

$$\begin{bmatrix} \mathbf{0} & -\boldsymbol{A}' & -\boldsymbol{I} \\ \boldsymbol{A} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\Lambda} & \mathbf{0} & \boldsymbol{X} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{x}^{\text{aff}} \\ \Delta\boldsymbol{\mu}^{\text{aff}} \\ \Delta\boldsymbol{\lambda}^{\text{aff}} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{g} - \boldsymbol{A}'\boldsymbol{\mu} - \boldsymbol{\lambda} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{X}\boldsymbol{\Lambda}\boldsymbol{e} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{r}_L \\ \boldsymbol{r}_A \\ \boldsymbol{r}_C \end{bmatrix} \tag{D.12}$$

Then we have to find the largest affine step

α^{aff} and β^{aff} such that

$$\boldsymbol{x} + \alpha^{\text{aff}}\Delta\boldsymbol{x}^{\text{aff}} \geq \mathbf{0}, \;\; \boldsymbol{\lambda} + \beta^{\text{aff}}\Delta\boldsymbol{\lambda}^{\text{aff}} \geq \mathbf{0} \tag{D.13}$$

Compute the affined duality gap

$$s^{\text{aff}} = (\boldsymbol{x} + \alpha^{\text{aff}}\Delta\boldsymbol{x}^{\text{aff}})'(\boldsymbol{\lambda} + \beta^{\text{aff}}\Delta\boldsymbol{\lambda}^{\text{aff}})/n \tag{D.14}$$

where n is the dimension of $\boldsymbol{x}$.

Thus for center step we have to introduce a duality measure :

$$s = \frac{\mathbf{x}'\lambda}{n},$$

and the centering parameter

$$\sigma = (s^{\text{aff}}/s)^3,$$

We solve the final equation after improving it by a corrector step and computing the centering parameter as

$$\begin{bmatrix} 0 & -A' & -I \\ A & 0 & 0 \\ \Lambda & 0 & X \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \mu \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} r_L \\ r_A \\ r_C + \Delta X^{\text{aff}} \Delta \Lambda^{\text{aff}} e - \sigma s e \end{bmatrix},$$

Further we compute the largest step length α and β such that

$$\boldsymbol{x} + \alpha \Delta \boldsymbol{x} \geq \mathbf{0}, \;\; \boldsymbol{\lambda} + \beta \Delta \boldsymbol{\lambda} \geq \mathbf{0} \tag{D.15}$$

and update $\boldsymbol{x}$, $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ by

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\mu} \\ \boldsymbol{\lambda} \end{bmatrix} \leftarrow \begin{bmatrix} \boldsymbol{x} + \eta \alpha \Delta \boldsymbol{x} \\ \boldsymbol{\mu} + \eta \beta \Delta \boldsymbol{\mu} \\ \boldsymbol{\lambda} + \eta \beta \Delta \boldsymbol{\lambda} \end{bmatrix} \tag{D.16}$$

Thus by improving corrector step and computing the centering parameter, the total step for solving next iteration is obtained.

The algorithm is specified here as below:

Predictor-Corrector Algorithm for QP

Compute $(\mathbf{x}_0, \mu_0, \lambda_0)$

for $k = 0, 1, 2, \ldots$

Set $(\mathbf{x}, \mu, \lambda) = (\mathbf{x}_k, \mu_k, \lambda_k)$ and solve (D.12)

for $(\Delta \mathbf{x}^{\text{aff}}, \Delta \mu^{\text{aff}}, \Delta \lambda^{\text{aff}})$;

Calculate $\hat{\alpha}_{\text{aff}} and \hat{\beta}_{\text{aff}}$ as in $(D.13)$;

Calculate $s^{\text{aff}} = (\mathbf{x} + \hat{\alpha}^{\text{aff}} \Delta \mathbf{x}^{\text{aff}})^T (\lambda + \hat{\beta}^{\text{aff}} \Delta \lambda^{\text{aff}})/n$;

Set centering paramenter to $\sigma = (s_{\text{aff}}/s)^3$;

Solve (D.1) for $(\Delta \mathbf{x}, \Delta \mu, \Delta \lambda)$;

Calculate $\hat{\alpha}_{\text{aff}} and \hat{\beta}_{\text{aff}}$ as in $(D.15)$;

Set $(\mathbf{x}_{k+1}) = (\mathbf{x}_k) + \hat{\alpha}(\Delta \mathbf{x})$;

Set $(\mu_{k+1}, \lambda_{k+1}) = (\mu_k, \lambda_k) + \hat{\beta}(\Delta \mu, \Delta \lambda)$;

end

APPENDIX E

Matlab Program for Soft MPC design

E.1 Initialization of MPC

```
%echo on
% Program for generating the Simulation response for the
% Soft MPC with Finite Impulse Response Methods for
%second order Transfer Function with delay time
%
% System Definition :
%
%Transfer Function Parameters
%
% K- Gain of the system
% T1,T2- Time constants
% beta - zero of the system
% Td - Time delay of the system
%
% Ts- Sampling Time
% Nsim - Simulation horizon
%
% Normal_MPC - Function for computing the control sequence
% using MPC without including soft constraint limits
% Soft_MPC - Function for MPC with  soft Constraints
%
% U - control input to the system
% D - Disturbance in the system
% Y - Output of the system including stochastic noises
```

```
% v -  Measurement Noise
% w -  Process Noise

% System Definition
close all
clear all
clc

% Parameters
K = [0.62 0.29;-15 5]
T1 = [45 2;60 14]
T2 = [8 38;0 1]
Td = [5 31.5;5 30.1]
beta = [0 8;0 0]
tol=1e-12;
%%%%
% K = [1.0 0;0 1]
% T1 = [5 0;0 5]
% T2 = [5 0;0 5]
% Td = [5 0;0 5]
% beta = [2 0;0 2]
%%%%
[nz,nu]=size(K)
dTs = 1.0;
n = 50;
Fs=18;
%% Design and Simulation of MPC Parameters

Nsim = 500;
N = 2*n;

% Noise sequence
```

```
Qv=[0.00 0 ;0 0.00];
Qw=[0.00 0;0 0.00];

 %% Input Limits

umin = [-15;-30];
umax = [15;30];
dumin = [-1.5;-0.3];
dumax = [1.5;0.3];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Normal MPC Parameters

Qz = [8 0;0 1.0]
S = [1.0e-8 0;0 1.0e-1]

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Soft MPC Parameters
Qzs = [1.8 0 ;0 0.0000000002] ;
Ss = [8000 0;0 40000];
Sn =[9000 0;0 250] ;    % Quadratic weight on Soft Limits
snlin=[0;0];     % Linear Weight on Soft Limits

%% System Model

[Uk,Sk,Vk]= svd(K)

Skmod= [1*Sk(1,:);1.0*Sk(2,:)]
Knew=Uk*Skmod*Vk'

[Ut,St,Vt]= svd(T1)
```

```
Stmod= [1.0*St(1,:);1.0*St(2,:)]
T1new=Ut*Stmod*Vt'

[Ud,Sd,Vd]= svd(Td)

Sdmod= [1*Sd(1,:);1.0*Sd(2,:)]
Tdnew=Ud*Sdmod*Vd'

%%
Knew=K

K1 = Knew%+[ 0.4  0.3 ;-8  4]
T11 = T1%+[ -15 -2 ;0  0]
T21 = T2
Td1 = Td +[0 0 ; 0 0]   %[ 5 -0.5 ;6 1]%Tdnew
beta1 = beta%+[-0 ;0  ]
%%
 [A,B,C,D,H1]=MIMOSSModel(nz,nu,N,Nsim,K1,T11,T21,beta1,
                                            Td1,dTs,tol);

Bd=B;
Nx = size(A,1);
Nu = size(B,2);
Ny = size(C,1);
Nd= size(Bd,2);

%% MPC definition
% Parameters
Khat =  K*1 ;
T1hat = T1*1;
T2hat = T2*1;
Tdhat = Td*1;
```

```
betahat = beta*1;

% Extract Impulse Response Coefficients (SISO) for system

[Ahat,Bhat,Chat,Dhat,Himp]=MIMOSSModel(nz,nu,N,Nsim,Khat,
                                T1hat,T2hat,betahat,Tdhat,dTs,tol);

%% Noise and disturbance Generation
R = ones(Ny,Nsim+1)*0;
[v,w] =MPC_noise(Qv,Qw,Nd,Ny,Nsim);

D=  0*ones(Nd,Nsim);
D(:,21:end) = 3*ones(2,480)*1;
Dw = D +w;
x0 = zeros(Nx,1) + 0*ones(Nx,1);
%%% Soft Limits

zmin = [0-0.8;0-50.0];
zmax = [0+0.8;0+50.0];

%% Soft MPC

[Y2,U2,R,FIRMPC2]=Soft_MPC_MIMO(Himp,A,B,Bd,C,Qzs,Ss,
                                Sn,snlin,R,Dw,v,x0,umin,umax,
                                dumin,dumax,zmin,zmax,N,Nx,Nu,Ny,Nsim);

%% Plot Results

  figure(1)
  subplot(211)
      set(gca,'FontName','Times','FontSize',Fs);
      plot(0:Nsim,Y2(1,:),'r','linewidth',2)
```

```
  hold on
  stairs(0:Nsim,R(1,:),'m--','linewidth',2)
  plot([0 Nsim],[zmax(1,:) zmax(1,:)],'m--','linewidth',1);
  plot([0 Nsim],[zmin(1,:) zmin(1,:)],'m--','linewidth',1);
  ylabel('Y1')
%  xlabel('time')
subplot(212)
  set(gca,'FontName','Times','FontSize',Fs);
  plot(0:Nsim,Y2(2,:),'r','linewidth',2)
 hold on
 stairs(0:Nsim,R(2,:),'m--','linewidth',2)
  plot([0 Nsim],[zmax(2,:) zmax(2,:)],'m--','linewidth',1);
  plot([0 Nsim],[zmin(2,:) zmin(2,:)],'m--','linewidth',1);
  ylabel('Y2')
 figure(2)

subplot(211)
  set(gca,'FontName','Times','FontSize',Fs);
  stairs(0:Nsim,[U2(1,:) U2(1,end)],'r','linewidth',2)
  hold on
  plot([0 Nsim],[umin(1,:) umin(1,:)],'m--','linewidth',2);
  plot([0 Nsim],[umax(1,:) umax(1,:)],'m--','linewidth',2);
  axis([0 Nsim 1.2*umin(1,:) 1.2*umax(1,:)])
  ylabel('U1')
   subplot(212)
  set(gca,'FontName','Times','FontSize',Fs);
  stairs(0:Nsim,[U2(2,:) U2(2,end)],'r','linewidth',2)
   hold on
  plot([0 Nsim],[umin(2,:) umin(2,:)],'m--','linewidth',2);
  plot([0 Nsim],[umax(2,:) umax(2,:)],'m--','linewidth',2);
  axis([0 Nsim 1.2*umin(2,:) 1.2*umax(2,:)])
  ylabel('U2')
```

```
  xlabel('time')

 figure(3)
 subplot(311)
   set(gca,'FontName','Times','FontSize',Fs);
   stairs(0:Nsim,[D(1,:) D(1,end)],'b','linewidth',2)
     stairs(0:Nsim,[D(2,:) D(2,end)],'r','linewidth',2)
   ylabel('D')
subplot(312)
set(gca,'FontName','Times','FontSize',Fs);
       plot(0:Nsim,v(1,:),'b','linewidth',2)
       ylabel('v')
subplot(313)
      set(gca,'FontName','Times','FontSize',Fs);
       plot(1:Nsim,w(1,:),'b','linewidth',2)
       ylabel('w')
       xlabel('time')
```

E.2 Soft MPC

```
function [Y,U,R,FIRMPC]=Soft_MPC_MIMO(Himphat,A,B,Bd,C,Qz,
                              S,Sn,snlin,R,Dw,v,x0,umin,umax,
                              dumin,dumax,zmin,zmax,N,Nx,Nu,Nz,Np)
%Soft_MPC- Program for computation and simulation of MPC
% with soft constraints
% The function is used to design, compute and simulate MPC with
%soft constraints using Finite Impulse Response co-efficients
%
%
% Himp - Impulse Response Coefficient
%
% MPC_noise - Function for generating measurement and process noises
```

```
% DesignConstMPC - Function for initialising and designing the
% weightage matrices used for MPC computation
% ClosedLoopSimulationConstMPC - Function for simulating the
% closed loopperformance of the system for MPC with soft
% output constraints
% U -  control input to the system
% D -  Disturbance in the system
% Z-   Output of the system without noise
% Y -  Output of the system including stochastic noises
% v -  Measurement Noise
% w -  Process Noise

%% Design and Simulation of the MPC system %%%%
MPCDataRegSpec=struct('Q',Qz,'S',S,'A',A,'B',B,'C',C, 'Himp',Himphat,
            'umin',umin,'umax',umax,'dumin',dumin,'dumax',dumax, ...
            'zmin',zmin,'zmax',zmax,'Sn',Sn,'snlin',snlin, ...
            'N',N,'Np',Np,'nx',Nx,'nu',Nu,'nz',Nz);


[FIRMPC,FIRMPCmem] = DesignConstMPC_MIMO(Himphat,Qz,S,Sn,snlin,N,
                              umin,umax,dumin,dumax,zmin,zmax);

[Y,Z,U,dU]= ClosedLoopSimulationConstMPC_MIMO(R,Dw,x0,A,B,Bd,C,
                                                  v,FIRMPC,FIRMI
```

E.3 MPC Design

```
function [FIRMPC,FIRMPCmem]=DesignConstMPC_MIMO(Himp,Qz,S,Sn,snlin,
                                         N,umin,umax,dumin,dumax,zmin
[nz,nu,n]=size(Himp);

Gamma = zeros(nz*N,nu*N);
```

```
for i=1:n
    A = diag(ones(N+1-i,1),1-i);
    B = Himp(:,:,i);
    Gamma = Gamma + kron(A,B);
end

Qzcal = kron(diag(ones(N,1),0),Qz);

Sncal = kron(diag(ones(N,1),0),Sn); % Weight Matrix on Soft limits

v = 2*ones(N,1);
v(N,1) = 1;
Hs =    kron(diag(v,0),S) ...
      - kron( diag(ones(N-1,1),-1), S) ...
      - kron( diag(ones(N-1,1), 1), S);

Mr = Gamma'*Qzcal;
H = Mr*Gamma + Hs;
%% Inclusion of soft constraint weights in H matrix

Hn = [H zeros(length(H(:,1))) ; zeros(length(H(1,:))) Sncal];
Hn = 0.5*(Hn+Hn);
sizHn=size(Hn);
Lambda = kron(diag(ones(N,1),0),eye(nu))
                    -kron(diag(ones(N-1,1),-1),eye(nu));
sizlam=size(Lambda);

Abar = [Lambda  zeros(N*nu,N*nz); Gamma  -eye(N*nz);
                                  Gamma eye(N*nz)];
sizAb=size(Abar);
%% structuring the values
FIRMPC = struct(...
```

```
                'H', H, ...
                'Hn', Hn, ...
                'Mr', Mr, ...
                'Lambda', Lambda, ...
                'umin', umin, ...
                'umax', umax, ...
                'zmin', zmin, ...
                'zmax', zmax, ...
                'dumin', dumin, ...
                'dumax', dumax, ...
                'Himp', Himp, ...
                'Qz', Qz, ...
                'S', S, ...
                'Sn', Sn, ...
                'snlin', snlin, ...
                'N', N, ...
                'nz', nz, ...
                'nu', nu, ...
                'Abar', Abar, ...
                'n', n);

FIRMPCmem = struct(...
                    'U0', zeros(nu,N), ...
                    'Uold', zeros(nu,n+1), ...
                    'um1',  zeros(nu,1), ...
                    'ym1',  zeros(nz,1), ...
                    'b0',   zeros(nz,1),...
                    'E0',  zeros(nu,N) );
```

E.4 Closed Loop Simulation

```
function [Y,Z,Uarray,dUarray]= ClosedLoopSimulationConstMPC_MIMO
```

```
(R,D,x0,A,B,Bd,C,v,FIRMPC,FIRMPCmem)

Nsim = size(D,2);
Z = zeros(FIRMPC.nz,Nsim+1);
Y = zeros(FIRMPC.nz,Nsim+1);
Uarray = zeros(FIRMPC.nu,Nsim);
bigval = 1e21;
x = x0;
dUarray = zeros(FIRMPC.nu,Nsim);
um1 = FIRMPCmem.um1;
UFlagGui=1;
yFlagVec=1;
for i=1:Nsim
    Z(:,i) = C*x;
    Y(:,i) = Z(:,i)+v(:,i);
    rArray = repmat(R(:,i),1,FIRMPC.N);
    [u0,info,FIRMPCmem] = ComputeFIRMPC_MIMO(Y(:,i),um1,
                          rArray,bigval,FIRMPC,FIRMPCmem);
    dU=u0-um1;
    um1 = u0;
    Uarray(:,i) = u0;
    dUarray(:,i) = dU;
    x = A*x+B*u0+Bd*D(:,i);
end
Z(:,Nsim+1) = C*x;
Y(:,Nsim+1) = Z(:,Nsim+1)+ v(:,Nsim+1);
```

APPENDIX F

CEMulator

F.1 Application

ECS/CEMulator is a high technological breakthrough in development of an advanced environment for training of process operators and engineers in the cement industry. Combining decades of process design and operation experience of FLSmidth, an extensive theoretical insight on process dynamics, and the latest software technology, FLSmidth Automation has developed an absolute realistic simulator of cement plant processes. The complete details of the ECS/CEMulator package has been given in FLSmidth.

F.1.1 Background

Contrary to most cement process simulators, ECS/CEMulator is developed on a full functional control systems platform enabling the complete set of functions and features of a modern control system environment for the users. Having a skilled team of operators plays a crucial role in beneficial and safe operation of industrial plants. Especially in the cement industry, with the significant high cost of investment, practical knowledge and experience of plant operation have a direct effect on production economy. Insufficient insight in process dynamics and interactions, high stress factors in real time operation conditions, and lack of adequate experience in utilizing the existing control system are typical reasons for incorrect operator actions. The consequences of this may result in low production quality, production interrupts, and equipment damage, in worst case risk on human safety. The increasing demand on production sustainability in the recent years has resulted in requirements of which the degree of fulfillment is effected by the level of skills of plant operators and engineers.

F.1.2 Benefits

ECS/CEMulator is an advanced and user-friendly cement process simulator which aims at:

- Process operator training in an absolutely realistic and risk-free environment
- Increasing operator skills in reaching pre-defined production quantity and quality targets
- Operator performance evaluation
- Increasing operator skills for optimal utilisation of a modern control system
- Enabling process engineers and designers to test their ideas before practical implementations

ECS/CEMulator combines two main data-engines for complete process unit simulation: I) thousands of mathematical model equation are solved to visualize the process dynamics and evolutions, and II) an actual Soft PLC containing a complete process unit PLC program is utilized to enable full digital I/O and sequence control and interlocking in the various groups of the process unit.

F.1.3 Limitations

- CEMulator data engine can be simulated only upto 20 times the real time speed, thus the immediate results on the simulation cannot be obtained
- Since its a property of FLSmidth its not possible to get the internal model design of the software
- The simulation can be only used with ECS SCADA package and cannot be used independently

REFERENCES

1. **Austin, L. G.**, **R. R. Klimpel**, and **P. T. Luckie**, *Process Engineering of Size Reduction: Ball Milling*. Society of Mining Engineers, New York, 1984.

2. **Automation** (2008). *ECS/Process EXpert brochure*. FLSmidth Private Limited.

3. **Bemporad, A.** and **M. Morari**, Robust model predictive control: A survey. *In* **A. Garulli**, **A. Tesi**, and **A. Vicino** (eds.), *Robustness in Identification and Control*, volume 245 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, 1999, 207–226.

4. **Bemporad, A.**, **M. Morari**, **V. Dua**, and **E. N. Pistikopoulos** (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, **38**, 3–20.

5. **Ben-Tal** and **Nemirovski**, Lectures on modern convex optimization. Philadelphia, 2001.

6. **Benzer, H.**, **L. Ergun**, **A. Lynch**, **M. Oner**, **A. Gunlu**, **I. Celik**, and **N. Aydogan** (2001). Modeling cement grinding circuits. *Minerals Engineering*, **vol 14**, 1469–1482.

7. **Boulvin**, **Renotte**, **V. Wouver**, **Remy**, **Tarasiewicz**, and **César** (1999). Modeling, simulation and evaluation of control loops for a cement grinding process. *European Journal of Control*, **5**, 10–18.

8. **Boulvin, M.**, **A. V. Wouwer**, **R. Lepore**, **C. Renotte**, and **M. Remy** (2003). Modeling and control of cement grinding processes. *IEEE Transactions on Control Systems Technology*, **vol 11**, 715–725.

9. **Boulvin, M.**, **A. V. Wouwer**, **C. Renotte**, **M. Remy**, and **R. Lepore**, Some observations on modeling and control of cement grinding circuits. *In Proceedings of the American Control Conference*. AACC, Philadelphia, Pennsylvania, 1998.

10. **Boyd, S.** and **C. Baratt**, *Linear Controller Design, Limits of Performance*. Prentice Hall, 1991.

11. **Boyd, S.** and **L. Vandenberghe**, *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

12. **Brosilow, C.** and **B. Joseph**, *Techniques of Model based Control*. Prentice Hall, 2002.

13. **Camacho** and **Bordons**, *Model Predictive Control*. Springer-Verlag, 1999.

14. **Camacho** and **Bordons**, *Model Predictive Control*. Springer, 2004.

15. **Cao, H.**, **G. Si**, **Y. Zhang**, **X. Ma**, and **J. Wang** (2008). Load control of ball mill by a high precision sampling fuzzy logic controllerwith self-optimizing. *Asian Journal of Control*, **10**, 621–631.

16. **Chen, X.-S.**, **Q. Li**, and **S.-M. Fei** (2008). Constrained model predictive control in ball mill grinding process. *Powder Technology*, **vol 186**, 31–39.

17. **Chen, X.-S.**, **S.-H. Li**, **J.-Y. Zhai**, and **Q. Li** (2009). Expert system based adaptive dynamic matrix control for ball mill grinding circuit. *Expert Systems with Applications*, **Vol.36,**, 716–723.

18. **Chidambaram, M.**, *Computer Control of Processes*. Narosa Publishing House, 2003.

19. **Clarke, D. W.** (1988). Application of generalized predictive control to industrial processes.

20. **Cleary, P. W.** (2006). Axial transport in dry ball mills. *Applied Mathematical Modelling*, **30**, 1343–1355.

21. **Concil, W. E.**, *Efficient Use of Energy Utilizing High Technology: An Assessment of Energy Use in Industry and Buildings*. World Energy Council, London, United Kingdom, 1995.

22. **Cutler** and **Ramakar** (1980). Dynamic matrix control - computer control algorithm. *Proceedings of the Joint Automatic Control Conference*.

23. **de Haas, B.**, **V. Werbrouck**, **G. Bastin**, and **V. Wertz**, Cement mill optimization: Design parameters selection of the LQG controller. 1995.

24. **Efe, M. Ö.** and **O. Kaynak**, Multivariable nonlinear model reference control of cement mills. *In IFAC 15th World Congress*. IFAC, Barcelona, Spain, 2002.

25. **Fletcher, R.**, *Practical Methods of Optimization*. wiley, 1987, second edition edition.

26. **FLSmidth** (). *ECS/CEmulator - the most realistic environment for operator training in the cement industry*. FLSmidth A/S- Automation.

27. **FLSmidthA/S** (2004). *FLSmidth Manual*. FLSmidthPrivate Limited.

28. **Fujimoto, S.** (1993). Reducing specific power usage in cement plants. *World Cement*, **vol 7**, 25–35.

29. **Garcia**, **Prett**, and **Morari** (1989). Model predictive control: theory and practice. *Automatica*, **25**, 335–348.

30. **Garriga, J. L.** and **M. Soroush** (2010). Model predictive control tuning methods : A review. *Ind. Engg. Chem. Res.*, **49**, 3505–3515.

31. **Gill, P.**, **W. Murray**, and **M. Wright**, *Practical Optimization*. Academic Press, London, 1981.

32. **Goodwin, G. C.**, **H. Haimovich**, **D. E. Quevedo**, and **J. S. Welsh** (2004). A moving horizon approach to networked control system design. *IEEE transactions on Automatic control*, **49**(9), 1427–1445.

33. **Goodwin, G. C.**, **M. M. Seron**, and **J. A. De Dona**, *Constrained Control and Estimation. An Optimization Approach*. Springer, New York, 2005.

34. **Grognard, F.**, **F. Jadot**, **L. Magni**, **G. Bastin**, **R. Sepulchre**, and **V. Wertz** (2001). Robust stabilization of a nonlinear cement mill model. *IEEE Transactions on Automatic Control*, **vol 46**, 618–623.

35. **GuruPrasath** and **Jorgensen**, Soft constraints for robust MPC of uncertain systems. *In ADCHEM 2009*. IFAC, Koc University, Istanbul, Turkey, 2009.

36. **GuruPrasath**, **J. B. Jorgensen**, and **Chidambaram** (2011). Control of cement ball mill circuit with large sample delay using mpc. *Powder Technology*.

37. **Guzman**, **Alamo**, **Berenguel**, **Dormido**, and **Camacho** (2009). A robust constrained reference governor approach using linear matrix inequalities. *Journal of Process Control*, **19**, 773–784.

38. **Hansson, A.** (2000). A primal-dual interior-point method for robust optimal control of linear discrete-time systems. *IEEE Transactions on Automatic Control*, **45**, 1639–1655.

39. **Havlena, V.** and **J. Findejs** (2005). Application of model predictive control to advanced combustion control. *Control Engineering Practice*, **13**, 671–680.

40. **Havlena, V.** and **J. Lu**, A distributed automation framework for plant-wide control, optimisation, scheduling and planning. *In 16th IFAC World Congress in Prague*. 2005.

41. **Herbst, J.**, **W. Pate**, and **A. Oblad** (1992). Model-based control of mineral processing operations. *Powder Technology*, **vol 69**, 21–32.

42. **Huusom, J. K.**, **A. D. Jensen**, **S. B. Jørgensen**, **M. Michelsen**, **J. Knudsen**, **B. Recke**, and **J. B. Jørgensen**, Modelling of cement grinding circuits for predictive control. *In IFAC World Congress 2005*. IFAC, Prague, Czech Republic, 2005.

43. **Iino**, **Y. Tomida**, **K. Fujiwara**, **H. Takagi**, **Y. Shigemasa**, and **T. Yamamoto** (1993). A new input/output mpc with frequency domain technique and its application to ethylene plant. *Industrial Electronics, Control and Instrumentation*, **1**, 457–462.

44. **Jankovic, A.**, **W. Valery**, and **E. Davis** (2004). Cement grinding optimisation. *Minerals Engineering*, **vol 17**, 1075–1081.

45. **Jayasundara**, **Yang**, **Yu**, and **Curry** (2008). Discrete particle simulation of particle flow in IsaMill - effect of grinding media properties. *Chemical Engineering Journal*, **vol 135**, 103–112.

46. **Jørgensen, J. B.**, **J. B. Rawlings**, and **S. B. Jørgensen**, Numerical methods for large scale moving horizon estimation and control. DYCOPS -7, Cambridge, MA, 2004.

47. **Karmarkar, N.** (1984). A new polynomial time algorithm for linear programming. *Combinatorica*, **4**, 373–395.

48. **Kassmann, D. E.**, **T. A. Badgwell**, and **R. B. Hawkins** (2000). Robust steady-state target calculation for model predictive control. *AIChE Journal*, **46**(5), 1007–1024.

49. **Kerrigan, E.** and **J. M. Maciejowski** (2000). Soft constraints and exact penalty functions in MPC. *Control Conference, Cambridge*.

50. **Kojima, A.** and **M. Morari** (2004). Lq control for constrained continuous-time systems. *Automatica*, **40**, 1143–1155.

51. **Lepore, R.**, **A. Wouwer**, and **M. Remy**, Nonlinear model predictive control of cement grinding circuits. *In ADCHEM 2003*. IFAC, Hong-Kong, 2003.

52. **Lepore, R.**, **A. V. Wouwer**, and **M. Remy**, Modeling and predictive control of cement grinding circuits. *In 15th IFAC World Congress*. IFAC, Barcelona, Spain, 2002.

53. **Lepore, R.**, **A. V. Wouwer**, **M. Remy**, and **P. Bogaerts**, State and parameter estimation in cement grinding circuits - practical aspects. *In DYCOPS-7*. IFAC, Cambridge, Massachussets, 2004.

54. **Lepore, R.**, **A. V. Wouwer**, **M. Remy**, and **P. Bogaerts**, Receding-horizon estimation and control of ball mill circuits. *In* **R. Findeisen**, **F. Allgöwer**, and **L. T. Biegler** (eds.), *Asssessment and Future Directions of Nonlinear Model Predictive Control*. Springer, Berlin, 2007*a*, 485–493.

55. **Lepore, R.**, **V. Wouwer**, **M. Remy**, and **P. Bogaerts**, Receding-horizon estimation and control of ball mill circuits. *In* **R. Findeisen**, **F. Allgöwer**, and **L. T. Biegler** (eds.), *Assessment and Future Directions of Nonlinear Model Predictive Control*. Springer, New York, 2007*b*, 485–493.

56. **Limon, D.**, **T. Alamo**, **F. Salas**, and **E. Camacho** (2006). On the stability of constrained mpc without terminal constraint. *IEEE transactions on Automatic control*, **51**(5), 832–836.

57. **Lin, C.-M.** and **P.-N. Chin** (1996). Cement roller press control by fuzzy logic reasoning. *Industrial Electronics*, **1**, 384–389.

58. **Maciejowski, J.**, *Predictive Control with Constraints*. Prentice Hall, Harlow, England, 2002.

59. **Magni**, **B. G**, and **W. V** (1999). Multivariable nonlinear predictive control of cement mills. *IEEE. Transactions on Control System Technologies*, **vol 7**, 502–508.

60. **Marafloti, G.**, **R. Bitmead**, and **M. Hovd**, Persistently exciting model predictive control using fir models. *In International conference on Cybernatics and Informatics*. 2010.

61. **Martin, G.** and **S. McGarel** (2001). Nonlinear mill control. *ISA Transactions*, **vol 40**, 369–379.

62. **Mayne, D.**, **J. Rawlings**, **C. Rao**, and **P. Scokaert** (2000). Constrained model predictive control: Stability and optimality. *Automatica*, **36**, 789–814.

63. **Mehrotra, S.** (1992). On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, **2**, 575–601.

64. **Miller, R. H.**, **I. kolmanovsky**, **E. G. Gilbert**, and **P. D. Washabaugh** (2000). Control of constrained nonlinear systems: A case study. *IEEE Control Systems Magazine*.

65. **Morari, M.** and **J. Lee** (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering*, **23**, 667–682.

66. **Muske, K. R.** and **J. B. Rawlings** (1993*a*). Model predictive control with linear models. *AIChE Journal*, **39**, 262–287.

67. **Muske, K. R.** and **J. B. Rawlings** (1993*b*). Model predictive control with linear models. *AIChE Journal*, **39**, 262–287.

68. **Powell, M.** and **A. McBride** (2006). What is required from DEM simulations to model breakage in mills? *Minerals Engineering*, **19**, 1013–1021.

69. **Prasath** and **Jorgensen**, Model predictive control based on finite impulse response models. *In ACC2008*. American Control Conference, Seattle, Washington, 2008.

70. **Prasath, G.** and **J. B. Jørgensen**, Moving horizon estimation based on finite impulse response models. *In Proceedings of the European Control Conference 2009*. 2009.

71. **Qin, S. J.** and **T. A. Badgwell** (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, **11**, 733–764.

72. **Rajamani** and **J. A. Herbst** (1991*a*). Optimal control of a ball mill grinding circuit - II. feedback and optimal control. *Chemical Engineering Science*, **vol 46**, 871–879.

73. **Rajamani, R. K.** and **J. A. Herbst** (1991*b*). Optimal control of a ball mill grinding circuit - i. grinding circuit modeling and dynamic simulation. *Chemical Engineering Science*, **vol 46**, 861–870.

74. **Ramasamy, M.**, **S. Narayanan**, and **C. Rao** (2005). Control of ball mill grinding circuit using model predictive control scheme. *Journal of Process Control*, **vol 15**, 273–283.

75. **Rao, C. V.**, **J. . B. Rawlings**, and **J. H. Lee** (2001). Constrained linear state estimation - a moving horizon approach. *Automatica*, **37**, 1619–1628.

76. **Rao, C. V.** and **J. B. Rawlings** (2002). Constrained process monitoring: A moving horizon approach. *AIChE Journal*, **48**, 97–109.

77. **Rhinehart, R.**, **M. Darby**, and **H. Wade** (2011). Editorial- choosing advanced control. *Editorial/ ISA Transactions*, **50**, 2–10.

78. **Richalet**, **Rault**, **Testud**, and **Papon** (1976). Algorithmic control of industrial processes. *4th IFAC symposium on identi?cation and system parameter estimation*, 1119–1167.

79. **Robertson** and **Lee** (1995). A least squares formulation for state estimation. *Journal of Process Control*, **5**, 291–299.

80. **Robertson**, **Lee**, and **Rawlings** (1996). A moving horizon based approach for least-squares estimation. *AIChE Journal*, **42**, 2209–2224.

81. **Rossiter, J.**, *Model based Predictive Control - A practical approach*. CRC Press, 2003.

82. **Roubal, J.** and **V. Havlena** (2005). Range control MPC approach for two-dimensional system. *IFAC World Congress 2005*, **vol 16**.

83. **Santos, L.**, **L. Biegler**, and **J. Castro** (2008). A tool to analyze robust stability for constrained nonlinear mpc. *Journal of Process Control*, **18**, 383–390.

84. **Scokaert, P. O. M.** and **J. B. Rawlings** (1999). Feasibility issues in linear model predictive control. *AIChE Journal*, **vol 45**, 1649–1659.

85. **Shao, J.**, *Mathematical Statistics*. Springer, 2003.

86. **Shinskey, F. G.**, *Process Control Systems. Application, Design, and Tuning*. McGraw-Hill, New York, 1988.

87. **Shridhar, R.** and **D. Cooper** (1997). A tuning strategy for unconstrained siso model predictive control. *Industrial Eng. Chemical Res.,*, **36**, 729–746.

88. **Topalov, A. V.** and **O. Kaynak** (2004). Neural network modeling and control of cement mills using a variable structure systems theory based on-line learning mechanism. *Journal of Process Control*, **vol 14**, 581–589.

89. **Touil**, **Belaadi**, and **Frances** (2008). The specific selection function effect on clinker grinding efficiency in a dry batch ball mill. *International Journal of Mineral Processing*, **87**, 141–145.

90. **Van Breusegem, V.**, **L. Chen**, **G. Bastin**, **V. Wertz**, **V. Werbrouck**, and **C. de Pierpont** (1996*a*). An industrial application of multivarible linear quadratic control to a cement mill circuit. *IEEE Transactions on Industry Applications*, **vol 32**, 670–677.

91. **Van Breusegem, V.**, **L. Chen**, **G. Bastin**, **V. Wertz**, **V. Werbrouck**, and **C. de Pierport** (1996*b*). An industrial application of multivariable linear quadratic control to a cement mill. *International Journal of Mineral Processing*, **44-45**, 405–412.

92. **Van Breusegem, V.**, **L. Chen**, **V. Werbrouck**, **G. Bastin**, and **V. Wertz** (1994). Multivariable linear quadratic control of a cement mill: An industrial application. *Control Engineering Practice*, **vol 2**, 605–611.

93. **Vandenberghe, L.**, **S. Boyd**, and **M. Nouralishadi**, Robust linear programming and optimal control. *In 15th IFAC World Congress 2002*. IFAC, Barcelona, Spain, 2002.

94. **Wardana, A. N.**, Pid-fuzzy controller for grate cooler in cement plant. *In Asian Control Conference*. 2004.

95. **Warren, A. L.** and **T. E. Marlin**, Constrained mpc under closed loop uncertainty. *In Proceedings of American Control Conference*. Boston, 2004.

96. **Wertz**, **Magni**, and **Bastin**, Multivariable nonlinear control of cement mills. *In* **F. Allgöwer** and **A. Zheng** (eds.), *Nonlinear Model Predictive Control*. Birkhäuser, Basel, Switzerland, 2000, 433–447.

97. **Zheng** and **M. Morari** (1995). Stability of model predictive control with mixed constraints. *Automatic control*, **40**(10), 1818–1823.

98. **Zheng, A.** (1998). Robust stability analysis of constrained model predictive control. *Proceedings of American Control Conference*, 786–790.

99. **ZimmerMann, H.**, *Fuzzy Set Theory and its Applications*. Allied Publishers Limited, 1996.

RELATED PUBLICATIONS

1. **M. GuruPrasath and John Bagterp Jørgensen**, “Model predictive control based on finite impulse response models”, American Control Conference 2008, Seattle, Washington, USA, June 11-13, 2008 ISBN 978-1-4244-2079-7.

2. **M. GuruPrasath and John Bagterp Jørgensen**, “Soft constraints for robust MPC of uncertain systems", ADCHEM 2009 Istanbul, Turkey: International Federation of Automatic Control (IFAC), Koc University, 2009.

3. **M. GuruPrasath and John Bagterp Jørgensen**, “Moving Horizon Estimation based on Finite Impulse Response Models”, Proceedings of ECC 2009, Budapest, Hungary, August 23-26, 2009 ISBN 978-963-311-369-1.

4. **M. GuruPrasath, Bodil Recke, M. Chidambaram and John Bagterp Jørgensen**, “Application of Soft Constrained MPC to a Cement Mill Circuit”,Dynamic control of Process Systems (DYCOPS 2010), (July 5-7, 2010), Leuven , Belgium.

5. **M. GuruPrasath, Bodil Recke, M. Chidambaram and John Bagterp Jørgensen**, “Robust MPC for a Cement Grinding Circuit”,Submitted to European Symposium on Computer-Aided Process Engineering -21,(Abstract has been conditionally accepted for full paper), [Special issue to be published in Computer Aided Chemical Engineering, Elsevier Science in 2011]

6. **M. GuruPrasath, Bodil Recke, M. Chidambaram and John Bagterp Jørgensen**, “Soft Constrained MPC to a Cement Mill Circuit”,Manuscript submitted to Journal of Process Control.

7. **M. GuruPrasath, John Bagterp Jørgensen and M. Chidambaram** , “Control of cement ball mill circuit with large sample delay using MPC”,Manuscript submitted to Powder Technology.