NONLINEAR CONTROL WITH STATE ESTIMATION AND POWER OPTIMIZATION FOR A ROM ORE MILLING CIRCUIT

by

Myrin Anand Naidoo

Submitted in partial fulfilment of the requirements for the degree

Master of Engineering (Electronic Engineering)

in the

Department of Electrical, Electronic and Computer Engineering
Faculty of Engineering, Built Environment and Information Technology
UNIVERSITY OF PRETORIA

October 2014

SUMMARY

NONLINEAR CONTROL WITH STATE ESTIMATION AND POWER OPTIMIZATION FOR A ROM ORE MILLING CIRCUIT

by

Myrin Anand Naidoo

Supervisor(s):	Prof I. K. Craig
Co-supervisor:	Mr J. D. le Roux
Department:	Electrical, Electronic and Computer Engineering
University:	University of Pretoria
Degree:	Master of Engineering (Electronic Engineering)
Keywords:	comminution, dynamic inversion, milling, model predictive control, neural network, particle filter, power optimization, run-of-mine, state estimation

A run-of-mine ore milling circuit is primarily used to grind incoming ore containing precious metals to a particle size smaller than a specification size. A traditional run-of-mine (ROM) ore single-stage closed milling circuit comprises of the operational units: mill, sump and cyclone. These circuits are difficult to control because of significant nonlinearities, large time delays, large unmeasured disturbances, process variables that are difficult to measure and modelling uncertainties. A nonlinear model predictive controller with state estimation could yield good control of the ROM ore milling circuit despite these difficulties. Additionally, the ROM ore milling circuit is an energy intensive unit and a controller or power optimizer could bring significant cost savings.

A nonlinear model predictive controller requires good state estimates and therefore a neural network for state estimation as an alternative to the particle filter has been addressed. The neural network approach requires fewer process variables that need to be measured compared to the particle filter. A neural network is trained with three disturbance parameters and used to estimate the internal states of the mill, and the results are compared with those of the particle filter implementation. The neural network approach performed better than the particle filter approach when estimating the volume of steel balls and rocks within the mill. A novel combined neural network and particle filter state es-

timator is presented to improve the estimation of the neural network approach for the estimation of volume of fines, solids and water within the mill. The estimation performance of the combined approach is promising when the disturbance magnitude used is smaller than that used to train the neural network.

After state estimation was addressed, this work targets the implementation of a nonlinear controller combined with full state estimation for a grinding mill circuit. The nonlinear controller consists of a suboptimal nonlinear model predictive controller coupled with a dynamic inversion controller. This allows for fast control that is asymptotically stable. The nonlinear controller aims to reconcile the opposing objectives of high throughput and high product quality. The state estimator comprises of a particle filter for five mill states as well as an additional estimator for three sump states. Simulation results show that control objectives can be achieved despite the presence of noise and significant disturbances.

The cost of energy has increased significantly in recent years. This increase in price greatly affects the mineral processing industry because of the large energy demands. A run-of-mine ore milling circuit provides a suitable case study where the power consumed by a mill is in the order of 2 MW. An attempt has been made to reduce the energy consumed by the mill in the two ways: firstly, within the nonlinear model predictive control in a single-stage circuit configuration and secondly, running multiple mills in parallel and attempting to save energy while still maintaining an overall high quality and good quantity. A formulation for power optimization of multiple ROM ore milling circuits has been developed. A first base case consisted not taking power into account in a single ROM ore milling circuit and a second base case split the load and throughput equally between two parallel milling circuits. In both cases, energy can be saved using the NMPC compared to the base cases presented without significant sacrifice in product quality or quantity.

The work presented covers three topics that has yet to be addressed within the literature: a neural network for mill state estimation, a nonlinear controller with state estimation integrated for a ROM ore milling circuit and power optimization of a single and multiple ROM ore milling circuit configuration.

OPSOMMING

NIE-LINEÊRE BEHEER MET TOESTANDSAFSKATTING EN ENERGIE-OPTIMERING VIR ’N MAALKRING WAT ONBEHANDELDE ERTS MAAL

deur

Myrin Anand Naidoo

Studieleier(s):	Prof I. K. Craig
Medestudieleier:	Mr J. D. le Roux
Departement:	Elektriese, Elektroniese en Rekenaar-Ingenieurswese
Universiteit:	Universiteit van Pretoria
Graad:	Magister in Ingenieurswese (Elektroniese Ingenieurswese)
Sleutelwoorde:	dinamiese inversie, energie-optimering, maling, model voorspellende beheer, neurale netwerk, onbehandelde erts, partikel-filter, toestandsafskatting, vermaling

’n Maalkring wat onbehandelde erts maal word hoofsaaklik gebruik om erts, wat kosbare metale bevat, fyn te maal tot ’n partikel grootte kleiner as ’n sekere spesifikasie. ’n Tradisionele enkelfase geslote-lus maalkring bevat die volgende funksionele eenhede: ’n meul, opvangbak en sikloon. Hierdie maalkringe is moeilik om te beheer as gevolg van aansienlike nie-lineariteite, groot tydsvertragings, sterk eksterne steurings, prosesveranderlikes wat moeilik is om te meet en onsekerhede in modelle. ’n Nie-lineêre-model voorspellende beheerder (NMVB) met toestandsafskatting kan die maalkring goed beheer ten spyte van hierdie probleme. Verder verbruik die maalkring baie energie en ’n beheerder of energie-optimeerder kan beduidende koste-besparings teweeg bring.

’n Nie-lineêre-model voorspellende beheerder benodig akkurate afgeskatte toestandswaardes en as sulks word ’n neurale netwerk vir toestandsafskatting as alternatief tot ’n partikel-filter aangespreek. Die neurale netwerk benadering benodig minder prosesveranderlikes wat gemeet is as die partikelfilter. Die neurale netwerk is opgelei met drie steuringsveranderlikes en word gebruik om die interne toestande van die meul te beraam. Die resultate verkry deur hierdie benadering word vergelyk met die verkry deur van ’n partikel-filter gebruik te maak. Die neurale netwerk benadering verkry beter resultate as die partikel-filter wanneer die volume van staal balle en rotse binne die meul beraam word.

’n Nuwe gekombineerde neurale netwerk en partikel-filter toestandsafskatter word voorgestel om op die resultate te verbeter wat die neurale netwerk verkry wanneer die volume van fyn erts, soliede erts en water binne die meul beraam word. Die akkuraatheid van afskatting wat die gekombineerde afskatter behaal is belowend in die teenwoordigheid van eksterne steurings, indien hierdie steurings minder sterk is as die wat gebruik is tydens die opleiding van die neurale netwerk.

Nadat toestandsafskatting aangespreek is, word daar gefokus op die implementasie van ’n nie-lineêre beheerder gekombineerd met volle toestandsafskatting vir ’n geslote-lus maalkring. Die nie-lineêre beheerder bestaan uit ’n sub-optimale nie-lineêre-model voorspellende beheerder en ’n dinamiese inversie beheerder. Dit laat toe vir vinnige beheer wat asimptoties stabiel is. Die nie-lineêre beheerder probeer om die teenstrydige doelwitte van hoë deurset en hoë kwaliteit te versoen. Die toestandsafskatter bestaan uit ’n partikel-filter vir die afskatting van die vyf toestande van die meul asook ’n addisionele afkatter vir die drie toestande van die opvangbak. Simulasieresultate dui aan dat beheerdoelwitte behaal kan word in die teenwoordigheid van ruis en sterk eksterne steurings.

Die koste van energie het drasties toegeneem in die afgelope paar jaar. Hierdie toename het ’n groot impak op die mineraalprosesseeringsbedryf aangesien die energieaanvraag so hoog is. ’n Maalkring wat onbehandelde erts maal is ’n goeie voorbeeld hiervan aangesien die energieverbruik van ’n meul in die orde van 2 MW is. ’n Poging is aangewend om die energie wat verbruik word in die meul te verminder op twee maniere: eerstens, deur die nie-lineêre-model voorspellende beheer van ’n enkel-fase maalkring, en tweedens wanneer ’n versameling meule in parallel bedryf word, om energie te bespaar en steeds ’n hoë kwaliteit en kwantiteit van produk te handhaaf. ’n Formulering vir energiebesparing wanneer ’n versameling maalkringe in parallel bedryf word, word voorgelê. ’n Grondslag vir vergelyking is opgestel vir wanneer ’n enkele maalkring beheer word sonder om energieverbruik in ag te neem, en ’n tweede grondslag is opgestel vir wanneer twee maalkringe in parallel bedryf word en die deurset en vrag gelyk tussen die twee gedeel word. In beide gevalle kan energiebesparings verkry word deur die NMVB (in vergelyking met die grondslag) sonder merkwaardige opoffering van die kwaliteit of kwantiteit van die produk.

Hierdie werk spreek drie onderwerpe aan wat nog nie voorheen in die literatuur behandel is nie, naamlik: ’n neurale netwerk vir toestandsafskatting van die meul, ’n nie-lineêre beheerder met geïntegreerde toestandsafskatting vir ’n maalkring wat onbehandelde erts maal, en energie-optimering vir ’n enkele maalkring asook vir veelvoudige maalkringe in parallel.

LIST OF ABBREVIATIONS

CV	Controlled variables
DI	Dynamic inversion
FOPTD	First order plus time delay
MIMO	Multiple-input multiple-output
MPC	Model predictive control
MV	Manipulated variables
NMPC	Nonlinear model predictive control
OP	Operating point
PDF	Probability density function
PID	Proportional-integral-derivative
RNMPC	Robust nonlinear model predictive controller
ROM	Run-of-mine

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The main goal of minerals processing is to convert ore with a low concentration of valuable minerals to a final product that contains a much higher concentration of the most valuable minerals. This circuit forms part of the minerals liberation process (Hodouin, 2011). These circuits are difficult to control because of the significant nonlinearities, large time delays, large unmeasured disturbances, process variables that are difficult to measure and modelling uncertainties (Olivier, Craig and Chen, 2012a; Remes, Aaltonen and Koivo, 2010). A nonlinear model predictive controller with state estimation could yield good control of the ROM ore milling circuit despite these difficulties. Additionally, the ROM ore milling circuit is an energy intensive unit and a controller or power optimizer could bring significant cost savings.

Nonlinear control makes use of a plant model that can cater for multiple equilibrium points whereas a linear or linearized model only considers one equilibrium point. Since the ROM ore milling circuit can operate at various set-points for mill load and circuit throughput, it would be of significant advantage to make use of a nonlinear model. This study addresses the use of a nonlinear controller as a control solution for the ROM ore milling circuit. A nonlinear controller requires state information, i.e. the volume of components within the mill and sump. Good state estimation can result in better control of the milling circuit and this study determines if a neural network can be used for state estimation as an alternative to the particle filter approach (Olivier, Huang and Craig (2012b)). A combination of a neural network and a particle filter is also investigated to improve the estimation results of the neural network approach by utilizing the particle filters estimation for the volume of fines, solids and water within the mill. A final investigation topic addresses the power consumption within a single and multiple mill configuration. Power can be saved in two scenarios; firstly, by trying to minimize the power draw from the mill motor when aiming to achieve set-point, and secondly, in a two parallel

mill configuration, by identifying the more efficient milling circuit and determining various mill load and throughput set-points to minimize total power consumption.

1.1 MOTIVATION AND BACKGROUND

The ROM ore milling circuit is the first step in the transformation process and improving the output of this process will improve all further processing units. The first and main objective is to keep the product quality and quantity consistent as possible while ensuring the load within the mill and sump volume does not exceed the limits. The ROM ore milling circuit is a difficult process to control. Factors that make the ROM ore milling circuit difficult to control are (Hodouin, Jamsa-Jounela, Carvalho and Bergh, 2001; Chen, Li and Fei, 2008):

- real-time measurements of component flow-rates (e.g. water flow-rate exiting the mill),
- characteristics of feed ore change and are not always available,
- reliability of measuring devices vary due to environmental conditions,
- constraints on manipulated and controlled variables,
- measurement noise, and
- inaccuracies in on-line measurements of flow-rates and particle size distributions.

Model predictive control (MPC) has the ability to handle constraints on manipulated and controlled variables and has been implemented in industries, such as the petrochemical industry, over the past three decades (Lee, 2011). The constraint handling ability of MPC ensures that the mill and sump are within operating regions and prevents undesirable scenarios such as mill overloading (Bouche, Brandt, Broussaud and van Wayne Drunick, 2005). A successful implementation of constraint handling was completed by Chen, Li, Fei and Zhai (2007a), where a hybrid control approach was used. The hybrid control approach combined override control and MPC to prevent overload of the mill and obtain an optimal feed rate set-point. A robust nonlinear model predictive controller (RNMPC) was developed by Coetzee, Craig and Kerrigan (2010) and showed good results in maintaining product quality despite disturbances. However, the study assumed full-state feedback and therefore a nonlinear controller with state estimation integrated has previously been unsolved.

A particle filter was used for state estimation by Olivier *et al.* (2012b) and can be seen as the most applicable to the current ROM ore milling circuit. The results showed that the first three states were estimated accurately despite disturbances; however, there is room for improvement when estimating the volume of rocks and steel balls within the mill. Additionally, the particle filter requires ambitious measurements, such as the component flow rates, in and out of the mill. This promotes the development of a state estimation method that only takes into account practical or traditional measurements such as particle size estimation, sump level, mill load, and the cyclone feed or sump density (Wei and Craig, 2009).

The two major objectives for implementing control on the ROM ore milling circuit are to stabilize the process as well as to optimize the economic performance of the process (Craig and MacLeod, 1995). A subset of objectives that contribute to these major objectives are:

i. to improve the quality of the product either by increasing fineness or by decreasing variations in product size,

ii. to maximize throughput,

iii. to minimize the resources such as steel balls used to produce the final product, and

iv. to minimize the power consumed for each ton of fines produced.

One can notice that it is impossible to satisfy all objectives at the same time, since sub-objectives (i) and (ii) have a negative effect on objectives (iii) and (iv). Hence, a trade-off between these objectives has to be found. The cost of electricity has increased significantly over the past few years and will continue to do so (Creamer, 2013; Matthews and Craig, 2013). The increasing cost motivates control objective (iv). However, aiming to minimize power could affect the product quality (objective (i)) and therefore affect recovery, the desired material after separation processes.

1.2 CONTRIBUTION

The following contributions have resulted from this work and are presented in the following order:

1. A neural network as an alternative state estimator for the mill model.

2. A combination of a neural network and a particle filter as a state estimator for a mill module.

3. A new control configuration for the ROM ore milling circuit was developed.

4. A state estimator/calculator for the sump is covered.

5. Simulation results with a nonlinear model predictive controller and a dynamic inversion controller are presented.

6. A combination of state estimation with a nonlinear controller is presented.

7. Power optimization within the nonlinear model predictive control is considered.

8. A conceptual supervisory power optimizer for multiple parallel ROM ore milling circuits has been developed.

1.3 PUBLICATIONS

The following publications have resulted from this work:

- Naidoo, M.A., Padhi, R., Craig, I.K., Olivier, L.E., le Roux, J.D., 2014, Augmented nonlinear MPC for control of a grinding mill circuit, Control Engineering Practice, Submitted for Review.

- Naidoo, M.A., Padhi, R. and Craig, I.K. August 2014., A New Nonlinear and Suboptimal Control Design Approach for Milling Circuits, In: Proc: 19th IFAC World Congress, Cape Town, South Africa.

- Naidoo, M., Olivier, L. and Craig, I. December 2013., Combined neural network and particle filter state estimation with application to a run-of-mine ore mill, 10th International Symposium on Dynamics and Control of Process Systems, pp. 397-402, Mumbai, India.

1.4 OVERVIEW OF STUDY

Fig. 1.1 illustrates a typical feedback control loop consisting of a plant, controller, measuring device and reference values. This study aims to complete this control loop by targeting each block within the control loop in a chapter. Chapter 2 describes the ROM ore milling circuit that was used through-

out the study. Chapter 3 proposes a combined neural network and particle filter approach for state estimation, essentially providing approximate values of the contents of the mill and sump. Chapter 4 consists of the new nonlinear control configuration integrated with state estimation. Chapter 5 looks at power optimization and development of a supervisory power optimizer for a parallel mill configuration.

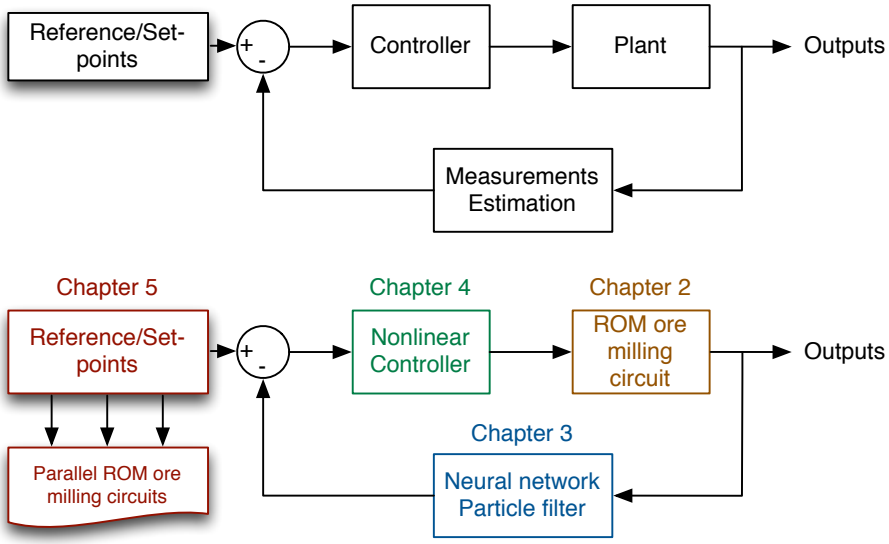

Figure 1.1: Overview of study.

CHAPTER 2

RUN-OF-MINE ORE MILLING CIRCUIT

2.1 INTRODUCTION

This chapter aims to provide a description of a ROM ore milling circuit and presents the milling circuit in state space format which is used in designing a state estimator, nonlinear controller and power optimizer in Chapters 3, 4 and 5 respectively. A brief background on the history of control of grinding mill circuits is covered.

2.2 DESCRIPTION OF THE RUN-OF-MINE ORE MILLING CIRCUIT

The goal of mineral processing is to convert ROM ore into a product that has a high concentration of the most valuable minerals. The ROM ore milling circuit, shown in Fig. 2.1, is the focus of this study, and is used to liberate valuable minerals such that it can be concentrated. The major disturbances affecting the ROM ore milling circuit are variations in feed size, grindability, model-mismatch and variables that are difficult to measure (Coetzee *et al.*, 2010; Remes *et al.*, 2010). For completeness, a brief description of the process is provided here, details of which can be found in Coetzee *et al.* (2010).

In the ROM ore milling circuit, ore containing valuable minerals (such as copper, platinum or gold) is fed into the mill where it is ground to a fine product. A semi-autogenous grinding (SAG) mill uses rocks and steel balls as the grinding medium and is used throughout this work. The mill discharges into a sump where the slurry is diluted with water to achieve the correct density before it is pumped to a hydrocyclone. Ore that cannot exit the mill (because of the mill discharge grate) is referred to as rocks whereas solids are able to pass through the mill discharge grate. Solids consist of product that

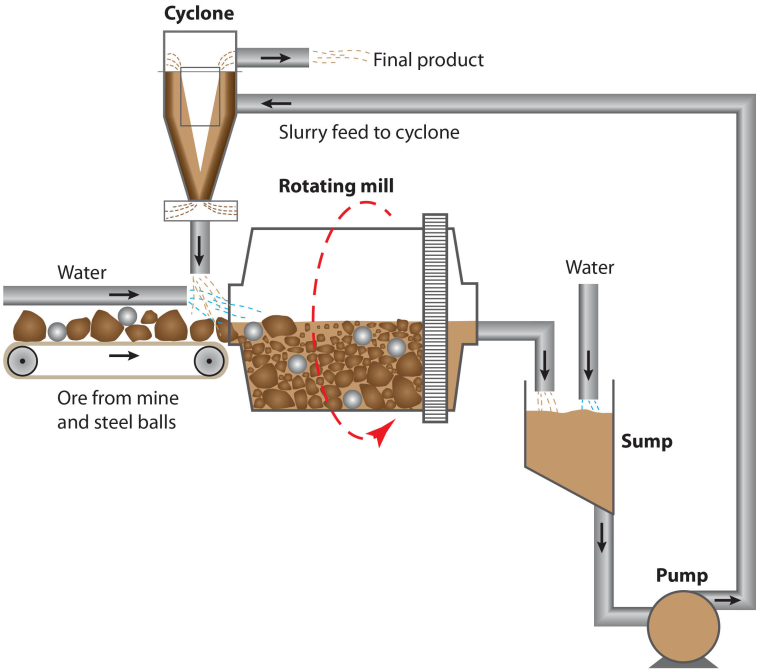

Figure 2.1: Run-of-mine milling circuit.

meet the specification size (fines) as well as product that is larger than the product specification size (coarse). The cyclone separates the coarse and fine particles, with the fine particles leaving the circuit as product while the coarse particles are recycled into the mill for further grinding.

A relationship between recovery, the desired material after the separation processes, and particle size (PSE) is given in Matthews and Craig (2013) and shown in Fig. 2.2. Results show that a finer grind results in better recovery, but this results in the throughput decreasing which increases the operating cost per ton produced.

2.2.1 Controlled and manipulated variables

In this study, manipulated variables include the mill feed solids (MFS), the mill inlet water (MIW), mill feed steel balls (MFB), the flow rate of the sump feed water (SFW), the cyclone feed flow (CFF) and the mill speed (α_{speed}). Table 2.1 shows the constraints and operating values for the manipulated variables and controlled variables. These operating values from Le Roux, Craig, Hulbert and Hinde (2013b) are used in Chapter 4 and the operating values from Coetzee *et al.* (2010) are used in Chapter 5.

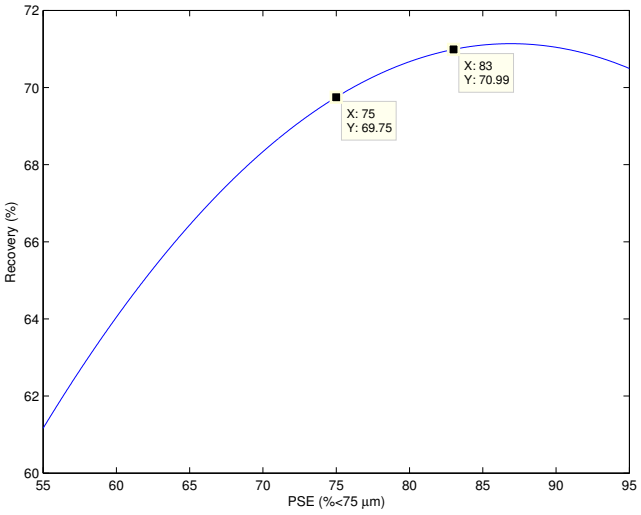

Figure 2.2: Recovery as a function of particle size.

The controlled variables for the milling circuit are typically the particle size estimate (PSE), percentage of the mill volume filled ($LOAD$), sump volume ($SVOL$) and cyclone feed density (CFD) (Wei and Craig, 2009). In this study, another controlled variable was introduced, namely throughput (THP). In past work, THP was not a controlled variable but rather seen as a product of the system while trying to maintain a good PSE (Coetzee *et al.*, 2010; Matthews and Craig, 2013). However, a high THP is also an objective in the control design approach presented here. To account for the additional controlled variable THP, mill speed (α_{speed}) is added as a manipulated variable. It should be noted that in practice a MPC typically outputs the set-points for local PID loops i.e. the set-point for mill speed. This study assumes therefore that a variable speed drive (VSD) is fitted on the mill motor.

2.2.2 Milling circuit model

The aim of this section is to present the *Hulbert*-model (Hulbert, 2005; Le Roux *et al.*, 2013b) in a more standardized state-space representation. This involves writing the change in state ($\dot{x}$) and output (y) equations in the following form:

$$\dot{x} = f(x,u,p) \tag{2.1}$$

$$y = g(x,u,p) \tag{2.2}$$

Table 2.1: Constraints and operating point (OP) for states, manipulated and controlled variables.

Variable	Min	Max	OP (Le Roux *et al.*, 2013b)	OP (Coetzee *et al.*, 2010)	Description
States					
X_{mw}	0	50	4.849	8.280	Volume of water in mill [m^3]
X_{ms}	0	50	4.898	9.510	Volume of solids in mill[m^3]
X_{mf}	0	50	1.087	3.340	Volume of fines in mill [m^3]
X_{mr}	0	50	1.816	20.90	Volume of rocks in mill [m^3]
X_{mb}	0	20	8.513	6.320	Volume of steel balls in mill [m^3]
X_{sw}	0	10	4.108	2.530	Volume of water in sump [m^3]
X_{ss}	0	10	1.876	0.640	Volume of solids in sump [m^3]
X_{sf}	0	10	0.416	0.230	Volume of fines in sump [m^3]
Manipulated variables					
MIW	0	100	4.639	27.17	Flow rate of water to the mill [m^3/h]
MFS	0	200	65.24	88.20	Flow rate of solids to the mill [t/h]
MFB	0	8	5.685	2.000	Flow rate of steel balls to the mill [t/h]
CFF	200	450	373.5	423.1	Flow rate of slurry to the cyclone [m^3/h]
SFW	0	300	140.5	261.6	Flow rate of water to the sump [m^3/h]
α_{speed}	0.4	1	0.712	0.820	Fraction of critical mill speed [%]
Controlled variables					
PSE	0.50	0.85	0.687	0.800	Product particle size [Fraction $< 75\ \mu$m]
$LOAD$	0.20	0.50	0.339	0.450	Total charge in the mill [Fraction of total volume]
$SVOL$	2	20	6.000	3.000	Volume of sump content []
THP	0	74	19.45	32.60	Throughput (solids) [m^3/h]
Important output variables					
P_{mill}	0	2000	1830	2000	Power draw of the mill motor [kW]
φ	0	1	0.510	0.510	Rheology factor
CFD	1	2	1.690	1.340	Cyclone feed density [kg/m^3]

The above equations show that the change in states and outputs are based on the state values (x), inputs (u) and parameters (p). It should be noted that this description does not explicitly show that the outputs are a function of the delayed states and inputs as well i.e. $y = f(x, x(t-T), u, p)$. There

are two time delays present within the milling circuit, between the sump and cyclone and cyclone and mill.

A brief description of the *Hulbert*-model is presented then followed by the state space description for the milling circuit. The ROM ore milling circuit considered in this study consists of four modules; feeder, mill, sump and cyclone.

Table 2.2 provides a description of the subscripts used in the next section for flow-rates (V) and states (X). The first subscript indicates the process unit considered and the second subscript specifies one of the five states. For the flow-rates, the final subscript shows if it is an inflow, outflow or underflow. The parameter values used in this article are shown in Table 2.3.

Table 2.2: Description of subscripts

Subscript	Description
$X_{\Delta-}$	m-mill; s-sump; c-cyclone
$X_{-\Delta}$	w-water; s-solids; c-coarse; f-fines; r-rocks; b-balls
$V_{--\Delta}$	i-inflow; o-outflow; u-underflow

2.2.2.1 Feeder module

The feeder is a straightforward module that consists of MIW, MFS and MFB as inputs. The following are discharges from the feeder module:

$$V_{fwo} = MIW \tag{2.3}$$

$$V_{fso} = \frac{MFS}{D_S}(1-\alpha_r) \tag{2.4}$$

$$V_{ffo} = \frac{MFS}{D_S}\alpha_f \tag{2.5}$$

$$V_{fro} = \frac{MFS}{D_S}\alpha_r \tag{2.6}$$

$$V_{fbo} = \frac{MFB}{D_B} \tag{2.7}$$

These outputs are fed to the mill module.

2.2.2.2 Mill module

The mill module consists of five volume classes. The mill receives water, solids, fines, rocks and steel balls from the feeder module. Note that the mill model makes use of five states to describe the contents of the mill (in m^3): water (X_{mw}), rocks (X_{mr}), solids (X_{ms}), fines (X_{mf}) and steel balls (X_{mb}). The following equations represent a population volume balance. A generalised mass balance equation consists of firstly subtracting the output from the inputs and secondly subtracting or summing any mass broken down or created within the module. Note that the discharge grate prevents the rocks and balls from exiting the mill. The mill module has the potential to model various mill types, such as semi-autogenous grinding (SAG) and ball mills.

$$\frac{d}{dt}X_{mw} = V_{mwi} - V_{mwo} \tag{2.8}$$

$$\frac{d}{dt}X_{ms} = V_{msi} - V_{mso} + RC \tag{2.9}$$

$$\frac{d}{dt}X_{mf} = V_{mfi} - V_{mfo} + FP \tag{2.10}$$

$$\frac{d}{dt}X_{mr} = V_{mri} - RC \tag{2.11}$$

$$\frac{d}{dt}X_{mb} = V_{mbi} - BC \tag{2.12}$$

The breakage functions terms are shown below:

$$RC = \frac{P_{mill} \cdot \varphi}{D_S \phi_r} \left(\frac{X_{mr}}{X_{mr} + X_{ms}} \right) \tag{2.13}$$

$$BC = \frac{P_{mill} \cdot \varphi}{\phi_b} \left(\frac{X_{mb}}{D_S \cdot (X_{mr} + X_{ms}) + D_B \cdot X_{mb}} \right) \tag{2.14}$$

$$FP = \frac{P_{mill}}{D_S \cdot \phi_f \cdot \left[1 + \alpha_{\phi_f} \cdot \left(\frac{X_{mw}+X_{mr}+X_{ms}+X_{mb}}{v_{mill}} - v_{P_{max}} \right) \right]} \tag{2.15}$$

The rheology factor within the mill is represented as:

$$\varphi = \left\{ \max \left[0, 1 - \left(\frac{1}{\varepsilon_{sv}} - 1 \right) \frac{X_{ms}}{X_{mw}} \right] \right\}^{0.5} \tag{2.16}$$

Note that the definition for the rheology factor is specific to this work. In general, the term rheology describes the behaviour of non-Newtonian fluids, with taking into account the applied stresses and strain rates. Fig. 2.3 shows the relationship between the hold-up of solids and water and the effect on the rheology factor. A good or recommended rheology factor for this work is 0.51 (Coetzee *et al.*, 2010). This implies that there should always be more solids than water within the mill.

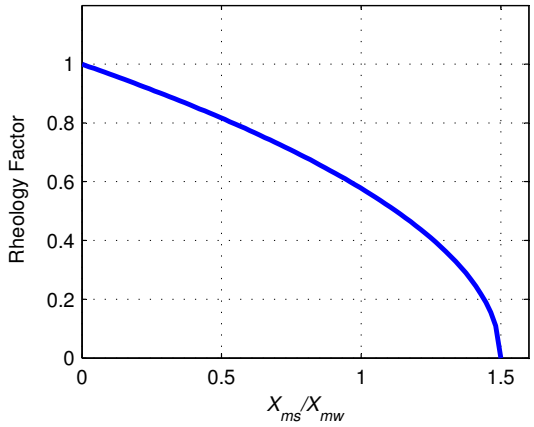

Figure 2.3: Empirically defined rheology factor (φ).

The mill model consists of three outputs because the rocks and steel balls are kept within the mill by a discharge grate. These flow-rates are shown below:

$$V_{mwo} = V_V \cdot \varphi \cdot X_{mw} \cdot \left(\frac{X_{mw}}{X_{ms} + X_{mw}} \right) \tag{2.17}$$

$$V_{mso} = V_V \cdot \varphi \cdot X_{mw} \cdot \left(\frac{X_{ms}}{X_{ms} + X_{mw}} \right) \tag{2.18}$$

$$V_{mfo} = V_V \cdot \varphi \cdot X_{mw} \cdot \left(\frac{X_{mf}}{X_{ms} + X_{mw}} \right) \tag{2.19}$$

$$V_{mro} = V_{mbo} = 0 \tag{2.20}$$

The mill consists of one controlled variable, *LOAD*, and one important output variable, P_{mill}. The effect of the total charge on mill power (Z_x) and the effect of slurry rheology on mill power (Z_r) are given by (2.23) and (2.24) respectively. To minimize mill power draw, Z_x and Z_r need to be as large as possible. There are two parameters that need to be taken into account regarding mill power draw i.e. mill filling for maximum power draw ($v_{P_{max}}$), and the rheology factor for maximum power draw ($\varphi_{P_{max}}$). If the aim of operation is to minimize mill power draw, then the difference between *LOAD* and $v_{mill} \cdot v_{P_{max}}$ should be maximized. Similarly with the difference between φ and $\varphi_{P_{max}}$ while operating within constraints. This means that a highly diluted or very thick slurry within the mill would minimize power consumption. These scenarios should be treated with caution as a diluted slurry could cause a puddle within the mill and a thick slurry could form a mud ball, which could negatively effect the overall quality and throughput of the milling circuit. An extremely high or

extremely low mill load should also be treated with caution.

$$LOAD = X_{mw} + X_{mr} + X_{ms} + X_{mb} \tag{2.21}$$

$$P_{mill} = P_{max}\{1 - \delta_{Pv} Z_x^2 - 2\chi_P \delta_{Pv} \delta_{Ps} Z_x Z_r - \delta_{Ps} Z_r^2\} \cdot (\alpha_{speed})^{\alpha_P} \tag{2.22}$$

$$Z_x = \frac{X_{mw} + X_{mr} + X_{ms} + X_{mb}}{v_{mill} \cdot v_{P_{max}}} - 1 \tag{2.23}$$

$$Z_r = \frac{\varphi}{\varphi_{P_{max}}} - 1 \tag{2.24}$$

The three outputs from the mill are directly fed into a sump.

2.2.2.3 Sump module

Similarly to the mill volume balance equations, the sump consists of three volume classes. The model of the sump uses three states to describe the contents of the sump: water (X_{sw}), fines (X_{sf}) and solids (X_{ss}). The sump consists of two manipulated variables, SFW and CFF. The population volume balance equations are shown below:

$$\frac{d}{dt} X_{sw} = V_{swi} - V_{swo} + SFW \tag{2.25}$$

$$\frac{d}{dt} X_{ss} = V_{ssi} - V_{sso} \tag{2.26}$$

$$\frac{d}{dt} X_{sf} = V_{sfi} - V_{sfo} \tag{2.27}$$

The discharge flow-rates of each volume are shown below:

$$V_{swo} = CFF \cdot \left(\frac{X_{sw}}{X_{sw} + X_{ss}} \right) \tag{2.28}$$

$$V_{sso} = CFF \cdot \left(\frac{X_{ss}}{X_{sw} + X_{ss}} \right) \tag{2.29}$$

$$V_{sfo} = CFF \cdot \left(\frac{X_{sf}}{X_{sw} + X_{ss}} \right) \tag{2.30}$$

The two measurable outputs from the sump are as follows:

$$SVOL = X_{sw} + X_{ss} \tag{2.31}$$

$$CFD = \frac{V_{swo} + D_s V_{sso}}{V_{swo} + V_{sso}} = \frac{X_{sw} + D_s X_{ss}}{X_{sw} + X_{ss}} \tag{2.32}$$

The outputs of the sump are controlled by a sump pump which pumps the slurry from the sump to the cyclone.

2.2.2.4 Cyclone module

The aim of the cyclone is to separate the slurry feed with regards to weight, which usually relates to the particle size. Ideally, the particles that satisfy the minimum specification size, 75 μm used in this work, all exit the cyclone through the overflow. The oversize or "off-spec" material is fed back into the mill exiting the cyclone through the underflow. In practice, the aim is that as much as possible fines pass to the overflow of the cyclone.

The flow-rates at the underflow of the classifier is defined as:

$$\frac{V_{ccu}}{V_{cci}} = \left(1 - C_1 \exp\left(\frac{-CFF}{\varepsilon_c}\right)\right)\left(1 - \left(\frac{F_i}{C_2}\right)^{C_3}\right)\left(1 - P_i^{C_4}\right) \tag{2.33}$$

$$V_{cwu} = \frac{V_{cwi}\left(V_{ccu} - F_u V_{ccu}\right)}{\left(F_u V_{cwi} + F_u V_{cfi} - V_{cfi}\right)} \tag{2.34}$$

$$V_{cfu} = \frac{V_{cfi}\left(V_{ccu} - F_u V_{ccu}\right)}{\left(F_u V_{cwi} + F_u V_{cfi} - V_{cfi}\right)} \tag{2.35}$$

The fraction solids in the total inflow volume is defined as:

$$F_i = \frac{V_{csi}}{CFF} \tag{2.36}$$

The fraction fines in the feed solids is defined as:

$$P_i = \frac{V_{cfi}}{V_{csi}} \tag{2.37}$$

The fraction solids in the underflow volume is defined as:

$$F_u = 0.6 - (0.6 - F_i) \cdot \exp\left(-V_{ccu}/(\alpha_{su}\varepsilon_c)\right) = \frac{V_{cfu} + V_{ccu}}{V_{cwu} + V_{cfu} + V_{ccu}} \tag{2.38}$$

The flow-rates at the overflow of the classifier is defined as:

$$V_{cwo} = V_{cwi} - V_{cwu} \tag{2.39}$$

$$V_{cso} = V_{csi} - V_{csu} \tag{2.40}$$

$$V_{cfo} = V_{cfi} - V_{cfu} \tag{2.41}$$

The cyclone module consists of two controlled variables, PSE and THP shown in (2.42) and (2.43) respectively.

$$PSE = \frac{V_{cfo}}{V_{cso}} \tag{2.42}$$

$$THP = V_{cso} \tag{2.43}$$

2.2.3 State space description

The circuit model for the robust nonlinear MPC implemented in simulation by Coetzee *et al.* (2010), was validated with real plant data by Le Roux *et al.* (2013b). The values for the mill and sump states at the operating point of the mill are shown in Table 2.1. The full state-space description of the milling circuit is shown in (2.44)-(2.57) and demonstrates the complexity and nonlinearity of the circuit. Five population balance differential equations describe the change in the states of the mill, and three population balance differential equations describe the change in the states of the sump. It is assumed that there is a transport delay of 30 s and 10 s between the sump and the cyclone (τ_{sc}) and between the cyclone and the mill (τ_{cm}) respectively.

2.2.3.1 Mill state equations

The mill contains five state equations shown in eq. (2.44)-(2.48).

$$\frac{d}{dt}X_{mw} = MIW - \frac{V_V \varphi X_{mw} X_{mw}}{X_{ms} + X_{mw}} + V_{cwu} \tag{2.44}$$

$$\frac{d}{dt}X_{ms} = \frac{MFS}{D_S}(1-\alpha_r) - \frac{V_V \varphi X_{mw} X_{ms}}{X_{ms} + X_{mw}} + \frac{P_{mill}\varphi}{D_S \phi_r}\left(\frac{X_{mr}}{X_{mr} + X_{ms}}\right) + V_{csu} \tag{2.45}$$

$$\frac{d}{dt}X_{mf} = \frac{MFS}{D_S}\alpha_f - \frac{V_V \varphi X_{mw} X_{mf}}{X_{ms} + X_{mw}} + V_{cfu} + \frac{P_{mill}}{D_S \phi_f \left[1 + \alpha_{\phi_f}\left(\frac{X_{mw}+X_{mr}+X_{ms}+X_{mb}}{v_{mill}} - v_{P_{max}}\right)\right]} \tag{2.46}$$

$$\frac{d}{dt}X_{mr} = \frac{MFS}{D_S}\alpha_r - \frac{P_{mill}\varphi}{D_S \phi_r}\left(\frac{X_{mr}}{X_{mr} + X_{ms}}\right) \tag{2.47}$$

$$\frac{d}{dt}X_{mb} = \frac{MFB}{D_B} - \frac{P_{mill}\varphi}{\phi_b}\left(\frac{X_{mb}}{D_S(X_{mr} + X_{ms}) + D_B X_{mb}}\right) \tag{2.48}$$

2.2.3.2 Sump state equations

The sump contains three state equations shown in eq. (2.49)-(2.51).

$$\frac{d}{dt}X_{sw} = \frac{V_V \varphi X_{mw} X_{mw}}{X_{ms} + X_{mw}} - \frac{CFF X_{sw}}{X_{sw} + X_{ss}} + SFW \tag{2.49}$$

$$\frac{d}{dt}X_{ss} = \frac{V_V \varphi X_{mw} X_{ms}}{X_{ms} + X_{mw}} - \frac{CFF X_{ss}}{X_{sw} + X_{ss}} \tag{2.50}$$

$$\frac{d}{dt}X_{sf} = \frac{V_V \varphi X_{mw} X_{mf}}{X_{ms} + X_{mw}} - \frac{CFF X_{sf}}{X_{sw} + X_{ss}} \tag{2.51}$$

2.2.3.3 Output equations

The circuit contains six outputs shown in (2.52)-(2.57).

$$P_{mill} = P_{max} \cdot (1 - \delta_{Pv} Z_x^2 - 2\chi_P \delta_{Pv} \delta_{Ps} Z_x Z_r - \delta_{Ps} Z_r^2)(\alpha_{speed})^{\alpha_P} \tag{2.52}$$

$$LOAD = X_{mw} + X_{ms} + X_{mr} + X_{mb} \tag{2.53}$$

$$SVOL = X_{ss} + X_{sw} \tag{2.54}$$

$$CFD = \frac{X_{sw} + X_{ss} D_S}{X_{sw} + X_{ss}} \tag{2.55}$$

$$PSE = \frac{V_{cfo}}{V_{cso}} \tag{2.56}$$

$$THP = V_{cso} \tag{2.57}$$

2.2.3.4 Intermediate equations

Equations (2.58)-(2.67) are intermediate equations for variables used in the state and output equations to complete the model description. The effect of the total charge on mill power (Z_x) and the effect of the slurry rheology (φ) on mill power (Z_r) are given by the empirically defined equations:

$$\varphi = max\left(0, \left(1 - \left(\frac{1}{\varepsilon_{sv}} - 1\right)\frac{X_{ms}}{X_{mw}}\right)^{0.5}\right) \tag{2.58}$$

$$Z_x = \frac{X_{mw} + X_{mr} + X_{ms} + X_{mb}}{v_{mill} v_{P_{max}}} - 1 \tag{2.59}$$

$$Z_r = \frac{\varphi}{\varphi_{P_{max}}} - 1 \tag{2.60}$$

The flow rate at the underflow of the cyclone is regarded as a mill input. The flow-rates of coarse (V_{ccu}), fines (V_{cfu}), and solids (V_{csu}) material as well as water (V_{cwu}) at the underflow are described as:

$$V_{ccu} = \frac{CFF\,(X_{ss} - X_{sf})}{X_{sw} + X_{ss}} \left(1 - C_1 \exp\left(\frac{-CFF}{\varepsilon_c}\right)\right) \left(1 - \left(\frac{X_{ss}}{C_2\,(X_{sw} + X_{ss})}\right)^{C_3}\right) \cdot \left(1 - \left(\frac{X_{sf}}{X_{ss}}\right)^{C_4}\right) \tag{2.61}$$

$$F_u = 0.6 - \left(0.6 - \frac{X_{ss}}{X_{sw} + X_{ss}}\right) \cdot \exp\left(-V_{ccu}/(\alpha_{su}\varepsilon_c)\right) \tag{2.62}$$

$$V_{cwu} = \frac{X_{sw}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{2.63}$$

$$V_{cfu} = \frac{X_{sf}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{2.64}$$

$$V_{csu} = V_{ccu} + V_{cfu} \tag{2.65}$$

The cyclone output flow-rates of the solids (V_{cso}) and fines (V_{cfo}) are:

$$V_{cso} = \frac{CFFX_{ss}}{X_{ss} + X_{sw}} - V_{ccu} - \frac{X_{sf}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{2.66}$$

$$V_{cfo} = \frac{CFFX_{sf}}{X_{ss} + X_{sw}} - \frac{X_{sf}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{2.67}$$

The parameter values used in this article are shown in Table 2.3 and were taken from Coetzee *et al.* (2010) (used in Chapter 3) and Le Roux *et al.* (2013b) (used in Chapter 4).

2.3 HISTORY OF CONTROL OF GRINDING MILL CIRCUITS

A recent literature survey on the control of grinding mill circuits is illustrated in Fig. 2.4 (Craig, 2012). The literature survey aims to illustrate the various control techniques that have been applied to grinding mill circuits over the years. The survey shows the new developments in control techniques and the shortcomings.

Traditionally ROM ore milling circuits are controlled by classical single-loop PID controllers (Wei and Craig, 2009) despite the multivariable nature of such circuits. Significant improvements in product quality, throughput and power consumption are possible when using multivariable control techniques. Numerous examples are available in literature, e.g. H_∞ and μ synthesis based controller designs (Craig and MacLeod, 1995; Craig and MacLeod, 1996), and linear model predictive control (Chen, Zhai, Li and Li, 2007b).

A multivariable control scheme was developed by Hulbert, Craig, Coetzee and Tudor (1990) and implemented on a ROM ore milling circuit. The study concluded that the milling circuit can be optimized by selecting suitable set-points. A year later, optimal control of a ball mill grinding circuit was completed by Rajamani and Herbst (1991) and a comparison between optimal control and PI control was investigated. The results of the study showed that an optimal controller can be implemented. However, a shortfall was that an on-line systems identification tool is required. Following this shortfall, Valenzuela, Najim, del Villar and Bourassa (1993) investigated a new process control

Table 2.3: Parameters and parameter values.

Parameter	Value (Le Roux *et al.*, 2013b)	Value (Coetzee *et al.*, 2010)	Description
Mill parameters			
α_f	0.055	0.1	Fraction of fines in the ore
α_r	0.465	0.1	Fraction of rocks in the ore
ϕ_f	29.57	28	Power needed per ton of fines produced [(kWh)/t]
ϕ_r	6.03	69	Rock abrasion factor [(kWh)/t]
ϕ_b	90	94	Steel abrasion factor [(kWh)/t]
ε_{sv}	0.6	0.6	Maximum fraction of solids by volume of slurry at zero slurry flow
δ_{Pv}	0.5	1	Power-change parameter for volume of mill filled
v_{pmax}	0.34	0.45	Fraction of mill volume filled for maximum power draw
v_{mill}	59.1	100	Mill volume [m^3]
χ_P	0	0	Cross-term for maximum power draw
δ_{Ps}	0.5	1	Power-change parameter for fraction solids in the mill
φ_{Pmax}	0.57	0.51	Rheology factor for maximum mill power draw
α_P	1	0.82	Fractional power reduction per fractional reduction from maximum mill speed
V_V	84	40	Volumetric flow per "flowing volume" driving force [h^{-1}]
α_{φ_f}	0.01	0.01	Fractional change in kW/fines produced per change in fractional filling of mill
Hydrocyclone parameters			
α_{su}	0.87	0.16	Parameter related to fraction solids in underflow
ε_s	128.85	184	Parameter related to coarse split [m^3/h]
C_1	0.6	0.6	Constant
C_2	0.7	0.7	Constant
C_3	4	3	Constant
C_4	4	3	Constant

technique that targets the issue of on-line systems identification and developed a learning controller that showed good robustness while requiring minimal computational load.

Computational load has to be considered when developing a controller. If the controller cannot com-

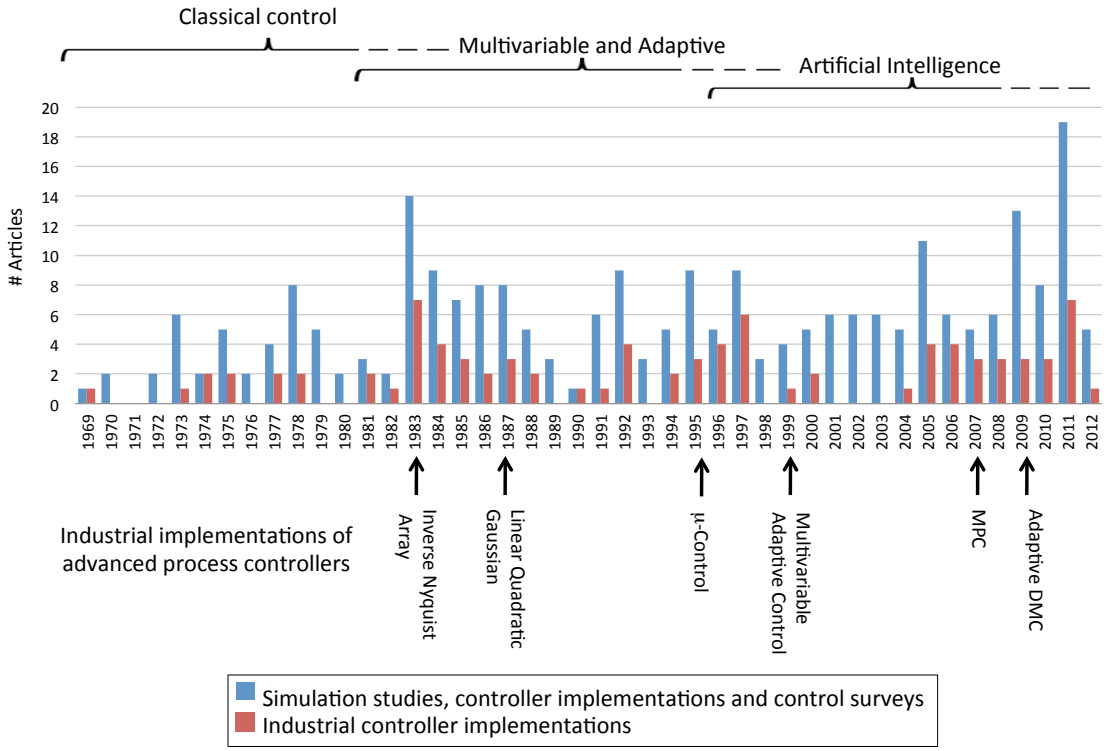

Figure 2.4: Literature survey on the history of grinding mill control.

pute a new control move within a sample time then the controller may be regarded as impractical. The work by presented in Coetzee *et al.* (2010) is an example of a robust nonlinear controller taking an average time of approximately 26 s per iteration and a maximum time of 123 s per iteration. The authors regarded this as not feasible for practical implementation as this is significantly higher than the recommended sampling time. Computation power has grown significantly over the past decade and techniques such as a direct neural network (NN) controller can make use of this development. A direct NN controller was developed by Duarte, Suarez and Bassi (2001) and compared to classical and adaptive multivariable control algorithms. Three neural networks were used to form a control solution, performing the roles of a state estimator, predictor and controller.

Multiple-input and multiple-output (MIMO) techniques were developed because of the interactions between input and output variables. These controllers, such as μ-synthesis controller design (Craig and MacLeod, 1995) and model predictive control (MPC) (Ramasamy, Narayanan and Rao, 2005) are able to perform well compared to decoupled proportional-integral-derivative (PID) control.

Fig. 2.4 shows the approximate years where advanced process controllers have been implemented in industry. The survey also shows that some multivariable control methods have fairly recently been implemented in an industry where single-loop PID controllers dominate (Wei and Craig, 2009). In particular, an MPC implementation on a grinding circuit was first reported in the literature as late as 2007 (Chen *et al.*, 2007b). Compare this to the ubiquitous nature of MPC in for example the petrochemical industry (Craig, Aldrich, Braatz, Cuzzola, Domlan, Engell, Hahn, Havlena, Horch, Huang, Khanbaghi, Konstantellos, Marquardt, McAvoy, Parisini, Pistikopoulos, Samad, Skogestad, Thornhill and Yu, 2011), there is great potential for increasing the number of MPC implementations on mineral processing plants. Fig. 2.5, reproduced from Wei and Craig (2009), shows that over 60% of milling circuits are controlled with PID controllers while less than 10% are controlled using model predictive control.

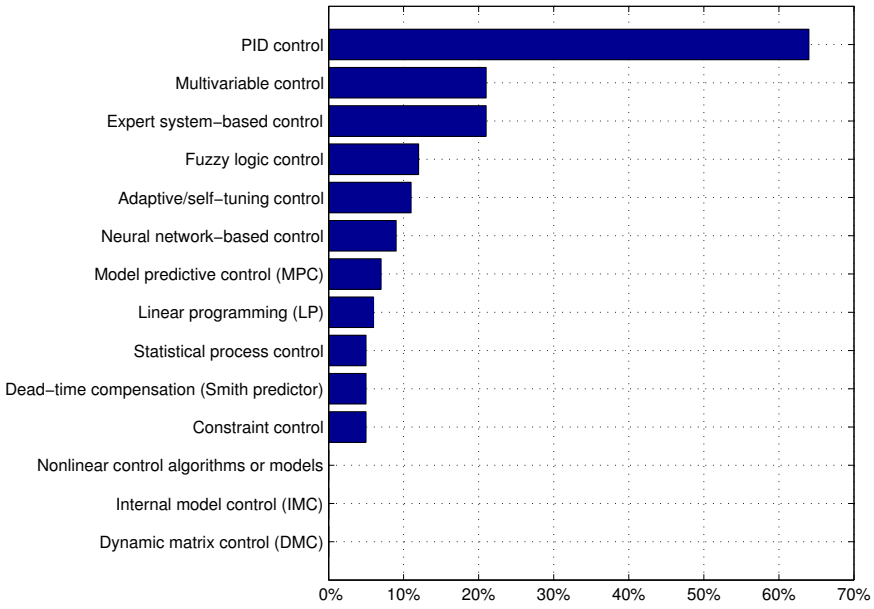

Figure 2.5: Control technologies used in milling circuits.

A robust nonlinear model predictive control (RNMPC), presented in Coetzee *et al.* (2010), for a ROM ore milling circuit has been successfully simulated. Despite disturbances to the inputs, often a case in the minerals processing industry where there is a variation in the ore size distribution, the results show that a NMPC is worth investigating further. The ROM ore milling circuit was analysed further by Olivier (2011) where a host of peripheral tools were developed (Olivier *et al.*, 2012a; Olivier *et al.*, 2012b; Olivier and Craig, 2013). The tools were developed with the operating philosophy

described in detail in Olivier (2011). The ROM ore milling circuit has a desired operating region that will aim to maximize the final product after leaching while minimizing production costs. This is achieved by choosing good set-point values for the particle size estimate (PSE), throughput (THP) and load within the mill ($LOAD$).

Model predictive control is highly dependent on the accuracy of the model within the controller. The model for the ROM ore milling circuit was verified by Le Roux *et al.* (2013b) with real plant data to support the model accuracy. Further research by Matthews and Craig (2013) makes use of a linear model predictive controller, verified plant model and an additional real time power optimizer. Power consumption is a trending topic and all industries are investigating power saving opportunities. The work by Matthews and Craig (2013) show that the cost per tons of fines produced can be reduced, provided the circuit is not operating at maximum throughput.

State estimation is another factor that model predictive control is highly dependant on. A run-of-mine (ROM) ore milling circuit is a process that is difficult to control because of significant model uncertainties, large unmeasured disturbances and process variables that are difficult to measure (Olivier *et al.*, 2012b). This justifies estimation methods for process measurements.

Currently, the extended Kalman filter (EKF) is the most common choice for soft sensing/estimation in minerals processing plants (Bouche *et al.*, 2005). Inherently, if the initial guess for the state estimate is incorrect or if the process model is inaccurate, because of for example linearization in an operating region. The EKF can be seen as providing a first-order approximation and these approximations can introduce large errors in the true posterior mean and covariance, and in some cases the filter results may diverge quickly. Additional flaws of the EKF are discussed by Van der Merwe (2004). Neural networks have previously been used to aid in the control of mineral grinding circuits. In a recent publication, a radial basis function neural network was used to successfully predict the in-mill slurry density and ball load volume in a ball-milling system (Makokha and Moys, 2012). Key process variables have hitherto been difficult or expensive to measure. Provided an offset error correction scheme is present, a neural network can be used as a direct neural controller and an inverse network controller (Flament, Thibault and Hodouin, 1993). An inverse network controller refers to a neural model based on inverting process dynamics allowing a process input to be computed from a desired process output. Stange (1993) built on this work and states that neural networks can be used as adaptive predictors. However, the issue with neural networks is that an extensive sampling campaign may be required to generate training data and the cost of the training campaign may not be warranted.

The benefit of a particle filter is that the particle filter does not require any training data and can deal with nonlinear non-Gaussian systems (Olivier *et al.*, 2012b). The particle filter does not suffer from the flaws of the EKF, e.g. even severely non-linear models may be used. Using inferential based measurements is an alternative approach. Herbst and Pate (1999) use this approach to estimate important process variables such as product particle size, ore grindability and mill filling. Additional process variables such as total mill load and ball load values were estimated by (Apelt, Asprey and Thornhill, 2001; Apelt, Asprey and Thornhill, 2002) using inferential based methods.

From the literature survey, a significant amount of work has been completed on the ROM ore milling circuit. A complete nonlinear model based controller integrated with full state estimation has yet to be completed. The work presented aims to fill this gap in the literature. In this study, a nonlinear controller combined with state estimation has been implemented on a ROM ore milling circuit. State estimation for the mill module has also been investigated and a combined neural network and particle filter approach is proposed. Additionally, power optimization has also been researched further.

Choosing the correct manipulated and controlled variables will have an impact on how applicable this simulation study is to industry. Wei and Craig (Wei and Craig, 2009) performed a worldwide survey in 2008 on grinding mill circuits in the minerals industry. The measured variables in over 80% of milling circuits are as follows: mill power, feed rate of water to the mill, feed density of the sump discharge slurry and sump level. The model shown in Fig. 2.1 requires the particle size to also be measured; this was measured in approximately 70% of milling circuits. The top three choices for controlled variables were product particle size, slurry level in the sump and sump discharge slurry density. In Craig and MacLeod (1996), Olivier *et al.* (2012a), Olivier *et al.* (2012b) and Coetzee *et al.* (2010), the mill load and sump level were controlled to stabilize the circuit and the particle size controlled to decrease variations in the product size. The top three choices for the manipulated variables were: the flow rate of water to the sump, flow rate of water to the mill and feed rate of solids to the mill. This conforms with the variables used in this study.

2.4 NON-LINEAR MODEL PREDICTIVE CONTROL

An advantage of non-linear model predictive control (NMPC) is that it allows the use of a non-linear plant model to be used for prediction. The basic control loop structure for NMPC is shown in Fig. 2.6. NMPC can be designed for a specific cost function to be minimised while keeping the state and control within specified constraints. NMPC aims to optimize future behaviour of the plant by testing

a set of manipulated variable actions and choosing a set that minimises the cost function (Qin and Badgwell, 2003).

Currently, the a main limitation with industrial MPC applications is the larger computational power required and alternative structures that reduce the computational burden should be investigated (Qin and Badgwell, 2003; Xi, Li and Lin, 2013).

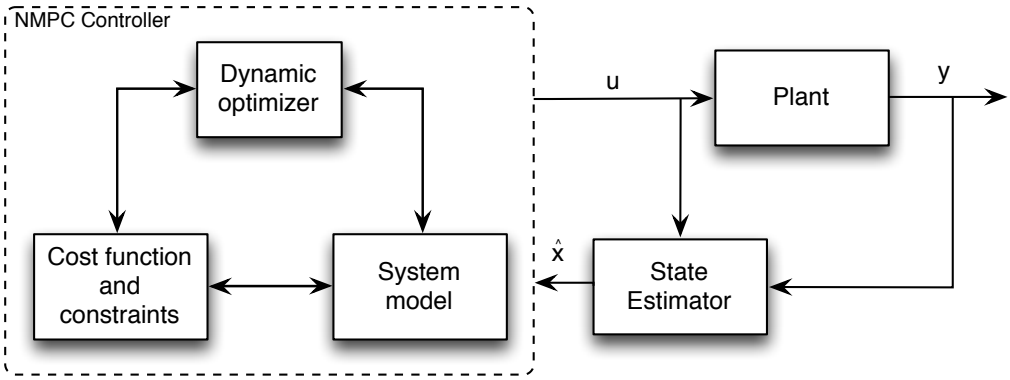

Figure 2.6: Generalized non-linear MPC control loop.

2.5 CONCLUSION

In this chapter, a nonlinear model describing the ROM ore milling circuit has been described and presented in state space format. This plant model will be used in the next three chapters to develop a neural network for state estimation, a nonlinear model predictive controller and a power optimizer. A brief look at the history of control of a grinding mill circuit showed that PID control is predominate in milling circuits and that there is reason to investigate the use of nonlinear model predictive control. A variety of estimation based methods have been implemented in the minerals processing industry. However, there is no single dominate estimation technique that is an ideal solution for this nonlinear estimation problem. This warrants the investigation of a neural network as a possible solution for estimating mill states. The cost of energy has increased significantly in recent years. This increase in price greatly affects the mineral processing industry because of the large energy demands. A run-of-mine ore milling circuit provides a suitable case study where the power consumed by a mill is in the order of 2 MW.

CHAPTER 3

COMBINED NEURAL NETWORK AND PARTICLE FILTER STATE ESTIMATION

3.1 INTRODUCTION

A run-of-mine (ROM) ore milling circuit is a process that is difficult to control because of significant model uncertainties, large unmeasured disturbances and process variables that are difficult to measure (Olivier *et al.*, 2012b). Process and state variables that are impractical to measure on-line motivate the investigation for state estimation. When designing a controller that requires full state feedback, the internal states of the mill must be accurately measured or estimated to achieve good control of this complex system. For example, model predictive control (MPC) requires full state feedback, which is difficult to achieve, in this case because of model inaccuracies, parameter variations and lack of on-line measurements. This work builds on the work of Olivier *et al.* (2012b) by investigating the use of a neural network to estimate the states of the mill. The neural network approach is compared to the particle filter technique implemented by Olivier *et al.* (2012b). Any improvements on state estimation will result in a more accurate closed-loop control. The internal states of a grinding mill model (Le Roux *et al.*, 2013b) are estimated using a neural network, and results are compared to those achieved when using a particle filter. A new method of state estimation using both a neural network and particle filter is also presented.

The application of robust non-linear MPC to a ROM ore milling circuit was presented by Coetzee *et al.* (2010). The controller described by Coetzee *et al.* (2010) requires full state feedback, an issue that is partially addressed by Coetzee (2009) and by Olivier *et al.* (2012b). The state estimation technique presented in this chapter could help improve the overall control accuracy of MPC and increase the

number of implementations in the mineral processing environment.

3.2 NEURAL NETWORKS

Neural networks have the ability to form predictive relationships between complex (such as non-linear) inputs and outputs. They also have the ability to provide fast inferred output process values that are difficult to measure (Willis, Montague, Di Massimo, Tham and Morris, 1992). This section gives a description of neural networks, similar to that given by Stange (1993).

A network consists of three major layers, namely the input layer, the hidden layer and the output layer. These three layers are shown in Fig. 3.1. The hidden layer shown in the figure is not limited to just one layer. Each node, shown as a circle in Fig. 3.1, operates by summing all the inputs and outputting a transformed value. The transformation comes from applying a weighting factor to the summed inputs. Training needs to be completed first to determine the weighting factors in the nodes. A training set of data is presented to the network and various training algorithms (such as back propagation) can be used. This can be optimised by minimising an objective function that sums the squared error between the predicted and actual output. This implies that with a significant amount of training data a complex function or plant can be approximated without the need for complex mathematics. The network requires three major settings, namely the number of layers, the number of nodes within each layer and the type of transformation function used in each node. Applying a neural network becomes a trial and error process as the network is highly dependant on the input-output data that it is presented with.

Neural networks are robust with respect to noisy plant data. According to Stange (1993), neural networks can be trained to replace existing controllers, as shown by Conradie and Aldrich (2001) where a neural network was used to control a ball mill grinding circuit. A symbolic adaptive neuro-evolution algorithm was developed by Conradie and Aldrich (2001) to eliminate the controller interactions and therefore making the controller more robust. This is feasible when controllers require practically difficult measurements to be made. Neural networks can be trained as adaptive predictors which means future values of measurements can be estimated provided significant amount of past data is available. Neural networks can be used for system identification if trained using input and output data. Additionally, neural networks can be trained to identify the inverse of the plant. This is achieved by using the outputs at the input layer and output layer contains the systems inputs. This enables inputs to be determined by specifying output values.

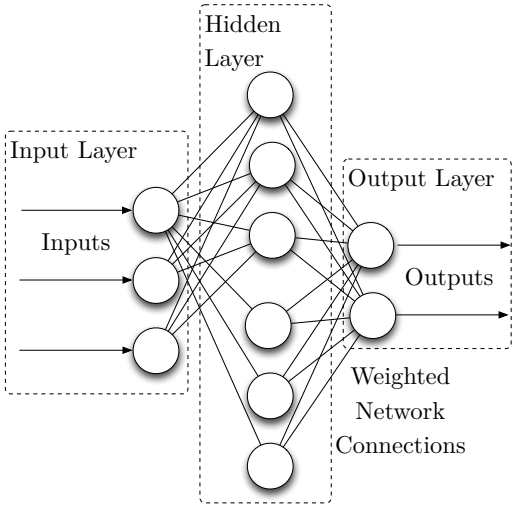

Figure 3.1: Generalized artificial neural network.

3.3 STATE ESTIMATION

3.3.1 Simulation setup

The aim of the state estimators are to estimate the five states that describe the contents of the mill (in m^3): water (X_{mw}), rocks (X_{mr}), solids (X_{ms}), fines (X_{mf}) and steel balls (X_{mb}). In order to illustrate the accuracy of the estimation algorithms, a simulation run is performed while the milling circuit is kept under feedback control by PI controllers for a 20-hour period. The same PI controllers and simulation environment were used by Olivier *et al.* (2012b) who provide further details about the controllers implemented. Disturbances are introduced as follows: the value of ϕ_f is decreased by 10% at time 3 h, the value of α_r is decreased by 10% at time 9 h and the value of α_f is increased by 10% at time 15 h. The three parameters α_f , α_r and ϕ_f are chosen as they have the largest influence on the operation of the mill (Olivier, 2011). The “true" simulated states are estimated and therefore no measurement noise is taken into account. Also, this comparison aims to show the estimation accuracy and not the noise-handling capability. A 10-second sampling time was used in the simulation.

3.3.2 Particle filter

3.3.2.1 Description

Particle filtering is the technique of implementing a recursive Bayesian filter by Monte Carlo simulations. The setup of the particle filtering simulation run is the same as that used by Olivier *et al.* (2012b), in which more information on the particle filter is presented. A brief description is given in this section.

Particle filtering relies on the technique of representing the posterior density function (pdf), which is used for estimation, by a set of random samples and associated weights. The locations of the particles represent the locations at which the pdf is evaluated and the sizes of the particles represent the associated weights, giving an indication of the value of the pdf at this location. This representation is expandable to an arbitrary number of dimensions and is applicable to any distribution, even multi-modal and other non-Gaussian distributions. As the number of particles becomes very large, this method of representing the pdf becomes equivalent to the functional description of the posterior pdf. The pdf at time t may then be approximated as (Arulampalam, Maskell, Gordon and Clapp, 2002):

$$p(x_t|Y_t) \approx \sum_{i=1}^{N_s} w_t^i \delta(x_t - x_t^i) \tag{3.1}$$

where N_s is the number of particles and $\{x_t^i, w_t^i\}_{i=1}^{N_s}$ is the set of particles and associated weights. These weights are defined to be (Ristic, Arulampalam and Gordon, 2004):

$$w_t^i \propto w_{t-1}^i \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, y_t)} \tag{3.2}$$

where $q(x_t^i|x_{t-1}^i, y_t)$ is a proposal distribution called an importance density. Ideally the importance density should be the true posterior distribution $p(x_t|Y_t)$ (Ristic *et al.*, 2004), but as this is not known in general, a proposal distribution is used.

The optimal importance density function, that is presented in (Olivier *et al.*, 2012b), minimizes the variance of importance weights conditioned upon x_{t-1}^i and y_t has been shown to be $q(x_t|x_{t-1}^i, y_t)_{opt} = p(x_t|x_{t-1}^i, y_t)$ resulting in

$$q(x_t|x_{t-1}^i, y_t)_{opt} = \frac{p(y_t|x_t, x_{t-1}^i)p(x_t|x_{t-1}^i)}{p(y_t|x_{t-1}^i)} \tag{3.3}$$

This optimal importance density is however only usable in a specific class of problems where it is

possible to sample from $p(x_t|x_{t-1}^i, y_t)$ and where

$$p(y_t|x_{t-1}^i) = \int p(y_t|x_t)p(x_t|x_{t-1}^i)dx_t \tag{3.4}$$

can be calculated up to a normalizing constant, which is not the case in general. One popular suboptimal choice is the transitional prior

$$q(x_t|x_{t-1}^i, y_t) = p(x_t|x_{t-1}^i) \tag{3.5}$$

which, if it is furthermore assumed that the process noise is additive zero-mean Gaussian noise, simply becomes

$$p(x_t|x_{t-1}^i) = \mathcal{N}(x_t; f_{t-1}(x_{t-1}^i), \mathbf{Q}_{t-1}) \tag{3.6}$$

which can easily be calculated. This assumption also means that particles can be drawn from a Gaussian distribution with a mean equal to the previous particle location propagation through the system equation and standard deviation equal to the noise standard deviation as $x_t^i \sim \mathcal{N}(f_{t-1}(x_{t-1}^i), \mathbf{Q}_{t-1})$.

3.3.2.2 State estimation

The particle filter has the following inputs (flow rates going in the mill): V_{mwi}, V_{msi}, V_{mfi}, V_{mri} and V_{mbi}. The following outputs are required: V_{mwo}, V_{mso}, V_{mfo}, $LOAD$ and P_{mill}. The particle filter is specified with 50 particles. A larger number of particles have been tried without too much improvement in the estimation results. In this study (as in Olivier *et al.* (2012b)) the transitional prior is used. The initial estimates of the mill states are randomly selected from a region ($\pm$ 0.01) around the actual initial values in each case. The “true" states and particle filter estimates are shown in Fig. 3.2.

3.3.3 Neural network

3.3.3.1 Training

The neural network requires the following input data: V_{mwi}, V_{msi}, V_{mfi}, V_{mri} and V_{mbi}. The neural network was trained using 480 hours of simulation data. This input data used to estimate the five states of the mill. Over-training the neural network is a factor that was considered when training the network, as an over-trained network will likely only work for the specified case it was trained on. A positive and negative 20% step change was made to ϕ_f, α_r and α_f. The data consisted of 20% positive

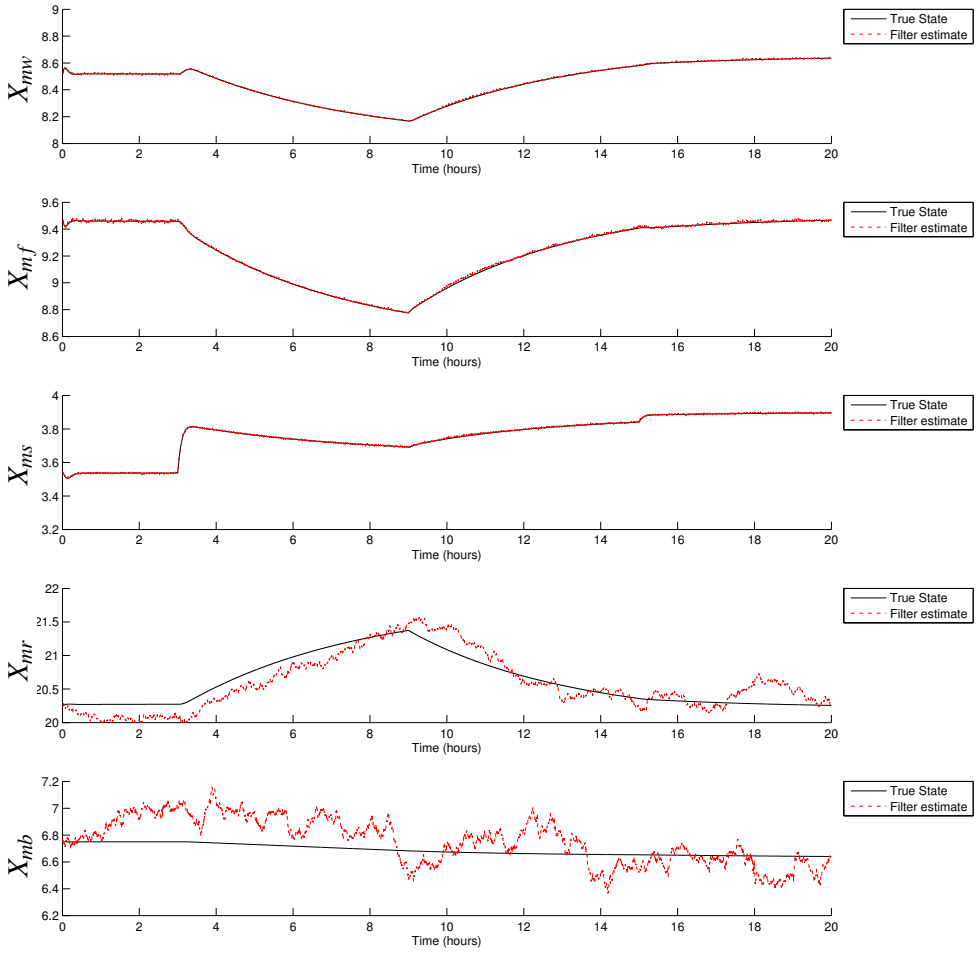

Figure 3.2: Particle filter state estimates

and negative step changes with various combinations of α_r, α_f and ϕ_f at 40 minutes, 480 minutes and 700 minutes respectively, as shown in Fig. 3.3 and Fig. 3.4. Disturbance steps in both directions for all three variables are shown in Fig. 3.5.

A two-layer feedforward backpropagation network with sigmoid hidden neurons and linear output neurons was trained for state estimation. The input and output data sets were randomized (maintaining input-output combination) as randomized combinations of the outputs according to the inputs is preferred compared to an ordered combination. The neural network was trained according to the Levenberg-Marquardt (Hagan, Demuth and Beale, 1996) optimization algorithm. This algorithm determines the weights and the bias values for the network. The network consisted of five hidden neurons.

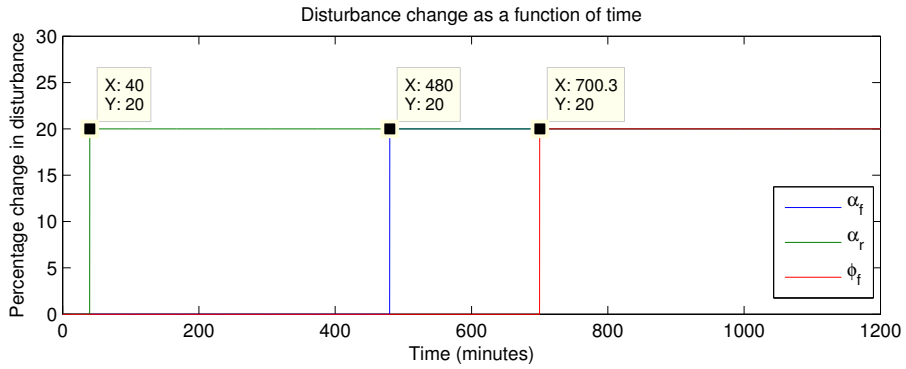

Figure 3.3: Training data set 1 with positive 20% disturbance changes

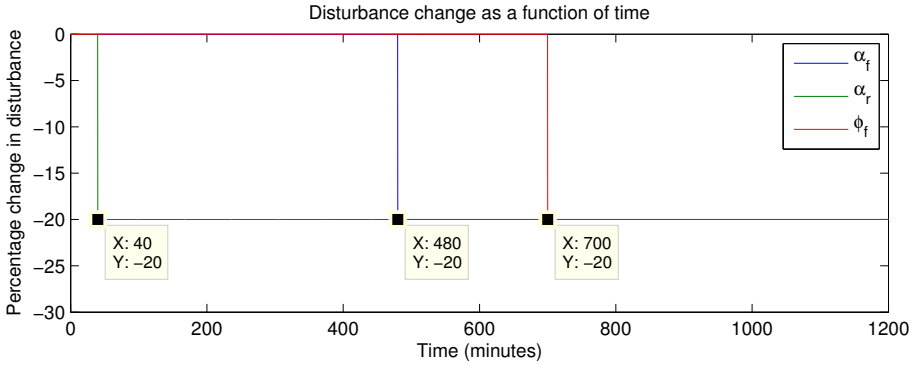

Figure 3.4: Training data set 2 with negative 20% disturbance changes

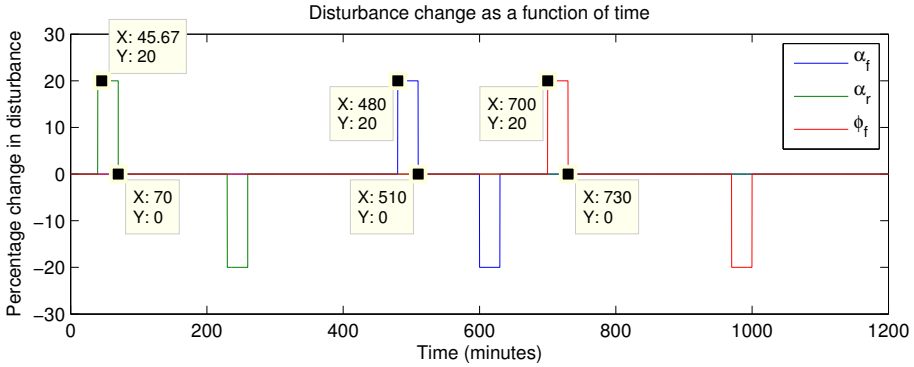

Figure 3.5: Training data set 3 with positive and negative disturbance changes

3.3.3.2 Results

The neural network was then tested on 10% changes using the same simulation environment as for the particle filter defined in section 3.3.1. The aim is to determine if the neural network can estimate the five mill states despite smaller changes made in the parameters, as compared to the training set.

The state estimation results for the neural network are shown in Fig. 3.6.

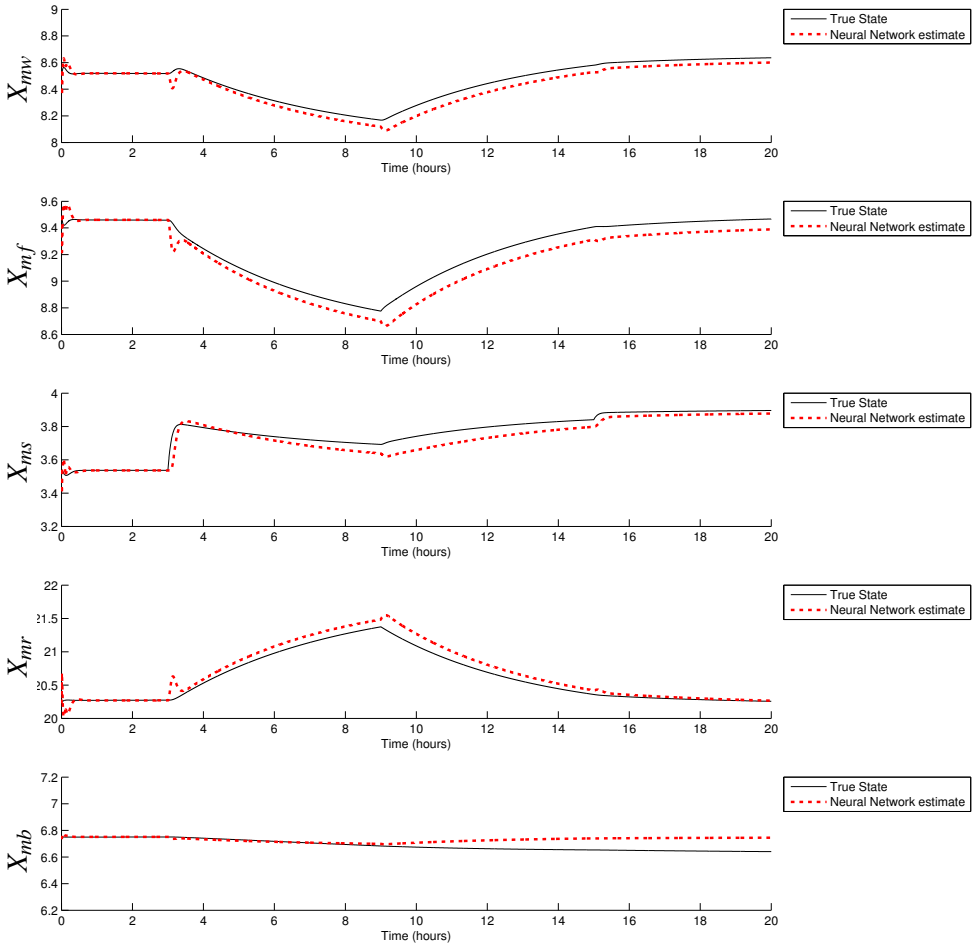

Figure 3.6: State estimation from neural network

The results show that the neural network performs well for the first four states and deviates after 10 h for X_{mb}, as shown in Fig. 3.6. The particle filter also does not estimate the hold-up of steel balls (X_{mb}) accurately, as shown in Fig. 3.2. A comparison between Fig. 3.2 and Fig. 3.6 indicates that the neural network performed better than the particle filter for states X_{mr} and X_{mb}; however, the particle filter's estimates for the first three states were more accurate.

When comparing the neural network and the particle filter estimators, it should be kept in mind that the neural network estimator estimated the mill states based on only the mill input data. The particle filter, however, requires output measurements and an initial estimate of the mill parameters (presented in Olivier *et al.* (2012b)) to function. These were provided to the particle filter estimator as described in section 3.3.2. The neural network estimator therefore had less information at its disposal when

estimating the mill states. Additional simulations showed that if the mill outputs, *LOAD* and P_{mill}, are included in the training data for the neural network then the accuracy is improved. Additionally, using accurate previous state values ($\mathbf{X}(t-1)$) to estimate the current state ($\mathbf{X}(t)$) do improve estimation accuracy. If the previous state value is not accurate and is provided as an input to the neural network then the output state estimate is poor.

3.3.4 Combination of particle filter and neural network

An alternative method of combining the neural network and particle filter is described in this section. Fig. 3.2 shows that the particle filter estimate is more accurate for the first three states than the neural network estimate (Fig. 3.6); however, the neural network estimate was more accurate for X_{mr} and X_{mb}. This section investigates the use of the particle filter to correct the offset found in the neural network estimates for the first three states, similar to offset correction described by Flament *et al.* (1993). For the first three states only, the average of the particle filter estimate at every hour is compared to the average of the neural network estimate. The average of the particle filter estimate is then subtracted from the neural network estimates. The results of the neural network method with particle filter offset correction at every hour are shown in Fig. 3.7. The performance index as a function of time is shown in Fig. 3.8. An average performance index was determined from ten simulation runs because particle filtering is a Monte Carlo method and will therefore have a difference index for each iteration.

The algorithm, which shows how the particle filter method was used as an offset correction method for the neural network method, is presented below. M is the time at which the offset error correction will be implemented. N is the number of values used to determine the average value of each method. At every hour the difference between the neural network estimation and particle filter estimation was calculated based on N samples of historical data. The offset correction was then implemented to the neural network estimation for M hours. The value of M was chosen based on an estimate on when lab samples would be available to infer the component outputs of the mill (V_{mwo}, V_{mso} and V_{mfo}). The particle filter approach can then be used with this output information. An estimate of six samples per hour resulted in the choices for N and M respectively. The value of M should be reduced if the milling circuit has the risk of becoming unstable before the next inferential calculation.

The combined method employs the particle filter method and therefore requires the mill parameter estimates.

Algorithm

- Initialize average arrays to 0.
- Using three FOR loops, calculate the average of N time steps back at every M for each state. This is done for both methods.
- Find the difference between the two matrices. This is the offset between the neural network and the particle filter.
- For M time steps forward create a new matrix that subtracts the difference from the neural network estimation.
- M was chosen to be 360 i.e. 60 minutes or 1 hour.
- N was chosen to be 6 i.e. 6 samples.

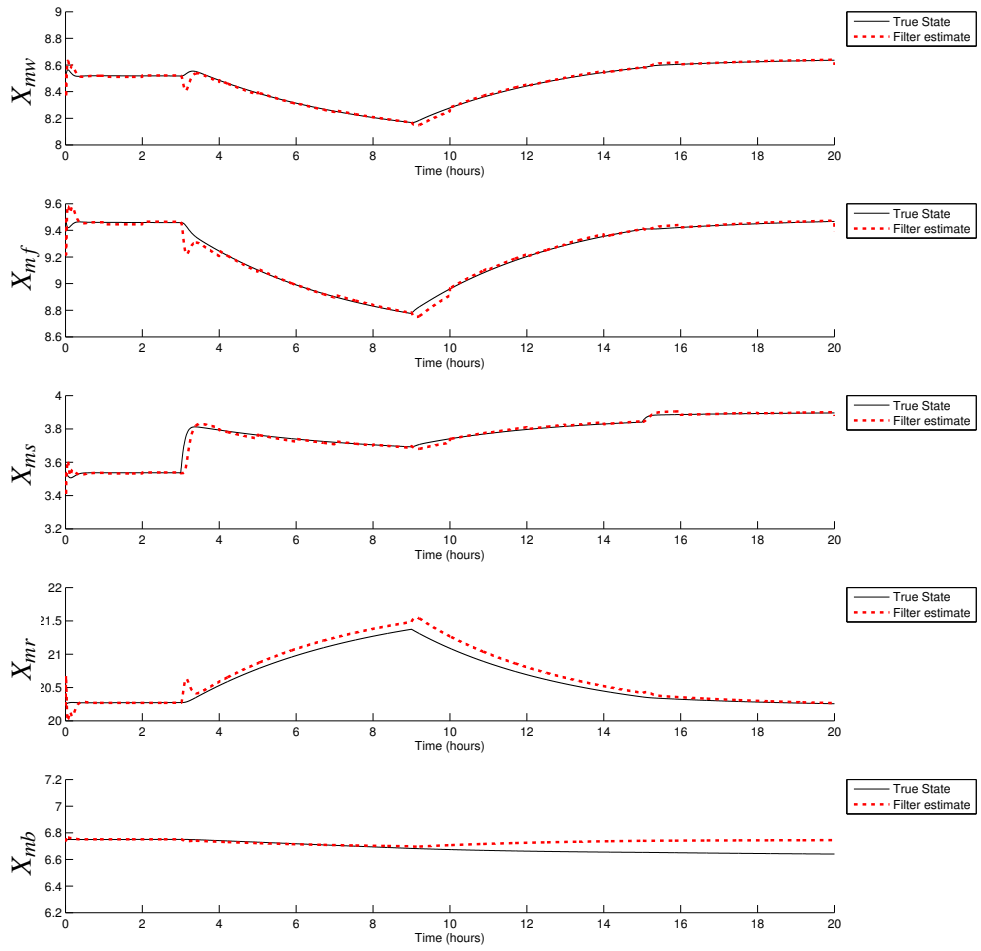

Figure 3.7: State estimation from neural network with particle filter correction

3.3.5 Comparison

The performance index shown in Fig. 3.8 clearly shows that an improvement has been made using the particle filter as an offset error correction method. The performance index used is shown in equation (3.7) where $\overline{X}_{yz}$ represents the ideal state and X_{yz} represents the estimated state.

$$PI = (\frac{\overline{X}_{mw} - X_{mw}}{\overline{X}_{mw}})^2 + (\frac{\overline{X}_{ms} - X_{ms}}{\overline{X}_{ms}})^2 + (\frac{\overline{X}_{mf} - X_{mf}}{\overline{X}_{mf}})^2 + (\frac{\overline{X}_{mr} - X_{mr}}{\overline{X}_{mr}})^2 + (\frac{\overline{X}_{mb} - X_{mb}}{\overline{X}_{mb}})^2 \tag{3.7}$$

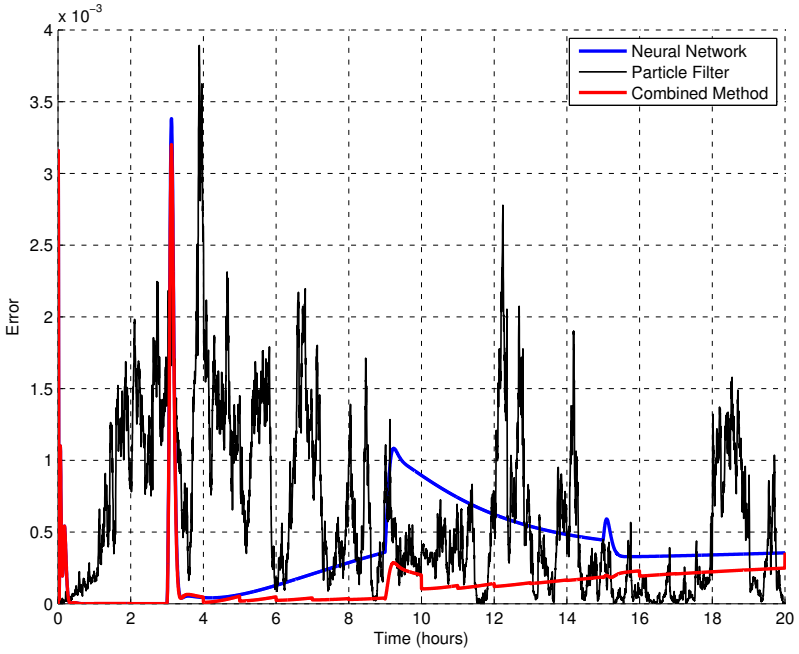

Figure 3.8: Squared error comparison of all three methods

Table 3.1 shows the results from 16 simulation scenarios. The first two scenarios illustrate the performance of the neural network training and therefore should have very small state errors. Tests number 3 to 7 illustrate a scenario when all three disturbance parameters are positive. This could occur when the parameters are underestimated. It should be noted that in Test number 7 and 12 the neural network could not estimate the states accurately, as a 30% disturbance change was made and the network was only trained on a 20% change. This shows that the neural network needs to be trained on a worse-case scenario of disturbance changes, up to 50% (Coetzee *et al.*, 2010), else the state estimation results are not reliable. Similarly, Test numbers 8 to 12 illustrate negative disturbance changes. Test numbers 13 to 16 illustrate a scenario when positive and negative disturbance changes

occur. The results show that as the magnitude of the disturbances increases, the estimation accuracy for both particle filter and neural network decreases. The particle filter results are more consistent compared to the neural network results. The combined method results are always better than the individual neural network and particle filter results except in Test number 7 and 12.

Table 3.1: State estimation validation tests and corresponding performance index results

Test No.	α_r		α_f		ϕ_f		State error[a]		
	Change (%)	Time (mins)	Change (%)	Time (mins)	Change (%)	Time (mins)	NN	PF	Combined
1	20	40	20	480	20	700	0.6893	16.849	0.4394
2	-20	40	-20	480	-20	700	0.8389	11.547	0.5505
3	5	540	5	900	5	180	0.3905	10.782	0.2399
4	10	540	10	900	10	180	1.0691	7.6039	0.7326
5	15	540	15	900	15	180	2.2655	15.126	1.4364
6	20	540	20	900	20	180	4.6991	13.889	2.2946
7	30	540	30	900	30	180	23.586	11.844	13.682
8	-5	540	-5	900	-5	180	1.2861	14.868	0.3885
9	-10	540	-10	900	-10	180	2.7745	13.913	1.0001
10	-15	540	-15	900	-15	180	3.4191	13.158	2.0793
11	-20	540	-20	900	-20	180	6.2087	17.003	3.5542
12	-30	540	-30	900	-30	180	75.278	13.967	14.469
13	-5	540	5	900	-5	180	1.4202	11.286	0.3881
14	-10	540	10	900	-10	180	2.6133	11.155	0.9994
15	-15	540	15	900	-15	180	3.3618	10.977	2.0905
16	-20	540	20	900	-20	180	7.7633	14.980	3.5137

[a] Based on the summation of equation 3.7 throughout a 20-hour simulation at the sampling interval of 10 seconds.

3.4 CONCLUSION

The work presented shows that it is possible to do internal state estimation for a milling circuit using a neural network trained on input data. The network was trained using disturbance changes in model parameters of 20%, and then used to predict the effect of 10% disturbances. A new method that combines a neural network and particle filter estimator for offset correction was presented. Initial results indicate that such a method can work well.

Training the neural network in practice will be difficult, as the method is highly dependent on the quality of the training data. It may also not be possible to train the neural network on one disturbance at a time, as was done here. Plants outside the minerals processing industry are often accompanied

by accurate simulators (Garatti and Bittanti, 2008) and according to Garatti and Bittanti (2008) a set of experiments using neural networks can be performed “virtually" by simulation trials. Further research should be done on how much training data, for this application, is required and to what extent the neural network method is accurate in mineral processing applications. A promising start has been made.

CHAPTER 4

CONTROL OF A ROM ORE MILLING CIRCUIT

4.1 INTRODUCTION

This chapter proposes a nonlinear suboptimal control approach with state estimation so that a high quality product at a high throughput is produced by the milling circuit despite large disturbances. The work presented in this chapter builds on the work of Olivier *et al.* (2012b) by developing a controller that uses the mill states provided by the particle filter. The novelty of the work presented in this chapter includes the estimation of the sump states, the augmented nonlinear controller and a combination of the particle filter with the control formulation i.e. including state estimation in the control loop. The formulations presented in this chapter aim to be as practically implementable as possible.

In previous work the particle filter required on-line measurements at the input and output of the mill. Because this is not always practically possible, various approximation and data filtering techniques are used based on practically viable measurements.

An innovative combined nonlinear model predictive control and nonlinear dynamic inversion design philosophy is presented in this chapter. State estimation of the contents of the mill and the sump, required for control, has also been adequately addressed. The proposed approach uses a nonlinear MPC for the overall circuit along with a fast acting dynamic inversion control for controlling the sump level. In summary, the proposed approach simultaneously meets the objectives of high product quality and high throughput, while ensuring manageable loads in the mill.

4.2 CONTROL FORMULATION

4.2.1 Control Objectives

The control objectives for the control formulation proposed are:

1. robust tracking of set-points or reference commands with independent control of PSE and THP,
2. accurate compensation for plant nonlinearities and
3. tight regulation of water volume in the sump.

A generalised control loop for mineral processing, developed by Hodouin (2011), is shown in Fig. 4.1. This figure illustrates how all the building blocks are connected to achieve successful control of a process. Successful control can be split up into the main controller (Block 2) and peripheral tools. Peripheral tools assist with determining the best control moves to overcome issues such as disturbances or model-plant mismatch. Block 1 and 6 can be optimized for power, material cost, minimum time etc. (e.g. using the supervisory control in Matthews and Craig (2013)), and block 5 has been studied by Venkatasubramanian, Rengaswamy, Kavuri and Yin (2003). Blocks 2, 3, 4, 7 and 8 are (partly) covered in this study.

In the proposed design, relevant physical limitations of the ROM ore milling circuit are also explicitly taken into account to the best possible extent. The variables in Table 2.1, excluding the rheology factor, are required to be measured online.

4.2.2 Control architecture

The ROM ore milling circuit is controlled using a combination of a NMPC and DI controller. The dynamics of the sump are significantly faster than the rest of the circuit and therefore it requires a fast acting controller. The dynamic inversion method was chosen for the fast dynamics while the NMPC manipulates the remaining variables. Instead of having these controllers isolated, the closed-form solution produced by the DI controller is used by the NMPC. As can be seen from (2.66) and (2.67), the volume of water in the sump (X_{sw}) has a significant effect on these equations and therefore on PSE and THP. This means that the dynamic inversion controller can assist the NMPC to aim for an optimal sump water set-point that can aid in minimising the objective function in the NMPC.

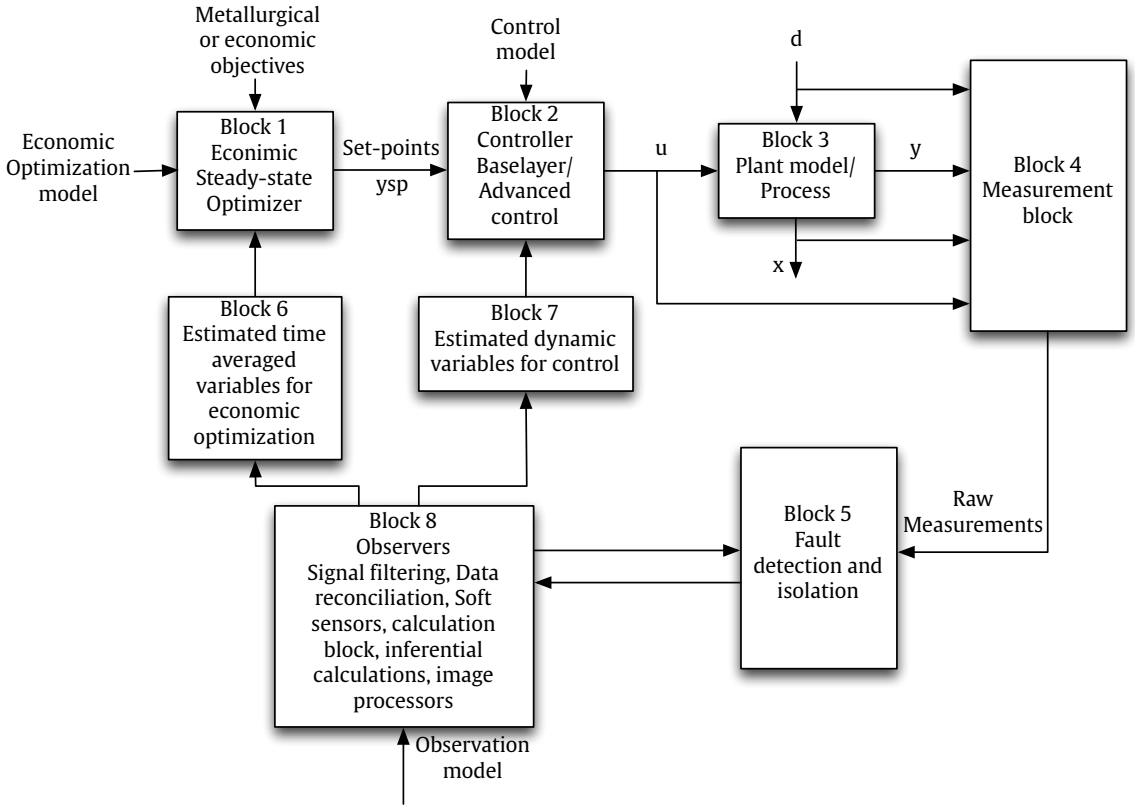

Figure 4.1: Generalised control loop for mineral processing.

Fig. 4.2 illustrates the connection between the NMPC and the DI controller. The volume of water in the sump is seen as a controlled variable by the DI controller. The desired value of the sump water (X^*_{sw}) is determined by the NMPC and fed to the DI controller as a set-point from which SFW is determined.

As shown in (2.56), X_{sw} has a significant impact on PSE, the ratio between the flow-rate of the fines and the flow-rate of solids out of the circuit. The sump constraints are taken into account using a convex combination, described in section 4.2.4.2, when the $SVOL$ reaches its physical limits.

4.2.3 Nonlinear model predictive control (NMPC)

The NMPC was designed using the full non-linear state-space equations as described in section 2.2.3. The objective of the controller is given by (4.1) where **u** represents the manipulated variables used to

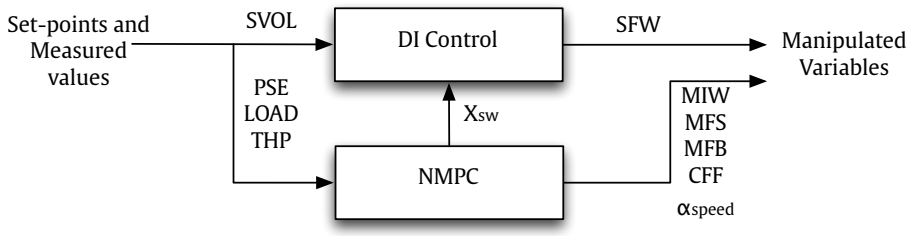

Figure 4.2: Block diagram illustrating the control configuration implemented.

solve the nonlinear optimal control problem by minimizing a performance index J.

$$\min_{\mathbf{u}} J(\mathbf{u}, \mathbf{x}_0, \mathbf{p}) \tag{4.1}$$

$$\text{such that } \mathbf{y} \in \mathbb{Y}, \mathbf{u} \in \mathbb{U} \tag{4.2}$$

$$\frac{d}{dt}\mathbf{x}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \tag{4.3}$$

$$\mathbf{y}(t) = g(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \tag{4.4}$$

In (4.1)-(4.13), $\mathbf{x}$ contains state variables with initial condition $\mathbf{x}_0$, $\mathbf{y}$ contains the controlled variables, $\mathbf{y}_{sp}$ contains the set-points, and $\mathbf{p}$ contains the parameter values for the system. The manipulated and controlled variables as given as follows.

$$\mathbf{u} = \{MIW, MFS, MFB, \alpha_{speed}, CFF, X^*_{sw}\}^T \tag{4.5}$$

$$\mathbf{y} = \{LOAD, PSE, THP\}^T \tag{4.6}$$

The upper and lower constraints for the controlled variables are given by $\mathbf{y}_u$ and $\mathbf{y}_l$ respectively. Similarly, for the input vector, $\mathbf{u}_u$ and $\mathbf{u}_l$ are the upper and lower constraints for the manipulated variables respectively.

$$\mathbf{u}_l = \{0, 0, 0, 0.4, 200, 2\}^T \tag{4.7}$$

$$\mathbf{u}_u = \{100, 200, 10, 1, 450, 12\}^T \tag{4.8}$$

$$\mathbf{y}_l = \{0.2, 0.5, 10\}^T \tag{4.9}$$

$$\mathbf{y}_u = \{0.5, 0.85, 30\}^T \tag{4.10}$$

$$\mathbb{U} = \{\mathbf{u} \in \mathbb{R}^{n_u} | \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u\} \tag{4.11}$$

$$\mathbb{Y} = \{\mathbf{y} \in \mathbb{R}^{n_y} | \mathbf{y}_l \leq \mathbf{y} \leq \mathbf{y}_u\} \tag{4.12}$$

The objective function for the NMPC is given as

$$J(\mathbf{u},\mathbf{x}_0,\mathbf{p}) = \sum_{n=1}^{N_p}(\mathbf{y}-\mathbf{y}_{sp})^T\mathbf{Q}_1(\mathbf{y}-\mathbf{y}_{sp}) + \sum_{n=1}^{N_p}\Delta\mathbf{y}^T\mathbf{Q}_2\Delta\mathbf{y} + \sum_{n=1}^{N_c}\Delta\mathbf{u}^T\mathbf{Q}_3\Delta\mathbf{u} \tag{4.13}$$

where $\Delta\mathbf{y} = \mathbf{y}^n - \mathbf{y}^{n-1}$ and $\Delta\mathbf{u} = \mathbf{u}^n - \mathbf{u}^{n-1}$. In general, the $\Delta\mathbf{u}$ term provides integral action as long as there are no steady-state targets for the manipulated variables in the objective function. However, in practice only this may be insufficient to get offset free tracking, because the state estimate returned by the observer as an initial condition for the prediction model might result in a solution that reaches the set-points during prediction, but a consistent disturbance or model mismatch is preventing that from happening in reality. The integrating action of the MPC through $\Delta\mathbf{u}$ will only work until the model is predicting that set-point will be reached. If the model is, however, not being updated to include the effects of disturbances or model mismatch, the results predicted by the model and reality will not match. Model parameter update is highly recommended, however, is out of the scope of this study.

The prediction horizon (N_p) should be chosen based on the longest settling time between the manipulated and controlled variables (Seborg, Edgar and Mellichamp, 2004). This results in an $N_p > 500$ because of the PSE-α_{speed} interaction, which is impractical to implement. The milling circuit model has a large range of such settling times, and to yield a controller that is feasible to implement, $N_p = 18$ (corresponding to 18 minutes with a sampling time of one minute) was chosen. Good results are achieved when the control horizon is selected to be 3 ($N_c = 3$). This is a good trade-off between aggressiveness and computational expense.

The matrices $\mathbf{Q}_1$, $\mathbf{Q}_2$ and $\mathbf{Q}_3$ weight the controlled variables, and the rates of change of the controlled variables and manipulated variables respectively. These are tuning parameters for the NMPC. PSE and THP have a significantly higher priority to meet set-point than $LOAD$. α_{speed} is included as a manipulated variable to control PSE and THP and is constrained between 0.4 and 1. In (4.15)-(4.24), y_{-max} and r_{-max} represent the allowed deviation for the controlled and manipulated variables respectively. Note that these are absolute value changes. The $\bar{q}_-$ values represent the priority of the controlled variables. The $\bar{r}_-$ values control the smoothness of the manipulated variables with a large

value resulting in a small rate of change.

$$\mathbf{Q}_1 = diag(q_1, q_2, q_3) \tag{4.14}$$

$$q_1 = \frac{\bar{q}_1}{y_{1_{max}}^2}, y_{1_{max}} = 0.05, \bar{q}_1 = 1 \tag{4.15}$$

$$q_2 = \frac{\bar{q}_2}{y_{2_{max}}^2}, y_{2_{max}} = 0.05, \bar{q}_2 = 10 \tag{4.16}$$

$$q_3 = \frac{\bar{q}_3}{y_{3_{max}}^2}, y_{3_{max}} = 0.5, \bar{q}_3 = 10 \tag{4.17}$$

All of the weights above have been normalized by an acceptable variation e.g. an acceptable variation for *PSE* is 0.05. The weight that determines the accuracy of the *LOAD* output was chosen to be 1. As *PSE* and *THP* are significantly more important than *LOAD*, these two outputs are given a weight of 10. The movement of the controlled variables is minimized with the weight $\mathbf{Q}_2 = 10^{-3} diag\,(q_1, q_2, q_3)$. The movement of the manipulated variables is minimized as follows:

$$\mathbf{Q}_3 = diag(r_1, r_2, r_3, r_4, r_5, r_6) \tag{4.18}$$

$$r_1 = \frac{\bar{r}_1}{r_{1_{max}}^2}, r_{1_{max}} = 20, \bar{r}_1 = 0.1 \tag{4.19}$$

$$r_2 = \frac{\bar{r}_2}{r_{2_{max}}^2}, r_{2_{max}} = 25, \bar{r}_2 = 10 \tag{4.20}$$

$$r_3 = \frac{\bar{r}_3}{r_{3_{max}}^2}, r_{3_{max}} = 2, \bar{r}_3 = 10 \tag{4.21}$$

$$r_4 = \frac{\bar{r}_4}{r_{4_{max}}^2}, r_{4_{max}} = 0.2, \bar{r}_4 = 1 \tag{4.22}$$

$$r_5 = \frac{\bar{r}_5}{r_{5_{max}}^2}, r_{5_{max}} = 50, \bar{r}_5 = 1 \tag{4.23}$$

$$r_6 = \frac{\bar{r}_6}{r_{6_{max}}^2}, r_{6_{max}} = 2, \bar{r}_6 = 10. \tag{4.24}$$

The *MIW* variable has been allowed to vary the most as this variable is practically the simplest to manipulate - it should however not be increased too quickly as this could have the effect of washing out the fines in the mill. *CFF* and α_{speed} control the speed of the sump pump and mill motor and are given a weight of 1. An additional constraint on α_{speed} was added to ensure smooth control: $\Delta\alpha_{speed} \leq 0.01$. It is undesirable to see major movements in raw ore and steel balls into the mill therefore these two manipulated variables have a weight of 10. Additionally, due to the slow grinding process, it is preferred that the sump water not move rapidly.

The *fmincon* function with the *active-set* algorithm in MATLAB[1] was used to determine the six control values. The objective function propagates the state and output functions for the prediction horizon. The Euler method was used for the propagation of states within the nonlinear model predictive controller. The fourth order Runge-Kutta method (Press, Teukolsky, Vetterling and Flannery, 1992) was used in the overall control loop after the control signals, obtained from the DI and NMPC controllers, were calculated.

4.2.4 Dynamic Inversion (DI)

The sump volume is traditionally controlled using a single-loop PI(D) controller. From (2.49), it is clear that the sump dynamics are nonlinear, even though a sump model for level control can often be approximated as an integrator depending on the shape of the sump (Craig and MacLeod, 1996). Dynamic inversion is used for $SVOL$ and X_{sw} control because of its simple design structure. No approximation of the system dynamics is necessary and the controller is easy to implement online. This method leads to a closed form solution for the controller and guarantees asymptotic stability for the error dynamics (Enns, Bugajski, Hendrick and Stein, 1994). The connection between the two controllers is simply that the NMPC sends a desired value of X_{sw} to the DI controller so that the NMPC can achieve specific aims. SFW has the largest impact on both X_{sw} and $SVOL$ and was therefore used as the manipulated variable.

Dynamic inversion allows the specification of a desired response path by choosing a proportional gain value (K_p) and an integral gain value (K_I) such that:

$$\frac{d}{dt}E + K_p E + K_I \int_0^t E d\tau = 0. \tag{4.25}$$

where $E = Y - Y_{sp}$ (the difference between the measured value (Y) and the set-point (Y_{sp})).

4.2.4.1 Synthesis of SFW

SFW consists of two parts: SFW_{SVOL} and $SFW_{X_{sw}}$. SFW_{SVOL} is the manipulated variable SFW that is used to control $SVOL$ only (ignoring the NMPC-DI connection) whereas $SFW_{X_{sw}}$ uses SFW to achieve the desired X_{sw}^* obtained from the NMPC. Equation (4.26) is the result for $SVOL$ control when $E = \Delta S_{vol} = SVOL - SVOL^*$ where $SVOL^*$ represents the upper bound, when the sump volume is greater than the midpoint of the sump, or lower bound, when the sump volume is below the midpoint.

[1] MATLAB is a registered trademark of The MathWorks Inc.

Equation (4.28) results from algebraic manipulation and solving for the control variable SFW.

$$\frac{d}{dt}(\Delta S_{vol}) + K_{p1}(\Delta S_{vol}) + K_{I1}\int_0^t (\Delta S_{vol})d\tau = 0 \tag{4.26}$$

$$\frac{d}{dt}(\Delta S_{vol}) = (V_{swi} + V_{ssi}) - CFF + SFW_{SVOL} \tag{4.27}$$

$$SFW_{SVOL} = (CFF - V_{swi} - V_{ssi}) - K_{p1}(\Delta S_{vol}) - K_{I1}\int_0^t (\Delta S_{vol})d\tau \tag{4.28}$$

K_{p1} was tuned to be 30, based on a desired settling time of approximately 2 minutes. K_{I1} was chosen to be ten times less than K_{p1}. This results in good sump volume control without exceeding any input constraints on SFW. The sump volume must be maintained between 2 and 20 m^3.

Similarly for sump water (X_{sw}) control: $E = \Delta X_{sw} = X_{sw} - X^*_{sw}$. The solution for $SFW_{X_{sw}}$ is shown in (4.29).

$$SFW_{X_{sw}} = \frac{CFFX_{sw}}{X_{sw} + X_{ss}} - \frac{V_V \phi X^2_{mw}}{X_{ms} + X_{mw}} - K_{I2}\int_0^t (\Delta X_{sw})d\tau - K_{p2}(\Delta X_{sw}) \tag{4.29}$$

K_{p2} was chosen to be ten times larger than the $SVOL$ K_{p1} value as fast control was desired. The K_{I2} value was also chosen to be ten times less than the K_{p2}.

4.2.4.2 Convex combination

A convex combination of SFW_{SVOL} and $SFW_{X_{sw}}$ was chosen for smooth control of both outputs ($SVOL$ and X_{sw}). Fig. 4.3 illustrates the proposed concept. A safe region is defined, where there is no concern of the sump overflowing or running dry, between the lower bound (LB) and upper bound (UB). In the safe region the focus of the control variable SFW can primarily be on controlling X_{sw}. As the sump volume approaches the constraints (Min and Max), more focus should be on bringing the sump volume back to the safe region. This means that SFW_{SVOL} should have preference over $SFW_{X_{sw}}$. This concept is mathematically shown in (4.30) and (5.16).

$$SFW = (1 - \lambda)SFW_{X_{sw}} + \lambda SFW_{SVOL} \tag{4.30}$$

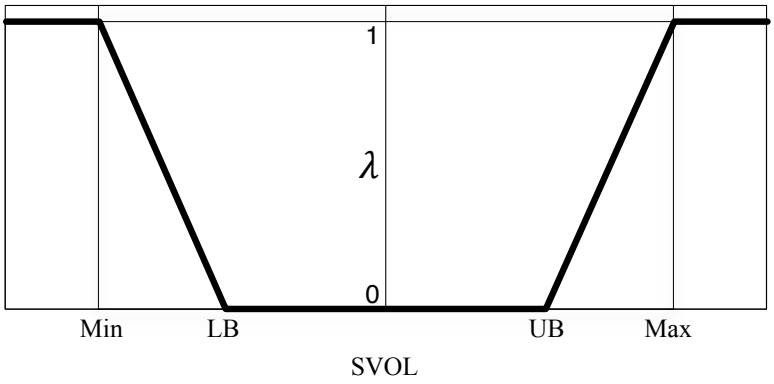

Figure 4.3: Conceptual figure illustrating convex combination.

$$\lambda(SVOL) = \begin{cases} 0 & \text{if } LB < SVOL < UB \\ \frac{1}{Min-LB}(SVOL - LB) & \text{if } Min \leq SVOL \leq LB \\ \frac{1}{Max-UB}(SVOL - UB) & \text{if } UB \leq SVOL \leq Max \\ & \text{if } SVOL < Min \text{ or} \\ 1 & SVOL > Max \end{cases} \tag{4.31}$$

where Lower Bound$(LB) = 4$, Upper Bound$(UB) = 16$,

$Min = 2,\ Max = 20$

4.3 STATE ESTIMATION

In a recent paper by Olivier *et al.* (2012b), a particle filter was used for estimation of the states representing the contents of a grinding mill. Only the volume of solids (X_{ms}), water (X_{mw}) and fines (X_{mf}) were estimated accurately since it was assumed that the flow-rate of each component out of the mill could be measured. However, measuring the flow-rate of these components is not practically possible. The volume of rocks (X_{mr}) and balls (X_{mb}) were not accurately estimated, because by definition these components do not exit the mill. The estimation accuracy of these two states (X_{mb} and X_{mr}) can be improved by using a neural network alongside the particle filter as discussed by Naidoo, Olivier and Craig (2013). Yet, only the particle filter was used in this study since it showed more accurate and reliable results in the presence of disturbances and noise. Additionally, the neural network approach requires a significant amount of training data that could be difficult to obtain from

an industrial plant. The combined method was not as accurate compared to using only the particle filter method when parameter variations of 30% (see Table 3.1) were introduced. In this study, seven parameters are varied from 5% to 50%. Therefore a more industrially viable and accurate option under these conditions, the particle filter method, was chosen.

Because there are no direct measurements of the discharge of the mill, and because balls and rocks do not discharge from the mill, observing the mill states is not straightforward. The only measurement available for the material in the mill is the *LOAD* measurement. However, the sump states are readily observable from the measurable outputs $SVOL$, CFD and PSE. If the sump states are known, it is possible to determine the flow-rate of solids, water and fines out of the mill by means of a flow-rate balance over the sump. Estimating the sump states separately from the mill states makes additional information available for a more accurate estimation of the mill states.

The next two subsections describe the process followed to estimate the sump and mill states.

4.3.1 State estimation for the sump states

The sump state equations are given in (2.49) to (2.51). The sump volume ($SVOL$) and cyclone feed density (CFD) contain two of the three sump states. Since the sump volume ($SVOL$) and cyclone feed density (CFD) are measurable, algebraic manipulation of (2.54) and (2.55) results in (4.32) and (4.33).

$$X_{ss_{est}} = SVOL\frac{(1-CFD)}{(1-D_s)} \tag{4.32}$$

$$X_{sw_{est}} = SVOL - X_{ss} \tag{4.33}$$

The third state, the volume of fines in the sump (X_{sf}), is difficult to estimate because a third measurement is required. Because the fraction of fines in the sump outflow is not measured, the measurement of the fines in the overflow of the cyclone (PSE) has to be used to estimate the volume of fines in the sump (X_{sf}). A weighted moving average is proposed to estimate the volume of fines in the sump (X_{sf}), as this ensures a smooth X_{sf} estimate. The mean of the previous four estimates are combined with the new estimated X^*_{sf}.

$$X_{sf_{est}}(k) = 0.5X^*_{sf} + 0.5\frac{1}{n}\sum_{i=1}^{n=4} X_{sf_{est}}(k-i) \tag{4.34}$$

X^*_{sf} is determined by writing the fraction of fines in the overflow of the cyclone (PSE) in terms of the states and manipulated variables and only then solving numerically for X^*_{sf} - see (4.35). This was

implemented using the *fsolve* function in MATLAB.

$$
\begin{aligned}
PSE =& \frac{100\left(A-\frac{CFFX_{sf}^{*}}{X_{ss}+X_{sw}}\right)}{A-\frac{CFFX_{ss}}{X_{ss}+X_{sw}}+\frac{C}{X_{ss}+X_{sw}}} \\
&\text{where} \\
A =& \left(\frac{X_{sf}^{*}}{X_{ss}+X_{sw}}\left(C-CFF(X_{sf}^{*}-X_{ss})\left(\left(\frac{X_{sf}^{*}}{X_{ss}}\right)^{C_4}-1\right)\times\right.\right. \\
& \left.\left.B\left(\left(\frac{X_{ss}}{C_2(X_{ss}+X_{sw})}\right)^{C_3}-1\right)\left(C_1\exp\left(-\frac{CFF}{\varepsilon_s}\right)-1\right)\right)\right/ \\
& (X_{sf}^{*}B-X_{sf}^{*}+X_{sw}B) \\
B =& \exp\left(-\frac{C}{\alpha_{su}\varepsilon_s(X_{ss}+X_{sw})}\right)\left(\frac{X_{ss}}{X_{ss}+X_{sw}}-0.6\right)+0.6 \\
C =& CFF(X_{sf}^{*}-X_{ss})\left(\left(\frac{X_{sf}^{*}}{X_{ss}}\right)^{C_4}-1\right)\left(\left(\frac{X_{ss}}{C_2(X_{ss}+X_{sw})}\right)^{C_3}-1\right) \\
& \left(C_1\exp\left(-\frac{CFF}{\varepsilon_s}\right)-1\right)
\end{aligned}
\tag{4.35}
$$

4.3.2 State estimation for the mill states

A description of the particle filter has been discussed in section 3.3.2. The particle filter is required to estimate the state vector $\mathbf{x}$, where $\mathbf{x} = \{X_{mw}, X_{ms}, X_{mf}, X_{mr}, X_{mb}\}$. The following inputs into the mill are required: MIW, MFS, MFB, α_{speed}, V_{cwu}, V_{csu} and V_{cfu}. The mill outputs are also required: V_{mwo}, V_{mso}, V_{mfo}, $LOAD$ and P_{mill}. A particle filter with 1000 particles is specified for estimating the mill states. Note that a larger number of particles was also tried without much improvement in the results. Table 4.1 illustrates that there is no significant improvement in state estimation when using 2000 particles however, there is a significant computation time difference. The computational time is the average time (in seconds) per iteration. As in Olivier *et al.* (2012b), the transitional prior is used as the importance density. This is a suboptimal choice when a more complex importance density is not warranted. The transitional prior is also chosen because of the ease of implementation and good accuracy. The initial estimates of the mill states are randomly selected from a region ($\pm$ 10%) around the actual initial values in each case. The formulation used here is exactly the same as described in Olivier *et al.* (2012b).

The particle filter requires the mill outflows exiting through the discharge grate and the cyclone underflows entering the mill. The flow-rate components into the mill (4.41)-(4.43) can be determined from knowing the sump states (X_{sw}, X_{ss}, X_{sf}) and using fraction solids in the total cyclone inflow volume F_i

Table 4.1: Comparison between number of particles.

Number of particles	Computational time (s)	Error*
10	0.0109	0.3190
50	0.0245	0.2140
250	0.0809	0.2189
500	0.1516	0.2899
1000	0.2917	0.1808
2000	0.5663	0.1759

*Error = sum(abs($X_{estimated} - X_{simulated}$))

in (4.36), fraction fines in the cyclone feed solids P_i in (4.37), fraction solids in the cyclone underflow volume F_u in (4.38) and the flowrate of coarse ore entering the cyclone V_{cci} in (4.39).

$$F_i = \frac{X_{ss}}{(X_{sw} + X_{ss})} \tag{4.36}$$

$$P_i = \frac{X_{sf}}{X_{ss}} \tag{4.37}$$

$$F_u = 0.6 - (0.6 - F_i) \cdot \exp\left(-V_{ccu}/(\alpha_{su}\varepsilon_c)\right) \tag{4.38}$$

$$V_{cci} = CFF\frac{(X_{ss} - X_{sf})}{(X_{ss} + X_{sw})} \tag{4.39}$$

$$V_{ccu} = V_{cci}\left(1 - C_1 \exp\left(\frac{-CFF}{\varepsilon_c}\right)\right)\left(1 - \left(\frac{F_i}{C_2}\right)^{C_3}\right)\left(1 - P_i^{C_4}\right) \tag{4.40}$$

$$V_{cwu} = \frac{X_{sw}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{4.41}$$

$$V_{cfu} = \frac{X_{sf}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{4.42}$$

$$V_{csu} = V_{ccu} + \frac{X_{sf}\left(V_{ccu} - F_u V_{ccu}\right)}{F_u X_{sw} + F_u X_{sf} - X_{sf}} \tag{4.43}$$

The outputs of the mill are directly fed into the sump and therefore the mill component outputs are the same as the sump inputs: $V_{mwo} = V_{swi}$, $V_{mso} = V_{ssi}$ and $V_{mfo} = V_{sfi}$. Equations (4.44) to (4.46) result from using the sump state equations (2.49) to (2.51) and are used in the particle filter.

$$V_{swi} = \frac{d}{dt}X_{sw} + CFF\frac{X_{sw}}{SVOL} - SFW \tag{4.44}$$

$$V_{ssi} = \frac{d}{dt}X_{ss} + CFF\frac{X_{ss}}{SVOL} \tag{4.45}$$

$$V_{sfi} = \frac{d}{dt}X_{sf} + CFF\frac{X_{sf}}{SVOL} \tag{4.46}$$

In the above equations, the derivative term can cause noise issues. Data filtering methods such as an exponential filter, moving-average filter or a noise-spike filter (Seborg *et al.*, 2004) applied to the derivative terms will assist with noisy system measurements. The above input and output equations for the mill module will allow the particle filter to be practically feasible.

Table 4.2 illustrates all the variables used in the proposed control configuration.

Table 4.2: Summary of the variables.

Type of variable	NMPC	DI controller	Sump state estimator	Mill state estimator	Plant
Manipulated variables (MVs)	MIW, MFS, MFB, CFF, α_{speed}, X_{sw}	SFW	-	-	MIW, MFS, MFB, CFF, α_{speed}, SFW
Controlled variables (CVs)	PSE, THP, $LOAD$	$SVOL$, X_{sw}	-	-	PSE, THP, $LOAD$, $SVOL$
Output variables (OVs)	P_{mill}, CFD, φ	-	-	-	P_{mill}, CFD, φ
State variables	X_{mw}, X_{ms}, X_{mf}, X_{mr}, X_{mb}, X_{sw}, X_{ss}, X_{sf}	X_{mw}, X_{ms}, X_{sw}, X_{ss}	-	-	X_{mw}, X_{ms}, X_{mf}, X_{mr}, X_{mb}, X_{sw}, X_{ss}, X_{sf}
Measured variables	Above MVs and CVs	Above MVs and CVs	$SVOL$, CFD, PSE	PSE, THP, $LOAD$, $SVOL$	-
Estimated variables	-	-	X_{sw}, X_{ss}, X_{sf}	X_{mw}, X_{ms}, X_{mf}, X_{mr}, X_{mb}	-

4.4 RESULTS

The aim of the results section is to illustrate how well the proposed control configuration is able to meet the control objectives set in section 4.2.1. It contains simulations that showcase the performance of the controller configuration at various set-points in the presence of noise and disturbances.

4.4.1 Simulation setup

The set-point for throughput (THP) was initially selected at 19.45 t/h and increased to 20.42 t/h after 3 hours and increased further to 21.4 t/h after 9 hours. Similarly, the particle size estimate (PSE) set-point was initially selected at 68.72 %, then increased to 72.16 % and further increased to 75.59 % after 3 and 9 hours respectively.

A twelve hour simulation was performed using the nonlinear controller and the particle filter for state estimation described earlier. A sampling time of 10 seconds was used. The NMPC used a one minute sampling time as the manipulated variables determined by the NMPC were only executed once a minute. The dynamic inversion control (i.e. SFW) executed every 10 seconds. These sampling time choices provided a good balance between the fast sump dynamics and the slow mill dynamics.

The *fmincon* function requires an objective function and can cater for various constraint functions (such as equality and inequality constraints). Within the objective function, the states and outputs of the mill and sump are propagated (which are a function of the control moves). The outputs are checked to be within inequality constraints defined in the constraint function. The constraint function comprises of inequality input and output constraints defined in Le Roux *et al.* (2013b). This hard limits the control moves and ensures that the inputs and outputs are always within the region of interest. The inequality constraints for the inputs take the general form $u^{min} \leq \overline{u}^1, \overline{u}^2, ..., \overline{u}^{N_c} \leq u^{max}$ and are given by:

$$0 \leq MIW \leq 100 \tag{4.47}$$

$$0 \leq MFS \leq 200 \tag{4.48}$$

$$0 \leq MFB \leq 8 \tag{4.49}$$

$$0.4 \leq \alpha_{speed} \leq 1 \tag{4.50}$$

$$|\alpha^N_{speed} - \alpha^{N-1}_{speed}| < 0.01 \tag{4.51}$$

$$200 \leq CFF \leq 450 \tag{4.52}$$

Similarly, the output constraints are given by:

$$0.5 \leq PSE \leq 0.85 \tag{4.53}$$

$$0.2 \leq LOAD \leq 0.5 \tag{4.54}$$

$$10 \leq THP \leq 30 \tag{4.55}$$

4.4.2 Noise and disturbances

The aim of including significant noise and disturbances is to provide some indication of how the controller might perform in practice. The model parameters and noise values were obtained from real plant data (Le Roux *et al.*, 2013b; Le Roux, Craig and Padhi, 2013a).

Noise values were randomly determined from the normal distribution for each of the outputs shown in Table 4.3. Table 4.3 shows the standard deviation of the noise.

Table 4.3: Output noise added to the system.

Output	Noise upper bound
$LOAD$	0.3396e-4 (0.01%)
$SVOL$	0.6e-3 (0.01%)
PSE	0.0137 (2%)
THP	0.1751 (1%)
P_{mill}	18.3 (1%)
CFD	0.0084 (0.5%)

In practice, parameters are constantly changing due to changes in the ore fed to the milling circuit. Therefore, the following parameters were varied sinusoidally as indicated: α_f, α_r, α_{su}, ε_s, ϕ_b, ϕ_f and ϕ_r by 50%, 50%, 5%, 5%, 5%, 50% and 20% respectively. Fig. 4.4 illustrates the parameter variations implemented in the twelve hour simulation. The percentages represent the maximum allowable amplitude in the sinusoidal variation, and the values used are given in Coetzee *et al.* (2010). A random amplitude variation occurs every 40 minutes given the sinusoidal function. The random variation follows a uniform distribution. The mean value of the parameter times the allowed percentage change gives the range of the distribution around the mean value. An example is shown in eq. (4.56) and

(4.57).

$$\alpha_r \sim U\left(\left(1-\frac{50\%}{2}\right)E(\alpha_r),\left(1+\frac{50\%}{2}\right)E(\alpha_r)\right) \tag{4.56}$$

$$\phi_r \sim U\left(\left(1-\frac{20\%}{2}\right)E(\phi_r),\left(1+\frac{20\%}{2}\right)E(\phi_r)\right) \tag{4.57}$$

The means that the distribution of ϕ_r is uniform where the minimum allowed value is $\left(1-\frac{20\%}{2}\right)E(\phi_r)$ and the maximum allowed value is $\left(1+\frac{20\%}{2}\right)E(\phi_r)$ and $E(\phi_r)$ is the expected value of ϕ_r.

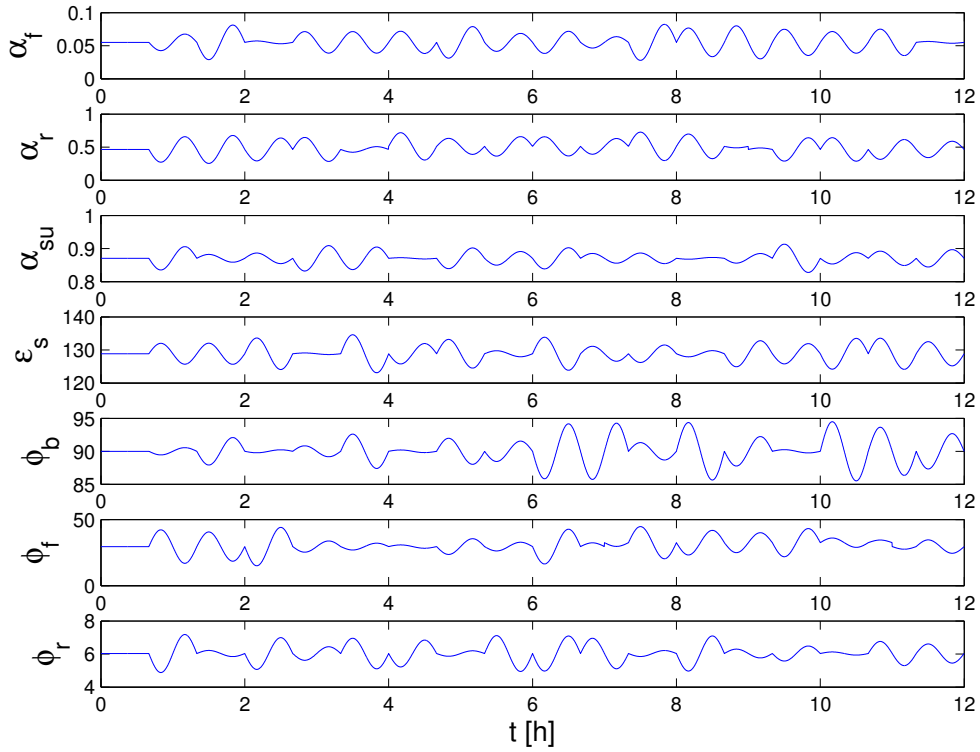

Figure 4.4: Parameter variations implemented in simulation.

Note that all these disturbances were only fed to the plant and not the state estimation models or the controller models.

4.4.3 Results with noise and disturbances

The accuracy of the particle filter described in section 4.3.2 is shown in Fig. 4.9. The added noise resulted in a deterioration of the state estimates, requiring modifications to the ideal state equations to improve accuracy. For the sump, the average values for $SVOL$ and CFD are used in the equations

describing the volume X_{ss} (4.32), X_{sw} (4.33) and X_{sf} (4.34). These average values were determined from the four most recent measured values. Equations (4.44)-(4.46) contain derivative terms which need to be filtered when noise is present. The MATLAB functions *polyfit* and *polyval* are used to fit a fifth order polynomial to the last 18 data points and the derivative of the polynomial function is used in (4.44)-(4.46). This resulted in a significantly smoother flow-rate term compared to using the last two data points and an algebraic difference equation.

Two simulations were implemented. The first simulation assumes full state feedback (no state estimation was used) to illustrate the performance of the controller under ideal conditions. The second simulation includes state estimation and illustrates the performance of the complete proposed control configuration.

Fig. 4.5 illustrates the performance for the *LOAD*, *SVOL* and *PSE*. *PSE* followed the set-point well despite the noise and parameter variation added. However the *LOAD* does not perform as well, this is due to the high weight, in the NMPC, for *PSE* and the low weight for *LOAD*. The accuracy of the *LOAD* can be easily increased by increasing the weight in the NMPC. The *SVOL* is safely within the bounds and there is no risk of overflowing or running dry.

The simulation with state estimation shows a more noisy output as the controller is now based on an estimated value as opposed to the actual state value.

Fig. 4.6 illustrates the controlled variable THP as well as additional outputs P_{mill} and CFD. THP is controlled well however, the variations, especially ϕ_f (based on numerous trial simulations), effect this controlled output the most. This shows that the proposed solution can perform well when aiming for a feasible desired quality and quantity. The simultaneous objectives of *PSE* and THP control have not been adequately covered by previously, see e.g. Coetzee *et al.* (2010).

When state estimation is included in the loop, the THP variation is larger than when using ideal states. Updating the parameters over time will increase the accuracy of THP significantly. Simultaneous parameter and state estimation (Olivier *et al.*, 2012b) should prove to be beneficial in this scenario.

Fig. 4.7 shows the performance of the dynamic inversion controller. From the figure it can be seen that the tracking performance is good. This is because the sump volume is safely within the constraints. If the sump volume moves towards the maximum or minimum bounds then the tracking performance

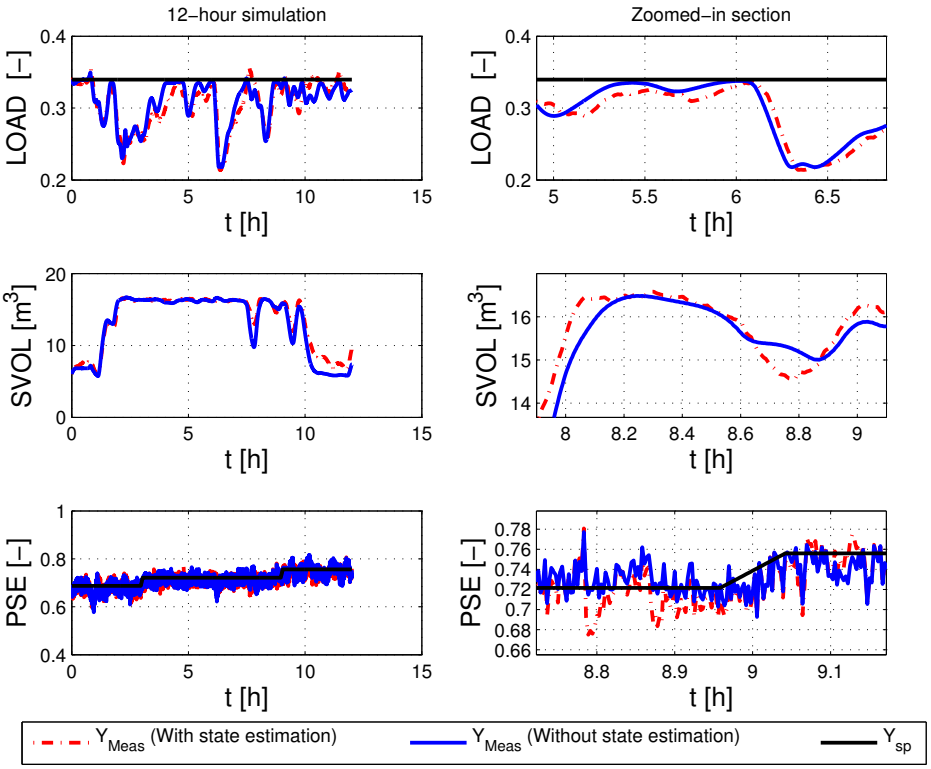

Figure 4.5: Controlled outputs with state estimation and control.

shown in the figure will deteriorate and more emphasis will be placed on moving away from limits, invoking the convex combination of (5.16).

In this scenario the upper bound was at 16 m^3 for the $SVOL$. After 8 hours, from Fig. 4.5, the $SVOL$ moves slightly above 16 m^3 and therefore there is a drive to bring the sump volume back within the bound. This results in X_{sw} reaching approximately 11.5 m^3 instead of 12 m^3 just after 8 hours.

Both simulations, with and without state estimation, show a slightly delayed response because of the average $SVOL$ and CFD values used for calculation of the sump states.

The three sump states are shown in Fig. 4.8 which show that the sump states can be estimated accurately despite the noise and disturbances. To ensure that these estimates are smooth, the average of the last five values for $SVOL$ and CFD were used in (4.32) and (4.33). The last state (X_{sf}) is calculated using the formulation in (4.34). The estimated states are marginally delayed compared to the actual state values, but this difference is small and can be regarded as negligible. Note that the sump states need to be as accurate and smooth as possible as the mill state estimation is dependent on

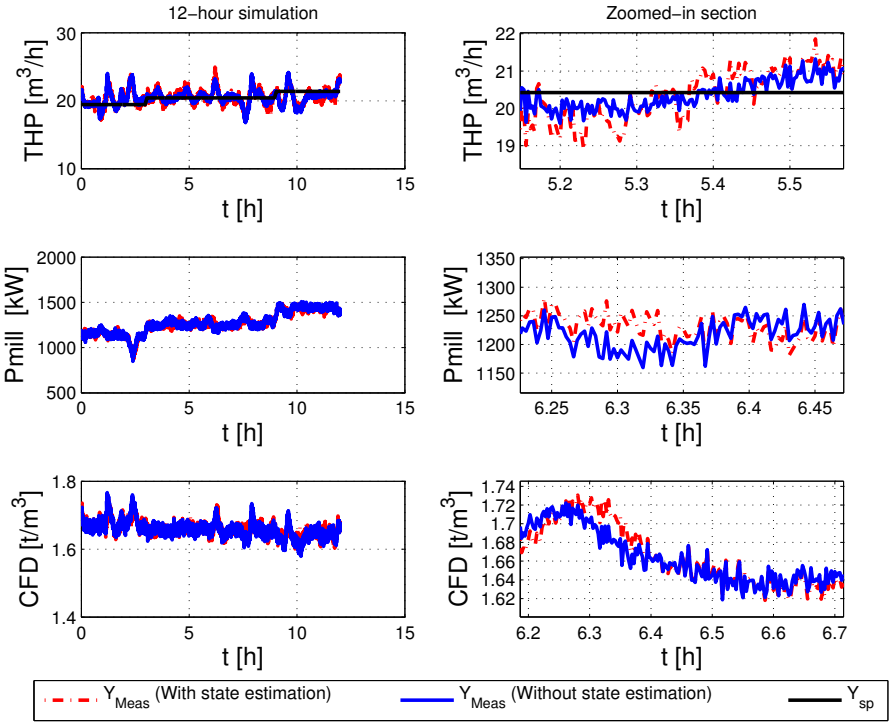

Figure 4.6: Controlled outputs with state estimation and control.

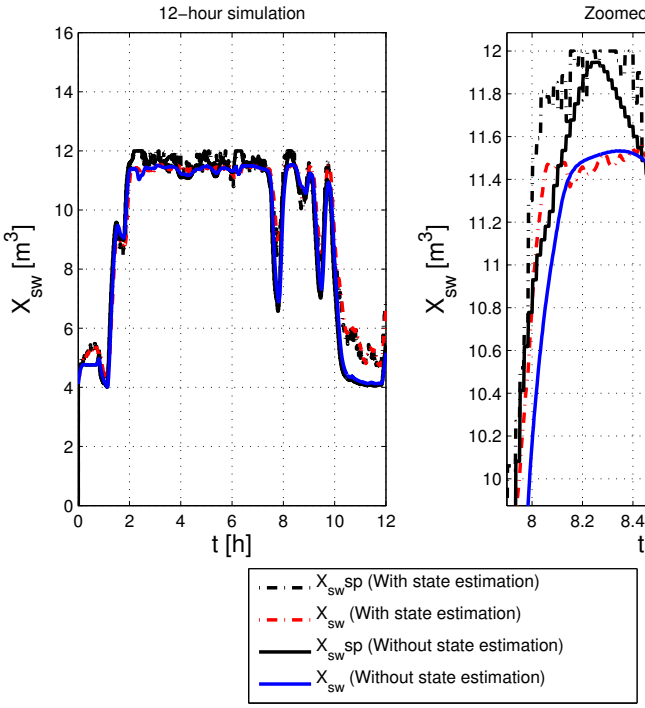

Figure 4.7: Dynamic inversion control.

these results.

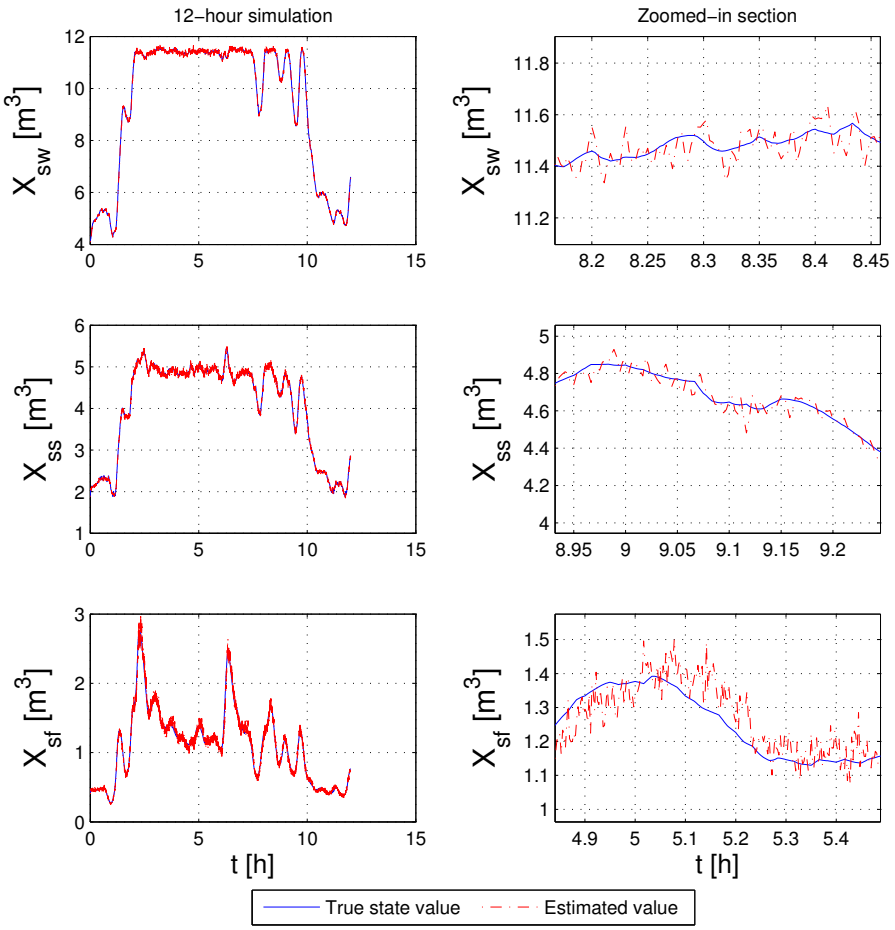

Figure 4.8: Sump state estimates.

The five mill states are shown in Fig. 4.9. The first two states are estimated accurately despite the noise and disturbances. The estimate of the third state (volume of fines) is deemed acceptable and is dependent on the accuracy of the estimate for X_{sf}. The estimate of the volume of rocks within the mill is acceptable even though α_r significantly affects this estimate. The estimate for the volume of steel balls is reasonable despite the added noise and disturbances. The rocks and steel balls do not leave the mill and are therefore more difficult to estimate.

Fig. 4.10 shows the manipulated variables for the mill. All the variables are within the constraints. The MIW hits the lower constraint of 0, if this is undesirable, the MIW could be removed as a manipulated variable and set as a fixed ratio to the MFS.

Fig. 4.11 shows the α_{speed}, SFW and CFF. SFW and α_{speed} do not hit any constraints. CFF

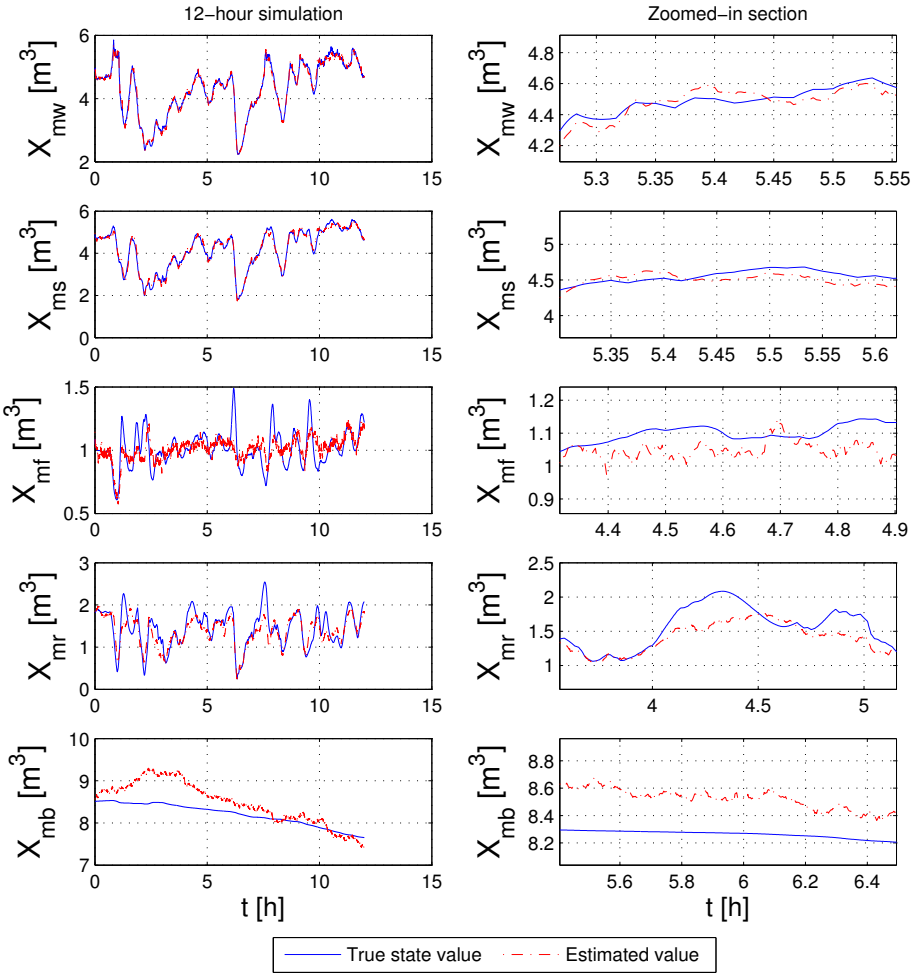

Figure 4.9: Mill state estimates.

does reach the upper and lower constraints, and THP and PSE do not reach set-point when this happens.

Fig. 4.12 represents a performance index that was used to compare the two simulations. The figure aims to illustrate the difference in error between using ideal state variables and using the state estimation proposed. The performance index is calculated based on (4.58) to (4.62). Equation (4.58) takes

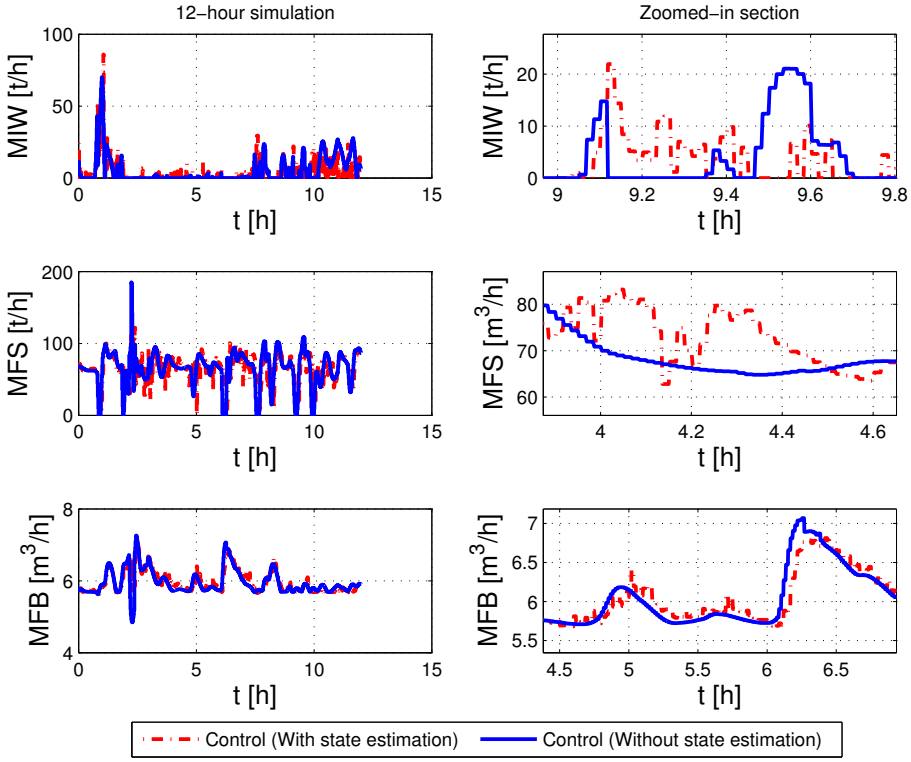

Figure 4.10: Manipulated variables.

a similar form as the objective function of the NMPC and uses similar weights to the NMPC.

$$J(\mathbf{u},\mathbf{x}_0,\mathbf{p}) = (\mathbf{y}-\mathbf{y}_{sp})^T\mathbf{Q}_a(\mathbf{y}-\mathbf{y}_{sp}) + \Delta\mathbf{u}^T\mathbf{Q}_b\Delta\mathbf{u} \tag{4.58}$$

$$\mathbf{u} = \{MIW, MFS, MFB, \alpha_{speed}, SFW, CFF\}^T \tag{4.59}$$

$$\mathbf{y} = \{LOAD, PSE, THP\}^T \tag{4.60}$$

$$\mathbf{Q}_a = \{1/0.05, 10/0.05, 10/0.5\} \tag{4.61}$$

$$\mathbf{Q}_b = \{0.1/20, 10/25, 10/2, 1/0.2, 0.1/50, 1/50\} \tag{4.62}$$

Fig. 4.12 shows that the proposed state estimation technique shows good overall control results. As expected, the error with state estimation is greater than when the ideal state values are used.

The nonlinear model predictive controller and state estimation was implemented using MATLAB R2012b. The controller formulation executed with an average time of 3.84 seconds and a maximum time of 7.02 seconds per iteration. The state estimation process executed with an average time of 0.345 seconds and a maximum of 0.59 seconds per iteration. An average time of 4.19 seconds is required to determine a single control move. This is promising when compared to the average computation

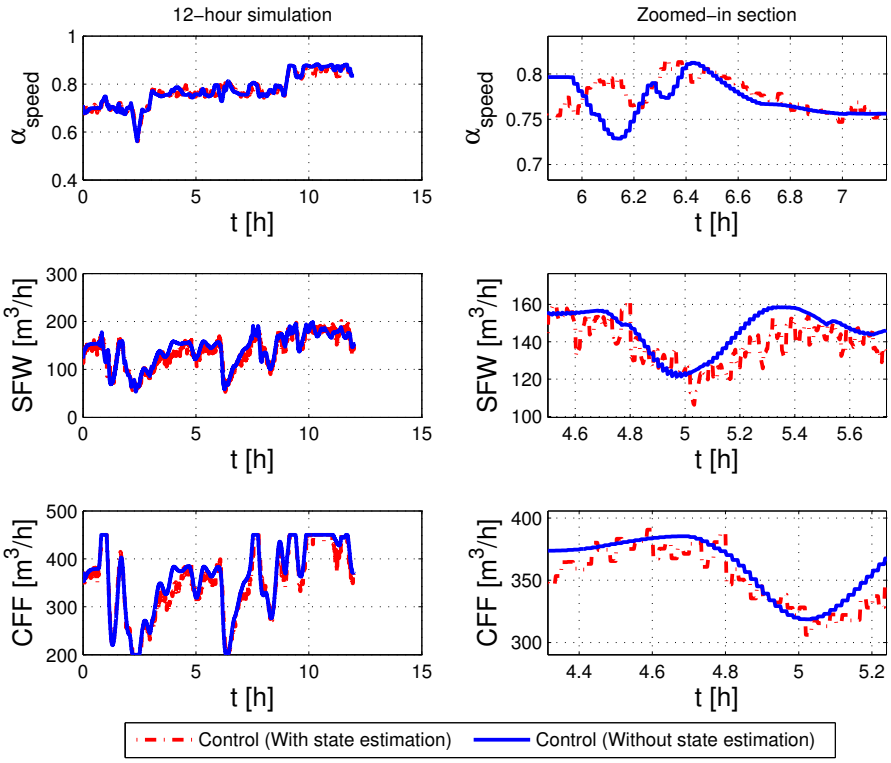

Figure 4.11: Manipulated variables.

time of about 26 seconds per iteration in Coetzee *et al.* (2010). However, in this study an Intel Core i7 2.7 GHz processor with 8 GB RAM was used and this is typically faster than the implementation platforms available on most mineral processing plants (Coetzee *et al.*, 2010).

4.4.4 Have the control objectives been met?

Objective 1

Robust tracking of set-points for *PSE* and *THP* is illustrated in Figures 4.5 and 4.6 respectively. Tracking was achieved despite the addition of significant noise and disturbances as described in section 4.4.2.

Objective 2

Good overall control has been achieved despite significantly varying the operation point as illustrated in Figures 4.5 and 4.6. Adequate state estimation was achieved for all eight states despite variations in the state variables as illustrated in Figures 4.8 and 4.9.

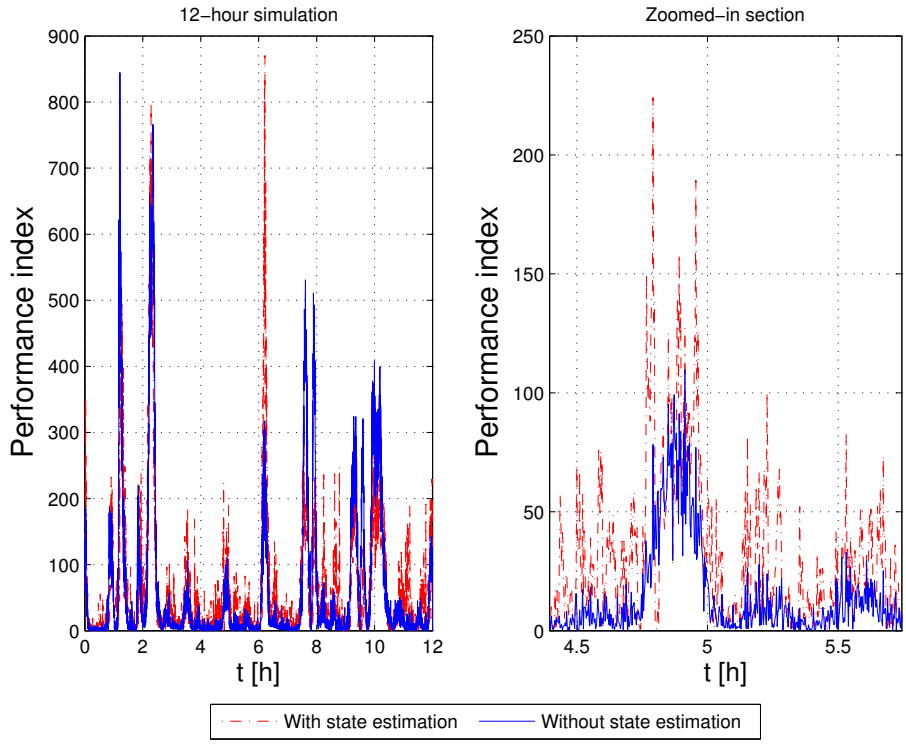

Figure 4.12: Performance index.

Objective 3

The sump volume has been successfully maintained within the sump limits as illustrated in Fig. 4.5. The water in the sump has been additionally optimized by a NMPC and DI controller as discussed in sections 4.2.3 and 4.2.4 and good control of sump volume and water is evident from Figures 4.5 and 4.7.

4.5 CONCLUSION

A combined nonlinear dynamic inversion and nonlinear model predictive control design approach is followed to propose a new effective control design philosophy for ROM grinding mill circuits, which meets the ambitious objectives of high product quality, high throughput and manageable loads in the mill. These objectives are an implicit result of good control of the milling circuit. The novel contribution of this work is the integration of a nonlinear controller and state estimator for the control of a ROM ore milling circuit. The results show that the particle size estimate (*PSE*) can be maintained adequately despite the introduction of significant noise and disturbances to the plant model. Six

parameters were constantly varied in a sinusoidal fashion with deviations of up to 50%. The sump volume and mill load where successfully maintained within the desired limits. All parameters as well as noise and disturbances were based on real plant data. Simulation results show that the proposed approach has the potential to successfully control a ROM milling circuit as only variables that can practically be measured are relied upon.

CHAPTER 5

POWER OPTIMIZATION

5.1 INTRODUCTION

The drive to reduce energy consumption is a growing interest because of the increase in the cost to produce energy. The cost of energy has increased significantly in recent years, and in South Africa an annual increase of 8 % is expected for the period 2013/2014 to 2017/2018 (Eskom, 2013). This increase in price greatly affects the mineral processing industry because of the large energy demands.

A run-of-mine (ROM) ore milling circuit provides a suitable case study where the power consumed by a mill is in the order of 2 MW. Grinding mill circuits have been identified as the most energy and cost intensive unit processes in the minerals processing industry (Wei and Craig, 2009) and hence energy savings can have a substantial impact. In a recent study (Matthews and Craig, 2013) a ROM ore milling circuit was power optimized using a time-of-use (TOU) tariff structure following the concept of demand side management. This study showed that power can be saved by implementing a real time optimizer (RTO) on a supervisory outer-loop level. For regulatory control, a linear model predictive controller (MPC) was used. This chapter, on the other hand, tries to save energy in the inner loop by additionally penalizing an energy factor as part of the cost function of the nonlinear model predictive control design (described in Chapter 4), and hence is fundamentally different from the RTO philosophy described in Matthews and Craig (2013).

In this chapter, an additional parallel milling circuit power optimization problem was researched. In the minerals processing industry a large amount of raw material is required to be processed and increasing production commonly requires a parallel mill configuration. Companies such as Anglo American Platinum operate with parallel milling circuits (Rule, 2011). Significant power is consumed

when running multiple milling circuits in parallel. This configuration yields an interesting power optimization problem due to the variability in a milling circuits power usage as a result from factors such as equipment degradation. Note that in this Chapter the parameter set from Coetzee *et al.* (2010) was used.

5.2 SINGLE ROM ORE MILLING CIRCUIT OPTIMIZATION

The aim of single ROM ore milling circuit optimization is to use the control configuration described in Section 4.2 with an additional power minimisation objective. The control system therefore needs to achieve a high throughput and maintain a high quality product while additionally aiming to minimise the power draw from the mill. Since the power draw from the mill is at least ten times more than any other unit in the milling circuit, the focus was predominately on the mill motor.

Nonlinear model predictive control allows for the set up of a unique objective function. In Chapter 4, it can be seen that the objective function (eq. (5.1)) was set up to minimise three expressions. The first expression ($\sum_{n=1}^{N_p}(\mathbf{y}-\mathbf{y}_{sp})^T\mathbf{Q}_1(\mathbf{y}-\mathbf{y}_{sp})$) aims to achieve set-point while the second and third expressions ($\sum_{n=1}^{N_p}\Delta\mathbf{y}^T\mathbf{Q}_2\Delta\mathbf{y}$ and $\sum_{n=1}^{N_c}\Delta\mathbf{u}^T\mathbf{Q}_3\Delta\mathbf{u}$) are used to minimise the movement of the controlled and manipulated variables. To minimise the power draw from the mill an additional power expression is added with results in the objective function:

$$J(\mathbf{u},\mathbf{x}_0,\mathbf{p}) = \sum_{n=1}^{N_p}(\mathbf{y}-\mathbf{y}_{sp})^T\mathbf{Q}_1(\mathbf{y}-\mathbf{y}_{sp}) + \sum_{n=1}^{N_p}\Delta\mathbf{y}^T\mathbf{Q}_2\Delta\mathbf{y} + \sum_{n=1}^{N_c}\Delta\mathbf{u}^T\mathbf{Q}_3\Delta\mathbf{u} + q_4\sum_{n=1}^{N_p}\frac{P_{mill}}{P_{max}}. \tag{5.1}$$

Note that the prediction horizon (N_p) equals 12 (two minutes). This value was chosen based on the work by (Coetzee *et al.*, 2010) where the study showed that it is possible to control the system with a prediction horizon of one minute and a sampling time of ten seconds. P_{mill} equals the power draw from the mill motor and P_{max} equals the maximum mill motor power draw.

5.2.1 Simulation setup

To illustrate the control capability and the effect of including a power factor in the nonlinear MPC objective function, an 8 h and 24 h simulation run was performed using the control formulation described in Section 4.2. The 8 h simulation run aims to illustrate the control performance despite disturbances and the 24 h simulation run aims to illustrate the control performance with various set-point changes to PSE and THP. Dynamic inversion and a nonlinear MPC was used to control

a nonlinear ROM ore milling circuit described in Section 4.2.2. However, in this chapter the DI controller and NMPC both have a sampling time of 10 seconds. State estimation is not the focus of this section and full state feedback is assumed. In practice, observers will have to be used as all states are not measured, see e.g. Chapters 3 and 4. Also, this study aims to show the accuracy and not the noise-handling capability of the controller. An additional constraint on α_{speed} was added to ensure smooth control: $\Delta\alpha_{speed} \leq 0.005$.

5.2.2 Control accuracy (8 h simulation)

Two input disturbances to SFW and MIW and one set-point change to THP was applied during the 8 h simulation. Additional spillage water was added to the sump between 2 h and 2.5 h with a magnitude of 30 m^3/h. The MIW was reduced by 10 m^3/h from 4 h to 4.5 h to simulate an input pipe leak. A THP set-point change was made at 6 h moving from 88 t/h to 90 t/h, PSE and $LOAD$ remained constant.

Fig. 5.1 illustrates how well the control configuration performs with disturbances and a set-point change. Note that the sump volume is within the desired limits. Fig. 5.2 illustrates how well the nonlinear dynamic inversion control technique operates. Fig. 5.3 illustrates resulting changes in additional important variables such as the power output, rheology factor and the cyclone feed density. All these outputs are within acceptable regions. Figures 5.4 and 5.5 show that the manipulated variables are all within constraints.

5.2.3 Power consumption comparison (24 h simulation)

Various set-point changes were made to the throughput (THP) and particle size (PSE) every 4 hours. The total charge of the mill ($LOAD$) was kept constant at 45% full. An energy factor was added to the nonlinear MPC to illustrate the additional possibilities that this control configuration possesses. The results show that the power from the mill is tightly linked with the circuit outputs i.e. changing P_{mill} will have a direct effect on whether the outputs reach set-point. The power consumed by the pump, after the sump, was regarded as negligible compared to the power consumed by the mill.

In the first run, the energy factor was not taken into account ($q_4 = 0$). Note that this run is regarded as the base case and that the energy savings have been compared to these results. The weight q_4 was introduced in the second simulation run with a starting value of 30. The error between the PSE

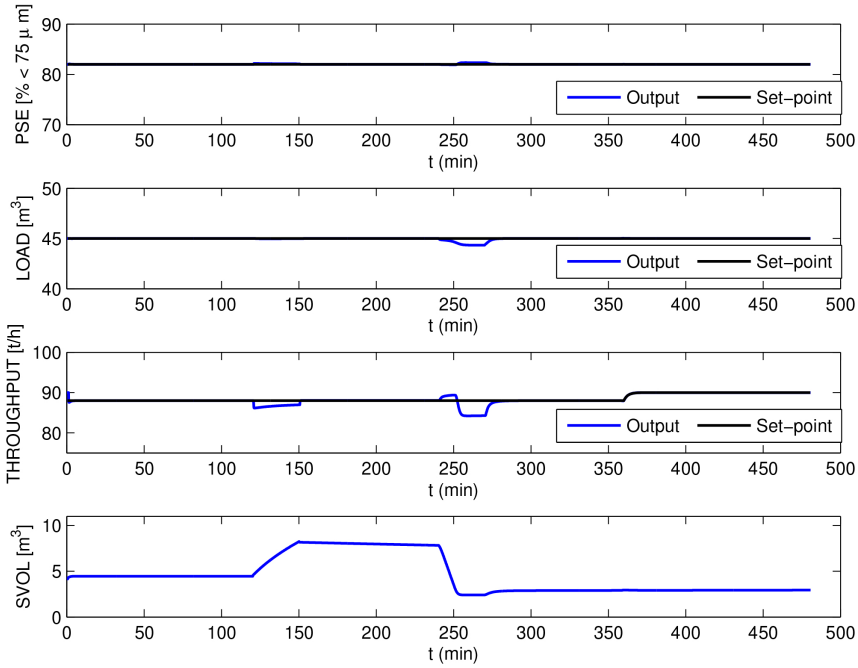

Figure 5.1: Output variables and set-point tracking

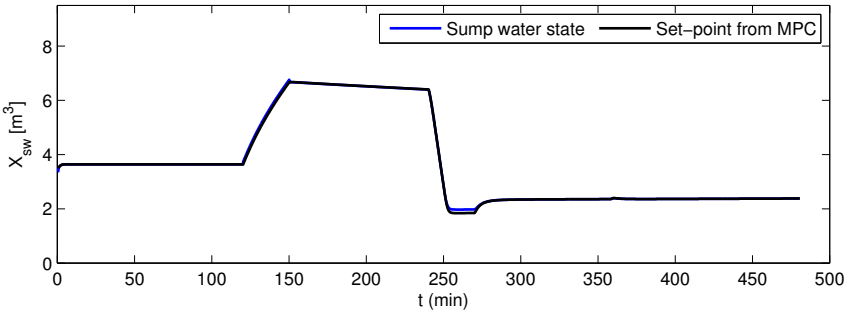

Figure 5.2: Dynamic inversion control. Desired and output value for the hold-up of sump water.

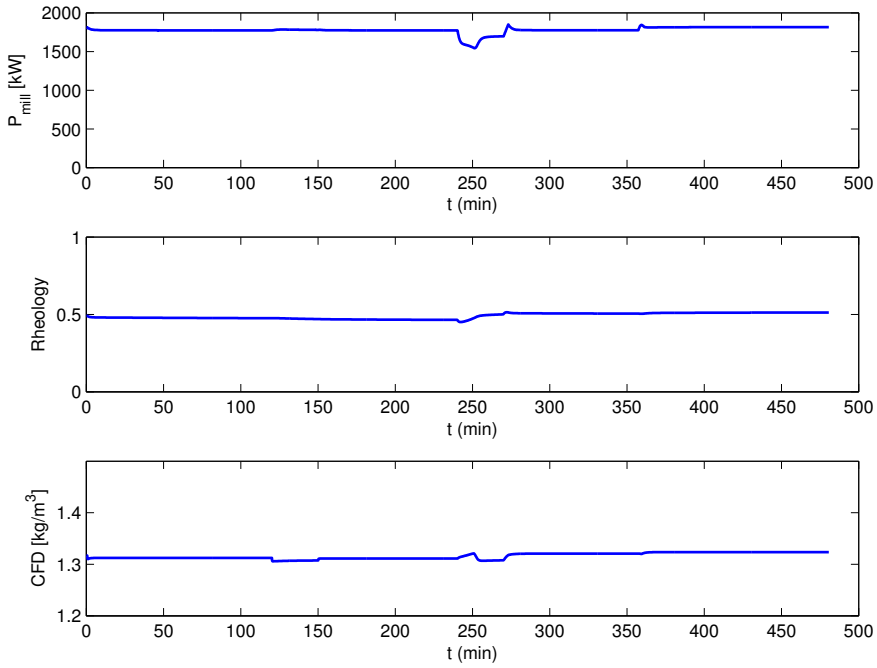

Figure 5.3: Important output variables for the milling circuit.

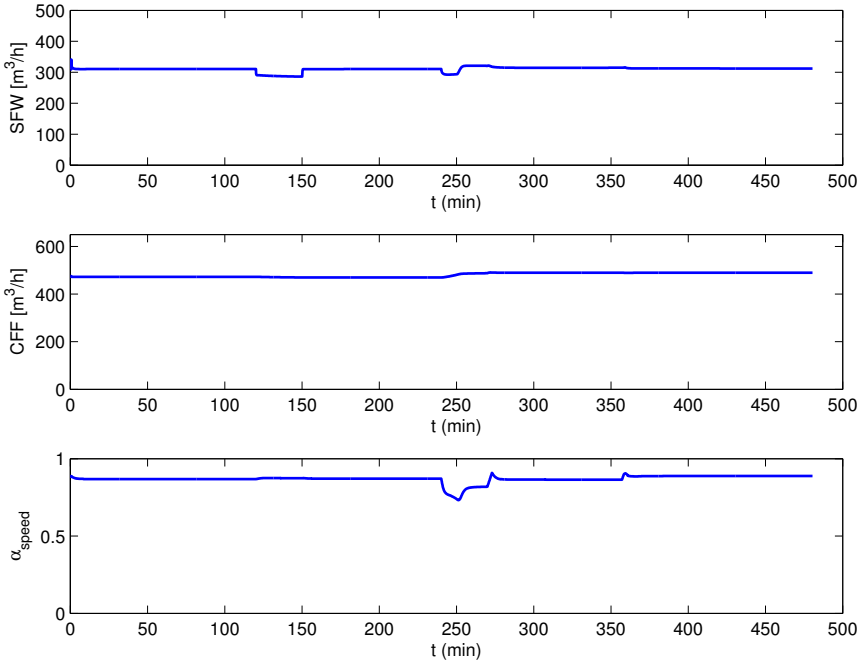

Figure 5.4: Manipulated variables.

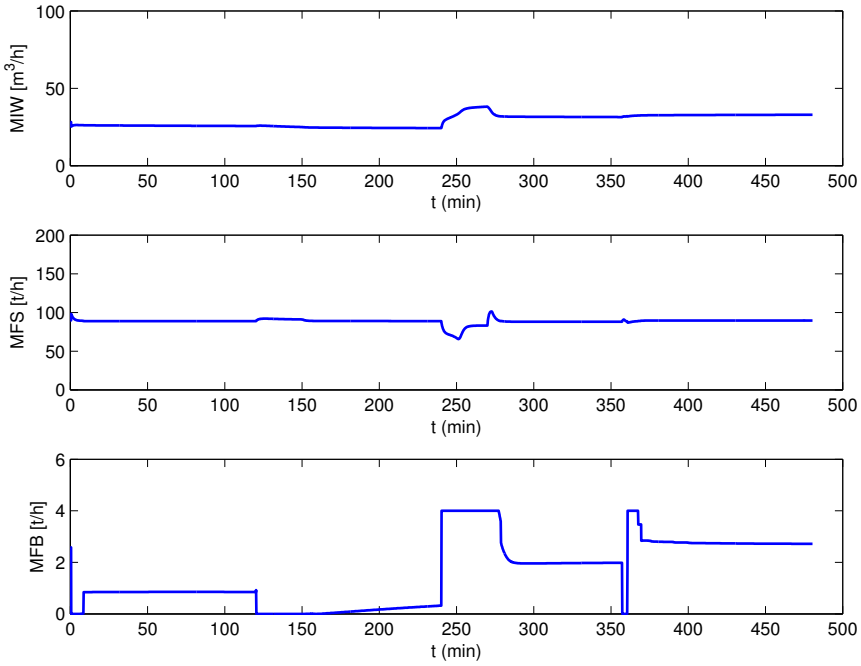

Figure 5.5: Manipulated variables.

measured value and the set-point value was determined every hour and if the error was larger than 0.1 %, the run was flagged as unsuccessful. Similarly for THP, an unsuccessful run resulted if the THP error was larger than 1 %. Once a run was flagged as unsuccessful, the run was restarted with a new q_4 value. The new q_4 value at the next step is 90 % of the current q_4 value.

The results show that in 24 h, and an energy term weight (q_4) of 18, a 332.7 kWh reduction in energy resulted with a 0.6% drop in THP and 0.1% drop in PSE. The simulation run with $q_4 = 18$ is regarded as the second and optimized run. Optimized in the sense that maximum power saving is achieved with a small effect on PSE and THP, $\leq 0.1\%$ and $< 1\%$ respectively. These results shows that the system is tightly integrated with power and that if the end result is to minimize power, there will, as expected, have to be some sacrifice in THP or PSE. The energy difference was calculated using the trapezoidal rule for the P_{mill} output variable.

Fig. 5.6 shows the difference in P_{mill} between the base case ($q_4 = 0$) and the minimized energy case ($q_4 = 18$). The spikes shown in Fig. 5.6 are mainly due to the mill motor speed during a set-point change. The second run showed that the controller slowly manipulated the mill motor speed to achieve set-point, the less aggressive movement resulted in the power difference. Fig. 5.7 illustrates

the difference in the additional outputs, rheology factor and cyclone feed density. The rheology factor shows very similar outputs while the density shows a slight change.

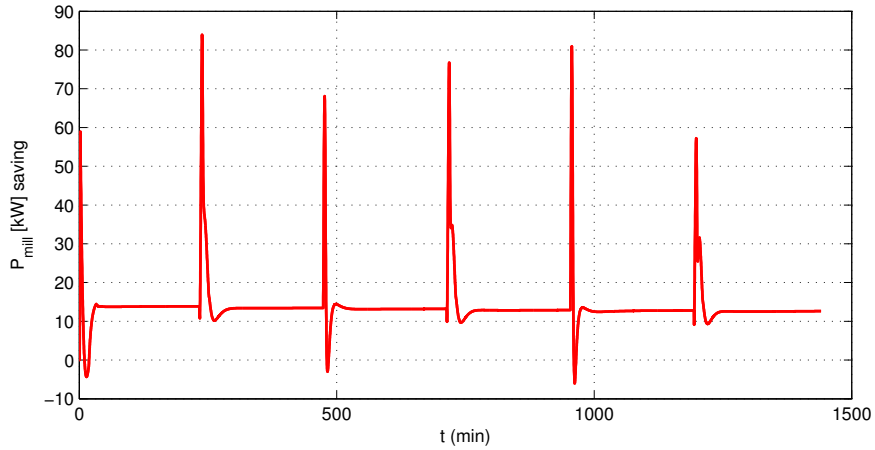

Figure 5.6: P_{mill} output difference between base case and optimized case in kW. Optimized case contains energy factor in NMPC objective function.

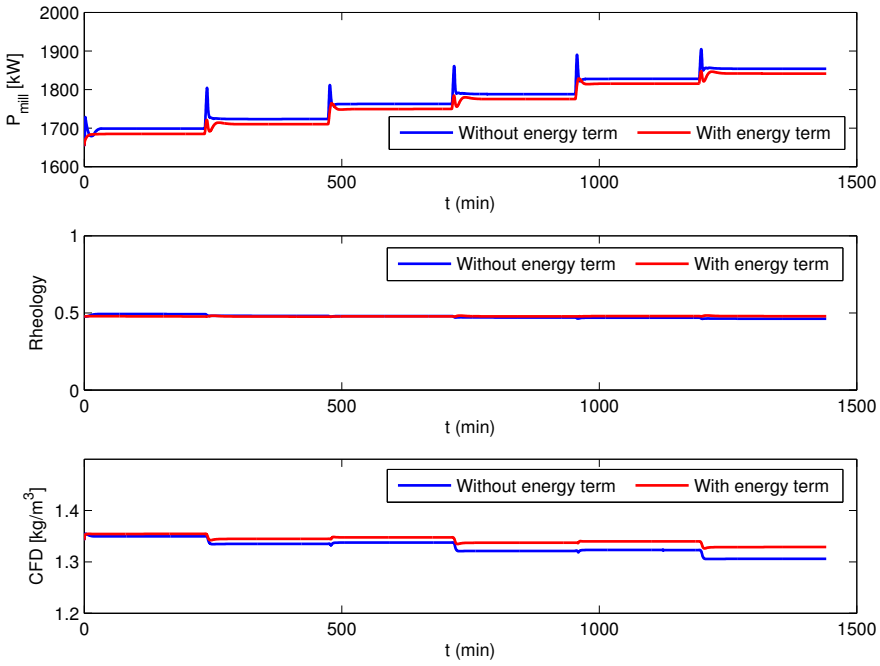

Figure 5.7: Output variables for base case and optimized case simulation.

PSE, *THP* and *LOAD* tracking is shown in Fig. 5.8. In the optimized run, the *LOAD* is around 47 m^3 (47% full). This result is expected as the P_{mill} value is reduced as the mill load moves away from

45% full. However, when looking at the parabolic mill power versus mill load curve it would be safer to operate at less than 45% full rather than close to the maximum constraint of 50%.

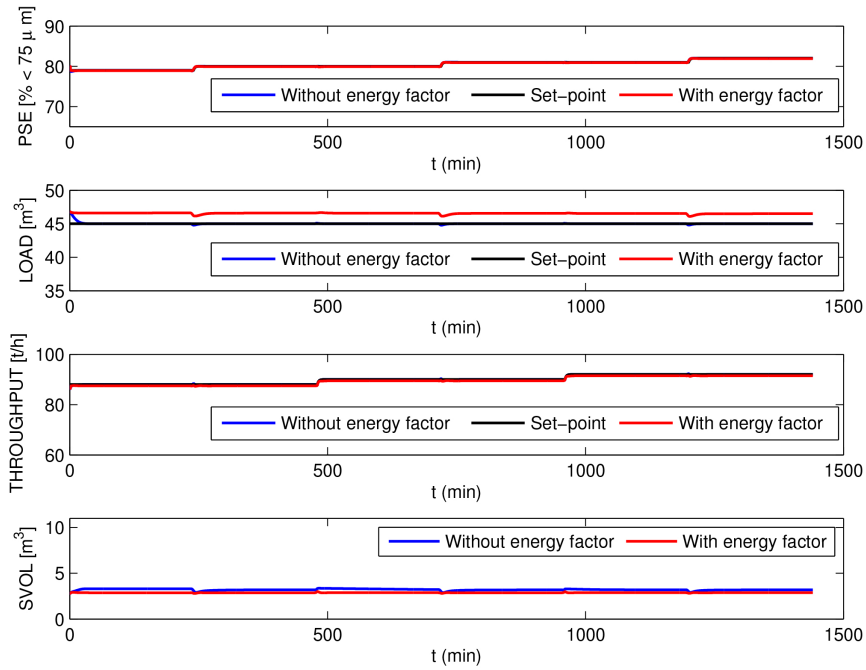

Figure 5.8: Controlled variables for base case and optimized case simulation.

Fig. 5.9 shows the dynamic inversion controller tracking the X_{sw} set-point from the NMPC. The energy factor run shows the effect of the convex combination (described in (4.30)). The water in the sump does not track the X_{sw} set-point from the NMPC perfectly because the controller is preventing the sump from running dry. Fig. 5.10 and 5.11 show that the manipulated variables are all within constraints.

Depending on the product processed and the type of mill used, the cost of electricity may be significantly less than the value of the product produced. However, as the cost of electricity increases the re-evaluation of cost-vs-profit will be worthwhile. The results show that this milling circuit model is close to power optimized when including α_{speed} as a manipulated variable.

The results show that to save energy the mill has to run between $45m^3$ and $50m^3$, shown in Fig. 5.8, and there should be no steel balls added to the mill, shown in Fig. 5.11. These conditions are undesirable as operating with a high mill load could result in a deteriorated breakage rate within the mill. Since no steel balls are added, the breakage rate could continue to deteriorate. The CFF is

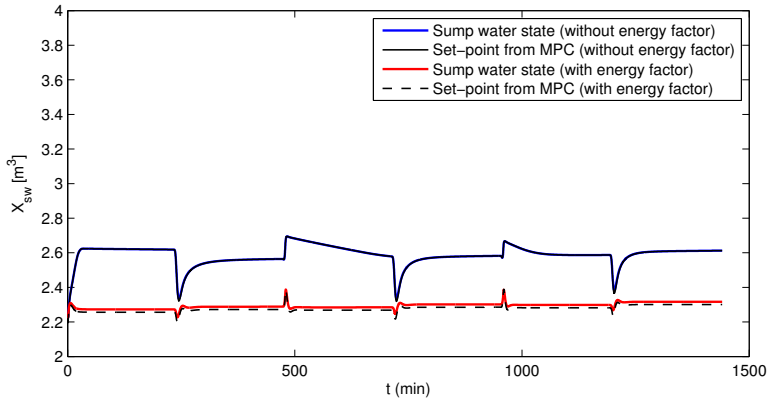

Figure 5.9: Dynamic inversion control for base case and optimized case.

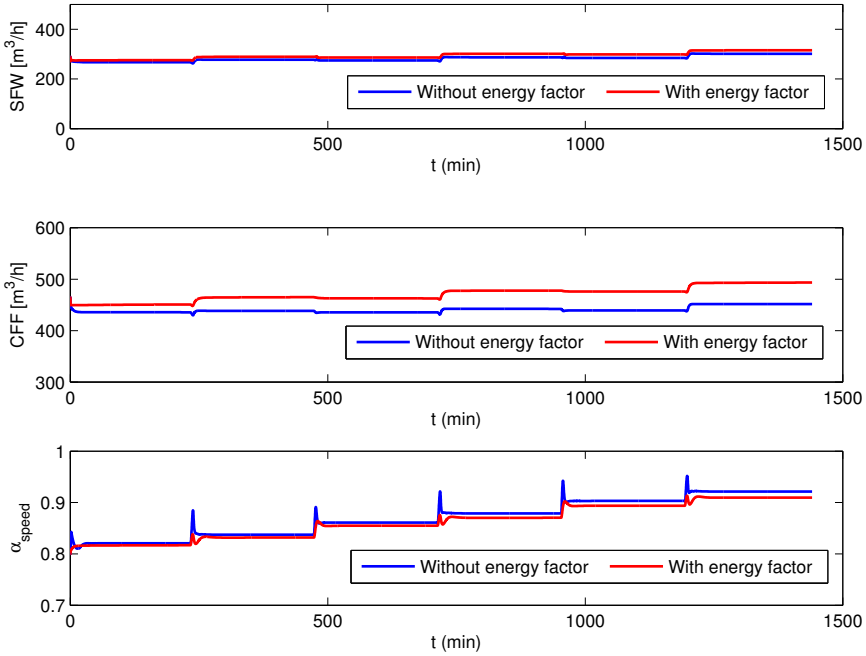

Figure 5.10: Manipulated variables for base case and optimized case simulation.

additionally very high, shown in Fig. 5.10, meaning that the sump pump is transferring as much as possible to the cyclone, this could be a result of the breakage within the mill.

An attempt has been made to overcome some of these issues by limiting the mill load below 45% in the NMPC. Additionally the weight of the mill load ($LOAD - LOAD_{sp}$) has been reduced from 1 to 0.01 in the NMPC. This aims to allow a larger movement of $LOAD$ to an energy optimized steady

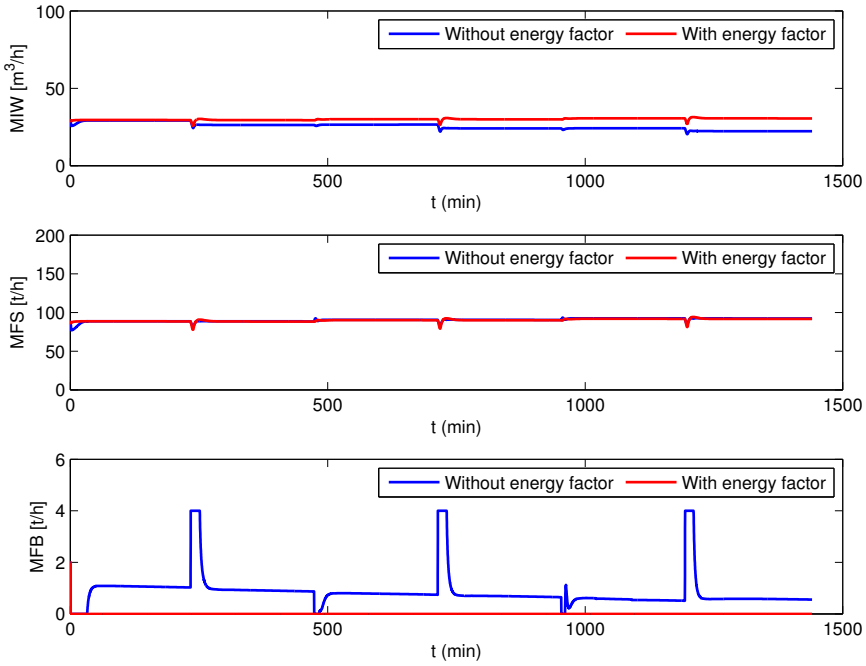

Figure 5.11: Manipulated variables for base case and optimized case simulation.

state value.

The results show that a 223.3 kWh reduction in energy resulted with a 0.4% drop in THP and 0.1% drop in PSE. This is 109.4 kWh less than initially with a 0.2% higher THP. The major difference is that the mill operates at $45m^3$ for both cases, shown in Fig. 5.14. This is a much better operating point than initially proposed at $47m^3$.

Fig. 5.12 shows a similar trend to Fig. 5.6 with spikes at set-point changes. Fig. 5.13 shows a similar trend to Fig. 5.7. PSE, $LOAD$ and THP are show good tracking despite the reduced $LOAD$ weight in the NMPC (Fig. 5.14). The DI controller, shown in Fig. 5.15, shows similar performance compared to Fig. 5.9.

The major difference in inputs can be seen in the steel balls rate, shown in Fig. 5.17, compared to Fig. 5.11. The addition of steel balls will assist with the breakage rate within the mill. However, the CFF, shown in Fig. 5.13, is still high and a concern as the pump is still operating close to the maximum limit.

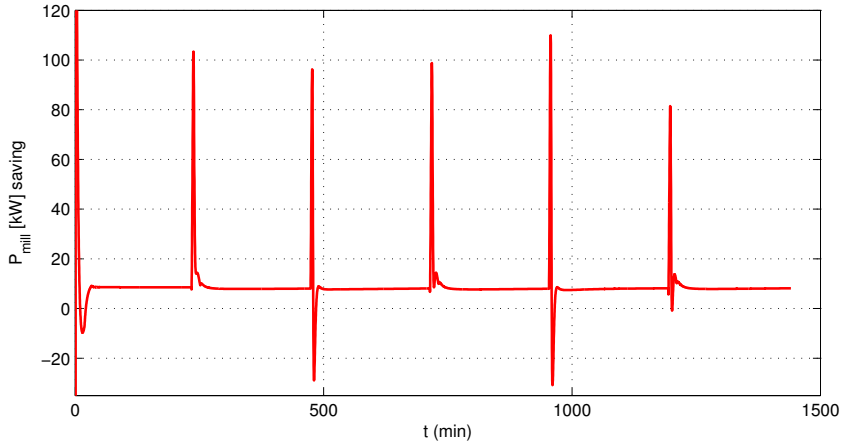

Figure 5.12: P_{mill} output difference between base case and optimized case in kW. Optimized case contains energy factor in NMPC objective function (*LOAD* constrained).

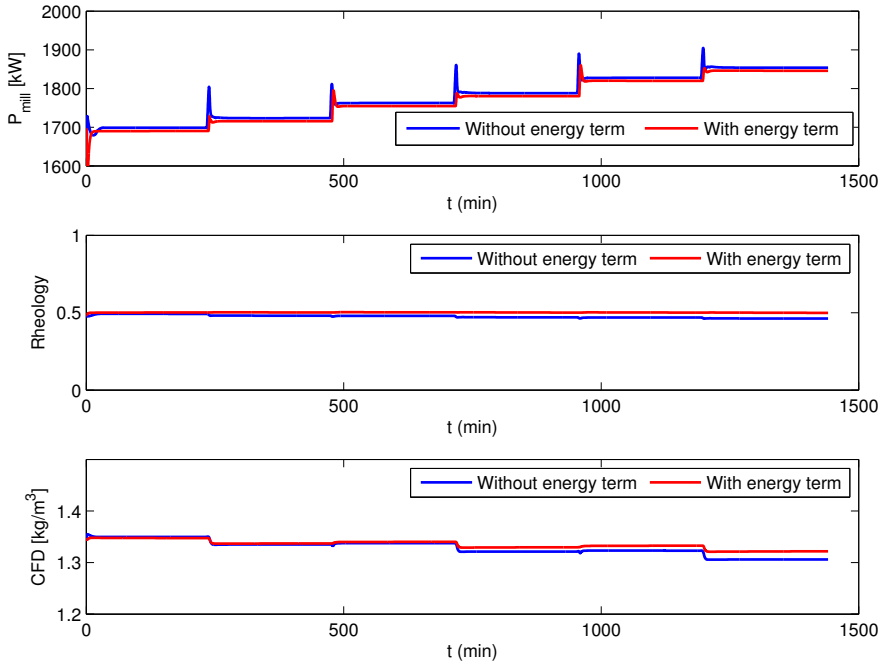

Figure 5.13: Output variables for base case and optimized case simulation (*LOAD* constrained).

The results from this section showed that energy can be saved in the inner loop but may result in an undesirable breakage rate within the mill. Additionally the cost of a higher CFF has to be taken into account. The next section targets power optimization when multiple mills are available, in an outer loop power optimizer.

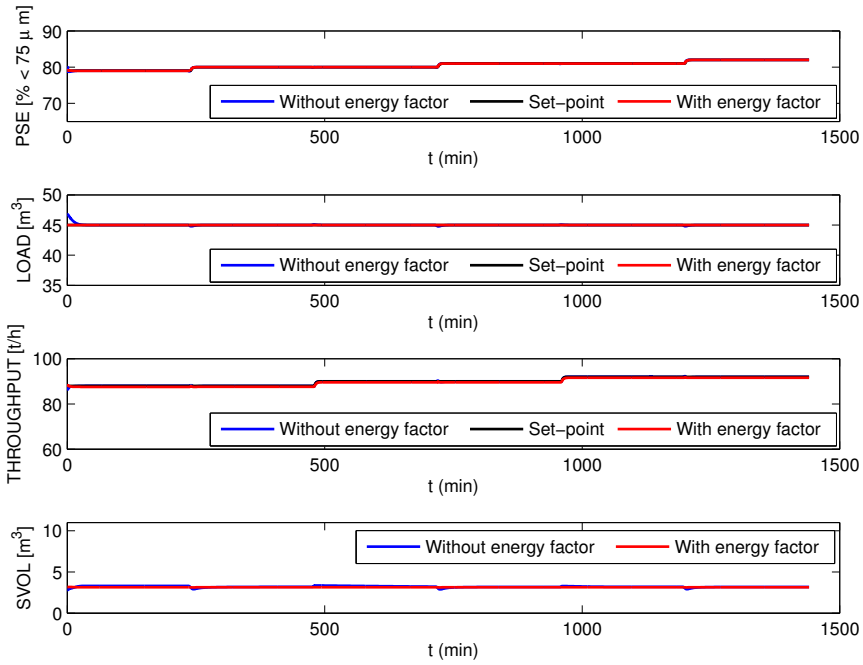

Figure 5.14: Controlled variables for base case and optimized case simulation (*LOAD* constrained).

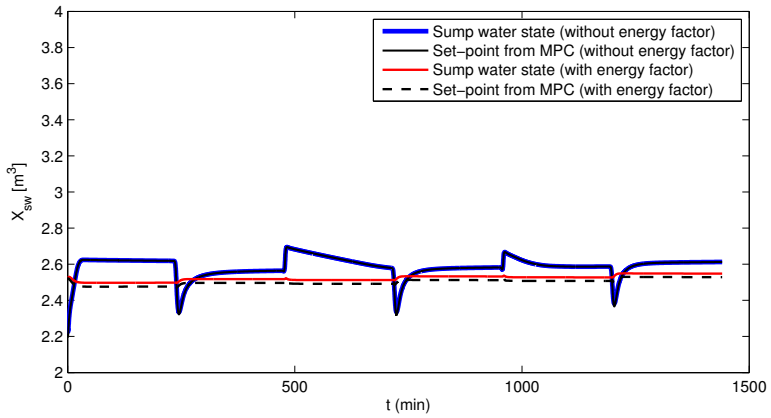

Figure 5.15: Dynamic inversion control for base case and optimized case (*LOAD* constrained).

5.3 MULTIPLE ROM ORE MILLING CIRCUIT OPTIMIZATION

This section considers minimizing the power consumption of two parallel ROM ore milling circuits. The aim of the power optimization is to come up with "dynamically changing set-point reference" for the operation of individual mills such that:

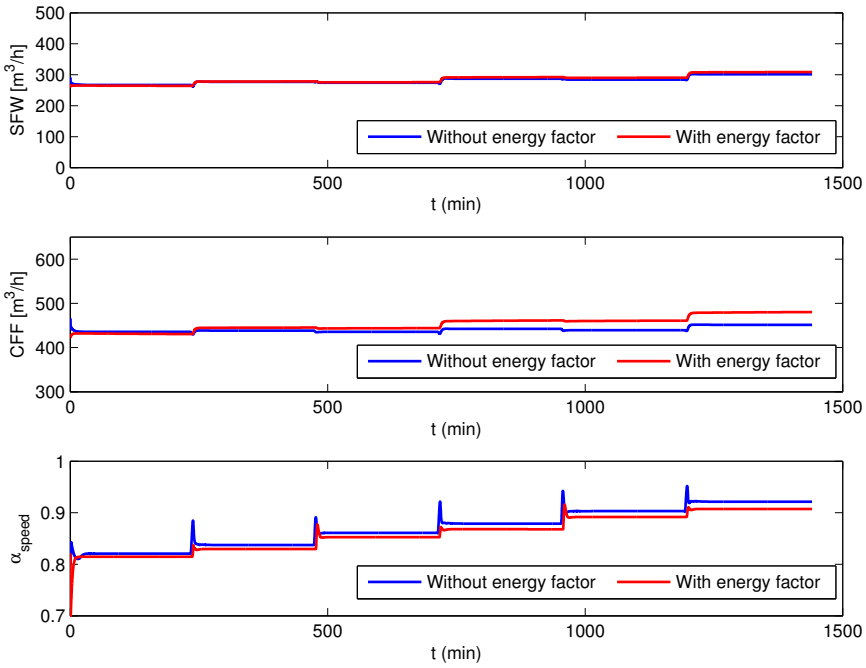

Figure 5.16: Manipulated variables for base case and optimized case simulation (*LOAD* constrained).

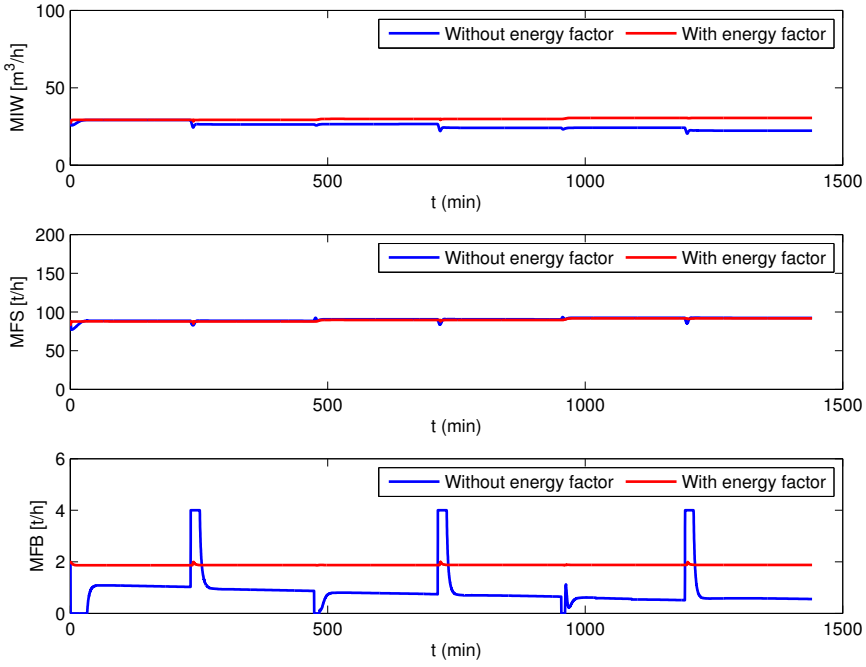

Figure 5.17: Manipulated variables for base case and optimized case simulation (*LOAD* constrained).

1. Total power usage should be minimized

2. Quality of product output (PSE) of individual mills must be maintained

3. Total throughput must remain same

4. Load in the mills should always be within the allowable limits

Fig. 5.18 illustrates the configuration that will be used to implement the power optimization. This figure shows that the complete control configuration presented in Chapter 4 was used to control each individual mill. The set-points, however, were determined by a power optimization block. The details of which will be covered in the sections that follow.

The first step in determining how to optimize two milling circuits for power is to determine the power consumption by each milling circuit. Note that this study only takes into account the mill motor power draw and not the power draw from any of the other modules. However, this method caters for the expansion to all other equipment. To determine the power draw from a milling circuit a neural network can be used. The neural network used the measurable outputs from the mill and determined the power consumption. This power estimation was used in the objective function of the set-point power optimizer block.

The $LOAD$ and THP set-point values are varied by the power optimizer. Since the aim is to maintain the product quality, each milling circuit receives the same PSE set-point. The aim for the sump volume ($SVOL$) is to be maintained within acceptable limits.

The power estimation that follows assumes that the milling circuit power consumption is comprised of only the mill motor draw. To expand the study to include all other components, the power measurement should be replaced with the summation of all the power draw components within the milling circuit. The assumption has been made since the mill motor power draw is at least ten times more than any other equipment in the milling circuit.

Two models for power estimation were implemented. Fig. 5.19 and Fig. 5.20 illustrate this concept. The first method consisted of using PSE, $LOAD$, THP, α_{speed} as inputs to a neural network and outputting mill power. While the second method contained the same inputs as well as α_{speed}, δ_{Pv}, δ_{Ps}, v_{Pmax} and φ_{Pmax}. This model was considered because these parameters directly affect the power consumption, as shown in eq. (2.52). The validation results showed that both methods predicted

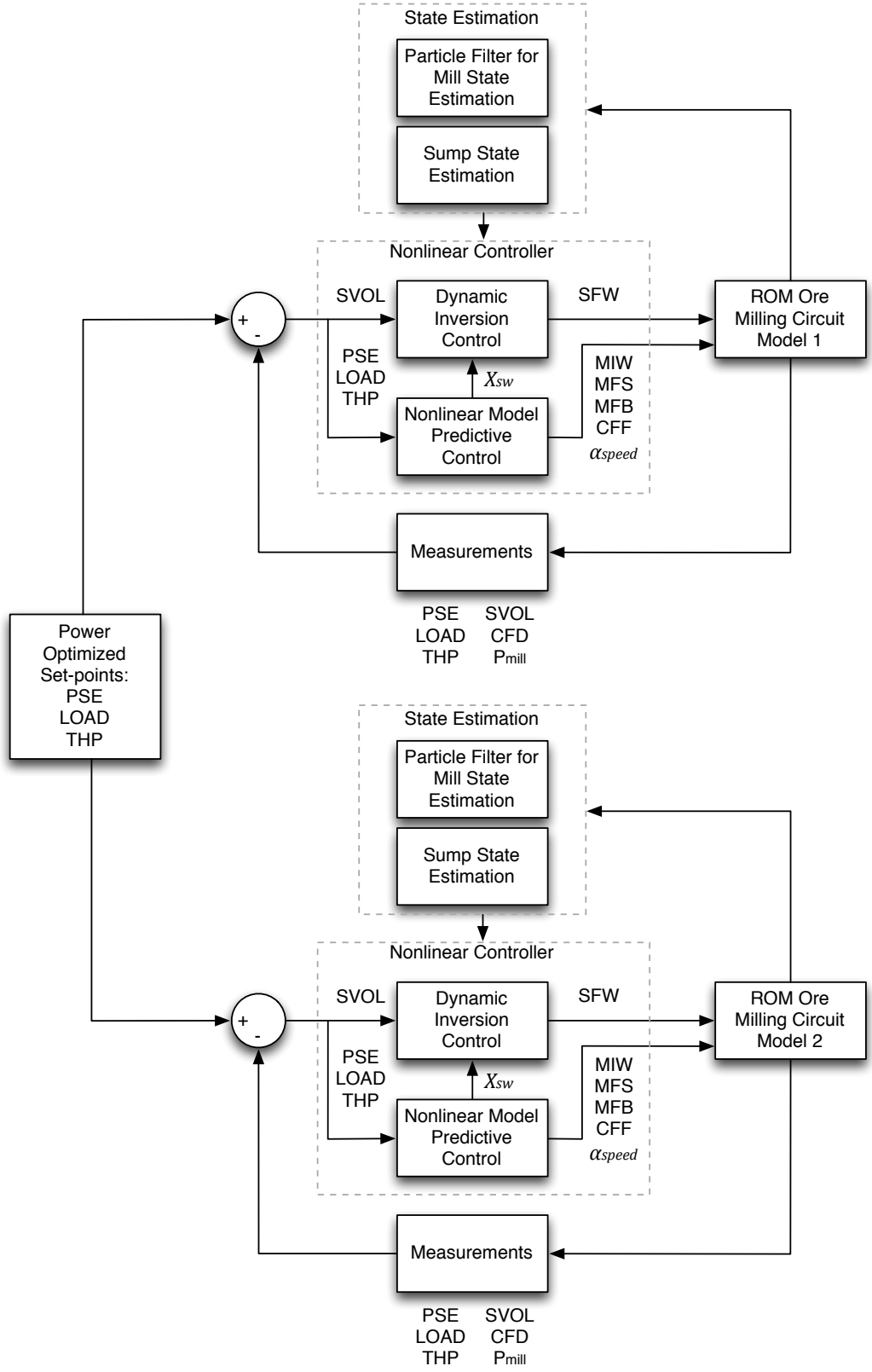

Figure 5.18: Power optimization complete control configuration.

the power very accurately. Therefore, only the first method was used as less information is required.

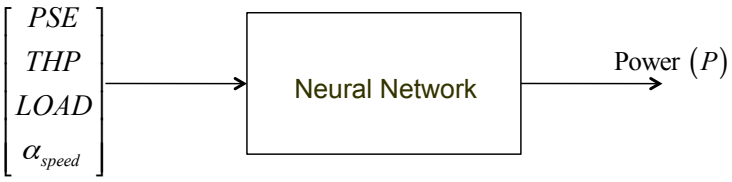

Figure 5.19: Formulation 1 using only measurable variables.

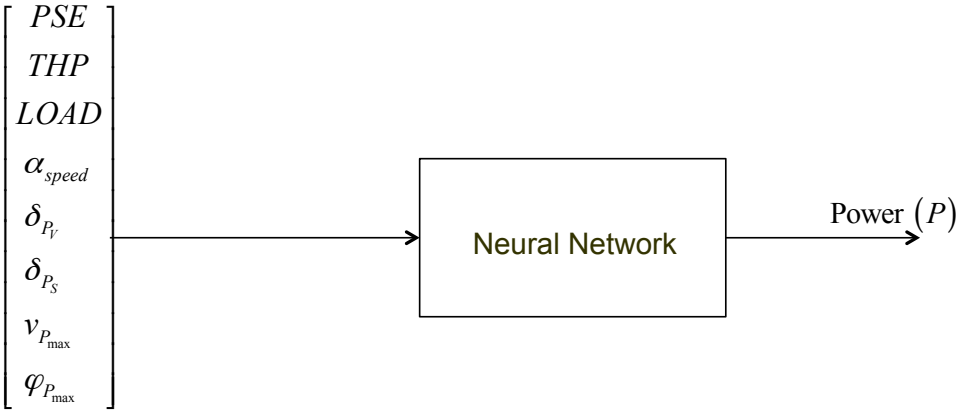

Figure 5.20: Formulation 2 using measurable variables and milling circuit parameters.

5.3.1 Neural network training

It is required as good practice and for good estimation to scale both the inputs and the outputs of the neural network. An additional requirement is to randomise the data set. This prevents that neural network from training the weights on cyclic data. The output of the network is scaled using the maximum power as shown in equation 5.2.

$$Output/Power(P) = \frac{P_{mill}}{P_{max}} \tag{5.2}$$

The inputs were normalised using the root mean square (rms) value as shown in equation 5.3.

$$x_{i_n} = \frac{x_i}{\overline{x_i}} \text{ where } \overline{x_i} = \frac{1}{N}\sqrt{\sum_j x_{ij}^2} \tag{5.3}$$

The network has the following structure: 4 X 6 X 6 X 1. The network used the 'tansig' basis functions for the hidden layers and the 'purelin' function for the output layer. The number of epochs was set

to 20. Fig. 5.21 shows the validation results for power estimation. From this figure it can be seen that the power estimation of P_{mill} can be accurately captured if PSE, $LOAD$, THP and α_{speed} are known.

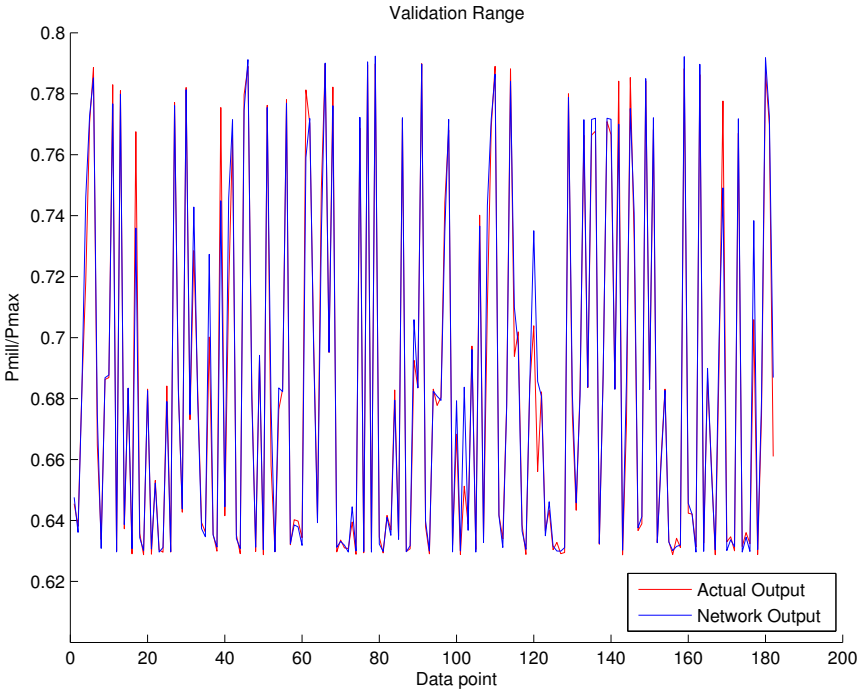

Figure 5.21: Validation of neural network with no noise

Fig. 5.22 shows the accuracy when the output variables are subjected to noise as in Section 4.4.2. The results show that power can be accurately estimated.

An attempt to estimate power with just PSE, $LOAD$ and THP was investigated. The results were unsatisfactory as the average error $(P_{mill}(measured) - P_{mill}(Estimated))$ was 50 kW (2.5 %). When including α_{speed} as an input to the neural network the average error was 7 kW (0.35 %).

Fig. 5.23 and 5.24 illustrate how different the power output changes as the THP and $LOAD$ vary. Also how different the power curves look with the following parameter value changes. These changes were implemented as ROM ore milling circuit 2.

1. α_f was increased by 10%

2. α_r was increased by 10%

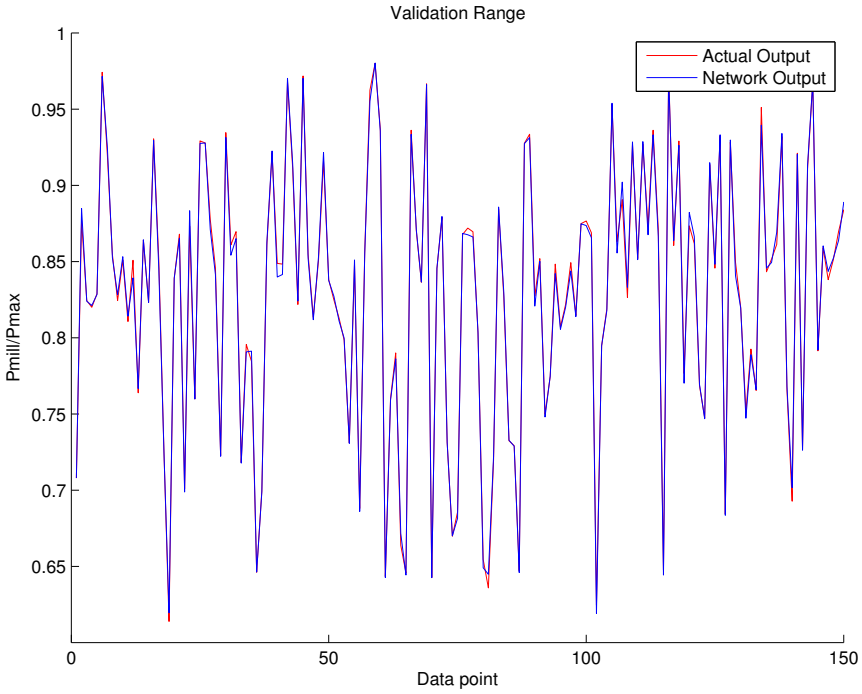

Figure 5.22: Validation of neural network with noise

3. ϕ_f was decreased by 10%

5.3.2 Power Optimization Formulation

A power formulation is applied to multiple mills as shown in (5.4) - (5.5). PSE_1 and PSE_2 have a desired value (80% less than 75 μm), shown as c_{11} and c_{21} respectively, as well as α_{speed1} and α_{speed2} are regarded as constants, shown as c_{12} and c_{22} respectively (because of the NMPC dependency). This leaves THP and $LOAD$ as variables for the power optimization. Therefore the power optimizer has four manipulated variables, as shown in (5.6) - (5.7), in a two parallel mill configuration. The neural network outputs $p1$ and $p2$ represent the power draw from the respective ROM ore milling circuit. The neural networks N_1 and N_2 have been trained using historical data where various step changes were made to $LOAD$ and THP.

$$\text{Plant 1} : p1 = N_1(PSE_1, THP_1, LOAD_1, \alpha_{speed1}) \tag{5.4}$$

$$\text{Plant 2} : p2 = N_2(PSE_2, THP_2, LOAD_2, \alpha_{speed2}) \tag{5.5}$$

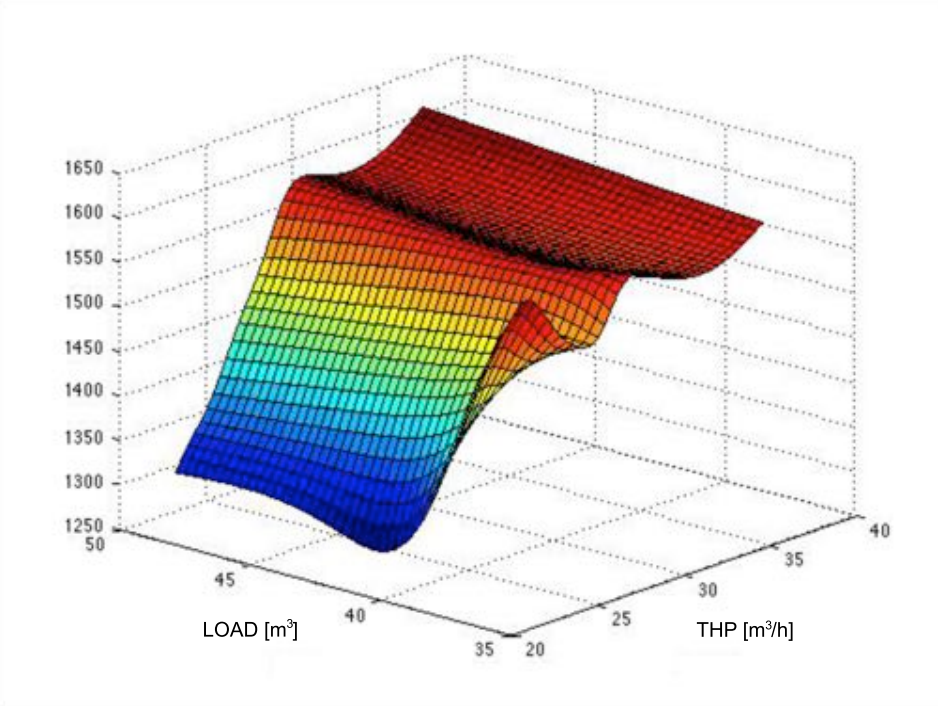

Figure 5.23: Surface plot of mill 1 power as a function of *LOAD* and *THP*.

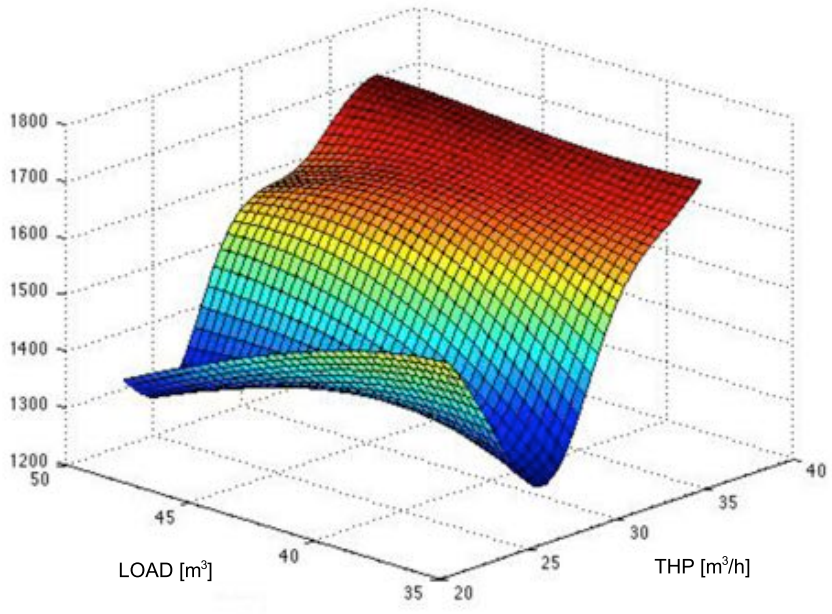

Figure 5.24: Surface plot of mill 2 power as a function of *LOAD* and *THP*.

The optimizer has a handle on THP and $LOAD$ for each milling circuit i.e. THP for circuit 1: u_{11}, THP for circuit 2: u_{21}, $LOAD$ for circuit 1: u_{12} and $LOAD$ for circuit 2: u_{22}.

$$\text{Plant 1} : p1 = N_1(c_{11}, u_{11}, u_{12}, c_{12}) \tag{5.6}$$

$$\text{Plant 2} : p2 = N_2(c_{21}, u_{21}, u_{22}, c_{22}) \tag{5.7}$$

5.3.2.1 Objective and constraint for optimizer

The main objective is the minimization of power consumption as shown in (5.8). The weights r_1 and r_2 are given a value of 1 but caters for further optimization. The weights could be used to further penalize a specific milling circuit.

$$\begin{aligned} J &= \frac{1}{2}\int_{t_0}^{t_f}(r_1 p_1^2 + r_2 p_2^2)dt \\ &= \frac{1}{2}\int_{t_0}^{t_f}(r_1 N_1(u_{11}, u_{12}) + r_2 N_2(u_{21}, u_{22}))dt \end{aligned} \tag{5.8}$$

An isoperimetric constraint on THP was implemented. An average of 90 t/h per mill was used to test this method i.e. two mills running for 8 hours have a desired throughput of 1440 tons. An 8 hour period was chosen as this is a typical period for the shift of a plant operator.

$$\int_{t_0}^{t_f}(THP_1 + THP_2)dt = C \text{ ((90 t/h)/mill)} \tag{5.9}$$

The outer loop formulation does not perturb the existing inner-loop control configuration (can be interpreted as 'guidance' of mills). This optimization is computationally simple as only one state equation has to be implemented (because of the isoperimetric constraint on throughput). However, success of the formulation critically depends on the successful training of the neural networks for a range of operation.

5.3.2.2 Cost function implementation

The cost function was formulated with the control (set-point) as piece-wise constant functions. This is illustrated in Fig. 5.25. Set-points for $LOAD$ and THP were divided into one hour intervals. This allowed for the output variables PSE, $LOAD$ and THP to reach set-point. Note that PSE was kept constant.

Equation (5.10) shows the cost function that was used in terms of piece-wise constant func-

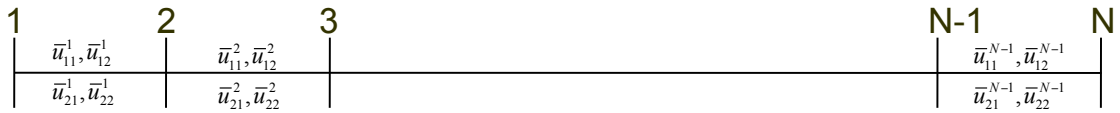

Figure 5.25: Piece-wise constant control

tions.

$$\begin{aligned} J &= \frac{1}{2}\int_{t_0}^{t_f} (r_1 N_1^2(u_{11},u_{12}) + r_2 N_2^2(u_{21},u_{22}))dt \\ &= \frac{1}{2}\int_{t_1}^{t_2} (r_1 N_1^2(\bar{u}_{11}^1,\bar{u}_{12}^1) + r_2 N_2^2(\bar{u}_{21}^1,\bar{u}_{22}^1))dt + ... \\ ...&\frac{1}{2}\int_{t_{N-1}}^{t_N} (r_1 N_1^2(\bar{u}_{11}^{N-1},\bar{u}_{12}^{N-1}) + r_2 N_2^2(\bar{u}_{21}^{N-1},\bar{u}_{22}^{N-1}))dt \\ &= \frac{\Delta T}{2}\left[\left(r_1 N_1^2(\bar{u}_{11}^1,\bar{u}_{12}^1) + r_2 N_2^2(\bar{u}_{21}^1,\bar{u}_{22}^1)\right) + ... \right. \\ &... + \left.\left(r_1 N_1^2(\bar{u}_{11}^{N-1},\bar{u}_{12}^{N-1}) + r_2 N_2^2(\bar{u}_{21}^{N-1},\bar{u}_{22}^{N-1})\right)\right] \end{aligned} \tag{5.10}$$

5.3.2.3 State constraint

As described earlier, the combined milling circuits need to produce a minimum throughput output over the working shift (8 hours). The throughput constraint was implemented as a state constraint. The state constraint was implemented as shown in (5.11).

$$\begin{aligned} \varphi(t_2) &= (\bar{u}_{11}^1 + \bar{u}_{21}^1)(t_2 - t_1) + \varphi(t_1) \\ \varphi(t_3) &= (\bar{u}_{11}^2 + \bar{u}_{21}^2)(t_3 - t_2) + \varphi(t_2) \\ \varphi(t_N) &= (\bar{u}_{11}^{N-1} + \bar{u}_{21}^{N-1})(t_N - t_{N-1}) + \varphi(t_2) \\ \varphi(t_N) &= \Delta T\left[(\bar{u}_{11}^1 + \bar{u}_{21}^1) + ... + (\bar{u}_{11}^{N-1} + \bar{u}_{21}^{N-1})\right] = C \\ &\left[(\bar{u}_{11}^1 + \bar{u}_{21}^1) + ... + (\bar{u}_{11}^{N-1} + \bar{u}_{21}^{N-1})\right] = \frac{C}{\Delta T} \end{aligned} \tag{5.11}$$

5.3.2.4 Position Constraints

The *THP* and *LOAD* set-points have constraints that are dependant on the physicals limits of the milling circuit. This is enforced using (5.12). A feasibility study was completed to determine the physical bounds of each milling circuit.

$$u^{min} \leq \bar{u}^1, \bar{u}^2, ..., \bar{u}^{N-1} \leq u^{max} \tag{5.12}$$

$$THP_{min} < u_{11}, u_{21} < THP_{max} \tag{5.13}$$

$$LOAD_{min} < u_{12}, u_{22} < LOAD_{max} \tag{5.14}$$

5.3.2.4.1 Feasibility Study The aim of the feasibility study was to determine the range of operation for each of the milling circuits in terms of $LOAD$ and THP variation.

After several trial simulations, it was noticed that at various $LOAD$ and THP combinations the system would not need reach set-point as the inputs to the milling circuit would saturate. This was a concern as the α_{speed} values, used in the neural network, had to be realistic unsaturated values. The study would allow the power optimization to extend the position constraints as much as possible. A study was conducted to determine where each milling circuit will reach set-point (shown as *) within 20 minutes subject to the following criteria.

1. $|PSE - PSE^*| < 1\ [\% < 75\mu m]$
2. $|LOAD - LOAD^*| < 1\ [m^3]$
3. $|THP - THP^*| < 5\ [t/h]$

The following jump constraints were implemented during the study (maximum step size change).

1. 5% of total load volume
2. 5% of average throughput

Fig. 5.26 and 5.27 illustrate the successful regions of operation for mill 1 and mill 2 as a function of $LOAD$ and THP. The red points indicate that the set-points were successfully reached. It should be noted that mill 2 has a significantly higher THP range compared to mill 1.

From the above study, the new constraints were determined as shown below. These are shown as lines on Fig. 5.26 and 5.27.

1. ROM ore milling circuit 1 constraints:
   (a) $35 < LOAD < 50$

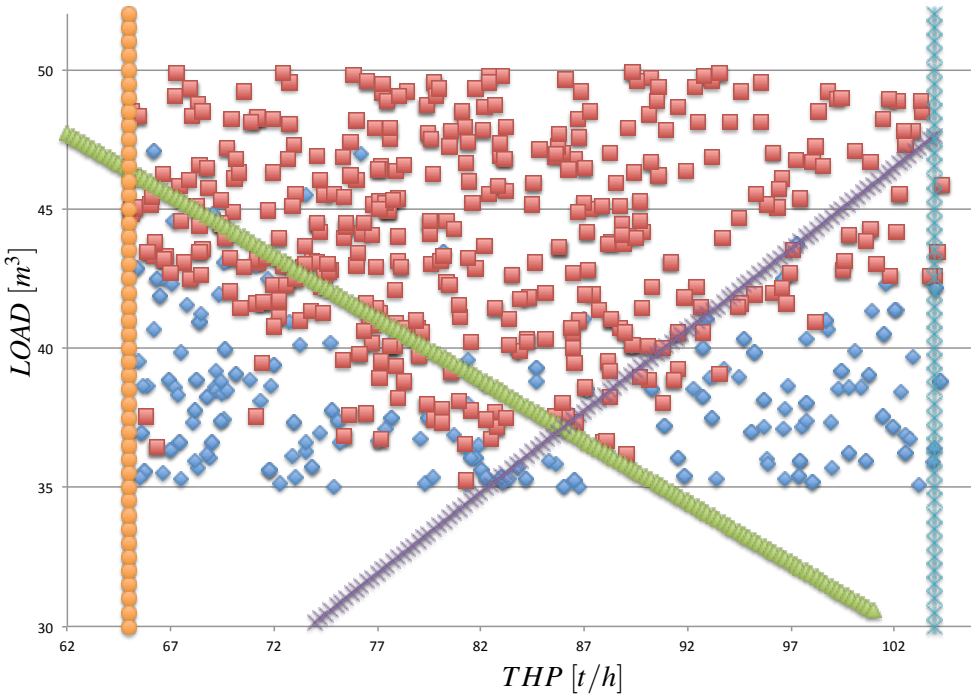

Figure 5.26: Feasibility study for mill 1.

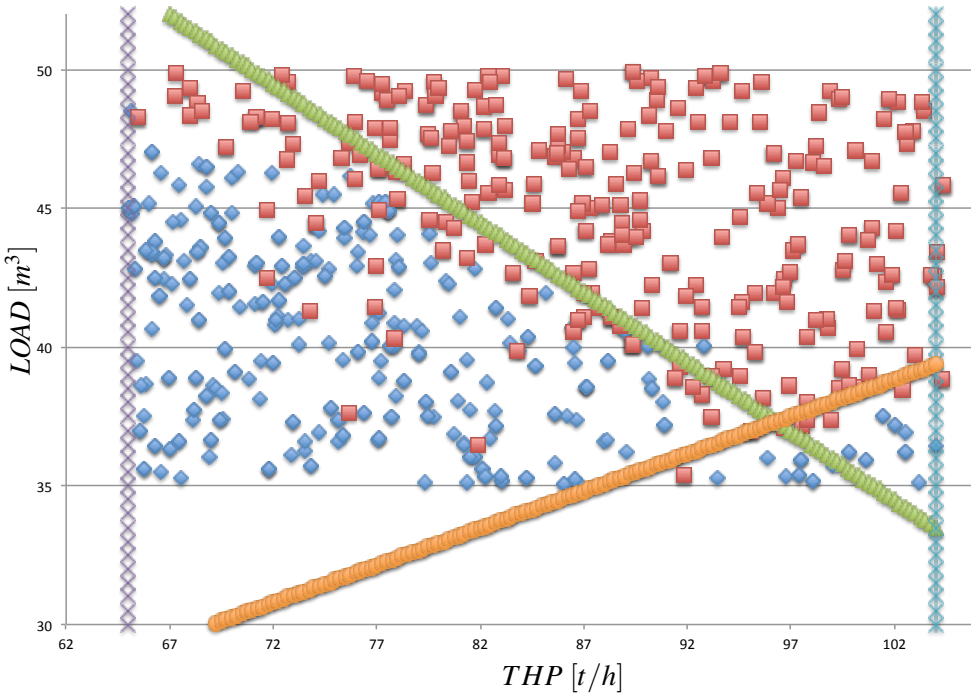

Figure 5.27: Feasibility study for mill 2

(b) $65 < THP < 105$

(c) -0.44*(THP)+74 $< LOAD$

(d) 0.583*(THP) - 15.58 < $LOAD$

2. ROM ore milling circuit 2 constraints:

(a) 35 < $LOAD$ < 50

(b) 70 < THP < 118

(c) -0.55*(THP)+88.06 < $LOAD$

(d) 0.268*(THP) + 10.15 < $LOAD$

5.3.2.5 Jump Constraints

Jump constraints restrict the movement of the set-point values (such as $LOAD$) as it is undesirable for the mill load to change by large amounts every hour. Likewise THP cannot change rapidly ensuring smooth operation.

$$\begin{aligned} \hat{u}_{12}^{min} \leq \left(\overline{u}_{12}^{i+1} - \overline{u}_{12}^{i}\right) \leq \hat{u}_{12}^{max} \\ \hat{u}_{22}^{min} \leq \left(\overline{u}_{22}^{i+1} - \overline{u}_{22}^{i}\right) \leq \hat{u}_{22}^{max} \\ \text{for } i = 1, 2, ..., (N-2) \end{aligned} \tag{5.15}$$

5.3.3 Implementation

The difficulty with the power optimization formulation is that the outer loop (set-point optimizer) is dependant on α_{speed} (which is a manipulated variable in the Chapter 4 control configuration). This is a problem because the optimization of power is based on a α_{speed} value (shown in (5.4)). The ideal scenario would be to set α_{speed} from the power optimizer ensuring that the power estimate and optimization is accurate. However, because of the inner loop objective to control both product quality and quantity this is not possible. Numerous attempts have been made to remove α_{speed} from the power optimizer loop. In most of the attempted scenarios, CFF would saturate and not be able to control the two conflicting output variables.

An implementation process was developed to try and work with the α_{speed} issue. Fig. 5.28 shows this implementation process and the inner and outer loop link. In the simulation, two mills run in parallel and have a desired THP value at the end of an eight hour shift. The quality of the product (PSE) must

remain at 80% throughout the shift. The desired throughput value results in an average of 180 t/h for both mills. The initial guess is to split this equally at 90 t/h for the shift. The process continues as follows.

1. Run eight hour simulation with all manipulated variables in the inner loop control configuration. The objective of the inner loop is to reach set-point. Dynamic inversion control configuration remains the same throughout optimization process. Once the simulation is completed, extract the eight α^1_{speed} values at i+45 min (where i=0,1,...,7) intervals. After an eight hour period, a vector containing both mills speed at the sample times specified is then passed to the outer loop optimizer.

2. The power optimization formulation operates using the α^1_{speed} values extracted and PSE at 80% for both milling circuits. The output of the formulation results in $LOAD$ and THP set points for the two ROM ore milling circuits.

3. An eight hour simulation is run with the new set points. The simulation set-up is that same as in step 1. The power and the throughput values are determined and compared to the values in the initial run. The eight α^2_{speed} values were extracted.

4. It was noticed that the α^2_{speed} values that resulted from the second, power optimized, run were not the same as the α^1_{speed} values initially extracted. This is intuitively correct as the milling circuit that consumed less power initially is then pushed to produce the majority of the throughput. Therefore a way to make the two different α_{speed} values converge had to be determined. Convergence would mean that the power optimizer is accurately predicting the power draw from the mills and therefore an optimal solution can be achieved. The first α_{speed} value is a result of initial set-points given and the second α_{speed} value is a result of new set-points from the power optimizer. Using the convex approach, an α^{new}_{speed} was determined using equation (5.16).

$$\begin{aligned} \alpha^{new}_{speed} &= \tau\alpha^1_{speed} + (1-\tau)\alpha^2_{speed} \\ \tau &= \pm 0.05 \end{aligned} \tag{5.16}$$

5. Another eight hour simulation was run with α_{speed} removed as a manipulated variable. The combined α^{new}_{speed} vector was used to set α_{speed} as a constant in this simulation. A feasibility check is completed at i+45 min (where i=0,1,...,7) intervals to determine if the circuit reaches the desired set-points using only the remaining manipulated variables. If unsuccessful then step

1 is repeated with $\tau = 0$ (i.e. α^2_{speed} values) else step 1 is repeated with α^{new}_{speed} values.

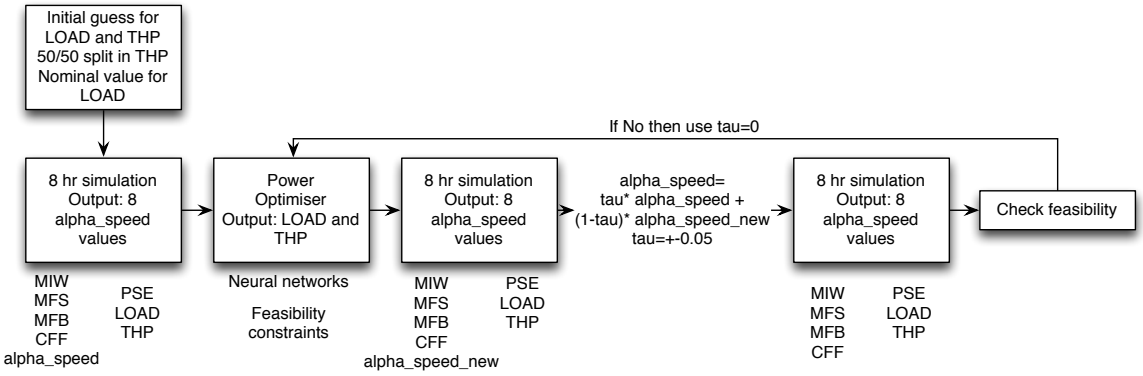

Figure 5.28: Block diagram illustrating inner and outer loop link.

5.3.4 Results

Fig. 5.29 shows the initial THP set-points and the set-points from the power optimizer. Initially the THP set-points were set to 33.33 m^3/h or 90 t/h for the eight hour shift for both ROM ore milling circuits. After the power optimizer, it can be seen that the throughput was decreased in milling circuit 1 (shown in blue) while the throughput was increased in milling circuit 2 (shown in green). This would indicate the milling circuit 2 consumes less power than milling circuit 1. This is a good result considering that in the milling circuit 2 the parameter ϕ_f (power needed per ton of fines produced) was decreased by 10 %.

Fig 5.30 shows the difference between the initial $LOAD$ set-points and the optimized $LOAD$ set-points. The variation in $LOAD$ set-points are due to the position constraints ensuring that each mill will reach set-point. These results are also positive, from (2.22)-(2.23) it can be seen that a $LOAD$ set-point of 45 m^3 will maximize the power draw. The optimizer aims to select a $LOAD$ set-point with the maximum deviation from 45 m^3 however, must still remain in the feasible region for the respective milling circuit.

The outputs of the two ROM ore milling circuits are shown in Fig. 5.31 and Fig. 5.32. Note that the initial conditions used were varied and therefore an initial jump is expected. These figures illustrate the variation in $LOAD$ and THP set-points. The set-point filter assists with the outputs reaching set-point quickly. The PSE is successfully maintained at 80 % $< 75\mu m$ for both milling circuit throughout the eight hour shift. Minor deviations can be seen to PSE every hour when the $LOAD$ and

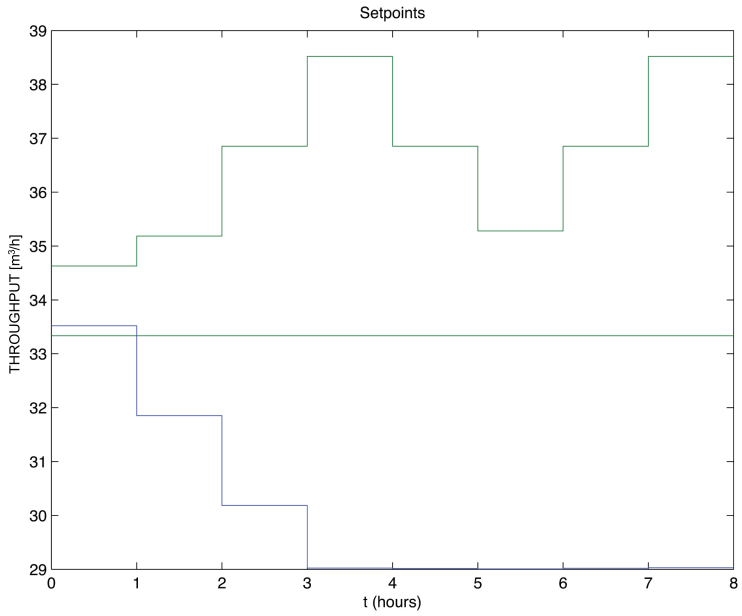

Figure 5.29: *THP* set-point changes for milling circuits obtained from the power optimizer. Milling circuit 1 is shown in blue and milling circuit 2 is shown in green.

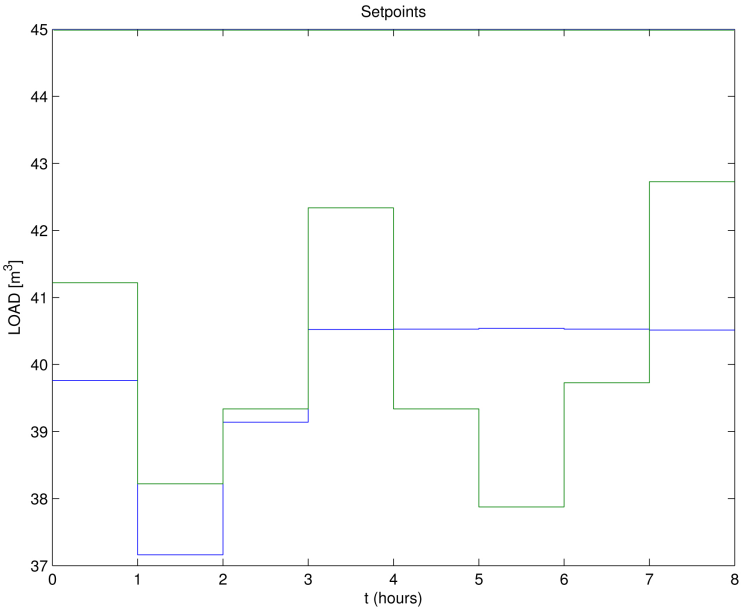

Figure 5.30: *LOAD* set-point changes for milling circuits obtained from the power optimizer. Milling circuit 1 is shown in blue and milling circuit 2 is shown in green.

THP change.

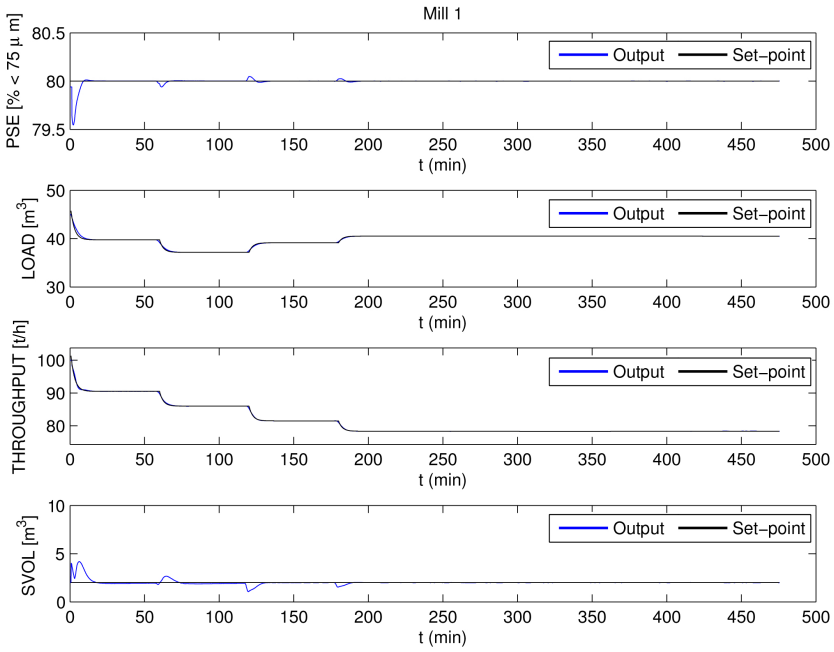

Figure 5.31: Output variables for milling circuit 1, note the *LOAD* and *THP* set-point changes.

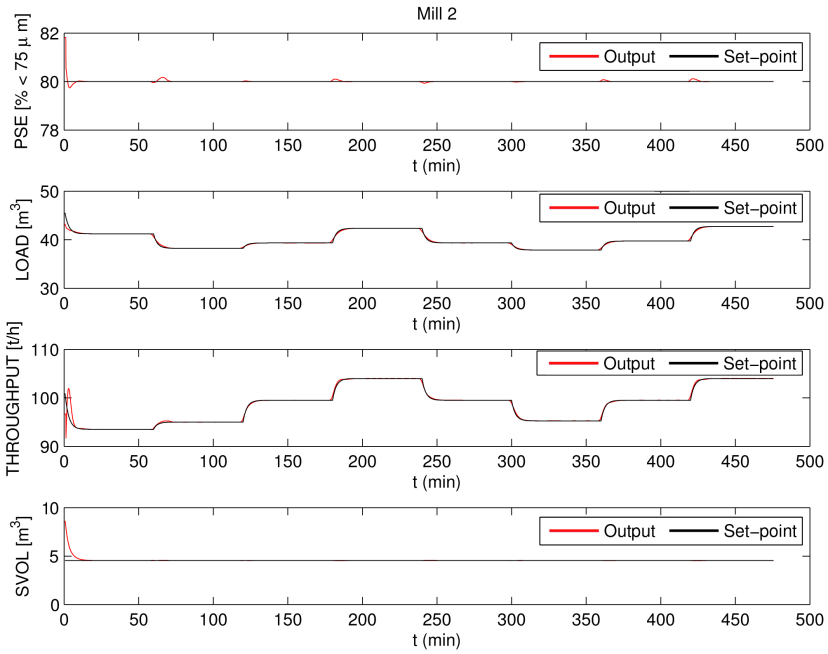

Figure 5.32: Output variables for milling circuit 2, note the *LOAD* and *THP* set-point changes.

The manipulated variables for the two milling circuits are shown in Fig. 5.33 and Fig. 5.34 for the respective ROM ore milling circuits. All the manipulated variables are within the constraints.

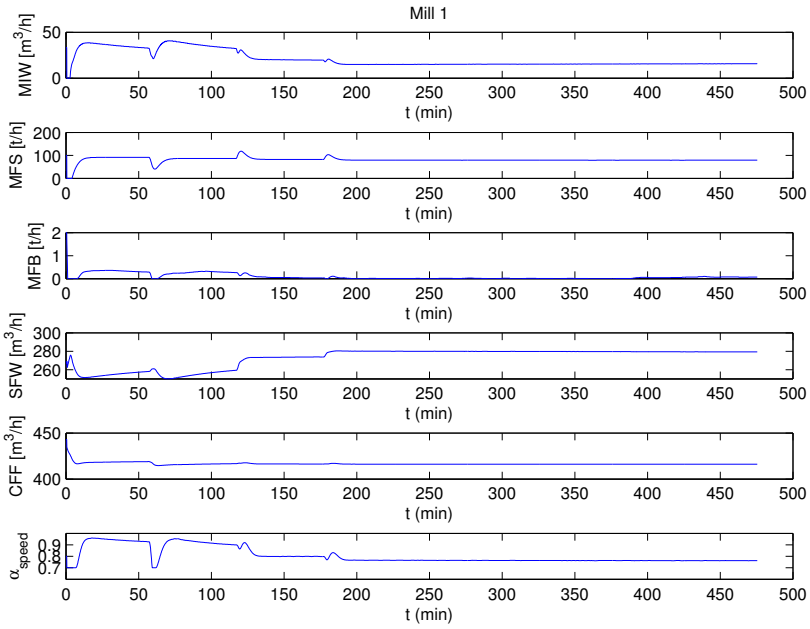

Figure 5.33: Manipulated variables for milling circuit 1.

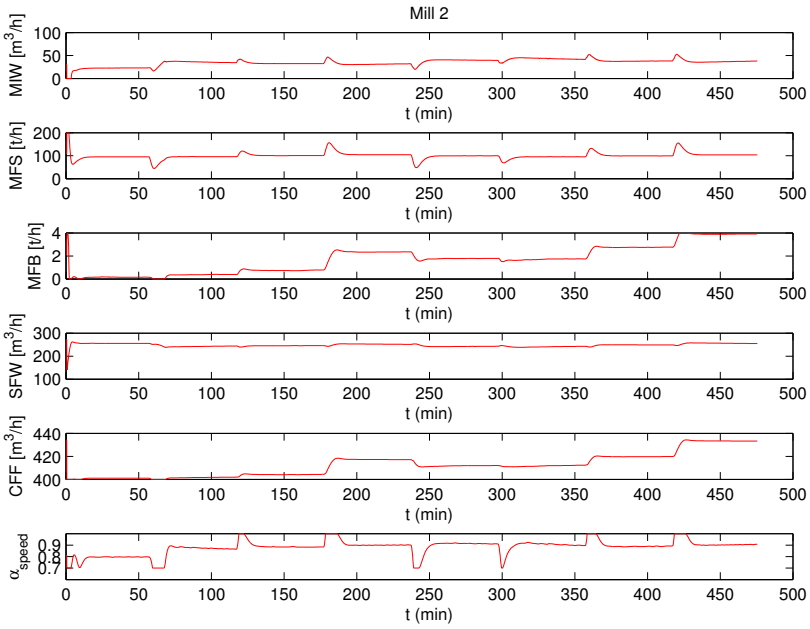

Figure 5.34: Manipulated variables for milling circuit 2.

The P_{mill} result for each mill is the main focus in this power optimization formulation. These results are shown in Fig. 5.35 and Fig. 5.36 for the respective milling circuits. The blue line indicates the initial power consumption based on a constant $LOAD$ and THP set-point of 45 m^3 and 90 t/h respectively.

Fig. 5.35 illustrates that the initial run for milling circuit 1 had an average P_{mill} value of approximately 1800 kW while the P_{mill} value for milling circuit 2 averaged around 1580 kW. This clearly shows that milling circuit 2 consumes less power than milling circuit 1. This result justifies the power optimizers THP set-point increase for milling circuit 2.

There will be an increase in power consumed by increasing the THP set-point of milling circuit 2. This is evident in Fig. 5.36 where the optimized output exceeds the initial output. The α_{speed} manipulated variable has a significant impact on this result. In milling circuit 1, there is a significant drop in the THP set-point which yields a lower power consumption. These figures illustrate the aim of the power optimizer which is to essentially minimize the combined area under the P_{mill} curves while still operating in a feasible region.

The initial simulation combined energy consumption of the two milling circuits was 26805.6 kWh in the eight hour shift. This was reduced by 2.5% to 26130.1 kWh. In the eight hour shift, energy was reduced by 675.5 kWh. This is a noteworthy amount of energy saving without significantly compromising on product quality or quantity. The product quality remained very similar before and after optimization, while a 0.2 % drop in overall throughput resulted. The initial throughput of both milling circuits was 1427.97 $t/8h$ or 89.25 (t/h)/mill and the optimized throughput resulted in 1425.11 $t/8h$ or an average of 89.07 (t/h)/mill. These results show that the power optimization formulation was successful in determining the milling circuit that consumed less power and then driving that milling circuit to produce more product.

On an actual plant, the power optimizer would only run once per eight hour shift. The optimizer would make use of sampled values from the previous shift to determine the new set-points for the next shift.

Additional results showed that a significant amount of power can be saved by sacrificing product quality or quantity, however this is not desirable in the current scenario. The cost of electricity is very small compared to the value of the product. Numerous other attempts have been made to increase the power saving and is discussed in the section that follows.

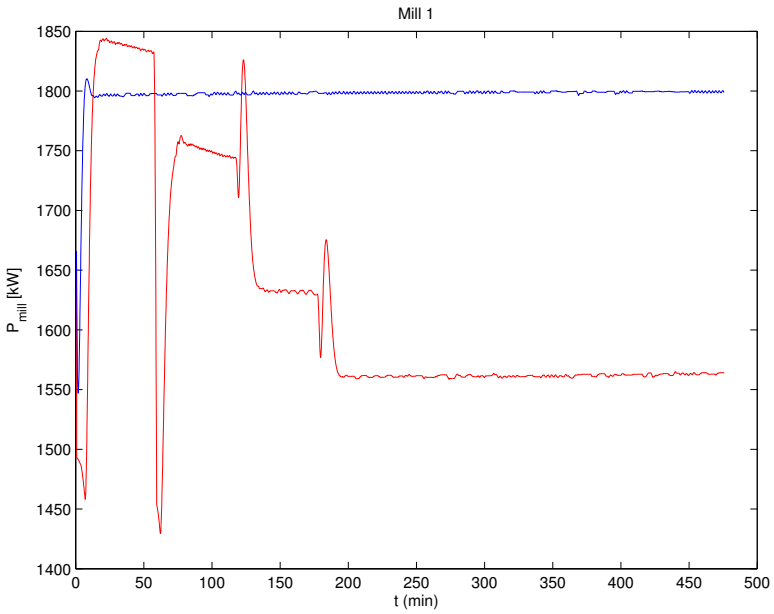

Figure 5.35: Power output from milling circuit 1. Initial run shown in blue and optimized run shown in red.

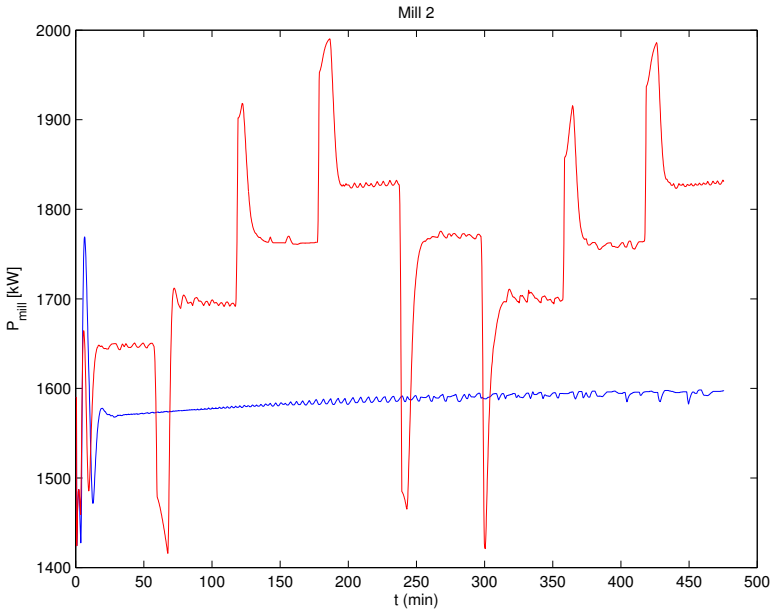

Figure 5.36: Power output from milling circuit 2. Initial run shown in blue and optimized run shown in red.

5.3.5 Additional attempts and drawbacks

The list that follows describes some of the attempts made and the drawbacks of each proposal.

- Changing the inner loop control strategy to exclude α_{speed} was considered. The region of operation is very small especially when trying to control two conflicting output variables. In many simulations it was seen that the MIW would increase significantly and result in the rheology factor exceeding 0.7. Once this happens then the milling circuit does not reach set-point.

- The number of control moves and prediction horizon for the NMPC was varied with no significant improvement. If the prediction horizon is too short then CFF often saturates.

- ROM ore milling circuit 2 parameters were varied in the opposite direction and similar power savings were seen. Additional parameters such as $\varphi_{P_{max}}$ and v_{pmax} were varied. There was no significant change in power savings.

- Alternative inner loop control configurations were considered, such as only using nonlinear model predictive control for the complete milling circuit. The $SVOL$ would saturate at the maximum level and therefore compromising on product quality while trying to increase the flow rate out of the sump. This could be further investigated and tuned to prevent $SVOL$ from reaching the upper constraint. Keeping MFB constant and MIW as a ratio to MFS also yielded positive saving results.

- Removing $LOAD$ as a set-point from the optimizer resulted in a positive savings however, not as much as when $LOAD$ is considered.

- An attempt to estimate the α_{speed} value using a neural network based on the $LOAD$-THP values was made. Unfortunately the accuracy was not acceptable and resulted in inaccurate P_{mill} estimates.

- The simulation time was extended to 24 hours and a similar power saving resulted.

- Setting α_{speed} a function of the THP set-point yielded very similar positive results i.e. $\alpha_{speed} = K \times THP$. However, the constant K would need to be determined based on the different milling circuit models.

5.4 CONCLUSION

The control configuration proposed in Chapter 4 is utilised in this chapter to propose a new effective control design philosophy for complex milling circuits, which meets the objectives of high product quality and manageable loads in the mill as well as minimizing the power consumption for the mill. Effectiveness of the proposed approach has been demonstrated by taking various combinations of throughput and load within the mill.

It can be noted that recent studies show that power can be saved by implementing a real time optimizer on a supervisory outer-loop level. This chapter, on the other hand, tries to save energy in the inner loop (by additionally penalizing an energy factor as part of the cost function). Hence, there is a scope to combine the two philosophies to obtain a better energy savings, which is a topic for future research. Future research should also contain noise and effects of modelling errors.

A multiple ROM ore milling circuit power optimization formulation has been developed. The formulation starts with estimating the power consumption using a neural network. The power consumption can be expanded to include all equipment within a ROM ore milling circuit however, just the mill is considered in this work. The formulation is highly dependant on α_{speed}, which is required as a manipulated variable in the inner loop control configuration. Despite this interaction, the power optimization formulation, in the outer loop, resulted in an energy saving of 675.5 kWh, in an eight hour period. Numerous attempts have been made to increase the power saving while keeping to the main goal to control product quality and quantity. The power optimization formulation successfully identified the more efficient ROM ore milling circuit and increased throughput while remaining in a feasible operating region.

CHAPTER 6

CONCLUSION

This dissertation proposes a nonlinear control solution for the run-of-mine ore milling circuit. The control solution comprises of a state estimator and nonlinear controller for the ROM ore milling circuit. Power optimization has also been considered, both in the inner loop control configuration as well as in an outer loop multiple milling circuit configuration. An attempt has been made to make the solution as practical as possible and does not require any ambitious measurements. The aim of the control solution is to always ensure a consistent product quality and quantity.

State estimation was initially studied since nonlinear model predictive control is highly dependent on accurate state estimates. A neural network has been compared to a particle filter for state estimation. A novel combination of the two state estimation methods has been proposed. The novel work also includes the control configuration proposed as well as the power optimization research.

The control configuration has been divided into a nonlinear model predictive controller and a dynamic inversion controller which can also be seen as an advanced process control configuration linked with a fast base layer configuration.

6.1 SUMMARY OF RESULTS

6.1.1 State estimation

Chapter 3 consisted of a neural network to estimate the internal states in a mill model. The neural network results were successful when the estimated five states were subjected to disturbances within the training data. However, when disturbances that were larger than in the training area the neural network method does not estimate accurately. The neural network method was compared to the

particle filter method where the particle filter method is more robust and can handle larger unknown disturbances and it was found that the neural network also makes use of fewer measurements. The particle filter method struggled with estimating the rocks and balls within the mill. This is because these two materials do not leave the mill and therefore there is no output measurement available to improve the state estimate.

A novel method of combining the particle filter and neural network estimates has been studied. The combined technique always performs better than the neural network alone however, when the disturbance is above the training region then the particle filter approach is more accurate. The combined method could be said to use sample data to determine state estimates using the particle filter. The particle filter estimates are then used to correct the offset present with the neural network results.

6.1.2 Control of a ROM ore milling circuit

Chapter 4 builds on the work by Coetzee *et al.* (2010) where a robust nonlinear model predictive controller was used for the control of the ROM ore milling circuit. Here on the other hand, the control of the milling circuit was broken up into a nonlinear model predictive controller and a dynamic inversion control technique. This control configuration meets the additional goal to achieve a high product quality and consistent product quantity. Downstream processes will benefit by ensuring that consistency of the ROM ore milling circuits outputs are maintained.

The novel work completed involves not only the design approach for the control configuration but also the integration with state estimation. This fills the gap in the literature with regard to a ROM ore milling circuit. In essence, a complete nonlinear control solution is presented while making use of minimal plant measurements.

The controller was tested with noise and disturbances and the particle size estimate was maintained despite this. The disturbances introduced were changes in the size distribution of the feed to the mill (the fraction of rocks and fines to the mill) and changes in the hardness of the rock (the power needed per ton of fines produced). For the disturbances, only the plant model was adjusted and not the model within the state estimator or controller.

The simulation results show that this control configuration has the potential to be successfully applied

to a ROM ore milling circuit.

6.1.3 Power optimization

Chapter 5 target two concepts, power optimization with a single milling circuit as well as multiple milling circuits that operate in parallel. In both of these cases, the underlying control configuration of Chapter 4 is followed. The first case however, penalizes power consumption within the objective function of the nonlinear model predictive controller. Therefore the nonlinear model predictive controller aims to reach set-point for PSE, THP and $LOAD$ while minimizing the power draw from the mill motor.

The results show that a energy reduction of 332.7 kWh resulted from the eight hour simulation. A drop of 0.6 % and 0.1% resulted for THP and PSE respectively. Various operating points were simulated to demonstrate the effectiveness of the proposed approach.

In the multiple ROM ore milling circuit configuration, an outer loop power optimizer was formulated. The outer loop formulation would provide set-points for PSE, THP and $LOAD$. The inner loop consisted of the nonlinear control configuration proposed. A scenario with two parallel ROM ore milling circuits was simulated with the second milling circuit slightly modified. The following modifications were made, α_f (fraction of fines in the ore) was increased by 10 %, α_r (fraction of rocks in the ore) was increased by 10 % and ϕ_f (power needed per ton of fines produced) was decreased by 10 %. Intuitively this would result in milling circuit 1 consuming more power for the same outputs as milling circuit 2. The power optimization formulation made use of a neural network to predict the milling circuits power consumption and with that information the more efficient milling circuit would operate with the majority of the throughput.

The results showed that milling circuit 2 was more efficient and the THP set-point was increased for that milling circuit compared to the base case. The difference between hourly set-points was restricted to ensure a smooth stable operation. The formulation is highly dependant on α_{speed} which is a required for the inner control to achieve a high throughput and product quality. The effect of α_{speed} resulted in the reduction of energy consumed by milling circuit 1 but an increase in energy consumed by milling circuit 2. The overall effect was a energy saving of 675.5 kWh in the eight hour shift.

6.2 SUGGESTION FOR FURTHER WORK

An attempt has been made in this work to increase the real world application of model predictive control in the minerals processing industry. State estimation is required for model predictive control and there is room to conduct a full comparison between state estimation techniques. The comparison would hopefully yield the best state estimator under various scenarios such as limited measurements available, significant noise, disturbance performance or model mismatch. The suggested state estimation techniques would e.g. use the particle filter, neural network, extended Kalman filter or the unscented Kalman filter.

The control configuration developed has many tuning parameters and because of the current computational power, some of these are not viable to constantly vary. Since computational power is increasing, future work should look at nonlinear model predictive control with a very large prediction horizon and increases the number of control moves made. Another concept/study could target optimizing the prediction horizon and other tunable variables for a model predictive controller based on a desired computational time.

Power optimization of two milling circuits in parallel was studied and the effect of scaling this formulation could potentially yield much better results. Suggestions for future work would be to implement no less than 5 milling circuits in parallel with a possible simpler inner loop control configuration. Each milling circuit should contain a slight varied parameter set. The power formulation can be extended to include all equipment in the milling circuit such as the pump after the sump.

REFERENCES

Apelt, T., Asprey, S. and Thornhill, N. (2001). Inferential measurement of SAG mill parameters, *Minerals Engineering* **14**(6): 575 – 591.

Apelt, T., Asprey, S. and Thornhill, N. (2002). Inferential measurement of SAG mill parameters II: state estimation, *Minerals Engineering* **15**(12): 1043 – 1053.

Arulampalam, M., Maskell, S., Gordon, N. and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, *IEEE Trans. Signal Process* **50**: 174–188.

Bouche, C., Brandt, C., Broussaud, A. and van Wayne Drunick (2005). Advanced control of gold ore grinding plants in South Africa, *Minerals Engineering* **18**(8): 866 – 876.

Chen, X., Li, Q. and Fei, S. (2008). Constrained model predictive control in ball mill grinding process, *Powder Technology* **186**(1): 31 – 39.

Chen, X., Li, Q., Fei, S. and Zhai, J. (2007a). Override and model predictive control of particle size and feed rate in grinding process, pp. 704–708.

Chen, X., Zhai, J., Li, S. and Li, Q. (2007b). Application of model predictive control in ball mill grinding circuit, *Minerals Eng.* **20**(11): 1099 – 1108.

Coetzee, L. (2009). *Robust Nonlinear Model Predictive Control of a Closed Run-of-Mine ore Milling Circuit*, PhD thesis, University of Pretoria.

Coetzee, L., Craig, I. and Kerrigan, E. (2010). Robust nonlinear model predictive control of a run-of-mine ore milling circuit, *IEEE Transactions on Control Systems Technology* **18**(1): 222 –229.

Conradie, A. V. E. and Aldrich, C. (2001). Neurocontrol of a ball mill grinding circuit using evolu-

tionary reinforcement learning, *Minerals Engineering* **14**(10): 1277 – 1294.

Craig, I. (2012). Grinding mill modelling and control: past, present and future, Plenary Lecture, 31st Chinese Control Conference, Hefei, China.

Craig, I., Aldrich, C., Braatz, R., Cuzzola, F., Domlan, E., Engell, S., Hahn, J., Havlena, V., Horch, A., Huang, B., Khanbaghi, M., Konstantellos, A., Marquardt, W., McAvoy, T., Parisini, T., Pistikopoulos, S., Samad, T., Skogestad, S., Thornhill, N. and Yu, J. (2011). Control in the process industries, in "The Impact of Control Technology", *Technical report*, IEEE Control Systems Society.

Craig, I. and MacLeod, I. (1995). Specification framework for robust control of a run-of-mine ore milling circuit, *Control Engineering Practice* **3**(5): 621 – 630.

Craig, I. and MacLeod, I. (1996). Robust controller design and implementation for a run-of-mine ore milling circuit, *Control Engineering Practice* **4**(1): 1 – 12.

Creamer, T. (2013). Eskoms industrial and mining customers face 9.6% tariff increase from April, http://www.engineeringnews.co.za/article/eskoms-industrial-and-mining-customers-face_96_tariff-increase-from-April-2013-03-22. Accessed October 7, 2013.

Duarte, M., Suarez, A. and Bassi, D. (2001). Control of grinding plants using predictive multivariable neural control, *Powder Technology* **115**(2): 193 – 206.

Enns, D., Bugajski, D., Hendrick, R. and Stein, G. (1994). Dynamic inversion: an evolving methodology for flight control design, *International Journal of Control* **59**(1): 71–91.

Eskom (2013). Tariffs and charges, http://www.eskom.co.za/CustomerCare/TariffsAndCharges/Pages/Tariffs_And_Charges.aspx. Accessed June 9, 2014.

Flament, F., Thibault, J. and Hodouin, D. (1993). Neural network based control of mineral grinding plants, *Minerals Engineering* **6**(3): 235 – 249.

Garatti, S. and Bittanti, S. (2008). Parameter estimation via artificial data generation with the "two-stage" approach, *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)* pp. 5600–5604.

Hagan, M. T., Demuth, H. B. and Beale, M. (1996). *Neural network design*, PWS Publishing Co., Boston, MA, USA.

Herbst, J. and Pate, W. (1999). Object components for comminution system softsensor design, *Powder Technology* **105**(1-3): 424–429.

Hodouin, D. (2011). Methods for automatic control, observation, and optimization in mineral processing plants, *Journal of Process Control* **21**(2): 211 – 225.

Hodouin, D., Jamsa-Jounela, S.-L., Carvalho, M. and Bergh, L. (2001). State of the art and challenges in mineral processing control, *Control Engineering Practice* **9**(9): 995 – 1005.

Hulbert, D., Craig, I., Coetzee, M. and Tudor, D. (1990). Multivariable control of a run-of-mine milling circuit, *J.S. Afr. Inst. Min. Metall.* **90**(7): 173–181.

Hulbert, D. G. (2005). Simulation of milling circuits: part 1 & 2, *Technical report*, Mintek, Johannesburg, South Africa.

Le Roux, J., Craig, I. and Padhi, R. (2013a). State and parameter estimation for a grinding mill circuit from operational input-output data, *10th IFAC International Symposium on Dynamics and Control of Process Systems*, Vol. 10, pp. 178–183. DOI: 10.3182/20131218-3-IN-2045.00046.

Le Roux, J., Craig, I., Hulbert, D. and Hinde, A. (2013b). Analysis and validation of a run-of-mine ore grinding mill circuit model for process control, *Minerals Engineering* **43-44**: 121–134.

Lee, J. (2011). Model predictive control: Review of the three decades of development, *International Journal of Control, Automation and Systems* **9**(3): 415–424.

Makokha, A. and Moys, M. (2012). Multivariate approach to on-line prediction of in-mill slurry density and ball load volume based on direct ball and slurry sensor data, *Minerals Engineering* **26**(1): 13–23.

Matthews, B. and Craig, I. (2013). Demand side management of a run-of-mine ore milling circuit, *Control Engineering Practice* **21**(6): 759–768.

Naidoo, M., Olivier, L. and Craig, I. (2013). Combined neural network and particle filter state estimation with application to a run-of-mine ore mill, *10th IFAC International Symposium on Dynam-*

ics and Control of Process Systems, pp. 397–402. DOI: 10.3182/20131218-3-IN-2045.00103.

Olivier, L. and Craig, I. (2013). Model-plant mismatch detection and model update for a run-of-mine ore milling circuit under model predictive control, *Journal of Process Control* **23**(2): 100–107.

Olivier, L., Craig, I. and Chen, Y. (2012a). Fractional order and BICO disturbance observers for a run-of-mine ore milling circuit, *Journal of Process Control* **22**(1): 3 – 10.

Olivier, L. E. (2011). *Peripheral control tools for a run-of-mine ore milling circuit*, Master's thesis, University Of Pretoria.

Olivier, L., Huang, B. and Craig, I. (2012b). Dual particle filters for state and parameter estimation with application to a run-of-mine ore mill, *Journal of Process Control* **22**(4): 710 – 717.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. (1992). *Numerical Recipes In C: The Art Of Scientific Computing*, second edn, Cambridge University Press.

Qin, S. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology, *Control Engineering Practice* **11**(7): 733 – 764.

Rajamani, R. K. and Herbst, J. A. (1991). Optimal control of a ball mill grinding circuit-II. Feedback and optimal control, *Chemical Engineering Science* **46**(3): 871–879.

Ramasamy, M., Narayanan, S. and Rao, C. (2005). Control of ball mill grinding circuit using model predictive control scheme, *Journal of Process Control* **15**(3): 273–283.

Remes, A., Aaltonen, J. and Koivo, H. (2010). Grinding circuit modeling and simulation of particle size control at Siilinjarvi concentrator, *International Journal of Mineral Processing* **96**(1): 70 – 78.

Ristic, B., Arulampalam, S. and Gordon, N. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, Boston.

Rule, C. (2011). Stirred milling at Anglo American Platinum, *in* K. Major, B. Flintoff, B. Klein and K. McLeod (eds), *International Conference on Autogenous Grinding, Semiautogenous Grinding and High Pressure Grinding Roll Technology*, Vancouver, B.C., Canada.

Seborg, D. E., Edgar, T. F. and Mellichamp, D. A. (2004). *Process dynamics and control*, 2nd edn, NJ: Wiley.

Stange, W. (1993). Using artificial neural networks for the control of grinding circuits, *Minerals Engineering* **6**(5): 479 – 489.

Valenzuela, J., Najim, K., del Villar, R. and Bourassa, M. (1993). Learning control of an autogenous grinding circuit, *International Journal of Mineral Processing* **40**(1-2): 45 – 56.

Van der Merwe, R. (2004). *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*, PhD thesis, University of Stellenbosch.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. and Yin, K. (2003). A review of process fault detection and diagnosis: Part III: Process history based methods, *Computers & Chemical Engineering* **27**(3): 327 – 346.

Wei, D. and Craig, I. K. (2009). Grinding mill circuits - a survey of control and economic concerns, *International Journal of Mineral Processing* **90**(14): 56 – 66.

Willis, M., Montague, G., Di Massimo, C., Tham, M. and Morris, A. (1992). Artificial neural networks in process estimation and control, *Automatica* **28**(6): 1181–1187.

Xi, Y.-G., Li, D.-W. and Lin, S. (2013). Model predictive control - status and challenges, *Acta Automatica Sinica* **39**(3): 222 – 236.

LIST OF FIGURES

LIST OF TABLES

Figure number	Page number	Source	Reference	Permission
Fig 1.1	5	Own	N.A.	N.A.
Fig 2.1	7	Own/UP	Control Department, UP	N.A.
Fig 2.2	8	Own/UP	Control Department, UP	N.A.
Fig 2.3	11	Own/UP	Control Department, UP	N.A.
Fig 2.4	19	Own/UP	Control Department, UP	N.A.
Fig 2.5	20	Own/UP	Control Department, UP	N.A.
Fig 2.6	23	Own	N.A.	N.A.
Fig 3.1	26	Own	N.A.	N.A.
Fig 3.2	28	Own	N.A.	N.A.
Fig 3.3	29	Own	N.A.	N.A.
Fig 3.4	29	Own	N.A.	N.A.
Fig 3.5	29	Own	N.A.	N.A.
Fig 3.6	31	Own	N.A.	N.A.
Fig 3.7	33	Own	N.A.	N.A.
Fig 3.8	34	Own	N.A.	N.A.
Fig 4.1	37	Own	N.A.	N.A.
Fig 4.2	38	Own	N.A.	N.A.
Fig 4.3	43	Own	N.A.	N.A.
Fig 4.4	50	Own	N.A.	N.A.
Fig 4.5	51	Own	N.A.	N.A.
Fig 4.6	51	Own	N.A.	N.A.
Fig 4.7	52	Own	N.A.	N.A.
Fig 4.8	52	Own	N.A.	N.A.
Fig 4.9	53	Own	N.A.	N.A.
Fig 4.10	54	Own	N.A.	N.A.
Fig 4.11	55	Own	N.A.	N.A.
Fig 4.12	56	Own	N.A.	N.A.
Fig 4.13	56	Own	N.A.	N.A.
Fig 4.14	57	Own	N.A.	N.A.
Fig 4.15	58	Own	N.A.	N.A.
Fig 4.16	58	Own	N.A.	N.A.
Fig 4.17	59	Own	N.A.	N.A.
Fig 5.1	64	Own	N.A.	N.A.
Fig 5.2	64	Own	N.A.	N.A.
Fig 5.3	65	Own	N.A.	N.A.
Fig 5.4	65	Own	N.A.	N.A.
Fig 5.5	66	Own	N.A.	N.A.
Fig 5.6	67	Own	N.A.	N.A.
Fig 5.7	67	Own	N.A.	N.A.
Fig 5.8	68	Own	N.A.	N.A.
Fig 5.9	68	Own	N.A.	N.A.
Fig 5.10	69	Own	N.A.	N.A.
Fig 5.11	69	Own	N.A.	N.A.
Fig 5.12	71	Own	N.A.	N.A.
Fig 5.13	72	Own	N.A.	N.A.
Fig 5.14	72	Own	N.A.	N.A.
Fig 5.15	73	Own	N.A.	N.A.
Fig 5.16	74	Own	N.A.	N.A.
Fig 5.17	75	Own	N.A.	N.A.
Fig 5.18	75	Own	N.A.	N.A.
Fig 5.19	77	Own	N.A.	N.A.

Fig 5.20	79	Own	N.A.	N.A.
Fig 5.21	79	Own	N.A.	N.A.
Fig 5.22	82	Own	N.A.	N.A.
Fig 5.23	83	Own	N.A.	N.A.
Fig 5.24	83	Own	N.A.	N.A.
Fig 5.25	84	Own	N.A.	N.A.
Fig 5.26	84	Own	N.A.	N.A.
Fig 5.27	85	Own	N.A.	N.A.
Fig 5.28	85	Own	N.A.	N.A.
Fig 5.29	87	Own	N.A.	N.A.
Fig 5.30	87	Own	N.A.	N.A.