UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

PERIPHERAL CONTROL TOOLS FOR A RUN-OF-MINE ORE MILLING CIRCUIT

by

Laurentz Eugene Olivier

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

November 2011

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

SUMMARY

PERIPHERAL CONTROL TOOLS FOR A RUN-OF-MINE ORE MILLING CIRCUIT

by

Laurentz Eugene Olivier

Study leader:	Prof. I. K. Craig
Department:	Electrical, Electronic and Computer Engineering
University:	University of Pretoria
Degree:	Master of Engineering (Electronic Engineering)
Keywords:	Fractional order, disturbance observer, dual estimation, milling, model-plant mismatch, particle filter, run-of-mine ore

Run-of-mine ore milling circuits are generally difficult to control owing to the presence of strong external disturbances, poor process models and the unavailability of important process variable measurements. These shortcomings are common for processes in the mineral-processing industry. For processes that fall into this class, the peripheral control tools in the control loop are considered to be as important as the controller itself. This work addresses the implementation of peripheral control tools on a run-of-mine ore milling circuit to help overcome the deteriorated control performance resulting from the aforementioned shortcomings.

The effects of strong external disturbances are suppressed through the application of a disturbance observer. A fractional order disturbance observer is also implemented and a novel Bode ideal cut-off disturbance observer is introduced. The issue of poor process models is addressed through the detection of significant mismatch between the actual plant and the available model from process data. A closed-form expression is given for the case where the controller has a transfer function. If the controller does not have a transfer function, a partial correlation analysis is used to detect the transfer function elements in the model transfer function matrix that contain significant mismatch.

The mill states and important mill parameters are estimated with the use of particle filters. Simultaneous state and parameter estimation is compared with a novel dual particle filtering scheme. A sensitivity analysis shows the class of systems for which dual estimation would provide superior

estimation accuracy over simultaneous estimation. The implemented peripheral control tools show promise for current milling circuits where proportional-integral-derivative (PID) control is prevalent, and also for advanced control strategies, such as model predictive control, which are expected to become more common in the future.

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

OPSOMMING

RANDBEHEERINSTUMENTE VIR 'N MAALKRING WAT ONBEHANDELDE ERTS MAAL

deur

Laurentz Eugene Olivier

Studieleier:	Prof. I. K. Craig
Departement:	Elektriese, Elektroniese en Rekenaar-Ingenieurswese
Universiteit:	Universiteit van Pretoria
Graad:	Magister in Ingenieurswese (Elektroniese Ingenieurswese)
Sleutelwoorde:	Breuk-orde, dubbel-afskatting, maling, model-aanleg-verskil, onbehandelde erts, partikel-filter, steuringsafskatter

Maalkringe wat onbehandelde erts maal is oor die algemeen moeilik om te beheer as gevolg van die teenwoordigheid van sterk eksterne steurings, onakkurate aanlegmodelle en metings van belangrike prosesveranderlikes wat ontbreek. Hierdie probleme is algemeen vir aanlegte in die mineraalprosesseringsbedryf. Vir aanlegte in hierdie klas word die randbeheerinstrumente as net so belangrik as die beheerder beskou. Hierdie verhandeling beskryf die implementering van randbeheerinstrumente vir 'n maalkring wat onbehandelde erts maal, om die verswakte beheerverrigting teen te werk wat veroorsaak word deur bogenoemde probleme.

Die impak van sterk eksterne steurings word teengewerk deur die implementering van 'n steuringsafskatter. 'n Breuk-orde-steuringsafskatter is ook geïmplementeer en 'n nuwe Bode ideale afsny-steuringsafskatter word voorgestel. Die kwessie van onakkurate aanlegmodelle word hanteer deur van die aanlegdata af vas te stel of daar 'n verskil is tussen die aanleg en die beskikbare model van die aanleg. 'n Uitdrukking word gegee vir hierdie verskil vir die geval waar die beheerder met 'n oordragsfunksie voorgestel kan word. Indien die beheerder nie 'n oordragsfunksie het nie, word van 'n parsiële korrelasie-analise gebruik gemaak om die element, of elemente, in die aanleg se oordragsfunksiematriks te identifiseer wat van die werklike aanleg verskil.

Die toestande en belangrike parameters in die meul word beraam deur van partikel-filters gebruik

te maak. Gelyktydige toestand- en parameter-beraming word vergelyk met ’n nuwe dubbel-partikel-filter skema. ’n Sensitiwiteitsanalise wys die klas van stelsels waarvoor dubbel-afskatting meer akkurate waardes sal gee as gelyktydige afskatting. Die voorgestelde randbeheerinstrumente is toepaslik vir huidige maalkringe waar PID-beheer algemeen is, asook vir gevorderde beheerstrategieë, soos model-voorspellende beheer, wat na verwagting in die toekoms meer algemeen sal word.

ACKNOWLEDGEMENT

I would like to thank my supervisor, Prof. Ian Craig, for showing me what research is and guiding me along this path. I am also grateful for your focus on publications as research output and your inputs in the process.

I would also like to thank Prof. YangQuan Chen from Utah State University for his interesting discussions about some of his multitude of ideas.

A great thank you goes to Prof. Biao Huang and his research group at the University of Alberta in Edmonton. Thank you, firstly for hosting my visit to Canada, and secondly for all the help and guidance during my stay. You really helped me achieve more than I hoped for in the short time I had in Canada.

A special thank you goes to my fiancée, Mariska, as well as to my parents; your support during my studies is much appreciated.

I would also like to thank God for allowing me the opportunity and ability to complete this work.

LIST OF ABBREVIATIONS

AG	Autogenous grinding
BIBO	Bounded-input bounded-output
BICO	Bode ideal cut-off
CSW	Cumulative sum of weights
DMC	Dynamic matrix control
DOB	Disturbance observer
EKF	Extended Kalman filter
FO-DOB	Fractional order disturbance observer
FO-LPF	Fractional order low-pass filter
FOPTD	First order plus time delay
IMC	Internal model control
INA	Inverse Nyquist array
IRID	Impulse response invariant discretisation
ISE	Integral squared error
MAP	Maximum a Posteriori
MIMO	Multi-input multi-output
MPC	Model predictive control
MPM	Model-plant mismatch
OE	Output error
PDF	Probability density function
PI(D)	Proportional-integral-derivative
RNMPC	Robust nonlinear model predictive control
ROM	Run-of-mine
SAG	Semi-autogenous grinding
SID	System identification
SIR	Sampling importance resampling
SISO	Single-input single-output
SMC	Sequential Monte Carlo
SVD	Singular value decomposition
UKF	Unscented Kalman filter

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

The process of extracting metals from run-of-mine (ROM) ore is an involved chain of steps as illustrated by Hodouin (2011), from whose work Fig. 1.1 is reproduced. Grinding ore down to a fine product is usually the first step in the metallurgical extraction process (Craig and MacLeod, 1995). The process of crushing and grinding the ore into minute fragments is known as comminution and comminution of metal-bearing ore is primarily required to liberate minerals to make them amenable to the subsequent metallurgical extraction steps (Marsden and House, 1992). These liberated minerals are then separated through a process such a flotation or leaching. At this step further regrinding may be specified as needed. In the separation process the rejects are removed and the final product, the metals, are extracted from the concentrate through some recovery process such as electrometallurgy or pyrometallurgy.

This work is specifically focussed on a ROM ore milling circuit, which accomplishes the crushing, grinding and size classification processes in a single unit-operation to produce liberated minerals from ROM ore. Ore, without any prior processing, is fed into the milling circuit and the liberated minerals, ground down to a specified size, are produced in the form of a slurry.

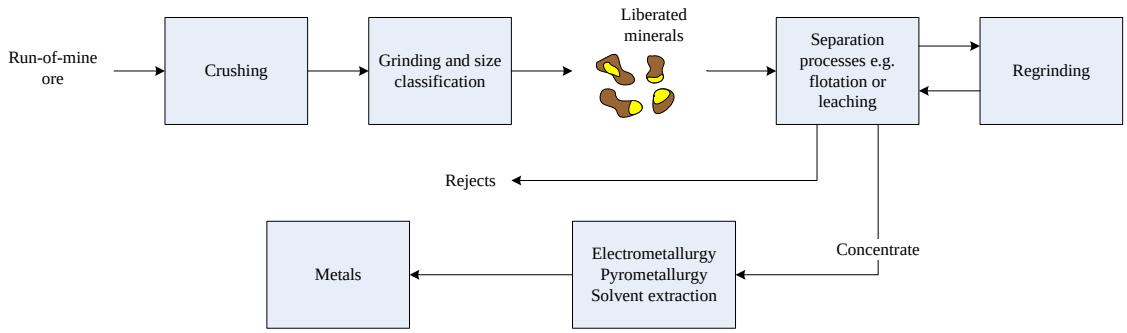

Figure 1.1: Simplified metallurgical extraction process from Hodouin (2011).

1.1 MOTIVATION

Milling is not only the first step in the metallurgical extraction process chain, but usually also the most expensive (Craig and MacLeod, 1995). Increasing milling efficiency will therefore lead to substantial economic benefits. Even larger financial benefits are obtained through the positive effect that more consistent milling has on the downstream extraction processes (Craig and MacLeod, 1995). This is because the downstream separation processes are more efficient when operated with a consistent feed stream.

Processes common in the mineral-processing industry, such ROM ore milling, are usually characterised by the presence of strong external disturbances, the availability of poor process models and process variables that are difficult to measure (Hodouin, 2011). For this class of processes the peripheral control tools in the control loop are considered to be as important as the controller itself (Hodouin, 2011).

Fig. 1.2 shows a generalised control loop as is very generally found in control systems analysis. It shows the general scheme used to control a process with some specified set-points while the process is affected by external disturbances. The process output is depicted as y, the external disturbance as d and the internal states of the process as x. This diagram can be expanded to that shown in Fig. 1.3, which shows more detail about the data-processing block of Fig. 1.2 as well as the origin of the set-points as may be supplied by the supervisory control layer. Expansion of the generalised control loop was done according to Hodouin (2011), and also incorporating ideas expressed by Venkatasubramanian et al. (2003) and Narasimhan and Jordache (2000).

From Fig. 1.3 it is clear that there are many operations, other than the controller itself, that contribute to the closed-loop system performance. Many of these operations form part of the class of processes that may collectively be called peripheral control tools, which is subsequently defined.

Peripheral control tools constitute all the elements in the control loop, other than the controller itself, that function to improve controller performance such as fault detection and isolation, data reconciliation, observers, soft sensors, optimisers and model parameter tuners. This work focuses on how peripheral control tools may be implemented to counteract the deteriorated control performance usually caused by

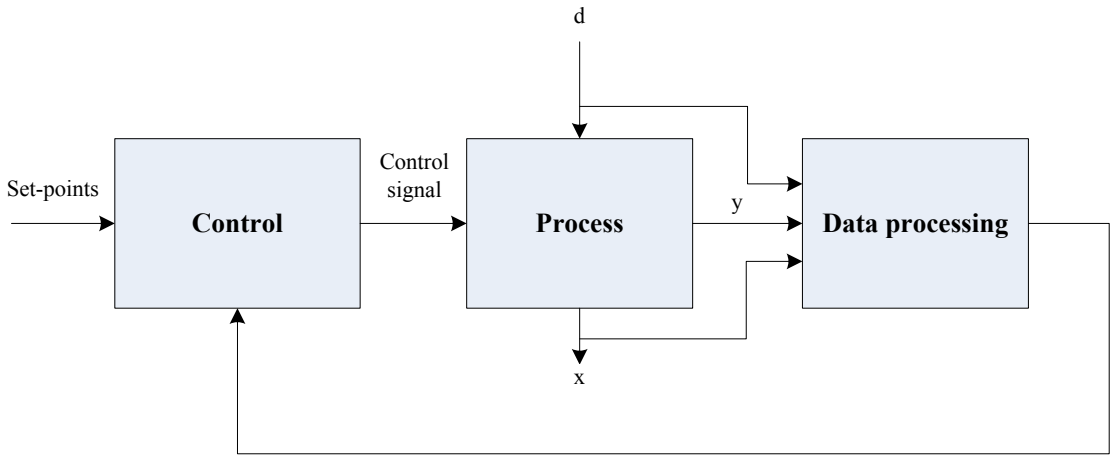

Figure 1.2: Generalised control loop.

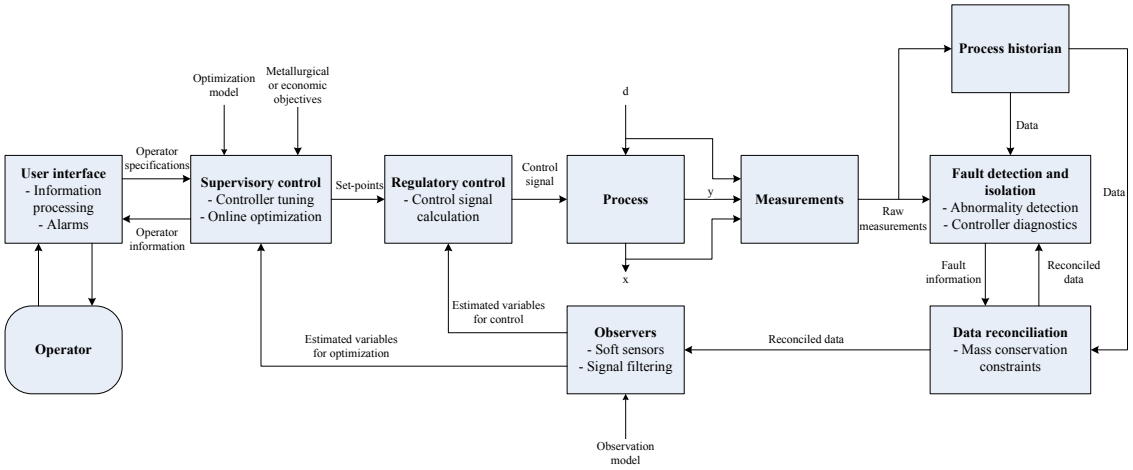

Figure 1.3: Expanded control loop.

- strong external disturbances,
- poor process models and
- unavailable process variable measurements,

with specific application to a ROM ore milling circuit.

1.2 BACKGROUND

Much work has already been done on the control of ROM ore milling circuits. A brief overview of the control methodologies previously applied is firstly given here, before some background will be given on peripheral control tools that are deemed appropriate for implementation, to remedy the problems with controlling a ROM ore milling circuit as given in the previous section.

1.2.1 Control of milling circuits

A comparison of proportional-integral (PI) control and optimal control was done by Rajamani and Herbst (1991). Control based on the inverse Nyquist array (INA) method was done by Hulbert et al. (1990). Successful implementation of this method however requires a good understanding of the process as well as the INA synthesis method. In the light of the difficulties with the INA control method, the use of μ-synthesis to design a robust multi-input multi-output (MIMO) controller was proposed by Craig and MacLeod (1995, 1996). The authors concluded that the significant modelling input required is probably not justified by the performance achieved.

Neural network based control of milling circuits has also been investigated by Conradie and Aldrich (2001); Duarte et al. (1999, 2001). Evolutionary reinforcement learning was employed by Conradie and Aldrich (2001), in which the controller learns through rewarding control moves based on how well they satisfy the objective. In this method the controller can potentially apply unwanted control moves in order to learn. It is therefore beneficial to train the controller on a simulation model first before application on a real plant. This does however require a relatively accurate simulation model. In Duarte et al. (1999, 2001) use was made of three neural networks to achieve the desired control performance, one for state estimation, one for control and one for state trajectory predictions for predictive control.

An early form of model predictive control (MPC) called dynamic matrix control (DMC) was compared to PI control and learning automata by Valenzuela et al. (1994). The authors concluded that DMC provided the best controller performance, hinting at the potential of model-based predictive control. Linear MPC based on a four-input-four-output model of the plant as well as on a three-input-three-output model was investigated by Chen et al. (2007) and Chen et al. (2008) respectively. In both cases the use of MPC provided better performance than decoupled PID control. A comparative simulation study between multi-loop PI control as well as unconstrained and constrained MPC on a two-input-two-output linear model was done by Ramasamy et al. (2005). Their conclusion was that MPC performed well compared to PI control.

Robust nonlinear model predictive control (RNMPC) was implemented by Coetzee (2009); Coetzee et al. (2010). It was found that the RNMPC controller can successfully control important process variables even under large disturbances. There is however a trade-off between complexity and the good performance achieved.

This discussion on control methodologies employed for grinding mill circuits is by no means exhaustive, but it does highlight some of the main contributions in terms of control for milling circuits. Even with extremely complex control methodologies, controller performance may still suffer in the presence of unavailable process variable measurements, poor process models and strong external disturbances. Peripheral control tools are the supplementary operations that can help to counter these effects.

1.2.2 Peripheral control tools and applications to milling circuits

A lot of work has gone into the development of control theory and the application thereof on grinding mill circuits, as discussed in the previous section. Less work has however gone into the application of peripheral control tools such as observers, soft sensors and fault-detection algorithms for milling circuits. This is despite the fact that the peripheral control tools are considered to be as important as the controller itself (Hodouin, 2011) in this framework.

Peripheral control tools do however cover a wide range of topics. The discussion in this work will be limited to the three main sources of deteriorated controller performance given by Hodouin (2011) and listed in Section 1.1. These are the presence of strong external disturbances, poor process models and unavailable process measurements. Some of the main sources of this work are briefly introduced

in the rest of this section, and will be elaborated on in subsequent chapters.

1.2.2.1 External disturbance rejection

Most control methodologies do not directly consider disturbance rejection as part of the controller design (Chen et al., 2009). For this reason the controller usually cannot reject strong external disturbances in a fast and effective way, rather disturbances are more sluggishly rejected through feedback regulation. The suppression of strong external disturbances is important in ROM ore milling, as the feed ore variations are often very large (Coetzee, 2009) and have a significant effect on the operation of the milling circuit.

Strong external disturbances may be suppressed in a much more effective way through the use of a disturbance observer (DOB) (Chen et al., 2009). The application of a DOB for rejection of external disturbances on a grinding mill circuit is presented in Chen et al. (2009). A DOB was used in conjunction with a PI controller, and the performance achieved was compared to that of a detuned multi-loop PI controller and a model predictive controller. A DOB has also been applied in conjunction with an MPC controller by Yang et al. (2010) with good effect. The DOB scheme is used in this work, mainly for the rejection of external disturbances, but as will be shown later, this scheme also offers other advantages with respect to the effects of model-plant mismatch.

1.2.2.2 Model-plant mismatch detection

Much research has gone into controller performance assessment, for which Harris et al. (1999); Qin (1998); Huang and Shah (1999) give thorough reviews. Several performance benchmarks exist but less work has gone into addressing performance diagnostics (Badwe et al., 2009). Patwardhan and Shah (2002) address this aspect where they list five possible sources of poor controller performance: (1) limitations imposed by the combination of process and controller design; (2) changes in plant dynamics; (3) varying disturbances; (4) sensor faults; and (5) process nonlinearity.

This work focuses on the identification of deteriorated controller performance due to changes in plant dynamics. In any model-based control strategy a plant model is usually derived during the commissioning phase of the plant. Should the dynamics of the plant change for some reason, the model will differ from the actual plant. This difference between the model and the actual plant is known as model-plant mismatch (MPM). This mismatch leads to deteriorated controller performance

(Patwardhan and Shah, 2002) and it is therefore very useful if the mismatch could be identified in a timely manner so that process re-identification may be done.

Analysis techniques concerned with MPM may be divided into three categories. Firstly there are methods that analyse overall control loop performance. In these methods the source of poor control loop performance is not always identified and could be any of the elements listed by Patwardhan and Shah (2002). These methods would only indicate whether a control loop is performing adequately, usually through some predefined metric. This group of methods is not of specific concern in this work, and is well documented elsewhere (Harris et al., 1999; Qin, 1998; Huang and Shah, 1999; Patwardhan and Shah, 2002).

Secondly, some methods focus on specifically determining whether MPM is the source of poor control performance. This may be called MPM detection. These methods are closely related to model validation techniques, as the aim is to determine whether the model is a true representation of the plant. It is however noted by Huang et al. (2003) that some models pass rigorous validation tests while the controller performance is not good, whereas other models fail rigorous validation tests and yet produce good control performance.

An example of such an analysis is given by Huang (2008). Here a method for control loop monitoring and diagnosis based on Bayesian methods is proposed. The proposed framework has flexibility in implementation, such as the ability to incorporate *a priori* process knowledge into the diagnosis. This method is also robust in the sense that it can handle missing data. This holistic diagnostic framework will be able to detect MPM if it is present.

The third group is those methods that aim to determine specifically in which input-output channel MPM is present. If the source of the mismatch can be identified, full model re-identification would not be necessary. This is advantageous, as process re-identification is a costly and time-consuming exercise (Conner and Seborg, 2005). This type of analysis may be called MPM identification.

A scheme for MPM identification is proposed by Kano et al. (2010) in which explanatory variables are selected. If a large number of past inputs contribute to the residual, the corresponding transfer function element is deemed to contain mismatch. An alternative method based on a partial correlation analysis is presented in Badwe et al. (2009). Another method, proposed by Selvanathan and Tangirala (2010), uses the plant-model ratio and identifies different signatures for this ratio based on the mismatch that may be present. This method has however only been proven for single-input single-output (SISO)

systems with simple transfer function models.

Only process models of poor quality are often available for control of ROM ore milling circuits (Olivier and Craig, 2011). In order to identify the specific element, or elements, in the model transfer function matrix that contain MPM, the partial correlation analysis presented by Badwe et al. (2009) is applied in Chapter 4.

1.2.2.3 State and parameter estimation

Owing to the nature of the ROM ore milling process environment, measurements of important process variables are often unavailable (Hodouin, 2011). In order to do effective control of these variables, inferential measurements of their values are an attractive option (Apelt et al., 2001). Although important process variables cannot easily be measured, a lot is known about the physics of the process and phenomenological models can often be used for inferential measurements (Herbst and Pate, 1999).

Some work has been done on the inferential measurements of important process variables, such as product particle size, ore grindability and mill filling (Herbst and Pate, 1999). There is also a series of articles by Apelt and co-workers (Apelt et al., 2001, 2002a,b; Apelt and Thornhill, 2009a,b) wherein the total mill load and ball load values are estimated, state and parameter estimation for a complex milling model is illustrated, and the use of the estimates in an MPC scheme is shown.

Parameters that define the operation of the milling circuit are also valuable to know, since they may be incorporated into the control scheme for better controller operation. Image-processing techniques are often employed (see for example Tessier et al. (2007); Maerz et al. (1996)) to measure the feed ore composition, as opposed to using inferential measurements. This approach will however not be pursued in this work.

The combined state and parameter estimation for the phenomenological milling circuit model described in Chapter 2 has previously been unsolved. The focus in this work will be on Bayesian state and parameter estimation for the mill.

Recursive Bayesian estimation methods are grouped into the following four categories by Van der Merwe (2004):

1. Gaussian approximate methods

2. Direct numerical integration methods

3. Sequential Monte Carlo methods

4. Variational Bayesian methods.

Gaussian approximate methods model the density functions of the random variables to be estimated with Gaussian distributions, and only need to update the mean and covariance at each iteration step. The ubiquitous Kalman filter (Kalman, 1960) is the optimal solution to the recursive Bayesian estimation problem if the underlying system is linear and all probability densities are Gaussian. For nonlinear systems, the most widely implemented alternative is the extended Kalman filter (EKF) (Jazwinski, 1970). This filter makes use of Jacobian matrices that represent the linearised system at the current operating point. Another extension to the regular Kalman filter is the unscented Kalman filter (UKF) (Julier and Uhlmann, 1997), which makes use of the unscented transform to calculate the transformed mean and covariance of a distribution which has undergone a nonlinear transformation. Other variations of the Kalman filter are also listed by Van der Merwe (2004), but the aforementioned variations are the most well known and widely applied.

Direct numerical integration methods, also known as grid-based methods, make use of an N-dimensional grid that tiles the state-space and the Bayesian recursion integrals are then approximated with large sums over the grid. These methods are computationally very intensive and as the dimensionality becomes even moderately high, computational effort becomes impractically large.

Sequential Monte Carlo methods (Doucet et al., 2001) also approximate the Bayesian integrals with finite sums, but unlike direct numerical integration methods the integration is done over an adaptive stochastic grid, which makes these methods practically implementable. These methods also do not make any explicit assumptions about the form of the distributions, and are therefore usable in possibly nonlinear and non-Gaussian systems.

Variational Bayesian methods approximate the true posterior distribution with a tractable approximate form. A lower bound on the likelihood of the posterior is then maximised with respect to the free parameters of this tractable approximation, through the use of Jensen's inequality and variational calculus (Van der Merwe, 2004). These methods are however not as popular as Gaussian approximate

and sequential Monte Carlo methods.

1.3 CONTRIBUTION

This work contributes to the implementation of peripheral control tools on a ROM ore milling circuit, which is usually plagued by the presence of strong external disturbances, poor process models and unavailable process variable measurements.

External disturbance rejection is done under a disturbance observer framework. An integer order disturbance observer is implemented, similar to that of Chen et al. (2009) as well as a fractional order disturbance observer (FO-DOB). Fractional order disturbance observers have previously been used in other applications, but is implemented here for the milling circuit to investigate the impact it has on controller performance. A novel Bode ideal cut-off disturbance observer (BICO-DOB) that makes use of a Bode ideal cut-off (BICO) filter is introduced and implemented, and the performance achieved is compared with that of the other two disturbance observer schemes.

A closed-form expression for the mismatch that may be present between the model and the actual plant is presented in this work. This expression is however only valid for a controller that can be represented by a transfer function.

Model-plant mismatch detection based on the partial correlation analysis method introduced by Badwe et al. (2009) is also implemented. This method is applicable to advanced control schemes, such as MPC, in which the controller does not have a transfer function. This allows timely detection of significant mismatch, as well as identifies the transfer function matrix element that contains the mismatch, which allows the application of partial process re-identification, as opposed to full process re-identification.

Combined state and parameter estimation for the mill is implemented through the use of particle filters. This is done based on a simultaneous estimation scheme as well as a dual estimation scheme and the performances achieved are compared. Simultaneous estimation, also referred to as joint estimation, is a well-known scheme for combined state and parameter estimation. The implementation of dual particle filtering for combined state and parameter estimation has however, to the knowledge of the author, not previously been implemented. A sensitivity analysis is then performed to identify the class of systems for which dual estimation produces superior results to simultaneous estima-

tion.

1.4 PUBLICATIONS

The following publications have resulted from this work:

- L.E. Olivier and I.K. Craig. "Parameter mismatch detection in a Run-of-Mine ore milling circuit under model predictive control," in *Proc. 18th IFAC World Congress, Milan*, 2011, pp. 9929 – 9934.

- L.E. Olivier, I.K. Craig and Y.Q. Chen. "Fractional order disturbance observer for a Run-of-Mine ore milling circuit," in *Proc. 10th IEEE Africon, Livingstone, Zambia*, 2011.

- L.E. Olivier, I.K. Craig and Y.Q. Chen. "Fractional order and BICO disturbance observers for a Run-of-Mine ore milling circuit," *Journal of Process Control*, 2012.

- L.E. Olivier, B. Huang and I.K. Craig. "Dual particle filters for state and parameter estimation with application to a Run-of-Mine ore mill," *Journal of Process Control*, 2012.

1.5 ORGANISATION

This dissertation consists firstly of a discussion of the ROM ore milling circuit in Chapter 2. The suppression of the effect of external disturbances through the use of a disturbance observer, and the way in which this concept may be extended to fractional order and BICO cases, is presented in Chapter 3. The effect on controller performance and the detection of MPM is presented in Chapter 4. In Chapter 5 state and parameter estimation are discussed before some concluding remarks and further suggestions are given in the concluding Chapter 6.

CHAPTER 2 RUN-OF-MINE ORE MILLING

2.1 INTRODUCTION

Run-of-mine ore milling is usually the first and most expensive unit operation in the extraction of precious metals from mined ore (Craig and MacLeod, 1995).

Several phenomenological models of varying complexity have been proposed to describe the operation of milling circuits. One package for simulating comminution circuits is JKSimMet as described by Morrison and Richardson (2002). This uses a complex model for circuit simulation with good accuracy, and is described by Morrison and Richardson (2002) as an industry-accepted tool for analysis, optimisation and design of comminution circuits.

For control purposes however, such complex models are not always meaningful. This is because advanced control techniques often require the states of the process for control and a model with too many states introduces unnecessary complexity.

A phenomenological model of reduced complexity that uses five states to describe the flow of material in the milling circuit is presented by Coetzee (2009); Coetzee et al. (2010). This non-linear model is used as the basis on which the proposed peripheral control tools are tested.

Section 2.2 will give a brief overview of the ROM ore milling operation and the rest of this chapter will describe the reduced complexity phenomenological model of Coetzee (2009); Coetzee et al. (2010), as well as give some general milling circuit operating practices.

2.2 PROCESS DESCRIPTION

The process of milling ROM ore into a fine product is illustrated by the layout of the milling circuit in Fig. 2.1. The description that follows will assume the milling of gold-bearing ore by a semi-

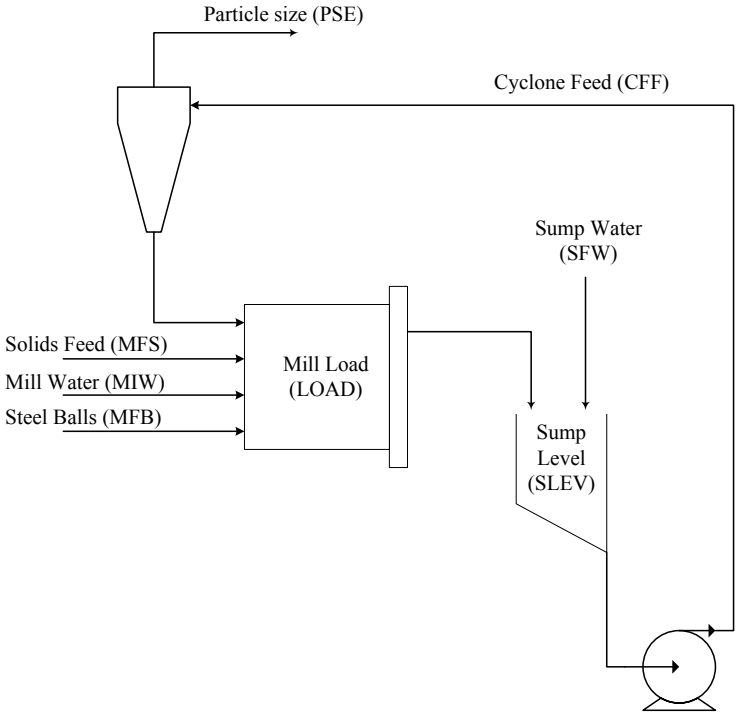

Figure 2.1: Run-of-mine ore milling circuit.

autogenous (SAG) mill in closed circuit.

Gold-bearing ore is transported from the mine and fed into the milling circuit from a conveyor belt at about 100 tons/hour. After mining no pre-processing is done on the ore before it enters the mill, hence the name run-of-mine ore. A typical mill has a length of 9 m and a diameter of 5 m (Stanley, 1987). The mill grinds the feed ore down to a product with a particle size of 80% smaller than 75 μm ($P_{80} = 75\mu$m) (Coetzee, 2009). Water is also fed into the mill, as well as steel balls, which help with the grinding of ore inside the mill. Sufficiently ground-down material, in the form of a slurry, is then discharged from the mill through an end-discharge grate into the sump. The discharge grate keeps large rocks and steel balls inside the mill while letting finely ground particles and water through.

The slurry is further diluted with water in the sump, which mainly acts as a buffer, before being pumped to a hydrocyclone for classification. The hydrocyclone has an internal diameter of 1 m. The out-of-specification material leaves the hydrocyclone as the underflow and is fed back into the mill for further re-grinding. The in-specification material leaves the hydrocyclone as the overflow; this is the product of the milling circuit that is fed to a downstream leaching process.

Table 2.1: Nomenclature used in the model equations.

Symbol	Description
Flow-rates used for internal flows: in (i), out (o), underflow (u)	
V_{wi}, V_{wo}, V_{wu}	Water $\left[\mathrm{m^3/h}\right]$
V_{si}, V_{so}, V_{su}	Solids $\left[\mathrm{m^3/h}\right]$
V_{fi}, V_{fo}, V_{fu}	Fines $\left[\mathrm{m^3/h}\right]$
V_{ci}, V_{co}, V_{cu}	Coarse (Solids - Fines) $\left[\mathrm{m^3/h}\right]$
V_{ri}, V_{ro}	Rocks $\left[\mathrm{m^3/h}\right]$
V_{bi}, V_{bo}	Balls $\left[\mathrm{m^3/h}\right]$
D_s, D_b	Density of feed ore (D_s) and steel balls (D_b) $\left[\mathrm{kg/m^3}\right]$

2.3 PROCESS MODEL

The full non-linear process model that is used to illustrate the performance of the peripheral control tools can be found in Coetzee (2009); Coetzee et al. (2010).

The model consists of separate modules for the feeder, mill, sump and hydrocyclone such that arbitrary circuit topologies may be constructed. The model uses five states, namely water, rocks, solids, fines and steel balls, to describe the flow of material through the milling circuit. Rocks are considered to be all the solid ore that cannot leave the mill through the end-discharge grate. All the ore than can leave the mill through the end-discharge grate is considered to be solids. Solids are then further divided into fines, the in-specification material, and coarse ore, the material that is small enough to exit through the mill end-discharge grate but not small enough to be considered in-specification. All the equations that constitute the non-linear model are based on these material classifications. Each of the modules that make up the milling circuit are described here. When reference is made to flows of constituents in the milling circuit, the nomenclature as given in Table 2.1 will be used.

The constraints and nominal values of the terms in the equations that will subsequently be used to describe the model are given in Table 2.2. All parameters and constants are given in Table 2.3.

Table 2.2: Constraints and operating point of the milling circuit.

Variable	Min	Max	OP	Description
X_{mw}	0	50	8.53	Hold-up of water in the mill [m^3]
X_{ms}	0	50	9.47	Hold-up of solids in the mill [m^3]
X_{mf}	0	50	3.54	Hold-up of fine ore in the mill [m^3]
X_{mr}	0	50	20.25	Hold-up of rocks in the mill [m^3]
X_{mb}	0	20	6.75	Hold-up of steel balls in the mill [m^3]
X_{sw}	0	40	21.99	Hold-up of water in the sump [m^3]
X_{ss}	0	30	5.84	Hold-up of solids in the sump [m^3]
X_{sf}	0	30	2.17	Hold-up of fine ore in the sump [m^3]
MIW	0	100	33.33	Flow-rate of water to the milling circuit [m^3/h]
MFS	0	200	100	Flow-rate of ore to the milling circuit [t/h]
MFB	0	4	2	Flow-rate of steel balls to the milling circuit [t/h]
CFF	400	500	442	Flow-rate of slurry to the cyclone [m^3/h]
SFW	0	400	267	Flow-rate of water to the sump [m^3/h]
PSE	60	90	80	Product particle-size [$\% < 75\mu m$]
LOAD	30	50	45	Total charge of the mill [%]
SLEV	2	37.5	30	Level of the sump [m^3]
φ	0	1	0.51	Rheology factor [dimensionless]
$THROUGHPUT$	0	200	200	Product throughput [t/h]
P_{mill}	0	2000	2000	Power draw of the mill motor [kW]

Table 2.3: Parameters and constants contained in the milling equations.

Parm	Nom	Min	Max	%Δ	Description
α_f	0.1	0.05	0.15	50	Fraction of fines in the ore [dimensionless]
α_r	0.1	0.05	0.15	50	Fraction of rocks in the ore [dimensionless]
ϕ_f	28	14	42	50	Power needed per ton of fines produced [$^{kW \cdot h}/_{t}$]
ϕ_r	69	55	83	20	Rock abrasion factor [$^{kW \cdot h}/_{t}$]
ϕ_b	94	89	99	5	Steel abrasion factor [$^{kW \cdot h}/_{t}$]
ε_{ws}	0.6	-	-	-	Maximum water-to-solids volumetric flow at zero pulp flow [dimensionless]
V_V	40	-	-	-	Volumetric flow per “flowing volume" driving force [h^{-1}]
P_{max}	2000	-	-	-	Maximum mill motor power [kW]
δ_{P_v}	1	-	-	-	Power change parameter for volume [dimensionless]
δ_{P_s}	1	-	-	-	Power change parameter for fraction solids [dimensionless]
$v_{P_{max}}$	0.45	-	-	-	Fraction of mill volume filled for maximum power [dimensionless]
$\varphi_{P_{max}}$	0.51	-	-	-	Rheology factor for maximum mill power [dimensionless]
α_{speed}	1.0	0.7	1.0	-	Fraction of critical mill speed [dimensionless]
α_P	0.82	-	-	-	Fractional power reduction per fractional reduction from maximum mill speed [dimensionless]
v_{mill}	100	-	-	-	Mill volume [m^3]
α_{ϕ_f}	0.01	-	-	-	Fractional change in kW/fines produced per change in fractional filling of mill [dimensionless]
χ_P	0	-	-	-	Cross term for maximum power [dimensionless]
ε_c	184	175	193	5	Coarse split [dimensionless]
α_{su}	0.16	0.15	0.17	5	Fraction solids in underflow [dimensionless]
C_1	0.6	-	-	-	Constant [dimensionless]
C_3	0.7	-	-	-	Constant [dimensionless]
C_4	3	-	-	-	Constant [dimensionless]
C_5	3	-	-	-	Constant [dimensionless]

2.3.1 Feeder module

The feeder is a simple unit that takes the total feed of material into the mill and divides it into the respective streams of the states of material inside the milling circuit. This simplistic model may be replaced by a model for an actual feeder such as a vibratory feeder. The flow-rates of the various constituents from the feeder are given by

$$V_{wo} \triangleq MIW \tag{2.1}$$

$$V_{so} \triangleq \frac{MFS}{D_s}(1-\alpha_r) \tag{2.2}$$

$$V_{fo} \triangleq \alpha_f \cdot \frac{MFS}{D_s} \tag{2.3}$$

$$V_{ro} \triangleq \alpha_r \cdot \frac{MFS}{D_s} \tag{2.4}$$

$$V_{bo} \triangleq \frac{MFB}{D_b}. \tag{2.5}$$

2.3.2 Mill module

The mill module has the capability of modelling various types of mills such as rod, ball, semi-autogenous grinding (SAG) and autogenous grinding (AG) mills. The mill receives the ore, balls and water and the model describes how the material proceeds through the mill according to the states of the model. The mill has five states, which are the hold-ups of the five material classifications used by the model, namely water, solids, fines, rocks and balls. The model includes the effect of the mill power and slurry rheology (Shi and Napier-Munn, 2002) on the breakage and power functions. The rheology factor, which relates to the fluidity of the slurry inside the mill, is defined as

$$\varphi \triangleq \left(\frac{\max\left[0, \left(X_{mw} - \left(\frac{1}{\varepsilon_{ws}} - 1\right) X_{ms}\right)\right]}{X_{mw}} \right)^{0.5}. \tag{2.6}$$

This factor varies between 0 for slurry that has a very low fluidity, e.g. thick mud, and 1 for high fluidity, such as for pure water. The mill power is defined as

$$P_{mill} = P_{max} \cdot \{1 - \delta_{P_v} Z_x^2 - 2\chi_p \delta_{P_v} \delta_{P_s} Z_x Z_r - \delta_{P_s} Z_r^2\} \cdot \left(\alpha_{speed}\right)^{\alpha_P} \tag{2.7}$$

where $Z_x = (X_{mw} + X_{ms} + X_{mr} + X_{mb}) / (v_{P_{max}} \cdot v_{mill} - 1)$ is the effect of the load inside the mill on power consumption and $Z_r = (\varphi / \varphi_{P_{max}} - 1)$ is the effect of the slurry rheology on the power consumption.

Inside the mill rocks will be ground down and may change state into solids and subsequently fines. The consumption of rocks inside the mill is given by

$$RC \triangleq \frac{1}{D_s \phi_r} \cdot P_{mill} \cdot \varphi \cdot \left(\frac{X_{mr}}{X_{mr} + X_{ms}} \right). \tag{2.8}$$

The production of fines inside the mill is given by

$$FP \triangleq \frac{P_{mill}}{D_s \phi_f \left[1 + \alpha_{\phi_f} \left(\frac{LOAD}{v_{mill}} - v_{P_{max}} \right) \right]} \tag{2.9}$$

where $LOAD = X_{mw} + X_{ms} + X_{mr} + X_{mb}$ is the total amount of material inside the mill. No distinction is made between fines produced directly from rocks and fines produced from coarse ore. Steel balls that help with the grinding of ore inside the mill also grind away over time. This effect is also modelled and the consumption of steel balls in the mill is given by

$$BC \triangleq \frac{1}{D_b \phi_b} \cdot P_{mill} \cdot \varphi \cdot \left(\frac{X_{mr}}{X_{mr} + X_{ms}} \right). \tag{2.10}$$

The discharge flow rates of material from the mill are then given by

$$V_{wo} \triangleq V_V \cdot \varphi \cdot X_{mw} \left(\frac{X_{mw}}{X_{mw} + X_{ms}} \right) \tag{2.11}$$

$$V_{so} \triangleq V_V \cdot \varphi \cdot X_{mw} \left(\frac{X_{ms}}{X_{mw} + X_{ms}} \right) \tag{2.12}$$

$$V_{fo} \triangleq V_V \cdot \varphi \cdot X_{mw} \left(\frac{X_{mf}}{X_{mr} + X_{ms}} \right) \tag{2.13}$$

and $V_{ro} = 0$ as well as $V_{bo} = 0$ because no rocks or balls can escape through the discharge grate. The time derivatives for the hold-ups of the various constituents of the mill charge are given by

$$\frac{dX_{mw}}{dt} \triangleq V_{wi} - V_{wo} \tag{2.14}$$

$$\frac{dX_{ms}}{dt} \triangleq V_{si} - V_{so} + RC \tag{2.15}$$

$$\frac{dX_{mf}}{dt} \triangleq V_{fi} - V_{fo} + FP \tag{2.16}$$

$$\frac{dX_{mr}}{dt} \triangleq V_{ri} - RC \tag{2.17}$$

$$\frac{dX_{mb}}{dt} \triangleq V_{bi} - BC. \tag{2.18}$$

2.3.3 Sump module

The sump mainly acts as a buffer between the mill and the hydrocyclone. The sump module used here is that of a mixed sump, where it is assumed that the water, fine ore and coarse ore are fully mixed.

The sump only has three states for the hold-up values of the three material classifications found in the sump, namely water, fines and solids. This is because it has been assumed that no rocks or steels balls are discharged from the mill. The time derivatives for the hold-ups of the various constituents of the slurry in the sump are given by

$$\frac{dX_{sw}}{dt} \triangleq V_{wi} + SFW - V_{wo} \tag{2.19}$$

$$\frac{dX_{ss}}{dt} \triangleq V_{si} - V_{so} \tag{2.20}$$

$$\frac{dX_{sf}}{dt} \triangleq V_{fi} - V_{fo}. \tag{2.21}$$

The discharge flow-rates of the constituents of the slurry from the sump are

$$V_{wo} \triangleq CFF \cdot \left(\frac{X_{sw}}{X_{ss} + X_{sw}} \right) \tag{2.22}$$

$$V_{so} \triangleq CFF \cdot \left(\frac{X_{ss}}{X_{ss} + X_{sw}} \right) \tag{2.23}$$

$$V_{fo} \triangleq CFF \cdot \left(\frac{X_{sf}}{X_{ss} + X_{sw}} \right). \tag{2.24}$$

The sump level is then given by

$$SLEV \triangleq X_{sw} + X_{ss} \tag{2.25}$$

and the cyclone feed density is defined as

$$CFD \triangleq X_{sw} + D_s \cdot \left(\frac{X_{ss}}{X_{ss} + X_{sw}} \right). \tag{2.26}$$

2.3.4 Hydrocyclone module

The hydrocyclone is a classification device that splits the input slurry with regard to weight. This relates to the size of the particles through the density. Smaller particles are forced out of the top of the cyclone as the overflow and is the product of the milling circuit. Heavier particles are forced out of the bottom of the cyclone as the underflow and are fed back into the mill for further grinding. The model used here for the hydrocyclone module is based on the empirical hydrocyclone models of Plitt and Nageswararao, as discussed in Nageswararao et al. (2004). The product size and density are accurately modelled by taking into account the effects of angular velocity of the particles inside the cyclone, slurry density and viscosity. The flow-rates of the constituents of the slurry at the underflow

of the cyclone are given by

$$V_{cu} \triangleq V_{ci} \cdot \left(1 - C_1 \cdot e^{-V_{ti}/\varepsilon_c}\right) \cdot \left(1 - \left[\frac{F_i}{C_3}\right]^{C_4}\right) \cdot \left(1 - P_i^{C_5}\right) \tag{2.27}$$

$$V_{wu} \triangleq V_{wi} \cdot \frac{V_{cu} - F_u \cdot V_{cu}}{F_u \cdot V_{wi} + F_u \cdot V_{fi} - V_{fi}} \tag{2.28}$$

$$V_{fu} \triangleq V_{fi} \cdot \frac{V_{cu} - F_u \cdot V_{cu}}{F_u \cdot V_{wi} + F_u \cdot V_{fi} - V_{fi}} \tag{2.29}$$

where

$$F_u \triangleq 0.6 - (0.6 - F_i) \cdot e^{-V_{cu}/\alpha_{su}\varepsilon_c}, \tag{2.30}$$

$$F_i \triangleq \frac{V_{si}}{V_{wi} + V_{si}} \tag{2.31}$$

and

$$P_i \triangleq \frac{V_{fi}}{V_{si}}. \tag{2.32}$$

The product particle size is defined as

$$PSE \triangleq \frac{V_{fo}}{V_{co} + V_{fo}} \tag{2.33}$$

where $V_{fo} \triangleq V_{fi} - V_{fu}$ and $V_{co} \triangleq V_{ci} - V_{cu}$. The product throughput is given by THROUGHPUT $\triangleq V_{co} + V_{fo}$.

2.4 OPERATING PHILOSOPHY

The milling circuit is typically operated to keep the product particle size (PSE) and mill load (LOAD) at specific set-points, the values of which are subsequently discussed. The sump level (SLEV) is not as critical and should generally just be kept from overflowing or running dry.

Fig. 2.2 shows the residue (gold not recovered) as a function of the particle size for leaching. This figure was derived by Craig et al. (1992) from industrial data. The figure shows that better recovery (i.e. a smaller residue) is achieved for a finer grind. Grinding the ore too fine does however decrease the throughput and increases grinding cost. The ore from the milling circuit should therefore be ground as finely as possible while maintaining the throughput and limiting grinding costs. It is for this reason that the particle size from a milling circuit is usually specified as a certain value, and should be maintained as consistently as possible at that value.

Similarly, a performance function for recovery as a function of the product particle size is given by Wei (2010) for flotation. This was reproduced and is shown here as Fig. 2.3. Here it is seen that maximum recovery is achieved for a specific particle size.

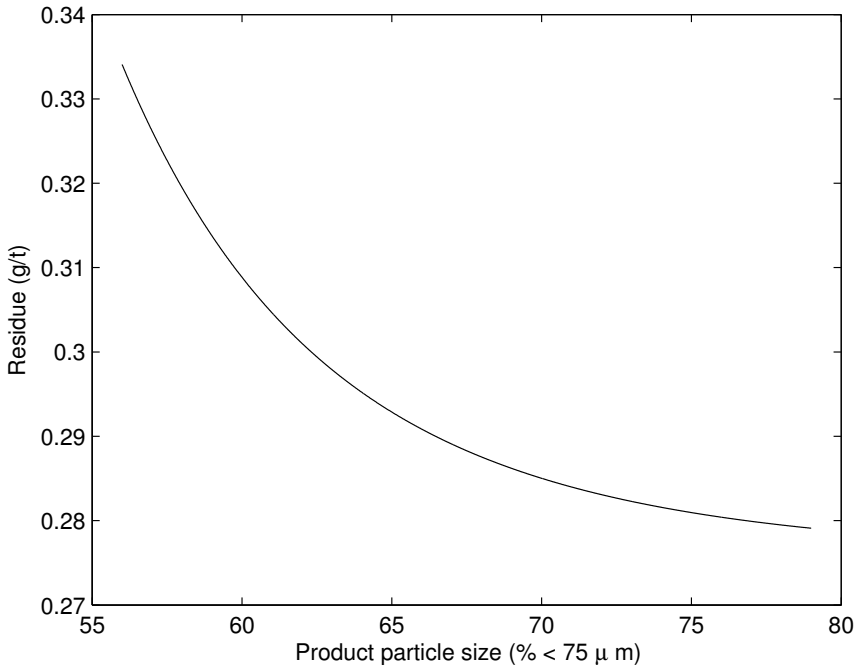

Figure 2.2: Recovery curve for leaching (reproduced from Craig et al. (1992)).

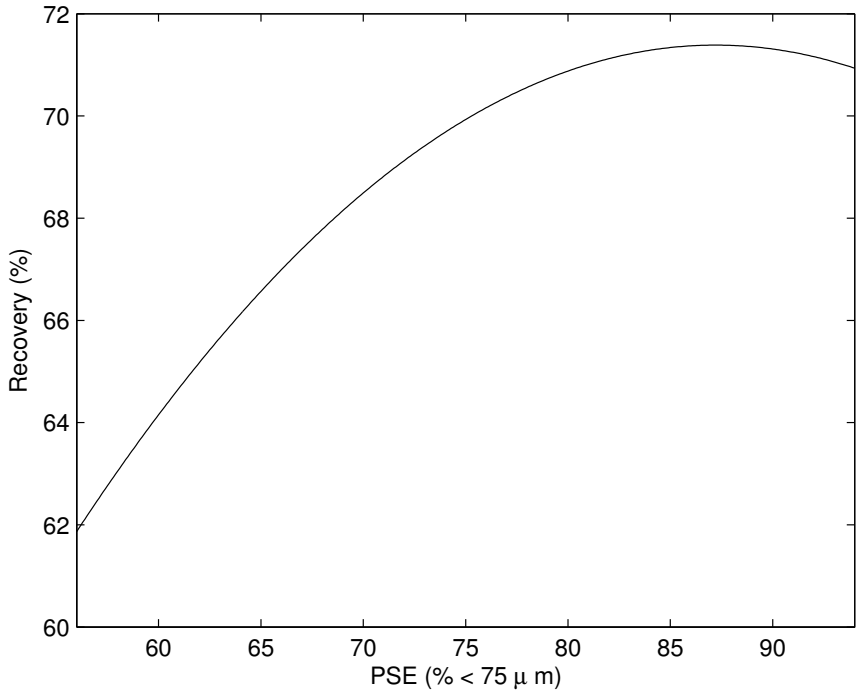

Figure 2.3: Recovery curve for flotation (reproduced from Wei (2010)).

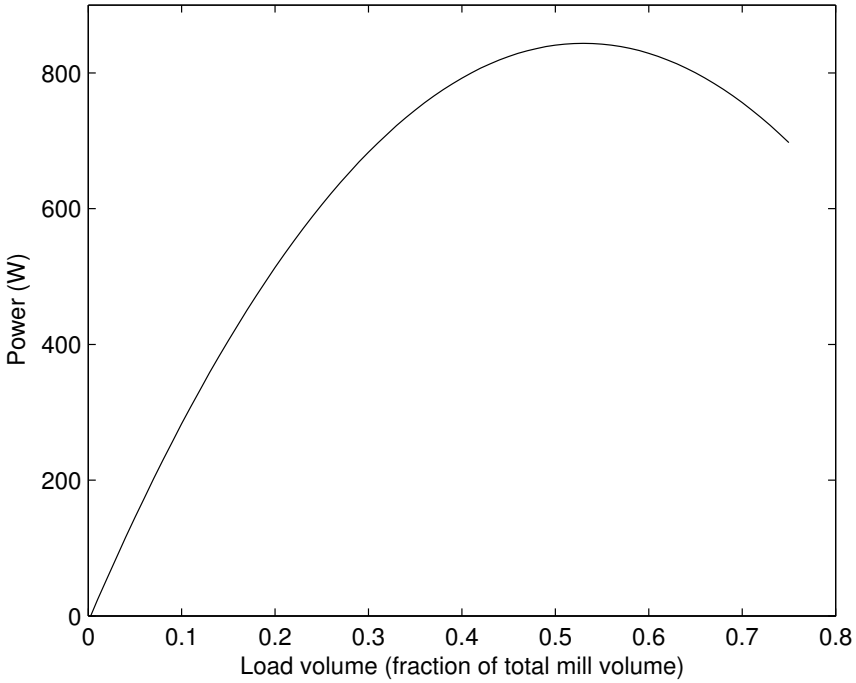

Figure 2.4: Mill power as a function of the load (reproduced from Van Nierop and Moys (2001)).

In both cases, for leaching and flotation, a specific particle size is therefore specified to maximise recovery but for leaching this is subject to the requirement that throughput should be maintained.

When comparing the value of the precious metal recovered to the grinding cost, it is usually seen that maximising throughput far outweighs grinding cost. It is for this reason that the mill is often run at maximum power such that throughput can be maximised with little regard for the power usage this incurs. There is also a strong relationship between the load inside the mill and the power used. Fig. 2.4 shows the mill power as a function of the load volume (given as a fraction of the total mill volume) as shown by Van Nierop and Moys (2001).

From Fig. 2.4 it is clear that there is a specific value for the fractional filling of the total mill volume at which mill power is maximised. The form of this power-load curve is general, although the actual load value for which the maximum is achieved depends on various factors. The load set-point is then specified to be at this point; for the phenomenological model used in this work, LOAD is specified to be 0.45.

The rheology factor as defined in Section 2.3.2 also has a large impact on the operation of the mill. A small rheology factor indicates that the slurry contains very little water and will be difficult to pump,

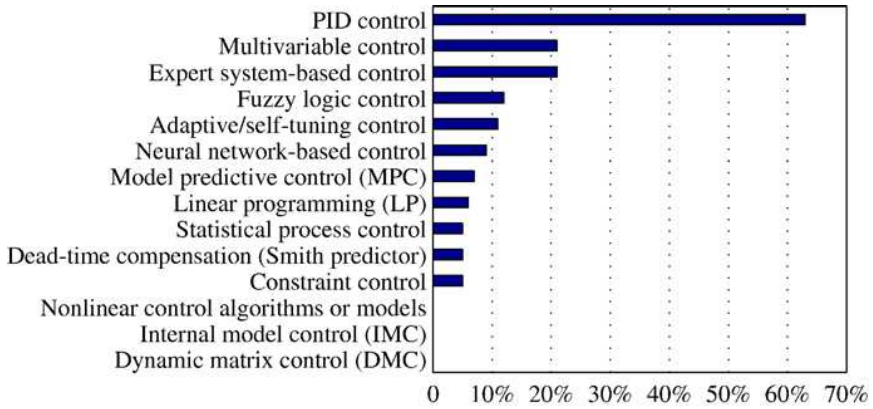

Figure 2.5: Control technologies used in milling circuits (from Wei (2010)).

while a large rheology factor indicates that too much water is present in the slurry and most of the pumping effort is simply to circulate water as opposed to circulating liberated minerals. There is a rheology factor at which maximum power for the mill is achieved and it is desirable to operate the mill at this rheology factor. The rheology factor is however not controlled as such, but a certain rheology factor can be more or less maintained by adding water into the mill at a constant ratio to the amount of solid ore being fed into the mill.

The load inside the mill and the sump level are both given by the integral of the nett amount of material flowing into the mill and sump respectively. Both of these quantities are therefore open-loop unstable in a bounded-input bounded-output (BIBO) sense. At the very least, stabilising control for the load and sump level are required. Other than this, control of the product particle size is also important to ensure the efficient operation of the downstream extraction process. The control objective can therefore be reduced to be: control the product particle size to a constant value while maintaining maximum throughput. In order to maintain the throughput, the load is to be controlled as discussed earlier, and the sump level should be stabilised.

Fig. 2.5 shows the control technologies used on milling circuits, as reported by Wei (2010). It is clear from the figure that PI(D) control is by far the most common control technology used on grinding mill circuits.

Decentralised control is common in which manipulated variables are grouped with controlled variables in SISO pairings (Hodouin, 2011). The simplicity of this control topology, coupled with the little knowledge needed to implement PID control as opposed to some advanced control scheme, is

listed by Wei and Craig (2009) as the most probable cause for the widespread use of PID control on milling circuits.

CHAPTER 3 DISTURBANCE OBSERVERS FOR EXTERNAL DISTURBANCE REJECTION

3.1 INTRODUCTION

One of the reasons why run-of-mine ore milling circuits are generally difficult to control is the presence of strong external disturbances. In order to ensure that material of consistent composition is passed to the downstream extraction process, disturbance rejection for the milling circuit is important.

Currently PID control is the most common control technology used in milling circuits (Wei and Craig, 2009). This is different in general from the process industries where MPC is most common (Bauer and Craig, 2008). This might be attributed to the fact that PID control is much easier to understand and maintain than an advanced control method such as MPC, even though the advantages of using MPC over PID in grinding circuit control are well documented (see for instance Ramasamy et al. (2005)). MPC does however struggle to produce good results in the presence of strong external disturbances and severe model-plant mismatch (Chen et al., 2009).

Disturbance observer based control helps to compensate for model-plant mismatch and external disturbances, both of which are common in ROM ore milling circuit control. A DOB has already been applied to a grinding mill circuit (Chen et al., 2009) with good results. The main contribution of this chapter is the presentation of the application of a fractional order disturbance observer (FO-DOB) to the ROM ore milling circuit (Olivier et al., 2011a), as well as the introduction of a novel BICO-DOB and the application thereof in the control of the milling circuit (Olivier et al., 2011b). These make use of a fractional order Q-filter and a BICO Q-filter respectively. This offers additional tuning freedom to optimise performance in the presence of strong external disturbances and severe MPM. These disturbance observers are implemented in conjunction with a standard PI controller to show how these schemes may improve controller performance.

Section 3.2 will first discuss the linearised milling circuit model used for the PI controller design as discussed in Section 3.3. The rest of the chapter will then be used to discuss the concept of using a disturbance observer, the implementation of a fractional order disturbance observer and finally the implementation of a BICO disturbance observer.

3.2 LINEARISED MILLING CIRCUIT MODEL

Model based milling circuit controllers, such as linear MPC, and common PI(D) tuning rules require a linearised plant model. Such a linear model of the plant is obtained through applying a standard system identification (SID) procedure as described by Ljung (1999), to the milling circuit model described in Chapter 2 around the operating point given in Table 2.2. The final linearised model for control is given by:

$$\begin{bmatrix} \Delta PSE \\ \Delta LOAD \\ \Delta SLEV \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \begin{bmatrix} \Delta CFF \\ \Delta MFS \\ \Delta SFW \end{bmatrix}. \tag{3.1}$$

In (3.1), the transfer function elements are (with all time constants and delays in hours):

$$\begin{aligned} g_{11} &= \tfrac{-3.63\times10^{-4}(-0.93s+1)}{(0.81s+1)(0.02s+1)}\exp(-0.011s), \\ g_{12} &= \tfrac{-0.0048}{1.02s+1}\exp(-0.064s), \\ g_{13} &= \tfrac{7.52\times10^{-4}}{0.60s+1}\exp(-0.011s), \\ g_{21} &= \tfrac{0.0173(7.6276s+1)}{(1.72s+1)(69.4s+1)}\exp(-0.014s), \\ g_{22} &= \tfrac{0.00375}{s}, \\ g_{23} &= \tfrac{-0.0025}{5.25s+1}\exp(-0.014s), \\ g_{31} &= \tfrac{-0.5651}{s}\exp(-0.014s), \\ g_{32} &= \tfrac{0.3611}{s}, \\ g_{33} &= \tfrac{0.6119}{s}. \end{aligned} \tag{3.2}$$

The model makes no reference to the manipulated variables MIW and MFB. In this study the value of MFB is kept constant at its nominal value of 2 [t/h]. The value of MIW is derived from the value of MFS such that a constant water-to-solids ratio is maintained for the feed into the mill, as discussed by Coetzee (2009).

The linearised plant model (3.2) is only used for controller design. The simulations that follow later

in this chapter are all based on the full nonlinear model of the ROM ore milling circuit discussed in Chapter 2.

3.3 PI CONTROLLER DESIGN

Decentralised PID control is the most common approach to feedback control in multi-variable mineral processing plants (Wei and Craig, 2009). Variables are usually coupled in SISO loops (Hodouin, 2011). Traditionally the input-output pairings on milling circuits are $SFW \rightarrow PSE$, $MFS \rightarrow LOAD$, and $CFF \rightarrow SLEV$ (Coetzee, 2009). It was however found by Coetzee (2009) that these pairings give rise to significant couplings that cause the sump to either overflow or run dry as soon as significant ore hardness and composition disturbances were introduced.

In this study the manipulated variables are paired with the controlled variables as $CFF \rightarrow PSE$, $MFS \rightarrow LOAD$, and $SFW \rightarrow SLEV$ for which Coetzee (2009) found much better robustness to feed disturbances that result in actuators saturating. The internal model control (IMC) tuning rules as presented in Seborg et al. (2003, p.308) are then applied to obtain three PI controllers. Pairing variables in SISO loops and independently designing a PI controller for each is not generally the best method for decentralised controller design. Despite the drawbacks of this method, it does usually satisfy metallurgical needs if the variable pairings are well chosen (Hodouin, 2011) and coupled with its simplicity makes it a common choice (Wei and Craig, 2009).

The IMC-based controller settings, as used here, explicitly specify a PI(D) controller through analytic tuning relations such that the closed-loop response takes the form of a low-pass filter

$$G_{cl,d} = \frac{1}{(\tau_c s + 1)^r}. \tag{3.3}$$

$G_{cl,d}$ is the desired closed-loop response, τ_c is the desired time constant for the response, which is specified to be 0.01 according to Coetzee (2009), and r is the order of the response, which is usually chosen to be 1 (Seborg et al., 2003), as was also done here. The aim is to derive tuning parameters that are applicable to the parallel form of representing a PI controller as

$$G_c(s) = K_c \left(1 + \frac{1}{\tau_I s}\right). \tag{3.4}$$

Controller for CFF $\rightarrow$ PSE **loop**

The model for the $CFF \rightarrow PSE$ loop is given by

$$g_{11} = \frac{-3.63 \times 10^{-4}(-0.93s+1)}{(0.81s+1)(0.02s+1)} \exp(-0.011s) \tag{3.5}$$

which is written into the the standard form

$$g_{11} = \frac{K(-\tau_3 s+1)}{(\tau_1 s+1)(\tau_2 s+1)} \exp(-\theta s). \tag{3.6}$$

The gain of the PI controller is then found from Seborg et al. (2003) through

$$K_c K = \frac{\tau_1 + \tau_2 + \frac{\tau_3 \theta}{\tau_c+\tau_3+\theta}}{\tau_c + \tau_3 + \theta}, \tag{3.7}$$

with $\tau_c = 0.01$, to be $K_{c,11} = 2.229 \times 10^3$. The integral time constant is found from Seborg et al. (2003) through

$$\tau_I = \tau_1 + \tau_2 + \frac{\tau_3 \theta}{\tau_c + \tau_3 + \theta} \tag{3.8}$$

to be $\tau_{c,11} = 0.84$.

Controller for MFS $\rightarrow$ LOAD **loop**

The model for the $MFS \rightarrow LOAD$ loop is given by

$$g_{22} = \frac{0.00375}{s} \tag{3.9}$$

which is in the the standard form

$$g_{22} = \frac{K}{s}. \tag{3.10}$$

The gain of the PI controller is then found from Seborg et al. (2003) through

$$K_c K = \frac{2}{\tau_c}, \tag{3.11}$$

with $\tau_c = 0.01$, to be $K_{c,22} = 5.327 \times 10^4$. The integral time constant is found from Seborg et al. (2003) through

$$\tau_I = 2\tau_c \tag{3.12}$$

to be $\tau_{c,22} = 0.02$.

Table 3.1: Tuning parameters for PI controllers.

Control loop	K_c	τ_I
$CFF \rightarrow PSE$	2.229×10^3	0.84
$MFS \rightarrow LOAD$	5.327×10^4	0.02
$SFW \rightarrow SLEV$	3.269×10^2	0.02

Controller for SFW $\rightarrow$ SLEV **loop**

The model for the $SFW \rightarrow SLEV$ loop is given by

$$g_{33} = \frac{0.6119}{s} \tag{3.13}$$

which is, like the model for g_{22}, in the the standard form

$$g_{33} = \frac{K}{s}. \tag{3.14}$$

The gain of the PI controller is then found from Seborg et al. (2003) through

$$K_c K = \frac{2}{\tau_c}, \tag{3.15}$$

with $\tau_c = 0.01$, to be $K_{c,33} = 3.269 \times 10^2$. The integral time constant is found from Seborg et al. (2003) through

$$\tau_I = 2\tau_c \tag{3.16}$$

to be $\tau_{c,33} = 0.02$.

The tuning parameters for all three controllers are summarised in Table 3.1.

3.4 DISTURBANCE OBSERVER BASED CONTROL

Disturbance observers have several attractive features. In the absence of large modelling errors, DOBs allow independent tuning of disturbance rejection and command following characteristics. Furthermore, compared to integral action, disturbances observers are more flexible, as they allow for the selection of the order, relative degree, and the bandwidth of the low-pass filter known as the disturbance observer filter or the Q-filter (see Fig. 3.1). Although it is known that by appending disturbance states to a traditional state estimator (Franklin et al., 1990) the disturbance compensation can be easily

handled, a DOB is in fact more welcomed by control practitioners because the DOB structure allows simple and intuitive tuning of the DOB loop gains independent of the state feedback gains.

As pointed out by Chen et al. (2004); Kempf and Kobayashi (1999), there is a trade-off between the phase margin loss and the strength of low frequency disturbance attenuation when applying a DOB. Given the required cut-off frequency of the Q-filter, it turns out that the relative degree of the Q-filter is the major tuning knob for this trade-off. As a motivation for the fractional order Q-filter, a solution based on an integer order Q-filter with variable relative degrees is introduced, which is the key contribution of Chen et al. (2001). A fractional order disturbance observer based on the fractional order Q-filter is presented, similar to the filter used by Chen et al. (2004).

A DOB in a process control context can be found in Zhu et al. (2005) for generic processes. However, for grinding mill processes, the application of a DOB presented by Chen et al. (2009) is more recent.

3.5 INTEGER ORDER DISTURBANCE OBSERVER

The layout of a conventional disturbance observer in a control loop is presented in Fig. 3.1. Signals $C(s)$, $U(s)$, $D_{ex}(s)$, $Y(s)$, $Y_{sp}(s)$, and $N(s)$ are the control signal, manipulated variable, external disturbance, controlled variable, set-point, and noise signal values respectively. The signals $\hat{D}(s)$ and $\hat{D}_f(s)$ are the estimates of the lumped disturbances before and after being filtered by the Q-filter. $G_c(s)$ is the controller, $P(s)$ is the actual plant, $G(s)$ is the nominal plant model (as shown in (3.2)) and $Q(s)$ is the Q-filter.

$\hat{D}(s)$ is given by

$$\hat{D} = G^{-1}(Y+N) - U \tag{3.17}$$

where reference to the Laplace variable s has been dropped for ease of representation. If the actual plant $P(s)$ is then equal to the plant model $G(s)$, the expression becomes

$$\begin{aligned} \hat{D} &= G^{-1}(P(D_{ex}+U)-N)-U && (3.18)\\ &= D_{ex}+U-G^{-1}N-U && (3.19)\\ &= D_{ex}-G^{-1}N, && (3.20) \end{aligned}$$

which gives a good estimate of the true external disturbance, except for the $G^{-1}N$ component which is due to noise. This suggests that the disturbance estimate should be filtered before being used for

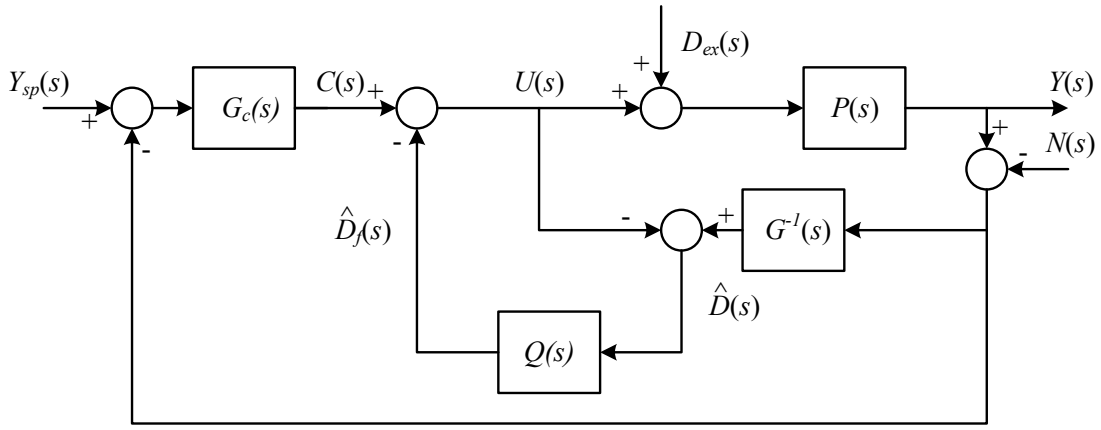

Figure 3.1: Block diagram of a conventional disturbance observer.

disturbance compensation.

When writing the output as (Chen et al., 2009)

$$Y(s) = G_{cy}(s)C(s) + G_{dy}(s)D_{ex}(s) + G_{ny}(s)N(s) \tag{3.21}$$

with

$$G_{cy}(s) = \frac{P(s)G(s)}{G(s) + (P(s) - G(s))Q(s)} \tag{3.22}$$

$$G_{dy}(s) = \frac{P(s)G(s)(1 - Q(s))}{G(s) + (P(s) - G(s))Q(s)} \tag{3.23}$$

$$G_{ny}(s) = \frac{P(s)G(s)}{G(s) + (P(s) - G(s))Q(s)} \tag{3.24}$$

it becomes clear that the Q-filter is the essence of the design of the disturbance observer. When $Q(s) \to 1$, then from (3.22) - (3.24),

$$G_{cy} \to G(s),\ G_{dy} \to 0,\ G_{ny} \to 1. \tag{3.25}$$

When $Q(s) \to 0$, then from (3.22) - (3.24),

$$G_{cy} \to P(s),\ G_{dy} \to P(s),\ G_{ny} \to 0. \tag{3.26}$$

In the light of the previous two equations, the Q-filter is designed to be in the form of a low-pass filter such that the lumped estimate of the disturbance approximately equals the actual lumped disturbance in the low-frequency range, i.e. when $Q(s) \to 1$ disturbances are rejected ($G_{dy} \to 0$). In the high frequency range noise is filtered out, i.e. when $Q(s) \to 0 \Rightarrow G_{ny} \to 0$.

Even though the proposed scheme might seem promising at this stage, the inverse of the nominal plant model (G^{-1}) is usually not realisable (Skogestad and Postlethwaite, 2005). This is due to the inverse of the nominal model usually having more zeros than poles, and the predictor element resulting from taking the inverse of a time delay that is usually present in the original model. In order to make the disturbance observer realisable, two techniques are employed. Firstly the model elements to be inverted are factored as

$$G_{ii} = G_+ G_-. \tag{3.27}$$

G_{ii} contains only the diagonal elements of G, and only these elements need to be inverted such that the lumped disturbance estimate will contain the uncompensated coupling effects as shown later in this section. G_+ contains any time delays and right-half plane zeros and is required to have a steady-state gain equal to 1 to ensure that the two factors in (3.27) are unique. This is similar to what is done when designing an IMC controller (Seborg et al., 2003). Only the minimum phase component of the model (i.e. G_-) is then inverted.

Secondly, the layout of the DOB is modified such that the inverse of the minimum-phase component of the model is cascaded with the Q-filter. The Q-filter is then specified with a minimum order based on the relative order of the plant model. The modified disturbance observer is shown in Fig. 3.2. It should also be noted from Fig. 3.2 that cascading the Q-filter with the inverse of the minimum-phase component of the model requires the addition of the same Q-filter in the left-side branch before the summing junction that produces $\hat{D}_f(s)$, such that the transfer function of the modified DOB remains equivalent to the original DOB. At this same summing junction, signals that travelled through the plant will have been delayed by any time delays present in the plant. The signals simply travelling through the Q-filter on the left should therefore also be delayed such that both sets of signals are delayed by similar amounts.

These modifications do imply that only G_- is now used as the effective plant model as opposed to G. G_- therefore becomes the model that the DOB wants to maintain and any difference between G_- and the actual plant P, as will inherently be present, will be lumped together with the estimate of the external disturbances. This is derived in Chen et al. (2009) and shown later in this section. It does mean that the disturbance estimate is not only of the external disturbances, as shown in (3.30), but is advantageous for controller performance.

The design of the DOB is independent of the design of the controller and thus the same PI controller as discussed in Section 3.3 can be used as the feedback controller for the DOB based control. Because

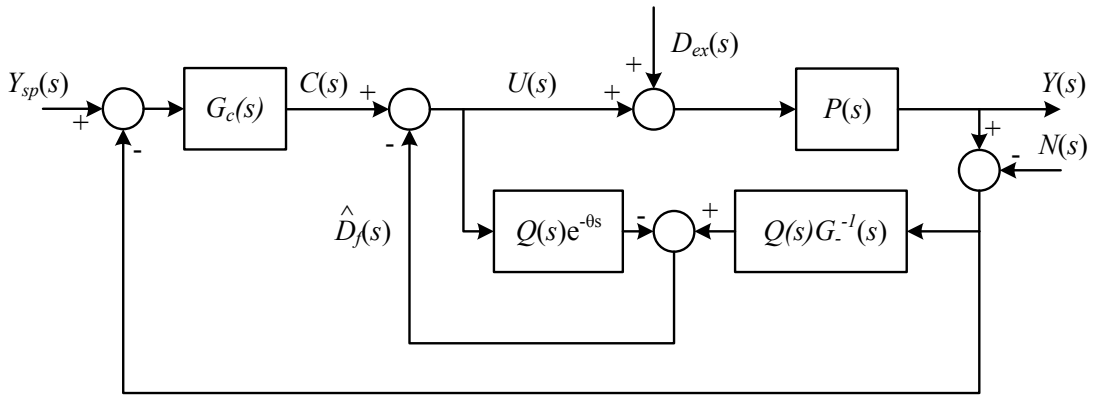

Figure 3.2: Block diagram of the modified disturbance observer.

this plant is represented by a three-input, three-output model the Q-filter is specified as a 3×3 matrix with diagonal elements of the form

$$Q_{ii} = \left(\frac{1}{\lambda_{ii}s+1}\right)^n \tag{3.28}$$

where λ_{ii} determines the filter bandwidth and n is the filter order ($n \in \mathbb{N}$). Based on the model presented in Section 3.2 the order of the Q-filter should be at least 2 such that the cascaded transfer function $Q(s)G_-^{-1}(s)$ is realisable.

As mentioned earlier in this section, only the diagonal elements of G_- are inverted and a diagonal Q-filter is used. This might intuitively give the impression that the internal plant couplings are merely discarded. Consider Fig. 3.3, which shows the modified disturbance observer for a single loop.

Here the diagonal element of the minimum phase part of the model is shown as $g_{-,11}(s)$ and the relevant diagonal element of the PI controller matrix is shown as $G_{c,11}(s)$. The output also includes the off-diagonal elements of the plant $p_{12}(s)$ and $p_{13}(s)$, which affect the output of this loop. The external disturbance affecting this loop has been moved to the output, which is equivalent to the situation shown in Fig. 3.2 if the disturbance is propagated through the appropriate disturbance transfer function $G_d(s)$. $D_{m,1}$ arises from any mismatch between g_{11} and p_{11} as

$$D_{m,1} = [p_{11}(s) - g_{11}(s)]U_1(s). \tag{3.29}$$

The model element $g_{11}(s)$ effectively becomes the plant which the disturbance observer maintains. The lumped disturbance estimate ($\hat{D}_f(s)$) given by the disturbance observer is

$$\hat{D}_f(s) = \hat{D}_{ex,1}(s) + \hat{D}_{m,1}(s) + \hat{D}_c(s) \tag{3.30}$$

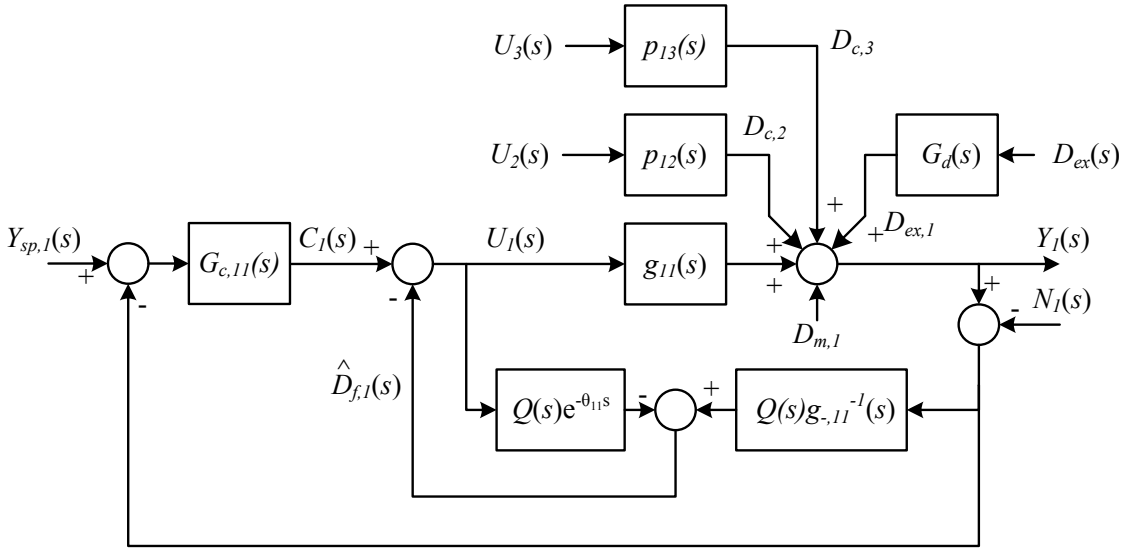

Figure 3.3: Block diagram of the modified disturbance observer for a single loop.

where $D_{ex,1}(s)$ is the external disturbance, $D_{m,1}(s)$ is the internal disturbance caused by MPM and $D_c(s) = D_{c,2}(s) + D_{c,3}(s)$ is the internal disturbance caused by unconsidered coupling effects. A similar result may be derived for the other two loops as well.

This is an interesting result because it shows that the disturbance observer will also help counter the effects of MPM and unconsidered coupling effects. It therefore implies that even without any external disturbances or MPM, the DOB will improve the set-point tracking performance of a diagonal PI(D) controller when applied to an inherently multi-variable plant.

3.6 FRACTIONAL ORDER DISTURBANCE OBSERVER

The concept of a FO-DOB was first proposed by Chen et al. (2004) in a motion control context and experimentally validated by Li and Hori (2007).

The two tuning parameters in the DOB Q-filter are the filter bandwidth and the filter order. Not limiting the filter order to the set of integers would therefore be a valuable aid in tuning such that the optimal filter order may be implemented. The concept of the fractional order disturbance observer is to represent the diagonal elements of the filter transfer function as

$$Q_{ii} = \left(\frac{1}{\lambda_{ii}s+1}\right)^{\alpha} \tag{3.31}$$

where $\alpha \in \mathbb{R}$. Details of how the fractional order low-pass filter is realised are given at the end of

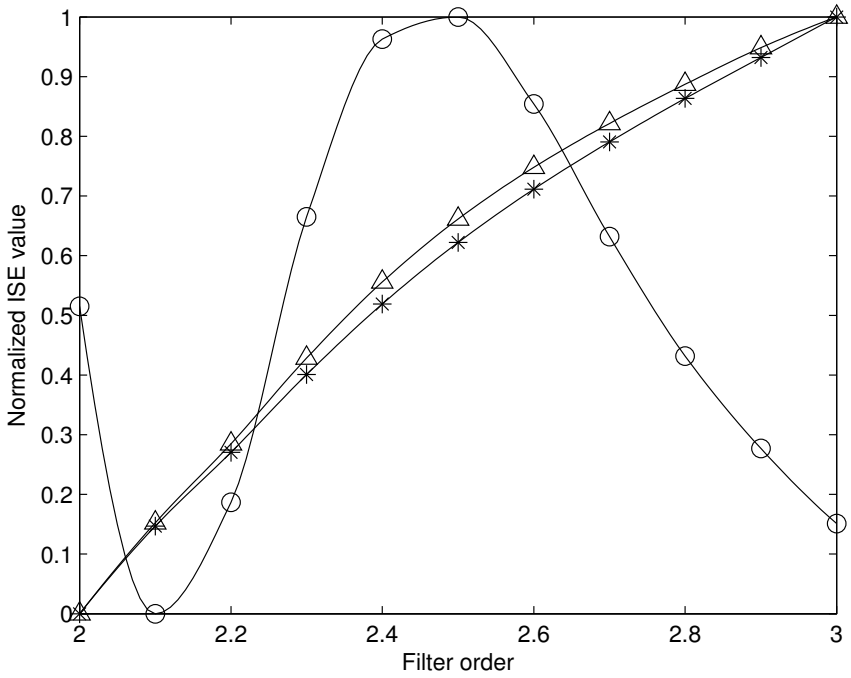

Figure 3.4: Normalised ISE values for different filter orders for α_r disturbance ($*$-markers), ϕ_f disturbance ($\triangle$-markers) and set-point change ($\bigcirc$-markers).

this section. Because the filter order is now not restricted to being an integer, an optimal value for the filter order may be specified based on the desired performance.

To determine the optimal filter order, disturbances were introduced into the system and set-point changes were made in the presence of MPM. The simulation conditions are as described in Section 3.8. The disturbances introduced here are changes in the fraction of rocks in the feed ore (α_r) and the amount of energy needed per ton of fines produced (ϕ_f). The former is an indication of the composition of the feed ore while the latter is an indication of the hardness of the ore. The integral squared error (ISE) values (normalised to be between 0 and 1) are shown in Fig. 3.4. The ISE will depend on the type of perturbation used, e.g. a step or a ramp. A step perturbation in the parameter values could occur in practice when the ore feed is switched to come from a different stockpile.

The feed variations in the ROM ore can introduce other forms of disturbances as well, such as those that arise from pile formation and reclaiming procedures that result in material segregation. Such drifting disturbances are however easier to deal with by the DOB than the step disturbances used in this study.

From Fig. 3.4 it is clear that smaller filter order values give better disturbance rejection performance. This is the same result that was found for linear models by Olivier et al. (2011a). It is interesting to note that the normalised ISE values for set-point tracking are not monotonic. For filter orders slightly larger than 2 a minimum is achieved before the ISE values increase to a maximum at an order equal to 2.5 before decreasing again as the filter order increases. This behaviour is not the same as was found for linear models by Olivier et al. (2011a). The difference between the result obtained here and that obtained by Olivier et al. (2011a) is that the non-linear plant model is used here. This non-monotonicity is also only seen for the set-point change where the plant changes operating point. This seems to indicate that there is no general rule for predicting the ISE values obtained from the nonlinear simulation.

In order to determine the filter order that achieves the best combined performance, the objective function

$$J = \gamma \sum_{1}^{M} \frac{1}{M} ISE_{sp,M} + (1-\gamma) \sum_{1}^{N} \frac{1}{N} ISE_{d,N} \tag{3.32}$$

should be minimised with respect to the disturbance observer filter order. In (3.32), γ is a weight in [0,1] that determines whether the emphasis is on disturbance rejection or set-point tracking, $ISE_{sp,M}$ and $ISE_{d,N}$ are the normalised ISE values for set-point tracking and disturbance rejection respectively, M is the number of different set-point changes to consider and N is the number of different disturbances under study. Practically the value of M should indicate the number of all common set-point changes. This value may be irrelevant if only disturbance rejection is of concern. N should indicate the number of all common and foreseeable disturbances that significantly affect the plant.

Setting $\gamma = 0.5$ (equal weight on set-point tracking and disturbance rejection) the objective function gives the optimal filter order as 2.1. Placing more importance on disturbance rejection than set-point tracking will reduce the optimal filter order. Because the optimal filter order is already quite close to that of the integer order DOB, $\gamma = 0.5$ is sufficient here for comparison purposes. The FO-DOB is therefore implemented with a Q-filter of order 2.1. The controller used is again the same PI-controller discussed in Section 3.3.

Calculating J according to (3.32) with the values as shown in Fig. 3.4 gives $J = 0.0583$ for the FO-DOB as opposed to $J = 0.2575$ for the integer order DOB. This suggests that there would be an improvement in overall performance with the FO-DOB implementation when considering set-point tracking and disturbance rejection.

Fractional order low-pass filter implementation

The implementation of a fractional order low-pass filter (FO-LPF) and its use in control are presented in Monje et al. (2010). The FO-LPF is of the form

$$H_{FOLPF}(s) = \left(\frac{1}{\lambda s+1}\right)^{\alpha} \tag{3.33}$$

where α is a real number and $\alpha \in (0,1)$. Code available from MATLAB Central[1] may be used to obtain an implementable finite dimensional transfer function

$$H_{FOLPF}(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1+\sum_{k=1}^{p} a_k z^{-k}} \tag{3.34}$$

which is based on the so-called "Impulse response invariant discretization (IRID)."

In order to be able to implement a fractional order filter for which the order is larger than 1, the FO-LPF in the form of (3.33) may be cascaded with an integer order filter of order n as

$$H(s) = \left(\frac{1}{\lambda_{FO}s+1}\right)^{\alpha}\left(\frac{1}{\lambda_{IO}s+1}\right)^{n} = \left(\frac{1}{\lambda s+1}\right)^{(\alpha+n)} \tag{3.35}$$

if $\lambda_{FO} = \lambda_{IO}$. The order of the overall filter is now $\alpha+n$ where $\alpha \in (0,1)$ and $n \in \mathbb{N}$. When two filters are cascaded to acquire the correct filter order, the bandwidth of the filter will invariably change. In order to acquire a filter of bandwidth ω_B with an order of $\alpha+n$, the filter time constant should be specified to be

$$\lambda = \frac{\sqrt{2^{1/(\alpha+n)}-1}}{\omega_B}. \tag{3.36}$$

3.7 BODE IDEAL CUT-OFF DISTURBANCE OBSERVER

Bode introduced the ideal cut-off filter in Bode (1945) such that the filter would produce an optimal trade-off between the desired phase margin and the most rapid cut-off. The transfer function of the filter is given by

$$\beta(s) = \frac{K}{\left(\sqrt{1+\frac{s^2}{\omega_o^2}}+\frac{s}{\omega_o}\right)^{2(1-\eta)}} \tag{3.37}$$

where ω_o is the cut-off frequency and η is a parameter relating to the filter order such that the roll-off at frequencies much larger than ω_o be $-40(1-\eta)$ dB/dec. The Bode plot of this transfer function (with $\omega_o = 1$ and $\eta = 0.5$) is shown in Fig. 3.5. The response of a first-order filter as given by (3.28) is also indicated in the figure.

[1] See `http://www.mathworks.com/matlabcentral/fileexchange/21365`

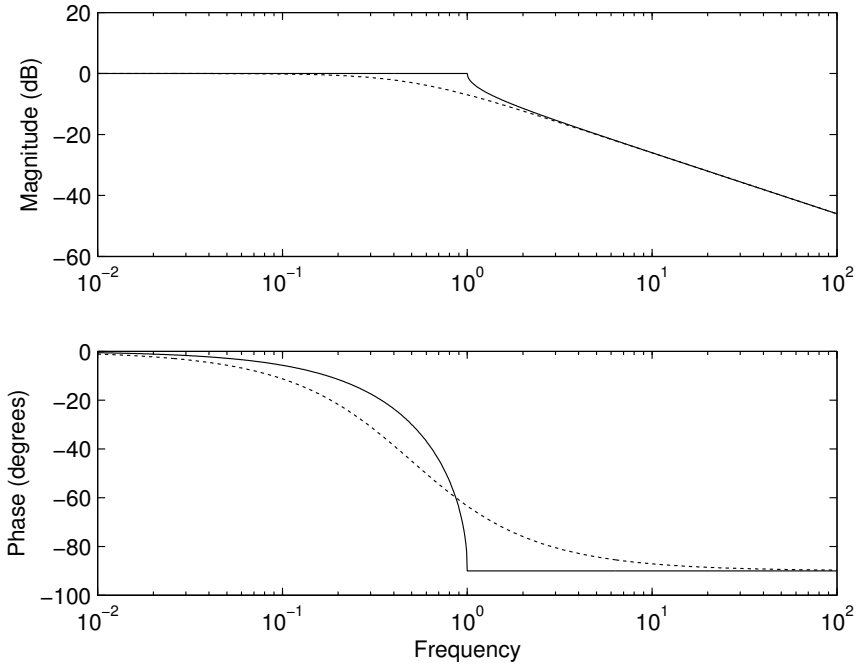

Figure 3.5: Bode plot of a BICO filter (solid line) and a regular first-order filter (dotted line).

It is notable from Fig. 3.5 that the BICO filter has a flat amplitude response in the pass-band of the filter and then a rapid cut-off. The phase of the BICO filter is approximately constant in the stop-band, which means that the phase margin would be relatively insensitive to uncertainties.

Details of how the BICO filter may be implemented are given at the end of this section.

For the BICO filter the two tuning parameters are ω_o and η. These parameter values are also determined through simulating different combinations and calculating the parameter values that produce the optimal overall performance. This procedure is similar to that discussed in Section 3.6 for the FO-DOB. Fig. 3.6 shows the ISE values obtained for different filter order and bandwidth values. The optimal value is achieved for $\omega_o = 2.75$ and $n = 1$. The relative order $n = 1$ is equivalent to $\eta = 0.5$ in equation (3.37).

BICO low-pass filter implementation

The relative degree of the IRID representation of the BICO filter is 0. Owing to this fact, the BICO filter should be cascaded with a low-pass filter of order 2 such that the transfer function $Q(s)G_-^{-1}(s)$ is realisable (based on the discussion of Section 3.5). This is similar to the method used in Section

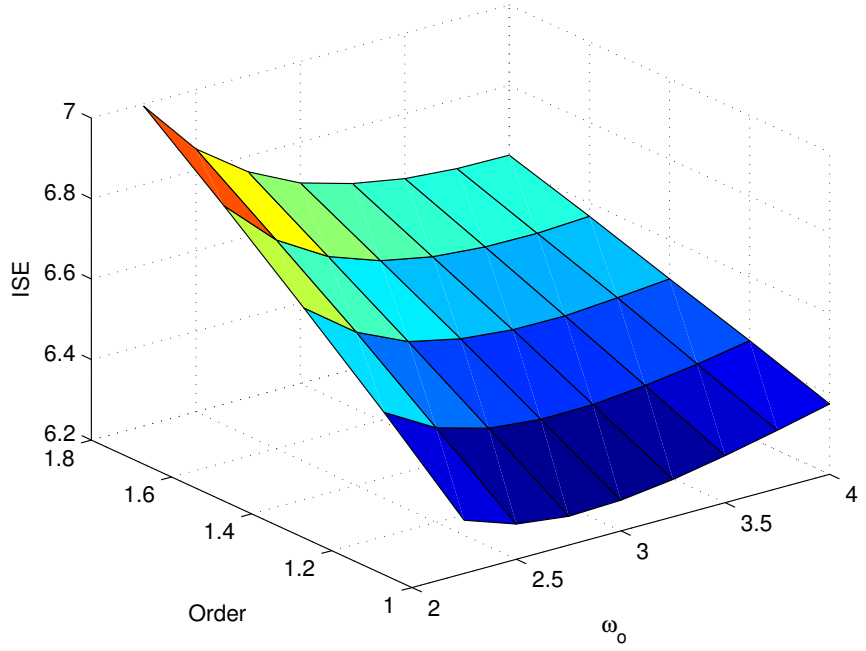

Figure 3.6: ISE values obtained with different filter bandwidth and filter order values for the BICO-DOB.

3.6 but here the order of the BICO filter is not restricted to be in (0,1) as the FO-LPF was.

Code for an impulse response invariant discretisation implementation for the BICO filter (similar to that shown in (3.34)) can be found from MATLAB Central[2].

3.8 DISTURBANCE OBSERVER SIMULATION RESULTS

The PI controller discussed in Section 3.3 as well as the three DOBs (as discussed in Sections 3.5, 3.6 and 3.7) are implemented on the nonlinear model of the ROM ore milling circuit. The external disturbance rejection capabilities of each, as well as the set-point tracking capabilities, are tested and the results are presented here.

3.8.1 External disturbance rejection results

External disturbance rejection is very important in ROM ore milling as a consistent product should be passed to the downstream process while parameters in the milling circuit change continuously. Firstly a change is made in the fraction of rocks in the ore entering the mill (α_r). This parameter

[2]See `http://www.mathworks.com/matlabcentral/fileexchange/28398/`

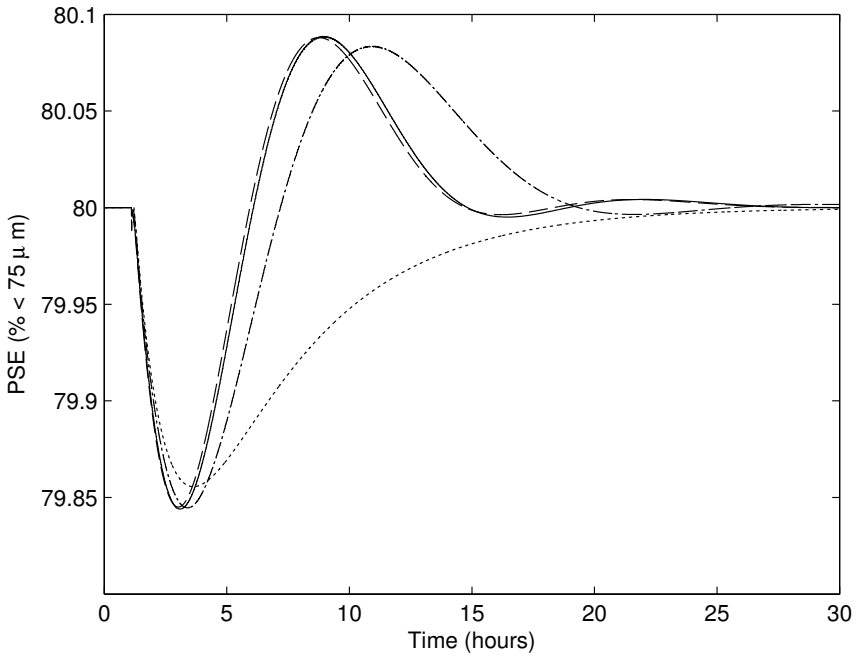

Figure 3.7: PSE in the presence of a strong external disturbance (change in α_r) with PI (dotted line), DOB (solid line), FO-DOB (dashed line) and BICO-DOB (dash-dot line).

changes frequently as the distribution of rocks on the stockpile rarely remains constant. The value of α_r is decreased by 20% at 1 hour and the effect this has on the PSE is shown in Fig. 3.7. The deviations in the PSE values may seem insignificant on a real plant but here the disturbances are kept relatively small such that the manipulated variables are not driven to their limits. This gives a much clearer comparison of the actual functioning of the controllers as opposed to comparing their constraint handling capabilities.

The ISE as well as the percentage deviation from the nominal value for all four control schemes are shown in Table 3.2.

Next a change is made in the amount of energy needed for a ton of fines produced (ϕ_f). This parameter also changes frequently as the hardness of the ore entering the mill changes. The value of ϕ_f is decreased by 15% at 1 hour and the effect this has on the PSE is shown in Fig. 3.8. The ISE as well as the percentage deviation for all four control schemes are shown in Table 3.2.

It is clear from Table 3.2 that the DOB based control schemes fare best at disturbance rejection.

Table 3.2: Result summary for disturbance rejection and set-point tracking.

		α_r-disturbance	ϕ_f-disturbance	SP-tracking
PI	ISE	6.314	10.864	22.439
	%OS	0.181	0.341	0.0
DOB	ISE	5.043	10.665	17.377
	%OS	0.194	0.351	0.201
FO-DOB	ISE	5.235	10.863	17.338
	%OS	0.195	0.344	0.198
BICO-DOB	ISE	6.271	11.773	17.059
	%OS	0.194	0.336	0.177

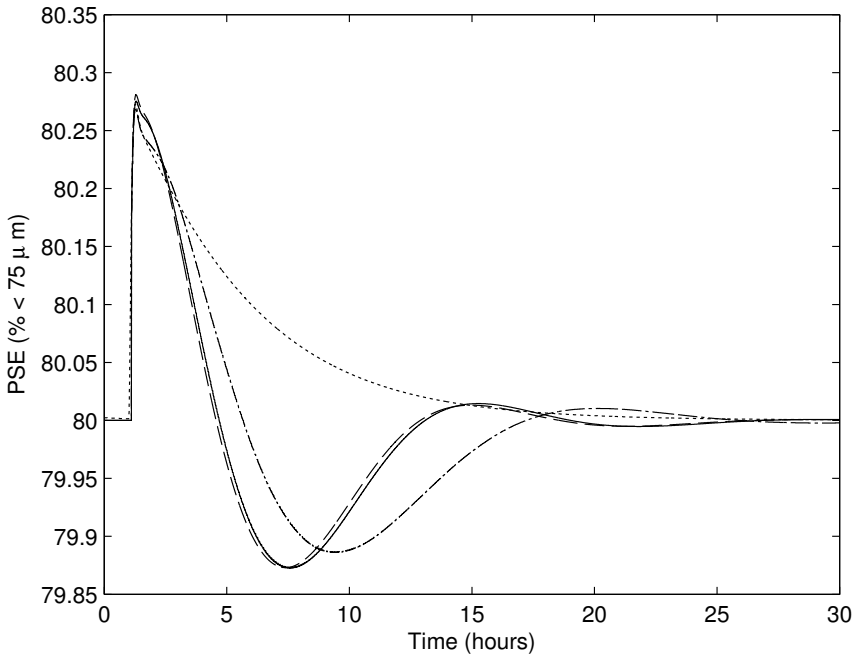

Figure 3.8: PSE in the presence of a strong external disturbance (change in ϕ_f) with PI (dotted line), DOB (solid line), FO-DOB (dashed line) and BICO-DOB (dash-dot line).

3.8.2 Set-point tracking in the presence of model-plant mismatch

The set-point tracking capability of each controller in the presence of MPM is presented here. Model-plant mismatch is a common occurrence on a ROM ore milling circuit (Olivier and Craig, 2011) and hinders the performance of most controllers (Badwe et al., 2009).

Some of the uncertainties present in actual milling models have been quantified by Craig and MacLeod (1995) and are known to be severe. For linear time-invariant models the introduction of MPM is as simple as changing the gain, time constant or time delay of the linear model as discussed by Badwe et al. (2009). For the nonlinear model however MPM can only be introduced through changes in the model parameters. These parameter value changes are made before the onset of the simulation.

The parameters whose values are changed are α_f, which is the fraction of fines in the feed ore, and α_r, which is the fraction of rocks in feed the ore. The values of both α_f and α_r are increased by 10%. Note that α_r was used as an external disturbance in Section 3.8.1 and here it is used to introduce MPM. By changing the nominal parameter values before the onset of the simulation, a linearised plant model different from the one used for controller design will result.

A step change is then made in the set-point for the PSE from 80% to 79%. The results are shown in Fig. 3.9. The summary of the performance of each controller is shown in Table 3.2.

It is clear from Table 3.2 that the addition of a disturbance observer to the normal PI controller gives better results than the PI controller alone because of the decrease in the ISE values. The rise times of the responses are also much smaller with the disturbance observer present.

The addition of the integer order DOB results in a substantial decrease in the ISE values shown in Table 3.2, as also reported by Chen et al. (2009). The order of the FO-DOB is close to the order of the integer order DOB and consequently the results are rather similar. The overall performance is however improved through the smaller value reported for J in (3.32). The BICO-DOB has poorer disturbance rejection performance than the other two DOB varieties. It does however give the best set-point tracking performance. The ease of implementation of a DOB and the substantial performance improvement it yields makes it an attractive addition to the control loop.

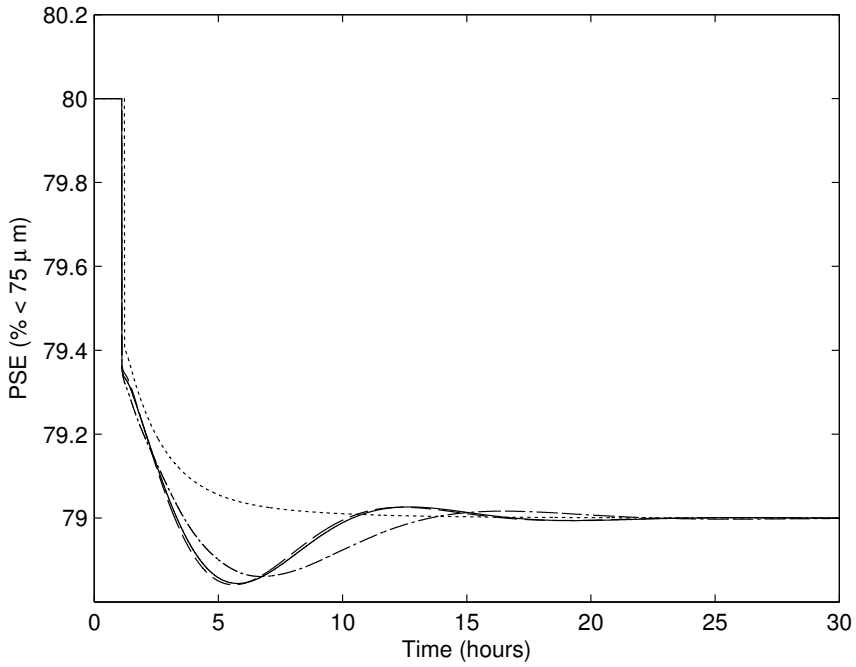

Figure 3.9: PSE in the presence of MPM with PI (dotted line), DOB (solid line), FO-DOB (dashed line) and BICO-DOB (dash-dot line).

3.9 CONCLUSION

Using a DOB, it is not only possible to attenuate the external disturbances in a grinding mill circuit, but also to reduce the effect of MPM as well as uncertainties from internal couplings. This chapter contributes a novel fractional order DOB for a ROM ore milling circuit where a fractional order low pass filter (Q-filer) is used in the DOB to offer additional tuning freedom to obtain optimal set-point tracking and disturbance rejection performance. The proposed FO-DOB scheme has been tested on a 3×3 nonlinear MIMO plant model, for disturbances that are common and significant as well as under model-plant mismatch which is also usually present, to evaluate the performance gained over the commonly used PI controller and an integer order DOB. The simulation results show that the FO-DOB is a useful additional method to consider when faced with external disturbances, MPM and unaccounted coupling effects.

A novel disturbance observer implemented with a BICO low-pass filter is also introduced. The simulation results show that the BICO-DOB gives relatively good performance under the circumstances tested here. It gives the best set-point tracking of all the implementations tested, as well as maintaining good disturbance-rejection capabilities.

CHAPTER 4 MODEL-PLANT MISMATCH DETECTION

4.1 INTRODUCTION

The performance of a model predictive controller depends on the quality of the plant model that is available. Often parameters in a ROM ore milling circuit are uncertain and inaccurate parameter estimation leads to a mismatch between the model and the actual plant. Although MPM is inevitable, timely detection of significant mismatch is desirable in order to prevent deteriorated control performance.

Grinding mill circuits are still predominantly controlled using single-loop PI(D) controllers (Wei and Craig, 2009) despite the significant advances that a technology such as MPC has made in the process industries (Qin and Bagwell, 2003). Perhaps one of the reasons for this is the fact that the dynamics of milling circuits can change significantly over time, leading to deteriorating controller performance. A technology that may aid the introduction of advanced control in grinding mill circuits is MPM detection (see e.g. Badwe et al. (2009)), as described in this chapter.

Once significant mismatch has been detected, the process control engineer can decide to re-identify the particular part of the plant model that contains significant mismatch. The controller can then be redesigned. These steps could be performed on-line.

This chapter starts with the derivation of a closed-form MPM expression in the case where the controller may be represented with a transfer function. This expression does however not hold for MPC, as the controller cannot be represented with a transfer function.

The rest of the chapter describes a simulation study of the application of an MPM detection strategy described by Badwe et al. (2009), to a ROM ore milling circuit under MPC control. The milling circuit model used is a linear time-invariant (LTI) approximation of the fundamental milling circuit model described by Coetzee (2009); Coetzee et al. (2010). Model-plant mismatch, motivated from

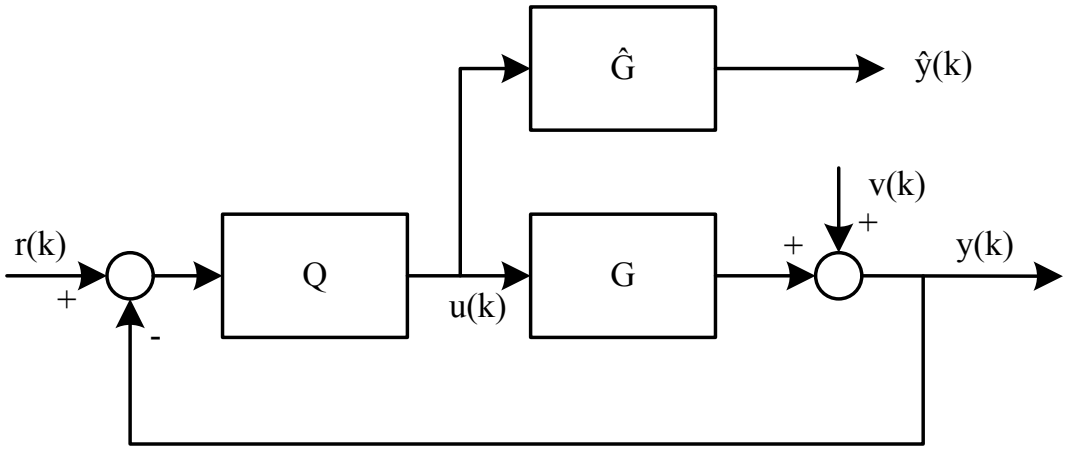

Figure 4.1: Block diagram of a control loop with model outputs being generated.

industrial experiments (Craig and MacLeod, 1995), is introduced in the model and its location in the multi-variable matrix model is correctly detected.

4.2 CLOSED-FORM MPM EXPRESSION

Consider the control loop shown in Fig. 4.1 in which G is the plant that generates the true output y, $\hat{G}$ is the model of the plant that generates the model output $\hat{y}$, Q is the controller, v is any disturbance that may be present and r is the reference signal (set-point).

The derivation is done for a MIMO system in which all signals may be vectors, and reference to time will be dropped for ease of representation. Let the residual (e) be the difference between the actual output and the model output as

$$e = y - \hat{y} \tag{4.1}$$

$$= Gu + v - \hat{G}u \tag{4.2}$$

$$= \Delta u + v \tag{4.3}$$

where $\Delta = G - \hat{G}$ is the mismatch. The control signal (u) is given by

$$u = Q(r-y) \tag{4.4}$$
$$u = Q(r-[Gu+v]) \tag{4.5}$$
$$u = Qr - QGu - Qv \tag{4.6}$$
$$(I+QG)u = Qr - Qv \tag{4.7}$$
$$u = (I+QG)^{-1} Q(r-v) \tag{4.8}$$
$$u = Q(I+GQ)^{-1}(r-v), \tag{4.9}$$

where the push-through rule for matrix manipulation (Skogestad and Postlethwaite, 2005, p.68) was used to go from (4.8) to (4.9). Substitution of (4.9) into (4.3) then gives

$$e = \Delta Q(I+GQ)^{-1}(r-v) + v \tag{4.10}$$
$$e = \Delta Q\left(I+\{\Delta+\hat{G}\}Q\right)^{-1}(r-v) + v \tag{4.11}$$
$$e = \Delta Q\left(I+\Delta Q+\hat{G}Q\right)^{-1}(r-v) + v \tag{4.12}$$
$$(e-v)(r-v)^{-1} = \Delta Q\left(I+\Delta Q+\hat{G}Q\right)^{-1} \tag{4.13}$$
$$(e-v)(r-v)^{-1}\left(I+\Delta Q+\hat{G}Q\right) = \Delta Q \tag{4.14}$$
$$(e-v)(r-v)^{-1}\left(I+\hat{G}Q\right) = \Delta Q - (e-v)(r-v)^{-1}\Delta Q \tag{4.15}$$
$$(e-v)(r-v)^{-1}\left(I+\hat{G}Q\right) = \left[I-(e-v)(r-v)^{-1}\right]\Delta Q. \tag{4.16}$$

The expression $G = \Delta + \hat{G}$ is used to go from (4.10) to (4.11). From (4.11) all the terms in the equation are known, save for the disturbance if it is unmeasured. This then leads, after further simplification, to a closed-form expression for the mismatch (Δ) as:

$$\Delta = \left[I-(e-v)(r-v)^{-1}\right]^{-1}(e-v)(r-v)^{-1}\left(I+\hat{G}Q\right)Q^{-1}. \tag{4.17}$$

This expression may be used to derive the mismatch if the disturbances are known. If the disturbances are however unmeasured, data from a period of operation free from significant disturbances can be used, and with $v = 0$, (4.17) becomes

$$\Delta = \left[I-er^{-1}\right]^{-1} er^{-1}\left(I+\hat{G}Q\right)Q^{-1}. \tag{4.18}$$

For signals (such as r) the Moore-Penrose pseudoinverse may be used to calculate the inverse. Sufficient excitation is required in either the disturbance or the reference signal in order for the application

of (4.17) to be sensible. If any of the signals to be inverted are singular, the expression will not produce the correct results although the Moore-Penrose pseudoinverse can still be calculated. In such a situation a small amount of noise can be added to the signal, and although it may still be ill-conditioned, the mismatch expression will be applicable. An example of such a situation is given at the end of this section.

The expression $G = \Delta + \hat{G}$ may now again be used to obtain the transfer function of the actual plant as

$$G = \left[I - (e-v)(r-v)^{-1}\right]^{-1}(e-v)(r-v)^{-1}\left(I + \hat{G}Q\right)Q^{-1} + \hat{G}. \tag{4.19}$$

Example of noise addition for correct matrix inversion

Say for some two-input two-output system the mismatch expression (4.17) should be applied but the reference signal is

$$r = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.20}$$

This signal is singular and the Moore-Penrose pseudoinverse, although it can be calculated, does not give sensible results. For this signal

$$r \times pinv(r) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \tag{4.21}$$

where $pinv(r)$ refers to the Moore-Penrose pseudoinverse. This result is clearly incorrect as it is known that any matrix multiplied by its inverse should produce the identity matrix. Adding zero-mean noise with a standard deviation of 0.001 to the original reference signal gives

$$r^* = \begin{bmatrix} -0.0003 & 0.9989 & 1.0017 & 0.9987 \\ -0.0008 & 0.0025 & 0.0003 & -0.0009 \end{bmatrix} \tag{4.22}$$

which is clearly very close to the original reference signal. For this signal

$$r^* \times pinv(r^*) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.23}$$

which is equal to the identity matrix.

4.3 PARTIAL CORRELATION ANALYSIS FOR MPM DETECTION

For controllers where no explicit transfer function is available, such as MPC (Qin and Bagwell, 2003), the control signal cannot be written as $u = Q(r-y)$ as was done in (4.4). The closed-form expression

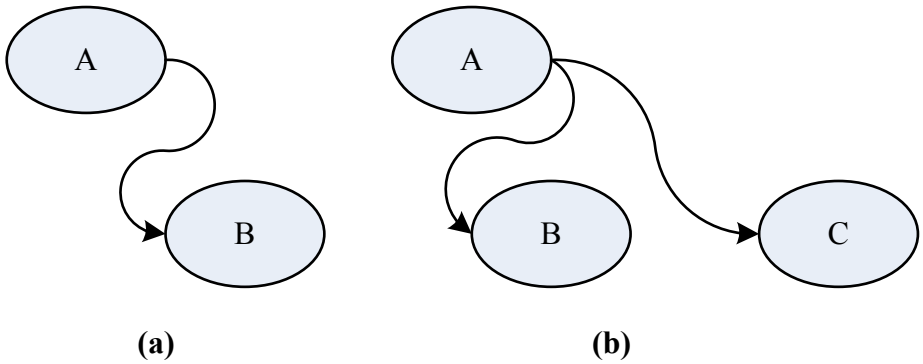

Figure 4.2: Indication of how spurious correlation may introduced.

for the mismatch will therefore not apply and an alternative method is needed to detect significant mismatch.

Recently Badwe et al. (2009) proposed a method for MPM detection in MPC applications based on partial correlation analysis between the model residuals and the manipulated variables. Partial correlation analysis helps in detecting hidden correlations as well as inhibiting the detection of spurious correlations. Fig. 4.2 shows an example of how spurious correlation may be generated. In (a) the event A causes the event B and consequently correlation exists between A and B implying this causation. In (b) the event A simultaneously causes events B and C. In this case correlation still exists between events B and C although no causal link exists.

With specific reference to the detection of mismatch in a multi-variable system, the situation shown in Fig. 4.3 is probable. When a set-point change is made, the MPC controller will probably react by changing all the manipulated variables by some amount, depending on the interactions between the loops. If mismatch exists in any of the transfer function matrix elements, a residual will be generated because of the mismatch. There will now still be correlation between this residual and the other manipulated variables, even though those manipulated variables did not cause the mismatch. Partial correlation is proposed to first break this apparent link before inferring causation from correlation.

Consider the closed-loop IMC structure represented in Fig. 4.4 (from Seborg et al. (2003)). Here G is an $n \times m$ MIMO plant, $\hat{G}$ is the model representing G and Q is a multi-variable controller. The plant and model outputs are $y(k)$ and $\hat{y}(k)$ respectively, $r(k)$ is the vector of references, $u(k)$ the manipulated variables and $v(k)$ the vector of disturbances.

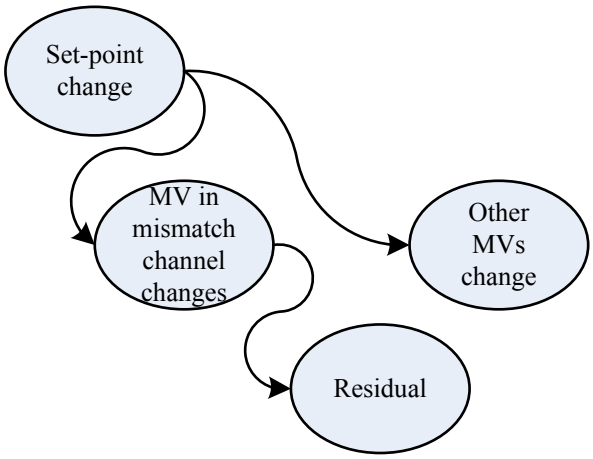

Figure 4.3: Spurious correlation in a multi-variable plant containing mismatch.

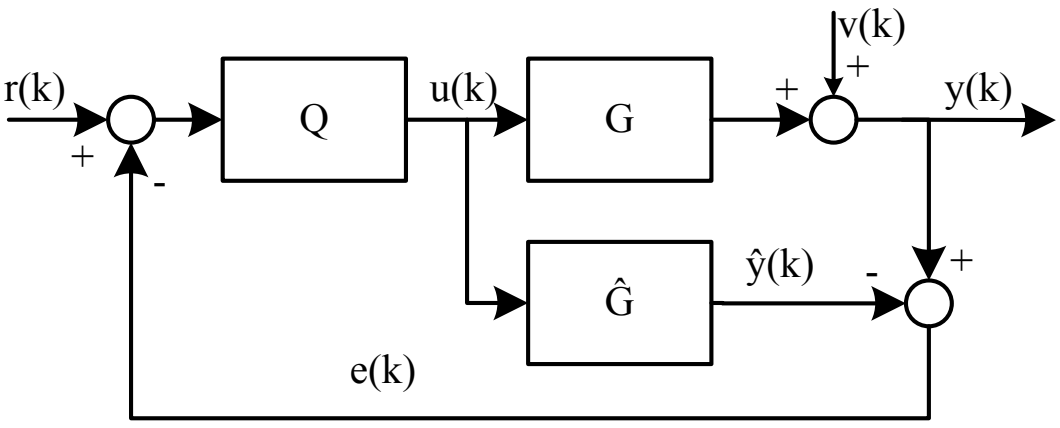

Figure 4.4: Closed-loop IMC structure.

The residuals ($e(k)$) are given by:

$$e(k) = y(k) - \hat{y}(k) = \Delta u(k) + v(k) \tag{4.24}$$

where ($\Delta = G - \hat{G}$) is the mismatch between the plant and the model. A correlation analysis between the signals $e(k)$ and $u(k)$ is expected to indicate the amount of mismatch Δ. Hidden and spurious correlations may however be present and a normal correlation analysis will not suffice. The following relations are obtained from Fig. 4.4:

$$e(k) = [I + \Delta Q]^{-1} \Delta Q r(k) + [I + \Delta Q]^{-1} v(k) \tag{4.25}$$

$$u(k) = \underbrace{Q[I + \Delta Q]^{-1}}_{S_{ru}} r(k) - \underbrace{Q[I + \Delta Q]^{-1}}_{S_{vu}} v(k) \tag{4.26}$$

where S_{ru} and S_{vu} are the input sensitivities from r and v respectively.

At each sampling instant the values of the manipulated variables are calculated based on the difference between the output and the reference vectors. Depending on the interactions in the model and the design of the controller, correlation may exist between manipulated variables. This may lead to the detection of spurious correlation or to the non-detection of hidden correlation between residuals and manipulated variables. This would in turn obscure the correct identification of the location of significant MPM. To overcome this Badwe et al. (2009) proposed the use of partial correlation analysis.

Data for analysis should be chosen from a period of time where there is sufficient set-point excitation. Since models are fitted to the sensitivity functions S_{ru} and S_{vu}, the set-points should be sufficiently exciting to ensure estimation accuracy. In order to ensure that MPM is not incorrectly identified owing to the presence of disturbances, the disturbance free components of the manipulated variables (MVs) are required. These are the components of the MVs needed to react to set-point changes and not for disturbance rejection. The disturbance-free components of the MVs are represented as $\hat{u}^r(k)$ and may be obtained as described by Badwe et al. (2009). The method used is based on fitting a model between set-points and the manipulated variables such that when no set-point changes have been made, any changes in the manipulated variables are considered to be due to disturbances.

Next, the component of each MV that is uncorrelated with all other MVs is computed. Each MV may be represented as

$$\hat{u}_i^r(k) = G_{u_i} \tilde{u}^r(k) + \varepsilon_{u_i}(k) \tag{4.27}$$

where G_{ui} is a model identified between u_i^r and all the other MVs, $\tilde{u}^r$ contains all the other MVs except for u_i, and ε_{u_i} is that component of u_i that is uncorrelated with all other MVs. The estimate of ε_{u_i} is then given by:

$$\hat{\varepsilon}_{u_i}(k) = \hat{u}_i^r(k) - G_{u_i}\tilde{u}^r(k). \tag{4.28}$$

A similar procedure is applied to calculate the component of each residual that is uncorrelated with all other MVs except u_i. Each residual is represented as

$$e_j(k) = G_{e_j}\tilde{u}^r(k) + \varepsilon_{e_j}(k). \tag{4.29}$$

Here G_{e_j} is the model identified between residual e_j and all other MVs except u_i. The estimate for ε_{e_j} is then given by

$$\hat{\varepsilon}_{e_j}(k) = e_j(k) - G_{e_j}\tilde{u}^r(k). \tag{4.30}$$

Non-zero correlation between $\hat{\varepsilon}_{e_j}$ and $\hat{\varepsilon}_{u_i}$ indicates the presence of model-plant mismatch in the $u_i - y_j$ channel. This MPM identification technique is applied to the ROM ore milling circuit to detect mismatch between parameters in the model and the actual plant.

4.4 MPC CONTROLLER DESIGN

In Section 3.2 a linearised plant model is presented that is used for the design of a PI controller. Here a linearised plant model is once again required for the implementation of a linear model predictive controller. Later in this chapter MPM, motivated by industrial experiments (Craig and MacLeod, 1995), will be applied and detected. In Craig and MacLeod (1995) the uncertainty in the model elements are given for the gain, time constant and time delay, for a first order plus time delay (FOPTD) model

$$G(s) = \frac{K}{\tau s + 1}e^{-\theta s}. \tag{4.31}$$

In order to use these uncertainties a linearised model is first derived in which all the transfer function elements are presented as FOPTD models. Some of the sub-models in the plant are however integrators and the slightly altered form

$$G(s) = \frac{K}{s + \alpha}e^{-\theta s} \tag{4.32}$$

will be used, which can be used to represent the FOPTD model as well as an integrator by simply letting $\alpha \to 0$.

Linearised plant model for MPC controller design

The linear model of the plant is obtained through applying a standard SID procedure as described by Ljung (1999), to the milling circuit model of Chapter 2. Step tests were performed around the operating point shown in Table 2.2. Operating data for 60 hours were collected and models were fitted for all nine elements of the transfer function matrix. The final model for control is then given by:

$$\begin{bmatrix} \Delta PSE \\ \Delta LOAD \\ \Delta SLEV \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \begin{bmatrix} \Delta CFF \\ \Delta MFS \\ \Delta SFW \end{bmatrix} \tag{4.33}$$

where g_{1j} is in the form

$$g_{1j} = \frac{k_{1j}}{s+\alpha_{1j}} e^{-\theta_{1j}s} \tag{4.34}$$

with $k_{11} = -2.4 \times 10^{-4}$, $k_{12} = -5.99 \times 10^{-4}$, $k_{13} = 1.45 \times 10^{-3}$; $\alpha_{11} = 0.5882$, $\alpha_{12} = 1.353$, $\alpha_{13} = 2.216$; and $\theta_{11} = 0.0111$, $\theta_{12} = 0.0639$, $\theta_{13} = 0.0111$.

The other six transfer functions are in the form

$$g_{ij} = \frac{k_{ij}}{s+10^{-6}} \tag{4.35}$$

with $k_{21} = 7.15 \times 10^{-4}$, $k_{22} = 7.22 \times 10^{-3}$, $k_{23} = -1.39 \times 10^{-3}$, $k_{31} = -0.60$, $k_{32} = 0.0097$, and $k_{33} = 0.774$. The models were derived in time units of hours with a sampling time of 10 seconds.

The model does not refer to the manipulated variables MIW and MFB. Once again the value of MFB is kept constant at its nominal value (as reported in Table 2.2). The value of MIW is derived from the value of MFS through a constant water-to-solids ratio into the mill as discussed by Coetzee (2009).

MPC controller specification

The controller for the milling circuit is a linear model predictive controller based on the linearised plant model discussed in the previous section. At each sampling instant the objective of the controller is to minimise some scalar performance index

$$\min_{u} V(u, x_0) \tag{4.36}$$

$$s.t.\ x \in X, u \in U \tag{4.37}$$

Table 4.1: MPC constraints and weights.

Variable	Min	Max	Weight	Description
MFS	0	200	0.01	Flow-rate of ore to the milling circuit [t/h]
CFF	400	500	0.01	Flow-rate of slurry to the cyclone [m^3/h]
SFW	0	400	0.01	Flow-rate of water to the sump [m^3/h]
PSE	60	90	100	Product particle-size [$\% < 75\mu$m]
LOAD	30	50	100	Total charge of the mill [%]
SLEV	2	37.5	1	Level of the sump [m^3]

$$\theta_c(x,u) \leq 0 \tag{4.38}$$

where $x : \mathbb{R} \to \mathbb{R}^{n_x}$ is the state trajectory, $u : \mathbb{R} \to \mathbb{R}^{n_u}$ is the control trajectory, x_0 is the initial state and $\theta_c(x,u)$ is the constraint vector. The solution to the optimisation problem provides a set of optimal control moves, the first of which is implemented and the optimisation problem is again solved at the next sampling instant.

At each sampling instant the controller calculates the required values for the manipulated variables CFF, SFW and MFS. The controller uses a prediction horizon of 20 and a control horizon of 5. The weights given to each controlled and manipulated variable are as shown in Table 4.1. The constraints imposed by the controller are the same as the constraints on the variables shown in Table 2.2 but are also included in Table 4.1. The Model Predictive Control toolbox in MATLAB (Bemporad et al., 2004) is then used to define the controller.

The simulation results in Fig. 4.5 show the outputs for nominal operation (without any parameter mismatch in the plant) of the controller over 5 hours. A set-point change is made for LOAD from 45% to 50% at time 1 hour. This is similar to the set-point change later employed to ensure sufficient excitation for the MPM detection algorithm. The set-points are indicated with dashed lines in the figure.

The manipulated variables for nominal operation are shown in Fig. 4.6. The constraints on each MV are indicated by dashed lines.

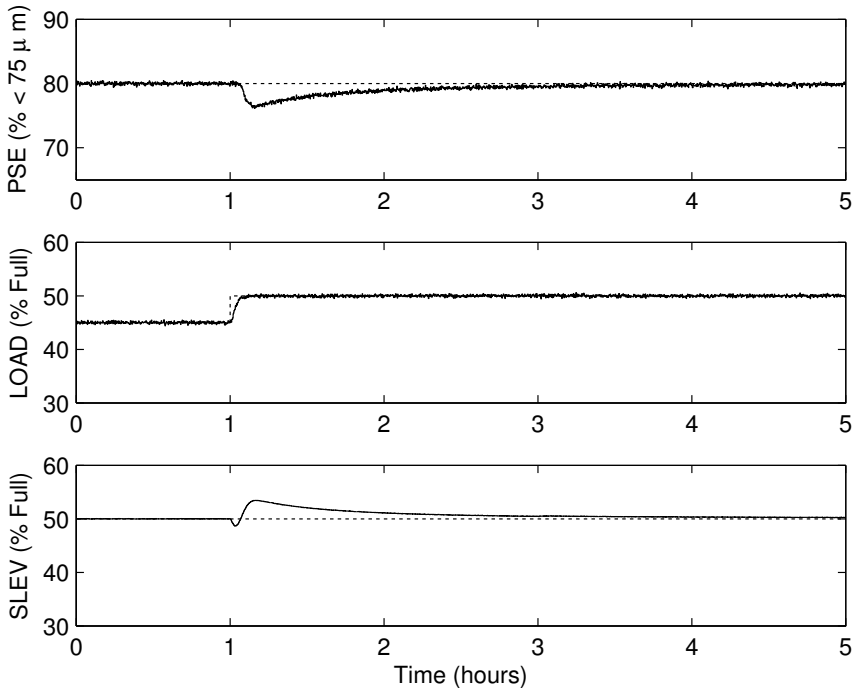

Figure 4.5: Outputs for nominal operation.

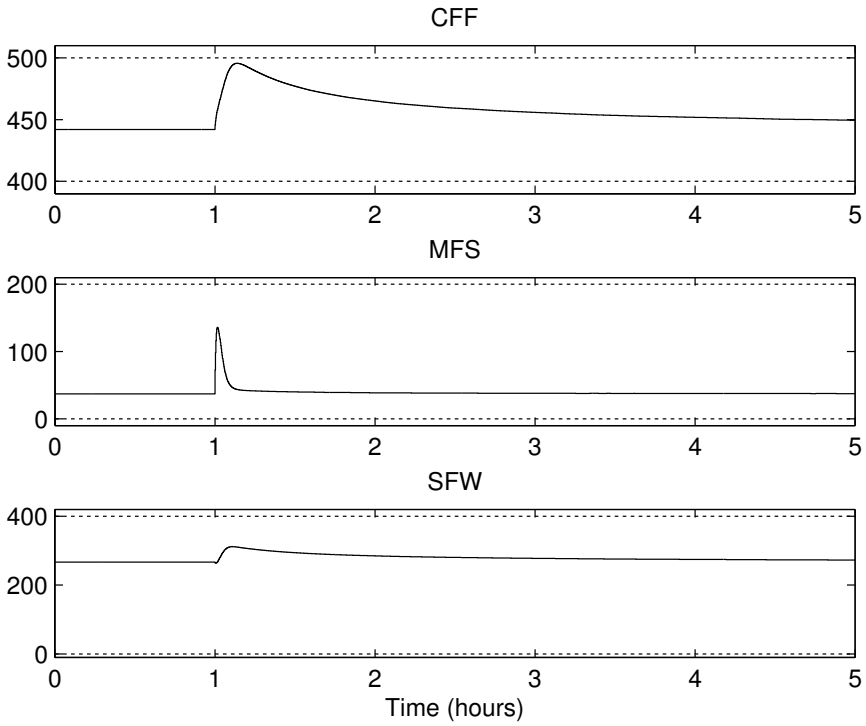

Figure 4.6: Manipulated variables for nominal operation.

4.5 MPM DETECTION RESULTS

When the parameter ϕ_f, which is an estimate of the hardness of the ore being fed into the mill, is changed from its nominal value to some perturbed value in the plant (G) while the nominal value is maintained in the model ($\hat{G}$), there is a discrepancy between the actual hardness of the ore in the mill and the estimate of the hardness. The hardness of the ore entering the mill is a variable that commonly varies during operation of the milling circuit.

A shown in Chapter 2, the hardness of the ore entering the mill affects the hold-up time of ore in the mill. In the linear model, the hold-up time is approximately equal to the time constants in the transfer functions of LOAD/CFF (g_{21}) and LOAD/SFW (g_{23}). It has been shown (Craig and MacLeod, 1995) that the time constant is given by $\tau = RC$ where C is the volume of material inside the mill and R is the inverse of the slurry discharge rate. The relative uncertainties of the time constants in the linearised transfer function has been investigated by Craig and MacLeod (1995). The relative uncertainty matrix for the time constants in the linearised model was found to be:

$$\tau_{ij}: \begin{bmatrix} 18\% & - & 19\% \\ 40\% & - & 60\% \\ - & - & - \end{bmatrix}. \tag{4.39}$$

With reference to this result the time constants for the transfer functions of LOAD/CFF and LOAD/SFW in the plant model are increased by 30%. The MPM detection algorithm is now expected to detect this parameter mismatch in the perturbed transfer functions.

A simulation run is initially performed with the nominal plant over 5 hours. The MPM detection algorithm correctly identifies no mismatch as no partial correlation between the residuals and the manipulated variables. This result is shown in Fig. 4.7. The plant model is then perturbed by increasing the time constants in the transfer function elements g_{21} as well as g_{23} and a simulation run is performed (once again over a period of 5 hours). The MPM detection algorithm is applied and the partial correlation plots are shown in Fig. 4.8.

From Fig. 4.8 it is seen that the mismatches in both g_{21} and g_{23} are correctly detected with clear non-zero correlation.

In order to ensure sufficient excitation in the generation of these results the reference value for LOAD was changed from 45% to 50% at time 1 hour. The models G_{u_i} and G_{e_j} determined by the MPM

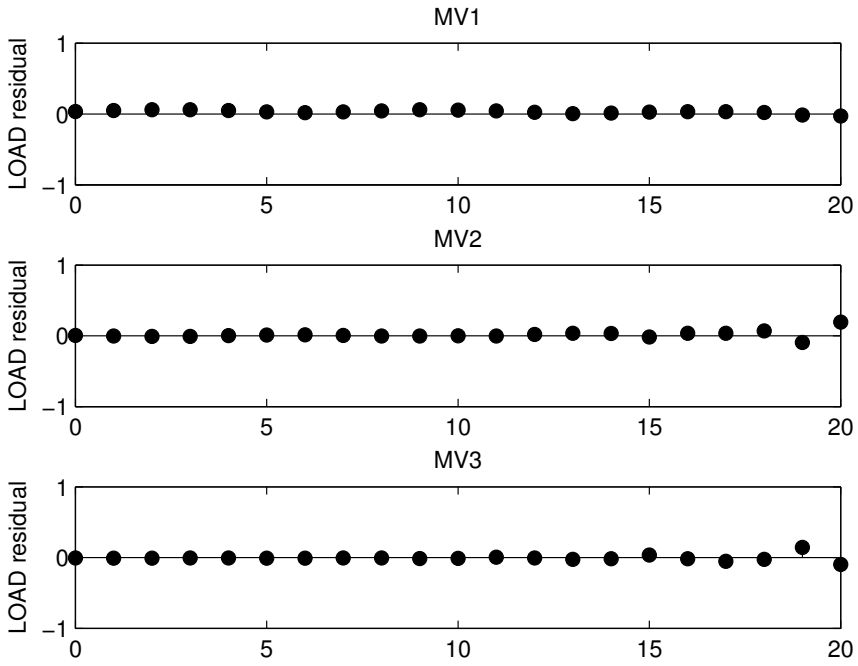

Figure 4.7: Partial correlation plots between the MVs and the LOAD residual with no mismatch present.

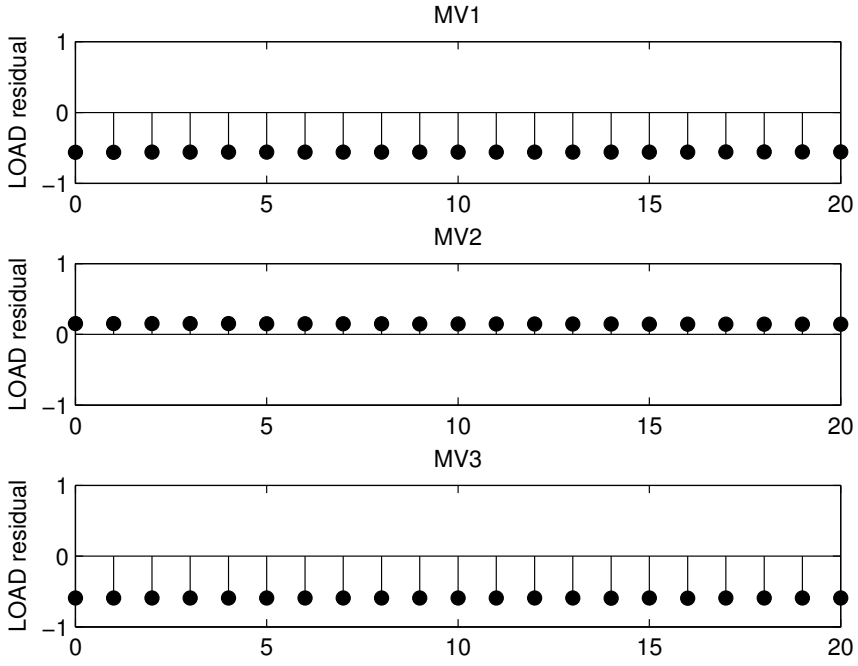

Figure 4.8: Partial correlation plots between the MVs and the LOAD residual for time constant mismatch in g_{21} and g_{23}.

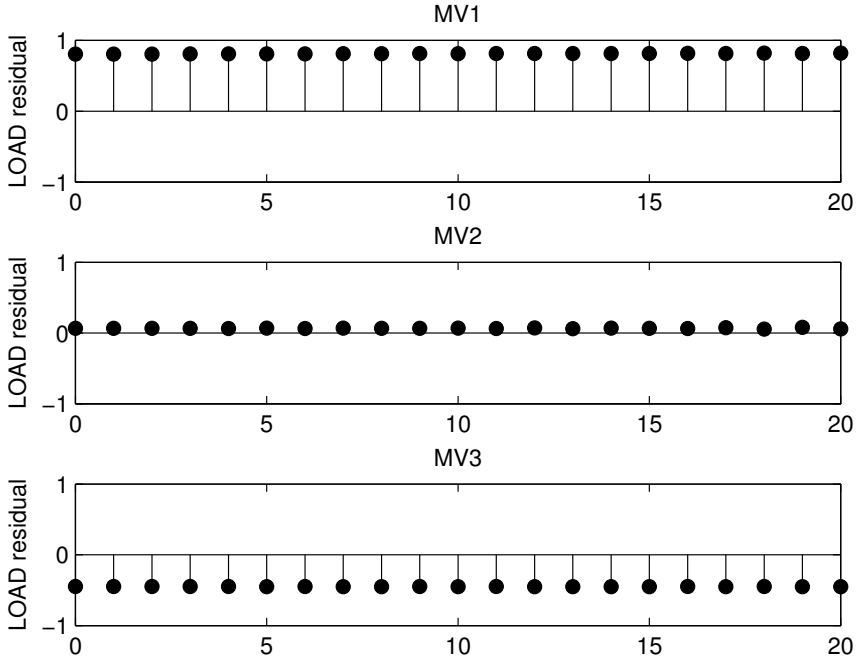

Figure 4.9: Partial correlation plots between the MVs and the LOAD residual for gain mismatch in g_{21}.

detection algorithm were based on the output error (OE) structure. The OE models were both specified with orders equal to 3.

The relative uncertainty matrix for the gains in the linearised model is reported by Craig and MacLeod (1995) to be:

$$k_{ij}: \begin{bmatrix} 31\% & 14\% & 35\% \\ 65\% & 11\% & 16\% \\ - & - & - \end{bmatrix}. \tag{4.40}$$

From the relative uncertainty matrix it is noted that the LOAD/CFF transfer function has a relatively large uncertainty. With reference to this result, another simulation run is performed in which the gain of the LOAD/CFF transfer function is perturbed. The gain of g_{21} is increased by 50% and the partial correlation plots are shown in Fig. 4.9.

From Fig. 4.9 it is clear that the significant mismatch is correctly identified to be in channel g_{21}. Some correlation does seem to exist in the g_{23} channel, but the amplitude is much smaller than the correlation for g_{21} and can easily be distinguished. For this simulation run a reference change was once again made for the LOAD in order to ensure sufficient excitation. The OE model structure with

order 3 was once again used to determine G_{u_i} and G_{e_j}.

In practice a set-point change for the LOAD is not very realistic. This is because the LOAD needs to be maintained at a specific value in order to achieve maximum throughput in the mill. Here the set-point change was made to ensure sufficient excitation in order to employ the MPM detection algorithm.

A power peak-seeking throughput optimiser is sometimes employed (Craig et al., 1992), which should ensure sufficient excitation in the LOAD signal. If load set-point changes are not common, sufficient excitation might result from parameter variations. Otherwise the control engineer may have to introduce set-point changes in the mill load when MPM is suspected in order to generate data for the detection algorithm.

4.6 CONCLUSION

This chapter describes a simulation study of the application of MPM detection to a ROM ore milling circuit under MPC control. The milling circuit model used is a linear time-invariant approximation of a fundamental milling circuit model. Model-plant mismatch, motivated from industrial experiments, is introduced in the model and its location in the multi-variable matrix model is correctly detected. For the MPM strategy to work adequately, sufficiently exited signals are required – if this does not occur during normal plant operation, set-point changes in the relevant outputs may have to be introduced.

CHAPTER 5 STATE AND PARAMETER ESTIMATION

5.1 INTRODUCTION

The values of the states of a system are important for advanced control techniques such as MPC (Qin and Bagwell, 2003) in which the state values are used to determine the optimal control signal. Owing to the nature of the milling process, measurements are usually not available, as measurement instruments capable of measuring the mill states are not available.

State estimation from input-output data may therefore be resorted to in order to infer the true state values. Even if measurement devices were available that could measure the actual state values, as they are in some other industries, they are usually more expensive and cumbersome to employ than a state estimation algorithm that can give good results. The employment of advanced control for which state feedback is needed is therefore usually coupled with the employment of some state estimation algorithm. This is especially true for nonlinear predictive control where nonlinear observers are necessary to produce estimates of the unmeasured state variables (Henson, 1998).

The parameters with large variances that are contained in the milling circuit have a big effect on the grinding performance (Coetzee et al., 2010) and accurate estimation of these parameter values would be valuable if they were to be incorporated into the control strategy.

In this chapter simultaneous state and parameter estimation concerning the mill is described. An overview of common estimation techniques is firstly given before the implemented particle-filtering technique is further discussed. This chapter also investigates the performance of dual estimation over simultaneous estimation in the particle-filtering framework. A sensitivity analysis compares the two estimation approaches and shows the class of systems for which dual estimation would provide superior results.

5.2 STATE ESTIMATION

Consider the general state-space representation of a dynamic system

$$x_{t+1} = f(x_t, u_t, \theta_t, v_t) \tag{5.1}$$

$$y_t = g(x_t, \theta_t, e_t) \tag{5.2}$$

where $x \in \mathbb{R}^n$ is the state vector and $y \in \mathbb{R}^m$ is the output vector, $f(\cdot)$ and $g(\cdot)$ are possibly nonlinear functions describing the state transitions and the outputs respectively, u_t contains the exogenous inputs, θ_t represents the parameters, v_t is the state noise and e_t is the measurement noise.

The objective is to estimate the true state and parameter values (x_t and θ_t), from the output measurements that are available (y_t).

5.2.1 State augmentation

In order to do simultaneous state and parameter estimation, the parameters are first augmented as states to the system, and then the procedure reduces to a pure state estimation problem. Consider the original system in (5.1) and (5.2); the original state vector (x_t) is now augmented with the parameters (θ_t) to be

$$x_t^* = \begin{bmatrix} x_t \\ \theta_t \end{bmatrix}. \tag{5.3}$$

Each parameter is now defined a state transition equation in the form

$$\theta_{k,t+1} = x_{n_x+k,t+1} = \theta_{k,t} + w_t \tag{5.4}$$

where w_t is the parameter noise, n_x is the original number of states and $k \in 1, \ldots, n_p$ with n_p the number of parameters to be estimated. The model used to describe the parameter transitions in (5.4) is called a random walk model, in which no dynamic transition term is included. This is because the parameters are not expected to change in a deterministic way through a dynamic model, as the states do. Any change in the parameter is attributed to the noise term only. This noise term will later be a tuning knob in the parameter estimation procedure as a large variance in w_t makes for faster tracking but with a larger estimation variance. A small variance in w_t will make the tracking much smoother but also much slower. The system equations now become

$$x_{t+1}^* = f(x_t^*, u_t, v_t^*) \tag{5.5}$$

$$y_t = g(x_t^*, e_t) \tag{5.6}$$

with v_t^* the augmented noise vector $v_t^* = \begin{bmatrix} v_t \\ w_t \end{bmatrix}$ comprising the original state noise and the parameter noise. The problem has now reduced to a pure state estimation problem, such that only state estimation can be discussed in general without excluding parameter estimation.

It should however be noted that this procedure generally makes linear state and parameter estimation problems nonlinear state estimation problems.

5.2.2 State estimation procedure

Let $Y_t = \{y_0, \ldots, y_t\}$ represent the sequence of all measurements from the initial time up to the current time, then the general state estimation problem is formulated as the solution of the conditional distribution function $p(x_t|Y_t)$, which is the distribution of the states given all the observations up to time t. The general solution to the latter is then given by (Bergman, 1999):

$$p(x_t|Y_{t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|Y_{t-1})dx_t \tag{5.7}$$

$$p(x_t|Y_t) = \frac{p(y_t|x_t)p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})} \tag{5.8}$$

where (5.7) is also known as the Chapman-Kolmogorov equation. Equations (5.7) and (5.8) give a recursive procedure for state estimation that consists of two steps:

1. Prediction step: Predict the pdf (probability density function), $p(x_t|Y_{t-1})$ from the state transition equation $f(\cdot)$ via (5.7).

2. Update step: Determine $p(x_t|Y_t)$ from $p(x_t|Y_{t-1})$ via (5.8).

Because $p(y_t|Y_{t-1})$ is a constant given all observations up to time t, it can be treated as a normalising constant and (5.8) can be represented as

$$p(x_t|Y_t) = \alpha p(y_t|x_t)p(x_t|Y_{t-1}). \tag{5.9}$$

The recursive procedure for Bayesian state inference can then be graphically represented as in Fig. 5.1 (from Huang (2011)).

This recursive procedure forms the basis of the optimal Bayesian solution but is only a conceptual solution, as the equations (5.7) and (5.8) are not computable in general (Arulampalam et al., 2002).

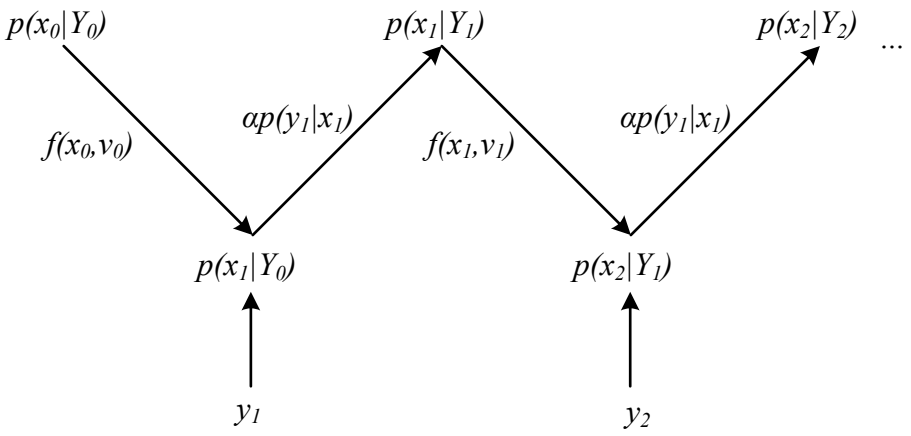

Figure 5.1: Recursive procedure for Bayesian state inference (reproduced from Huang (2011)).

Solutions do exist in special cases such as for the Kalman filter, but when the analytic solution is intractable, approximations are needed that converge to the optimal Bayesian solution. The Kalman filter and some common approximate methods are subsequently discussed.

Once the conditional distribution function $p(x_t|Y_t)$ is known, the state estimate is determined as a point estimate from the distribution as

$$\hat{x}_t = \Theta\left(p(x_t|Y_t)\right) \tag{5.10}$$

where $\Theta(\cdot)$ is a nonlinear function operator. Generally the mean, given by

$$\bar{x}_{t|t} = \int x_t p(x_t|Y_t) dx_t, \tag{5.11}$$

or the Maximum a Posteriori (MAP) estimate, given by

$$MAP(x_{t|t}) = \max\{p(x_t|Y_t)\}, \tag{5.12}$$

is used to calculate the point estimate. The notation $x_{t|t}$ describes the value of x_t given all the measurements up to time t. This notation will also be used in the rest of this chapter. The choice of the function to use in determining the point estimate depends on the form of the distribution. For Gaussian distributions both functions return the same values, but for bi-modal distributions the results will be much different. For a bi-modal distribution the MAP estimate is usually preferred.

5.3 KALMAN FILTERING

Kalman filtering, as introduced by Kalman (1960), is a popular way of solving the optimal Bayesian estimation problem. The Kalman filter gives optimal estimates if the underlying system equations are

linear and the posterior distribution is assumed to be Gaussian at every time step (Huang, 2011), and hence fully specified by the mean and covariance.

If $p(x_{t-1}|Y_{t-1})$ is Gaussian, $p(x_t|Y_t)$ will also be Gaussian provided (Ho and Lee, 1964):

1. The state and measurement noise are additive Gaussian noise with known statistics,

2. the state transition equations are known and linear and

3. the measurement equations are known and linear.

This requires that it be possible to write the system equations as

$$x_{t+1} = F_t x_t + B_t u_t + v_t \tag{5.13}$$

$$y_t = H_t x_t + e_t, \tag{5.14}$$

where F_t, B_t and H_t are matrices describing the state transition and output equations respectively. Zero mean Gaussian noise is assumed for the state and output noise with covariances Q and R respectively.

The prediction step of the optimal Bayesian estimation solution is then performed, given the distribution at the previous time step ($p(x_{t-1}|Y_{t-1})$) characterised by the mean $x_{t-1|t-1}$ and covariance $P_{t-1|t-1}$, by calculating the predicted mean and covariance as:

$$\hat{x}_{t|t-1} = F_t \hat{x}_{t-1|t-1} + B_t u_t \tag{5.15}$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t. \tag{5.16}$$

Once the current measurement becomes available the innovation (measurement residual) is calculated as

$$\tilde{y}_t = y_t - H_t \hat{x}_{t|t-1}, \tag{5.17}$$

and the innovation covariance is calculated as

$$S_t = H_t P_{t|t-1} H_t^T + R_t. \tag{5.18}$$

The optimal Kalman gain is calculated from

$$K_t = P_{t|t-1} H_k^T S_t^{-1}, \tag{5.19}$$

and then the updated state estimate (mean) and updated estimate covariance is calculated as

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t \tilde{y}_t \tag{5.20}$$

$$P_{t|t} = (I - K_t H_t) P_{t|t-1}. \tag{5.21}$$

The regular Kalman filter does not require the underlying system equations to be linear nor that the probability densities be Gaussian (Van der Merwe, 2004), but is no longer optimal if either of these restrictions do not hold. If the underlying system is nonlinear, the EKF, as discussed in the following section, is a very popular alternative.

5.3.1 Extended Kalman filter

The EKF (Jazwinski, 1970) is probably the most popular estimation algorithm for nonlinear systems (Van der Merwe, 2004). The basic idea of the EKF is to first linearise the system equations around the current mean and covariance by using a first order Taylor series expansion. This does however require the system equations to be differentiable.

Given that the system is now represented by (5.5) and (5.6), the Jacobians of the state transition equations are calculated from

$$\hat{F}_t = \nabla_x f(x, \bar{v}, u_t)|_{x=\hat{x}_{t-1|t-1}} \tag{5.22}$$

$$G_v = \nabla_v f(\hat{x}_{t-1|t-1}, v, u_t)|_{v=\bar{v}} \tag{5.23}$$

and then the prediction step of the Bayesian inference algorithm can be applied by using these linearised versions of the state transition equations. The predicted state mean and covariance are then calculated from

$$\hat{x}_{t|t-1} = f(\hat{x}_{t-1|t-1}, u_t, \bar{v}) \tag{5.24}$$

$$P_{t|t-1} = \hat{F}_t P_{t-1|t-1} \hat{F}_t^T + G_v Q_t G_v^T. \tag{5.25}$$

The observation model Jacobians are calculated from

$$\hat{H}_t = \nabla_x h(x, \bar{e})|_{x=\hat{x}_{t|t-1}} \tag{5.26}$$

$$D_e = \nabla_v h(\hat{x}_{t|t-1}, e)|_{e=\bar{e}} \tag{5.27}$$

and when the current measurement becomes available the update step can be applied. Firstly the innovation is calculated as

$$\tilde{y}_t = y_t - h(\hat{x}_{t|t-1}, e), \tag{5.28}$$

and then the innovation covariance is calculated as

$$S_t = \hat{H}_t P_{t|t-1} \hat{H}_t^T + D_e R_t D_e^T . \tag{5.29}$$

The equations for approximated optimal Kalman gain as well as for the updated state estimate and estimate covariance are the same as they were for the regular Kalman filter, but the linear observation matrix (H) should now be substituted for the observation Jacobian ($\hat{H}$).

The simple extension from the regular Kalman filter is evident, and often the EKF is sufficient for nonlinear state estimation. It should however be remembered that random variables with inherent uncertainty are estimated; this is disregarded by the linearisation around a single point, which has large implications for the accuracy and consistency of the EKF algorithm (Van der Merwe, 2004).

Another drawback of the EKF is the need to calculate the Jacobian matrices. For simple systems this is not much of a problem, but for more complex systems of high dimensionality this quickly becomes significant. The EKF also only uses a first order Taylor series approximation of the nonlinear system, which becomes inaccurate if the nonlinearities are severe. To overcome this problem, a second order EKF may be used, which uses a second order Taylor series expansion. This will increase the accuracy of the estimated underlying nonlinear system, but significantly increases the implementation complexity, as the Hessian matrices must now also be calculated. This is once again a major drawback for complex and high dimensionality systems.

5.3.2 Unscented Kalman filter

The amount of computational effort required to calculate Jacobian matrices, and possibly Hessian matrices, coupled with the fact that the EKF is in some cases inaccurate and even unstable (Van der Merwe, 2004), prompted the need for the introduction of a new extension to the Kalman filter.

The UKF was introduced by Julier and Uhlmann (1997) and is based on the unscented transform[1], which is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation. Because a Gaussian distribution is fully specified by its mean and covariance, it is easier to approximate a Gaussian distribution than it is to approximate some arbitrary nonlinear function or transformation (Julier and Uhlmann, 1997). The mean and covariance of the transformed

[1]The reason why this transformation is named “unscented” is unknown despite the efforts of Van der Merwe (2004) to discover the origin.

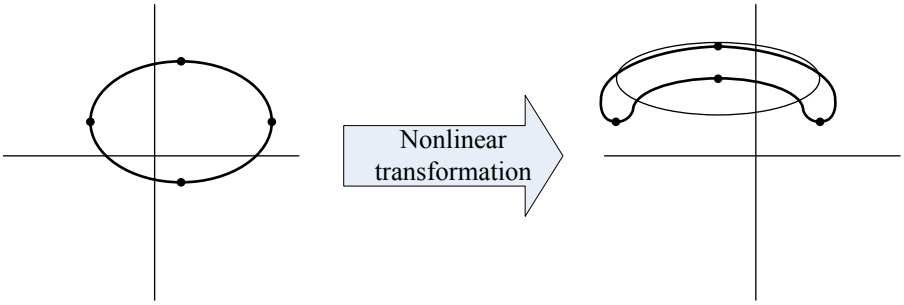

Figure 5.2: Illustration of the unscented transform.

distribution are calculated by transforming several deterministically chosen sample points (known as sigma points) through the nonlinear function.

The unscented transform is illustrated in Fig. 5.2. From the original distribution on the left side of the figure, with mean and covariance equal to $\bar{x}$ and P_x, a set of deterministically chosen sigma points are then transformed through the known nonlinear equations, as shown on the right side of the figure. The estimate of the output statistics $\bar{y}$ and P_y are then calculated from the transformed sigma points. The Gaussian distribution represented by the output statistics are also shown on the right side of the figure. The UKF therefore makes use of statistical linearisation, as opposed to the algebraic linearisation used by the EKF.

The sigma points, $S_i = \{w_i, \mathscr{X}_i\}$, used to calculate the transformed statistics, are deterministically chosen such that they completely capture the true mean and covariance of the prior random variable x. A possible selection scheme that satisfies this requirement (Van der Merwe, 2004) is then used to calculate the sigma points:

$$\begin{aligned}
\mathscr{X}_i &= \bar{x} & w_i &= \tfrac{\lambda}{L+\lambda} & i &= 0 \\
\mathscr{X}_i &= \bar{x} + \left(\sqrt{(L+\lambda)P_x}\right)_i & w_i &= \tfrac{1}{2(L+\lambda)} & i &= 1,\ldots,L \\
\mathscr{X}_i &= \bar{x} - \left(\sqrt{(L+\lambda)P_x}\right)_i & w_i &= \tfrac{1}{2(L+\lambda)} & i &= L+1,\ldots,2L
\end{aligned} \tag{5.30}$$

for the $2L+1$ sigma points, where L is the dimensionality of the random variable x. The weight associated with the i-th sigma point is w_i and $\Sigma_{i=0}^{2L} w_i = 1$. λ is given by

$$\lambda = \alpha^2(L+\kappa) - L \tag{5.31}$$

where κ and α are scaling parameters and $\left(\sqrt{(L+\lambda)P_x}\right)_i$ is the i-th column (or row) of the matrix square root of the weighted covariance matrix. κ should be chosen to be non-negative to ensure positive semi-definiteness of the covariance matrix; the specific value of κ is not critical and is usually

chosen as $\kappa = 0$. α determines the spread of the sigma points and should be chosen between 0 and 1.

The predicted state mean and covariance are then calculated as

$$\mathscr{X}_{t|t-1} = f(\mathscr{X}_{t-1}, u_{t-1}) \tag{5.32}$$

$$\hat{x}_{t|t-1} = \sum_{i=0}^{2L} w_i^m \mathscr{X}_{i,t|t-1} \tag{5.33}$$

$$P_{t|t-1} = \sum_{i=0}^{2L} w_i^c \left(\mathscr{X}_{i,t|t-1} - \hat{x}_{t|t-1}\right)\left(\mathscr{X}_{i,t|t-1} - \hat{x}_{t|t-1}\right)^T. \tag{5.34}$$

Here w^m is the weight use to calculate the mean and w^c is the weight used to calculate the covariance; $w^m = w$ as given in (5.30); $w^c = \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta)$ for $i = 0$ and $w^c = w^m$ for $i = 1, \ldots, 2L$. Here β is another scaling parameter that affects the weight of the zeroth sigma point for the calculation of the covariance. The value of β should be non-negative and can be used to incorporate knowledge of the higher order moments of the distribution. For a Gaussian prior the optimal choice is $\beta = 2$ (Julier, 2002). The transformed sigma points are then calculated and used to calculate the estimated output mean and covariances as

$$\mathscr{Y}_{t|t-1} = h\left(\mathscr{X}_{t|t-1}\right) \tag{5.35}$$

$$\hat{y}_{t|t-1} = \sum_{i=0}^{2L} w_i^m \mathscr{Y}_{i,t|t-1} \tag{5.36}$$

$$P_{\tilde{y}} = \sum_{i=0}^{2L} w_i^c \left(\mathscr{Y}_{i,t|t-1} - \hat{y}_{t|t-1}\right)\left(\mathscr{Y}_{i,t|t-1} - \hat{y}_{t|t-1}\right)^T \tag{5.37}$$

$$P_{xy} = \sum_{i=0}^{2L} w_i^c \left(\mathscr{X}_{i,t|t-1} - \hat{x}_{t|t-1}\right)\left(\mathscr{Y}_{i,t|t-1} - \hat{y}_{t|t-1}\right)^T. \tag{5.38}$$

The Kalman gain is then calculated as

$$K_t = P_{xy} P_{\tilde{y}}^{-1} \tag{5.39}$$

and finally the updated state mean and covariance are calculated

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - \hat{y}_{t|t-1} \tag{5.40}$$

$$P_{t|t} = P_{t|t-1} - K_t P_{\tilde{y}} K_t^T. \tag{5.41}$$

The UKF is a great improvement on the EKF, firstly because there is no need for the calculation of the Jacobian or Hessian matrices. This allows the UKF to deal with discontinuous functions and functions that are continuous but have points of non-differentiability. Secondly, high order information about

the distribution can be captured using only a few sigma points (Julier and Uhlmann, 1997), which makes the algorithm computationally efficient. In the original presentation of the UKF, Julier and Uhlmann argued that because of the ease of implementation and improved accuracy of the UKF, its use is recommended over the EKF in virtually all applications.

The posterior distribution is however still approximated to be Gaussian, which is a major limitation for systems where the nonlinearities in the process equations are severe, or where the posterior distribution is multi-modal.

5.4 SEQUENTIAL MONTE CARLO METHODS

Sequential Monte Carlo (SMC) methods make no explicit assumptions about the form of the posterior distribution and are therefore applicable to general nonlinear, non-Gaussian systems (Douchet et al., 2000). These methods approximate the Gaussian integrals, as listed in Section 5.2.2, with finite sums calculated about a set of weighted samples drawn from a proposal distribution.

Particle filtering is a technique of practically implementing a recursive Bayesian filter by Monte Carlo simulations. The required posterior probability density function (pdf) is represented by a set of random samples and associated weights. The idea of representing a pdf in this fashion is illustrated in Fig. 5.3. 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 posterior density function at time t may then be approximated as (Arulampalam et al., 2002):

$$p(x_t|Y_t) \approx \sum_{i=1}^{N_s} w_t^i \delta(x_t - x_t^i) \tag{5.42}$$

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 et al., 2004)

$$w_t^i \propto w_{t-1}^i \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, y_t)} \tag{5.43}$$

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 $p(x_t|Y_t)$ but as this is not known in general, a proposal

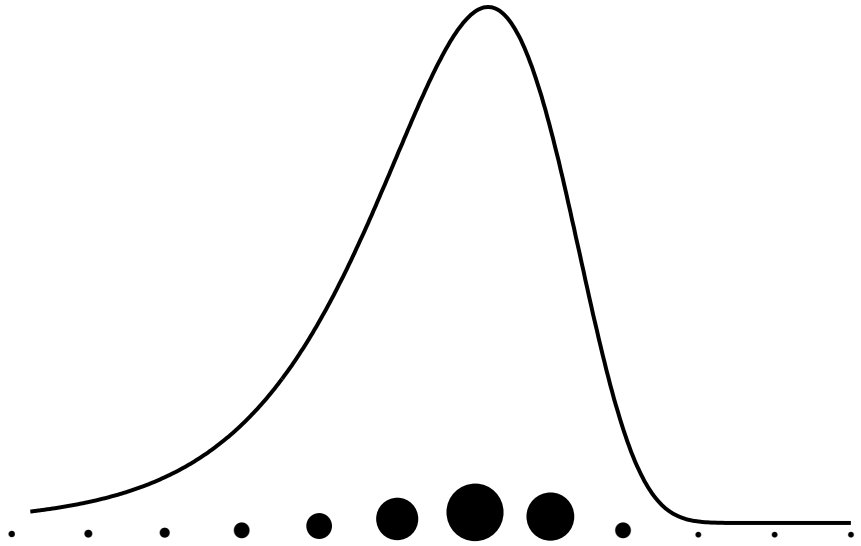

Figure 5.3: PDF represented by particles and associated weights.

distribution is used. The importance density plays an important role in the design and is discussed in Section 5.4.2.

5.4.1 Degeneracy

It has been shown (Douchet et al., 2000) that the variance of the importance weights of (5.43) can only increase over time. This has a harmful effect on accuracy and is known as degeneracy because after only a couple of iteration steps all but one particle will have negligible normalised weights. Practically this means that a lot of computational time is spent to update particles that produce negligible contributions to the approximation of the posterior distribution. An indication of the degree of degeneracy is the effective number of particles defined by

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_s} (w_t^i)^2}. \tag{5.44}$$

If the effective number of particles becomes significantly smaller than the actual number of particles used, the degeneracy in the algorithm is significant.

Degeneracy may be eliminated through the use of resampling that eliminates particles with low importance weights and multiplies particles with high importance weights. This means that the particles located at x_t^i with weights w_t^i are replaced by particles located at x_t^{i*} with uniform weights. One pos-

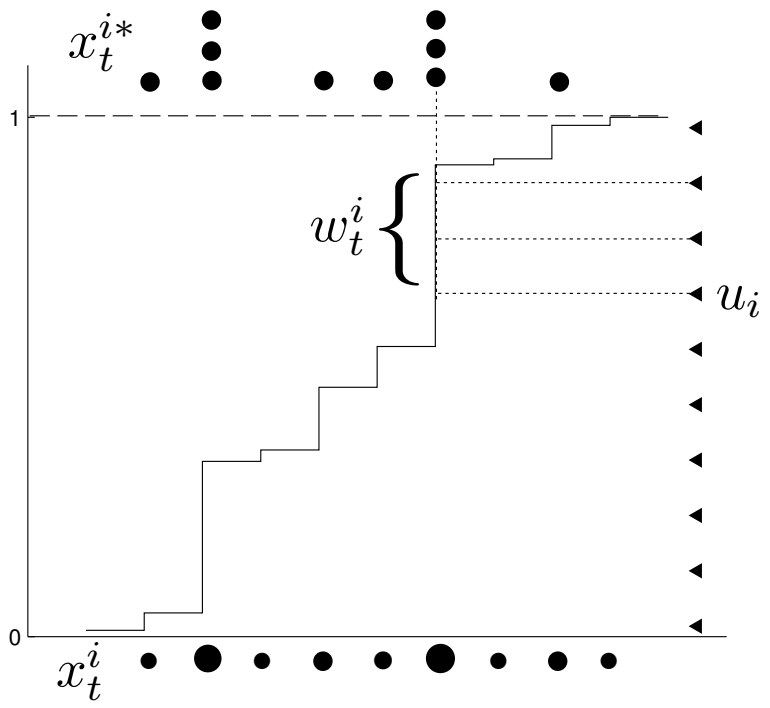

Figure 5.4: Resampling procedure.

sible resampling scheme is systematic resampling (Kitagawa, 1996), for which the pseudo-code is listed as Algorithm 1. The algorithm uses the cumulative sum of weights (CSW) defined as

$$CSW_i = \sum_{j=1}^{i} w_t^j \tag{5.45}$$

and draws samples from $u \in [0,1]$ to map new samples from the CSW. This process is illustrated in Fig. 5.4 where the original particles that give form to the CSW are shown at the bottom of the figure. Samples are uniformly drawn at the points indicated by the left-pointing triangles and based on the value of the CSW function at that point, the resampled particles (x_t^{i*}) are determined from the locations of the original particles (x_t^i). From Fig. 5.4 it is visible that particles with larger weights have a bigger chance of being drawn (and consequently duplicated) than particles with smaller weights.

5.4.2 Importance density

The optimal importance density function that minimises the variance of importance weights conditioned upon x_{t-1}^i and y_t has been shown (Douchet et al., 2000) to be

$$q(x_t|x_{t-1}^i, y_t)_{opt} = p(x_t|x_{t-1}^i, y_t) \tag{5.46}$$

$$= \frac{p(y_t|x_t, x_{t-1}^i)p(x_t|x_{t-1}^i)}{p(y_t|x_{t-1}^i)}. \tag{5.47}$$

Algorithm 1 Systematic resampling

$[\{x_t^{j*}, w_t^j\}_{j=1}^N] = \text{RESAMPLE}[\{x_t^i, w_t^i\}_{j=1}^N]$

- Initialise the CSW: $c_1 = w_t^1$
- FOR $i = 2 : N$
  - Construct CSW: $c_i = c_{i-1} + w_t^i$
- END FOR
- Begin at bottom of the CSW: $i = 1$
- Draw a starting point: $u_1 \sim \mathscr{U}[0, \frac{1}{N}]$
- FOR $j = 1 : N$
  - Move along the CSW: $u_j = u_1 + \frac{1}{N}(j-1)$
  - Find the smallest value of i such that $u_j \leqslant c_i$
  - Assign sample: $x_t^{j*} = x_t^i$
  - Assign weight: $w_t^j = \frac{1}{N}$
- END FOR

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

$$p(y_t|x_{t-1}^i) = \int p(y_t|x_t)p(x_t|x_{t-1}^i)dx_t \tag{5.48}$$

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

$$q(x_t|x_{t-1}^i, y_t) = p(x_t|x_{t-1}^i) \tag{5.49}$$

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

$$p(x_t|x_{t-1}^i) = \mathscr{N}(x_t; f_{t-1}(x_{t-1}^i), \mathbf{Q}_{t-1}) \tag{5.50}$$

which can easily be calculated.

5.4.3 Sampling importance resampling particle filter

Now that all the elements of the particle filtering algorithm have been described, the complete algorithm can be given. The sampling importance resampling (SIR) particle filter (Gordon et al., 1993) is a version of the general particle filtering algorithm (Arulampalam et al., 2002) in which the importance density is chosen as the transitional prior and resampling is done at each step. The assumptions

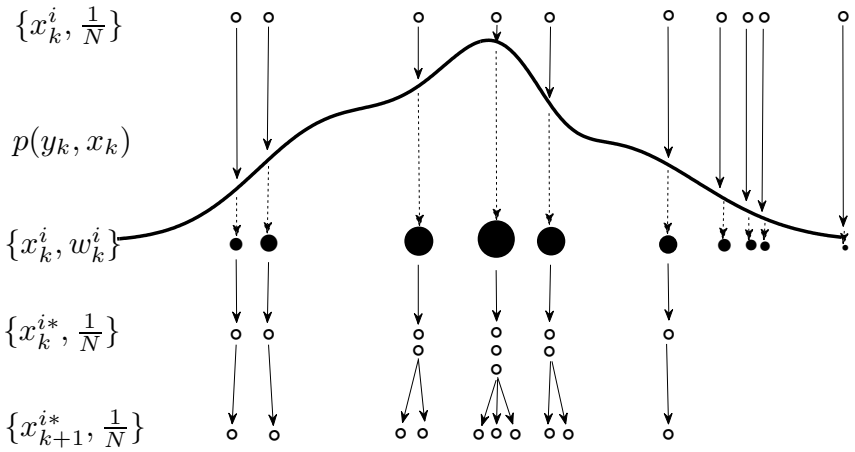

Figure 5.5: One iteration of the SIR particle filter algorithm (adapted from Ristic et al. (2004)).

required to use the SIR particle filter are very weak (Ristic et al., 2004), making it suitable for a wide variety of implementations. The SIR type particle filter is used throughout in this study.

One iteration of the SIR particle filter algorithm is presented in Fig.5.5; the pseudo-code describing the algorithm is listed as Algorithm 2. In Fig. 5.5 it can be seen that the uniformly weighted particles $\{x_k^i, \frac{1}{N}\}$ from the previous iteration are available, and used as the input to the algorithm. The particles are allocated their normalised weights based on the value of posterior density. The particles are still located at the same locations but now have specific weights and are given by $\{x_k^i, w_k^i\}$. Next the particles are resampled, and it can be seen that particles with large weights are duplicated while particles with small weights are discarded. The resampled particles are represented in the figure as $\{x_k^{i*}, \frac{1}{N}\}$, and it should be noted that the particles once again have uniform weights. The particle locations for the next iteration are then determined by randomly sampling around the current locations based on the specified noise variance.

Because resampling is done at every time step, it is unnecessary to pass the weights on between successive iterations, as all the weights are uniform $w_t^i = \frac{1}{N}$.

Algorithm 2 SIR particle filter

$[\{x_t^i\}_{i=1}^N] = \text{SIR}[\{x_{t-1}^i\}_{i=1}^N, y_t]$

- FOR $i = 1 : N$
  - Draw $x_t^i \sim p(x_t|x_{t-1})$
  - Calculate $\tilde{w}_t^i = p(y_t|x_t)$
- END FOR
- Calculate the sum of the weights: $\Sigma_w = \text{SUM}[\{\tilde{w}_t^i\}]$
- FOR $i = 1 : N$
  - Normalise: $w_t^i = \frac{\tilde{w}_t^i}{\Sigma_w}$
- END FOR
- Resample using Algorithm 1

  $[\{x_t^i\}_{i=1}^N] = \text{RESAMPLE}[\{x_t^i, w_t^i\}_{i=1}^N]$

5.5 DUAL PARTICLE FILTERING

Under certain circumstances, which will be explored in Section 5.6, simultaneous state and parameter estimation through augmenting the parameters as states might not produce good results. In these situations dual estimation may be implemented, which makes use of two particle filters running in parallel, as shown in Fig. 5.6. Here one particle filter is used solely for state estimation and the other solely for parameter estimation.

In the dual estimation framework the two particle filters operate in an iterative fashion. At each time step the current state estimate is given as a known input to the particle filter doing parameter estimation. Similarly, the current parameter estimate is given as a known input to the particle filter doing the state estimation. The state estimator draws particles as

$$x_t^i \sim p(x_t|x_{t-1}^i, \hat{\theta}_{t-1}) \tag{5.51}$$

and then the weights are calculated as

$$\tilde{w}_{t,x}^i = p(y_t|x_t^i). \tag{5.52}$$

The parameter estimator draws particles as

$$\theta_t^i \sim p(\theta_t|\theta_{t-1}^i, \hat{x}_{t-1}) \tag{5.53}$$

and then the weights are calculated as

$$\tilde{w}_{t,\theta}^i = p(\hat{x}_t|\theta_t^i, \hat{x}_{t-1}). \tag{5.54}$$

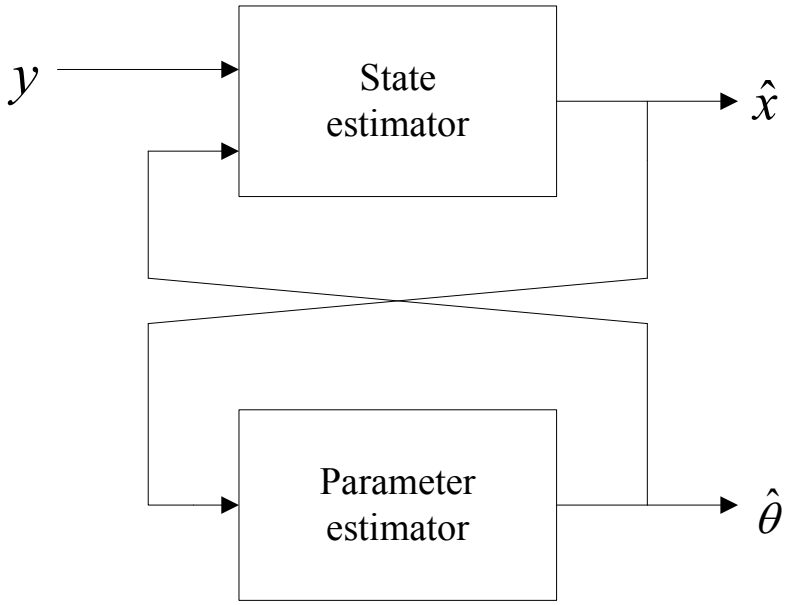

Figure 5.6: Dual state and parameter estimation.

The SIR particle filter given in Algorithm 2 is still directly applicable in this case although the samples are drawn from different distributions and different transitional priors are used. This method has previously been implemented with dual Kalman filers as discussed by Van der Merwe (2004) but as far as the author knows it has not been done with dual particle filters.

5.6 SENSITIVITY ANALYSIS

The effect a parameter or a state has on the output is a good indication of how easily that parameter or state can be estimated from output data. The investigation of these effects is done through a sensitivity analysis (Jayasankar et al., 2009).

The output sensitivity for the simultaneous estimation algorithm with respect to the augmented state vector (x^*) from Section 5.2.1 gives the output sensitivity with respect to the states and parameters

and is given by

$$S_{y,s} = \frac{\partial y}{\partial x^*} \tag{5.55}$$

$$= \frac{\partial g}{\partial x}\frac{\partial x}{\partial x^*} + \frac{\partial g}{\partial \theta}\frac{\partial \theta}{\partial x^*} \tag{5.56}$$

$$= \frac{\partial g}{\partial x}\begin{bmatrix} \frac{\partial x}{\partial x} \\ \frac{\partial x}{\partial \theta} \end{bmatrix} + \frac{\partial g}{\partial \theta}\begin{bmatrix} \frac{\partial \theta}{\partial x} \\ \frac{\partial \theta}{\partial \theta} \end{bmatrix} \tag{5.57}$$

$$= \frac{\partial g}{\partial x}\begin{bmatrix} \mathrm{I} \\ \frac{\partial x}{\partial \theta} \end{bmatrix} \tag{5.58}$$

$$= \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial x}\frac{\partial x}{\partial \theta} \end{bmatrix} \tag{5.59}$$

where the fact that in this case $\frac{\partial g}{\partial \theta} = 0$ was used to go from (5.57) to (5.58). This is because the parameters are not contained in the output equations, but only in the state transition equations. The first row of (5.59) applies to state estimation and the second row to parameter estimation. The partial derivatives $\frac{\partial g}{\partial x}$ can easily be determined from $y = g(x^*)$ but the derivatives $\frac{\partial x}{\partial \theta}$ are not directly calculable for lack of a direct expression for the state values. The time derivative of the states are however known through (5.5) from which $\frac{\partial x}{\partial \theta}$ may be derived by writing

$$\frac{dx}{dt} = f(x,u), \tag{5.60}$$

then taking the partial derivative to θ on both sides to give

$$\frac{\partial}{\partial \theta}\left(\frac{dx}{dt}\right) = \frac{\partial f}{\partial \theta} + \frac{\partial f}{\partial x}\frac{\partial x}{\partial \theta} \tag{5.61}$$

and then changing the order of differentiation to give

$$\frac{d}{dt}\left(\frac{\partial x}{\partial \theta}\right) = \dot{S}_x = \frac{\partial f}{\partial \theta} + \frac{\partial f}{\partial x}S_x. \tag{5.62}$$

S_x is then found by integrating the function (5.62) from the initial time (t_o) to the final time (t_f).

For the dual estimation algorithm, a change in a parameter causes a change in the states and then the state transition equations are used to calculate the particle weights for parameter estimation (see (5.54)). This means that the sensitivity function to consider in this case is $\frac{\partial x}{\partial \theta} = S_x$ as defined previously. This change in the state will then cause the output to change, from which the particle weights are calculated for state estimation (see (5.52)). The sensitivity function describing this effect is $\frac{\partial y}{\partial x}$. The output sensitivity function for the dual estimation algorithm is therefore given by

$$S_{y,d} = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial x}{\partial \theta} \end{bmatrix}. \tag{5.63}$$

The only difference between the sensitivity functions $S_{y,s}$ and $S_{y,d}$ is therefore the extra $\frac{\partial g}{\partial x}$ term which is present in the second row of $S_{y,s}$ pertaining to parameter estimation. This is because in the simultaneous estimation algorithm a change in a parameter value must firstly cause a change in the state values before having an effect on the outputs, from which both the state and parameter values are then inferred. In the dual estimation scheme a change in a parameter value has a direct effect on the state transition equations from which the parameter values are inferred.

This implies that for systems where $\frac{\partial g}{\partial x}$ is ill-conditioned, deteriorated accuracy for parameter estimation would result, using the simultaneous formulation.

In order to determine the conditioning of the matrix, the condition number is calculated. This is the ratio (C) of the largest to the smallest singular values of a matrix (Datta, 1994). The condition number of a matrix gives an estimate of the worst-case loss of precision when solving a linear system with that matrix. The condition number for an $n \times n$ square matrix with elements randomly chosen from a normal distribution is of the order $\sqrt{n}$ (Turing, 1948). This result suggests that this problem would be more common in systems with high dimensionality.

5.7 OBSERVABILITY ANALYSIS

A dynamic system given by

$$\dot{x} = Ax + Bu \tag{5.64}$$

$$y = Cx + Du \tag{5.65}$$

is said to be state observable (or commonly simply said to be observable) if, for any time $t_1 > 0$, the initial state $x(0) = x_0$ can be determined from the time history of the input $u(t)$ and the output $y(t)$ in the interval $[0, t_1]$ (Skogestad and Postlethwaite, 2005). The linear system is observable if, and only if, the observability matrix

$$\mathscr{O} \triangleq \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{5.66}$$

has full column rank (rank equal to n, where n is the number of rows, or columns, in A).

For regular Kalman filtering, the asymptotic stability of the filter is guaranteed only if the dynamics of

the underlying system are linear and the system is observable and controllable (Cohn and Dee, 1988). It has also been shown (Reif et al., 1999) that if, among other conditions, a nonlinear system satisfies a nonlinear observability rank condition for every set of states over the time history, the mean square estimation error is exponentially bounded given that the initial estimation error is sufficiently small. This implies that observability is one of the requirements to ensure good estimation accuracy.

Similarly for the particle-filtering framework, convergence of the state estimate to the true state of the system depends on the system properties such as observability (Juloski et al., 2003). This does not mean that systems with unobservable states will necessary produce poor estimation results, but good estimation cannot be guaranteed for these systems. Conversely, observability of the system does not guarantee good estimation accuracy, but does go some way in quantifying the estimation accuracy that can be expected. The observability of the mill states are tested in this section.

In order to test the observability of a nonlinear system, such as given in (5.5) and (5.6), the matrix of all repeated Lie derivatives is firstly defined as

$$l(x) \triangleq \begin{bmatrix} L_f^0(g) \\ L_f^1(g) \\ \vdots \\ L_f^{n-1}(g) \end{bmatrix} \tag{5.67}$$

where $L_f^i(g)$ is the i-th Lie derivative of $g(\cdot)$ with respect to $f(\cdot)$. The Lie derivative of $g(\cdot)$ with respect to $f(\cdot)$ is defined to be

$$L_f(g) = \frac{\partial g}{\partial x} \cdot f \tag{5.68}$$

$$= \begin{bmatrix} \frac{\partial g}{\partial x_1} & \cdots & \frac{\partial g}{\partial x_n} \end{bmatrix} \begin{bmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{bmatrix}. \tag{5.69}$$

Higher order Lie derivatives can then be calculated as

$$L_f^i(g) = \frac{\partial}{\partial x}\left[L_f^{i-1}(g)\right] \cdot f. \tag{5.70}$$

The observability space of the nonlinear function is then given by

$$\mathscr{O} = \left.\frac{\partial l(x,u_0)}{\partial x}\right|_{x=x_0} \tag{5.71}$$

and must have full column rank for the nonlinear system to be observable.

Calculating the rank of the observability space for simple nonlinear systems to determine whether the system is observable is a powerful tool, but as soon as the dimensionality of the system becomes large or the equations become complex, the calculation of repeated Lie derivatives quickly turns into a non-trivial task.

This is especially true for the milling circuit equations where even the second order Lie derivatives become very complex, and because the mill has five states the Lie derivatives need to be calculated up to the fifth order. Even if the repeated Lie derivatives are calculated with some software package, the rank of the resulting observability space is needed, and since this is generally calculated by making use of the singular value decomposition (SVD) or the determinant of the matrix, symbolic manipulators struggle to produce reliable results. For example, given the matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \tag{5.72}$$

Matlab returns 2 as the rank of the matrix, which gives the impression that the matrix does have full column (or row) rank. For this example it is easy to verify that in the case where $a = 1$, $b = 1$, $c = 2$ and $d = 2$ the rank of the matrix is 1, and does not have full column rank. The indeterminacy of the true rank of a symbolic matrix from Matlab only gets worse for more complex matrices.

It is however known (Diop and Fliess, 1991) that a nonlinear system is observable at a specific operating point if, and only if, a linearised version of the system is observable at that operating point. This implies that the straightforward observability test for a linear system, given earlier in this section, can be performed on a linearised version of the nonlinear system, to evaluate the observability of the nonlinear system. This test will not provide a verdict on the global observability of the nonlinear system, but the local observability information it supplies is a good initial indication. This is especially true if the plant is not expected to undergo significant changes in its operating point.

In order to perform this test it is necessary to calculate a linearised state-space representation of the mill at the normal operating point. The normal nonlinear system equations are firstly used to test the observability of the mill as is. This test will indicate whether reasonable estimation accuracy can be expected for state estimation of the mill. Then different combinations of parameters will be augmented to the normal system equations and the observability of the augmented system will be tested. This will indicate the different parameter combinations that can be estimated (with relative accuracy), simultaneously with the mill states. The reader is reminded that observability of the system does not guarantee good estimation but is an indication of the estimation accuracy that may be expected.

The linearised state-space model is in the form

$$\partial x_{t+1} = \bar{A}\partial x + \bar{B}\partial u \tag{5.73}$$

$$\partial y = \bar{C}\partial x + \bar{D}\partial u \tag{5.74}$$

where $\bar{A}$, $\bar{B}$, $\bar{C}$ and $\bar{D}$ are the linearised state-space matrices given by

$$\bar{A} = \frac{\partial f}{\partial x^T}\bigg|_{op} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \\ \frac{\partial f_{n_x}}{\partial x_1} & & \frac{\partial f_{n_x}}{\partial x_{n_x}} \end{bmatrix}\bigg|_{op} \tag{5.75}$$

$$\bar{B} = \frac{\partial f}{\partial u^T}\bigg|_{op} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_{n_u}} \\ \vdots & \ddots & \\ \frac{\partial f_{n_x}}{\partial x_1} & & \frac{\partial f_{n_x}}{\partial x_{n_u}} \end{bmatrix}\bigg|_{op} \tag{5.76}$$

$$\bar{C} = \frac{\partial g}{\partial x^T}\bigg|_{op} = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_{n_x}} \\ \vdots & \ddots & \\ \frac{\partial g_{n_y}}{\partial x_1} & & \frac{\partial g_{n_y}}{\partial x_{n_x}} \end{bmatrix}\bigg|_{op} \tag{5.77}$$

$$\bar{D} = \frac{\partial g}{\partial u^T}\bigg|_{op} = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \cdots & \frac{\partial g_1}{\partial u_{n_u}} \\ \vdots & \ddots & \\ \frac{\partial g_{n_y}}{\partial u_1} & & \frac{\partial g_{n_y}}{\partial u_{n_u}} \end{bmatrix}\bigg|_{op} \tag{5.78}$$

where n_x, n_y and n_u are respectively the number of states (augmented states), outputs and inputs; op in the equations indicate the operating point, which is

$$OP = \begin{bmatrix} X_{mw} \\ X_{ms} \\ X_{mf} \\ X_{mr} \\ X_{mb} \\ \alpha_f \\ \alpha_r \\ \phi_f \\ MFS \\ MIW \\ MFB \end{bmatrix} = \begin{bmatrix} 8.53 \\ 9.47 \\ 3.54 \\ 20.25 \\ 6.75 \\ 0.1 \\ 0.1 \\ 28 \\ 100 \\ 33.33 \\ 2 \end{bmatrix} \tag{5.79}$$

as was presented in Table 2.2.

Table 5.1: Observability test results.

# of parameters	Parameters	$n_x + n_p$	Rank	Full rank
0	–	5	5	Yes
1	α_f	6	6	Yes
1	α_r	6	6	Yes
1	ϕ_f	6	6	Yes
2	α_f, α_r	7	7	Yes
2	α_f, ϕ_f	7	6	No
2	α_r, ϕ_f	7	7	Yes
3	α_f, α_r, ϕ_f	8	7	No

The linearised matrices are then substituted into the linear observability matrix (5.66) and the rank is determined. The results are shown in Table 5.1. In the table $n_x + n_p$ denotes the number of augmented states and therefore the required rank of the observability matrix for full column rank.

The three parameters α_f, α_r and ϕ_f are specifically chosen as they have the largest influence on the operation of the mill, and it would be beneficial to know their true values.

From Table 5.1 it is noticeable that the original system and most of the augmented systems are observable around the operating point. The only two augmented systems that are not observable are the sixth and last entries in the table. In both of these systems α_f and ϕ_f are simultaneously augmented to the original system, resulting in the augmented system not being observable. This result suggests that good estimation accuracy should not necessarily be expected when trying to estimate the values of α_f and ϕ_f simultaneously. This result is not surprising, as α_f and ϕ_f are contained in the same equation (see equation (2.17) in Section 2.3.2).

For this reason the main analysis in the next section is based on estimating the values of ϕ_f and α_r, while treating α_f as a known variable. The results achieved with dual particle filters for estimating all three parameters as well as the states will however be presented in Section 5.8.2 to show the results obtained when the system is not observable.

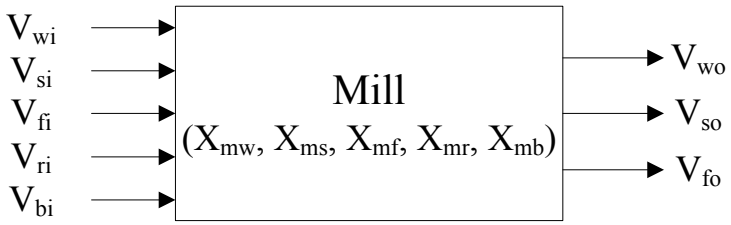

Figure 5.7: Mill, showing component flows and states.

5.8 STATE AND PARAMETER ESTIMATION FOR THE MILL

As discussed in Chapter 2, the mill has five states, which are the hold-ups of the five classifications of material in the mill, as shown in Fig. 5.7. Fig. 5.7 also shows the streams into and out of the mill that have been broken up into these separate components. Note again that no rocks nor steel balls flow out of the mill, as they cannot escape through the end discharge grate.

The state transition equations for the mill, which will be used in the particle filtering algorithm as $f(\cdot)$, were given in Chapter 2. The output equations ($g(\cdot)$) as used in the particle filtering algorithm are

$$g(x) = \begin{bmatrix} V_{wo} \\ V_{so} \\ V_{fo} \\ LOAD \\ P_{mill} \end{bmatrix}. \tag{5.80}$$

All five mill states as well as both the parameters α_r and ϕ_f are to be estimated from these equations.

5.8.1 State and parameter estimation results

In order to illustrate the accuracy of the simultaneous estimation as well as the dual estimation algorithms, a simulation run for the milling circuit is performed while kept in feedback control by the PI controller of Section 3.3 over a period of 20 hours. The value of ϕ_f is decreased by 20% at time 2 hours, the value of α_r is decreased by 20% at time 8 hours and the value of α_f is increased by 20% at time 14 hours. The parameter values are shown in Fig. 5.8. The output values of (5.80) that result from these parameter changes are shown in Fig. 5.9.

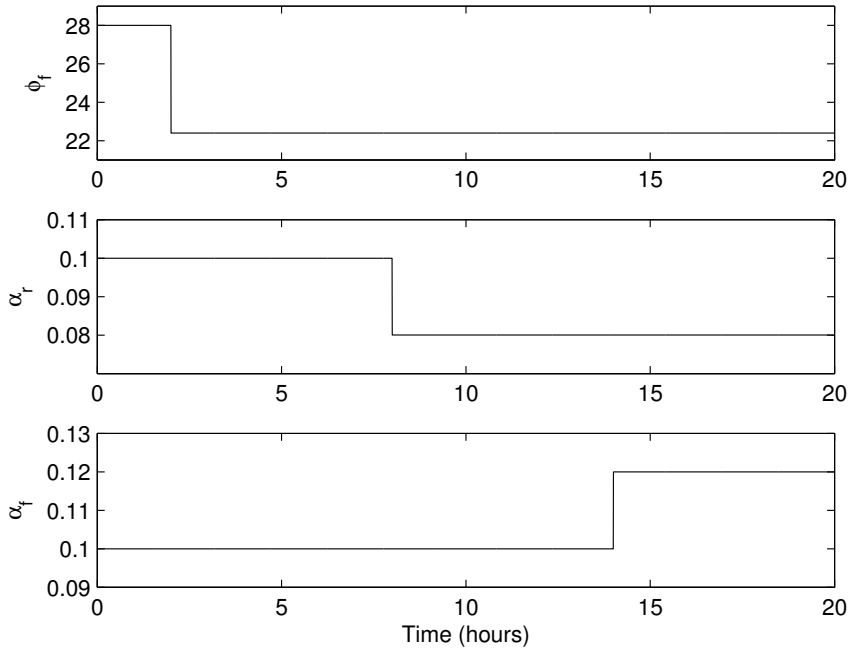

Figure 5.8: True parameter value changes for estimation.

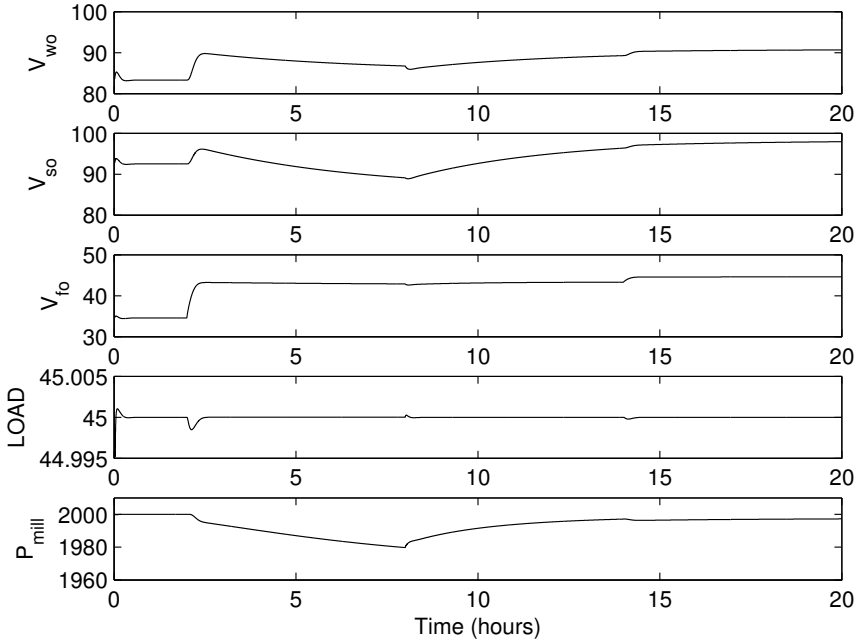

Figure 5.9: True output values for estimation.

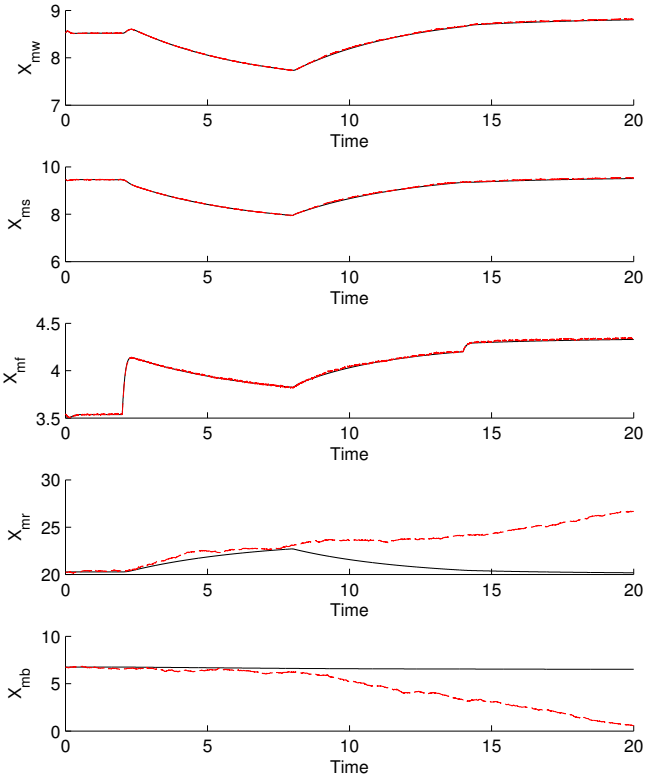

Figure 5.10: State estimates from simultaneous estimation algorithm.

The data generated by the simulation are processed by both estimation schemes to determine estimates of all five of the states and both the parameters ϕ_f and α_r. The particle filters used by the simultaneous as well as the dual estimation schemes are specified with 50 particles each. Even though the dual estimation scheme makes use of two particle filters, the sum of the dimensions equals the dimensionality of the simultaneous estimation scheme such that an equal number of function evaluations are required. The initial estimates of the states and parameters are randomly selected from a region (± 0.01) around the actual initial values in each case.

The state estimation results from the simultaneous estimation algorithm are shown in Fig. 5.10 and the parameter estimates are shown in Fig. 5.11. The state estimation results from the dual estimation algorithm are shown in Fig. 5.12 and the parameter estimates are shown in Fig. 5.13.

The condition number of the matrix $\frac{dg}{dx}$ for the entire simulation run is shown in Fig. 5.14. Here it can be seen that the condition number of this matrix is very large for the entire simulation run, resulting in ill-conditioning of the system.

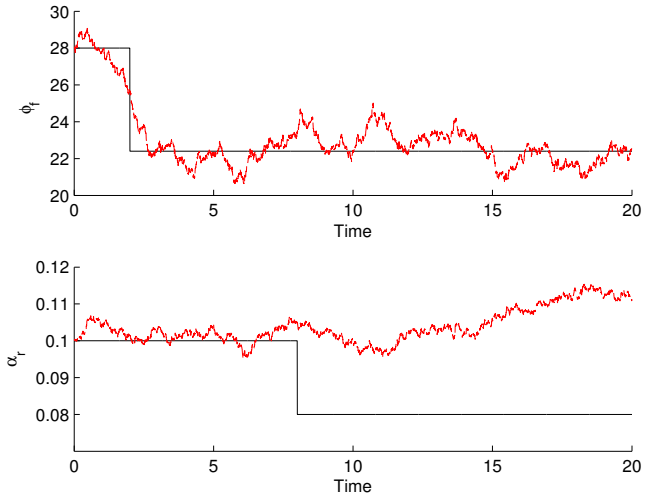

Figure 5.11: Parameter estimates from simultaneous estimation algorithm.

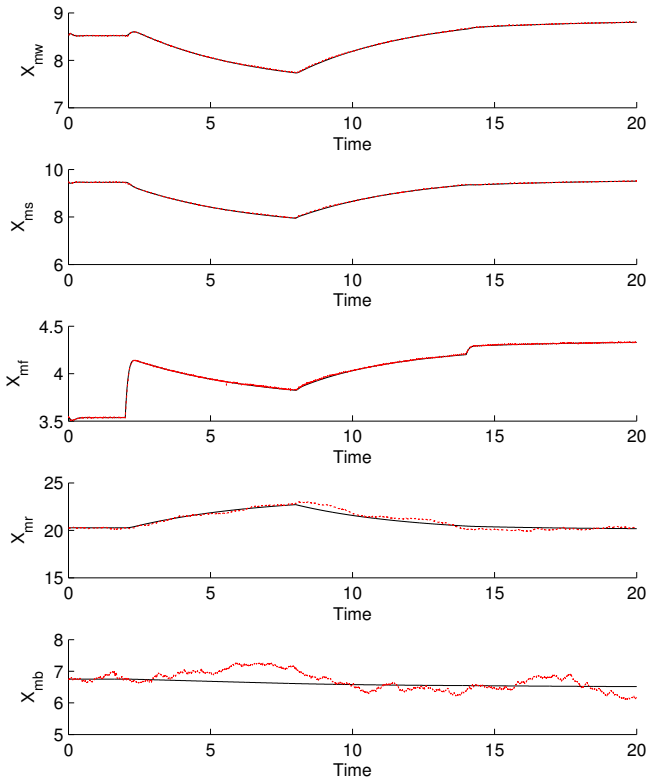

Figure 5.12: State estimates from dual estimation algorithm.

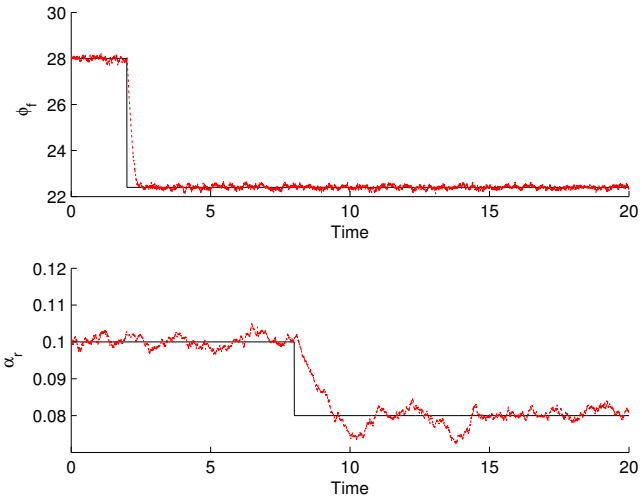

Figure 5.13: Parameter estimates from dual estimation algorithm.

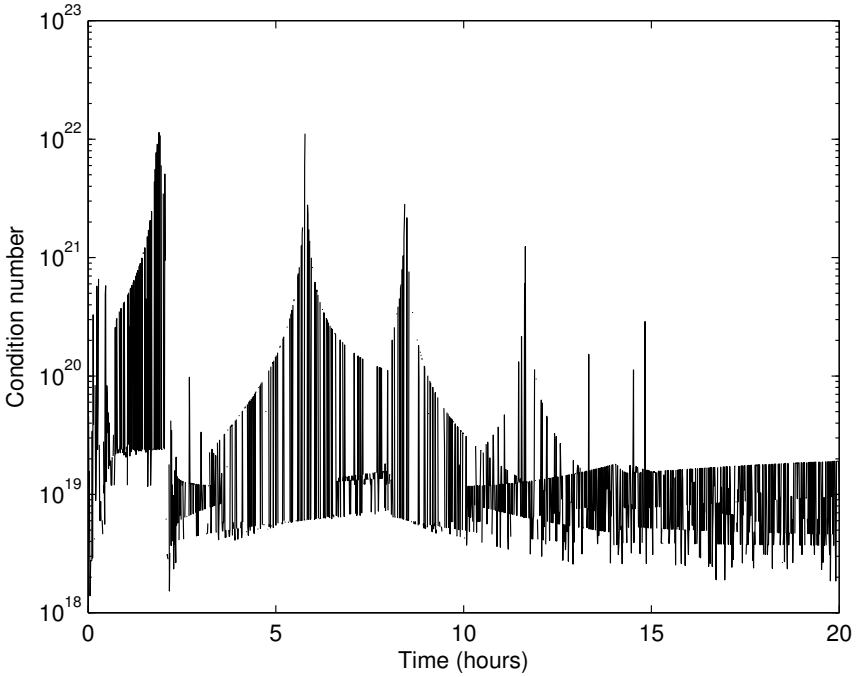

Figure 5.14: Condition number of $\frac{dg}{dx}$.

It is clear from the results that the dual state and parameter estimation scheme produces superior results to the simultaneous estimation scheme for this system. It is especially noticeable that when α_r changes value, the simultaneous estimation scheme cannot correctly attribute the amount of change in the output caused by a change in the parameter and the amount of change caused by the subsequent change in the state value. For this reason the accuracy of state and parameter estimation deteriorates.

5.8.2 State and parameter estimation results (unobservable system)

From the previous section it is clear that good estimation accuracy is achieved when using the dual estimation scheme to estimate all five states and the two parameters α_r and ϕ_f. The same simulation dataset used in the previous section to test the simultaneous and dual estimation algorithms is once again used in a dual estimation framework to estimate the values of the states and three parameters, α_r, α_f and ϕ_f. In Section 5.7 it was shown that this augmented system is unobservable, suggesting that poor estimation accuracy might result. The results obtained for the state estimates are shown in Fig. 5.15 and the parameter estimates are shown in Fig. 5.16.

In comparing these results with Figs. 5.12 and 5.13, it can be seen that the state estimates are still rather accurate, except for the estimate of X_{mr} which is not as good. The estimates for the parameters α_r and ϕ_f are still rather good, but the estimate for α_f is not very accurate. The deteriorated accuracy of the estimate for X_{mr} may be expected, as α_f is present in the state transition equation of X_{mr}, and the value of α_f is poorly estimated.

This section shows that system properties such as observability do have an effect on the estimation accuracy that may be expected. In these cases it may be necessary to include an additional measurement that can be used to extract the value of a specific parameter (or state) more accurately from the process data.

5.9 CONCLUSION

A change in a parameter value causes the state transition equation to change value. At this point the dual estimation algorithm will use the state transition equation information to infer the values of the parameters. The states will then change to cause a change in the outputs. At this point the dual estimation algorithm uses the outputs to determine the state values. It is only at this point where

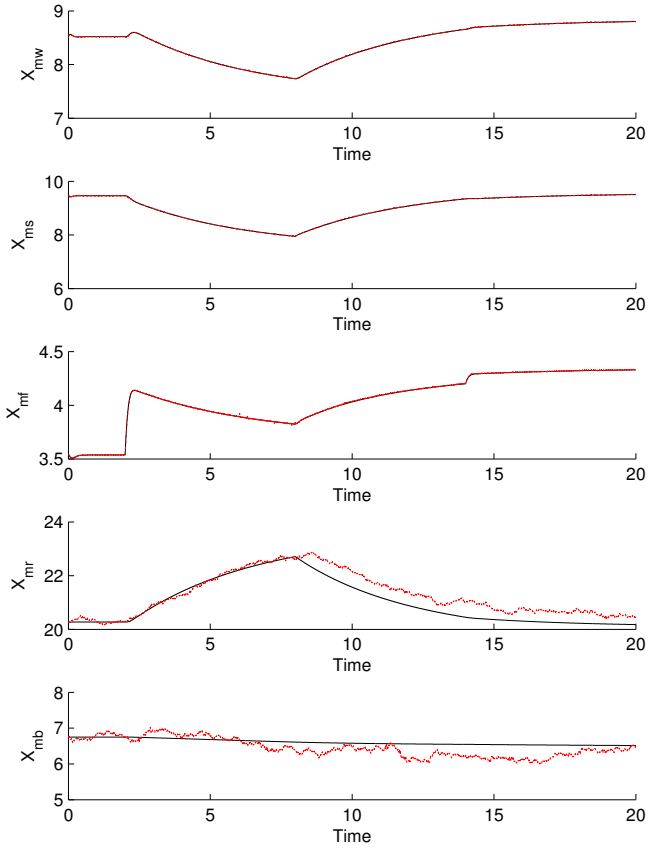

Figure 5.15: State estimates from dual estimation algorithm while estimating three parameters.

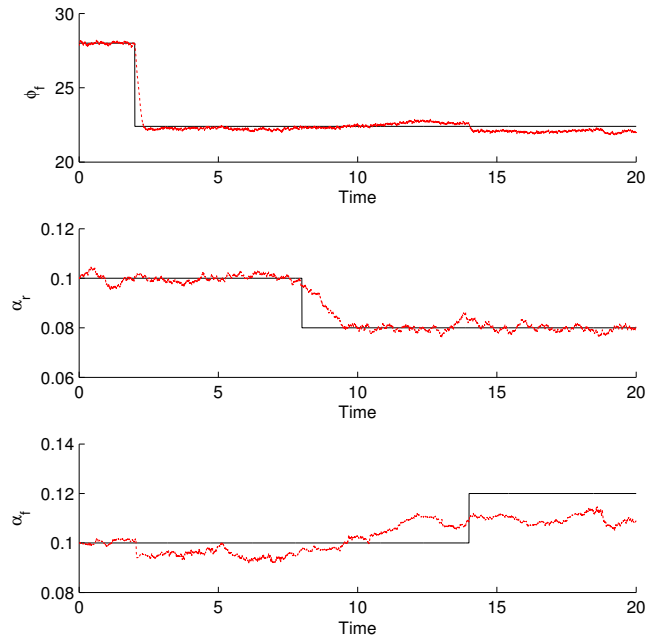

Figure 5.16: Parameter estimates from dual estimation algorithm while estimating three parameters.

the simultaneous estimation algorithm infers the values of the states and parameters. Disregarding the information supplied by the state transition equations in the simultaneous estimation scheme may lead to deteriorated estimation accuracy, depending on the conditioning of the matrix $\frac{\partial g}{\partial x}$. This result was obtained from the sensitivity analysis done for both estimation schemes in Section 5.6.

This simulation study shows how the dual estimation scheme outperforms simultaneous estimation in such a situation.

Run-of-mine ore milling circuits are generally difficult to control, partly because of unavailable process variable measurements. This study shows that accurate state and parameter estimation can be done for a grinding mill with the application of dual particle filters. These estimates may then be used in an advanced process control scheme to improve control performance.

CHAPTER 6 CONCLUSION

Run-of-mine ore milling circuits are generally difficult to control owing to strong external disturbances, poor process models and the unavailability of important process variable measurements (Hodouin, 2011). This work touched on each of these difficulties, firstly through the application of various DOBs to suppress the effects of external disturbances in Chapter 3. Secondly, MPM detection was implemented in Chapter 4 to determine whether the available process model adequately represents the actual plant. Thirdly, combined state and parameter estimation for the mill was implemented in Chapter 5. Simultaneous estimation was compared to dual estimation in a particle-filtering framework, and a sensitivity analysis also showed the class of systems for which dual estimation would provide superior results.

6.1 SUMMARY OF RESULTS

In Chapter 3 the results for the implementation of an integer order DOB, a fractional order DOB and a BICO DOB on the nonlinear milling circuit model were shown. The results indicated that all three DOB based control schemes produced improved controller performance (in an ISE sense) in the presence of external disturbances as well as for set-point tracking in the presence of MPM.

The order of the FO-DOB for optimal overall performance was close to the order of the integer order DOB, and consequently the results are rather similar. The added complexity of implementing a FO-DOB for this system might not be justified by the marginal improvement in performance, but this would not necessarily be the case in general. For systems where the optimal fractional order is around 0.5 (between two integer orders) the performance would be less similar to the integer order DOB, and the implementation of a FO-DOB would be warranted.

The BICO disturbance observer produced the best results for set-point tracking in the presence of mismatch. Again the performance improvement should be weighed against the added complexity of

implementation. Implementing a DOB is however not particularly complex in general, and could be a valuable addition to a control loop. It is also advantageous that the DOB is implementable in conjunction with the ubiquitous PI(D) controller, for which there is great potential of implementation, especially in the minerals processing industry.

Chapter 4 showed a closed-form expression for the mismatch between the actual plant and the available model. This expression is calculable from input-output data where some excitation is present, given that the controller has a transfer function. This once again has large scope for implementation, as PI(D) controllers are expressed with explicit transfer functions. Model-plant mismatch is common in ROM ore milling circuits and this expression lends itself to the possibility of updating the plant model on-line, based on the mismatch that may be present.

Model predictive controllers are however not generally expressible with explicit transfer functions from the error to the control signal. For a plant controlled by an MPC controller, a different method of MPM detection is needed. The method used here is based on a partial correlation analysis and does indicate the correct transfer function element(s) in the MIMO transfer function matrix that contains mismatch. The method models the dependency of manipulated variables on one another as well as of residuals on manipulated variables and subtracts these before doing the correlation analysis to eliminate spurious correlations. The results are rather sensitive to the models used in this step, as may be expected, but this reduces the robustness of the method.

Once the transfer function element, or elements, containing mismatch have been identified, partial process re-identification may be done. This is less time-consuming and less costly than full process re-identification, which would have been needed if the transfer function elements could not be identified.

In Chapter 5 the results for combined state and parameter estimation for the mill were presented. The simultaneous estimation scheme showed rather poor results, and the sensitivity analysis showed that this could be attributed to the poor conditioning of the $\frac{\partial g}{\partial x}$ term. Dual particle filters, which do not suffer deteriorated estimation accuracy if $\frac{\partial g}{\partial x}$ is ill-conditioned, were implemented and showed improved estimation accuracy. The results provided by this novel method are deemed sufficiently accurate for incorporation into an advanced control strategy.

Estimation results were also presented where the underlying augmented system is unobservable. Here the state estimates and some of the parameter estimates were still rather accurate. In order to increase

the estimation accuracy of the inaccurately estimated parameters, it is considered necessary to include more measurements. This estimation scheme does in a sense try to extract a lot of information from the process data by using the phenomenological model equations. It is expected to be difficult to extract more information from the same amount of data. At some point the estimation will become inaccurate and more data will be needed.

6.2 CONCLUDING REMARKS

The implemented peripheral control tools have aimed to address the three main sources of deteriorated control performance on a ROM ore milling circuit listed by Hodouin (2011), namely the presence of strong external disturbances, poor process models and the unavailability of important process variable measurements.

The implementation of several DOB variations was shown to deal well with external disturbances. A fractional order disturbance observer was implemented as a novel application. A new BICO disturbance observer was also introduced and the performance gained over the regular integer order DOB was explored.

The focus for MPM was aimed at detecting significant mismatch between the available model and the actual plant. An explicit expression was given for the mismatch in the case where the controller can be expressed with a transfer function. In the case where the controller does not have a transfer function, such as for MPC, a partial correlation analysis approach was followed to detect the mismatch.

The problem of the unavailability of process variable measurements was dealt with by implementing Bayesian estimation. Particle filtering was chosen for combined state and parameter estimation. A novel dual particle filtering approach was proposed, and the class of systems for which this framework is expected to provide superior results was derived.

Throughout this work some emphasis has been placed on the immediate impact these peripheral control tools can have on current milling circuits by implementing them in conjunction with the widely used PI(D) control topology. Some attention was also given to the impact that these tools may have in the future, where it is expected that more advanced control strategies such as MPC will be more commonly employed.

6.3 SUGGESTIONS FOR FURTHER WORK

Much work has gone into the development of adequate control strategies for ROM ore milling circuits. In this field, where the peripheral control tools are regarded to be as important as the controller itself, much more research effort into peripheral control tools is expected. This applies to the implementation of existing peripheral control tools to ROM ore milling circuits, and also to the development of new strategies to assist the controller.

The disturbance observers used in this work were applied in conjunction with PI controllers. Disturbance observers have been used on grinding mill circuits in conjunction with MPC (Yang et al., 2010) and it is suggested that there is more scope for contribution to this area. The FO-DOB and BICO-DOB topologies used in this work could still be applied with MPC to investigate their impact. Formal tuning strategies for both of these implementations are also lacking.

This work has addressed the detection of significant MPM. There is however still some scope for automatically updating the process model on-line once MPM has been detected. If the controller has a transfer function the closed-form MPM expression given in Chapter 4 can be used for this task, but the possible presence of unmeasured disturbances should still be addressed. The MPM detection strategy based on the partial correlation analysis gives the transfer function elements containing mismatch. It is at this stage still up to the control engineer to re-identify these models. The model re-identification could be automated, depending on the amount of excitation in the process data, which may also be tested automatically.

Dual particle filtering for combined state and parameter estimation was employed here. There is firstly some scope for further work in the particle-filtering theory, and secondly scope for expanding the implementation. A formal mathematical analysis to compare simultaneous estimation and dual estimation is needed. Specific reference is made here to the difference in how they handle the covariances in the data. Estimates produced by particle filters are also expected to converge to the true variable values, depending on the number of particles used and how close the initial estimate is to the true variable value. Expressions for these may be difficult to obtain, as particle filtering is a Monte Carlo method, but bounds on these quantities would be useful and are currently lacking.

The state estimation results would presumably be most valuable for use in an advanced control strategy where full-state feedback is required. Such an advanced control strategy was proposed by

Coetzee et al. (2010) where full-state feedback was assumed. The sump in this model also has three states and their values will be needed if the milling circuit is to be controlled by a strategy requiring full-state feedback. Once the values of the parameters that affect the mill are known, they can be included in the control strategy for more effective milling. Relationships should however be established for how the manipulated variable values should be changed, based on the known parameter values.

BIBLIOGRAPHY

Apelt, T. A. and Thornhill, N. F. Inferential measurement of SAG mill parameters IV: inferential model validation. *Minerals Engineering*, 22:1032 – 1044, 2009a.

Apelt, T. A. and Thornhill, N. F. Inferential measurement of SAG mill parameters V: MPC simulation. *Minerals Engineering*, 22:1045 – 1052, 2009b.

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

Apelt, T. A., Asprey, S. P., and Thornhill, N. F. Inferential measurement of SAG mill parameters II: state estimation. *Minerals Engineering*, 15:1043 – 1053, 2002a.

Apelt, T. A., Asprey, S. P., and Thornhill, N. F. Inferential measurement of SAG mill parameters III: inferential models. *Minerals Engineering*, 15:1055 – 1071, 2002b.

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

Badwe, A. S., Gudi, R. D., Patwardhan, R. S., Shah, S. L., and Patwardhan, S. C. Detection of model-plant mismatch in MPC applications. *Journal of Process Control*, 19:1305–1313, 2009.

Bauer, M. and Craig, I. K. Economic assessment of advanced process control - A survey and framework. *Journal of Process Control*, 18:2 – 18, 2008.

Bemporad, A., Morari, M., and Ricker, N. L. *Model Predictive Control Toolbox for Matlab – User's Guide*. The Mathworks Inc., http://www.mathworks.com/access/helpdesk /help/toolbox/mpc/, 2004.

Bergman, N. *Recursive Bayesian estimation: Navigation and tracking applications*. PhD thesis, Linkoping University, 1999.

Bode, H. W. *Network Analysis and Feedback Amplifier Design*. Van Nostrand, New York, 1945.

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

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

Chen, X. S., Yang, J., Li, S. H., and Li, Q. Disturbance observer based multivariable control of ball mill grinding circuits. *Journal of Process Control*, 19:1205 – 1213, 2009.

Chen, Y. Q., Ooi, K. K., Ding, M. Z., Tang, L. L., and Soh, K. T. An efficient sensorless rotational vibration and shock compensator (RVSC) for hard disk drives with higher TPI. *US PTO Published Patent Applications*, page US20010036026, 2001.

Chen, Y. Q., Vinagre, B. M., and Podlubny, I. Fractional order disturbance observer for vibration suppression. *Nonlinear Dynamics*, 38(1–4):355–367, 2004.

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

Coetzee, L. C., Craig, I. K., and Kerrigan, E. C. Robust nonlinear model predictive control of a run-of-mine ore milling circuit. *IEEE Trans. Control Syst. Technol.*, 18:222 – 229, 2010.

Cohn, S. E. and Dee, D. P. Observability of discretized partial differential equations. *SIAM Journal on Numerical Analysis*, 25:586 – 617, 1988.

Conner, J. S. and Seborg, D. E. Assessing the need for process re-identification. *Industrial and Engineering Chemistry Research*, 44:2767 – 2775, 2005.

Conradie, A. V. E. and Aldrich, C. Neurocontrol of a ball mill grinding circuit using evolutionary reinforcement learning. *Minerals Engineering*, 14:1277 – 1294, 2001.

Craig, I. K. and MacLeod, I. M. Specification framework for robust control of a run-of-mine ore

milling circuit. *Control Eng. Practice*, 3:621 – 630, 1995.

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

Craig, I. K., Hulbert, D. G., Metzner, G., and Moult, S. P. Optimized multivariable control of an industrial run-of-mine circuit. *Journal of the South African Institute of Mining and Metallurgy*, 92: 169 – 176, 1992.

Datta, B. N. *Numerical Linear Algebra and Applications*. Pacific Grove: Brooks/Cole, 1994.

Diop, S. and Fliess, M. Nonlinear observability, identifiability, and persistent trajectories. In *Proc. 30th IEEE conference on Decision and Control*, volume 1, pages 714 – 719, Brighton, UK, December 1991.

Doucet, A., De Freitas, N., and Gordon, N. *Sequential Monte-Carlo Methods in Practice*. New York: Springer, 2001.

Douchet, A., Godsill, S., and Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10:197 – 208, 2000.

Duarte, M., Suarez, A., and Bassi, D. Multivariable predictive neuronal control applied to grinding plants. In *Proc. of the 2nd Int. Conference on Intelligent Processing and Manufacturing of Materials, 1999. IPMM '99*, pages 975 – 982, Honolulu, HI, USA, 1999.

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

Franklin, G. F., Powell, J. D., and Workman, M. L. *Digital Control of Dynamic Systems*. Addison-Wesley, Reading, MA, 1990.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F Radar and Signal processing*, 140:107 – 113, 1993.

Harris, T. J., Seppala, C. T., and Desborough, L. D. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *Journal of Process Control*, 9:1 – 17, 1999.

Henson, M. A. Nonlinear model predictive control: current status and future directions. *Computers and Chemical engineering*, 23:187 – 202, 1998.

Herbst, J. A. and Pate, W. T. Object components for comminution system softsensor design. *Powder Technology*, 105:424 – 429, 1999.

Ho, Y. C. and Lee, R. C. K. A Bayesian approach to problems in stochastic estimation and control. *IEEE Transactions on Automatic control*, 9:333 – 339, 1964.

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

Huang, B. Bayesian methods for control loop monitoring and diagnosis. *Journal of Process Control*, 18:829 – 838, 2008.

Huang, B. Dynamic modeling, identification and predictive control with applications in solid oxide fuel cells. Personal communication, 2011.

Huang, B. and Shah, S. L. *Performance Assessment of Control Loops: Theory and Applications*. London: Springer, 1999.

Huang, B., Malhotra, A., and Tamayo, E. C. Model predictive control relevant identification and validation. *Chemical Engineering Science*, 58:2389 – 2401, 2003.

Hulbert, D. G., Craig, I. K., Coetzee, M. L., and Tudor, D. Multivariable control of a run-of-mine milling circuit. *Journal of the South African Institute of Mining and Metallurgy*, 90:173 – 181, 1990.

Jayasankar, B. R., Ben-Zvi, A., and Huang, B. Identifiability and estimability study for a dynamic solid oxide fuel cell model. *Computers and Chemical Engineering*, 33:484 – 492, 2009.

Jazwinski, A. *Stochastic processes and filtering theory*. Academic, New York, 1970.

Julier, S. J. The scaled unscented transformation. In *Proc. of the American control conference*, pages 4555 – 4559, 2002.

Julier, S. J. and Uhlmann, J. K. A new extension of the Kalman filter to nonlinear systems. In *Proc. of*

AeroSense: The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Controls, volume 3068, pages 182 – 193, 1997.

Juloski, A. L., Heemels, W. P. M. H., Boers, Y., and Verschure, F. Two approaches to state estimation for a class of piecewise affine systems. In *Proc. 42nd IEEE Conference of Decision and Control*, pages 143 – 148, Maui, Hawaii, USA, 2003.

Kalman, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35 – 45, 1960.

Kano, M., Shigi, Y., Hasebe, S., and Ooyama, S. Detection of significant model-plant mismatch from routine operation data of model predictive control system. In *Proc. of 9th International Symposium on Dynamics and Control of Process Systems (DYCOPS 2010) Leuven, Belgium, 2010*, pages 677 – 682, Leuven, Belgium, 2010.

Kempf, C. J. and Kobayashi, S. Disturbance observer and feedforward design for a high-speed direct-drive positioning table. *IEEE Trans. Control Syst. Technol.*, 7:513 – 526, 1999.

Kitagawa, G. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5:97 – 112, 1996.

Li, W. and Hori, Y. Vibration suppression using single neuron-based PI fuzzy controller and fractional-order disturbance observer. *IEEE Trans. Ind. Electron.*, 54(1):117–126, 2007.

Ljung, L. *System Identification – Theory for the User*. Upper Saddle River, NJ: Prentice Hall, 2nd edition, 1999.

Maerz, N. H., Palangio, T. C., and Franklin, J. A. Wipfrag image based granulometry system. In *Proc. of the FRAGBLAST 5 Workshop on Measurement of Blast Fragmentation*, pages 91 – 99, Montreal, Quebec, Canada, 1996.

Marsden, J. O. and House, C. I. *The Chemistry of Gold Extraction*. New York : Horwood, 1992.

Monje, C. A., Chen, Y. Q., Vinagre, B., Xue, D., and Feliu, V. *Fractional Order Systems and Controls - Fundamentals and Applications*. Advances in Industrial Control Series, Springer-Verlag, London, ISBN: 978-1-84996-334-3, 2010.

Morrison, R. D. and Richardson, J. M. *Mineral processing plant design, practice, and control proceedings*, chapter JKSimMet: A Simulator for analysis, optimisation and design of comminution circuits. Colorado: Society for Mining, Metallurgy, and Exploration, Inc. (SME), 2002.

Nageswararao, K., Wiseman, D. M., and Napier-Munn, T. J. Two empirical hydrocyclone models revisited. *Minerals Engineering*, 17:671 – 687, 2004.

Narasimhan, S. and Jordache, C. *Data reconciliation and gross error detection: an intelligent use of process data*. Houston: Gulf Pub. Co., 2000.

Olivier, L. E. and Craig, I. K. Parameter mismatch detection in a run-of-mine ore milling circuit under model predictive control. In *IFAC World Congress 2011*, pages 9929–9934, Milan, Italy, 2011.

Olivier, L. E., Craig, I. K., and Chen, Y. Q. Fractional order disturbance observer for a run-of-mine ore milling circuit. In *IEEE Africon 2011*, Zambia, 2011a.

Olivier, L. E., Craig, I. K., and Chen, Y. Q. Fractional order and BICO disturbance observers for a run-of-mine ore milling circuit. *Journal of Process Control*, 2011b.

Patwardhan, R. S. and Shah, S. L. Issues in performance diagnostics of model-based controllers. *Journal of Process Control*, 12:413 – 427, 2002.

Qin, S. J. Control performance monitoring – review and assessment. *Computers and Chemical Engineering*, 23:173 – 186, 1998.

Qin, S. J. and Bagwell, T. A. A survey of industrial model predictive control technology. *Control Eng. Practice*, 11:733 – 764, 2003.

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

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

Reif, K., Günther, S., Yaz, E., and Unbehauen, R. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic control*, 44:714 – 728, 1999.

Ristic, B., Arulampalam, S., and Gordon, N. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, Boston, 2004.

Seborg, D. E., Edgar, T. F., and Mellichamp, D. A. *Process Dynamics and Control*. Hoboken, NJ: Wiley, 2nd edition, 2003.

Selvanathan, S. and Tangirala, A. K. Diagnosis of poor control loop performance due to model-plant mismatch. *Industrial and Engineering Chemistry Research*, 49:4210 – 4229, 2010.

Shi, F. N. and Napier-Munn, T. J. Effects of slurry rheology on industrial grinding performance. *Int. J. Mineral Process.*, 65:125 – 140, 2002.

Skogestad, S. and Postlethwaite, I. *Multivariable feedback control : analysis and design*. Chichester, England: Wiley, 2nd edition, 2005.

Stanley, G. G. *The extractive metallurgy of gold in South Africa*. Johannesburg : South African Institute of Mining and Metallurgy, 1987.

Tessier, J., Duchesne, C., and Bartolacci, G. A machine vision approach to on-line estimation of run-of-mine ore composition on conveyor belts. *Minerals Engineering*, 20:1129 – 1144, 2007.

Turing, A. M. Rounding-off errors in matrix processes. *Quart. J. Mech. Appl. Math.*, 1:287 – 308, 1948.

Valenzuela, J., Bourassa, M., Najim, K., and Del Villar, R. Dynamic matrix control of an autogenous grinding circuit. *Minerals Engineering*, 7:105 – 114, 1994.

Van der Merwe, R. *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*. PhD thesis, Oregon Health & Science University, 2004.

Van Nierop, M. A. and Moys, M. H. Exploration of mill power modelled as function of load behaviour. *Minerals Engineering*, 14:1267 – 1276, 2001.

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

Wei, D. *Development of performance functions for economic performance assessment of process control systems*. PhD thesis, University of Pretoria, 2010.

Wei, D. and Craig, I. K. Grinding mill circuits - a survey of control and economic concerns. *Int. J. Miner. Process.*, 90:56 – 66, 2009.

Yang, J., Li, S., Chen, X., and Li, Q. Disturbance rejection of ball mill grinding circuits using DOB and MPC. *Powder Technology*, 198:219 – 228, 2010.

Zhu, H. D., Zhang, G. H., and Shao, H. H. Control of the process with inverse response and dead-time based on disturbance observer. In *Proc. of the American Control Conference*, pages 4826 – 4831, Portland, OR, USA, 2005.

LIST OF FIGURES

LIST OF TABLES

“If we knew what it was we were doing, it would not be called research, would it?”

– Albert Einstein