Estimation and control of froth flotation units used in Coal Handling and Preparation Plants (CHPPs)

by

Kevin Arulmaran

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Chemical Engineering

Department of Chemical and Materials Engineering

University of Alberta

Abstract

The processing of coal in coal handling and preparation plants (CHPPs) produces a significant amount of valuable fine coal which is then recovered using froth flotation. As froth flotation units are regulated using simple, conventional control methods the application of modern control methods is one option to improve the efficiency and profitability of these units. There are various issues to consider before modern control methods can be applied, namely: state estimation, handling model plant mismatch (MPM) and handling measurement delays. This thesis compares the performance of various modern control methods in the nominal case (no MPM and no measurement delays) and in the case of MPM with no measurement delays with the aim of identifying the best control method for each scenario. State estimation methods are also tested in the nominal case, with MPM and with measurement delays to identify the best one for each scenario.

For state estimation, three methods are investigated: an extended Kalman filter (EKF), moving horizon estimator (MHE) and nonlinear observer. The EKF is found to give the best estimation performance in the nominal case. Although MHE has the potential to give better estimates, it does not do so due to the small estimation window. The estimation window needs to be small to keep computation time reasonable as there are a lot of system states. The nonlinear observer performs the worst because it does not use noise information. All three methods have similar performance when there is MPM as the MPM masks noise information. All three methods also have similar performance when measurements are delayed as the large measurements delays in the froth flotation system mean that state prediction dominates over state estimation. The EKF is picked as the best estimator for this system on the basis of its estimation performance and computation time.

A multiple model (MM) based approach is proposed to obtain unbiased state es-

timates in the presence of MPM caused by parameter mismatch. For a deterministic linear system operating at steady state, it is proved that unbiased output estimates guarantee unbiased states estimates provided certain conditions are met. This approach is successfully applied to the froth flotation system.

A model predictive controller (MPC) and economic model predictive controller (EMPC) are tested for the nominal case of no MPM and accurate state estimates available. Both are found to stabilize the system and give similar economic performance. For the case of offset free control in the presence of parameter mismatch, we present two methods: an offset free MPC using augmented models and a model identification based method which combines the MM state estimation method we propose with a conventional MPC. The offset free MPC fails to achieve offset free control when using an output disturbance model. Although the model ID based approach achieves offset free control there is no guarantee that it will work in the general case.

Acknowledgements

First and foremost I wish to thank my supervisor Dr. Jinfeng Liu without whose guidance this thesis would never have been completed. It was a pleasure to work alongside my colleagues in the CPC group: Su Liu, Jannatan Nahar, Jayson McAllister, Xunyuan Yin, Tianrui An, Mohammad Rashedi, An Zhang and others. I am grateful to the Honda-McNeil family (Jamie, Hiromi and Makoto) for including me in Sunday night family dinners. Finally I would like to thank my parents and brother on whose love and support I can always depend.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

The processing of coal in coal handling and preparation plants (CHPPs) produces a significant amount of valuable fine coal which is then recovered using froth flotation (Canright *et al.*, 1981). Typically, froth flotation units are regulated using simple, conventional control methods like base level control which contributes to the inefficiency of the froth flotation operation (Shean and Cilliers, 2011). Improving the efficiency of these units would improve the profitability of the entire plant. One way to improve froth flotation efficiency is to implement modern control methods such as model predictive control (Mayne *et al.*, 2000; Rawlings, 2000) or economic model predictive control (Rawlings *et al.*, 2012; Liu *et al.*, 2015).

These modern control methods typically require knowledge of all the system states. As it is in general difficult or expensive to measure all the system states, state estimation techniques can be used to reconstruct the entire system states using a model of the process and measurements of a few output variables. There are many options for state estimation with each option having advantages and disadvantages.

Inspired by the above, this thesis focuses on testing multiple state estimation methods and control strategies for the froth flotation system. Estimation performance will be tested under various scenarios: normal operation, delayed or missing measurements and with model plant mismatch (MPM). Control strategies will be compared during normal operation with all the system states directly available. Combined estimation and control performance in the presence of MPM caused by parameter mismatch will also be investigated.

1.2 Thesis outline and contributions

A description of the froth flotation system is provided in Chapter 2 along with a first principles system model. The design of a sensor network that uses the minimum number of sensors to guarantee observability of the entire system is covered. Three state estimators (extended Kalman filter, moving horizon estimator and nonlinear observer) are designed and the performance of these estimators is compared under normal conditions, with delayed or missing measurements and in the presence of MPM. It is shown that state estimates are biased when there is MPM.

In Chapter 3, the issue of obtaining unbiased state estimates in the presence of MPM caused by parameter mismatch is investigated. An algorithm is presented which can be used to obtain unbiased state estimates if certain conditions are met. A flowchart of the algorithm is provided and a system consisting of two continuous stirred tank reactors (CSTRs) is used to illustrate each stage of the algorithm. Finally, the algorithm is applied to the froth flotation system and is shown to successfully provide unbiased state estimates.

Chapter 4 covers the application of different controllers to the froth flotation system. A model predictive controller (MPC) and economic model predictive controller (EMPC) are tested on the system when there is no MPM and with all the states directly available to the controllers. The issue of offset free control in the presence of parameter mismatch is also investigated and two solutions are proposed: offset free nonlinear model predictive control (Morari and Maeder, 2012) and a model identification based approach which combines the state estimation method introduced in Chapter 3 with a conventional MPC.

Finally, Chapter 5 provides a summary of the thesis results and looks at directions for future work.

Chapter 2

State estimation of the froth flotation system

2.1 Introduction

Commonly used state estimation methods for nonlinear systems include the extended Kalman filter (EKF) (Einicke, 2012), the moving horizon estimator (MHE) (Robertson *et al.*, 1996) and nonlinear observers (Ciccarella *et al.*, 1993; Gauthier *et al.*, 1992). The EKF is an extension of the Kalman filter to nonlinear systems based on successive linearizations of the nonlinear system. Although it is not an optimal filter the EKF is still widely used due to its simplicity. There are variations of the EKF that address deficiencies of the EKF but we only investigates the standard EKF. The MHE is an optimization based state estimator that can handle nonlinear systems and system constraints (unlike the EKF which uses linear approximations of the system and ignores constraints). As a result, the MHE can have better estimation performance than the EKF but also has a greater computational burden. Nonlinear observers are designed for deterministic nonlinear systems. They are capable of handling nonlinear systems explicitly (like the MHE) but do not have the computational burden associated with the MHE. The main drawbacks to nonlinear observers are that they ignore noise information and can be difficult to design.

In this chapter, we consider the application of these three state estimation methods to a typical froth flotation process. We first introduce a process model developed at Oak Ridge National Laboratory (Canright *et al.*, 1981) for the coal froth flotation process. Based on currently available instrumentation used in froth flotation, we

then carry out sensor network design and determine a minimum set of sensors that provides observability of the entire froth flotation process. Subsequently, we design the EKF, the MHE and a nonlinear observer for the froth flotation process. The nonlinear observer is designed via an overlapping subsystem decomposition approach. Two different scenarios are considered: estimation without measurement delays and estimation with measurement delays. The performances of the three state estimation methods are compared with and without MPM. It is found in both scenarios that the EKF is better in terms of estimation precision while the nonlinear observer has the smallest computational burden.

2.2 Process description and modeling

Figure 2.1 shows a typical froth flotation unit consisting of 5 tanks in series and Figure 2.2 shows the cross section of an individual tank. Feed slurry consisting of coal, ash (non-coal solids) and water enters the first tank. The maximum particle size in the feed is controlled and the solid (coal and ash) content of the feed ranges from 0 to 30 wt%. Reagents (frother and collector) are added either at the inlet or upstream of the first tank (Canright *et al.*, 1981) with the optimal reagent concentration dependent on the solid concentration and coal type. The frother promotes the formation of froth while the collector promotes the preferential attachment of air bubbles to coal particles which then carry the coal particles to the top of the tank (the froth). The froth is removed using paddles and sent to a disk filter system for drying. The underflow (also known as tailings) from each tank is sent to the next tank in the series and the tailings from the final tank are sent to a static thickener.

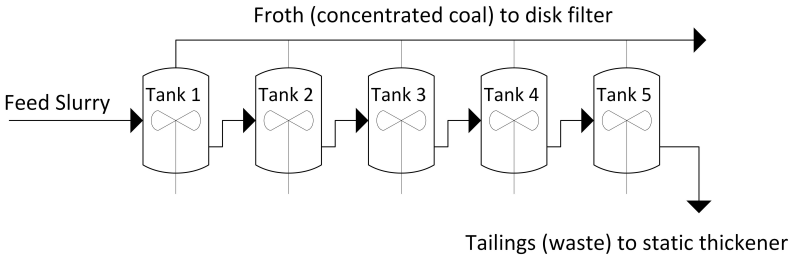

Figure 2.1: Schematic diagram of a froth flotation unit. Adapted from Canright *et al.*, 1981.

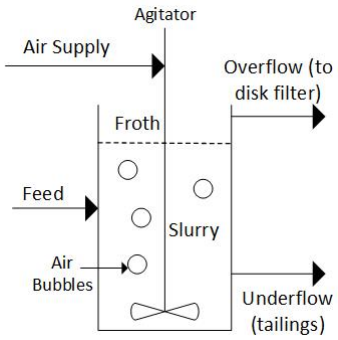

Figure 2.2: Schematic diagram of an individual flotation tank.

A first order kinetic model is used to model the flotation process with each tank modeled as a continuous stirred tank reactor (CSTR). The following assumptions are made in this model (Canright *et al.*, 1981):

1. The contents of each tank are well mixed, i.e, no concentration gradients within a tank.

2. There is no intermixing between tanks.

3. The slurry level in each tank does not vary.

The dynamics of the process can be characterized by the dynamics of the solids (coal and ash), liquids (mainly water) and ash (non-coal solids). Mass balances are completed for the 5 tanks for a total of 15 states. The equations describing the dynamics of tank i, where $i = 1, \ldots, 5$, are as follows (Canright *et al.*, 1981):

$$\frac{dc_{st_i}}{dt} = \frac{\dot{V}_{uf_{i-1}}}{V_i}(c_{st_{i-1}}) - \frac{\dot{V}_{uf_i}}{V_i}(c_{st_i}) - r_i \tag{2.1a}$$

$$\frac{dc_{lt_i}}{dt} = \frac{\dot{V}_{uf_{i-1}}}{V_i}(c_{lt_{i-1}}) - \frac{\dot{V}_{uf_i}}{V_i}(c_{lt_i}) - \frac{\beta_i}{V_i} \tag{2.1b}$$

$$\frac{dc_{at_i}}{dt} = \frac{\dot{V}_{uf_{i-1}}}{V_i}(c_{at_{i-1}}) - \frac{\dot{V}_{uf_i}}{V_i}(c_{at_i}) - \frac{\dot{A}_i}{V_i} \tag{2.1c}$$

where the subscript $i = 1, \ldots, 5$ indicates the tank number, c_{st_i} is the solids concentration (kg/m^3), c_{lt_i} is the liquids concentration (kg/m^3), c_{at_i} is the ash concentration (kg/m^3), $\dot{V}_{uf_i}$ is the volumetric rate of the underflow (m^3/min) and V_i is the slurry volume (m^3). r_i is the rate of solid removal as defined below, β_i is the mass flow rate of liquid to overflow (kg/min) and $\dot{A}_i$ is the mass flow rate of ash to overflow

(kg/min). c_{st_0}, c_{lt_0}, c_{at0} are the solids, liquids and ash concentration of the feed respectively. $\dot{V}_{uf_0}$ is the volumetric flow rate of the feed. On the right hand side of the above model, the first term of each equation represents the amount entering from the previous tank, the second term represents the amount leaving in the underflow and the third term represents the amount leaving in the overflow.

The rate of solid removal r_i is given by (Canright *et al.*, 1981):

$$r_i = f_r k(c_{st_i} - c_\infty) \tag{2.2}$$

where r_i is the flotation rate (kg m^{-3} min^{-1}), k is the rate constant (l/min), c_∞ is the equilibrium solids concentration (kg/m^3) and f_r is the correction factor for industrial scale reactions (as k and c_∞ values are obtained from lab experiments). The values of k and c_∞ depend on frother loading (g/kg dry feed), collector loading (g/kg dry feed) and coal type.

The following equations are also defined for each tank i, where $i = 1, \ldots, 5$ (Canright *et al.*, 1981):

$$\dot{V}_{uf_{i-1}} = \dot{V}_{uf_i} + \dot{V}_{of_i} \tag{2.3}$$

$$\dot{V}_{of_i} = \frac{\beta}{\rho_l} + \frac{r_i V_i}{\rho_c} \tag{2.4}$$

$$\dot{A}_i = x_{A_i} \left(\sum_{j=1}^{i} \dot{M}_{sof_j} \right) - \sum_{j=1}^{i-1} \dot{A}_j \tag{2.5}$$

$$R_i = \left(\frac{c_{sof_i} \dot{V}_{of_i}}{c_{sof_i} \dot{V}_{of_i} + c_{st_i} \dot{V}_{uf_i}} \right) (100 - R_{i-1}) + R_{i-1} \tag{2.6}$$

$$x_{A_i} = g(R_i) \tag{2.7}$$

where ρ_c is the density of coal (kg/m^3), ρ_l is the density of liquid (kg/m^3), R_i is the cumulative solid recovery (%) at tank i, x_{A_i} is the cumulative mass fraction of ash in the overflow solids at tank i, $\dot{M}_{sof_i}$ is the mass flowrate of solids in the overflow (kg/min), $\dot{V}_{of_i}$ is the volumetric flowrate of the overflow (m^3) and c_{sof_i} is the concentration of solids in the overflow (kg/m^3). $g(R_i)$ is an empirical function of R_i for a given frother and collector loading obtained from Canright *et al.*, 1981.

The process model can be written in the following compact form:

$$\dot{x}(t) = f(x(t), u(t), w(t)) \tag{2.8}$$

where the state vector is $x = \left[x_1^T,\ x_2^T,\ x_3^T,\ x_4^T,\ x_5^T\right]^T$ with $x_i^T = [c_{st_i}, c_{lt_i}, c_{at_i}]$ for $i = 1, \ldots, 5$ and w denotes random process noise. u denotes the inputs with $u = [u_1, u_2]^T$ where u_1 is the frother loading and u_2 is the collector loading.

2.3 Sensor network design

The success of state estimation depends on the available measurements. The purpose of this section is to determine which states can be measured using currently available sensors and to determine the smallest set of measurements which will allow all the system states to be estimated reliably.

2.3.1 Measurements

The following measurements are available for the froth flotation process:

1. Solid/liquid concentration in each tank. Slurry density can be measured using X-ray fluorescence (XRF) analyzers or gamma density gauges. This information can then be used to calculate the solid and liquid concentration. XRF dynamics are in the order of 10-20 minutes but provide high accuracy (between 1% and 6%) (Shean and Cilliers, 2011). Gamma density gauges are non-intrusive but sensitive to location as air bubbles affect measurements (Shean and Cilliers, 2011).

2. Ash concentration in each tank. XRF can be used to obtain elemental assay data which can be used to find ash concentration. Machine vision methods can also be used to find ash concentration. Machine vision is faster than XRF with dynamics in the order of 1 min versus 10-20 minutes for XRF (Shean and Cilliers, 2011) but is less accurate with up to 30% error at low ash concentrations (Hargrave *et al.*, 1996; Zhang *et al.*, 2014).

While it is possible to use sensors to measure all 15 states directly, it is worthwhile to determine if a smaller number of sensors will result in an observable system as sensors are costly to install and replace.

2.3.2 Observability testing

A system is said to be observable if, for a time interval $[t_0, t_1]$, given the input and output measurements over the interval it is possible to solve for the initial state $x(t_0)$ (Bay, 1998). Using this information it is then possible to solve for $x(t)$ over the entire interval. We assume that the measurements from the sensors can be modeled using the following equation:

$$y_i(t) = h_i(x(t)) + v_i(t) \tag{2.9}$$

where y_i is the sensor measurement with corresponding measurement noise v_i for $1 \leq i \leq s$ with s being the number of measurements.

For nonlinear systems, the observability test is to check if (Marino and Tomei, 1995):

$$\text{rank}\{O\} = n$$

with

$$O = \begin{bmatrix} dh(x) \\ \vdots \\ d(L_f^{n-1}h(x)) \end{bmatrix} \tag{2.10}$$

where $h(x(t)) = [h_1(x), \ldots, h_s(x)]^T$, n is the size of the system and $L_f^k h$ $(1 \leq k \leq n-1)$ is the k-th Lie derivative of h with respect to f.

This observability test requires the calculation of high order Lie derivatives and their differentials which in general is a difficult task. As a result, a different approach has to be used to test observability. The approach we used is to test the observability of linear approximations of the nonlinear system at different points along a typical state trajectory. The linear approximation of the nonlinear system at a point $x(t)$ will have the form (assuming zero noise without loss of generality):

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{2.11a}$$

$$y(t) = C(t)x(t) \tag{2.11b}$$

where $A(t)$ and $C(t)$ are found by taking the Jacobian of (2.8) and (2.9) respectively at $(x(t), u(t))$ as shown below:

$$A(t) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{x=x(t),u=u(t)} \qquad C(t) = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_s}{\partial x_1} & \cdots & \frac{\partial h_s}{\partial x_n} \end{bmatrix}_{x=x(t)} \tag{2.12}$$

The observability of the linear approximation (2.11) at $(x(t), u(t))$ is tested by checking if (Bay, 1998):

$$\text{rank}\left\{\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}\right\} = n \tag{2.13}$$

If the linear approximation is observable at all the different points along the trajectory, we claim that the nonlinear system is observable in a neighborhood of the trajectory. Using the above approach, the froth flotation process is found to be observable using only two measurements, c_{lt_5} (x_{14}) and c_{at_5} (x_{15}), resulting in the following time invariant (constant) C matrix:

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.14}$$

Kalman decomposition can be carried out according to (Bay, 1998) to determine the set of observable states associated with each measurement. It is found that measuring c_{lt_5} allows observability of c_{st_i} and c_{lt_i} for all $i = 1, \ldots, 5$ (no ash states observable) and measuring c_{at_5} allows observability of c_{st_i} and c_{at_i} for all $i = 1, \ldots, 5$ (no liquid states observable).

2.4 State estimation without measurement delays

In this section, we consider state estimation of the froth flotation process without measurement delays. Three typical state estimation methods are considered: (1) an extended Kalman filter (EKF), (2) moving horizon estimator (MHE) and (3) a nonlinear observer.

2.4.1 Modeling of measurements

We assume that c_{lt5} and c_{at5} measurements are available at discrete sampling times $t_k = t_0 + k\Delta$ where $t_0 = 0$ is the initial time, k denotes non-negative integers and Δ is a fixed sampling time. The measurements at t_k are modelled as follows:

$$y(t_k) = Cx(t_k) + v(t_k) \tag{2.15}$$

2.4.2 Design of the EKF

EKFs are discrete time filters for nonlinear systems based on successive linearizations of the nonlinear system (Chui and Chen, 1991). There are two steps in an EKF: a prediction step and an update step. The EKF used in this work is designed as follows:

1. Prediction step:

$$\hat{x}(t_k|t_{k-1}) = \hat{x}(t_{k-1}) + \int_{t_{k-1}}^{t_k} f(\hat{x}(t), u(t), 0)dt \tag{2.16a}$$

$$P(t_k|t_{k-1}) = A(t_{k-1})P(t_{k-1})A(t_{k-1})^T + Q \tag{2.16b}$$

   where $\hat{x}(t_k|t_{k-1})$ denotes the prediction of the state at time t_k based on the state estimate $\hat{x}(t_{k-1})$ at t_{k-1}, $P(t_{k-1})$ is the error covariance matrix of $\hat{x}(t_{k-1})$, $P(t_k|t_{k-1})$ is the error covariance matrix of $\hat{x}(t_k|t_{k-1})$ and Q is the covariance of the process noise. $A(t_{k-1})$ is the discretized Jacobian of f with respect to x at t_{k-1}. $u(t)$ is the known, piecewise constant input.

2. Update step:

$$K(t_k) = P(t_k|t_{k-1})C^T[CP(t_k|t_{k-1})C^T + R]^{-1} \tag{2.17a}$$

$$\hat{x}(t_k) = \hat{x}(t_k|t_{k-1}) + K(t_k)(y(t_k) - C\hat{x}(t_k|t_{k-1})) \tag{2.17b}$$

$$P(t_k) = (I - K(t_k)C)P(t_k|t_{k-1}) \tag{2.17c}$$

   where R is the covariance of the measurement noise and $K(t_k)$ is the filter gain at t_k (Chui and Chen, 1991).

2.4.3 Design of the MHE

The moving horizon estimator (MHE) design for the system represented by (2.8) and (2.9) has the following form at time t_k (Rao and Rawlings, 2002):

$$\begin{aligned}\min_{\hat{x}(t_{k-N+1}),w(t_{k-N+1}),\ldots,w(t_{k-1})} \Gamma(\hat{x}(t_{k-N+1})) + \sum_{i=k-N+1}^{k-1} w(t_i)^T Q^{-1} w(t_i) + \ldots \\ \cdots \sum_{i=k-N+1}^{k} v(t_i)^T R^{-1} v(t_i)\end{aligned} \tag{2.18a}$$

$$\text{s.t} \qquad \dot{\hat{x}}(t) = f(\hat{x}(t), u(t), w(t)) \tag{2.18b}$$

$$y(t) = C\hat{x}(t) + v(t) \tag{2.18c}$$

$$w(t) \in \mathbb{W}, v(t) \in \mathbb{V}, \hat{x}(t) \in \mathbb{X} \tag{2.18d}$$

where N is the size of the estimation window and $\hat{x}(t_{k-N+1})$ is the MHE estimate of the system state obtained at t_{k-N+1}. Q and R are the covariance matrices of the process noise (w) and the measurement noise (v) respectively. u is the known, piecewise constant input. w and v are assumed to be piecewise constant during a sampling period Δ. $\mathbb{W}$, $\mathbb{V}$, and $\mathbb{X}$ represent the set of all possible values of w, v and x respectively. The MHE attempts to minimize the cost function (2.18a) by optimizing $\hat{x}(t_{k-N+1}), w(t_{k-N+1}), \ldots, w(t_{k-1})$ which is equivalent to optimizing $\hat{x}(t_{k-N+1}), \ldots, \hat{x}(t_k)$ due to the piecewise constant noise assumption.

The $\Gamma(\hat{x}(t_{k-N+1}))$ term is known as the arrival cost and summarizes past information that is not included in the estimation window, i.e. information up to t_{k-N}. The arrival cost has a significant impact on the performance of the MHE but is difficult to determine exactly for constrained nonlinear systems. While the arrival cost can be omitted if a sufficiently large N is used, this increases the computational burden and reduces the speed of the MHE. There are a number of methods to approximate the arrival cost with one such method based on an EKF as follows:

$$\Gamma(z) = (z - \hat{x}(t_{k-N+1}))^T P^{-1}(t_{k-N+1}|t_{k-N})(z - \hat{x}(t_{k-N+1})) \tag{2.19}$$

where z is the MHE estimate of $x(t_{k-N+1})$ at the current time t_k and $\hat{x}(t_{k-N+1})$ is the MHE estimate of $x(t_{k-N+1})$ obtained at t_{k-N+1}. $P(t_{k-N+1}|t_{k-N})$ is obtained from an EKF that is running parallel to the MHE.

2.4.4 Design of the nonlinear observer

Observer design for nonlinear systems is in general challenging. Most approaches either require Lyanpunov functions that are difficult to find, do not guarantee convergence to the actual states or are focused on single input single output (SISO) systems (Ciccarella *et al.*, 1993). There are only a few methods that are extensible to multi input multi output (MIMO) systems (Ciccarella *et al.*, 1993). In this work, the nonlinear observer is designed based on a subsystem decomposition approach in-

spired by results on distributed state estimation (Zeng and Liu, 2015; Zhang and Liu, 2014; Zhang and Liu, 2013).

The system is split into two overlapping subsystems according to the observable states associated with each of the two measurements. The states of the two subsystems are $x_{s1}(t) = [c_{st_1}(t), c_{lt_1}(t), c_{st_2}(t), c_{lt_2}(t), ..., c_{st_5}(t), c_{lt_5}(t)]^T$ and $x_{s2}(t) = [c_{st_1}(t), c_{at_1}(t), c_{st_2}(t), c_{at_2}(t), ..., c_{st_5}(t), c_{at_5}(t)]^T$ respectively. The two subsystems have the following forms:

$$\dot{x}_{s1}(t) = f_{s1}(x_{s1}(t), u(t), w_{s1}(t)) \tag{2.20a}$$

$$y_{s1}(t) = C_{s1}x_{s1}(t) + v_{s1}(t) \tag{2.20b}$$

$$\dot{x}_{s2}(t) = f_{s2}(x_{s2}(t), u(t), w_{s2}(t)) \tag{2.20c}$$

$$y_{s2}(t) = C_{s2}x_{s2}(t) + v_{s2}(t) \tag{2.20d}$$

where w_{s1} and w_{s2} are the process noise and v_{s1} and v_{s2} are the measurement noise in their respective subsystems. C_{s1} and C_{s2} are constant matrices such that:

$$C_{s1} = C_{s2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.21}$$

The observer for a given subsystem is designed following Ciccarella *et al.*, 1993. The nonlinear observers for the two subsystems are as follows::

$$\begin{array}{rcl} \dot{\hat{x}}_{s1}(t) & = & f_{s1}(\hat{x}_{s1}, u(t), 0) + \left(O_{s1}(\hat{x}_{s1}(t), u(t))\right)^{-1} K_{s1}(y_{s1}(t) - C_{s1}\hat{x}_{s1}(t)) \\ \dot{\hat{x}}_{s2}(t) & = & f_{s1}(\hat{x}_{s2}, u(t), 0) + \left(O_{s2}(\hat{x}_{s2}(t), u(t))\right)^{-1} K_{s2}(y_{s2}(t) - C_{s2}\hat{x}_{s2}(t)) \end{array} \tag{2.22}$$

where O_{s1} and O_{s2} are the observability matrices for subsystems 1 and 2 respectively as defined in (2.10). K_{s1} and K_{s2} are finite gain vectors that can be tuned to change the speed of convergence (Ciccarella *et al.*, 1993) and $\hat{x}_{s1}$ and $\hat{x}_{s2}$ are the estimates of the subsystem states. f_{s1}, f_{s2}, C_{s1} and C_{s2} in (2.22) are the same as in (2.20). Due to the difficulty of calculating O_{s1} and O_{s2} explicitly, they are approximated following the approach in Section 2.3.2. As the c_{st_i} $(i = 1, \ldots, 5)$ states appear in both subsystems, there are two ways to obtain the state estimate for the entire system:

$$\hat{x} = [\hat{x}_{s1,1}, \hat{x}_{s1,2}, \hat{x}_{s2,2}, \hat{x}_{s1,3}, \hat{x}_{s1,4}, \hat{x}_{s2,4}, \ldots, \hat{x}_{s1,9}, \hat{x}_{s1,10}, \hat{x}_{s2,10},] \tag{2.23a}$$

$$\hat{x} = [\hat{x}_{s2,1}, \hat{x}_{s1,2}, \hat{x}_{s2,2}, \hat{x}_{s2,3}, \hat{x}_{s1,4}, \hat{x}_{s2,4}, \ldots, \hat{x}_{s2,9}, \hat{x}_{s1,10}, \hat{x}_{s2,10},] \tag{2.23b}$$

where $\hat{x}$ represents the estimate of the entire system state. $\hat{x}_{s1,i}$ and $\hat{x}_{s2,i}$, $i = 1, \ldots, 10$, represent the i-th element in x_{s1} and x_{s2} respectively. If y_{s1} is known to be more

reliable than y_{s2}, (2.23a) should be used; if y_{s2} is more reliable, (2.23b) should be used.

2.4.5 Simulation results

Model parameters and simulation settings

The model parameters used in the simulations are shown in Table 2.1. Parameters are obtained from Canright *et al.*, 1981.

Table 2.1: Froth flotation model parameters used in simulations.

k	=	3.81 l/min
c_∞	=	17.91 kg/m^3
f_r	=	0.2
ρ_l	=	1000 kg/m^3
ρ_s	=	1299 kg/m^3
β	=	290.3 kg/min
V	=	18 m^3
c_{st_0}	=	10 wt% of feed
c_{lt_0}	=	90 wt% of feed
c_{at_0}	=	24.5 wt% of solids

The volumetric flow of feed to the first tank ($\dot{V}_{uf_0}$) is set to be a sinusoid ($21.71 + 7.60\sin(\omega t)$ m^3/min) with $\omega = \frac{2\pi}{10}$ rad/min. It is assumed that the feed flow rate up to the current time is available to all three state estimation methods; at t_k, $\dot{V}_{uf_0}(0)$ up to $\dot{V}_{uf_0}(t_k)$ are known. The inputs are set to constant values with $u_1 = 0.125$ g/kg dry feed and $u_2 = 0.3125$ g/kg dry feed. This results in constant values for k and c_∞.

Samples of the two measured states (c_{lt5} and c_{at5}) are taken every 1 minute; that is, $\Delta = 1$ min. Both measurements are subject to measurement noise. The measurement noise is Gaussian white noise generated as follows:

$$v(t_k) \backsim [N(0, 50^2), N(0, 6^2)]^T \tag{2.24}$$

As mentioned in Section 2.3.1 liquid concentrations can be measured with $1 - 6\%$ accuracy while ash measurements can have errors of up to 30% at low ash concentrations. Typical values of c_{lt_i} and c_{at_i} are 1000 kg/m^3 and 20 kg/m^3 respectively; the standard deviations of the noise correspond to a 5% error in liquid measurements and a 30% error in ash measurements.

The process is simulated by using the classical-Runge-Kutta method to integrate the system model (2.8) using an integration step of $h = 0.10$ min. Random noise is added to the right-hand-side of system model (2.8) to simulate process noise/disturbances. The process noise associated with tank i, $i = 1, \ldots, 5$, is Gaussian white noise generated as follows:

$$w_i(t) \backsim [N(0, 3^2), N(0, 7^2), N(0, 0.25^2)]^T \tag{2.25}$$

The parameters used in the EKF are $Q = \text{diag}([3^2, 7^2, 0.25^2, 3^2, 7^2, 0.25^2, 3^2, 7^2, 0.25^2, 3^2, 7^2, 0.25^2, 3^2, 7^2, 0.25^2]^T)$, $R = \text{diag}([10^2, 5^2]^T)$ and $P(0) = \text{diag}([100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100]^T)$ where $\text{diag}(v)$ represents a diagonal matrix whose diagonal elements are the elements of a vector v. Q, R and $P(0)$ are diagonal matrices with the assumption that the process noise, measurement noise and state estimation errors are all uncorrelated.

The estimation windows size in the MHE is picked to be $N = 3$. A small window size is used due to the large number of system states to reduce the complexity of the MHE optimization problem. Q and R are the same as those used in the EKF. The $\Pi(0)$ used in the arrival cost (2.19) is equal to $P(0)$ used in the EKF. The state estimate at the start of the window (z from (2.19)) for tank i, $i = 1, \ldots, 5$, has the lower and upper bounds $[0, 800, 0]$ and $[200, 1000, 30]$ respectively. The process noise $w_i(t)$ associated with tank i, $i = 1, \ldots, 5$, has the lower and upper bounds $[-100, -100, -100]$ and $[100, 100, 100]$ respectively. There are no bounds on measurement noise.

The nonlinear observer parameters are as follows:

$$\begin{aligned} K_{s1} = K_{s2} = &[1.500 \cdot \epsilon, 5.987 \cdot \epsilon^2, 6.375 \cdot \epsilon^3, 13.191 \cdot \epsilon^4, 9.915 \cdot \epsilon^5, 13.537 \cdot \epsilon^6, \\ &6.693 \cdot \epsilon^7, 6.477 \cdot \epsilon^8, 1.655 \cdot \epsilon^9, 1.144 \cdot \epsilon^{10}]^T \end{aligned} \tag{2.26}$$

where $\epsilon = 1/750,000$ for K_{s1} and $\epsilon = 1/1,750,000$ for K_{s2}. The nonlinear observer we use is designed for continuous systems with continuous measurements available. In the froth flotation system, measurements are only obtained at discrete sampling intervals of 1 min which is too large for the measurements to be considered continuous. Due to this lack of measurement information, the observer cannot be as aggressive which results in the small gain values. If a smaller sampling time was used then larger

values of ϵ could be used. Another reason for the small values of ϵ are the large values of elements in the O_{s1}^{-1} and O_{s2}^{-1} matrices. The initial state used in the simulations is:

$$x(0) = [64.5, 956.8, 23.7, 42.0, 975.7, 21.7, 31.8, 991.6, 20.7, 24.4, 994.2, 18.4, 21.5, 983.3, 15.8]^T \tag{2.27}$$

and the initial guess provided to the state estimators:

$$\hat{x}(0) = [80.0, 950.0, 30.0, 35.0, 950.0, 15.0, 40.0, 990.0, 25.0, 15.0, 980.0, 15.0, 30.0, 1000.0, 24.0]^T \tag{2.28}$$

All simulations are carried out with an Intel Core i5 computer at 3.30 GHz and 8.00 GB RAM. The MHE optimization problems are solved using the open source interior point optimizer Ipopt (Wächter and Biegler, 2006).

Results without model plant mismatch

In this section, we evaluate the performance of the three state estimation methods without considering model plant mismatch.

Figure 2.3 shows the actual state trajectory and the estimates obtained using the EKF, MHE and nonlinear observer when there is no model plant mismatch. From the figure, it can be seen that all the three methods track the trend of the actual trajectories well. The normalized error is used to compare the performance of these methods as it reduces the effects of states having different orders of magnitude. Figure 2.4 shows the trajectories of the normalized error for the three state estimation methods with the normalized error defined as:

$$e(t_k) = \sqrt{\sum_{i=1}^{15} (e_i(t_k))^2} \tag{2.29}$$

where $e(t_k)$ is the normalized error at t_k and $e_i(t_k)$ is the normalized error in state i, $i = 1, \ldots, 15$, defined as follows:

$$e_i(t_k) = \frac{\hat{x}_i(t_k) - x_i(t_k)}{\max(\hat{x}_i - x_i)} \tag{2.30}$$

where the maximum error for a given state i is the largest error for state i between all three estimation methods. The average normalized error is: EKF - 1.38, MHE - 1.40 and observer - 1.54. The EKF and MHE have similar performance as the system is

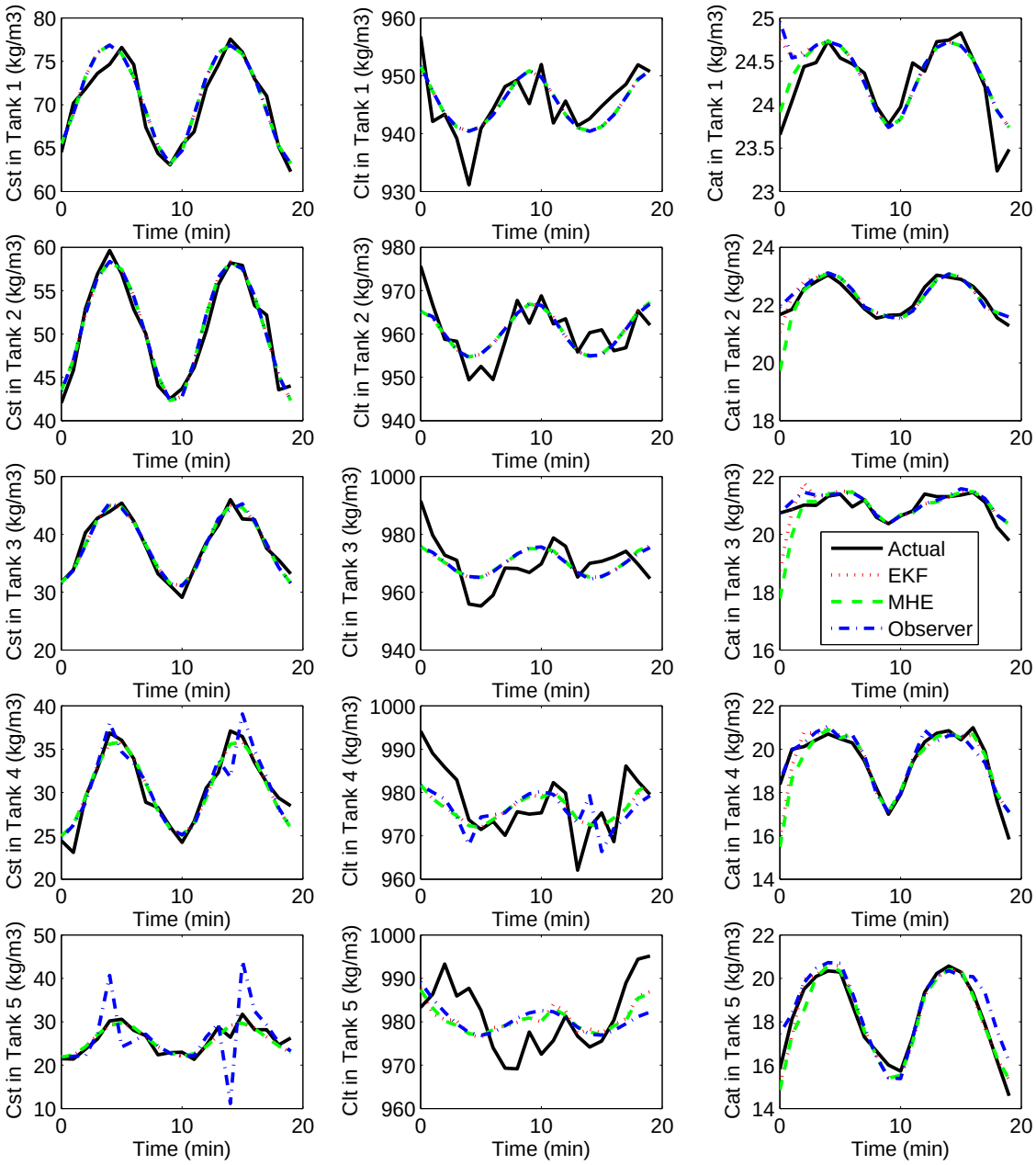

Figure 2.3: Trajectories of the actual process states and estimates given by the three estimation methods for no model mismatch and no measurement delay.

not operating near its constraints and the MHE window size is small. The observer performs worse than the EKF and MHE as it ignores noise information while the EKF and MHE both use noise information. For this process, using larger values of N in the MHE may provide slightly improved estimation performance but at the cost of a significant increase in evaluation time.

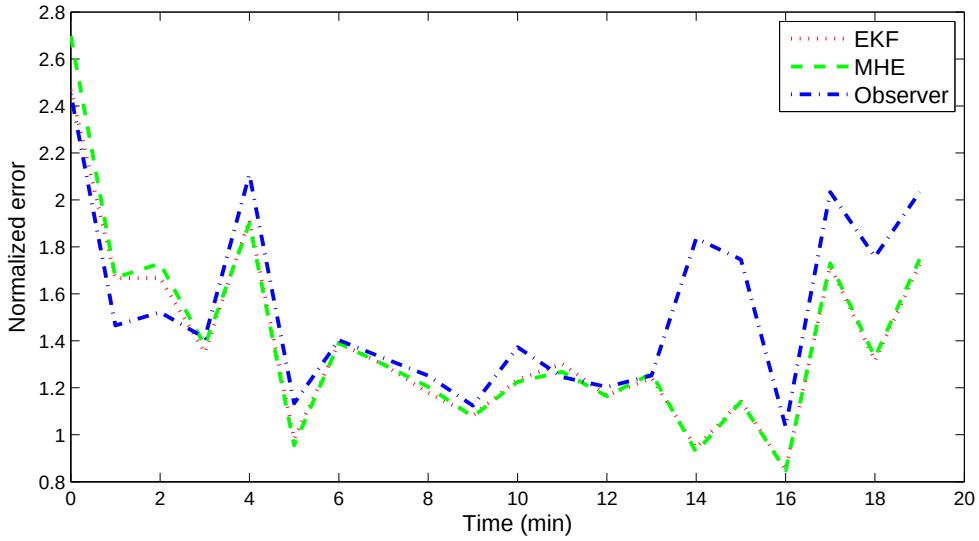

Figure 2.4: Trajectories of the normalized estimation error of the three estimation methods for no model mismatch and no measurement delay.

Results with model plant mismatch

In this section, the model used in the estimation methods (EKF, MHE, nonlinear observer) is not the same as the actual plant. The mismatch is introduced by setting r_i and x_{A_i}, $i = 1, \ldots, 5$, from (2.2) in the estimation models to be 20% higher than the actual (plant) values of r_i and x_{A_i}, $i = 1, \ldots, 5$.

Figure 2.5 shows the actual state trajectory and the estimates obtained using the EKF, MHE and nonlinear observer when there is a model plant mismatch. Figure 2.6 shows the trajectories of the normalized error for all three estimation methods. The performance of all three state estimation methods is worse than when there is no model plant mismatch, with the average normalized error being: EKF - 2.12, MHE - 2.14 and observer - 2.14. The observer has similar performance to the EKF and MHE because the noise information is masked by the model plant mismatch which reduces

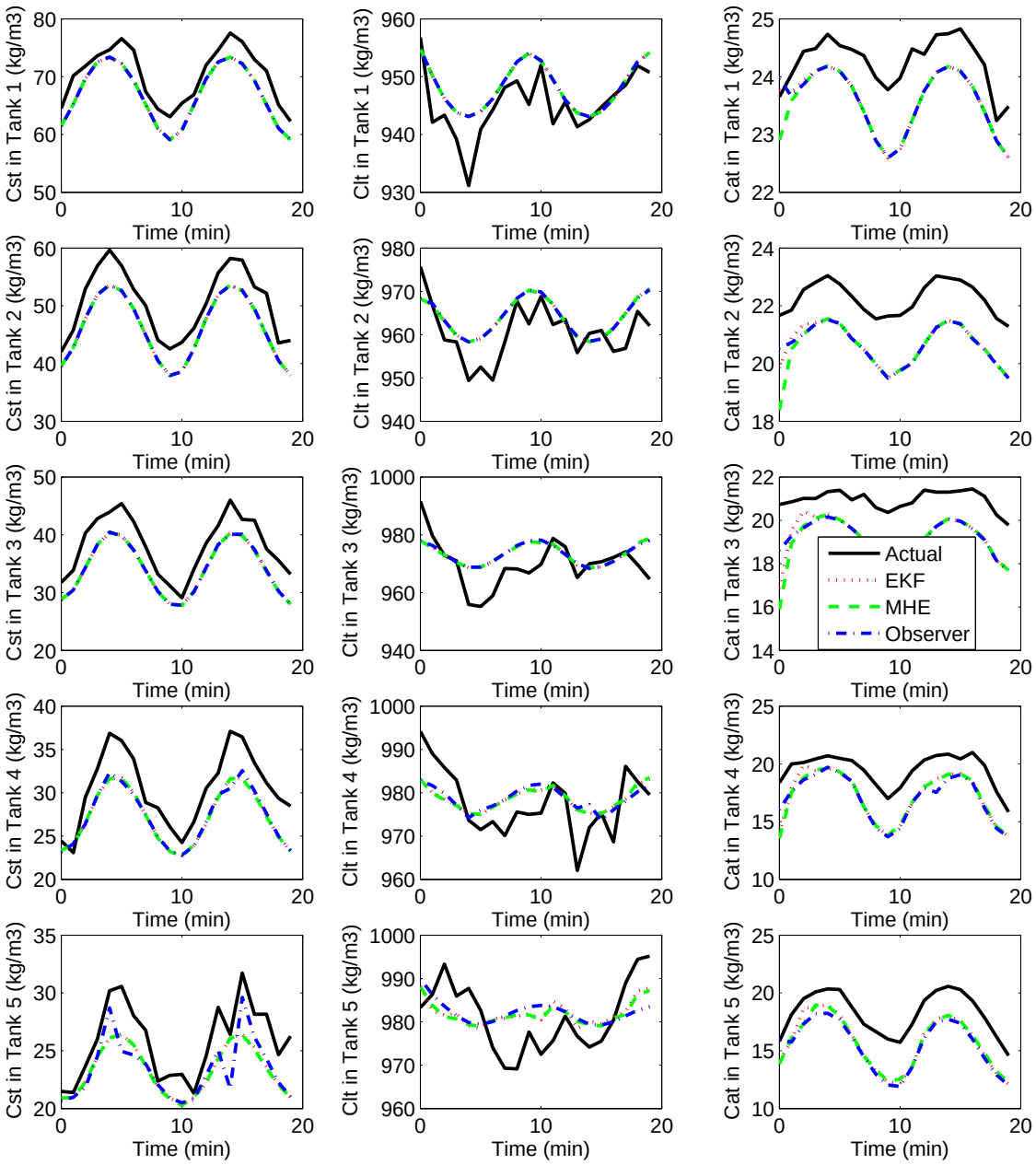

Figure 2.5: Trajectories of the actual process states and estimates given by the three estimation methods for model mismatch and no measurement delay.

the benefits of using the noise information. All three consistently underestimate solid and ash concentrations and usually overestimate liquid concentrations. This is because the models suggest that more solids/ash are being removed than is actually the case as r_i and x_{A_i} are higher in the model than the actual plant.

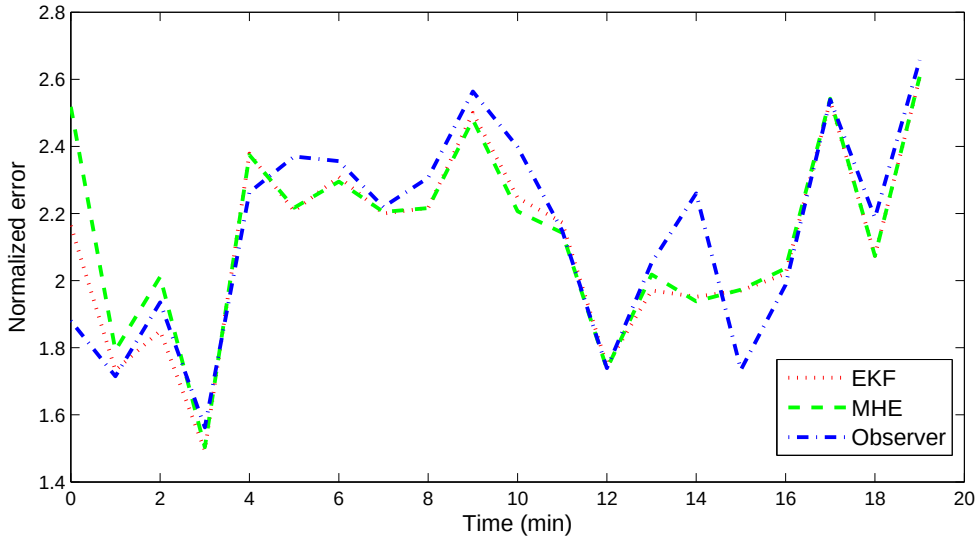

Figure 2.6: Trajectories of the normalized estimation error of the three estimation methods for model mismatch and no measurement delay.

2.5 State estimation subject to measurement delays

2.5.1 Modeling of measurements

In the previous section, both measurements (y_{s1} and y_{s2}) were assumed to arrive at the EKF/MHE/observer immediately (without delay). In this section the measurements are assumed to have random delays such that $y_{s1}(t_k)$ arrives at $t_{k+d_1(t_k)}$ and $y_{s2}(t_k)$ arrives at $t_{k+d_2(t_k)}$ where $d_1(t_k)$ and $d_2(t_k)$ are the delays associated with $y_{s1}(t_k)$ and $y_{s2}(t_k)$ respectively. It is assumed that samples are taken at each time step and the delays are due to sample processing time and communication delays. At a given time instant, we must rearrange the received measurements in the order they were taken before carrying out state estimation. This step is necessary as multiple measurements can arrive simultaneously due to the random delays. It is assumed that samples are

tagged with the time when they are taken.

2.5.2 Design of the EKF, MHE and observer

The EKF is designed based on the filter recalculation method described in Gopalakrishnan *et al.*, 2011. After the rearrange step, the EKF is recalculated from when the oldest measurement received at the current time was taken. If any measurements along the trajectory are available they are included in the EKF calculations. If at a given time any measurements are missing, the missing measurement is set to be the same as the predicted value i.e., $y(t_k) = C\hat{x}(t_k|t_{k-1})$. This is the equivalent of setting (2.17b) in the original EKF design to be:

$$\hat{x}(t_k|t_k) = \hat{x}(t_k|t_{k-1}) \tag{2.31}$$

The MHE is designed based on Valencia *et al.*, 2011. The window size for the MHE is set to the difference between the maximum possible delay ($\max(d_1, d_2)$) and the minimum possible delay ($min(d_1, d_2)$). The MHE cost function (2.18a) is modified as follows: the arrival cost is omitted due to the large window size and any terms involving missing measurements (v_k terms when no measurement is available) are omitted.

For the nonlinear observers, the estimated state trajectory is recalculated from when the oldest measurement received at the current time was taken (similar to the EKF). If any measurements along the trajectory are available they are included in the observer calculations. If no measurement is available for a subsystem at a given time then a predictor is used for that subsystem. The predictor equation is:

$$\dot{\hat{x}}_l(t) = f(\hat{x}_l(t), u(t), 0); \tag{2.32}$$

where l is the subsystem number ($s1$ or $s2$). The predictor is simply the first term of the observer (2.22). If at a given time one subsystem has a measurement available while the other does not, then the state estimates from the subsystem with the measurement are used to update the overlapping state estimates from the other subsystem. Once the overlapping states have been updated the observer or predictor update to the next time step continues as normal. Figure 2.7 illustrates this procedure. In the figure, $y_{s1}(t_{k-4})$, $y_{s2}(t_{k-3})$ and $y_{s1}(t_{k-1})$ are available at the current time t_k.

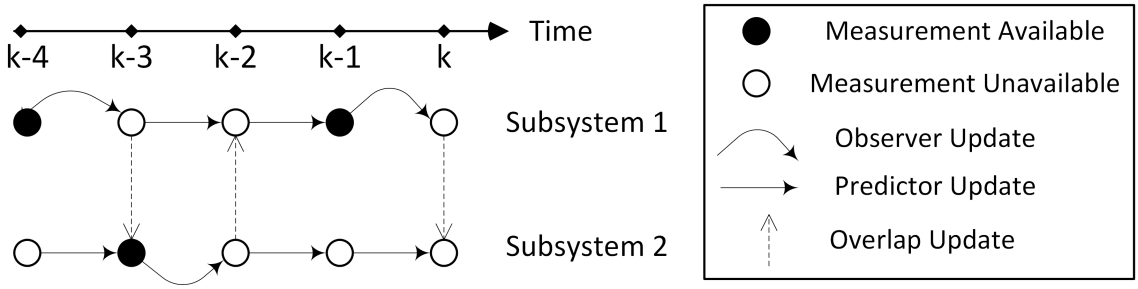

Figure 2.7: Outline of observer/predictor procedure

2.5.3 Simulation Results

Model parameters and simulation settings

The model parameters and simulation settings are the same as in the no delay case (Section 2.4.5). The random delay is such that $10 \leq d_1(t), d_2(t) \leq 20$. These delay values are chosen based on the dynamics of the XRF sensors used. The EKF and observer parameters are the same as the no delay case. All MHE parameters except window size are the same as the no delay case. The MHE window size (N) is 10.

Results without model plant mismatch

In this section, results are presented for when there is no model plant mismatch. Figure 2.8 shows the actual trajectory and the estimates obtained using the EKF, MHE and nonlinear observer. Figure 2.9 shows the trajectories of the normalized error for the three methods. The average normalized error for all estimation methods is 1.70. It is not possible to differentiate between the three methods because: (a) with random delay between 10 and 20, at a given sampling time the EKF, MHE and nonlinear observer at best only receive measurements taken 10 sampling periods ago. Thus state prediction dominates over state estimation in all three; and (b) the froth flotation process is an open-loop stable process which makes differences in the state estimates die out in the prediction phase if measurements are not available.

Results with model plant mismatch

In this section, the model used in the estimation methods (EKF/MHE/nonlinear observer) is not the same as the actual plant. The mismatch is introduced the same way as in the no measurement delay case, by setting r_i and x_{A_i}, $i = 1, \ldots, 5$, from

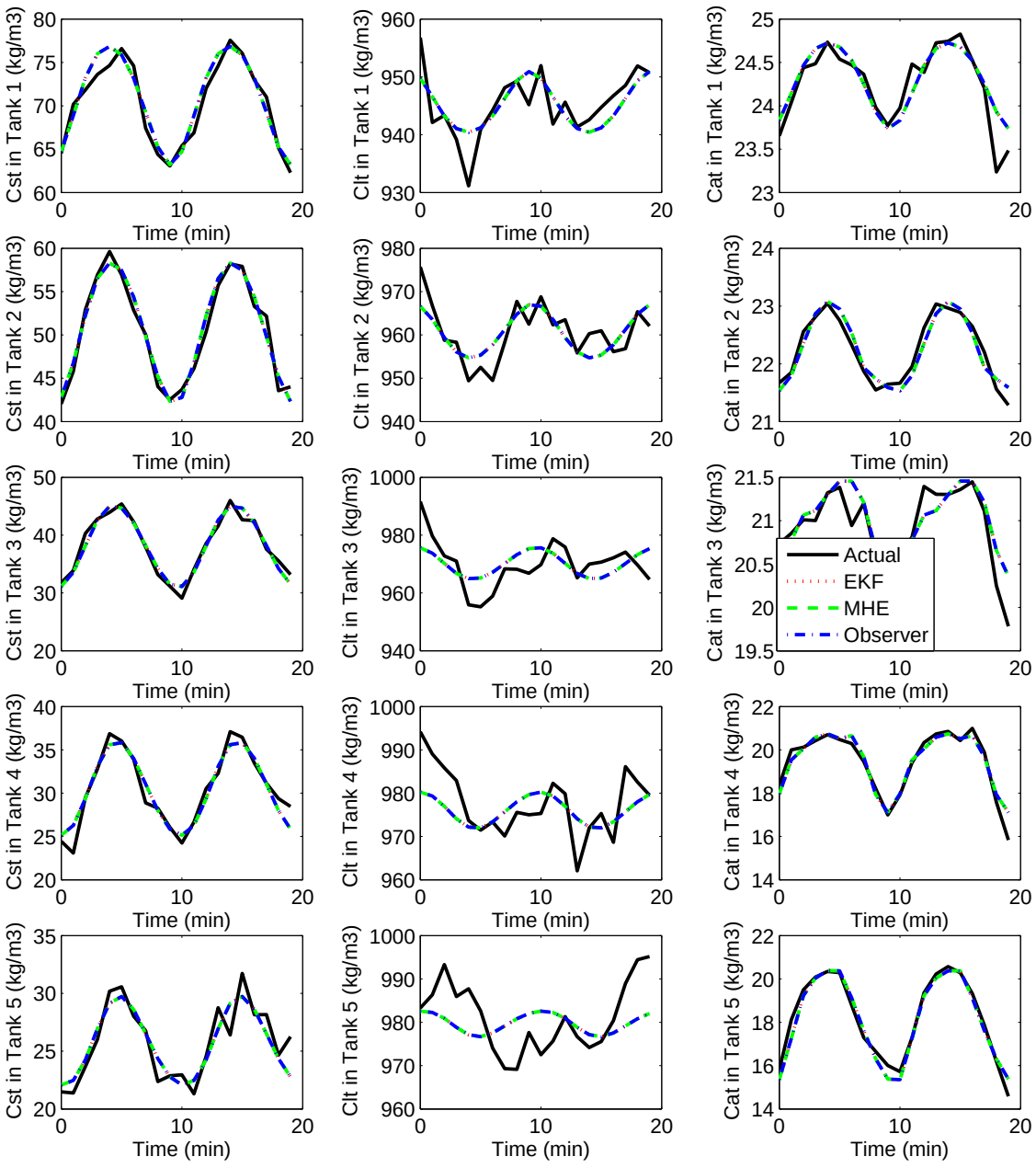

Figure 2.8: Trajectories of the actual process states and estimates given by the three estimation methods for no model mismatch and measurement delay.

(2.2) in the estimation models to be 20% higher than the actual (plant) value of r_i and x_{A_i}, $i = 1, \ldots, 5$.

Figure 2.10 shows the actual trajectory and the estimates obtained using the EKF, MHE and nonlinear observer. Figure 2.11 shows the trajectories of the normalized

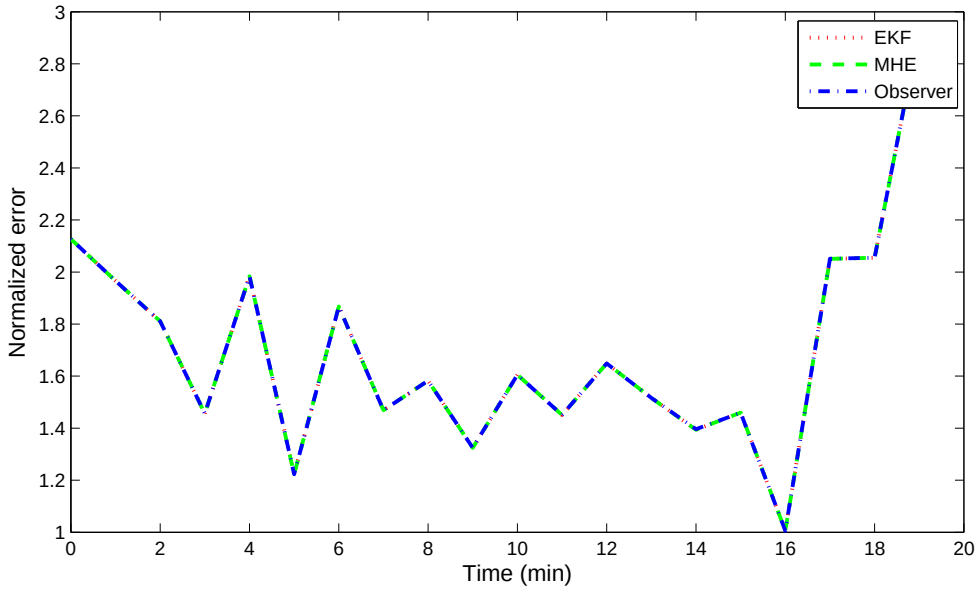

Figure 2.9: Trajectories of the normalized estimation error of the three estimation methods for no model mismatch and measurement delay.

error for both estimation methods. The average normalized error for all three estimation methods is 2.42. Just as in the no delay case, the estimation methods consistently underestimate solid and ash concentrations and usually overestimate liquid concentrations because the models suggest that more solids/ash are being removed than is actually the case due to higher r_i and x_{A_i} in the model than the actual plant.

2.6 Conclusions

In this chapter, three state estimation methods (EKF, MHE and nonlinear observer) for a coal froth flotation process were compared. A typical 5 tank flotation system was used in simulations. The performance under various conditions (with/without random measurement delay and with/without model plant mismatch) was investigated.

The EKF gave the best estimation performance under normal conditions (no measurement delay and no model plant mismatch). The MHE performed worse than the EKF because the system was not operating near its constraints and the estimation window size was small. The MHE also had a higher computational load than the EKF: the runtime of the MHE was significantly longer than the sampling time. The nonlinear observer ran the fastest but had the worst estimation performance. All three

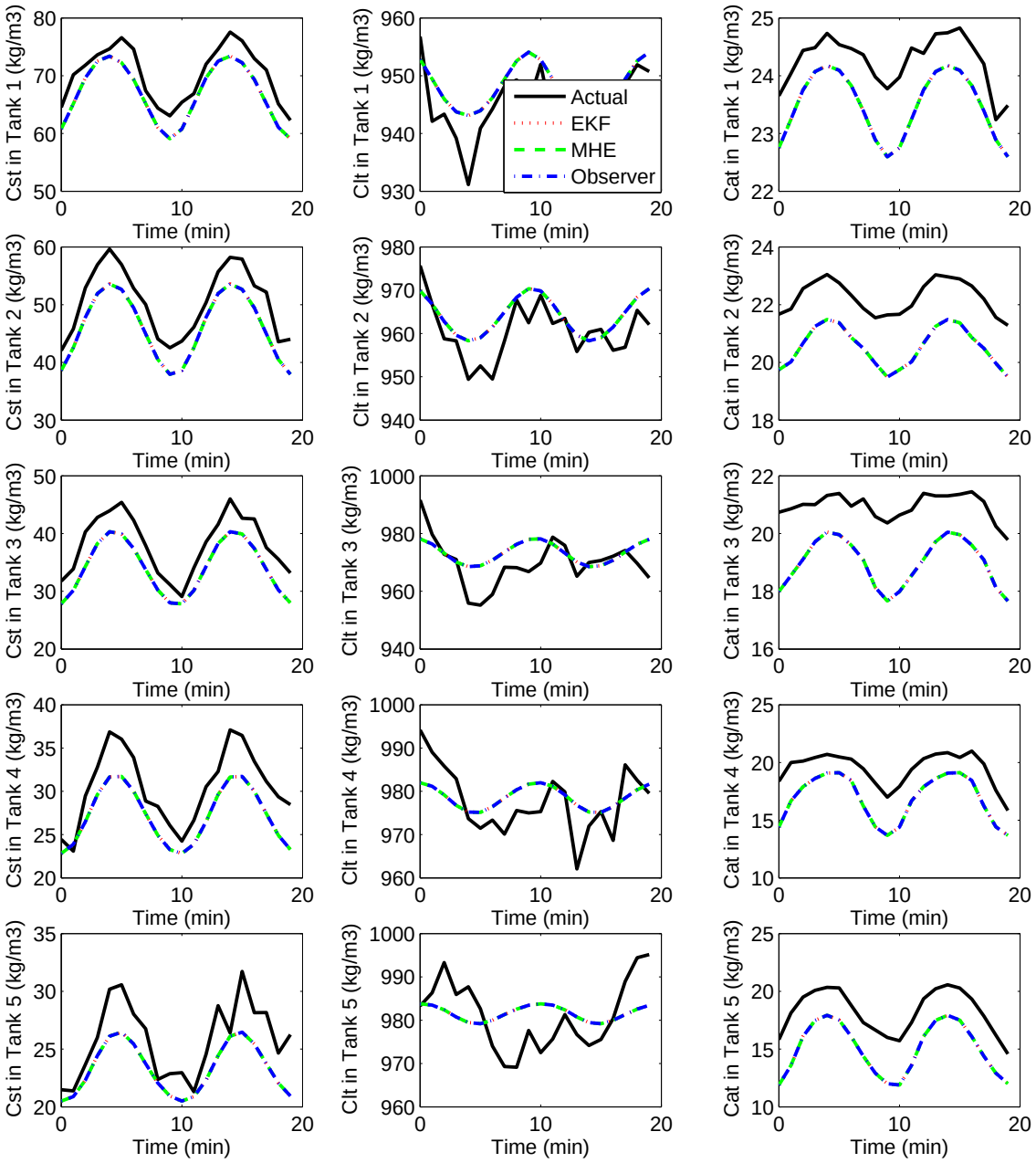

Figure 2.10: Trajectories of the actual process states and estimates given by the three estimation methods for model mismatch and measurement delay.

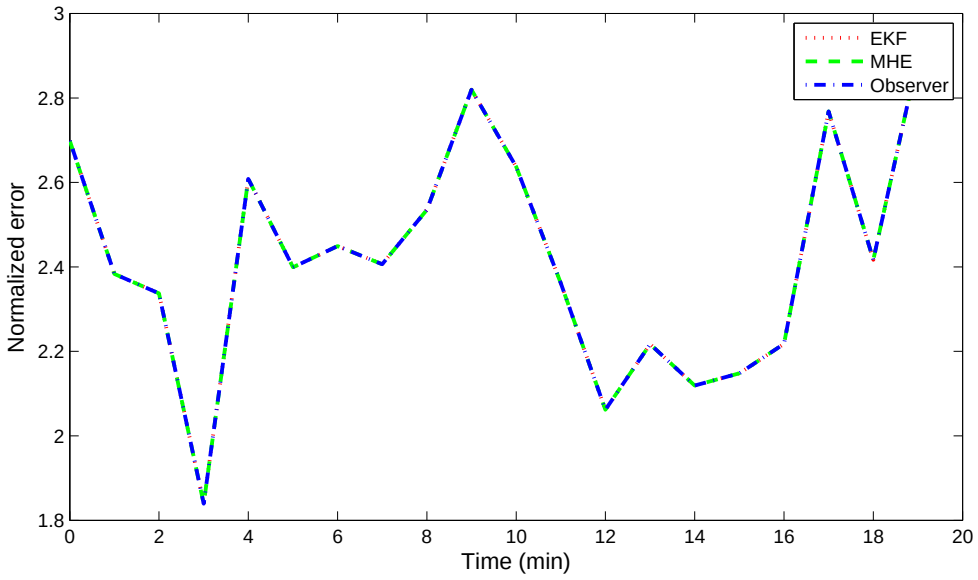

Figure 2.11: Trajectories of the normalized estimation error of the three estimation methods for model mismatch and measurement delay.

methods had similar estimation performance when there was model plant mismatch.

In the presence of measurement delay, all three methods gave similar performance with or without model plant mismatch. This was because the lack of measurements meant that state prediction dominated over state estimation. As the froth flotation system is open-loop stable, this meant that differences between estimates by the different methods disappeared in the prediction phase. Overall, the EKF is the best choice of estimator for this system as it provides a good balance of speed and accuracy.

Chapter 3

Multiple model based approach to state estimation in the presence of model plant mismatch (MPM)

3.1 Introduction

State estimation plays a fundamental role in the success of MPC (Muske and Badgwell, 2002). Model based state estimation methods are sensitive to model-plant mismatch (MPM) which is always present in real systems due to factors such as external disturbances, parameter drift, sensor failures and others (Botelho *et al.*, 2015). Due to the popularity of MPC, it becomes necessary to design state estimation methods that provide accurate results even in the presence of MPM.

Numerous approaches have been proposed in the literature to address this issue. One approach, as described in Parlos *et al.*, 2002, is to use a data driven model to capture the effects of the mismatch. However, while this method can provide unbiased state estimates it does not provide any information about the cause of the mismatch. This is because the data driven model has no physical significance while the states and parameters in chemical systems do.

An alternative approach, as described in Salhi and Bouani, 2016, is to reduce model plant mismatch by updating the system parameters at each time step by solving an optimization problem. This approach provides diagnostic information through the updated parameter estimates but has a high computational load as an optimization problem has to be solved at each time step. If a large number of parameters have to be optimized then computation time can be significant.

Similar to offset free model predictive control, it is possible to augment the system model with constant step disturbances and use special observer designs to obtain unbiased estimates (Lee *et al.*, 2012; Yang and Liu, 2016). However these approaches suffer the same downside as Parlos *et al.*, 2002 in that they do not provide any information about the potential cause of the mismatch.

A common approach in tracking and fault detection applications for mechanical and electrical systems is the use of Multiple Model (MM) methods (Semerdjiev *et al.*, 2000; Jiao *et al.*, 2015; Pitre *et al.*, 2005; Amirzadeh *et al.*, 2011; Huang and Leung, 2005; Zhang and Li, 1998). These methods work by having multiple models (each model representing one mode of the system) run in parallel. Model probabilities and the overall estimate are obtained according to the rules of the particular method being used. It is possible to use augmented models with MM methods to obtain updated parameter estimates which provide diagnostic information (Semerdjiev *et al.*, 2000). MM methods offer the advantage of low computational cost while still providing diagnostic information, however there are only limited examples of their use in chemical systems (Chetouani, 2008; Kuure-Kinsey and Bequette, 2010). Furthermore, these works focus on unbiased output estimation and control which does not necessarily guarantee unbiased state estimation. The design of an appropriate model set to use with MM methods is also challenging as there needs to be enough separation between models based on output residuals (Zhang and Li, 1998) and enough models to capture the range of system dynamics but using too many models will decrease performance (Li, 2002).

In this chapter, we present an algorithm for the use of a MM method for state estimation in the presence of MPM caused by parameter mismatch. The model set used in our approach includes models augmented with different parameters (augmented models do not have the same states). Guidelines for model set design and assumptions on model properties are also presented. Simulations examples are included to illustrate the effectiveness of this approach.

3.2 Problem description

Consider a nonlinear plant whose dynamics are characterized as follows:

$$\dot{x} = f(x, \theta^*, u, w) \tag{3.1a}$$

$$y = g(x) + v \tag{3.1b}$$

where θ^* is the vector of actual plant parameters ($\theta^* \in \mathbb{R}^m$), x is the state vector ($x \in \mathbb{R}^n$), u is the vector of inputs ($u \in \mathbb{R}^p$) and y is the vector of outputs ($y \in \mathbb{R}^q$). w and v represent process noise and measurement noise respectively. We assume a model of the nonlinear plant is developed and has the form:

$$\dot{x} = f(x, \theta, u, 0) \tag{3.2a}$$

$$y = g(x) \tag{3.2b}$$

where θ is the vector of model parameters ($\theta \in \mathbb{R}^m$).

A common approach to deal with parameter uncertainty is to augment the uncertain parameters as states and use the augmented model in estimators (Semerdjiev *et al.*, 2000). Parameters can be added to the state vector to obtain augmented systems of the form:

$$\dot{\tilde{x}} = \begin{bmatrix} \dot{x}_x \\ \dot{x}_\theta \end{bmatrix} = \begin{bmatrix} \tilde{f}(\tilde{x}, \tilde{\theta}, u, 0) \\ 0 \end{bmatrix} \tag{3.3a}$$

$$y = \tilde{g}(\tilde{x}) \tag{3.3b}$$

where $\tilde{x}$ represents the augmented state vector ($\tilde{x} \in \mathbb{R}^{\tilde{n}=n+r}$) consisting of the original state vector ($x_x \in \mathbb{R}^n$) and the added parameters ($x_\theta \in \mathbb{R}^r$). $\tilde{\theta}$ is the vector of remaining parameters ($\tilde{\theta} \in \mathbb{R}^{m-r}$).

The objective of this work is to outline a Multiple Model (MM) approach that uses augmented models of the form (3.3) and is capable of providing unbiased state estimates when the following assumptions are satisfied:

Assumption 3.2.1 *There is only parametric mismatch in the model and the actual plant parameters (θ^*) are time invariant.*

Assumption 3.2.2 *Each parameter has a unique effect on the output, i.e. each unique set of parameter values results in a unique y trajectory for a given constant input.*

Assumption 3.2.3 *The effects of the parameters are distinguishable from noise.*

Assumptions 3.2.2 and 3.2.3 are needed to ensure that the parameters can be distinguished from each other based on noisy output measurements which is a necessary condition for the success of any MM method (Zhang and Li, 1998). Some additional assumptions are also required and these are presented in Section 3.4.3.

3.3 Illustrative example - Two CSTRs

In this section, we present a plant which will be used to illustrate each step of our algorithm. The plant model is obtained from Sun and El-Farra, 2008 and a schematic diagram is provided in Figure 3.1. The plant consists of two well-mixed, non-isothermal continuous stirred tank reactors (CSTRs) where three parallel, irreversible, elementary, exothermic reactions take place: $A \xrightarrow{k_1} B$, $A \xrightarrow{k_2} U$ and $A \xrightarrow{k_3} R$. A is the reactant, B is the desired product and U and R are undesired byproducts. Under standard modeling assumptions, the plant model is:

$$\dot{T}_1 = \frac{F_0}{V_1}(T_0 - T_1) + \frac{F_r}{V_1}(T_2 - T_1) + \sum_{i=1}^{3} G_i(T_1)c_{A1} + \frac{Q_1}{\rho c_p V_1} \tag{3.4a}$$

$$\dot{c}_{A1} = \frac{F_0}{V_1}(c_{A0} - c_{A1}) + \frac{F_r}{V_1}(c_{A2} - c_{A1}) - \sum_{i=1}^{3} R_i(T_1)c_{A1} \tag{3.4b}$$

$$\dot{T}_2 = \frac{F_1}{V_2}(T_1 - T_2) + \frac{F_3}{V_2}(T_{03} - T_2) + \sum_{i=1}^{3} G_i(T_2)c_{A2} + \frac{Q_2}{\rho c_p V_2} \tag{3.4c}$$

$$\dot{c}_{A2} = \frac{F_1}{V_2}(c_{A1} - c_{A2}) + \frac{F_3}{V_2}(c_{A03} - c_{A2}) - \sum_{i=1}^{3} R_i(T_2)c_{A2} \tag{3.4d}$$

where $R_i(T_j) = k_i \exp(-E_i/RT_j)$, $G_i(T_j) = (-\Delta H_i/(\rho c_p))R_i(T_j)$ for $j = 1, 2$. T_j, c_{Aj}, Q_j and V_j represent the temperature of the reactor, the concentration of A, the rate of heat input to the reactor and the reactor volume respectively with the subscript representing the CSTR number. ΔH_i, k_i, E_i, $i = 1, 2, 3$ represent the enthalpies, pre-exponential constants and activation energies of the three reactions respectively. c_p and ρ are the heat capacity and density of fluid in the reactor. The state vector is $x = [T_1, c_{A1}, T_2, c_{A2}]^T$ and the input vector is $u = [Q_1, Q_2]^T$. There are two measured outputs, $y = [x_1, x_3]^T$. Nominal parameter values, steady states and associated inputs are provided in Table 3.1.

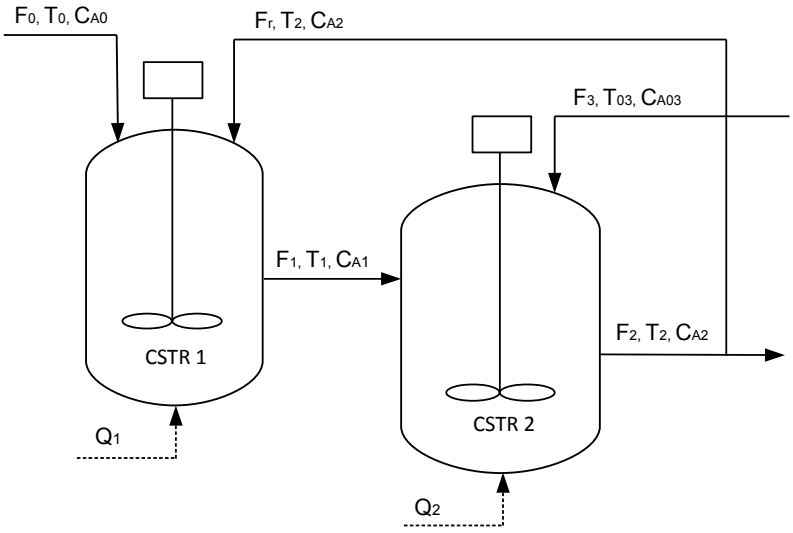

Figure 3.1: Schematic diagram of 2 CSTR example

Table 3.1: Nominal parameters, steady states and steady state inputs for two CSTRs

F_0	=	4.998 m^3/h	k_1	=	3.0×10^6 h^{-1}
F_1	=	39.996 m^3/h	k_2	=	3.0×10^5 h^{-1}
F_3	=	30.0 m^3/h	k_3	=	3.0×10^5 h^{-1}
F_r	=	34.998 m^3/h	E_1	=	5.0×10^4 kJ/kmol
V_1	=	1.0 m^3	E_2	=	7.53×10^4 kJ/kmol
V_2	=	3.0 m^3	E_3	=	7.53×10^4 kJ/kmol
R	=	8.314 kJ/kmol K	c_p	=	0.231 kJ/kg K
T_0	=	300.0 K	ρ	=	1000.0 kg/m^3
T_{03}	=	300.0 K	T_1^s	=	303.7 K
c_{A0}	=	4.0 $kmol/m^3$	c_{A1}^s	=	2.5 $kmol/m^3$
c_{A03}	=	2.0 $kmol/m^3$	T_2^s	=	302.9 K
ΔH_1	=	-5.0×10^4 kJ/kmol	c_{A2}^s	=	2.3 $kmol/m^3$
ΔH_2	=	-5.2×10^4 kJ/kmol	Q_1^s	=	1.0×10^5 kJ/h
ΔH_3	=	-5.4×10^4 kJ/kmol	Q_2^s	=	1.0×10^5 kJ/h

The objective is to estimate the four states based on the two measured outputs in the presence of parameter uncertainty. In this example, we show that an estimator based on augmented models may give unbiased output estimates but that this does not guarantee unbiased state estimates. To emphasize the effects of the parameters, process and measurement noise are not considered in this example.

Let us consider a scenario where the actual $F_3 = 50$ m^3/h (for an unidentified reason) while the value used in the model is $F_3 = 30$ m^3/h (the nominal value). Figure 3.2 shows the trajectories of the estimates given by an EKF using a model augmented with T_0 and T_{03} and a non-augmented model (where parameter uncertainty is not considered). The initial state of the EKF is the nominal steady state (when $F_3 = 30$ m^3/h). It can be seen from the figure that there is parameter uncertainty as the actual outputs move away from their nominal values. It can also be seen that if parameter uncertainty is not considered (EKF with a non-augmented model), the EKF gives biased output estimates as well as biased state estimates. Further, it can be seen that the EKF with the augmented model gives unbiased output estimates even though the augmented parameters are not the ones that contain mismatch. Thus, unbiased output estimates do not guarantee unbiased state estimates as can be seen from the figure.

This motivates our present work to develop a procedure for the design of estimators based on augmented models which will guarantee unbiased states estimates if unbiased output estimates are obtained.

3.4 Proposed solution

This section provides an overview of our procedure for state estimation using a MM approach. A flowchart of the propsed procedure is presented in Figure 3.3. The different steps are explained in the following subsections.

3.4.1 Parameter set selection

The first stage is to determine which parameters contain or are likely to contain mismatch. This can be carried out by using old process data or prior knowledge about the system. For example, rate constants found by extrapolation are likely to

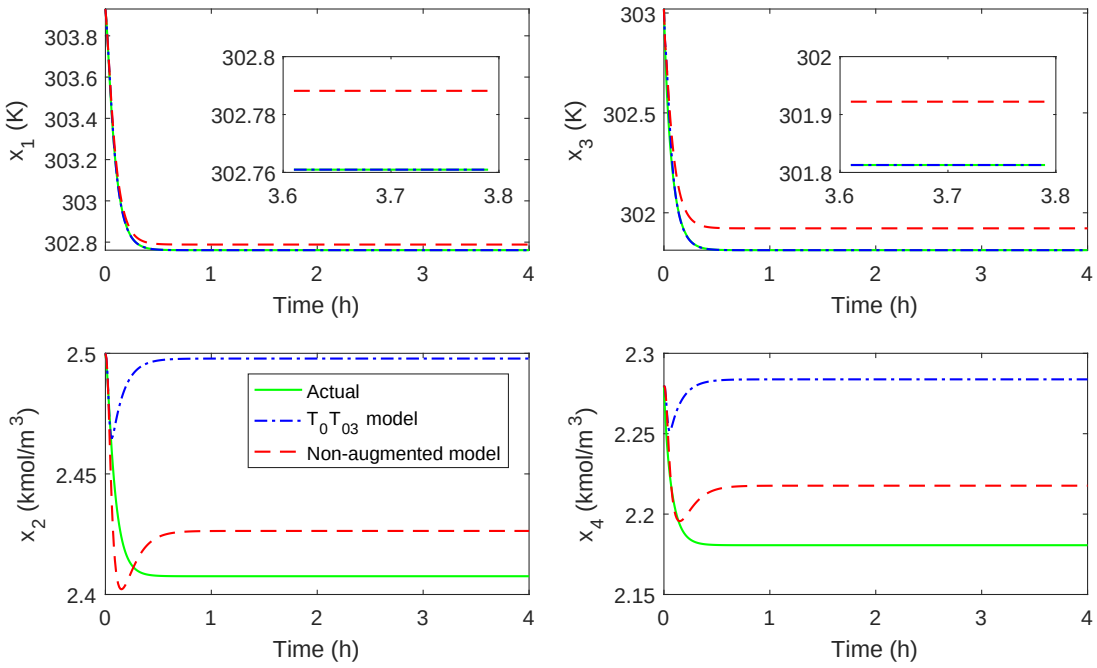

Figure 3.2: Trajectory of the actual states and estimates given by an EKF using a model augmented with T_0 and T_{03} and a non-augmented model with model plant mismatch in F_3.

contain mismatch (Canright *et al.*, 1981). There is no restriction on the number of parameters selected at this stage. Let $\mathbb{S}$ be the set of parameters selected.

For the two CSTR example, we may select $\mathbb{S} = \{F_0, F_r, F_3, k_1, V_1, V_2, T_0, T_{03}\}$ as these parameters are likely to have mismatch. Mismatch in F_0, F_r and F_3 can be caused by factors such as valves sticking or pipe fouling. Mismatch in k_1 may result from improper mixing in the CSTRs or improper calculation of k_1 from experiments. Mismatch in V_1 and V_2 may result from valves sticking or tuning errors in the CSTR level controllers. Mismatch in T_0 and T_{03} can be caused by fouling in boilers, coolers or heat exchangers used to heat or cool the feeds (F_0 and F_3).

3.4.2 Parameter sensitivity of outputs

The next step is to determine the sensitivity of the outputs to each parameter in $\mathbb{S}$ to ensure that the parameters satisfy assumption 3.2.3. If a parameter has a negligible effect on all the outputs it will not be possible to detect mismatch in that parameter using the output data even if there is no noise. There are four possible results for

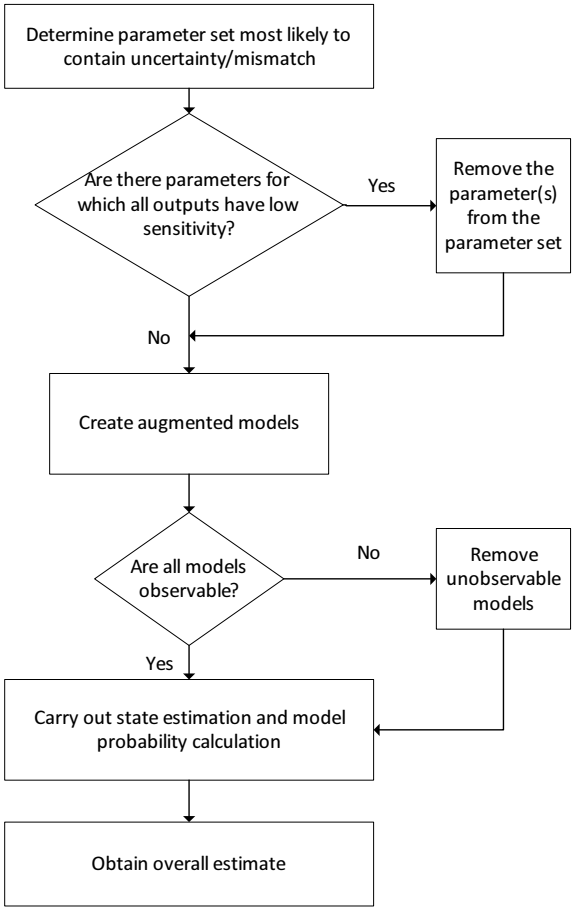

Figure 3.3: A flowchart of the proposed procedure for obtaining unbiased state estimates in the presence of parameter uncertainty.

parameter sensitivity:

1. Outputs and unmeasured states are sensitive to the parameters.
2. Outputs are sensitive to the parameters but unmeasured states are not sensitive.
3. Outputs are not sensitive to the parameters but the unmeasured states are sensitive.
4. Neither outputs nor unmeasured states are sensitive to the parameters.

In the first two cases the parameters should be left in $\mathbb{S}$. In the last case the parameters can be removed from $\mathbb{S}$ without significantly affecting estimation performance. In the third case, state estimates will be affected by the parameters however it will not be possible to detect mismatch in these parameters based on the outputs. Thus these parameters can be removed from $\mathbb{S}$ as well. Additional measurements can be taken if accurate estimates of the affected states are critical (such as for safety reasons). Sensitivity is quantitative so while a sufficiently large mismatch in the parameters from cases three and four could be detected, a sufficiently large mismatch may be unlikely or even impossible based on the physical constraints of the system. As a result, leaving these parameters in $\mathbb{S}$ will increase computation time while not improving estimation performance as the model set will be larger due to the inclusion of models augmented with these insensitive parameters. Two methods to calculate parameter sensitivity are provided in the following subsections.

Step test

The step test only considers steady state information and can only handle variations in a single parameter. However, unlike the gramian approach discussed in the next section, it can identify the sensitivity of unmeasured states to the parameters and is faster to calculate than a gramian. As such, it can be used as a preliminary test to reduce the size of $\mathbb{S}$ before carrying out the gramian test. The procedure for the step test is as follows:

1. Carry out simulations using the system model (3.2) and nominal parameter values (θ_{nom}) to find the nominal steady state (y^s_{nom}).

2. For a given non-zero input, increase (or decrease) a parameter (θ_i) by a fixed percentage while holding the other parameters unchanged. Run the simulation to find the new steady state.

3. Calculate the normalized steady state gain $K_{ij} = \dfrac{(y_j^s - y_{j,nom}^s)/y_{j,nom}^s}{(\theta_i - \theta_{i,nom})/\theta_{i,nom}}$ for $j = 1, \ldots, q$ (each output).

4. Repeat steps 2 and 3 for $i = 1, \ldots, s$ (each parameter in $\mathbb{S}$).

For a given θ_i, if the sum of the absolute normalized steady state gains $K_i = \sum_{j=1}^{q} |K_{ij}|$ is smaller than a predetermined threshold ϵ (i.e., $K_i < \epsilon$), then the outputs are said to be insensitive to that parameter and the parameter should be removed from $\mathbb{S}$. As step test results depend on the operating region, the parameter sensitivity should be checked over a typical range of manipulated input values.

The reason non-zero inputs are used in step tests is based on a property of linear systems. In general, stable non-singular linear systems of the form:

$$\dot{x} = Ax + Bu \tag{3.5}$$

will converge to a zero steady state if $u = 0$ regardless of A. If $u = 0$ parameters will have no effect on the steady state and it will not be possible to distinguish between them based on step tests. Thus nonzero inputs are imposed for the sake of generality.

Empirical observability gramian approach

The observability gramian provides a quantitative measure of observability however is difficult to calculate analytically for nonlinear systems. As a result, empirical observability gramians have been developed as local approximations to the analytical gramians for nonlinear systems (Geffen, 2008). Using these gramians for sensitivity calculations provides two major advantages over the use of step tests: transient information is used and it is possible to handle variations in multiple parameters. However gramian calculation is more computationally intensive than the step test. The procedure for calculating empirical gramians is as follows (Geffen, 2008):

1. Pick a nominal operating point (states and parameters). Add the parameters under consideration to the state vector to obtain an augmented state vector ($\tilde{x} \in \mathbb{R}^{\tilde{n}}$).

2. Define step change directions for the augmented state vector. One option is to use two level factorial design but partial factorial designs may be used to reduce computation time. These directions are listed in a semi-orthogonal matrix T with each column being one direction.

3. Define n_d step magnitudes (positive values) on a percentage scale denoting c_d as the magnitude for $d = 1, \ldots, n_d$ and define a scaling matrix S to convert the steps from percentage to actual values.

4. Define a number of experiments with one experiment for each combination of step direction and magnitude. The initial condition for each experiment is given by $\tilde{x}^{id}(0) = c_d S T e_i + \tilde{x}_{nom}$ where e_i is a standard unit vector in $\mathbb{R}^{\tilde{n}}$ and the superscript id denotes the experiment.

5. Simulate the trajectory of each experiment until it reaches steady state. This data can be used to calculate the observability gramian (W_O) as follows:

$$W_O = \sum_{d=1}^{n_d} \frac{1}{n_d \; c_d^2} \int_0^{t_f^{id}} T \; \Psi^d(t) \; T^T \; dt \tag{3.6}$$

where t_f^{id} is the time to steady state for experiment id and $\Psi^d(t)$ is an $\tilde{n} \times \tilde{n}$ matrix with the ij element defined as:

$$\Psi_{ij}^d(t) = (y^{id}(t) - y^{id}(t_f^{id}))^T \; (y^{jd}(t) - y^{jd}(t_f^{jd})) \tag{3.7}$$

The observability gramian has the structure

$$W_O^{\tilde{n} \times \tilde{n}} = \begin{bmatrix} W_X^{n \times n} & W_{X\theta}^{n \times m} \\ W_{\theta X}^{m \times n} & W_\theta^{m \times m} \end{bmatrix} \tag{3.8}$$

where W_θ is the identifiability gramian of the parameters. The eigenvalues of W_θ provide information about the identifiability of the parameters with smaller eigenvalues indicating lower sensitivity. Parameters with significant components along eigenvectors of eigenvalues below a certain threshold will not be identifiable. This threshold will be zero in an ideal case but in reality will always be non-zero due to numerical errors (Geffen, 2008). The threshold can be approximated as the Frobenius-norm of E where E is found by linearizing the system about its nominal point and taking the difference between the empirical observability gramian and analytical observability gramian of this linearized system.

Two CSTR example

The results of the sensitivity tests for the two CSTR example are summarized in Table 3.2. Step tests were carried out by increasing parameters by 20% from their nominal values one at a time. Gramian tests were conducted by changing parameters one at a time by ±10% and ±20% of their nominal values.

Table 3.2: Results of parameter sensitivity analysis using step tests and empirical observability gramians for two CSTRs

	Step test (Sum of absolute normalized gains)	Empirical Gramian (Normalized eigenvalues)
F_0	0.004	7.150×10^{-9}
F_r	0.002	7.507×10^{-8}
F_3	0.014	5.810×10^{-6}
k_1	2.578×10^{-6}	6.032×10^{-11}
V_1	0.001	4.443×10^{-8}
V_2	0.002	2.278×10^{-9}
T_0	0.404	0.002
T_{03}	1.964	1.000

Both empirical gramian and step test indicate that F_0, F_r, F_3, k_1, V_1 and V_2 are not sensitive and should be removed from $\mathbb{S}$. Removing these parameters results in $\mathbb{S} = \{T_0, T_{03}\}$.

3.4.3 Augmented model creation

The first step to creating augmented models is to determine how many states can be added to each model. For deterministic linear systems operating at steady state with mismatch in only the A matrix, two assumptions must hold for unbiased output estimates to guarantee unbiased state estimates. These assumptions are derived below.

Consider a stable linear plant (A_{plant} has only negative eigenvalues):

$$\dot{x}_{plant} = A_{plant}x_{plant} + Bu \tag{3.9a}$$

$$\dot{x}_{plant} = (A_{model} + \Delta A)(x_{model} + \Delta x) + Bu \tag{3.9b}$$

$$y_{plant} = Cx_{plant} \tag{3.9c}$$

where ΔA and Δx represent the model plant mismatch in the A matrices and states respectively.

As there is only mismatch in A_{plant}, the model of this plant is given by:

$$\dot{x}_{model} = A_{model}x_{model} + Bu \tag{3.10a}$$

$$y_{model} = Cx_{model} \tag{3.10b}$$

For a given constant nonzero input u^s, the system will converge to a nonzero steady state. At this steady state, we can check if output estimates are unbiased by directly comparing the values of y^s_{plant} and y^s_{model}. If we have unbiased output estimates, we can derive the following:

$$y^s_{plant} = y^s_{model} \tag{3.11a}$$

$$Cx^s_{plant} = Cx^s_{model} \tag{3.11b}$$

$$C(x^s_{model} + \Delta x^s) = Cx^s_{model} \tag{3.11c}$$

$$C\Delta x^s = 0 \tag{3.11d}$$

The following is also true at this steady state as the B and u are known exactly:

$$A_{plant}x^s_{plant} = -Bu^s \tag{3.12}$$

$$A_{model}x^s_{model} = -Bu^s \tag{3.13}$$

From this we can derive the following:

$$A_{plant}x^s_{plant} = A_{model}x^s_{model} \tag{3.14a}$$

$$(A_{model} + \Delta A)(x^s_{model} + \Delta x^s) = A_{model}x^s_{model} \tag{3.14b}$$

$$A_{model}\Delta x^s + \Delta A x^s_{model} + \Delta A \Delta x^s = 0 \tag{3.14c}$$

For any row where the model matches the plant perfectly, that row of ΔA will be zero. The minimum number of zero rows in ΔA will occur when the mismatched parameter(s) and augmented parameter(s) are on different rows. This minimum (n_{min}) is equal to the number of rows (n) minus the number of mismatched parameters and augmented parameters ($n_{mis} + r$), i.e. $n_{min} = n - (n_{mis} + r)$.

Let $A_{model,i}$ representing row i of A_{model} for $i = 1, \ldots, n$. In order to satisfy (3.14c), $A_{model,i}\Delta x^s = 0$ for any row of A_{model} corresponding to a zero row of ΔA.

Considering (3.11d) and the above statements we obtain:

$$\underbrace{\begin{bmatrix} A_{model,i} \\ C \end{bmatrix}}_{T} \Delta x^s = 0 \tag{3.15}$$

for i = {Rows of A_{model} corresponding to zero rows of ΔA}.

If T is full column rank, the only solution is $\Delta x^s = 0$ which means we have achieved unbiased state estimation. T can only be full column rank if the number of rows $(n - (n_{mis} + r) + q)$ is greater than or equal to the number of columns (n).

This provides us with some additional assumptions needed to guarantee unbiased state estimation if unbiased output estimates are obtained:

Assumption 3.4.1 *The number of outputs must be greater than or equal to the number of mismatched parameters plus the number of augmented parameters i.e.* $q \geq n_{mis} + r$.

Assumption 3.4.2 *As the location of the mismatch is not known,* T *must be full column rank for ALL possible combinations of* n_{min} *rows of* A_{model}.

It is important to note that these assumptions do not guarantee the mismatched parameter has been identified correctly but only that state estimates are unbiased.

Assumption 3.4.1 can be checked for the stochastic nonlinear systems we consider directly while assumption 3.4.2 can be checked using linearized system models. Although not proven, it can be seen from simulation results that the mismatch can be correctly identified if these two assumptions (and the ones presented in Chapter 3.2) hold.

Once the number of parameters to augment has been decided, candidate models can be created using combinations of parameters from $\mathbb{S}$. For the two CSTR example, there are two outputs and we assume only one parameter is mismatched thus we can augment one parameter. This results in two augmented models: one augmented with T_0 and the other augmented with T_{03}.

3.4.4 Model observability

Given an observable linear system (A, C), the number of constant step disturbance states added must be less than or equal to the number of outputs to ensure that the resulting linear augmented system $(\tilde{A}, \tilde{C})$ is observable (Muske and Badgwell, 2002). In our algorithm, the added states are constant parameters thus the augmented model will be nonlinear even if the original system is linear. As a result, the above condition

does not guarantee observability of the augmented model and the observability of each augmented model must be checked before it can be added to the model set. Model observability is checked using the methods presented in Chapter 2.3.2. Any unobservable models should be removed from the model set.

3.4.5 State estimation

After designing the model set, state estimation should be carried out using all the models in the model set simultaneously. Any nonlinear model based state estimator can be used with common choices being the extended Kalman filter (EKF) or the moving horizon estimator (MHE). Appropriate constraints should be set on the augmented state vector if the estimator is capable of handling constraints.

3.4.6 Overall estimate

At each time step, the state estimates from the model with the highest probability are used.

Model probability is found according to the formula used in the Autonomous Multiple Model (AMM) algorithm which is as follows (Pitre, 2004):

$$L_{i,k} = \frac{\exp(-0.5 \cdot \epsilon_{i,k}^T \cdot S_{i,k}^{-1} \cdot \epsilon_{i,k})}{|2\pi S_{i,k}|^{1/2}} \tag{3.16a}$$

$$p_{i,k} = \frac{p_{i,k-1} L_{i,k}}{\sum_{j=1}^{t} p_{j,k-1} L_{j,k-1}} \tag{3.16b}$$

where $L_{i,k}$ represents the likelihood of model i at time k, $p_{i,k}$ represents the probability of model i at time k and t is the number of models in the model set. $\epsilon_{i,k}$ represents the one step ahead measurement prediction error of model i at time k with $S_{i,k}$ representing the covariance of this error.

An artificial lower limit δ is set on model probabilities with any probability that drops below δ set to δ. This prevents models from becoming inactive as the recursive nature of the probability calculation means that if a model probability drops to zero then it cannot be nonzero in future time steps (Kuure-Kinsey and Bequette, 2010).

If $S_{i,k}$ is not available due to the choice of estimator, an alternate option for calculating model probabilities is to use the following equation for model likelihood

(Kuure-Kinsey and Bequette, 2010):

$$L_{i,k} = \exp(-0.5 \cdot \epsilon_{i,k}^T \cdot \Delta \cdot \epsilon_{i,k}) \tag{3.17}$$

where Δ is a tuning parameter that is the same for all models in the model set.

The AMM algorithm is used over newer MM algorithms such as Generalized Pseudo Bayesian (GPB) and Interacting Multiple Models (IMM) for the following reasons:

1. AMM assumes the true system does not change over time while GPB/IMM assume that the true system changes over time according to a Markov or semi-Markov chain. In our case the true system parameters and structure are constant with respect to time.

2. GPB/IMM fuse the estimates obtained from the different models which is only possible if all the models share the same states.

If assumptions 3.2.1-3.2.3, 3.4.1 and 3.4.2 are satisfied there are two possible options when comparing model probabilities:

1. There is ONE clear winner and no output bias. This indicates that the mismatch has been identified correctly and occurs when the number of mismatched parameters (n_{mis}) is equal to the number of added states (r).

2. There are MANY winners and no output bias. This occurs when the number of mismatched parameters (n_{mis}) is less than the number of added states (r). All models which contain the mismatched parameters will have similar (or equal) probabilities.

If any of the assumptions are violated, there are two possible options regardless of the number of winning models:

1. There is output bias in the winning model(s). This indicates that there is no augmented model that fully captures the mismatch and there will be bias in the state estimates.

2. There is no output bias in the winning model(s). This does not provide any meaningful information as there is no guarantee that state estimates are not biased or that the mismatched parameter(s) have been identified correctly.

For our two CSTR example, let us consider the case where the actual $T_0 = 320$ K while the value used in the model is $T_0 = 300$ K. Figure 3.4 shows the trajectory of the actual states and the estimates given by EKFs using a model augmented with T_0 and a model augmented with T_{03}. The initial state for both EKFs is the nominal steady state and nominal parameter values ($T_0 = 300$ K). Assumptions 3.2.1-3.2.3, 3.4.1 and 3.4.2 are satisfied thus it can be seen that only the correct model (T_0) removes both output and estimation bias. Figure 3.5 shows the trajectory of the T_0 and T_{03} model probabilities obtained using the proposed algorithm. It can be seen that the correct model is picked.

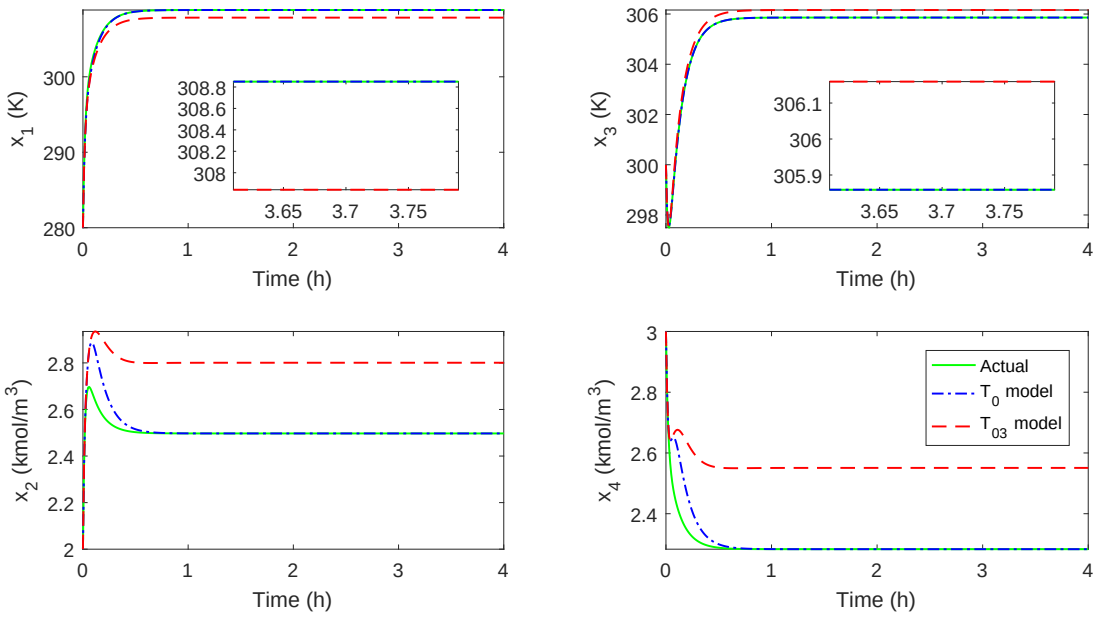

Figure 3.4: Trajectory of the actual states and estimates given by an EKF using a model augmented with T_0 and a model augmented with T_{03} with model plant mismatch in T_0.

3.5 Application to a froth flotation system

The proposed algorithm is tested on the froth flotation system presented in Chapter 2.2. Model parameters used are the same as in Chapter 2.4.5 except for the volumetric

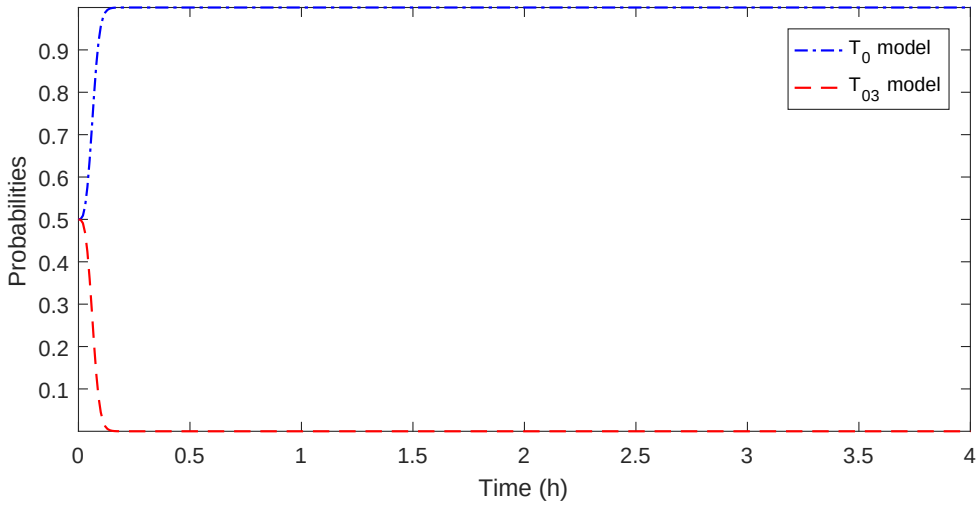

Figure 3.5: Trajectory of T_0 and T_{03} model probabilities obtained using the proposed algorithm with model plant mismatch in T_0.

feed flow to the first tank ($\dot{V}_{uf_0}$) which is constant instead of time varying. The volumetric flow of feed liquid to the first tank is constant at 20 m^3/min while the volumetric flow of feed solids (and total feed) depends on the mass fraction of solids in the feed (c_{st_0}).

We select $\mathbb{S} = \{c_{st_0}, \beta, k, c_\infty, c_{at_0}\}$. Mass fraction of solids in feed is selected because it is not controlled and can vary between $0 - 30\%$ (Canright *et al.*, 1981). β is selected arbitrarily and mismatch in β could arise as a results of problems with the motors driving the froth removal paddles. k and c_∞ are selected because they are calculated by extrapolating from batch data and depend on the coal type thus are highly likely to contain mismatch. Mass fraction of ash in the feed is selected because it depends on the coal type and can vary between samples of the same coal type.

Sensitivity analysis was carried out on all the identified parameters and the results are summarized in Table 3.3. Step tests were carried out by increasing parameters by 20% from their nominal values one at a time. Gramian tests were conducted by changing parameters one at a time by $\pm 10\%$ and $\pm 20\%$ of their nominal values.

Based on these results, we removed β from $\mathbb{S}$ resulting in $\mathbb{S} = \{$solids fraction in feed (c_{st_0}), k, c_∞, ash fraction in feed (c_{at_0})$\}$.

Models were augmented with one parameter resulting in four augmented models: c_{st_0} model, k model, c_∞ model and c_{at_0} model. Simulations were carried out to test

Table 3.3: Results of parameter sensitivity analysis using step tests and empirical observability gramians for froth flotation

	Step test (Sum of absolute normalized gains)	Empirical Gramian (Normalized eigenvalues)
c_{st_0}	0.619	0.010
β	0.059	1.105×10^{-5}
k	0.348	0.001
c_∞	0.324	0.003
c_{at_0}	1.478	1

the efficacy of the algorithm using these models with model plant mismatch introduced by setting the actual c_{at_0} to 31.85% of c_{st_0} (1.3 times the nominal value of 24.5% of c_{st_0}). Gaussian measurement noise was generated as $v \backsim [N(0,1), N(0,1)]^T$. Gaussian process noise was generated for each tank i, $i = 1, \ldots, 5$, as $w_i(t) \backsim [N(0,1), N(0,1), N(0,1)]^T$. Small noise values were used to avoid violating assumption 3.2.3. The probability threshold δ was set as $\delta = 0.001$.

Figure 3.6 shows the trajectory of the actual states and the estimates given by an EKF using a k model, c_∞ model and c_{at_0} model for the last tank (tank 5) with model plant mismatch in c_{at_0}. Estimates from the c_{st_0} model are not shown because the c_{st_0} model is not observable along the entire trajectory. Figure 3.7 shows the trajectory of the c_{st_0} model, k model, c_∞ model and c_{at_0} model probabilities using the proposed algorithm with model plant mismatch in c_{at_0}. Although the wrong model (c_∞ model) is picked initially, the algorithm eventually picks the correct model (c_{at_0} model). This is due to the initial lack of separation between the c_∞ and c_{at_0} models based on noisy output residuals. The separation between these two models increases as the plant approaches steady state thus the correct model is picked when the plant is closer to steady state. It can also be seen that only the correct model (c_{at_0}) model is capable of removing bias in the c_{st} estimates.

Simulations were also carried out to test the case where some of assumptions 3.2.1-3.2.3, 3.4.1 and 3.4.2 were violated. The augmented models, measurement noise and process noise were the same as in the previous case but model plant mismatch was introduced by setting the actual c_{st_0} and actual c_{at_0} to be 1.3 times their nominal values. In this situation assumptions 3.4.1 and 3.4.2 are violated. Figure 3.8 shows the trajectory of the actual states and the estimates given by an EKF using a k

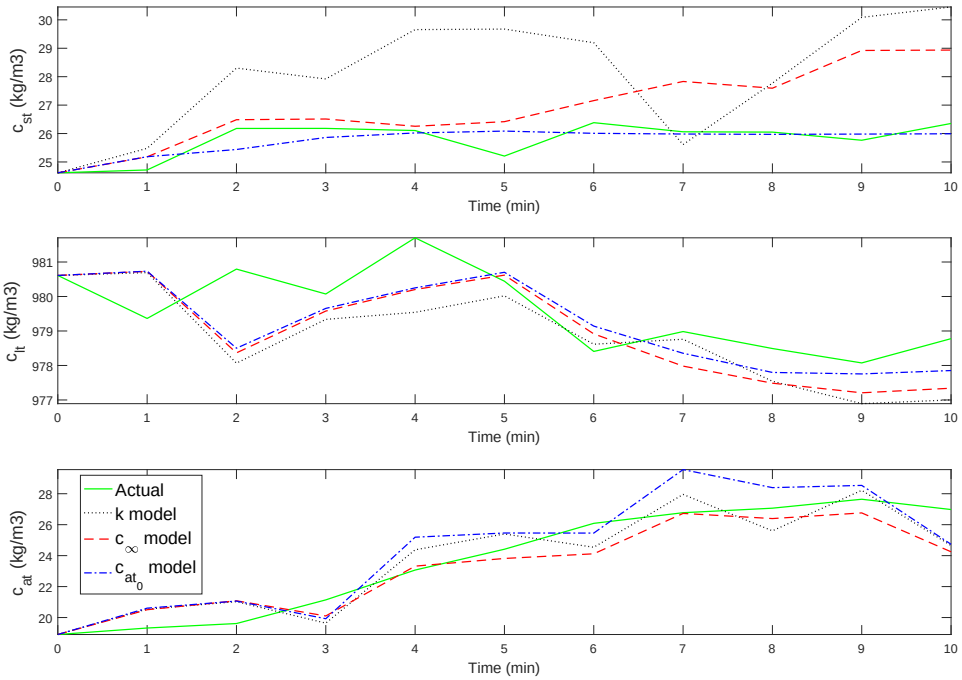

Figure 3.6: Trajectory of the actual states and the estimates given by an EKF using a k model, c_∞ model and c_{at_0} model for the last tank (tank 5) with model plant mismatch in c_{at_0}.

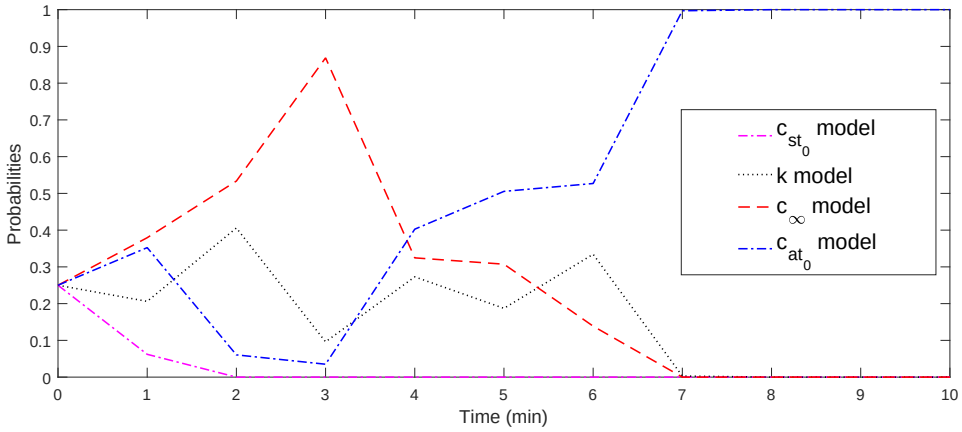

Figure 3.7: Trajectory of the c_{st_0} model, k model, c_∞ model and c_{at_0} model probabilities obtained using the proposed algorithm with model plant mismatch in c_{at_0}.

model, c_∞ model and c_{at_0} model for the last tank (tank 5) in this case. Estimates obtained using the c_{st_0} model are not shown as the c_{st_0} model does not converge. Figure 3.9 shows the trajectory of the c_{st_0} model, k model, c_∞ model and c_{at_0} model probabilities using the proposed algorithm in this case. It can be seen that although one model (c_{at_0}) is the clear winner it does not remove bias in the c_{st} estimates. As assumptions 3.4.1 and 3.4.2 are violated, unbiased output estimates do not guarantee unbiased state estimates.

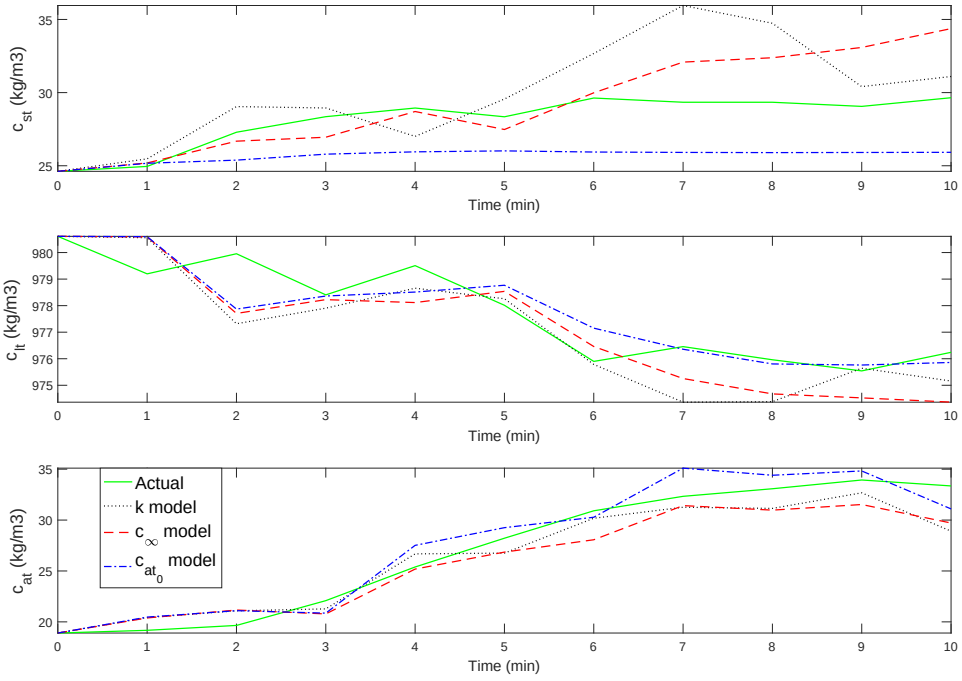

Figure 3.8: Trajectory of the actual states and the estimates given by an EKF using a k model, c_∞ model and c_{at_0} model for the last tank (tank 5) with model plant mismatch in c_{at_0} and c_{st_0}.

3.6 Conclusions

In this chapter, a method for obtaining unbiased state estimates in the presence of parameter mismatch was detailed. Necessary assumptions for the success of the method were provided. Proof was given for the simple case of deterministic linear systems operating at steady state. The algorithm was shown to work on a two CSTR system and the froth flotation system when the necessary assumptions were met.

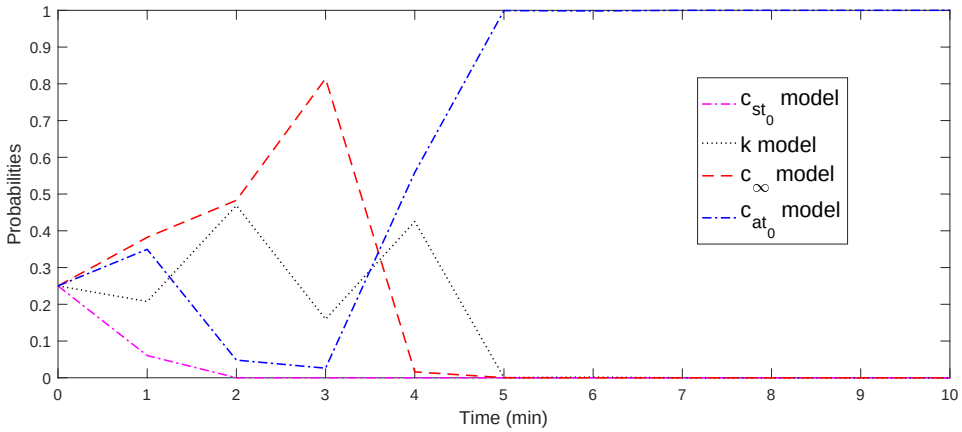

Figure 3.9: Trajectory of the c_{st_0} model, k model, c_∞ model and c_{at_0} model probabilities obtained using the proposed algorithm with model plant mismatch in c_{at_0} and c_{st_0}.

Chapter 4

Control of the froth flotation system

4.1 Introduction

Model predictive control (MPC) is a highly popular control strategy in industry due to its ability to handle constraints on the control and states (Mayne *et al.*, 2000). In MPC, an optimization problem is solved at each time step to determine the optimal input sequence over a horizon while taking into account system dynamics, state constraints and input constraints (Mayne *et al.*, 2000). Only the first element of this input sequence is applied to the system. MPC tries to move the system to a target steady state that is picked to optimize plant economics. This target is determined in a separate layer called the real-time optimization (RTO) layer by solving an economic optimization problem.

Recently, an alternative control strategy known as economic model predictive control (EMPC) has been gaining attention (Liu and Liu, 2016). Like MPC, EMPC can handle constraints on the control and states. It too solves an optimization problem at each time step to determine the optimal input sequence over a horizon with only the first element of this input sequence applied to the system. However unlike MPC, EMPC does not try to move the system to a target steady state. It instead aims to optimize plant economics by solving an economic optimization problem directly to calculate the input sequence.

A known issue with MPC is its inability to track a reference signal without offset if there is model plant mismatch (MPM). One approach to offset free reference tracking

is provided in Morari and Maeder, 2012. Their approach involves augmenting the system model with a disturbance model to capture the effects of the mismatch. The augmented model is then used to calculate new MPC targets and input sequences. There are many choices for the disturbance model and the disturbance model does not need to accurately model the mismatch for this approach to succeed.

An alternate approach to offset free reference tracking is to remove the model plant mismatch by updating the model then using the updated model to calculate the input sequence. This approach can be considered a model identification (ID) based approach. There are many options for model identification but in this work we consider the MM approach presented in Chapter 3. This ID based approach offers the potential of better transient performance than the approach in Morari and Maeder, 2012 as the updated model should reflect the true plant dynamics better than the arbitrary disturbance model picked in Morari and Maeder, 2012.

In this chapter, we present the details of controller design. A MPC and EMPC are designed for use in the nominal case (no model plant mismatch and accurate state estimates available). A nonlinear offset free MPC as proposed by Morari and Maeder, 2012 and a model identification based approach are designed for use when there is parameter mismatch between the model and the plant.

4.2 Preliminaries

The plant and nominal model used in this chapter are defined in Chapter 3.2 as (3.1) and (3.2) respectively. In this chapter, the outputs are obtained at discrete sampling intervals and inputs are piecewise constant. Plant parameters (θ^*) and model parameters (θ) are assumed to be time invariant.

4.3 Model predictive control (MPC)

The optimization problem solved by the MPC used in this work takes the following form (Mayne *et al.*, 2000):

$$\min_{u(t_k),\ldots,u(t_{k+N})} \sum_{i=k}^{k+N} (\check{x}(t_i) - x^s)^T Q (\check{x}(t_i) - x^s) + \sum_{i=k}^{k+N} (u(t_i) - u^s)^T R (u(t_i) - u^s) \quad (4.1a)$$

$$\text{s.t} \qquad \dot{\check{x}}(t) = f(\check{x}(t), \theta, u(t), 0) \quad (4.1b)$$

$$\check{x}(t_k) = x(t_k) \tag{4.1c}$$

$$\check{x}(t) \in \mathbb{X}, u(t) \in \mathbb{U} \tag{4.1d}$$

where N is the horizon length, Q and R are weighting matrices on the state and input deviations respectively, $\mathbb{X}$ is the set of all possible values of x and $\mathbb{U}$ is the set of all possible values of u. x^s is the target for the system steady state with u^s the associated input target. $\check{x}$ is the predicted state trajectory based on the initial state $x(t_k)$ and model (4.1b). Only the first element of the calculated input sequence $(u(t_k))$ is applied to the system and the optimization problem is solved at the next time step to calculate a new input sequence.

4.4 Economic model predictive control (EMPC)

The EMPC combines economic optimization and control of the system by solving an economic cost function instead of the MPC cost function (4.1) to calculate the input sequence (Ellis *et al.*, 2014). The optimization problem solved by the EMPC takes the following form (Liu *et al.*, 2015; Liu and Liu, 2016):

$$\min_{u(t_k),\ldots,u(t_{k+N})} \sum_{i=k}^{k+N} l(\check{x}(t_i), u(t_i)) + c(\check{x}(t_{k+N}), N_h) \tag{4.2a}$$

$$\text{s.t} \qquad \dot{\check{x}}(t) = f(\check{x}(t), \theta, u(t), 0) \tag{4.2b}$$

$$\check{x}(t_k) = x(t_k) \tag{4.2c}$$

$$\check{x}(t) \in \mathbb{X}, u(t) \in \mathbb{U} \tag{4.2d}$$

As with the MPC, N is the horizon length, $\mathbb{X}$ is the set of all possible values of x and $\mathbb{U}$ is the set of all possible values of u. $\check{x}$ is the predicted state trajectory based on the initial state $x(t_k)$ and model (4.2b). Only the first element of the calculated input sequence $(u(t_k))$ is applied to the system and the optimization problem is solved at the next time step to calculate a new input sequence. The $l(\check{x}, u)$ term is the economic cost function that describes the economic cost of the system at a given time. The $c(\check{x}(t_k), N_h)$ term is known as the terminal cost and summarizes the economic performance of the system under an asymptotically stabilizing controller $u = k(x)$

over the terminal horizon N_h as follows:

$$c(x(t_k), N_h) = \sum_{i=k}^{k+N_h} l(\check{x}(t_i), u(t_i)) \tag{4.3}$$

The terminal cost is used to improve closed loop stability of the EMPC (Liu *et al.*, 2015; Liu and Liu, 2016).

4.5 Offset free model predictive control

The offset free MPC presented in Morari and Maeder, 2012 is one approach to offset free output tracking in the presence of MPM. While their approach is developed for discrete systems, it is applicable to the continuous systems we consider due to our restrictions that the inputs are piecewise constant and the outputs are obtained at discrete sampling intervals. The procedure is described below.

The nominal model (3.2) is augmented with disturbance states to capture the effects of the parameter mismatch. The augmented model does not need to model the real mismatch accurately in order to achieve offset free control thus the disturbance states do not need to be system parameters. This results in an augmented model of the form:

$$\dot{\tilde{x}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{d}(t) \end{bmatrix} = \begin{bmatrix} f_{aug}(x(t), \theta, u(t), 0, d(t)) \\ 0 \end{bmatrix} \tag{4.4a}$$

$$y(t) = g_{aug}(x(t), d(t)) \tag{4.4b}$$

where $\tilde{x}$ represents the augmented state vector ($\tilde{x} \in \mathbb{R}^{\tilde{n}}$) consisting of the system states ($x \in \mathbb{R}^n$) and the disturbance states ($d \in \mathbb{R}^r$). θ is the vector of system parameters ($\theta \in \mathbb{R}^m$).

This model is used with a nonlinear estimator such as an extended Kalman filter to obtain estimates of the states and disturbance at each time instant (denoted by $\hat{x}(t_k)$ and $\hat{d}(t_k)$ respectively).

For a given reference signal y_{ref}, the following problem is solved at each time step:

$$0 = f_{aug}(\bar{x}, \theta, \bar{u}, 0, \hat{d}(t_k)) \tag{4.5a}$$

$$y_{ref} = g_{aug}(\bar{x}, \hat{d}(t_k)) \tag{4.5b}$$

where $\bar{x} \in \mathbb{X}$ and $\bar{u} \in \mathbb{U}$ are the state and input targets used to calculate the input sequence using a standard MPC formulation as follows:

$$\min_{u(t_k),\ldots,u(t_{k+N})} \sum_{i=k}^{k+N} (\check{x}(t_i) - \bar{x})^T Q(\check{x}(t_i) - \bar{x}) + \sum_{i=k}^{k+N} (u(t_i) - \bar{u})^T R(u(t_i) - \bar{u}) \tag{4.6a}$$

$$\text{s.t} \quad \dot{\check{x}}(t) = f_{aug}(\check{x}(t), \theta, u(t), 0, \hat{d}(t_k)) \tag{4.6b}$$

$$\check{x}(t_k) = \hat{x}(t_k) \tag{4.6c}$$

$$\check{x}(t) \in \mathbb{X}, u(t) \in \mathbb{U} \tag{4.6d}$$

The first element of the calculated input sequence ($u(t_k)$) is applied to the system and the procedure is repeated at the next time instant.

This method guarantees offset free reference tracking when the following assumptions are met:

Assumption 4.5.1 *The reference signal (y_{ref}) and parameter mismatch are constant.*

Assumption 4.5.2 *The number of disturbance states (r) is equal to the number of inputs (p) which is equal to the number of outputs (q). $r = p = q$.*

Assumption 4.5.3 *For the augmented model (4.4) there exist a $x^* \in \mathbb{X}$ and $d^* \in \mathbb{D}$ for all $y \in \mathbb{Y}$ and $u \in \mathbb{U}$ such that*

$$x^* = f_{aug}(x^*, \theta, u, 0, d^*) \tag{4.7a}$$

$$y = g_{aug}(x^*, d^*) \tag{4.7b}$$

with (x^*, d^*) a unique solution to (4.7) for a given (u, y).

Assumption 4.5.4 *For the augmented model (4.4) there exist a $x^* \in \mathbb{X}$ and $u^* \in \mathbb{U}$ for all $y_{ref} \in \mathbb{Y}$ and $d \in \mathbb{D}$ such that*

$$x^* = f_{aug}(x^*, \theta, u^*, 0, d) \tag{4.8a}$$

$$y_{ref} = y = g_{aug}(x^*, d) \tag{4.8b}$$

with (x^*, u^*) a unique solution to (4.8) for a given (y_{ref}, d).

Assumption 4.5.5 *The observer is nominally error free at steady state.*

Assumption 4.5.6 *The controller is nominally error free at steady state for all* $d \in \mathbb{R}$ *and* $y_{ref} \in \mathbb{Y}$ *which yield strictly feasible targets* $(\bar{x}, \bar{u})$.

The advantage of this approach over the model identification (ID) based approach presented in the next section is that the augmented model does not need to model the real mismatch accurately.

4.6 Model ID based approach to offset free control

This approach aims to achieve offset free reference tracking by updating the model to accurately reflect the real plant. Model update is carried out using the MM method outlined in Chapter 3. The updated model and estimates are used with the MPC formulation:

$$\min_{u(t_k),\ldots,u(t_{k+N})} \sum_{i=k}^{k+N} (\check{y}(t_i) - y_{ref})^T Q (\check{y}(t_i) - y_{ref}) \tag{4.9a}$$

$$\text{s.t} \quad \dot{\check{x}}(t) = \tilde{f}(\check{x}(t), \tilde{\theta}, u(t), 0) \tag{4.9b}$$

$$\check{y}(t) = \tilde{g}(\check{x}) \tag{4.9c}$$

$$\check{x}(t_k) = \tilde{x}(t_k) \tag{4.9d}$$

$$\check{x}(t) \in \tilde{\mathbb{X}}, u(t) \in \mathbb{U} \tag{4.9e}$$

where the model used in the MPC is the model with the highest probability according to the MM method in Chapter 3. The success of this approach depends on the success of the model identification stage thus assumptions 3.2.1-3.2.3,3.4.1 and 3.4.2 must be satisfied. It is also assumed that the reference signal y_{ref} is constant. This method offers the potential of better transient performance than the offset free MPC presented in the previous section as the best model will accurately reflect the actual plant dynamics. The better the model represents the true plant, the better the transient performance (Morari and Maeder, 2012).

4.7 Application to the froth flotation system

4.7.1 Without model plant mismatch

Simulation settings

The simulation parameters used are the same as in Chapter 2.4.5 with the exception of the volumetric flow of feed to the first tank ($\dot{V}_{uf_0}$) which is constant at 21.71 m^3/min. There is no process or measurement noise ($w = v = 0$) and the states are directly available to the controllers.

In the MPC, $Q = \text{diag}([1, 1, 2, 1, 1, 2, 1, 1, 2, 1, 1, 2, 1, 1, 2])$ and $R = \text{diag}([0.5, 0.5]^T)$ where $\text{diag}(v)$ represents a diagonal matrix whose diagonal elements are the elements of a vector v. The control horizon for the MPC and EMPC is $N = 5$ and terminal window sizes of $N_h = 10, 15$ are used in the EMPC. As the froth flotation system is open loop stable, the controller used in the EMPC terminal cost takes the form $u(t) = [0.125, 0.3125]^T$.

The economic cost function used in the EMPC takes the form $l(x, u) =$ (Cost of $u \times u$) - (Price of coal $\times$ coal recovered). Input prices were obtained from ICIS Chemical Business magazine and coal prices were obtained from the U.S. Energy Information Administration website. This cost function was solved to obtain the economic optimum steady state which was used as the state and input targets in the MPC. The profit at t_k is simply $-l(x(t_k), u(t_k))$.

Simulation results

Figure 4.1 shows the trajectory of states in the last tank (tank 5) with an MPC and EMPC under these simulation settings. It can be seen that all three controllers are stable. Figure 4.2 shows the profit at each time step t_k with the three controllers. It can be seen that the economic performance of all three is similar. This result is not surprising as there is no guarantee that the EMPC has better economic performance than the MPC.

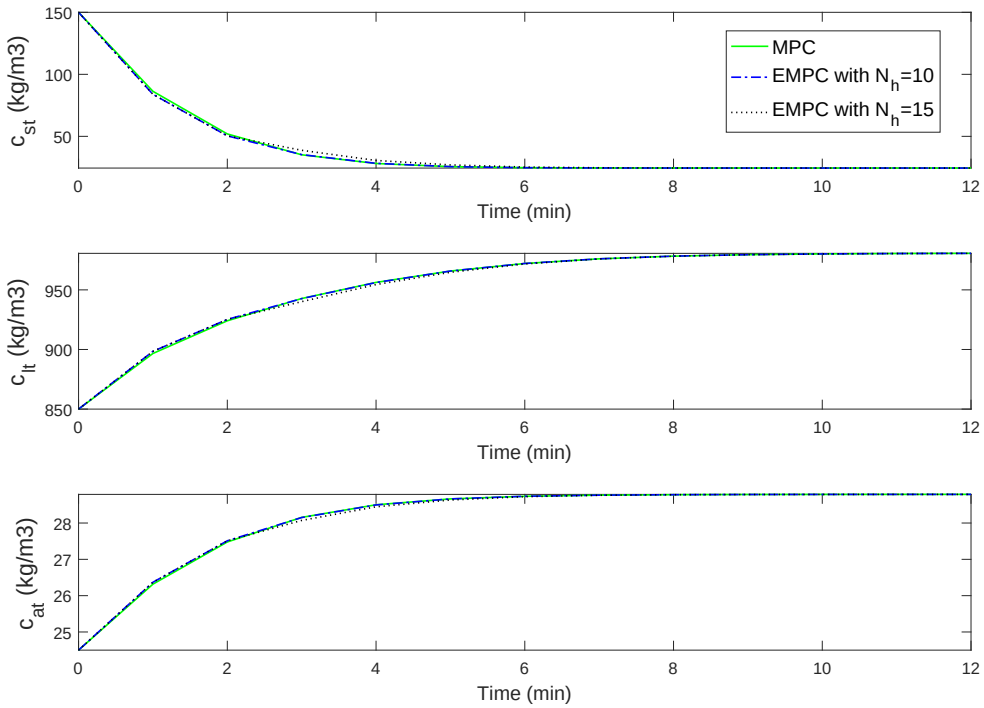

Figure 4.1: Trajectory of the states in tank 5 using an MPC with $N = 5$ and EMPC with $N = 5$ and $N_h = 10, 15$ when states are directly available to the controllers and no MPM

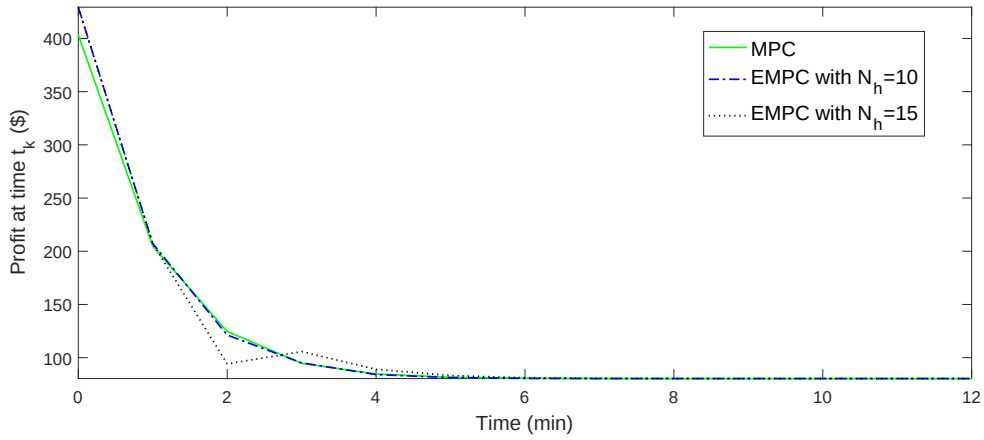

Figure 4.2: Profit at each step using an MPC with $N = 5$ and EMPC with $N = 5$ and $N_h = 10, 15$ when states are directly available to the controllers and no MPM

4.7.2 With model plant mismatch

Simulation settings

The simulation parameters used are the same as in Chapter 2.4.5 with the exception of the volumetric flow of feed to the first tank ($\dot{V}_{uf_0}$) which is constant instead of time varying. MPM was introduced by setting the plant $c_{st_0} = 0.2$ instead of the nominal model value of $c_{st_0} = 0.1$. This results in $\dot{V}_{uf_0} = 23.85$ m^3/min. There is no process or measurement noise ($w = v = 0$).

For the offset free MPC approach, $Q = \text{diag}([1, 1, 2, 1, 1, 2, 1, 1, 2, 1, 1, 2, 1, 1, 2])$ and $R = \text{diag}([0.5, 0.5]^T)$. Only output disturbances were used in the augmented model resulting in an augmented model of the form:

$$\dot{\tilde{x}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{d}(t) \end{bmatrix} = \begin{bmatrix} f_{aug}(x(t), \theta, u(t), 0)) \\ 0 \end{bmatrix} \tag{4.10a}$$

$$y(t) = g_{aug}(x(t), d(t)) \tag{4.10b}$$

For the model ID based approach, the model set used contained a c_{st_0} model and a c_{at_0} model. A k model and c_∞ model were not used as k and c_∞ depend on u thus will not be constant during closed loop operation.

For both approaches, an EKF was used to obtain state estimates.

In order to ensure feasibility of the reference signal, y_{ref} was calculated based on the true plant and was set to $y_{ref} = [968.5, 30.5]^T$ which corresponds to a plant input of $u = [0.08, 0.20]^T$.

Simulation results

Figure 4.3 shows the reference signal and the outputs obtained using the offset free MPC approach and the model ID based approach. It can be seen that the model ID based approach achieves offset free output tracking while the offset free MPC approach does not. The offset free MPC approach fails as assumption 4.5.4 is violated. During simulation, there are time steps where there is no value of $x \in \mathbb{X}$ and $u \in \mathbb{U}$ which satisfy (4.7) for the given value of $(y_{ref}, \hat{d})$. Figure 4.4 shows the associated inputs for this case.

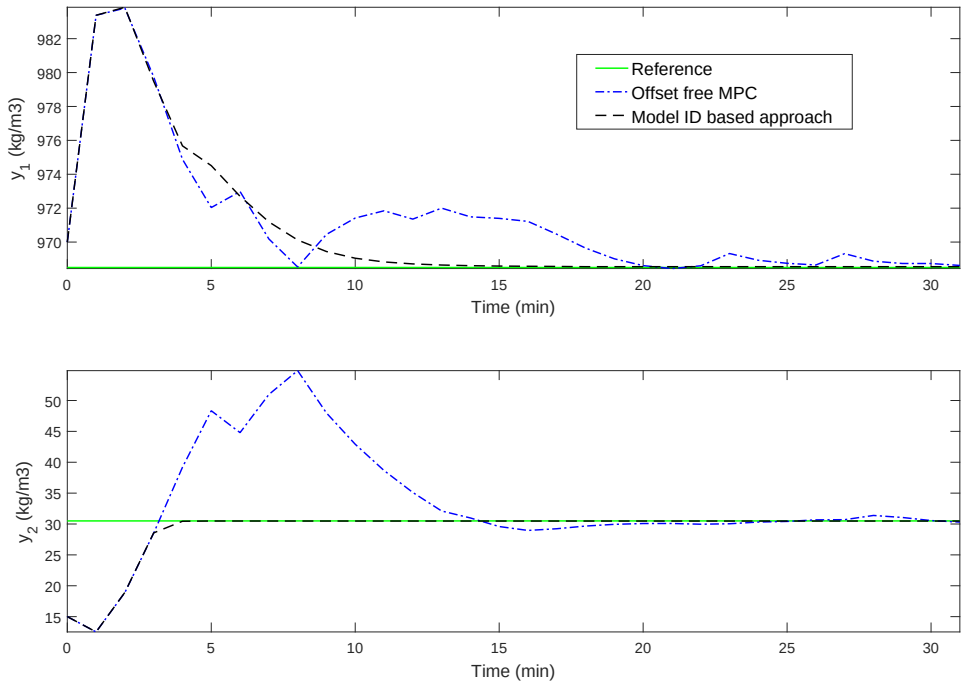

Figure 4.3: Trajectory of the outputs using an offset free MPC and model ID based approach in the presence of mismatch in c_{st_0}

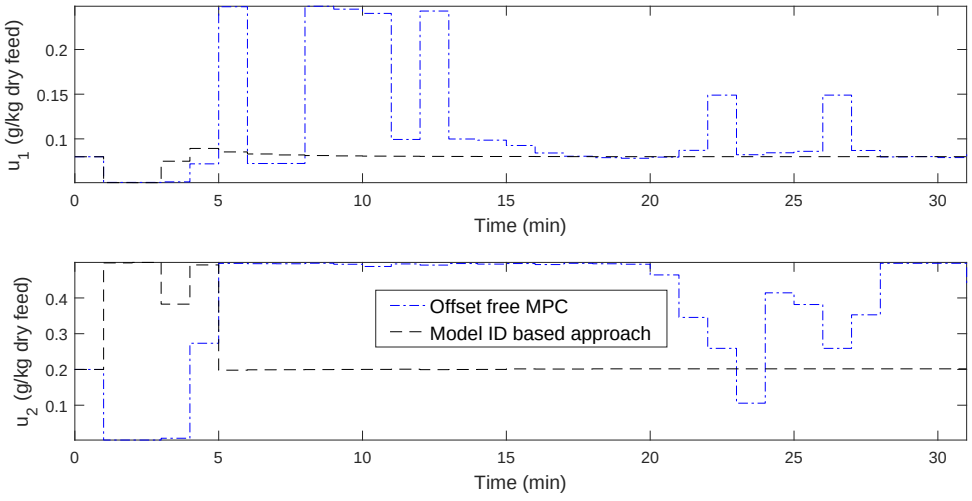

Figure 4.4: Trajectory of the inputs using an offset free MPC and model ID based approach in the presence of mismatch in c_{st_0}

4.8 Conclusions

In this chapter, MPC and EMPC were compared to see which gave better performance when there was no model plant mismatch and the states were directly available to the controllers. It was found that both controllers were stable and had similar economic performance even though the MPC does not have the explicit goal of optimizing economic performance unlike the EMPC. As there is no guarantee that the EMPC will have better economic performance than the MPC, this result is not surprising.

As conventional MPC does not provide offset free output tracking in the presence of MPM, an offset free MPC formulation and a model ID based approach to offset free control were investigated. Only MPM caused by parameter mismatch was considered. It was found that the offset free MPC could not provide offset free output tracking when using an output disturbance model as some assumptions necessary for the success of this method were not met. It was also found that the model ID based approach could successfully track the reference signal without offset.

Chapter 5

Conclusions

5.1 Summary

The goal of this thesis was to test modern control methods on a froth flotation system. As state estimation plays a crucial role in the success of modern control methods, various state estimators were also tested on this system. A first principles model of the froth flotation system as developed by Canright *et al.*, 1981 was used in simulations.

In Chapter 2, an extended Kalman filter (EKF), moving horizon estimator (MHE) and nonlinear observer were tested on the froth flotation system. It was found that the EKF gave the lowest estimation error in the ideal case (no model plant mismatch and no delay in measurements). Although MHE has the potential give better performance than the EKF, it did not do so in this case due to the small size of the estimation window which was necessary due to the large number of states in the froth flotation process. The nonlinear observer performed the worst as it did not use any noise information. All three methods had similar performance when there was model plant mismatch as the MPM masked the noise information which nullified the advantage of the EKF and MHE over the observer. It was also found that state estimates were biased in the presence of MPM. All three methods had similar performance when measurement were delayed (both with and without model plant mismatch). As measurement delays were large, state prediction dominated over state estimation in all three estimators. As the froth flotation system is open loop stable, all three estimators converged to the same estimates. The EKF was picked as the best estimator for this system as it gave the best balance of estimation performance and computation time.

In Chapter 3, an approach to unbiased state estimation in the presence of param-

eter mismatch was presented. This approach used multiple models (MM) running in parallel and assigned probabilities to each model based on their one step ahead output prediction error. The state estimate at each time was obtained from the model with the highest probability. It was proved that unbiased output estimates guaranteed unbiased state estimates for a deterministic linear system operating at steady state provided certain conditions were met. A flowchart of the procedure was provided and the procedure was successfully applied to the froth flotation system to obtain unbiased state estimates.

Various controllers were tested in Chapter 4. A model predictive controller (MPC) and economic model predictive controller (EMPC) were tested for the nominal case of no model plant mismatch and accurate state estimates available. The economic optimum steady state was found by solving the economic cost function used in the EMPC and set as the state and input targets for the MPC. It was found that the MPC and EMPC both stabilized the system and had similar economic performance. As there is no guarantee that EMPC provides better economic performance, this result was not surprising. To solve the issue of offset free control in the presence of parameter mismatch, two methods were proposed: an offset free MPC as presented in Morari and Maeder, 2012 and a model identification based method which combined the state estimation method proposed in Chapter 3 with a conventional MPC. The performance of the two methods was tested when parameter mismatch was introduced to the system. The offset free MPC could not provide offset free control when using an output disturbance model as some necessary conditions were not met. The model ID based approach successfully provided offset free control. Although the model ID based approach worked in this case, it is not rigorously proved thus is not guaranteed to work in the general case even if the conditions for unbiased state estimation are met.

5.2 Directions for future work

Potential directions for future work are listed below:

- Apply state estimation methods not covered in this work to the forth flotation system. Some choices are unscented Kalman filters (UKF) and particle filters.

- Apply control strategies not covered in this work to the froth flotation system. Some choices are the zone MPC and chance constrained MPC.
- Extend the proof provided in Chapter 3 to determine conditions for which unbiased output estimates guarantee the parameter mismatch has been identified correctly and not just that states estimates are unbiased. The proof can also be extended to the nonlinear case.
- Extend the multiple model approach for estimation to cases where MPM is not only caused by parameter mismatch.
- Extend the model ID based approach for control to cases where the MPM is not only caused by parameter mismatch.

Bibliography

Amirzadeh, A., A. Karimpour and A. Moeini (2011). An IMM algorithm based on augmented Kalman filter for maneuvering target tracking. *Scientific Research and Essays* **6**(34), 6787–6797.

Bay, J. S. (1998). *Fundamentals of Linear State Space Systems*. McGraw-Hill.

Botelho, V., J. O. Trierweiler, M. Farenzena and R. Duraiski (2015). Methodology for detecting modelplant mismatches affecting model predictive control performance. *Industrial & Engineering Chemistry Research* **54**(48), 12072–12085.

Canright, G. S., C. H. Brown, G. O. Allgood and W. R. Hamel (1981). Dynamic modeling and control analysis of froth floatation and clean coal filtration as applied to coal benefictiation. Technical report. Oak Ridge National Laboratory. Oak Ridge, Tennessee 37830.

Chetouani, Y. (2008). Design of a multi-model observer-based estimator for Fault Detection and Isolation (FDI) strategy: application to a chemical reactor. *Brazilian Journal of Chemical Engineering* **25**(4), 777–788.

Chui, C. K. and G. Chen (1991). *Kalman Filtering: With Real-Time Applications*. Springer-Verlag.

Ciccarella, G., M. D. Mora and A. Germani (1993). A Luenberger-like observer for nonlinear systems.. *International Journal of Control* **57**(3), 537.

Einicke, G. A. (2012). *Smoothing, filtering and prediction: estimating the past, present and future*. Intech, Croatia.

Ellis, M., H. Durand and P. D. Christofides (2014). A tutorial review of economic model predictive control methods. *Journal of Process Control* **24**(8), 1156 – 1178.

Gauthier, J. P., H. Hammouri and S. Othman (1992). A simple observer for nonlinear systems: applications to bioreactors. *IEEE Transactions on Automatic Control* **37**, 875–880.

Geffen, D. (2008). Parameter identifiability of biochemical reaction networks in systems biology. Master's thesis. Queen's University. Kingston, Ontario, Canada.

Gopalakrishnan, A., N. S. Kaisare and S. Narasimhan (2011). Incorporating delayed and infrequent measurements in extended kalman filter based nonlinear state estimation.. *Journal of Process Control* **21**, 119 – 129.

Hargrave, J. M., N. J. Miles and S. T. Hall (1996). The use of grey level measurement in predicting coal flotation performance. *Minerals Engineering* **9**, 667 – 674.

Huang, D. and H. Leung (2005). An expectation-maximization-based interacting multiple model approach for cooperative driving systems. *IEEE Transactions on Intelligent Transportation Systems* **6**(2), 206–228.

Jiao, Z., W. Li, Q. Zhang and P. Wang (2015). Tracking for the maneuvering target based on multiple model and moving horizon estimation. In: *2015 IEEE International Conference on Information and Automation*. pp. 1936–1941.

Kuure-Kinsey, M. and B. W. Bequette (2010). Multiple model predictive control strategy for disturbance rejection. *Industrial & Engineering Chemistry Research* **49**(17), 7983–7989.

Lee, D. J., Y. Park and Y. s Park (2012). Robust H_∞ sliding mode descriptor observer for fault and output disturbance estimation of uncertain systems. *IEEE Transactions on Automatic Control* **57**(11), 2928–2934.

Li, X. R. (2002). Model-set design for multiple-model method. Part I. In: *Proceedings of the Fifth International Conference on Information Fusion*. Vol. 1. IEEE. pp. 26–33.

Liu, S. and J. Liu (2016). Economic model predictive control with extended horizon. *Automatica* **73**, 180 – 192.

Liu, S., J. Zhang and J. Liu (2015). Economic MPC with terminal cost and application to an oilsand primary separation vessel. *Chemical Engineering Science* **136**, 27–37.

Marino, R. and P. Tomei (1995). *Nonlinear Control Design: Geometric, Adaptive, and Robust*. Prentice-Hall.

Mayne, D. Q., J. B. Rawlings, C. V. Rao and P. O. M. Scokaert (2000). Constrained model predictive control: Stability and optimality. *Automatica* **36**, 789–814.

Morari, M. and U. Maeder (2012). Nonlinear offset-free model predictive control. *Automatica* **48**(9), 2059–2067.

Muske, K. R. and T. A. Badgwell (2002). Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control* **12**(5), 617–632.

Parlos, A. G., S. K. Menon and A. F. Atiya (2002). An adaptive state filtering algorithm for systems with partially known dynamics. *Journal of Dynamic Systems, Measurement, and Control* **124**(3), 364.

Pitre, R. (2004). A comparison of multiple-model target tracking algorithms. Master's thesis. University of New Orleans. New Orleans, Louisiana, United States.

Pitre, R. R., V. P. Jilkov and X. R. Li (2005). A comparative study of multiple-model algorithms for maneuvering target tracking. In: *Proceedings of the SPIE*. Vol. 5809. pp. 549–560.

Rao, C. V. and J. B. Rawlings (2002). Constrained process monitoring: Moving-horizon approach. *AIChE Journal* **48**(1), 97–109.

Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE Control Systems Magazine* **20**, 38–52.

Rawlings, J. B., D. Angeli and C. N. Bates (2012). Fundamentals of economic model predictive control. In: *Proceedings of the 51st IEEE Conference on Decision and Control*. Maui, Hawaii, USA. pp. 3851–3861.

Robertson, D. G., J. H. Lee and J. B. Rawlings (1996). A moving horizon-based approach for least-squares estimation. *AIChE Journal* **42**, 2209–2224.

Salhi, H. and F. Bouani (2016). Nonlinear parameters and state estimation for adaptive nonlinear model predictive control design. *Journal of Dynamic Systems, Measurement, and Control* **138**(4), 044502–044502.

Semerdjiev, E., L. Mihaylova and X. R. Li (2000). Variable- and fixed-structure augmented ZMM algorithms using coordinated turn model. In: *Proceedings of the Third International Conference on Information Fusion FUSION 2000*. Vol. 1. pp. MOD2/25–MOD2/32.

Shean, B. J. and J. J. Cilliers (2011). A review of froth flotation control.. *International Journal of Mineral Processing* **100**, 57 – 71.

Sun, Y. and N. H. El-Farra (2008). Quasi-decentralized model-based networked control of process systems. *Computers & Chemical Engineering* **32**(9), 2016–2029.

Valencia, F., J. D. Lopez, A. Marquez and J. J. Espinosa (2011). Moving horizon estimator for measurement delay compensation in model predictive control schemes.. *IEEE Conference on Decision & Control & European Control Conference* p. 6678.

Wächter, A. and L. T. Biegler (2006). On the implementation of primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming* **106**, 25–57.

Yang, Q. and Y. Liu (2016). Adaptive state estimation of multi-input and multi-output non-linear systems with general uncertainties both in the state and output equations. *IET Control Theory Applications* **10**, 354–362.

Zeng, J. and J. Liu (2015). Distributed moving horizon state estimation: Simultaneously handling communication delays and data losses. *Systems & Control Letters* **75**, 56–68.

Zhang, H. Y., Y. L. Kuang, G. H. Wang and L. Ji (2014). Soft sensor model for coal slurry ash content based on image gray characteristics.. *Internaional Journal of Coal Preparation and Utilization* **34**, 24 – 37.

Zhang, J. and J. Liu (2013). Distributed moving horizon state estimation for nonlinear systems with bounded uncertainties. *Journal of Process Control* **23**(9), 1281–1295.

Zhang, J. and J. Liu (2014). Observer-enhanced distributed moving horizon state estimation subject to communication delays. *Journal of Process Control* **24**(5), 672–686.

Zhang, Y. and X. R. Li (1998). Detection and diagnosis of sensor and actuator failures using IMM estimator. *IEEE Transactions on Aerospace and Electronic Systems* **34**(4), 1293–1313.