SOFT SENSORS FOR PROCESS MONITORING OF COMPLEX PROCESSES

A Dissertation

by

MITCHELL ROY SERPAS

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2012

Major Subject: Chemical Engineering

SOFT SENSORS FOR PROCESS MONITORING OF COMPLEX PROCESSES

A Dissertation

by

MITCHELL ROY SERPAS

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Juergen Hahn
Committee Members,	Sergiy Butenko
	Carl Laird
	Mahboobul Mannan
Head of Department,	Charles Glover

August 2012

Major Subject: Chemical Engineering

ABSTRACT

Soft Sensors for Process Monitoring of Complex Processes. (August 2012)

Mitchell Roy Serpas, B.S., Louisiana State University

Chair of Advisory Committee: Dr. Juergen Hahn

Soft sensors are an essential component of process systems engineering schemes. While soft sensor design research is important, investigation into the relationships between soft sensors and other areas of advanced monitoring and control is crucial as well. This dissertation presents two new techniques that enhance the performance of fault detection and sensor network design by integration with soft sensor technology. In addition, a chapter is devoted to the investigation of the proper implementation of one of the most often used soft sensors. The performance advantages of these techniques are illustrated with several cases studies.

First, a new approach for fault detection which involves soft sensors for process monitoring is developed. The methodology presented here deals directly with the state estimates that need to be monitored. The advantage of such an approach is that the nonlinear effect of abnormal process conditions on the state variables can be directly observed. The presented technique involves a general framework for using soft sensor design and computation of the statistics that represent normal operating conditions.

Second, a method for determining the optimal placement of multiple sensors for processes described by a class of nonlinear dynamic systems is described. This approach is based upon maximizing a criterion, i.e., the determinant, applied to the empirical observability gramian in order to optimize certain properties of the process state estimates. The determinant directly accounts for redundancy of information,

however, the resulting optimization problem is nontrivial to solve as it is a mixed integer nonlinear programming problem. This paper also presents a decomposition of the optimization problem such that the formulated sensor placement problem can be solved quickly and accurately on a desktop PC.

Many comparative studies, often based upon simulation results, between Extended Kalman filters (EKF) and other estimation methodologies such as Moving Horizon Estimation or Unscented Kalman Filter have been published over the last few years. However, the results returned by the EKF are affected by the algorithm used for its implementation and some implementations may lead to inaccurate results. In order to address this point, this work provides a comparison of several different algorithms for implementation.

To my wife and parents

ACKNOWLEDGMENTS

I would like to acknowledge that I have only been able to accomplish this task with the help of my committee members, fellow group mates, family and friends.

Foremost is the advice, academic but also career and personal, that I have received throughout my time at Texas A&M from Dr. Juergen Hahn. He has always gone above and beyond the call of duty to help me in the pursuit of my interests. I have learned so much from his vast knowledge and wonderful engineering mindset. It is clear to me that he truly cares about the success of his students.

In addition, I would like to thank my other committee members, Drs. Sergiy Butenko, Carl Laird, and Sam Mannan. I believe that through this wide range of areas of interest that each member offers, my education is diverse and complete. In addition to serving on my comittee, each has been personally involved in my work at Texas A&M. I would like to thank Dr. Butenko for his excellent instruction; his class was my first introduction to graduate optimization. I would especially like to thank Dr. Laird, whom I had the opportunity to work with and learn from on a regular basis, for his time and effort in furthering my education. Also, I would like to thank Dr. Mannan for his extensive work with the Mary Kay O'Connor Process Safety Center, as all of my interactions with the him and with the center have been so helpful.

I am very grateful to have had an environment such as we have within our Process Systems Engineering group full of such excellent students. I appreciate all of the helpful conversations in learning with George Abbott, Loveleena Bansal, Arjun Bhadouria, Yunfei Chu, Yankai Cao, Wei Dai, Gabe Hackebeil, Jacky Huang, Jia Kang, Sean Legg, Shreya Maiti, Vishal Manhindrakar, Angelica Mann, Cheryl Qu,

Daniel Word, and Yu Zhu.

Without the encouragement and support of my family and friends, I would not have been able to excel as I did. Thank you to all for living this part of my life with me. Special thanks to my wife, who moved away from her family and friends to join me here at Texas A&M. Also, I am so grateful to my parents and sister who have supported me in so many ways my entire life.

Most importantly, I would like to thank my Lord Jesus Christ. All of my success is due to Him.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

CHAPTER I

INTRODUCTION

As processes trend towards safer, cleaner, more energy efficient, and more profitable production in chemical plants and refineries, a continually increasing reliance on advanced monitoring and control is necessary. The use of the diverse set of tools made available through process systems engineering is now crucial to operation of manufacturing plants on many different levels. The domain of the Process Systems Engineer in practice consists of the interactions between the process and the operator. This is true across a large range of time and distances, from planning and scheduling the interactions between facilities spread across the globe years in advance to basic proportional control of a quick acting valve. Using mathematical and computational techniques, it is the job of the engineer to optimize this interaction by attempting to increase the amount of information gleaned from the process and ensure that the plant remains as close as possible to the operators desired trajectory. On the interface between this arena and the process lies the sensor. The sensor is any way that the process is physically measured, and constitutes the source of data, both historical and current, for use by the process systems engineering strategies. Conversely, the operator interacts with the system by providing goals for the process to attain. Everything in between, such as system control, analysis, optimization, simulation, monitoring, filtering and fault detection, form the suite of technologies that must attempt to perfect the communication between the two.

One critical piece of this arrangement is a soft sensor. A soft sensor is a mathematical technique used to infer important process variables that are not physically

---

The journal model is *Computers and Chemical Engineering*.

measured. This is accomplished by the use of the combination of current data of the plant with some previous knowledge of the plants behaviour, i.e. some type of plant model. The inclusion of the additional information about the process contained in the system model and the procedures to analyse the current state of the system allow for the enhancement of the information available. In other words, the estimated variables produced by soft sensors are often used with other techniques such as advanced process control, fault detection, process monitoring and product quality control to obtain higher performance. For example, no matter how fine-tuned a control system is, it cannot properly control system variables that it cannot observe. One role of a soft sensor is to provide an increased availability of variables as feedback to the control system. Because of these types of relationships (illustrated in Figure 1), the connections between soft sensors and other areas of process systems engineering should be thoroughly and carefully investigated. This constitutes the broad motivation for this dissertation.

Soft sensors can generally be in one of two subtypes based on the type of plant model it uses. The model can either be a historical data based model or a first principles model. While the former is used often in many fields, its is outside the scope of this dissertation. The later type uses a physically meaningful model of the internal dynamics of a process system, and is typically derived from accurate mathematical descriptions of known phenomena. A process model is usually developed initially based on general laws related to the process and further honed by parameter estimation. In this way, sometimes massive amounts of historical experience is accurately and concisely incorporated into the system model. Fundamentally, this is the reason for the enhance performance that comes with the inclusion of soft sensors. Of course system modeling is inherently imperfect and remains a very active area of research of its own (Chu, 2011). Because of model inaccuracies, all work regarding soft sensors

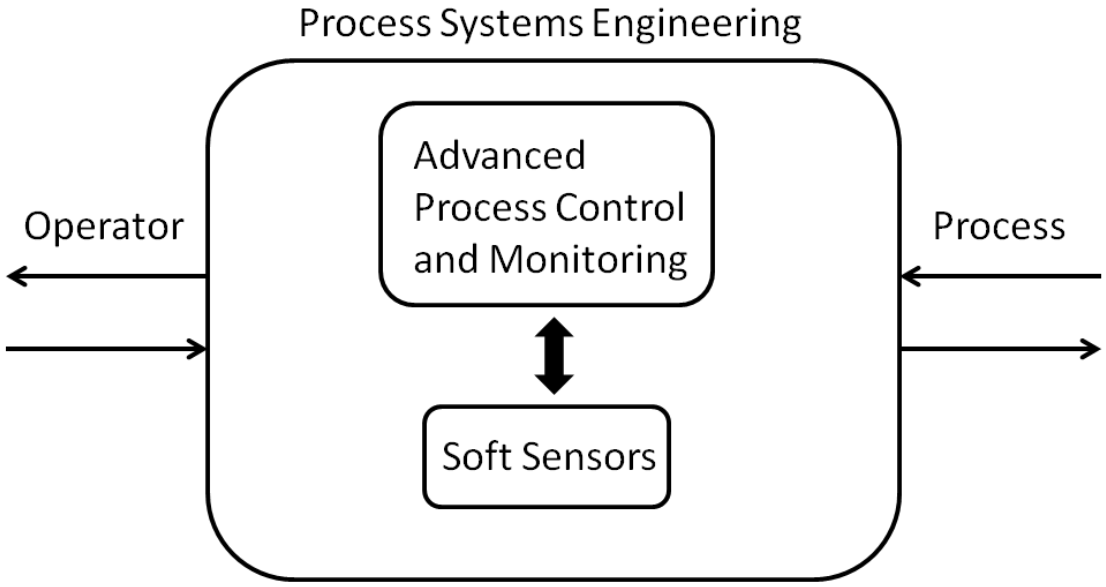


Fig. 1. Illustration of the relationship of soft sensors to process systems engineering

must be assumed to be only as accurate as the process model being used, at best.

A. Fault Detection

As the accuracy of modelling and soft sensor design continues to increase, the use of estimated variables for critical process monitoring and control tasks does as well. One crucial part of many process monitoring strategies is fault detection. Whether it be for an early warning safety limit system on a critical process variable, or for quality control on one parameter of a final product, fault detection is often relied upon to

relay important information about the state of the process. There are many types of fault detection schemes, including threshold detection, quick change detection, and model based fault detection. One key component in many of them is defining exactly where the variable of interest is supposed to be, and how much is it allowed to vary around that point before declaring a fault. This often translates mathematically into finding the statistical mean and variance of the variable.

Finding thresholds for fault detection is an often performed task in industrial application and usually involves statistical analysis of historical data. In other words, the variation in the variable during "normal operation" are used to determine thresholds for future fault detection. The issue becomes more complex however when there is a desire to use a soft sensor application in order to perform fault detection on unmeasured variables. In the general case, the thresholds that are proper for measured variables obtained from historical data will not be the correct thresholds for unmeasured variables.

This dissertation presents a method to use the information about the process previously known, such as variance, in combination with the soft sensor model, to perform fault detection on estimated state variables. A fault detection scheme is developed, including computation of the upper and lower control limits, which performs fault detection directly on variables that are estimated by soft sensors. This process is intentionally general in that it can be used with many of the available fault detection methods. The use of the additional process information contained in the process model enhances the performance of fault detection schemes that make use of a normal operation threshold.

B. Sensor Network Design

As mentioned previously, a control system can only be expected to perform as well as the quality of information it receives. Similarly, the accuracy of the soft sensor's estimations is dependent on its two sources of information about the process, namely the process model and the current state of the process. The former is related to the level of model detail and precision available. However, as mentioned earlier these are part of the considerations taken into account in the modeling process and are outside the scope of this work. Therefore, it is assumed that the most accurate model available is employed. Instead, this work focuses on the latter component. The inputs to the soft sensor calculations, i.e. the sensor or network of sensors, determine the best case accuracy of state estimation. In other words, if the placement of sensors throughout the process is not carefully considered, the soft sensor may be using less than ideal data. This is one of the main motivations for the field of Sensor Network Design.

In most industrial plant units there is a plethora of potential measurement locations. Because of this large number of possibilities and the complexities inherent to chemical engineering systems, the "best" location for sensor placement is often not obvious. Furthermore, the capital and maintenance costs of sensors produces a desire to employ a minimum number of sensors whenever possible.

There are several unattractive issues with much of the state of the art in sensor network design. Firstly, much of the groundwork has been in attempting to determine the single best sensor. However, in industrial application, the interest will almost always be on an entire network of sensors. This greatly complicates the problem as it introduces redundancy of information about the process. Secondly, many of the communications in sensor network design literature focus on steady state linear

models of processes. Although this allows for useful solutions to be found, it is only practical for certain specific situations, as most systems require dynamic control and exhibit nonlinear behaviour.

This dissertation presents a method for determining the optimal sensor network design for a general non-linear system. Specifically, the determinant of the empirical observability gramian is maximized using a hybrid approach to solve the necessary Mixed Integer Nonlinear Programming Problem.

C. Extended Kalman Filter Implementations

There exists a dichotomy between first principles modeling of physical systems, which is usually represented by continuous time ordinary differential equations, and measurement and computational techniques, which are fundamentally restricted to discrete time formulations. In addition to this gap, first principle models are almost always non-linear in nature, but many of the most used process systems techniques are derived and implemented using assuming a linear model. Because of these situations, the concepts of linearization, or alternatives of the same goal, and discretization are essential tools in systems engineering. The situation is complicated by the fact that there are many different ways of executing these goals. These competing technologies vary widely in accuracy and computational expense depending on the situation. Therefore it is crucial that the implementation of these strategies be done with extreme attention to the details.

This point is particularly pertinent to the application of soft sensors. There have been many simulation based performance comparisons between different soft sensors recently. These typically use Extended Kalman filters (EKF) performance as a benchmark for newer methods created as an attempt to rectify some of the flaws

of the EKF. These include Moving Horizon Estimation, Unscented Kalman Filter, and Particle Filtering. However, the results returned by the EKF are affected by the algorithm used for its implementation and some implementations of EKF lead to inaccurate results. In order to address this point, this dissertation investigates several different algorithms for implementing EKF. Advantages and drawbacks of different EKF implementations are discussed in detail and illustrated in a comparative simulation study.

This dissertation is organized as follows. Chapter II contains a thorough review of the previous literature and a review of techniques relevant to this work. Next, Chapter III presents a new technique for using fault detection with soft sensors. Then, Chapter IV contains a complete method for determining the optimal sensor network design. Chapter V offers a detailed look at implications of improper implementations of the EKF. Finally, Chapter VII concludes with a summary and discussion of future work.

CHAPTER II

LITERATURE REVIEW

The study of the science of soft sensors is a wide and diverse area of research. Logically, the central area of investigation details soft sensor design. This main topic can be broadly divided into soft sensor types based on data driven modeling and those based on first principles based modeling (Lin *et al.* , 2005). There is a very large collection of literature involving the design and application of the former, including current textbooks and state of the art survey papers (Fortuna *et al.* , 2007; Lin *et al.* , 2007; Kadlec *et al.* , 2009). However, the focus of this dissertation is restricted to the latter, namely soft sensors using first principle based models. This type of modeling is advantageous as it retains the physical meaning connected with important process variables, as is often desired in systems encountered in chemical engineering systems. A more thorough review of previous work concerning model based soft sensor design is detailed in the next section, followed by a review of literature in related fields such as soft sensors used with fault detection and sensor network design.

A. Soft Sensor Design

Model based soft sensor design techniques play a key role process monitoring and various types of estimators have been developed. Model based soft sensors, often called state filter or state estimators, can trace their origins to one of two historically important state estimators, the Luenberger Observer and the Kalman filter (KF). The first was developed by David Luenberger during his dissertation project (Luenberger, 1964). The Luenberger Observer is used on deterministic linear systems in order to reconstruct unmeasured dynamic states. It is tuned to guarantee stability and ideally produces a diminishing error in the absence of unknown disturbances and perfect

model knowledge. The Kalman Filter was introduced in a seminal paper by Rudolph Kalman (Kalman, 1960). As the term "filter" suggests, the Kalman Filter is designed to infer unobserved states, while simultaneously removing unwanted random noise inherent to all real systems. Although both state estimators continue be used in both academia and industrial applications, the Kalman Filter is by far the more popular for application in petrochemical process industries (de Assis, 2000; Soroush, 1998). Accordingly, the focus of this dissertation is on soft sensors, or state estimators, of the Kalman Filter "type".

1. The Kalman Filter

A general dynamic, stochastic, continuous, linear time invariant model is written as follows:

$$\dot{x} = Ax + Bu + Gw \tag{2.1}$$

$$y = Cx + Du + v \tag{2.2}$$

Here, x, y, and u are vectors containing the variables of the internal system states, the outputs, and inputs of the system respectively. Matrices A, B, G, C and D contain the system parameters. Additionally, w and v are vectors representing the random disturbances present in the system, specifically dynamic model noise and measurement noise. It is assumed that both are zero mean Gaussian white noise processes, given by $w \sim N(0, Q)$ and $v \sim N(0, R)$. The noise terms are assumed to be uncorrelated in time and uncorrelated to each other and the states.

This continuous time model involving differential equations often results naturally from linearization of physically meaningful natural laws such as the energy balance resulting from the first law of thermodynamics. However, because actual data is inherently discrete and due to the simplifications that result from using only

algebraic equations, there is often a desire to represent this model in a discrete time form (2.3)-(2.4).

$$x_k = A_d x_{k-1} + B_d u_k + G_d w_k \tag{2.3}$$

$$y_k = C_d x_k + D_d u + v_k \tag{2.4}$$

where each of the vectors are now defined at each time point k, and the matrices are denoted as the discrete time form. This equivalent model is obtained through the following transformation:

$$A_d = e^{AT} \tag{2.5}$$

$$B_d = A^{-1}(A_d - I)B \tag{2.6}$$

$$C_d = C \tag{2.7}$$

$$D_d = D \tag{2.8}$$

$$G_d = A^{-1}(A_d - I)G \tag{2.9}$$

T is the sampling interval of the system. Because of the reasons mentioned above, most of the methods in this work assume a discrete time form. Accordingly the accent $_d$ will be dropped for simplicity of notation.

Assuming a model given by equations (2.3)-(2.4), the Kalman Filter is given by the following set of recursive equations.

$$P_k^- = AP_{k-1}A^T + GQG^T \tag{2.10}$$

$$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1} \tag{2.11}$$

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \tag{2.12}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \tag{2.13}$$

$$P_k = (I - K_k C)P_k^- \tag{2.14}$$

At each time step the state estimate $\hat{x}_k$ is calculated, in addition to the error covariance matrix $P_k = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]$. Here, $\hat{x}_k^-$ represents the propagated, or a priori, state estimate, which is simply the state variables propagated from one time step to the next using the system model equations. The term $\hat{x}_k$ represents the updated state estimate, where the update is an error term multiplied by the optimal Kalman gain, and added to the a priori estimate. The calculation of the Kalman gain, denoted by K_k, is included in the recursive format of the state dynamics and is accomplished using the noise covariance terms Q and R. The gain term represents a trade-off between the accuracy of the model versus the precision of the data. Determination of these factors is discussed in section 3.

2. State Estimation for Nonlinear Systems

For unconstrained linear systems, the KF provides the optimal state estimation in the least squared sense (Mariani & Corigliano, 2005). The KF is applied to real world, non-linear systems via local linearizations, providing a suboptimal estimate (Grewal & Andrews, 2001). The disadvantages include its restriction to the area directly around a chosen steady state, possible existence of offset between the estimate and actual state value, and that it does not easily allow inclusion of system constraints. This latter point is especially pertinent in chemical engineering systems as most all state variables have at least non-negativity constraints. Despite this, the KF is a very often implemented state estimator, and has formed the basis for development for many nonlinear filters. These extensions include the widely used Extended Kalman Filter (Schmidt *et al.* , 1976; Anderson & Moore, 1979; Sorenson, 1985; Grewal & Andrews, 2001), which utilizes a dynamically updated first order linearization approximation of nonlinear systems. Here, the a priori density is assumed Gaussian, and is captured by the propagation of the mean and covariance only. A basic, first order Taylor Series

linearization is performed on the system model.

The EKF is applied to a nonlinear system matching the general discrete time formulation of the following set of equations.

$$x_k = f(x_{k-1}, u_k, w_k) \tag{2.15}$$

$$y_k = h(x_k, u_k, v_k) \tag{2.16}$$

Here, the variable definitions match those of equations (2.3)-(2.4) only the models functions f and g are no longer restricted to the linear relationships posed there. In order to overcome this difference the matrices $\tilde{A}_k = \frac{df}{dx^T}$, $\tilde{C}_k = \frac{dh}{dx^T}$, and $\tilde{G}_k = \frac{df}{dw^T}$ can be calculated as approximations of the linear system at each time step. This allows the EKF to outperform a simple "one-time" linearization of the process by updating the linearization as the system moves to a different place in state space. However, the EKF is still reliant on the accuracy of first order approximation and still does not easily allow for the inclusion of system constraints.

Another useful filter for nonlinear systems derived in the spirit of the KF is the Unscented Kalman Filter (Julier & Uhlmann, 1997; Wan & Van Der Merwe, 2000; Xiong *et al.* , 2006; Pan *et al.* , 2005), which combines the KF with an unscented transformation. The UKF is comparatively new but is being applied in chemical engineering systems (Julier & Uhlmann, 2004, 1997; Romanenko & Castro, 2004; Romanenko *et al.* , 2004). The UKF propagates a deterministically chosen set of sample points through the nonlinear system so that the mean and covariance can be calculated without the use of linearization. In this way, the error covariance matrix, P_k in equation (2.14), is given by the following equation.

$$P_k = \sum_i [w^i (z_k^i - \hat{x}^k)(\eta_k^i - \hat{x}^k)^T] \tag{2.17}$$

Here, w^i represents the weight given to each sample z_k^i. Equation (2.17) shows that in the UKF the state estimate error is not propagated through system linearization but uses a specified number and type of sample points in order to obtain an "average" of the system around the current location of the system in state space. In addition to this, the UKF provides the advantage of not requiring the system's partial derivatives for its computation.

A large and diverse class of state filters, broadly called sample based filters, have been in development recently. The author is including in this category Particle Filters, Cell Filters, Ensemble Kalman Filters and others (Lang *et al.* , 2007; Prakash *et al.* , 2010; Ungarala, 2011). The unifying concept in this class of state estimators is that they propagate randomly chosen sample points through the nonlinear system and allow for the approximation of any non-Gaussian distribution. Although the inner working of these filters is varied and very different from the UKF, the final error covariance matrix calculation is similar to equation (2.17), as they are sample based. Much research has been done in attempting to incorporate state constraints into particle filtering directly, and the proper way to accomplish this task remains open. Although random sample filters may currently be too computationally intensive for many applications, the allowance for any general shaped probability density for the random disturbances entices many to work in this field.

Finally, probably the most popular nonlinear state estimator in recent research is the Moving Horizon Estimator. This will be discussed in the next sub-section as the framework is best understood from an independent treatment.

3. Moving Horizon Estimation and Arrival Cost

Moving Horizon Estimation has become an increasingly popular strategy for state estimation over the last number of years (Rawlings & Bakshi, 2006; Zavala

et al. , 2008). This is due to the fact that it allows for directly dealing with model nonlinearities and state variable constraints. Consider a system described by the following discrete time, general nonlinear recursive model.

$$x_{k+1} = f(x_k, u_k) + w_k \tag{2.18}$$

$$y_k = h(x_k) + v_k \tag{2.19}$$

The solution to the following optimization problem provides the optimal state estimate in the least squares sense, as displayed in (Rawlings & Bakshi, 2006).

$$\begin{aligned} min_{X(T)} & \left[V_0(x(0)) + \sum_{j=1}^{T-1} L_w(w(j)) + \sum_{j=0}^{T} L_v(y(j) - h(x(j))) \right] \\ \text{s.t. } & x_{j+1} = f(x_j, u_j) + w_j \\ & V_0(x) = -log(p_{x(0)}(x)) \\ & L_w(w) = -log(p_w(w)) \\ & L_v(v) = -log(p_v(v)) \end{aligned} \tag{2.20}$$

The objective function is a sum of three terms V_0, L_w, and L_v, which represent the uncertainty or state estimate error due to the initial state, dynamic model noise, and measurement noise, respectively. This formulation of the state estimation problem into an optimization problem, known as full information estimation, allows for directly dealing with nonlinear system models and for the explicit inclusion of system constraints. Both of these less than ideal situations occur frequently in chemical engineering processes. However, the computational burden problem defined in equation (2.20) increases dramatically as time horizon and system size increase. In response to this difficulty, the Moving Horizon Estimator (MHE) was developed to use the same

formulation as equation (2.20) on a fixed window of data, size N.

$$min_{X(T-N:T)} \left[V_{T-N}(x(T-N)) + \sum_{j=T-N}^{T-1} L_w(w(j)) + \sum_{j=T-N}^{T} L_v(y(j) - h(x(j))) \right] \tag{2.21}$$

Here V_{T-N} is known as the arrival cost. The arrival cost is responsible for including information about the system at the beginning of the horizon. More specifically, this represents the *a priori* density function. If the system is linear and unconstrained, the Kalman Filter provides the exact *a priori* density in the form of its state estimate and error covariance matrix (Rao & Rawlings, 2002). In this case, the arrival cost can be expressed as follows.

$$V_{T-N}(x(T-N)) = (x(T-N) - \hat{x}(T-N))^T P_{T-N}^{-1}(x(T-N) - \hat{x}(T-N)) \tag{2.22}$$

Here, V_{T-N} is the error covariance matrix from the Kalman filter defined by the following equation.

$$P_T = (x(T) - \hat{x}(T))^T (x(T) - \hat{x}(T)) \tag{2.23}$$

The error covariance matrix evolves according to the following dynamic equation.

$$P_T = AP_{T-1}A^T + GQG^T - AP_{T-1}C^T(CP_{T-1}C^T + MRM^T)^{-1}CP_{T-1}A^T \tag{2.24}$$

The steady state solution can be determined for linear systems by solving the related discrete time algebraic Riccati equation.

For general nonlinear systems, the exact representation of this density is non-trivial, and is typically approximated in one of many available methods. The main open area of research for MHE is in determining good approximations of the arrival cost. This is critical to the industrial application of MHE because the poorer the

estimation of the arrival cost, the larger the estimation horizon must be in order to provide satisfactory performance. However, the larger the estimation horizon (and therefore the computational cost), the longer the state estimation takes at each time step. The goal of arrival cost research is to find a useful compromise in this tradeoff (Ungarala, 2009).

The arrival cost may be calculated using any nonlinear filter that propagates the mean and covariance (Robertson & Lee, 1995; Ungarala, 2009). Recent literature is full of attempts to use a large variety of nonlinear filters for this approximation. The traditional approach for estimating the arrival cost for MHE on nonlinear systems is the Extended Kalman Filter (Ungarala, 2009). Using EKF to calculate the arrival cost follows the approach outlined above where $A = \frac{df}{dx^T}$, $C = \frac{dh}{dx^T}$, and $M = \frac{dh}{dv^T}$ in equation (2.24). The disadvantages of using EKF to calculate the arrival cost include inability to include system constraints, and the calculation of the covariance matrix through system linearization introduces errors (Vachhani *et al.* , 2006). These two problems have been shown to introduce unwanted errors and have led to a major response in the literature for investigating the development of other approaches (Haseltine & Rawlings, 2005).

The Unscented Kalman Filter has been proposed for use in computing the arrival cost as an improvement from the EKF. The approach of using UKF for the arrival cost estimation has been implemented and compared to results using EKF to calculate the arrival, and reported improved performance in all cases studied (Qu & Hahn, 2009b; Ungarala, 2009). In order to deal with constraints it was proposed using a constrained version of UKF for calculating the arrival cost using clipping to satisfy the bounds (Kandepu *et al.* , 2008). However, this has been criticized as ad hoc and potentially sub optimal (Lopez-Negrete & Biegler, 2011).

Many types of sample based filters have been proposed for use in calculating the

arrival cost as well (Lopez-Negrete *et al.* , 2009; Ungarala, 2009). Typically the limitation in using these types of filters is computation cost. However, research continues into how best to formulate the techniques in sample based filters to create efficient algorithms. As this field develops these types of filters may be used increasingly to compute arrival cost.

Moving Horizon Estimation remains one of the most promising solutions to the constrained nonlinear online estimation problem. However, due to high computation costs in the complex optimization problem, live industrial implementation is restricted to low horizon lengths. This requires an accurate estimation of the arrival cost. The best arrival cost calculation seems to be theoretically difficult, especially when considering the inclusion of state constraints. Many of the techniques described above show improved performance over the basic EKF approach, especially the constrained sample based filters.

B. Crucial Related Fields

While much work has been done focusing specifically on soft sensor design, there are also many related fields that require thorough investigation. Because soft sensors are pieces in a larger framework that is process monitoring, many of these peripheral areas of research arise due to the interactions between the soft sensors and the tools used to monitor and control the process. For example, advanced control techniques often rely on soft sensors for accurate and current values for the internal states of the process. The following subsection details a review of literature related to Fault Detection use with Soft Sensors. Next, a subsection discussing the work related to Sensor Network Design is given. Finally a discussion on determination of the Kalman Filter tuning parameters, Q and R.

1. Fault Detection

Fault detection has been described as the number one priority for improvement of industrial process operations (Venkatasubramanian *et al.* , 2003a). A fault can be defined as an a non-permitted deviation of a characteristic property which leads to the inability to fulfill the intended purpose (Isermann, 1984). However, even strictly within Chemical Engineering related research, fault detection is a wide and diverse area of investigation, both in application and approach, as shown in several comprehensive reviews (Kramer & Mah, 1994; Venkatasubramanian *et al.* , 2003b,c,a; Zhang & Jiang, 2008). Quantitative fault detection can be broadly divided into techniques that use historical process data, and techniques using model-based methods.

Within the set of process history techniques, one main category deals with statistical methods (Qin, 2003). The most commonly used technique is principal component analysis, a type of data/model reduction (Chiang *et al.* , 2000; Li *et al.* , 2000). Another widely-used approach uses neural networks for fault detection (Venkatasubramanian & Chan, 1989), which is often used for nonlinear systems (Nahas *et al.* , 1992).

The second set of methods, model-based methods, has also been extensively discussed in the literature (Frank, 1990; Frank *et al.* , 2000; Katipamula & Brambley, 2005). Model-based approaches can be broken up into three areas: diagnostic observers, parity space methods, and fault detection filters (Frank *et al.* , 2000). These methods, in different ways, perform fault detection by monitoring some model error residual. However, it has been shown that these different methods are closely related and often return similar or identical results (Ding *et al.* , 1998; Frank *et al.* , 2000; Gertler, 1991; Hou & Mller, 1994). Applications of model-based approaches to industrial processes have also been reported (Chetouani, 2004; Huang *et al.* , 2003). Model

based approaches are typically derived for linear systems, however, some extensions to nonlinear systems have also been reported (Venkatasubramanian *et al.* , 2003a).

One of the common themes among all of these fault detection approaches is that they compare current measurements to values that are regarded as representing normal plant operation. The differences among the techniques result from how these nominal values are computed and how the comparison between the current measurements and their nominal values are made. One area that has received little to no attention in the literature is that fault detection can be performed on physically important, but unmeasured, variables, if soft sensors are employed. Himmelblau (Watanabe & Himmelblau, 1983) mentioned that this is a possibility, however, did not propose a solution as to how such a fault detection scheme should be implemented. Even now, according to Fortuna (Fortuna *et al.* , 2007), the main application of soft sensors for fault detection is still to compare the soft sensor prediction with an actual measurement and not perform fault detection on the soft sensor predictions themselves.

A large number of different fault detection methods have been developed and implemented. While it is beyond the scope of this work to review all of these techniques here, there are concepts that can be found universally across all schemes. Every process monitoring approach compares one or more variables that are monitored to values of these variables that reflect normal operation. The decision if a process is operating within normal parameters is usually made by comparing the variable against a threshold value.

Possibly the simplest of these monitoring techniques is limit checking using thresholds. Here, the mean and variance, defined below, are used to determine if

the process is experiencing a fault or not (Isermann, 2006).

$$\mu_i = E[y_i(t)] \tag{2.25}$$

$$\bar{\sigma}_i^2 = E[(y_i(t) - \mu_i)^2] \tag{2.26}$$

These statistics can be applied to determine the presence of faults in a number of ways, including application of hypothesis tests, control chart techniques, and adaptive thresholds (Himmelblau, 1978; Hofling & Isermann, 1996; Scharf & Demeure, 1991). The general approach for using these techniques is to compare current live data with some expected statistics of that data. More specifically for limit checking, the residual used for fault detection is determined by the difference between the expected value of the process and the actual value.

$$r_k = y_k - y_{ss} \tag{2.27}$$

This residual can then be compared to some threshold. This same principle similarly applies to many fault detection schemes as a residual is calculated as the difference between data and the normal operating condition, and this residual is compared to a threshold, as shown below.

$$r_k < r_{min}\, OR\, r_k > r_{max} \rightarrow \text{fault} \tag{2.28}$$

Regardless of the method used for determining the residual, almost all fault detection methods require determination of this threshold. The success of a fault detection scheme can only be as good as the accuracy of the boundaries chosen for the fault decision. Despite the large amount of research dealing with fault detection methods, little effort has been put into the computation of appropriate thresholds. This finding is surprising as determining an appropriate threshold is a key parameter for process

monitoring, even for such a simple and widely applied method such as limiting checking (Ding, 2008; Ding *et al.* , 2004). To the authors knowledge, there is no work previously done on threshold computation specifically for use with soft sensors.

2. Sensor Network Design

There have been many contributions advancing the field of sensor network design. The solution to this problem is highly valued due to evidence that poor sensor network design can cause a wide range of problems for plant operation (Van den Berg *et al.* , 2000). The goal of sensor network design for chemical engineering systems is to choose an optimal group of sensors from many available measurement locations based on evaluating the capital and maintenance costs of the sensors and some performance criteria. The performance criteria has taken on many different forms throughout the research literature, but the general concept remains to maximize the accuracy and precision of the monitoring systems knowledge of the system. Some groups focus on minimizing undesired disturbances, or noise, in the measurements, and others attempt to isolate the richest data set, assuming excellent measurement and model quality. In addition, others focus solely on the cost minimization piece of the problem formulation. Simultaneous to these research efforts in proper problem formulation, there have been many advances in solution strategies for the programming problems that result.

Specifically, there are many interesting contributions focusing on minimizing cost while also maximizing precision and reliability via variability (Kotecha *et al.* , 2008; Luong *et al.* , 1994; Bagajewicz & Sanchez, 2000; Bagajewicz, 1997; Ali & Narasimhan, 1993; Madron & Veverka, 1992). Nguyen and Bagajewicz extended current linear techniques to nonlinear systems (Nguyen & Bagajewicz, 2008). Over the years, other work has focused specifically on the maximization of observability in ranking sensor

network designs (Georges, 1995; Kister, 1990; Waldraff *et al.* , 1998; Raghuraj *et al.* , 1999). However, there has also been work specifically in observability for nonlinear systems using other metrics, including Lie-algebra techniques, gram determinant test of independence and nonlinear observability gramians (Hermann & Krener, 1977; Scherpen, 1993). Some researchers use the state error covariance matrix from the Kalman Filter (equation (2.14)) instead (Kadu *et al.* , 2008; Mellefont & Sargent, 1978; Mehra, 1976; Muske & Georgakis, 2003; Musulin *et al.* , 2005). Also, several researchers have presented approaches for optimal monitoring design using Principle Component Balance combined statistics (Brewer *et al.* , 2007; Stanimirovic *et al.* , 2008; Zumoffen & Basualdo, 2010).

The many techniques represented here have been applied to a variety of chemical systems. Recently, Sumana and Venkateswarlu investigated different observability measures based on the empirical observability gramian applied to reactive distillation (Sumana & Venkateswarlu, 2009). The trace of the observability gramian has been applied to fault detection in micro chemical plants (Tonomura *et al.* , 2008). Some work on the increasingly popular water network security problem via sensor network design (Busch *et al.* , 2009; Mann *et al.* , 2011). There is also significant work on related research that requires only observability or redundancy of measurements and focuses on minimizing the cost of sensor networks (Nguyen & Bagajewicz, 2008; Uci ski & Patan, 2010). In a somewhat different approach, some research is focused on designing optimal sensor networks while explicitly considering optimal control (Fraleigh *et al.* , 2003; Vande Wouwer *et al.* , 2000). In addition, there have been related contributions focusing on catastrophic event detection (Berry *et al.* , 2006; Legg *et al.* , 2012).

The field of sensor network design for soft sensing based on maximization of information gathered by the sensor takes advantage of several other areas of pro-

cess systems engineering in order to accomplish its tasks. The following subsections present a review of these areas including some review of the pertinent literature. First, the concept of experimental design is introduced and its relationship to sensor network design is discussed. Second, a specific class of programming problems, MAX-DET optimization, is stated briefly. Next, the central concept of system observability is presented, for linear and nonlinear systems. Finally, a short subsection on a recent solution technique for Mixed Integer Non-Linear Programming Problems is included.

a. Experimental Design

The goal of experimental design is to determine the best set of conditions for determining optimal values for estimated parameters. In experimental design, a quantitative measure of goodness of design is the Fisher Information Matrix (Walter & Pronzato, 1990). The Fisher Information Matrix is related to the covariance between the measured variables and the estimated parameters. Just as with the observability gramian, in order to perform optimization, a scalar function of the matrix must be chosen. There are many functions used for this task, such as the E-optimality criterion, D-optimality criterion and the A-optimality criterion (Kiefer, 1992). These criteria optimize the least eigenvalue, the determinant, and trace, respectively, of the covariance matrix, represented below by Σ.

$$\Phi_E = \max\left[\lambda_{min}\left(\Sigma\right)\right] \tag{2.29}$$

$$\Phi_D = \max\left[\det\left(\Sigma\right)\right] \tag{2.30}$$

$$\Phi_A = \min\left[\text{trace}\left(\Sigma\right)\right] \tag{2.31}$$

Though each criterion has its own purpose, the determinant of the Fisher Information Matrix is the most widely accepted optimality criterion for experimental design

(Emery & Nenarokomov, 1998; Uci ski, 2005; Walter & Pronzato, 1990; Wynn, 1972). Therefore the optimization problem is generally described as equation (2.30).

b. MAX-DET Optimization

Because of its usefulness in experimental design and other fields, the problem class of maximization of the determinant of a matrix has been previously studied thoroughly. It is critical that this problem formulation be investigated as the determinant introduces nonlinearity to the objective function, thereby significantly increasing the difficulty of the problem. It has been shown that the optimization problem resulting from maximizing the determinant of a positive semi-definite matrix, as written in equation (2.32), is convex (Atkinson *et al.* , 2007; Vandenberghe *et al.* , 1998).

$$\begin{aligned} \min_{x} \quad & c^T x + \log\det[G(x)^{-1}] \\ \text{s.t.} \quad & H(x) > 0 \\ & F(x) \geq 0 \end{aligned} \tag{2.32}$$

In addition to guaranteeing a global solution, this property allows the use of problem solving techniques that are designed for the convex class of problems.

c. Observability

Observability is a property of a dynamic system which determines if it is possible to reconstruct the values of the states from the measurements. As such, observability is a property which is commonly used in some form for sensor network design (Damak *et al.* , 1993; Mller & Weber, 1972; Waldraff *et al.* , 1998). A linear system is said to be observable if the initial state can be determined by some finite set of measurement

data (Brogan, 1985). For linear systems of the form,

$$\dot{x} = Ax + Bu \tag{2.33}$$

$$y = Cx + Du \tag{2.34}$$

the observability gramian (eq. (2.35)) and observability matrix (eq. (2.36)) are defined by the following equations:

$$W_{O,linear} = \int_0^\infty e^{A^T t} C^T C e^{At} dt \tag{2.35}$$

$$Q = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{2.36}$$

Here, the internal system state vector is represented by $x \in R^{n\times 1}$, the input or manipulated variable vector is $u \in R^{o\times 1}$, and the measurement or system output vector is given by $y \in R^{m\times 1}$.

For general nonlinear systems given by the following general formulas,

$$\dot{x} = f(x, u) \tag{2.37}$$

$$y = g(x, u) \tag{2.38}$$

calculation of these observability matrices analytically is not possible, as the relationships between variables are dependent on the location in state space. One solution would be to simply obtain the observability gramian of a linearized system around a specific operating point. This likely will produce misleading results, however, since they will only be valid when the system is exactly at the chosen steady state. Instead, the empirical observability gramian offers an alternative for use directly on nonlinear

systems (Singh & Hahn, 2005b). The empirical observability gramian is defined as follows:

$$W_o = \sum_{l=1}^{r} \sum_{m=1}^{s} \frac{1}{(rsc^2)} \int_0^\infty T_l \Psi^{lm} T_l^T dt \tag{2.39}$$

$$\Psi_{ij}^{lm}(t) = (y^{ilm}(t) - y_{ss})^T (y^{jlm}(t) - y_{ss}) \tag{2.40}$$

$$T^n = \left\{ T_1, ..., T_r; T_i \in R^{nxn}, T_i^T T_i = I, i = 1, ..., r \right\} \tag{2.41}$$

$$M = \{c_1, ..., c_s; c_i \in R, c_i > 0, i = 1, ..., s\} \tag{2.42}$$

$$E^n = \{e_1, ..., e_n; \text{unit vectors in } R^n\} \tag{2.43}$$

Here, y^{ilm} is the system output given by the initial condition $x(0) = c_m T_l e_i + x_{ss}$.

Using this approximation, observability information can be found that provides some sort of average around an operating point. The empirical observability gramian has been shown to both offer an accurate approximation for nonlinear systems and has been shown to be equivalent to true observability gramians when applied to linear systems (Singh & Hahn, 2005a). Furthermore, as the specified range around the operating point is reduced, the empirical observability gramians results reduce to that of the observability gramian of the linearized system.

Both the observability gramian and the observability matrix are useful in determining the binary decision of whether a system is fully observable or not. In each case, if the rank is equal to the number of states, n, then the system is fully observable. However, the answer to the question of how to quantify a scale of observability for ranking various systems is not quite as clear. Several metrics based on producing a scalar value from the observability gramian have been used to rank increasing observability (Mller & Weber, 1972). Certain metrics, such as the determinant, which are known to be useful for this were not used in past work (Singh & Hahn, 2006; Sumana & Venkateswarlu, 2009). This seems to be due to the computational com-

plexity that these metrics cause when applied to sensor network design. For example, the trace is popular, possibly due to the simplicity of the resulting optimization problem. However, most scalar functions other than the determinant do not include all of the observability information from the matrix as they only operate on certain terms in the matrix.

d. MINLP Solution Methods

Mixed Integer Nonlinear Programming problems (MINLPs) are often considered extremely difficult to solve, and solution methods are an open area of research. A set of hybrid solution techniques for solving convex MINLPs is presented in recent literature (Bonami *et al.* , 2008) and is briefly introduced here.

The solution technique detailed in literature operates on a subclass of MINLPs whose continuous relaxation is convex. This type of formulation is shown in equation (2.44).

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ \text{s.t.} \quad & g(x,y) \leq 0 \\ & x \in X \cap Z^n \\ & y \in Y \end{aligned} \tag{2.44}$$

Here, the functions $f : X \times Y \rightarrow R$ and $g : X \times Y \rightarrow R$ are continuously twice differentiable and convex. Note that the MINLP in the previous formulation can be reformulated as an MINLP with a linear objective function by introducing a new variable a which is minimized subject to the additional constraint f(x,y)= a. The solution procedure begins by introducing linearizations of the objective function and constraints to build the relaxed problem in the form of a mixed-integer linear pro-

gramming problem (MIP). The relaxed problem has the form of Equation (2.45).

$$
\begin{aligned}
\min_{x,y} \ & \alpha \\
\text{s.t.} \ & \bigtriangledown f\left(x^k, y^k\right)^T \begin{pmatrix} x - x^k \\ y - y^k \end{pmatrix} + f(x^k, y^k) \leq \alpha, \forall (x^k, y^k) \in T \\
& \bigtriangledown g\left(x^k, y^k\right)^T \begin{pmatrix} x - x^k \\ y - y^k \end{pmatrix} + g(x^k, y^k) \leq 0, \forall (x^k, y^k) \in T \\
& x \in X \cap Z^n \\
& y \in Y \\
& \alpha \in R
\end{aligned}
\tag{2.45}
$$

Here T represents a set of points (x^k, y^k) where the functions f and g are linearized. Due to convexity of the functions f and g, this problem is guaranteed to provide a lower bound on the original objective function. Bonami et al. show that by successivly strengthening the relaxed subproblem with new linearization points the algorithm converges to an optimal solution of the original MINLP in a finite number of steps, provided that assumptions on convexity, differentiability, and constraint qualifications hold (Bonami *et al.* , 2008). Application of this solution technique to the sensor location problem is discussed in further detail in the following section.

3. Estimation of Unknown Disturbance Statistics

In order to tune Kalman Filter type state estimators, information about the dynamic state model error and the measurement noise must be known. As described shown in Section 1 for KF and EKF particularly, the Q and R covariance matrices must be determined before implementing the filter. Currently, the most popular technique in process systems research for accomplishing this goal is the Autocovari-

ance Least Squares (ALS) method (Odelson *et al.* , 2006b,a; Rajamani & Rawlings, 2009; Odelson, 2003; Rajamani *et al.* , 2007; Åkesson *et al.* , 2008). ALS is part of a larger group of techniques for estimating these covariance matrices called correlation techniques. This set of techniques accomplishes the estimation using the residuals from state filtering and was pioneered by several researchers previously (Mehra, 1970, 1972; Belanger, 1974; Carew & Bélanger, 1973). In addition, there are other ways historically used for estimating these covariance matrices that have not been focused on recently for various reasons. These areas include Bayesian, maximum likelihood, and covariance matching methods (Kashyap, 1970; Bohlin, 1976; Hilborn & Lainiotis, 1969; Alspach, 1974; Bunn, 1981). Finally, there is a growing interest in researching optimal adaptive filtering with application to electrical engineering systems (Karasalo & Hu, 2011; Ding *et al.* , 2007; Yang & Gao, 2006).

The ALS method is a technique to estimate the covariance of the model and measurement noise terms in a state space model, using information from real process data. Much of the difficulty in this estimation process is due to the goal of simultaneously estimating the unknown covariance of both noises. The ALS technique for linear systems is steady state data from a time-invariant system that can be described by the following discrete time equations.

$$x_k = Ax_{k-1} + Bu_{k-1} + Gw_{k-1} \tag{2.46}$$

$$y_k = Cx_k + Du_k + v_k \tag{2.47}$$

$$Q \equiv E[w_k w_k^T] \tag{2.48}$$

$$R \equiv E[v_k v_k^T] \tag{2.49}$$

Here, $x \in R^n$ is the vector of states, $y \in R^m$ is the vector of measurements, $w \in R^g$ is the vector of random state disturbances, $v \in R^m$ is vector of measurement noise, and

$u \in R^o$ is the vector of inputs. Each process variable is defined at every time step k.

First, a discrete time linear filter is used on the data to produce estimates of the state variables, from chosen "best guess" filter gain, L.

$$\hat{x}_k^+ = \hat{x}_k^- + L(y_k - C_k\hat{x}_k^-) \tag{2.50}$$

$$\hat{x}_k^- = A\hat{x}_{k-1}^+ + Bu_{k-1} \tag{2.51}$$

Then, the innovations term is formed at each step by subtracting the actual historical data from the "estimated data", and the error term is defined as the difference between the actual state value and the propagated state estimate at each time step.

$$Y_k \equiv y_k - C\hat{x}_k^- \tag{2.52}$$

$$e_k \equiv x_k - \hat{x}_k^- \tag{2.53}$$

Now, with substitution, the dynamic model for the error term is obtained.

$$e_k = \bar{A}e_{k-1} + \bar{G}\bar{w}_{k-1} \tag{2.54}$$

$$Y_k = Ce_k + v_k \tag{2.55}$$

$$\bar{A} \equiv A - ALC \tag{2.56}$$

$$\bar{G} \equiv [G, -AL] \tag{2.57}$$

$$\bar{w}_k \equiv [w_k, v_k]^T \tag{2.58}$$

$$\bar{Q} \equiv E[\bar{w}_k\bar{w}_k^T] = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \tag{2.59}$$

Next, the propagation of the variance of the error term is given by the following.

$$P_k \equiv E[e_k e_k^T] \tag{2.60}$$

$$P_k = \bar{A}P_{k-1}\bar{A}^T + G\bar{Q}\bar{G}^T \tag{2.61}$$

The steady state value of the variance of the error term is simply given by the solution to the Lyapunov equation.

$$P = \bar{A}P\bar{A}^T + \bar{G}\bar{Q}\bar{G}^T \tag{2.62}$$

The next piece of information needed is the correlation matrix of the innovations term, defined by the following equation.

$$S_N \equiv \begin{bmatrix} c_0 & \cdots & c_N \\ \vdots & \ddots & \vdots \\ c_N & \cdots & c_0 \end{bmatrix} \equiv E \begin{bmatrix} Y_k Y_k^T & \cdots & Y_k Y_{k+N}^T \\ \vdots & \ddots & \vdots \\ Y_{k+N} Y_k^T & \cdots & Y_{k+N} Y_{k+N}^T \end{bmatrix} \tag{2.63}$$

Here, N is the number of time steps chosen for the correlation. Each term in this matrix can be estimated from the innovations, by the following often used technique.

$$c_j \approx \frac{1}{N-j} \sum_{i=1}^{N_data} Y_{i+j} Y_i^T \tag{2.64}$$

Next, the equation for the correlations is determined, by plugging in the dynamic equations shown previously.

$$c_0 = E[Y_k Y_k^T] = CPC^T + R \tag{2.65}$$

$$c_j = E[Y_{k+j} Y_k^T] = C\bar{A}PC^T - CALR \tag{2.66}$$

Finally, the relationship between the innovations term obtained from data and the covariance of the noise can be established. By using the above equations, the Lyapunov equation for the value of P and the estimated value for each innovations correlation, the equation can be derived and solved for Q and R. However, in most cases the N is chosen large enough where the solution is not unique, and an optimization strategy is needed. An optimization approach is also necessary in order that positivity constraints can be placed on the variances. In this case, the problem is

formulated as follows.

$$\begin{aligned} &\min_{x} \ \|Zx - b\|_2 \\ &\text{s.t. } x \leq 0 \\ &Z = \begin{bmatrix} ((C \otimes O)(I_{n^2} - \bar{A} \otimes \bar{A})^{-1} \\ ((C \otimes O)I_{n^2} - \bar{A} \otimes \bar{A})^{-1}(AL \otimes AL) + (I_p \otimes G)) \end{bmatrix}^T \\ &x = \begin{bmatrix} Q|_s \\ R|_s \end{bmatrix} \end{aligned} \tag{2.67}$$

Here, $Q|_s$ and $R|_s$ represent the variance matrices vectorized. Also, b represents the estimate of the first column of the correlation matrix S_N which can be estimated as discussed earlier. Several of the matrices in Z are not defined here, but are built from the system model parameters. As is clear here, the size of the programming problem is determined not only by the size of the system (i.e. states, measurements, and random disturbances), but also the chosen N parameter. N can be described as the size of the window of data analyzed for the estimation.

a. ALS Technique for Nonlinear Systems

Generally, state space models are used to describe the dynamic action of internal states of system, in addition to the inputs and outputs (or measurements). Because they are typically more connected to physical interpretations of physical phenomena, systems of differential equations are often used to describe the dynamics of internal states. However, algebraic difference equations can also be used, and are often used in estimation techniques for nonlinear systems because of nice mathematical properties. The ALS method for nonlinear systems assumes that the system being studied can represented by the following set of discrete time equations, and practically, many

systems can be described in this way.

$$x_k = f(x_k, u_k) + g(x_k, u_k)w_k \tag{2.68}$$

$$y_k = g(x_k, u_k) + v_k \tag{2.69}$$

The ALS method presented previously can be modified and applied to systems modelled by nonlinear systems of equations, and time-variant linear systems. In general, the modification consists of linearizing the system at each time step around the state estimate, similar in concept to the procedure of the Extended Kalman filter.

When the system has been linearized, the following property must hold in the limit as k goes to infinity, in order for the ALS method to be applicable.

$$\prod_{i=0}^{k} \bar{A}_i \approx 0 \tag{2.70}$$

Given this property, the derivation of the ALS technique for nonlinear systems follows the same ideas as the linear case. The main difference is the inclusion of this product term, multiplying the linearized system from the initial time until time k. There is a simplification though, where the product is only used from time t_m to time k. This adds an additional parameter to the nonlinear ALS. Once again, the estimation depends on information gained from a window sized N, but also, each piece of information is obtained using t_m number of historical data. This extra parameter is necessary for the nonlinear or time-variant version of ALS because the correlation matrix is changing with time.

CHAPTER III

FAULT DETECTION APPROACH FOR SYSTEMS INVOLVING SOFT SENSORS

A. Introduction

Process monitoring forms an essential element of operating modern industrial processes. Various types of control charts have been developed and extensively used to monitor the variability of process variables such that process faults can be detected. These fault detection techniques generally monitor if the values of process variables are within acceptable parameters, e.g., upper and lower bounds of the process variable value (Isermann, 1997). One challenge for process monitoring is that the important variables, e.g. the concentrations of key components in a reactor, may not be measured directly and the control charts cannot be applied directly to monitor their variability. One solution to this problem is using the measured variables, e.g. the temperature of a reactor, to monitor process conditions. Data from these measured variables can then be used as an indicator of the operating condition if the measured variables are affected by the unmeasured ones. Monitoring measured data directly is a simple way to detect faults; however, this approach is not always trivial as the relationship between the measurements and the process fault may be non-trivial. One result of this is that many multi-variable monitoring techniques, such as principal component analysis or partial least squares, have been developed as usually several measured variables need to be used to determine the state of a process.

Alternatively, the measured process variables can be used to predict the unmeasured ones by employing some form of soft sensor which infers the value of the

unmeasured variables from a combination of data and a system model (Fortuna *et al.* , 2005; Kadlec *et al.* , 2009; Lin *et al.* , 2007). The estimated value can be used for monitoring instead of the original data. This approach offers the benefit of additional information contained in the model. However, this approach needs to take into account that the predicted variables are not only affected by the process operations and measurement accuracy but also by the soft sensor design. For variables that are directly measured, upper and lower control limits are often given as a function of the covariance matrix of the measured variables. However, the situation is more complex if the variables are determined from soft sensors as the covariance matrix of the predicted variables is affected by the soft sensor design and is not equal to the covariance matrix of the original measurements. Instead, the covariance matrix of the estimated variables has to be computed if limits for the operating conditions are to be determined.

As soft sensors begin to replace some plant measurements for process monitoring, it is important to adapt techniques, such as fault detection, to variables that are computed from soft sensors. The concept of using soft sensor predictions directly for process monitoring, instead of comparing soft sensor predictions to actual measurements, is not a new idea (Watanabe & Himmelblau, 1983), however, there seems to be little literature focusing on this concept. This work addresses this point and presents a new, generally applicable approach for fault detection schemes for processes involving soft sensors. Specifically, the presented approach will directly use state estimates of unmeasured process variables to determine if the values of these variables are corresponding to normal operating conditions.

This chapter is organized as follows. Section B presents the description of the process monitoring scheme for linear and, under certain conditions, for nonlinear systems. Next, in Section C, simulation results from the application of the technique

are shown and discussed, including a comparison to standard techniques. Finally, Section D presents the conclusions.

B. Fault Detection Approach Involving Soft Sensors

1. Motivation for Fault Detection Approach

Kalman filters such as the one given by equations (2.10)-(2.11) allow the prediction of the values of unmeasured states of a process. These predicted state values can be used for process monitoring. However, the statistics of the estimated states are different from the statistics of the actual states as measurement noise, model accuracy, and the soft sensor design/tuning affect the statistical properties of the estimated states. It is the purpose of this work to develop a process monitoring technique which can take these factors into account.

The motivation of this work is to develop an approach to fault detection that can be combined with process monitoring strategies that include soft sensors. The strategy employed here determines the mean and covariance matrix of the state estimate from the system model to be used and from the methodology for computing the filter gain. Given these statistical values, thresholds for fault detection are calculated. These thresholds can then be used for fault detection, where one common method uses three times the standard deviation above and below the steady state value, for a control chart threshold. With these thresholds, most of the fault detection techniques currently used for monitoring measured variables can be adapted to the case where soft sensor state estimates are available instead of measurements. This is beneficial primarily because it allows application of fault detection directly on physically meaningful internal states of the system that are unmeasured. Secondly, this approach offers the potential for increasing fault detection performance in cer-

tain circumstances, e.g., for cases where the measurements are nonlinearly related to the process variable to be monitored, but where this relationship is known with a reasonable level of accuracy.

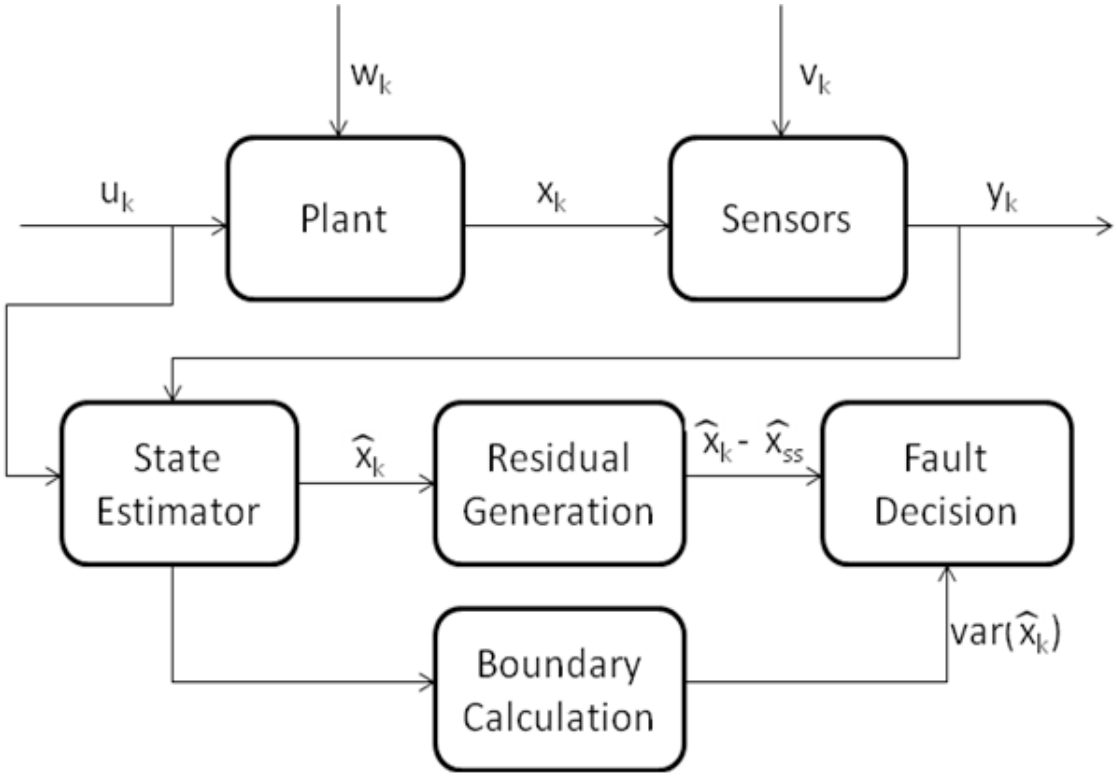


Fig. 2. Flowchart of presented soft sensor-based fault detection scheme

The presented approach is conceptually illustrated in Figure 2. In a first step, the state estimator receives data about the inputs and outputs of the plant, and computes the state estimates. Then, the residual generator compares the state estimates to their expected values. These expected values are determined from operating characteristics of the system and the chosen filter. Finally, a fault detection decision is made by comparing the computed residual to some threshold. It should be pointed out that standard procedures can be used for all of the individual blocks shown in Figure 2,

however, the contribution of this work is to show the integration of these tasks and to use the predicted statistical variations of the state estimates for fault detection. In addition, this work offers a detailed discussion of the threshold calculation required for implementation of the presented approach. This procedure is intentionally kept general, so that it may be used for a variety of available state estimation and fault detection tools.

One key component of this strategy is that the thresholds of the estimates must be determined from the covariance matrix of the state estimates. This covariance matrix will be a function of the system parameters, including the dynamic state model and measurement noises, Q and R.

$$\mathrm{VAR}(\hat{x}) = E[(\hat{x} - E[\hat{x}])(\hat{x} - E[\hat{x}])^T] = f(Q, R) \tag{3.1}$$

In the general case, an estimate of this covariance is attainable via Monte Carlo simulations. However, depending on the type of filter employed, there may be an analytical solution. In the next section this analytical calculation is performed for a linear system using a steady state linear filter, followed by a section detailing the estimation in the general case.

2. Calculation of Variance Matrix for Linear Systems

This section will analytically determine the necessary normal operating statistical values for linear systems with a linear filter. Specifically, the steady state mean and variance of the state estimate vector are calculated. Though these statistics can be calculated from a Monte Carlo simulation, as will be shown in Section 3.3, such an approach requires a large amount of computation time and its accuracy is not guaranteed. The following derivation offers an analytical solution of the desired values for a linear system. One implementation of a linear filter for a stable model given by

equations (2.3) and (2.4) is given by:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \tag{3.2}$$

$$\hat{x}_k = \hat{x}_k^- + L_k(y_k - C\hat{x}_k^-) \tag{3.3}$$

where the A, B, and C matrices are identical to the matrices from equations (2.3) and (2.4) and L refers to the gain. It is further assumed that the data y is collected while the system is at steady state and that the gain is constant.

Equations (2.3), (2.4), (3.2), and (3.3) can be combined into a single equation for the propagation of the error updated state estimate:

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_k + K(CA(x_{k-1} - \hat{x}_{k-1}) + CGw_k + v_k) \tag{3.4}$$

In order to determine the normal operating condition of the process, the expected value of the state estimation model can be taken.

$$E[\hat{x}_k] = (A - LCA)E[\hat{x}_{k-1}] + Bu_k + LCE[y_k] \tag{3.5}$$

The assumption is made that the expected value from one time step the next does not change at steady state. This allows for solving for the steady state, resulting in:

$$E[\hat{x}_{ss}] = (I - A - LCA)^{-1}[Bu_{ss} - LCBu_{ss} + Ly_{ss}] \tag{3.6}$$

$$\hat{x}_{ss} = (I - A - LCA)^{-1}[B - LCB + LC(I - A)^{-1}B]u_{ss} \tag{3.7}$$

Next, the steady state variance of the state estimate needs to be computed. In order to do so, a new term is defined as the estimation error given by the difference between the state estimate and the actual state:

$$e_k = x_k - \hat{x}_k \tag{3.8}$$

The linear filter as given by equation (3.4) can be written in terms of the estimation error, and the estimation error can be formulated in terms of its own dynamic recursive equation:

$$\hat{x}_k = A\hat{x}_{k-1} + LCAe_{k-1} + Bu_k + LCGw_k + Lv_k \tag{3.9}$$

$$e_k = x_k - \hat{x}_k = Ae_{k-1} - LCAe_{k-1} + Gw_k - LCGw_k - Lv_k \tag{3.10}$$

A new state vector that includes the estimated state and the estimation error can be created, resulting in the following system:

$$z_k = \begin{pmatrix} A & LCA \\ 0 & A - LCA \end{pmatrix} z_{k-1} + \begin{pmatrix} B \\ 0 \end{pmatrix} u_k + \begin{pmatrix} LCG & L \\ G - LCG & -L \end{pmatrix} \begin{pmatrix} w_k \\ v_k \end{pmatrix} \tag{3.11}$$

$$z_k \equiv \begin{pmatrix} \hat{x}_k \\ e_k \end{pmatrix} \tag{3.12}$$

$$\Pi_k = \Lambda\Pi_{k-1}\Lambda^T + \Gamma\Phi\Gamma^T \tag{3.13}$$

$$\Lambda \equiv \begin{pmatrix} A & LCA \\ 0 & A - LCA \end{pmatrix} \tag{3.14}$$

$$\Gamma \equiv \begin{pmatrix} LCG & L \\ G - LCG & -L \end{pmatrix} \tag{3.15}$$

$$\Phi \equiv \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix} \tag{3.16}$$

$$\Pi_k = \begin{pmatrix} \mathrm{VAR}[\hat{x}_k] & \mathrm{COV}[\hat{x}_k e_k^T] \\ \mathrm{COV}[e_k \hat{x}_k^T] & \mathrm{VAR}[e_k] \end{pmatrix} \tag{3.17}$$

Using the form shown in equation (3.11), the variance of the state vector, including the variance of the state estimates, can be found by the discrete time Lyapunov

equation (3.13). The steady state variance of the state estimates can be determined by the steady state solution of the Lyapunov equation (3.13). It is worth noting that if the gain is the steady state Kalman Filter gain, the variance of the estimate error matrix (lower right block of Π_k) will be equivalent to that automatically produced by the KF (P_k in equation (2.14)), and the covariance between the state estimate and the estimate error will be a matrix of zeros. However, if a case arises where the chosen gain is not the KF gain, then the variance calculated using this method still provides the state estimate variance. In other words, whether or not the gain is the Kalman Filter, the upper left block of Π_kk produces the variance of the state estimates as desired. Moreover, this approach for calculating the variance matrix for linear systems is applicable to any general linear system of the form of equations (2.3) and (2.4) with any chosen linear filter.

An alternative derivation for calculating the same value without having to solve the Lyapunov equation is shown in the Appendix. It can easily be seen from the equations provided in the Appendix that the variance of the state estimate is linearly dependent on the variance of the random input to the system.

Given the model of the system and information about the noise term, the steady state solution of equation (3.13) provides the variance of the state estimate, and equation (3.5) provides the steady state mean. The mean and covariance matrix can be used to determine the upper and lower threshold limits for fault detection in process monitoring. These control limits can be used for monitoring the values of the predicted state variables in a similar manner as one could use the statistics of the actual states if these were measured. It is one of the contributions of this work to derive how these control limits of the predicted states can be computed.

3. Calculation of Variance Matrix for Nonlinear Systems

In the previous section, the expected mean and variance of a given linear system with a linear filter was explicitly determined. This allows for calculation of thresholds for process monitoring as described in Section 3.1. However, given a general nonlinear system, where one of the many possible filters for nonlinear systems is applied, a general formula for calculating the covariance matrix is not available. Instead, it is possible to determine estimates of these values using a Monte Carlo approach. Assume that the system model and state estimator are given by the following equations:

$$x_k = f(x_{k-1}, u_k, w_k) \tag{3.18}$$

$$y_k = h(x_k, u_k) + v_k \tag{3.19}$$

$$\hat{x}_k = g(\hat{x}_{k-1}, y_k, f, h, Q, R) \tag{3.20}$$

where all variables are defined as previously except f, h, and g represent general nonlinear dynamic state, measurement and recursive state estimator equations. The procedure simulates these equations for long periods of time where the system is driven by the random inputs wk and vk. This will produce a matrix of data of dimension ns x nt, where ns refers to the number of states and nt corresponds to the number of data points simulated. The state estimate variance can then be estimated using the following equations:

$$\hat{\mu}_{\hat{x}} \approx \frac{1}{n_t} \sum_{i=1}^{n_t} \hat{x}_i \tag{3.21}$$

$$\text{VAR}[\hat{x}] \approx \frac{1}{n_t} \sum_{i=1}^{n_t} [(\hat{x}_i - \hat{\mu}_{\hat{x}}) \times (\hat{x}_i - \hat{\mu}_{\hat{x}})^T] \tag{3.22}$$

The described approach has the advantage that it is general in nature. However, drawbacks of this approach include significant computational effort and that it is

difficult to determine just how much data is required to obtain sufficient accuracy for a proper threshold.

One (sub-optimal) alternative to the technique presented in this subsection can be used if an Extended Kalman Filter is employed for state estimation. Since the Extended Kalman Filter is an extension of the Kalman Filter, an estimate of the covariance of the state estimate can be calculated using the procedure described in Section 3.2. The system parameter matrices used in the calculation should be determined as follows, and evaluated at the system steady state:

$$A = \frac{df}{du^T} \tag{3.23}$$

$$B = \frac{df}{dx^T} \tag{3.24}$$

$$G = \frac{df}{dw^T} \tag{3.25}$$

$$C = \frac{dg}{dx^T} \tag{3.26}$$

$$d = \frac{dg}{du^T} \tag{3.27}$$

The steady state Kalman filter gain should be used for L, which can be determined by solving the algebraic Riccati equation:

$$P_{ss}^{-} = AP_{ss}^{-}A^T - (AP_{ss}^{-}C^T)(CP_{ss}^{-}C^T + R)^{-1}(CP_{ss}^{-}A^T) + GQG^T \tag{3.28}$$

$$L = K_{ss} = P_{ss}^{-}CT(CP_{ss}^{-}C^T + R)^{-1} \tag{3.29}$$

These matrices can be used as part of the Lyapunov equation (3.13), which can then be solved at steady state.

C. Case Studies

This section illustrates the presented technique via several case studies. The methodology is applied to two linear examples in Section 1 and a nonlinear system in Section b. An illustrative comparison between results returned by the presented technique and a traditional fault detection method is made in Section 3.

1. Application to Linear Systems

This subsection presents two case studies: the first example investigates process monitoring of an isothermal CSTR while the second one deals with a linearized model of a distillation column.

a. Isothermal CSTR

Assume that a series reaction is occurring in an isothermal reactor where the first reaction is reversible while the second reaction is irreversible. The model equations are given by:

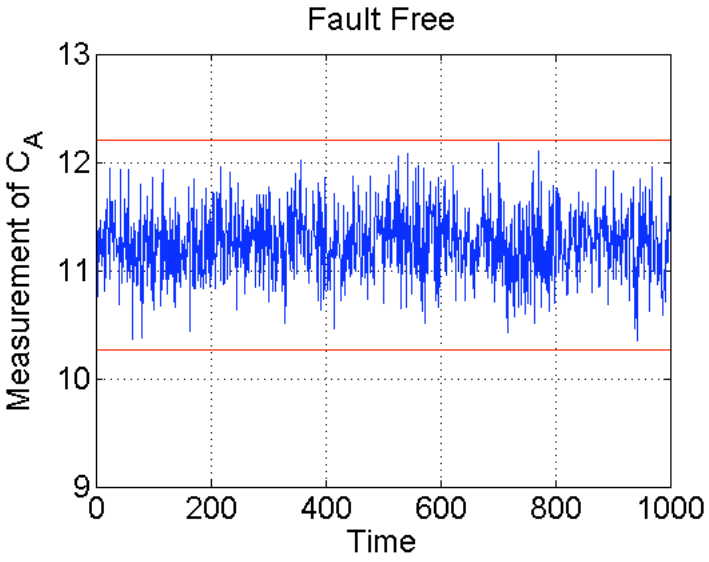
$$
\begin{aligned}
A &\overset{k_f,k_r}{\longleftrightarrow} B \xrightarrow{k_2} C \\
\frac{dC_A}{dt} &= -k_f C_A + k_r C_B + F C_{Ain} - F C_A + w_1 \\
\frac{dC_B}{dt} &= k_f C_A - k_r C_B - k_2 C_B - F C_B + w_2 \\
\frac{dC_C}{dt} &= k_2 C_B - F C_C + w_3 \\
y &= \begin{bmatrix} C_A \\ C_B \\ C_C \end{bmatrix} + v
\end{aligned}
\tag{3.30}
$$

In this example, the concentration of A is monitored and used as an indicator for process faults. The system parameters were chosen as $k_f = 400$, $k_r = 10$, $k_2 = 100$,

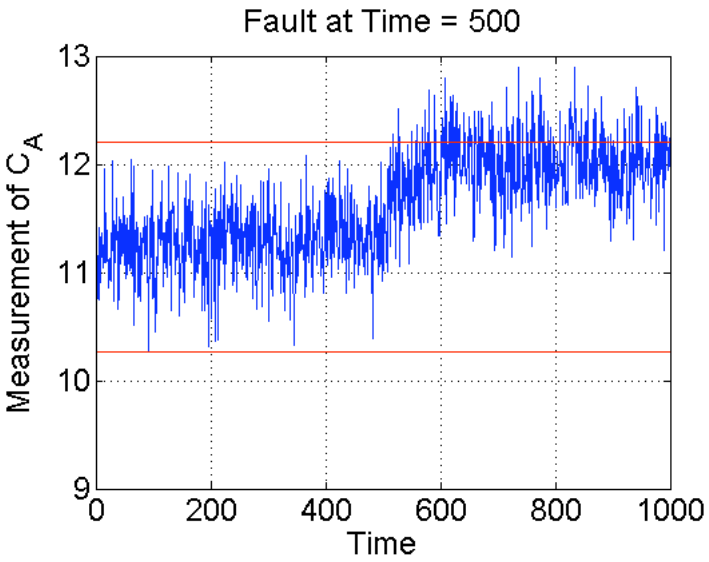
$F = 2000$, and $C_{A,in} = 10$. For this simulation both process and measurement noise is used, $w \in R^3$ and $v \in R^3$, respectively. The noise terms are Gaussian with distributions given by $N \sim (0, 0.1I_3)$ for w and$N \sim (0, I_3)$ for$v$. Given these random inputs the process operates with expected noise levels around its steady state. Threshold values of 3σ, i.e. 3 times the standard deviation are used for fault detection. In this case, $\sigma = 0.323$. A typical snapshot of the control chart is shown in Figure 3. It can be seen that the process is operating normally as the monitored variable C_A remains within its bounds. More specifically, the process was simulated for long periods of time in order to observe that the theoretically predicted 99.7% of points are approximately within these boundaries (Montgomery *et al.* , 2009).

Next, the same process is simulated again, but this time a fault is introduced. The flow rate into the reactor is increased by 7% at time 500, simulating an upstream problematic valve. The fault can clearly be seen in Figure 3(b) as the process variable is outside of the fault boundaries. The situation changes, however, if the concentration of A would not be measured, i.e., the process variable is still important for process monitoring, but the measurement is not available. In order to illustrate this point, a Kalman Filter is implemented to infer the value of this process variable.

When the same bounds that were derived for the measured variables are applied to the predicted variables, it is not possible to detect this fault. This scenario is illustrated in Figure 4, where the red lines represent the upper and lower control limits based upon the statistics of the concentration of species A if it were directly measured. However, if the fault boundaries are recomputed based upon the presented approach, they are found to be much smaller, specifically $s = 0.03$. Using these new bounds, which are shown by the black lines, the state estimates can be used to accurately predict the faulty behavior.



(a) Fault free process monitoring



(b) Fault present

Fig. 3. Typical control chart snapshot

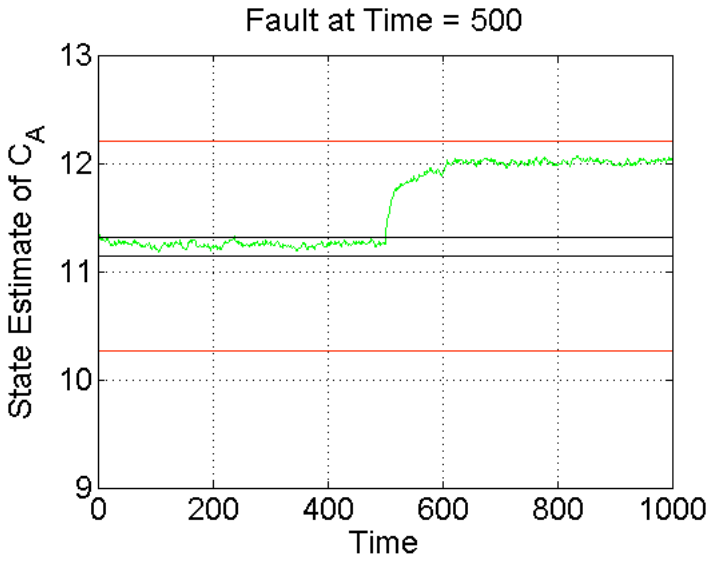


Fig. 4. State estimate (green) fault detected by calculated boundaries (black), but not by boundaries based upon statistics of the measurement of C_A (red)

b. Distillation Column with 30 Trays

A model of a distillation column with 30 trays, a full condenser, and a reboiler is shown below (Hahn & Edgar, 2002).

$$
\begin{aligned}
\frac{dx_{A,1}}{dt} &= \frac{V}{A_{condenser}}(y_{A,2} - x_{A,1}) \\
\frac{dx_{A,i}}{dt} &= \frac{1}{A_{tray}}[L_1(x_{A,i-1} - x_{A,i}) - V(y_{A,i} - y_{A,i+1})], \text{for } \; i = 2, ..., 16 \\
\frac{dx_{A,17}}{dt} &= \frac{1}{A_{tray}}[Fx_{A,Feed} + L_1 x_{A,16} - L_2 x_{A,17} - V(y_{A,17} - y_{A,18})] \\
\frac{dx_{A,i}}{dt} &= \frac{1}{A_{tray}}[L_2(x_{A,i-1} - x_{A,i}) - V(y_{A,i} - y_{A,i+1})], \text{for } \; i = 18, ..., 31 \\
\frac{dx_{A,32}}{dt} &= \frac{1}{A_{reboiler}}[L_2 x_{A,31} - (F - D)x_{A,32} - V y_{A,32}] \\
\alpha_{AB} &= \frac{y_A(1 - x_A)}{(1 - y_A)x_A} \\
y &= [x_{A,1} \;\; x_{A,17} \;\; x_{A,32}]^T
\end{aligned}
\tag{3.31}
$$

The simulation and filter used in this section make use of a linearization of this model around the steady state corresponding to: $\alpha_{AB} = 1.6$, $x_F = 0.5$, $x_D = 0.935$, $x_B = 0.065$, and $\frac{L_1}{D} = RR = 3.0$. As with the previous case study, additive noise enters both the dynamic state equation and the measurement equation with variances of 10-5 and 10-1. In this study, only the concentrations of the feed and the two exits of the column are measured. This is an attempt to mimic the realistic situation in which internal column measurements are usually impractical and unavailable.

Despite the limited number of measurements, the presented approach has been used to obtain thresholds for fault detection for all of the unmeasured states. One representative result of this is discussed next. In this example the measurement noise variance is significantly increased from $10e^{-3}$ to $10e^{-1}$ at time step 500. Figure 5(a) shows the process monitoring bounds for the concentration of species A on tray 20.

The blue curve represents the value of the measured variable, and the red bounds are the 3σ bounds for that variable, here $\sigma = 0.032$. It can be clearly seen that the measurement after time step 500 is not accurate as the majority of the measurements are outside of the 3σ bounds. However, if an observer is used then the estimated state trajectory is shown as the green curve in Figure 5(a). The state estimate clearly remains inside the bounds determined from the original measurements. However, these bounds will need to be modified as the variance of the state estimate is different from the variance of the measurement of this state, now $\sigma = 0.004$. The updated 3σ bounds are given by the black lines in Figure 5(b). It can clearly be seen that the technique correctly identifies a fault, albeit a measurement and not a process fault, using these updated upper and lower control limits.

2. Application to a Nonlinear System

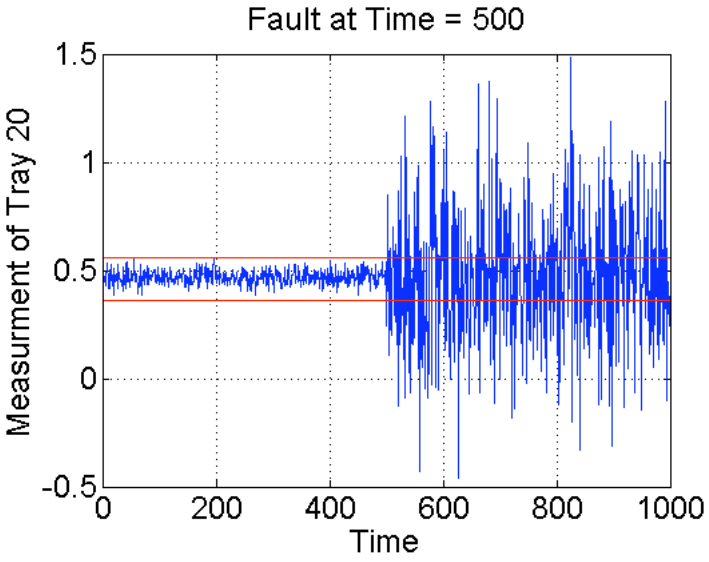
The nonlinear system under investigation in this subsection deals with a non-isothermal CSTR in which a van de Vusse reaction is taking place:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

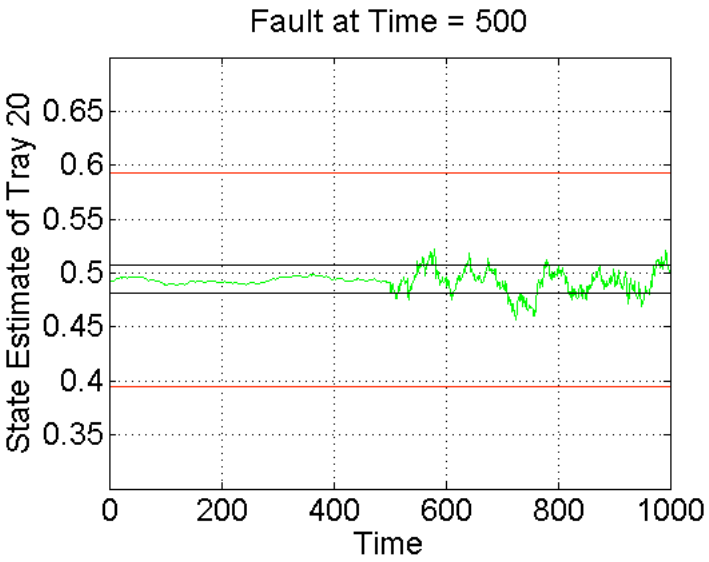
$$2A \xrightarrow{k3} D$$

The equations governing this CSTR are given by

$$\begin{aligned}
\frac{dC_A}{dt} &= \frac{F}{V}(C_{A,in} - C_A) - k_1 e^{\frac{-E_1}{RT}} C_A - k_3 e^{\frac{-E_3}{RT}} C_A^2 \\
\frac{dC_B}{dt} &= -\frac{F}{V} C_B + k_1 e^{\frac{-E_1}{RT}} C_A - k_2 e^{\frac{-E_2}{RT}} C_B \\
\frac{dT}{dt} &= \frac{1}{\rho c_p}[+k_1 e^{\frac{-E_1}{RT}} C_A(-\Delta H_1) + k_2 e^{\frac{-E_2}{RT}} C_B(-\Delta H_2) + k_3 e^{\frac{-E_3}{RT}} C_A^2(-\Delta H_3)] \\
&\quad + \frac{F}{V}(T_{in} - T) + \frac{Q}{V \rho c_p}
\end{aligned} \tag{3.32}$$



(a) Monitoring measured variable



(b) Monitoring unmeasured variable

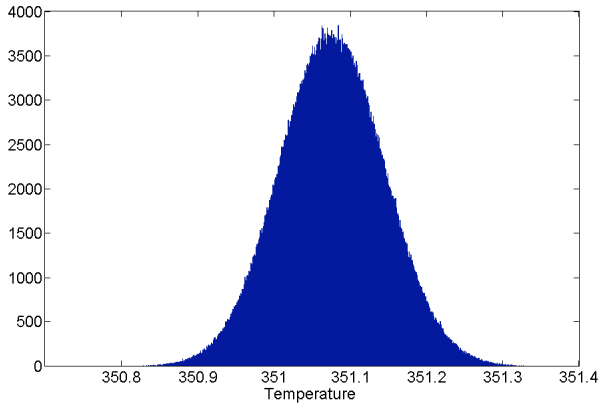
Fig. 5. Simulation of fault in distillation column simulation

and the values of the parameters for this model can be found in (Hahn & Edgar, 2001). The only input is the volumetric flow into the reactor (F), and it was chosen as 25 L/hour so that the system is operating close to the most nonlinear region. The initial conditions are given by the steady state operating point corresponding to this nominal input value, namely xo = [2.98, 0.905, 351]T. For simulation of this nonlinear system, the random state model noise and measurement noise entered the system according to the following general equation.

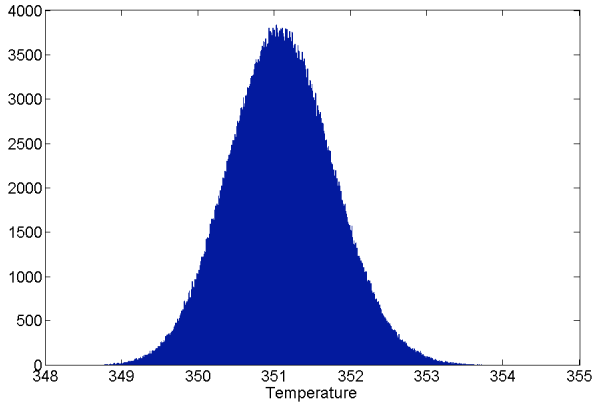
$$\dot{x} = f(x, u) + Gw \tag{3.33}$$

$$y(t) = Cx(t) + v \tag{3.34}$$

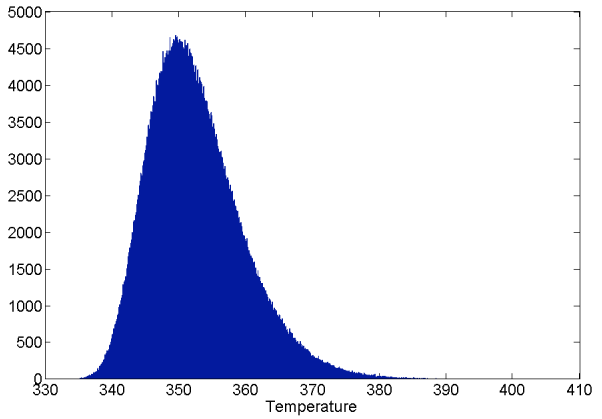
Here the noise enters the system in a linear fashion and is assumed to be taken from a white noise Gaussian distribution. However, this does not guarantee that the distribution of the states will remain Gaussian as the distribution is affected by the nonlinear function f. In situations with relatively small noise variances a Gaussian approximation is often appropriate. For this system, Monte Carlo simulations were used and it was found that the state vector appears Gaussian in shape for realistic values of the model noise covariance. Figures 6(a) - 6(c) shows the histogram of the temperature for simulations of the presented nonlinear system for 106 time units, with state model noise variances of $10e^{-3}$, $10e^{-1}$, and $10e^{1}$, respectively. Even though the probability distribution starts to take on a nonsymmetrical shape, this only happens at temperatures changes which are outside of a realistically acceptable range, e.g., resulting in temperature variations of 40 degrees or more. Therefore, it can safely be assumed that each state follows a Gaussian distribution for small values of the variances in the noise, such as $10e^{-3}$.



(a) $\sigma = 10e^{-3}$



(b) $\sigma = 10e^{-1}$



(c) $\sigma = 10e^{1}$

Fig. 6. Distribution of temperature state variable from varying model noise variance

The noise levels for this investigation were chosen as

$$Q = \begin{bmatrix} 10^{-3} & 0 & 0 \\ 0 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix} \tag{3.35}$$

$$R = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-4} \\ 0 & 0 \end{bmatrix} \tag{3.36}$$

for the state disturbance and sensor noise covariance matrices, respectively. A process monitoring scheme was applied to this reactor for the unmeasured concentration of species A for two different cases: 1) using the bounds calculated from the variance of directly measuring C_A and 2) using the bounds determined by the presented approach. For both cases, the only measurement is the reactor temperature. This mimics many practical situations in which the concentrations in a reactor are important for monitoring the state of the system, but concentrations are not measured due to online concentration measurements being non-trivial. One representative result of this simulation is shown in Figure 7. A fault of a 1% increase in the inlet flow was introduced at time step 500. It can be seen that the bounds computed from the direct measurement of C_A (given by the red lines, $\sigma = 0.014$) cannot detect the fault, whereas the bounds computed by the presented technique (given by the black lines, $\sigma = 0.006$) accurately detect the fault.

3. Comparison with Techniques that only Use Available Measurements

One question that naturally arises in the context of this paper is how the presented technique compares to existing approaches that directly use measured data for

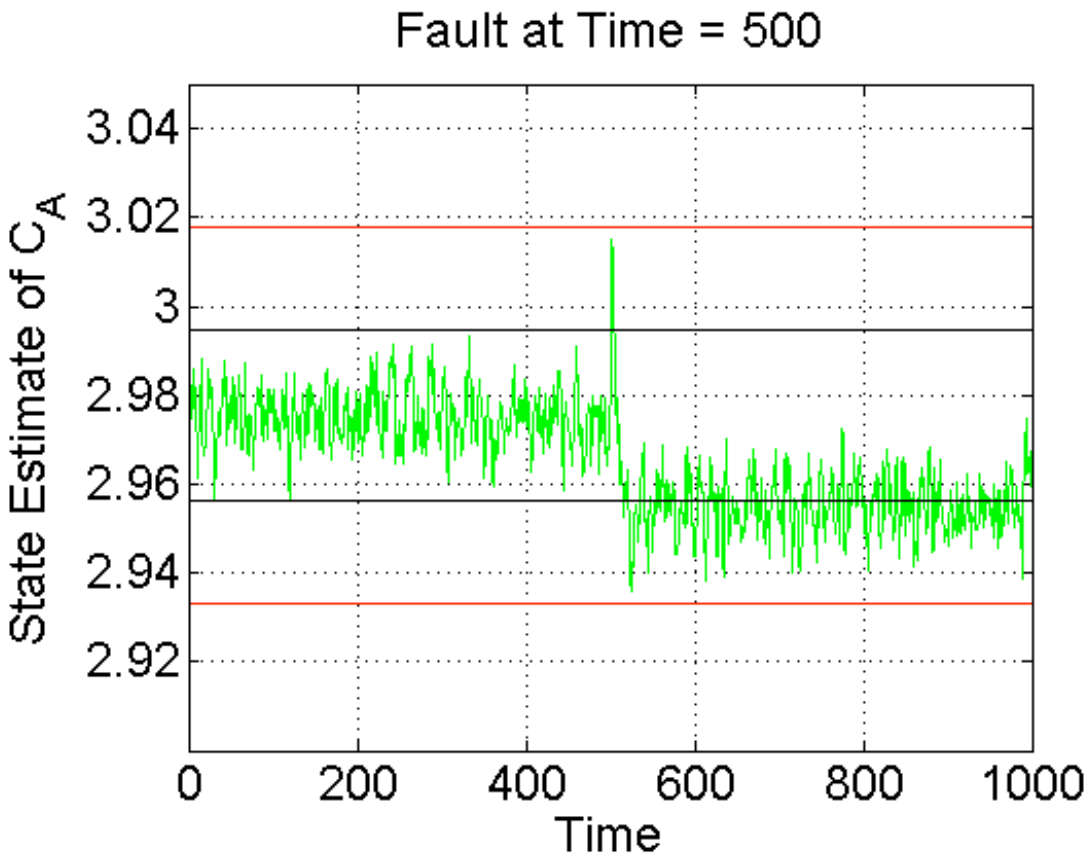


Fig. 7. Fault detected while monitoring state estimate (green) with calculated bounds (black)

fault detection. As discussed previously, most currently implemented process monitoring techniques perform fault detection on a residual calculated from the current data. The difference between these classical approaches and the presented technique is that, besides data, the model included in the soft sensor is also used in this work. A soft sensor technique can sometimes take advantage of the model resulting in a more rapid and accurate fault detection. A detailed comparison of the results that can be obtained for the example shown in the previous subsection is performed here.

In the case study dealing with the linear isothermal CSTR, faults were assumed

to be sufficiently large so that they can be detected by visual inspection of the process monitoring charts. However, not all faults can be detected easily by such an approach. Suppose the fault is a change in the raw material concentration, i.e., in this case the inlet concentration C_A is assumed to increase by 2%. The concentration C_C is measured and the increase in C_A is reflected in the variations of the measured data. However, the measurements are relatively insensitive to the fault and monitoring data will cause a delayed detection of the fault or even miss it. On the other hand, fault detection via monitoring estimated state values can result in improved performance. To illustrate this point, the CSTR model is simulated for 1,000 time steps after the fault occurs. Normally there are about 3 out of 1,000 points beyond the 3σ bound (.27%). However, there will be significantly more points outside of the bounds if a fault has occurred.

To compare the performance of fault detection methods, two criteria are calculated. The first one is the time at which the 4th point beyond the bounds appears. The appearance of the point is an indicator for presence of the fault and the earlier it is found the more quickly the fault is detected. Due to the stochastic property of the system the time of appearance of the 4th point will change for different runs. In order to address this point, a Monte Carlo simulation with 10,000 runs is applied. Figure 8(a) shows the distribution of the time point when the original measured data or the estimated state values are monitored. It can be seen that monitoring state estimates, even of unmeasured states, can detect the fault more quickly than monitoring the original data.

Another criterion to compare the performance of fault detection methods is the total number of points which are outside of the normal operating bounds over the entire simulation interval. The more points that appear outside of the bounds, the stronger the evidence for detecting the fault. Since the number is also dependant on

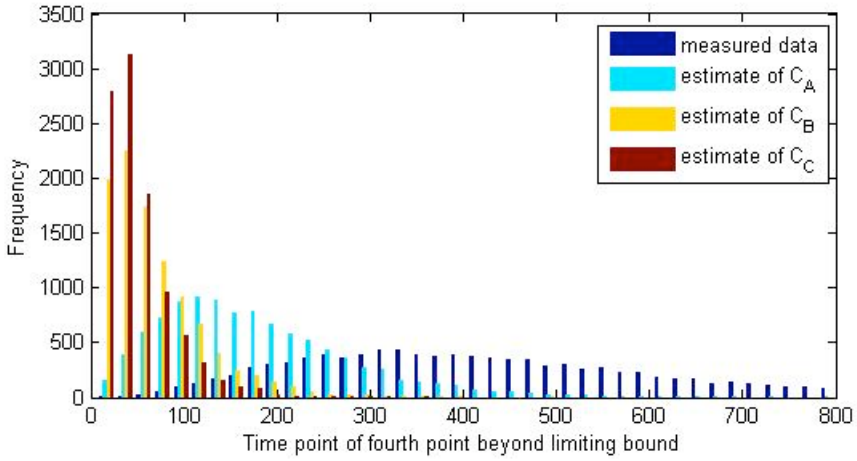
the noise added in a single run, 10,000 Monte Carlo simulations have been used to compute the distribution of the numbers shown in Figure 8(b). It can be seen that there are more points of the estimated state values that appear outside of the bounds than points of the measured variables that appear outside of their respective bounds. Figure 8(b) also indicates that the estimate of C_C has more time points at which the bounds are violated than if the estimates of any of the other states are used.

In order to investigate that an increase in the sensitivity of the state estimate to faults does not result in a large number of false positives, another type of assessment has also been used for comparing the presented approach to traditional fault detection techniques.

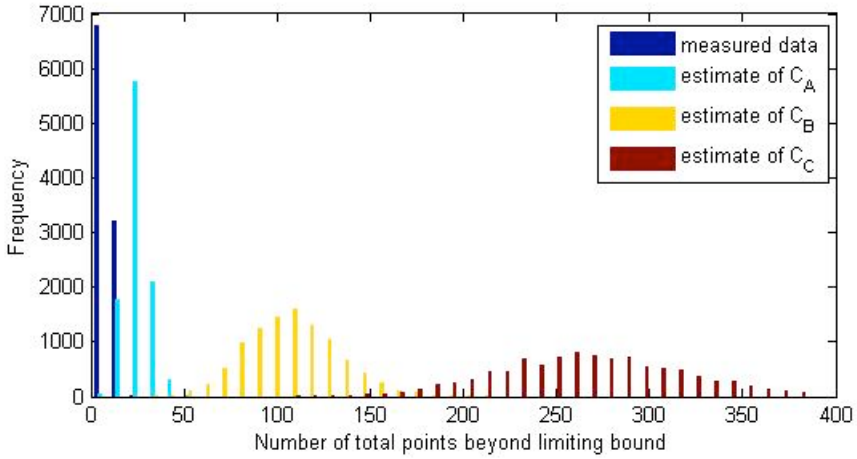
For this investigation, the model of the nonlinear CSTR was simulated for 100 time steps with a deterministic fault being introduced at the beginning of the simulation. This simulation is repeated 1,000 times with the fault being randomly generated from a uniform distribution between 0 and 1, for each simulation. This additive type of fault model represents a situation such as a leak or certain types of environmental changes.

A fault of less than 0.04 was chosen as the fault detectability limit. A different choice of this threshold will simply affect the number of false positives versus false negatives and the total percent of cases in the No Fault category; the threshold can be selected for desired results. In each simulation a 99.99% confidence level hypothesis test on the mean was performed on the data to determine if a fault was present or not present. Table I below shows the results in the case where the fault is affecting C_A. The only variable measured is the temperature.

The first section of Table I, labeled State Variable, shows the results for performing the hypothesis test directly on the internal states. The next two sections show the performance when using the measurements and state estimates, respectively. Each



(a) The time point at which the 4th point appears beyond the bound



(b) The number of total points beyond the bound in the simulation interval

Fig. 8. Distribution of the monitored value in Monte-Carlo simulation

value represents the percentage of simulations in which a fault was detected, or not detected. For example, the upper left hand entry shows that 96% of the times where there was truly a fault present, a fault was detected.

It can be seen that when fault detection is performed on the state estimate, the performance is as good as the case in which the actual measurement is used. If monitoring is done on either concentration of species A or B, the only option would be to perform the fault detection using the state estimate. If it is desired to monitor temperature directly, the state estimate shows equivalent results to that of the measurement. This is due to the process in which the state value is estimated by the soft sensor. Information from the available measurement is used to filter the values and reconstruct the others. In this way, information about the fault is contained in all of the state estimates.

While these types of results cannot necessarily be expected for all nonlinear systems, the technique performed well in this case as well as all other ones to which the methodology was applied.

Table I. Comparison of fault detection performance when limit checking is performed on measurements and on state estimates

State Variable	C_A		C_B		T	
	Fault	No Fault	Fault	No Fault	Fault	No Fault
Fault Detected	96	0	89	0	99	0
Fault Not Detected	4	100	11	100	1	98
Measurement	C_A		C_B		T	
	Fault	No Fault	Fault	No Fault	Fault	No Fault
Fault Detected	N/A	N/A	N/A	N/A	96	0
Fault Not Detected	N/A	N/A	N/A	N/A	4	100
State Estimate	C_A		C_B		T	
	Fault	No Fault	Fault	No Fault	Fault	No Fault
Fault Detected	96	0	96	0	96	0
Fault Not Detected	4	100	4	100	4	100

D. Summary and Conclusions

This chapter developed an approach for process monitoring and fault detection involving soft sensors. The process monitoring strategy compares state estimates to bounds that represent state values derived from normal operation of the process. It is important to point out that these bounds are different from the bounds that would be derived if the states were measured instead of estimated. The reason for this is that the soft sensor design has a direct impact on the variance of these state estimates. Accordingly, the bounds need to be adjusted based upon the soft sensor design. While this may seem like a trivial point, this point has, to the best of the knowledge of these authors, not been addressed in the literature.

The approach presented here was kept general such that it can be used regardless of the employed type of soft sensor. As a special case, the equations for a linear system where a linear filter is used for state estimation have been derived. In addition, an approximation of the bounds for use with a nonlinear system using the EKF was shown. The technique was illustrated by applying it to three examples and also performing comparisons to standard process monitoring schemes which only use process data. The presented approach of performing limit checking on physically meaningful state estimates of unmeasured variables, rather than simply on available measurements only, was shown to offer improved performance.

CHAPTER IV

SENSOR LOCATION FOR NONLINEAR DYNAMIC SYSTEMS VIA OBSERVABILITY ANALYSIS AND MAX-DET OPTIMIZATION

A. Introduction

Accurate and current values of process variables are essential in order for manufacturing plants to maximize profits and meet all safety, health, and environmental requirements. Specifically, proper data collection is crucial for many process systems techniques including control, state and parameter estimation, and fault detection. Due to the sometimes extremely large number of important process variables and measurement difficulty often associated with them, the objective to know the state of the plant is in competition with the capital and maintenance costs of additional sensors. Therefore, optimization of the placement of sensors is a vital part of plant design and operation.

Sensor network design research based on observability, or sensitivity of the system outputs to the initial state values, as a metric is not new in the literature. Classically, the use of a scalar function of the observability gramian to determine the best sensor network for linear systems has been proposed (Mller & Weber, 1972). Several scalar metrics which are focused on one key part of the matrix were used including smallest eigenvalue, condition number, lower bound for rank deficiency, spectral norm, and trace (Georges, 1995; Van den Berg *et al.* , 2000; Waldraff *et al.* , 1998). Linear systems analysis has the obvious drawback of only being applicable to real systems over a small range of operation. Thus, efforts turned to attempting to determine optimal sensor network designs for state estimation of nonlinear systems (Lopez & Alvarez, 2004; Vande Wouwer *et al.* , 2000). The use of the empirical observability gramian,

an approximation of the observability gramian for nonlinear systems, was shown to be effective in applying previous work for linear systems to dynamic nonlinear systems (Singh & Hahn, 2005a, 2006).

Despite this progress, most all of this research is resigned to using scalar metrics which only use some small part of the information available in the observability matrix, as mentioned above. The determinant of the observability matrix is mentioned as useful, but often not incorporated into the final approach. This may be due to the complexity that the determinant causes in the final problem formulation. However, the determinant advantageously includes all of the terms in the matrix. In addition, in the related field of optimal experimental design, the D(eterminant)-optimality criterion has become one of the most commonly used metrics (Atkinson *et al.* , 2007; Joshi & Boyd, 2009; Uci ski, 2005; Vandenberghe *et al.* , 1998). Using this method, the determinant of the Fischer Information Matrix is used to rank different arrangements of sensors in order to obtain the best experimental design.

This paper mainly provides two advances in this effort, and in so doing creates a generic approach for finding the optimal sensor network design for nonlinear systems. In order to choose the best sensor network, a metric must be defined. In this work, the determinant of the empirical observability gramian is chosen. However, for systems that are unobservable or marginally observable, this metric creates numerical problems. As a way of producing meaningful results, state space reduction is performed on the matrix before the determinant. Finally, this paper incorporates methods from the optimization literature for efficiently solving the mixed integer nonlinear programming problem (MINLP) that results from the maximization of the determinant. This combination of proposed approaches utilizes the information provided by the observability gramian in order to determine the best sensor network design.

This chapter is organized as follows. Section B defines the sensor network prob-

lem to be solved and discusses the optimization problem that results, including procedures for its solution. Section C includes results from case studies and discussion of the results, followed by a conclusion section.

B. Formulation of Sensor Location Problem: Determinant of Gramian

In this work, a general procedure is outlined for producing the optimal sensor network design for general nonlinear systems described by equations (2.37) and (2.38). This procedure uses the determinant of the empirical observability gramian as the metric for the sensor network design. In addition, a methodology for dealing with nearly unobservable systems through model reduction is presented, and an algorithm for solving the resulting mixed integer nonlinear programming problem is introduced.

In general, the objective of this work is to maximize the determinant of the observability gramian by varying the network of sensors placed. As mentioned above, there are several scalar functions to choose from, but the determinant is useful for several reasons. First, the determinant includes information from all elements of the matrix. This is important, in general, for properly ranking the networks. More importantly, however, is the impact on information redundancy when placing multiple sensors. The nave application of many other available scalar functions works well for placing a single sensor, but fails to capture information redundancy present in the system when placing multiple sensors. The determinant takes this information into account as it is a function the entire matrix of values, including covariance terms. Based on this choice of the determinant as the scalar metric, the problem can be formulated mathematically as follows:

$$\max_{\chi} \det\left[W_o\left(\chi\right)\right] \tag{4.1}$$

Here, χ is used to represent the set of sensors chosen in the network. In other words, the goal is to maximize the determinant of the observability matrix W_o, by varying the set of sensors to be placed in the system, χ. In order to create observability gramians for the placing multiple sensors, the matrices are added according to the following equation.

$$W_o(\chi) = \sum_i \chi_i W_{o,i} \tag{4.2}$$

Here, χ_i is a binary variable set to 1 if a sensor is selected to be placed at location i and 0 otherwise. For example, if it is desired to place sensor at state 2 and 5, the observability gramian would be the sum of $W_{o,2}$ and $W_{o,5}$. This procedure effectively creates observability gramians for each set of sensors in a network by adding the information from each sensor individually and the information available from the combination of sensor found in the covariance terms. One drawback to the selection of the determinant of the empirical observability matrix is the extremely small values produced from near singular matrices. This occurs often in systems with many states, in situations where few are being measured. In order to allow for comparison of reasonable values, the presented method uses a simple matrix reduction technique to strip out only the most important information from the almost unobservable system. Simple dimension reduction via singular value decomposition gives the following set of equations.

$$W_o = USV^T \tag{4.3}$$

$$W_(o, reduced) = U_{1:p} W_o U_{1:p}^T \tag{4.4}$$

Where the first p columns of U are used to reduce the size of W_o from $n \times n$ to $p \times p$ to include only the state space directions of the largest p singular values.

Technically this allows for extracting the most important information from a matrix. Using equations (4.2) and (4.4) the problem formulation is now written as:

$$\max_{\chi} \det \left[\sum_{i=1}^{m} \chi_i U_{1:p} W_{o,i} U_{1:p}^T \right] \tag{4.5}$$

In summary, the procedure for finding the optimum sensor network design is as follows. First, empirical observability gramians are calculated where it is assumed that only one state is measured at a time. In other words, for a system with n states, n gramian matrices of size nxn are calculated. This is done according to equations (2.39)-(2.40), with q = 1. These matrices represent the observability gramian for each possible case of the system where only one sensor is placed. Next, the gramians are reduced in size based on analysis of the singular values of the observability gramian calculated for the case where all sensors are present, as in equations (4.3)-(4.4). These matrices are then used in the problem formulation defined in equation (4.5), which results in a Mixed Integer Nonlinear Programming problem, or MINLP.

Typically, MINLP problems are difficult to solve, however, a second primary advantage to using the determinant as the chosen scalar function is that the optimization is related to a well known class of convex problems, specifically MAX-DET optimization (Vandenberghe *et al.* , 1998). This property of guaranteed convexity leads to a tractable solution procedure. In order to apply this approach, equation (4.5) is rewritten in the following equivalent way, according to the standard presented in equation

(2.32).

$$\begin{aligned} P : \min \log & \left\{ \det \left[\left(\sum_{i=1}^{m} \chi_i U_{1:p} W_{o,i} U_{1:p}^T \right)^{-1} \right] \right\} \\ \text{s.t.} \quad & \sum_{i=1}^{n} \chi_i \geq r \\ \chi \in & \{0,1\} \end{aligned} \tag{4.6}$$

Here, r is the maximum number of sensors to be placed. The reduction matrices $U_{1:p}$ are assumed to be included in the problem formulation. For convenience, this problem is labeled P. Here it is noted that each gramian sub-matrix $W_{o,i}$is positive semi-definite such that $z^T W_{o,i} z = 0$ for all nonzero z. This property guarantees that the continuous relaxation of the objective function from equation (4.6) is convex (Vandenberghe *et al.* , 1998). Therefore, this problem can be solved using the global optimization technique found in (Bonami *et al.* , 2008), which is explained in more detail in the remainder of this section.

The following notation is introduced to refer to the objective function from Equation (4.6) as $G(\chi)$ and its derivative defined over the continuous domain $\chi_i \in \{0,1\}$, excluding the point where $\chi_i = 0$ for all i as $\bigtriangledown_j G(\chi)$.

$$G(\chi) = \log \left\{ \det \left[\left(\sum_{i=1}^{m} \chi_i W_{o,i} \right)^{-1} \right] \right\} \tag{4.7}$$

$$\bigtriangledown_j G(\chi) = - \operatorname{trace} \left[\left(\sum_{i=1}^{m} \chi_i W_{o,i} \right)^{-1} W_{o,j} \right] \tag{4.8}$$

$$\tag{4.9}$$

Now a mixed-integer linear programming problem can be formulated which serves as an outer approximation to the original MINLP from P. For convenience, this formu-

lation is labeled $P^{OA}(T)$, or the outer approximation of P at T.

$$\begin{aligned} P^{OA}(T) : \min_{\chi} \; & \alpha \\ \text{s.t.} \quad & \bigtriangledown G(\bar{\chi})^T(\chi - \bar{\chi}) + G(\bar{\chi}) \leq \alpha, \forall \bar{\chi} \in T \\ & \sum_{i=1}^{n} \chi_i \leq r \\ & \chi \in \{0,1\} \end{aligned} \tag{4.10}$$

Here T is some set of points, not necessarily feasible to P, where linearizations of the original objective function from equation (4.6) are generated. Any optimal solution of $P^{OA}(T)$ provides a lower bound of the optimal solution of P. This is illustrated in Figure 9 below.

This figure shows that, for any set of linearization points T, $\alpha_i = G(\chi_i)$ holds, where (α_i, χ_i) is the optimal solution of $P^{OA}(T)$. This leads to an iterative approach for finding the optimal solution to P where the linear relaxation $P^{OA}(T)$ is solved successively, at each iteration updating the set of linearizations points T. Note that the optimization procedure described in (Bonami *et al.* , 2008) may require the solution of a nonlinear probramming problem (NLP) to obtain the upper bound on the original objective function. The NLP is obtained by fixing all integer variables in the MINLP version of the problem to the current value objtained by solving the MIP relaxation. However, in this study only a function evalution of the original objective function is needed because the MINLP consists soley of integer variables χ. Termination of the solution algorithm occurs when an optimal solution (α^*, χ^*) of $P^{OA}(T)$ is obtained for which $G(\chi^*) = \alpha^*$ or when the difference between $G(\chi^*)$ and α^* is within some tolerance ϵ. This iteration process is illustrated in Figure 10.

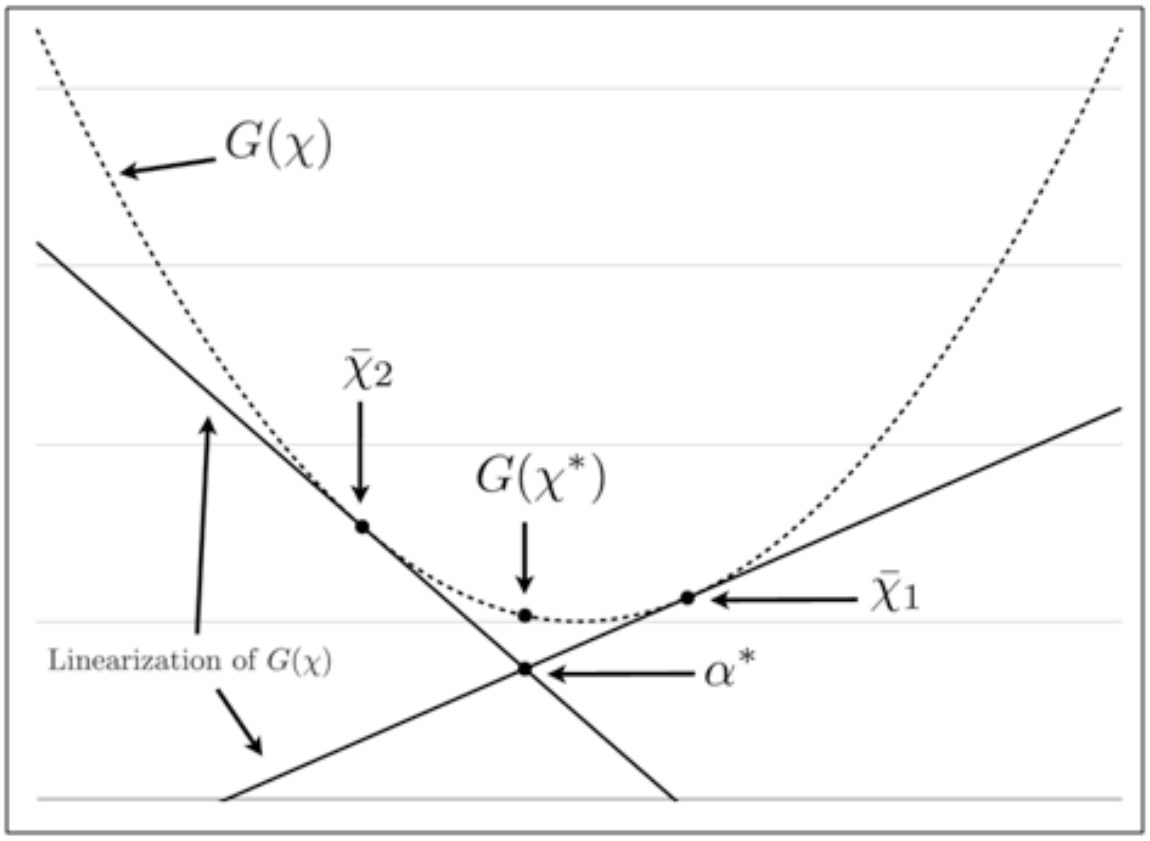


Fig. 9. Linearizations of a convex objective function which provide a lower bound on the optimal solution

C. Case Studies and Discussion

The following section details two case studies that are used to illustrate the technique presented above. First is a 32 stage binary distillation column, followed by a 120 state packed bed reactor.

1. Distillation Column

The first case study consists of a nonlinear 30 tray distillation column model, separating cyclohexane and n-heptane. The feed, with composition $x = 0.5$, is intro-

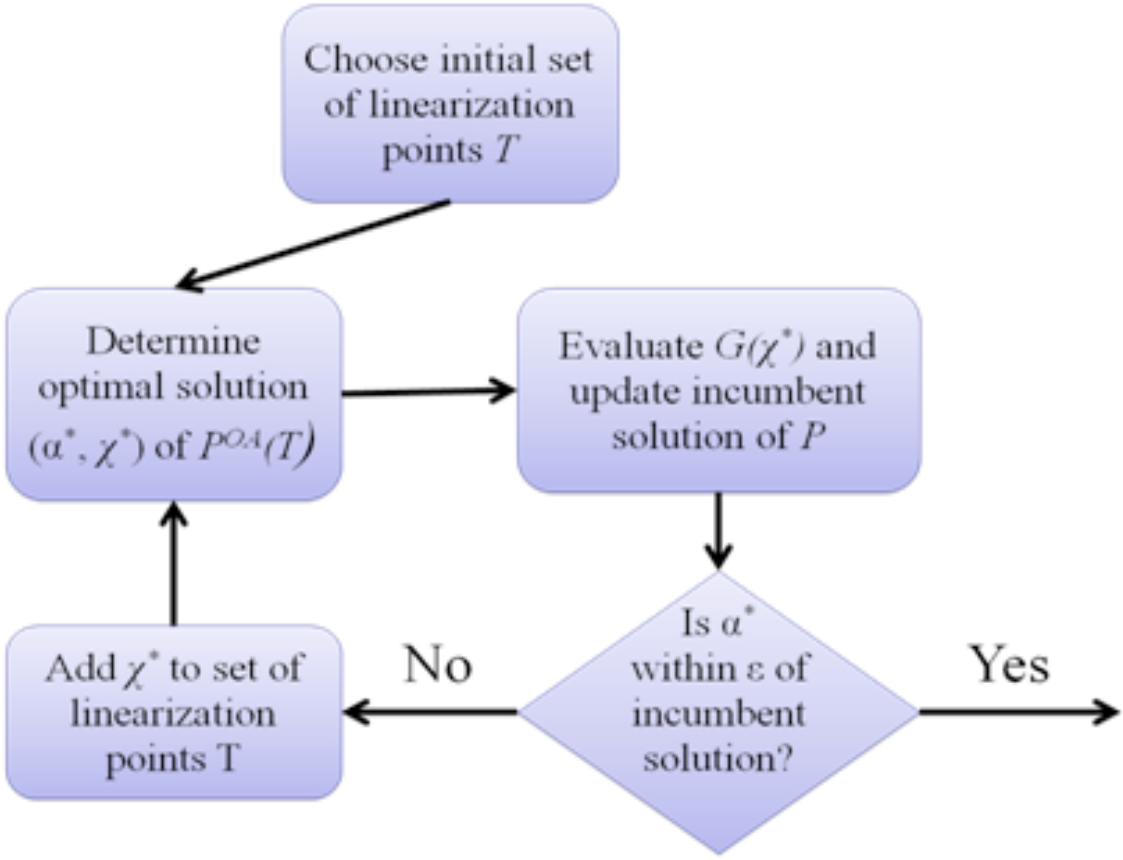


Fig. 10. Work flow diagram using linear under estimators to solve the MAX-DET optimization problem

duced on the 17th tray and the relative volatility, α_{AB}, is constant and equal to 1.6. The distillate and bottom product compositions are 0.935 and 0.065, respectively, and the reflux ratio is chosen to be 3. This model is also used in previous work on sensor network design, and its details are available there (Singh & Hahn, 2005a).

First, the single sensor case is analyzed. Each state is chosen individually as the only measurement. In order to produce the observability gramians, the following parameters are chosen; $T = [I, -I], M = 0.05$, and E^{32} is comprised of 32 orthogonal unit vectors. This produces 32 empirical observability gramians according to equa-

tions (2.39)-(2.40). Simply taking the determinant of these 32 matrices should allow for the ranking of the single sensor networks. However, in a system this large, having only one measurement makes the observability gramian very close to singular at best. In this case, values produced range from -1e-314 to 3e-97, effectively zero. Simple state space reduction will extract the important information from the observability gramians. In order to decide how many states to extract, the singular values of the observability gramian from the fully observed system, q = 32, must be analyzed. As seen in Figure 11, over 90% of the information is present in the first four eigenvalues.

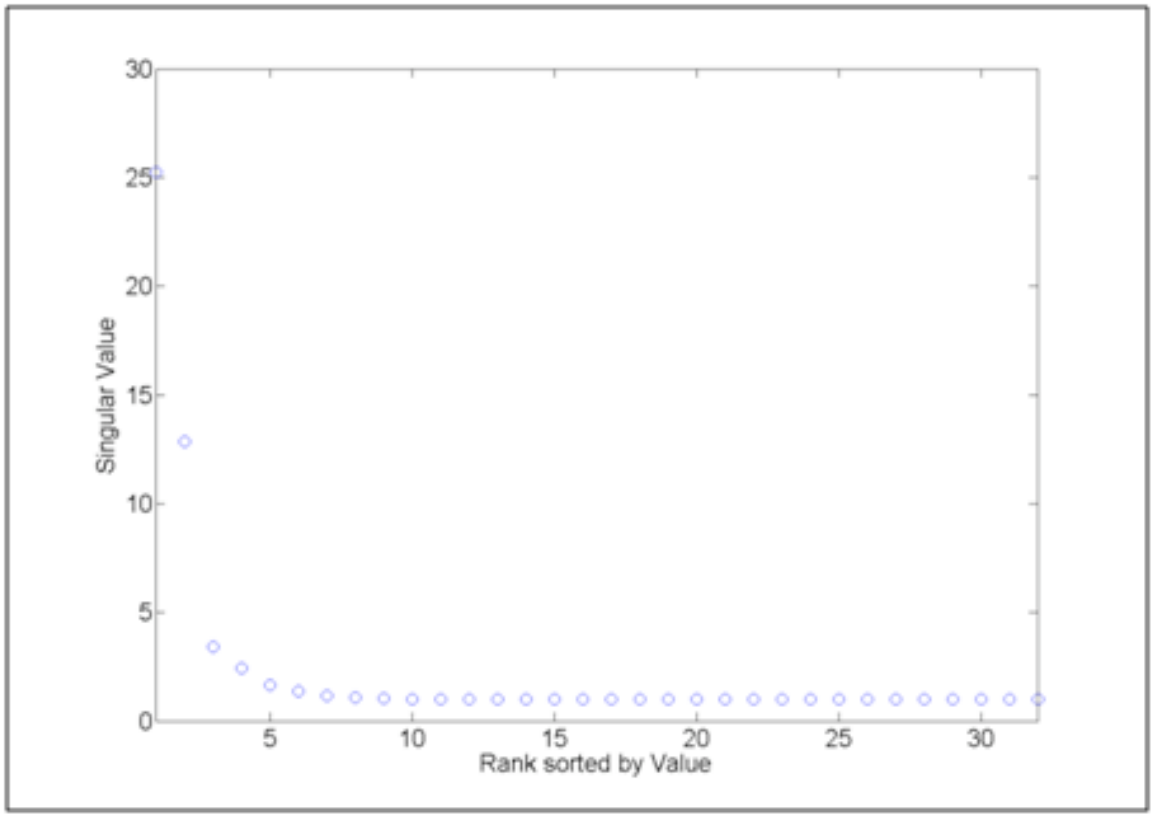


Fig. 11. Singular values of empirical observability gramian of fully observed system

The results from performing the reduction technique on each matrix according

to equation (4.3), allow the determination of the best sensor network. The next step is to compare the determinants of the empirical observability gramians based on each individual sensor being used alone. These results are plotted in Figure 12, and tray 26 is seen to be the best choice for placing one sensor.

Next, the problem of how best to place two sensors together is evaluated. Although trays 25 and 27 give the next highest values of the observability metric from the one sensor analysis, this choice of location for placing the second sensor will involve mostly redundant information.

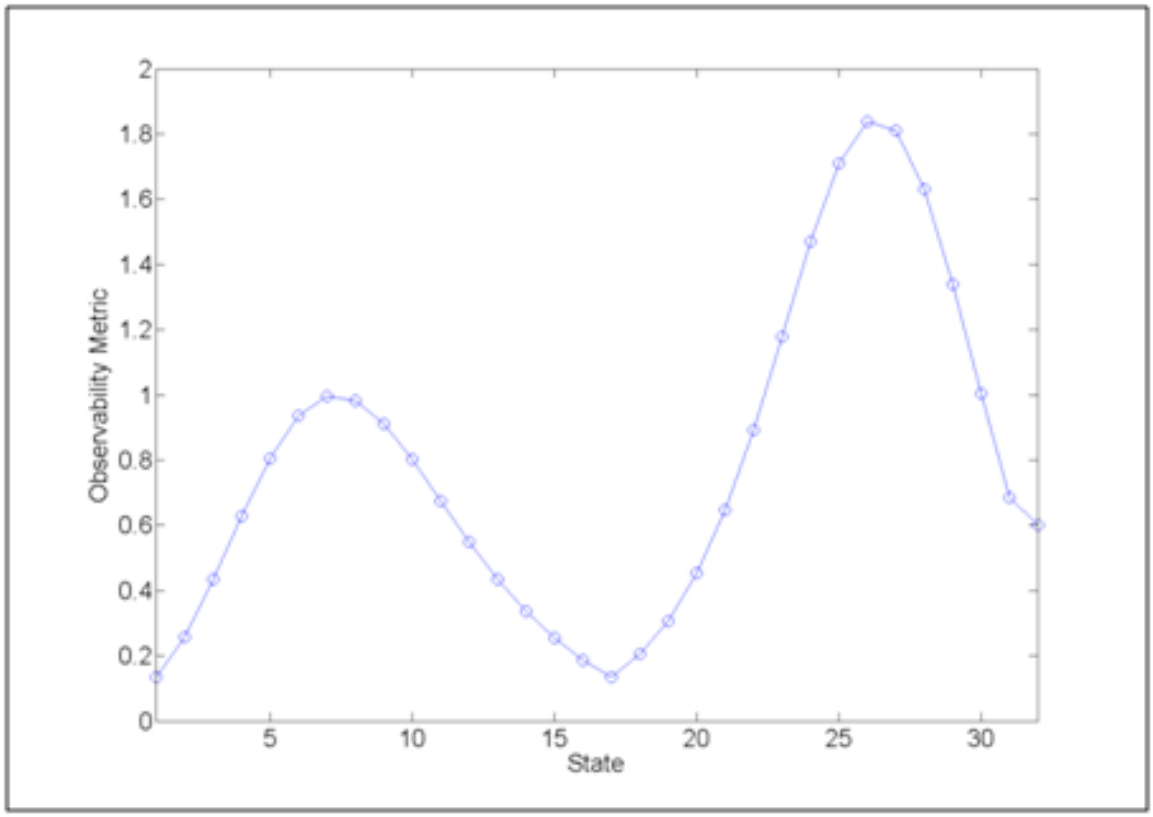


Fig. 12. Observability ranking for one sensor placement

In this case, there are only 469 feasible solutions for the two sensor problem,

indicating that total enumeration is tractable. The optimum placement of sensors is chosen to be tray 7 and tray 27. The results for the top 10 highest ranking determinants are displayed in Table II. The determinant is automatically taking into account redundancy of information, as the sensors are spread across the column, with one in each of the rectifying and stripping sections.

Table II. Results for placing two sensors

1st Sensor Tray	2nd Sensor Tray	Determinant
7	27	4.4284
6	27	4.3347
7	26	4.3249
6	26	4.2364
8	27	4.1774
7	28	4.1334
8	26	4.0754
6	28	4.0429
8	28	3.9017
5	27	3.8788
7	25	3.8753

In a similar manner, the results for each number of sensors placed can be determined. In each case, the size of the reduced observability gramian is chosen equal to the number of sensors desired. Following this methodology, Table III is produced. The results indicated that the redundancy of information is continually taken into account by the determinant of the reduced observability gramian as the sensor are spread about the distillation column as the number of sensors increases. As expected, both total enumeration and the optimization scheme produce exactly the same results shown in Table III.

Table III. Results for placing one through seven sensors

Number of Sensors	Sensor Location
1	26
2	7, 27
3	5, 26, 32
4	4, 11, 25, 32
5	1, 6, 7, 25, 32
6	1, 7, 13, 19, 26, 32
7	1, 5, 10, 15, 21, 27, 32

As the number of sensors placed increased, the number of possibilities increases dramatically, such that it begins to become unrealistic to be computed through full enumeration. In this case study, if it is desired to place 12 sensors, for example, among the 32 possible locations, there are over 2e8 different combinations. In order to solve this increasingly computationally intensive problem, the presented solution technique is utilized to find solutions in a reasonable amount of time. Table IV shows the optimal sensors locations found for placing 8 through 12 sensors.

Table IV. Results for placing 8 through 12 sensors

Number of Sensors	Sensor Location
8	1, 5, 9, 13, 18, 23, 28, 32
9	1, 4, 8, 11, 15, 19, 23, 28, 32
10	1, 4, 7, 10, 14, 17, 21, 25, 29, 32
11	1, 4, 7, 10, 13, 16, 18, 22, 25, 29, 32
12	1, 4, 7, 10, 13, 16, 18, 23, 26, 27, 29, 32

Convergence of the solution took under 6 minutes for each of the placements in Table IV using a Apple Macbook Pro with a 2.4GHz Intel Core i7 Quad Core Processor and 8GB of RAM. The algorithm was formulated using the open source software package

Pyomo (Hart *et al.* , 2012). Figure 13 shows the convergence behavior of the solution for the placement of 12 sensors.

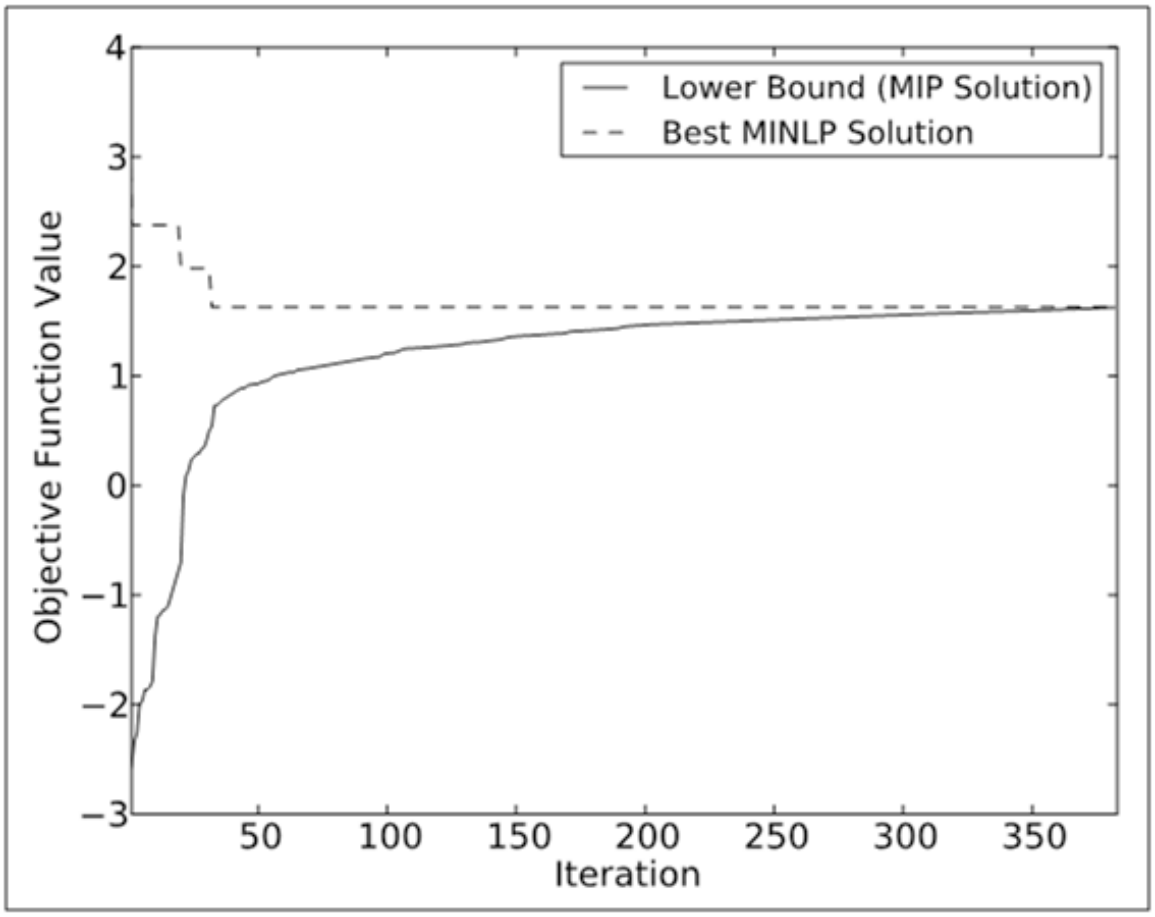


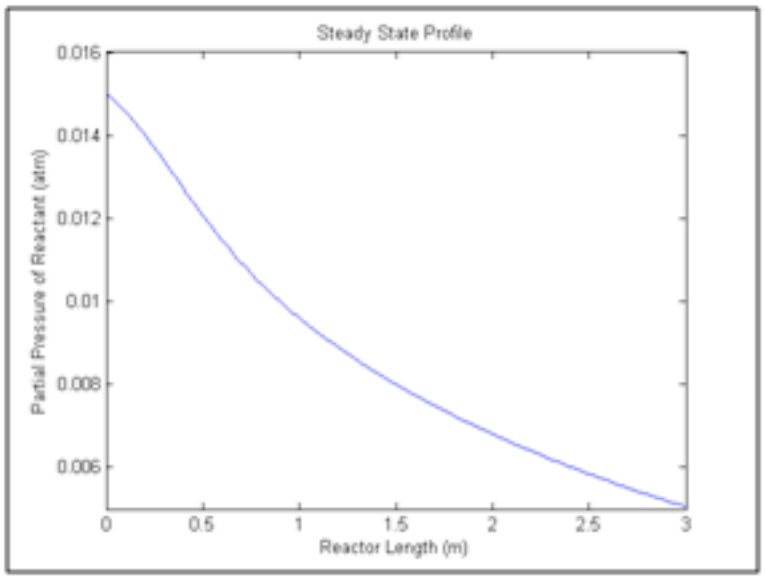
Fig. 13. Convergence of the optimization algorithm for placing 12 sensors

In this figure, the solid line represents the objective function value for the relaxed problem, and the dashed line represents the current best MINLP solution (incumbent). The figure shows the algorithm proved optimality of the incumbent solution in fewer than 400 iterations, with a high quality solution being found in fewer than 50 iterations. Thus, the efficiency of solution of the presented technique can be seen from this case study.

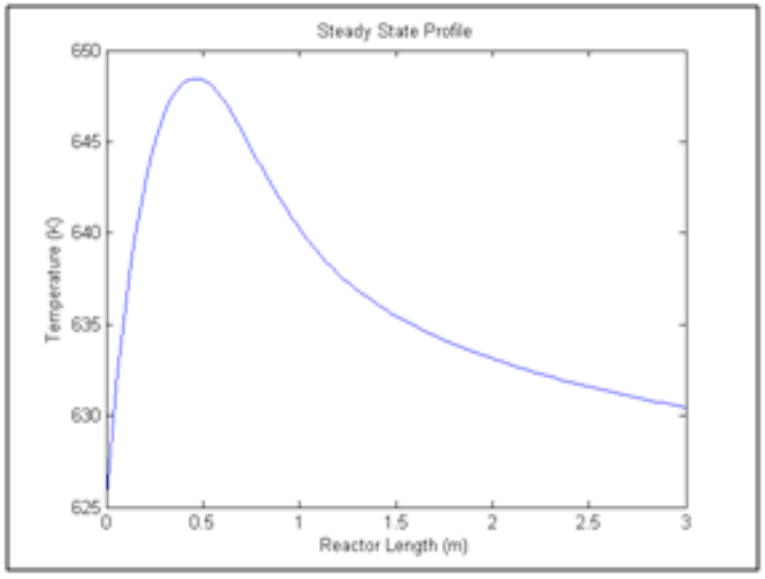
2. Solid Catalyst Packed Bed Reactor

The next case study is to determine the optimal sensor network design on a packed bed reactor described in several previous works (Singh & Hahn, 2005a; Van Welsenaere & Froment, 1970). The reaction involves producing phthalic anhydride from the catalytic oxidation of o-xylene. The model is implemented using 60 discretizations, with concentration and temperature being state variables at each point along the reactor, resulting in a total of 120 nonlinear equations. The initial conditions for the partial pressure of the reactant and the bulk temperature are p(t,0) = 0.015 atm and $T(t,0) = 625K$. The steady state profiles of the reactor's reactant concentration and temperature are shown in Figure 14. This type of reactor model is often studied due to "hot spots" in the temperature profile that are nontrivial to control. This case study is no exception; the spike in temperature around 0.45 m can be seen in the steady state profile.

As a more complex model with significantly more states, it is infeasible to compute results from total enumeration in a satisfactory amount of time. For instance, choosing to place 10 sensors along this reactor offers more than 1e14 possible sensor combinations. In addition to these differences between this case and the previous one, this model offers two different types of states to measure at each location, namely temperature and concentration. However, in this case it is almost always desirable to place temperature indicators over concentration sensors simply due to capital and maintenance costs, as well as reliability issues. First, the simple case of placing one sensor is analyzed. In addition to discovering where the most critical point along the reactor is for measurement, an analysis can be made about the tradeoffs between the two types of measurements. The results for placing one sensor are as follows and are displayed in Figure 15. For temperature, the maximum observability metric



(a) Partial pressure of reactant



(b) Reactor temperature

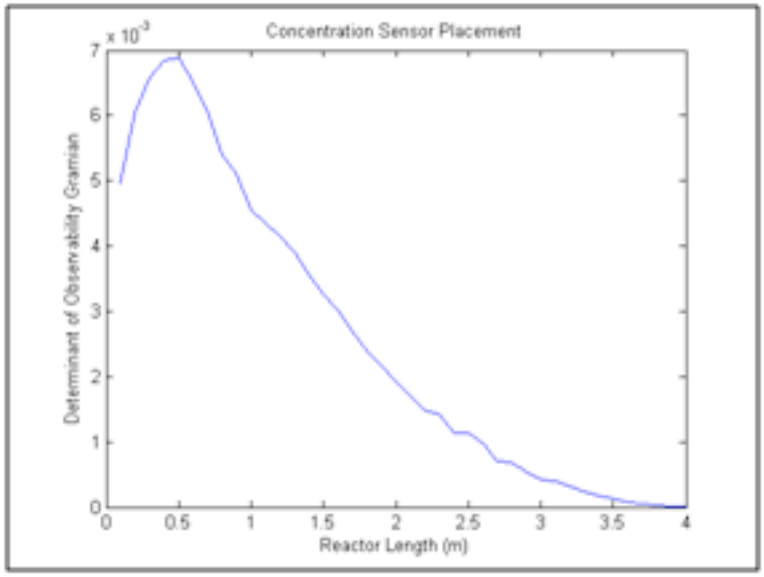
Fig. 14. Steady state profile of temperature and reactant partial pressure along the length of the reactor

was produced at 0.9 m along the reactor with a value of 3. For partial pressure, a maximum value of 0.007 was found at 0.9 m. These results seem to indicate that the most sensitive point for taking measurements is near the hotspot of the reactor.

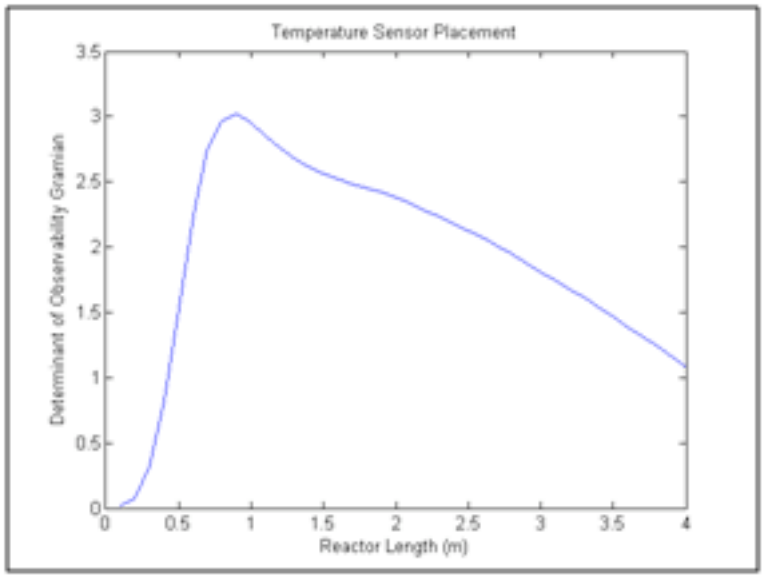
Next, the multisensor case is analyzed. The results for optimal sensor locations chosen for placing 3, 4, and 10 sensors are displayed in Table V. The second column lists the optimal sensor locations shown as number of discretizations points along the reactor. For example, state 30 out of a total of 60 of the 3 meter distance represents 1.5 m. The third column displays the time to solution in seconds. The solutions were solved to a relative solution gap of 0.001%, where the gap is defined as the percent difference of the objective function value of the MINLP and the relaxed NLP. The results indicate that even for this larger problem, solution times are feasible. When choosing 10 sensor locations from 120 possible, a solution was found in about three minutes. These times were validated through repetitive runs of the solver and each time produced similar timing results. Also, the results again confirm that redundancy of information is accounted for as some sensor locations are concentrated near the hot spot of the reactor, but as more are added they are spread throughout the column in order to best measure the states.

Table V. Multiple sensor placement results

Number of Sensors	Sensor Location	Optimization Solution Time (sec)
3	39, 40, 41	0.7
4	18, 45, 46, 47	1.0
10	12, 13, 23, 30, 36, 43, 50, 57, 58, 59	57



(a) Partial pressure of reactant



(b) Reactor temperature

Fig. 15. Results for placing only one sensor

D. Summary and Conclusion

Determination of the best locations for sensors is crucial for process monitoring and control. This paper presents a general procedure for obtaining the optimal sensor network design using observability analysis for dynamic nonlinear systems. The empirical observability gramian is first determined from computations on simulated or real plant data. If the gramian is near singular for low numbers of sensors, it is reduced in size based on a singular value decomposition analysis. Finally the network is optimized based on finding the maximum determinant of the feasible gramians. The presented technique incorporates the full observability gramian, taking advantage of all of the information available. Also, a decomposition for solving the resulting MINLP is applied. This method is a result of combining techniques in previous work in sensor network design, research into optimal experimental design, and optimization literature. The procedure is illustrated in two case studies, and results show that it is possible to place a reasonable number of sensors even for nonlinear systems of reasonable size.

CHAPTER V

INVESTIGATION OF DIFFERENT EXTENDED KALMAN FILTER IMPLEMENTATIONS

A. Introduction

Extended Kalman filters have found wide-spread use in nonlinear state and parameter estimation. In order to apply Kalman filters to nonlinear systems, EKF uses a first-order Taylor series expansion to linearize the nonlinear model along its trajectory and assumes a Gaussian noise distribution. This inevitably leads to EKF's limitations when applied to nonlinear systems or non-Gaussian noise processes. Numerous estimation methodologies have been proposed in the literature to address the problems that the EKF encounters and comparisons between the presented new approaches and EKF have often been made. To name a few, Rawlings and coworkers compared Moving Horizon Estimation (MHE) to EKF for estimation of constrained problem and pointed out that EKF may fail to converge to the true values (Haseltine & Rawlings, 2005). Chen *et al*. investigated particle filtering (PF) performance in industrial batch processes and compared the results to EKF (Chen *et al.* , 2005). Similar comparisons on a continuous stirred-tank reactor (CSTR) were made by Chen and his coworkers (Chen *et al.* , 2004). In the area of unscented Kalman filter (UKF) applications, Romanenko *et al*. applied EKF and UKF to a nonlinear exothermic chemical CSTR and a pH system (Romanenko *et al.* , 2004; Romanenko & Castro, 2004). The authors showed improvements in the performance of UKF over EKF in both cases. Kandepu *et al*. conducted comparisons between UKF and EKF for four cases including a Van der Pol oscillator and a reversible reaction (Kandepu *et al.* , 2008). They also showed that the UKF performs better than the EKF in terms

of robustness and speed of convergence. Contrary to some of these findings, the work by Qu and Hahn found the difference between UKF and EKF to be minor for mildly nonlinear systems and the advantages of UKF over EKF can be mainly seen for highly-nonlinear systemsv(Qu & Hahn, 2009a). It is evident from several of these findings that only a few generally applicable conclusions can be drawn when comparing different estimators. This is especially so as the implementation of an estimator can affect its performance.

This work performs a detailed study of EKF implementations with a focus on several key procedures such as discretization, first order linearization and computation of the Jacobian matrix for nonlinear continuous-time model functions. Comparisons among the implementations are made based upon a chemical reactor model with Van de Vusse reaction kinetics.

This Chapter is organized as follows. Different approaches for EKF implementations are presented and discussed in Section B. Section C compares the performance of each EKF implementation based upon application to a continuous stirred tank reactor exhibiting nonlinear dynamic behavior. Concluding remarks are given in Section D.

B. Implementations of EKF and Discussions

1. Continuous Extended Kalman Filter

The EKF was introduced with several other state estimation techniques in Chapter II. However, only the discrete form was discussed.For different models, there are different forms for EKF. For continuous-time models with discrete-time measure-

ments, as given by

$$\dot{x}(t) = f(x(t), u(t)) + Gw(t) \tag{5.1}$$

$$y_k = h(x(t_k)) + v_k \tag{5.2}$$

$$x(0) \sim N(\bar{x}_0, P_{x_0}),\ w(t) \sim N(0, Q),\ v_k \sim N(0, R_k), \tag{5.3}$$

where $x \in \mathbb{R}^n$ is a vector of the state variables; $w \in \mathbb{R}^n$ is a vector of plant noise; $y_k \in \mathbb{R}^m$ is a vector of the measured variables and $v_k \in \mathbb{R}^m$ is a vector of measurement noise, the following equations define the continuous-discrete form of the EKF:

Prediction equations:

$$\begin{aligned} \dot{\hat{x}} &= f(\hat{x}, u) \\ \dot{P} &= A(\hat{x})P + PA(\hat{x}) + GQG^{'} \\ \hat{y}_k &= h(\hat{x}(t_k), u) \end{aligned} \tag{5.4}$$

Update equations:

$$\begin{aligned} K_k &= P(t_k)H_k^{'}(H_k P(t_k) H_k^{'} + R)^{-1} \\ P_k &= (I - K_k H_k)P(t_k) \\ \hat{x}_k &= \hat{x}(t_k) + K_k(y_k - \hat{y}_k) \end{aligned} \tag{5.5}$$

where $A(\hat{x}) \approx \frac{\partial f}{\partial x}|_{\hat{x}}$ and $H_k \approx \frac{\partial h}{\partial x}|_{\hat{x}(t_k)}$ are the matrices of the linearized system model, and computed as functions of the estimate for linearization about the estimated trajectory. Lyapunov equations need to be solved at each step for computing the Kalman gain K_k and updating state estimates, $\hat{x}_k$.

Finite difference is the most commonly used method found in the literature for model discretization or computing a Jacobian matrix $A(\hat{x})$ for EKF (Haseltine & Rawlings, 2005; Romanenko & Castro, 2004). The estimation errors and computation times are greatly dependent on the step size for computing the finite difference.

In order to implement an extended Kalman filter, attention must be paid to discretization and linearization of nonlinear continuous-time model along its trajectory. The order in which discretization and linearization are performed may result in differences in performance. The approach used for discretizing systems, such as Euler's method or Runge-Kutta method, may also lead to different results. In addition, computation of Jacobian matrices via sensitivity equations or via finite differences affect the accuracy of EKF.

Taking these points into account, several different EKF implementations are discussed in this section. The reason for doing so is that when results for EKF and other estimation methods are reported in the literature, there is often very little discussion of the discretization scheme used, yet the choice of a discretization scheme has a major effect on the outcome. In this work, the system model is assumed to be a nonlinear continuous dynamic system with discrete measurements, such as the one shown in Equations (5.1) - (5.3).

The functions f and h are differentiable functions of the state vector x, $w \in \mathbb{R}^n$ is a vector of plant noise, with $E[w] = 0$ and $E[ww^T] = Q$; $y_k \in \mathbb{R}^m$ is a vector of the measured variables and $v_k \in \mathbb{R}^m$ is a vector of measurement noise, with $E[v_k] = 0$ and $E[v_k v_k^T] = R_k$; n is the number of states, m refers to the number of measurement variables. The distributions of w and v are Gaussian. The initial value x_0 is also a Gaussian random variable with known mean $\bar{x}_0$ and known $n \times n$ covariance matrix P_{x_0}. The sampling time for measurements is T. $x(t)$,$u(t)$ and $w(t)$ are referred to x, u and w, respectively, in the rest of the paper unless specified.

2. Implementations Via Linearization and Continuous KF for Covariance Prediction

This algorithm (**Algorithm 1**) linearizes the model along its trajectory and then predicts the covariance matrix P via continuous KF.

The state estimate $\hat{x}$ is computed from the nonlinear differential equation, i.e.,

$$\dot{\hat{x}} = f(\hat{x}, u)^1 \tag{5.6}$$

The Jacobian matrix $A(\hat{x})$ of $f(\hat{x})$ is found to be

$$A(\hat{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial \hat{x}_1} & \cdots & \frac{\partial f_1}{\partial \hat{x}_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial \hat{x}_1} & \cdots & \frac{\partial f_m}{\partial \hat{x}_n} \end{bmatrix}. \tag{5.7}$$

The covariance matrix P is then propagated through the Lyapunov equation

$$\dot{P} = A(\hat{x})P + PA(\hat{x})^{'} + GQG^{'}. \tag{5.8}$$

Since the initial values $\hat{x}_0$ and P_0 are known, ODEs (5.6) $\sim$ (5.8) form an initial value problem that can be solved using commercial ODE solvers such as Matlab® ′s *ode*45.

The predictions at any sampling point kT are given by

$$\hat{x}(kT) = \hat{x}_k^-, \ P(kT) = P_k^-. \tag{5.9}$$

---

[1]Consider the first-order Taylor series expansion of $f(x, u)$ about the current estimate (i.e., conditional mean) $\hat{x}$:

$$f(x, u) \cong f(\hat{x}, u) + \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}} [x - \hat{x}],$$

where $\hat{x}$ is close to x. Taking the expectation of both sides of the above equation gives

$$E\{f(x, u)\} = f(\hat{x}, u),$$

Therefore the state estimate $\hat{x}$ is predicted via $\dot{\hat{x}} = f(\hat{x}, u)$.

Since measurements are only available at the sampling time, the Kalman gain is calculated based on the predicted discrete covariance:

$$K_k = P_k^- H_k^{'}(H_k P_k^- H_k^{'} + R)^{-1}, \text{where } H_k = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_k^-}. \tag{5.10}$$

In a last step, corrections are made based upon the predictions and the new available measurement.

$$P_k = (I - K_k H_k) P_k^- \tag{5.11}$$

$$\hat{x}_k = \hat{x}_k^- + K_k[y_k - h(\hat{x}_k^-)]. \tag{5.12}$$

Table VI provides a summary of this algorithm. In this algorithm, both mean $\hat{x}$ and covariance matrix P are solved in a continuous manner. The numerical solver determines the step size for integration of $\hat{x}$ and P during each sampling interval. This increases the accuracy of integration compared to methods with a fixed step size. This method produces an error resulting from linearization at each integration step only.

Table VI. Summary of procedure for algorithm 1

Initialization	$\hat{x}_0 = \bar{x}_0,\ P_0 = P_{x_0}$
Prediction	$\dot{\hat{x}} = f(\hat{x}, u)$ $\dot{P} = A(\hat{x})P + PA(\hat{x})^{'} + GQG^{'}$, where $A(\hat{x}) = \left.\frac{\partial f}{\partial x}\right\|_{x=\hat{x}}$ $\hat{x}(kT) = \hat{x}_k^-,\ P(kT) = P_k^-$
Kalman gain	$K_k = P_k^- H_k^{'}(H_k P_k^- H_k^{'} + R)^{-1}$, where $H_k = \left.\frac{\partial h}{\partial x}\right\|_{x=\hat{x}_k^-}$
Correction	$P_k = (I - K_k H_k) P_k^-$ $\hat{x}_k = \hat{x}_k^- + K_k[y_k - h(\hat{x}_k^-)]$

Remark: There are possible alternatives to this implementation. In Algorithm 1, the Jacobian matrix A is considered to be time-varying for solving the ODEs. If A is assumed to be time-invariant, the matrix could be calculated at each sampling interval, i.e., $A(\hat{x}_k) = \frac{\partial f}{\partial x}\Big|_{x=\hat{x}_k}$. The ODEs (5.6) and (5.8) thus can be solved separately (**Algorithm 1.1**), which may reduce computation costs. However,the computation accuracy may be decreased concurrently since the error is affected by linearization at each sampling interval, which is usually significantly larger than the integration step size used in Algorithm 1.

3. Implementations Via Linearization and Discrete KF for Covariance Prediction

In the second algorithm (**Algorithm 2**), linearization of the nonlinear continuous-time model along its trajectory is performed and the covariance predictions are computed from discrete information.

As in Algorithm 1, the nonlinear differential equation

$$\dot{\hat{x}} = f(\hat{x}, u) \tag{5.13}$$

is used for predicting the state vector $\hat{x}_k$. The Jacobian matrix A is then computed at each sampling time,

$$A(\hat{x}_k) = \frac{\partial f}{\partial x}\Big|_{x=\hat{x}_k}. \tag{5.14}$$

The linearized model is given by

$$\dot{\tilde{x}} = A(\hat{x}_k)\tilde{x} + B\tilde{u} + f(\hat{x}_k, u_k), \tag{5.15}$$

where $\tilde{x} = \hat{x} - \hat{x}_k$ and $\tilde{u} = u - u_k$.

Solving the ODEs (5.15) and (5.8) for computing the discrete covariance matrix

P results in

$$\hat{x}_{k+1} = A_k \hat{x}_k + B_k \tag{5.16}$$

$$P_{k+1}^{-} = A_k P A_k^{'} + G Q_k G^{'} \tag{5.17}$$

where $A_k = e^{A(\hat{x}_k)T}$ is the state transition matrix, $GQ_kG^{'} = \int_0^T e^{A(\hat{x}_k)\tau} GQG^{'} e^{A(\hat{x}_k)^{'}\tau} d\tau$ is the process noise matrix, and $B_k = \int_0^T e^{A(\hat{x}_k)\tau}[Bu(T-\tau) + f(\hat{x}_k, u_k)]d\tau$ is the input matrix with $B = \frac{\partial f}{\partial u}\Big|_{u=u_k}$.

Corrections for mean and covariance matrices using a Kalman filter are computed by equations (5.10) $\sim$ (5.12) as discrete measurements are the source for estimation updates.

Table VII summarizes the procedure of this algorithm. During each measurement sampling interval, the nonlinear system is considered as a linear first-order system with constant coefficients, which results in equations (5.16) and (5.17). This incurs a larger error for computing the covariance matrix P than Algorithm 1, where P is integrated using a time-varying state transition matrix A.

An alternative (**Algorithm 2.1**) to this algorithm is to use an Euler approximation for discretization of continuous-time models. The matrices for covariance prediction are then replaced by the following:

$$A_k = I + A(\hat{x}_k)T,\ Q_k = QT,\ \text{and}\ B_k = BTu(kT). \tag{5.18}$$

Due to the lower accuracy of Euler's method compared to, e.g., a Runge-Kutta method used by conventional ODE solvers, Algorithm 2.1 will in theory result in poorer performance than Algorithm 2. The errors are due to both linearization and discretization of nonlinear continuous-time models. Additionally, when large sampling times are used, this method may produce unstable results.

Table VII. Summary of procedure for algorithm 2

Initialization	$\hat{x}_0 = \bar{x}_0,\ P_0 = P_{x_0}$
Prediction	$\dot{\hat{x}} = f(\hat{x}, u)$ $P_{k+1}^{-} = A_k P A_k^{'} + G Q_k G^{'}$ where $A(\hat{x}_k) = \frac{\partial f}{\partial x}\Big\vert_{x=\hat{x}_k},\ A_k = e^{A(\hat{x}_k)T},$ $G Q_k G^{'} = \int_0^T e^{A(\hat{x}_k)\tau} G Q G^{'} e^{A(\hat{x}_k)^{'}\tau} d\tau$
Kalman gain	$K_k = P_k^{-} H_k^{'} (H_k P_k^{-} H_k^{'} + R)^{-1}$, where $H_k = \frac{\partial h}{\partial x}\Big\vert_{x=\hat{x}_k^{-}}$
Correction	$P_k = (I - K_k H_k) P_k^{-}$ $\hat{x}_k = \hat{x}_k^{-} + K_k[y_k - h(\hat{x}_k^{-})]$

4. Implementations Via Discretization Followed by Linearization

In this algorithm (**Algorithm 3**), the nonlinear continuous-time model is discretized first and then linearized along its trajectory. It is well known that ODE solvers or Euler approximations are two common methods used for discretization of continuous-time models. Euler's method discretizes models with a fixed step size while the step size for discretization is adjusted for different dynamic behaviors when ODE solvers are used. Therefore ODE solvers can result in more accurate discrete data than if Euler approximations are used.

In order to use ODE solvers such as Matlab® ′s *ode*45 for discretization, however, a continuous noise signal $w(t)$ in equation (5.3) is needed. This is unlikely to be simulated and implemented in a digital computer. As a compromise, a discrete signal $w_k = w(kT)$ can be generated. After solving an initial value problem for finding a

solution of the ODE

$$\dot{x} = f(x, u), \tag{5.19}$$

w_k is linearly added to $x_k = x(kT)$ at each sampling interval kT.

Once a discrete-time model is obtained, the next step is to compute the Jacobian matrix A_k of the nonlinear function f at each time step kT. Since numerical discretization of the continuous-time model is executed at the first step, no analytical form for the model is available and A_k also needs to be computed numerically. One approach uses the sensitivity matrix

$$\dot{A} = \frac{\partial f}{\partial x'} A. \tag{5.20}$$

The state vector predictions $\hat{x}_k^-$ and the Jacobian matrix A_k of f can be solved simultaneously

$$\dot{\hat{x}} = f(\hat{x}, u) \tag{5.21}$$

$$\dot{A} = \frac{\partial f}{\partial \hat{x}'} A \tag{5.22}$$

with $\hat{x}_0 = \bar{x}_0$ and $A_0 = I$.

Remarks: One alternative (**Algorithm 3.01**) is to compute A_k by using finite difference such as central differences, i.e.,

$$A_k = \frac{f(\hat{x}_k + \Delta x) - f(\hat{x}_k - \Delta x)}{2\Delta x}. \tag{5.23}$$

Numerically it is non-trivial to find an appropriate difference Δx. A finite difference method may be less accurate than solving the sensitivity equation.

Once the state vector estimate $\hat{x}_k^-$ and the matrix A_k are computed using equa-

tions (5.21) and (5.22), the covariance matrix P is computed

$$P_{k+1}^{-} = A_k P_k A_k^{'} + G Q_k G^{'}, \tag{5.24}$$

where Q_k is approximated by QT.

The Kalman gain is calculated in the same discrete-time form as in equation (5.10),

$$K_k = P_k^{-} H_k^{'} (H_k P_k^{-} H_k^{'} + R)^{-1}, \text{where } H_k = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_k^{-}}. \tag{5.25}$$

Updates for the state estimate $\hat{x}_k$ and the covariance matrix P_k are made in the same manner as in equations (5.11) and (5.12),

$$P_k = (I - K_k H_k) P_k^{-} \tag{5.26}$$

$$\hat{x}_k = \hat{x}_k^{-} + K_k[y_k - h(\hat{x}_k^{-})]. \tag{5.27}$$

Table VIII. Summary of procedure for algorithm 3

Initialization	$\hat{x}_0 = \bar{x}_0,\ P_0 = P_{x_0},\ A_0 = I$
Prediction	$\dot{\hat{x}} = f(\hat{x}, u), \dot{A} = \frac{\partial f}{\partial \hat{x}'} A$ $\hat{x}(kT) = \hat{x}_k^{-},\ A(kT) = A_k$ $P_{k+1}^{-} = A_k P_k A_k^{'} + G Q_k G^{'}$
Kalman gain	$K_k = P_k^{-} H_k^{'} (H_k P_k^{-} H_k^{'} + R)^{-1}$, where $H_k = \left.\frac{\partial h}{\partial x}\right\|_{x=\hat{x}_k^{-}}$
Correction	$P_k = (I - K_k H_k) P_k^{-}$ $\hat{x}_k = \hat{x}_k^{-} + K_k[y_k - h(\hat{x}_k^{-})]$

Table VIII summarizes the procedure, where the sensitivity matrix is used for computing the Jacobian matrix of the nonlinear function f at each time step.

As mentioned in Algorithm 2.1, a discretization of continuous-time models can

be replaced by an Euler approximation. Algorithm 3 then results in:

$$\hat{x}_{k+1}^{-} = \hat{x}_k + Tf(\hat{x}_k, u_k) \tag{5.28}$$

$$A_k = I + T\frac{\partial f(\hat{x}, u_k)}{\partial \hat{x}}\bigg|_{\hat{x}=\hat{x}_k} \tag{5.29}$$

$$P_{k+1}^{-} = A_k P_k A_k^{'} + GQTG^{'}, \tag{5.30}$$

which is referred to as **Algorithm 3.1** here. The Kalman gain and correction equations remain the same. Unstable filters may be generated when the step size is large for the Euler approximation. This method results in errors from linearization of the nonlinear model along its trajectory at each sampling interval and discretization of continuous-time models for computing the covariance matrix, similar to Algorithm 2.1. Additionally, an error is resulting from prediction of the states using an Euler approximation. It is estimated that this method performs worse than Algorithm 2.1.

5. Discussions

The algorithms discussed in the last three subsections involve linearization and discretization of nonlinear continuous-time models at different steps. The sequence and the specific technique for executing them define each algorithm. To be more specific, Algorithms 1 and 1.1 execute linearization of the model along its trajectory first and then use a continuous KF to predict the covariance P. Algorithms 2 and 2.1 also linearize the model along its trajectory first but use a discrete KF to propagate P. Algorithms 3, 3.01 and 3.1 first discretize the model and then linearize the model along its trajectory for computing P. With respect to linearization, Algorithm 1, 3 and 3.01 evaluate the Jacobian matrix A continuously while the others compute it only at the sampling time.

In spite of the classifications, all algorithms share the same formulation for state

predictions and mean and covariance corrections with the exception of Algorithm 3.1 where the state is predicted via an Euler approximation. The main differences between the methods are given by the computation of the covariance matrix P and the approaches used to evaluate the Jacobian matrix A. Table IX summarizes the results. Algorithm 1 and 1.1 propagate P via a continuous KF while a discrete KF is used to compute P for the other algorithms. Further, Algorithms 1, 3 and 3.01 evaluate A continuously while Algorithms 1.1, 2 and 2.1 compute A at each sampling point x_k.

Table IX. Summary of the algorithms

<table>
<tr><td>Initialization</td><td colspan="3">$\hat{x}_0 = \bar{x}_0,\ P_0 = P_{x_0},\ A_0 = I$</td></tr>
<tr><td>Mean
Prediction</td><td colspan="3">$\dot{\hat{x}} = f(\hat{x}, u)$
$(Exception : \hat{x}^-_{k+1} = \hat{x}_k + Tf(\hat{x}_k, u_k)\ for\ Algorithm\ 3.1)$</td></tr>
<tr><td rowspan="3">Covariance
Prediction</td><td>Evaluate A:</td><td>at $x(t)$</td><td>at x_k</td></tr>
<tr><td>$\dot{P} = A(\hat{x})P + PA(\hat{x})' + GQG'$</td><td>Alg. 1</td><td>Alg. 1.1</td></tr>
<tr><td>$P^-_{k+1} = A_k P_k A'_k + GQ_k G'$</td><td>Alg. 3, 3.01</td><td>Alg. 2, 2.1, 3.1</td></tr>
<tr><td>Kalman gain</td><td colspan="3">$K_k = P^-_k H'_k (H_k P^-_k H'_k + R)^{-1}$, where $H_k = \frac{\partial h}{\partial x}\Big|_{x=\hat{x}^-_k}$</td></tr>
<tr><td>Correction</td><td colspan="3">$P_k = (I - K_k H_k) P^-_k$
$\hat{x}_k = \hat{x}^-_k + K_k[y_k - h(\hat{x}^-_k)]$</td></tr>
</table>

Figure 16 provides a graphic overview of the discussed EKF implementations for a continuous-time model with discrete-time measurements.

Remarks:

1. Algorithms 1 and 1.1 predict P via solution of the Lyapunov Eq. $\dot{P} = A(\hat{x})P + PA(\hat{x})' + GQG'$ which distinguishes them from the other algorithms. The dif-

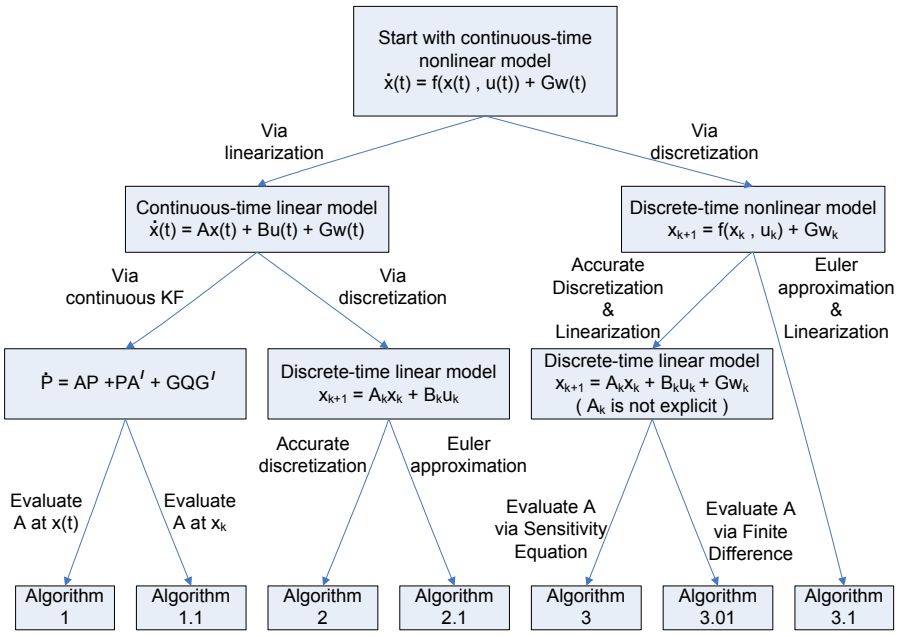


Fig. 16. Comparison of different algorithms for implementing EKF.

ference between 1 and 1.1 lies in the Jacobian matrix calculation of model functions. Algorithm 1 computes the Jacobian A at each integration step $x(t)$ while Algorithm 1.1 computes A only at time kT which implies that A_k is time-invariant during each integration period. Algorithm 1.1 should be less accurate than Algorithm 1 though it can save computation time.

2. Algorithms 2, 2.1, 3, 3.01 and 3.1 propagate P in a discrete manner $P_{k+1}^{-} = A_k P_k A_k' + G Q_k G'$.

3. Algorithms 2 and 2.1 linearize the model along its trajectory first and then carry out discretization of the state equations. Algorithm 2 uses $A_k = e^{A(\hat{x}_k)T}$ for discretization while Algorithm 2.1 makes use of an Euler approximation which results in $A_k = I + A(\hat{x}_k)T$. If the sampling time T is small, then Algorithm 2.1 may produce results comparable to Algorithm 2. Additional

note: For computing the integral $\int_0^T e^{A(\hat{x}_k)\tau} GQG' e^{A(\hat{x}_k)'\tau} d\tau$ for Algorithm 2, a simple alternative involving the matrix exponential computation is presented by Van Loan (Van Loan, 1978).

4. Algorithms 3, 3.01 and 3.1 implement discretization of the model first and then use linearization for the same model along its trajectory. Similar to Algorithms 2 and 2.1, the discretization scheme distinguish Algorithms 3 and 3.01 from Algorithm 3.1 where an Euler approximation is used. Therefore both Algorithm 3 and 3.01 should be superior to Algorithm 3.1 in terms of accuracy. Algorithm 3 and 3.01 differ in the approach for computing the Jacobian matrix A. The former uses solution of the sensitivity equation while finite differences are chosen for the latter.

5. Algorithms 2.1 and 3.1 are identical except for how they predict the state $\hat{x}$. The former uses direct integration while the latter makes use of an Euler approximation. Therefore Algorithm 2.1 potentially performs better than 3.1. However, both of them may produce unstable filters when the sampling interval is large due to the Euler approximation for discretizing the model.

6. Algorithm 2 is identical to Algorithm 1.1 although the implementations are not the same. Please refer to the Appendix for a detailed proof.

C. Case Studies

To evaluate the performance of EKF using each implementation, the algorithms given in Section B have been applied to models including ones with mild as well as some with strong degrees of nonlinearity and a large number of scenarios such as different operating conditions, different tuning parameters Q and R, and different

process and measurement noise levels. Monte Carlo simulations of 50 runs were carried out for each scenario. The performance is evaluated by the overall mean-squared error (MSE). The MSE is first averaged over all simulations for each time point and then over time to indicate the long-term behavior of each estimator and the distribution of errors over time. Due to the space constraints, this section includes only one case study for a model exhibiting strongly nonlinear behavior. However, the presented results are representative for other cases that have been investigated.

The considered system is an isothermal nonlinear CSTR with a competing side reaction governed by van de Vusse reaction kinetics used for the production of Cyclopentanol (Stack & Doyle III, 1997):

$$\begin{aligned} &\mathrm{A} \xrightarrow{\mathrm{k_1}} \mathrm{B} \xrightarrow{\mathrm{k2}} \mathrm{C} \\ &2\mathrm{A} \xrightarrow{\mathrm{k3}} \mathrm{D}. \end{aligned} \tag{5.31}$$

Component A is the the reactant cyclopentadiene, B is the product cyclopentanol, C and D are the side products cyclopentandiol and dicylopentadiene. The nonlinear system model is given by the following three differential equations:

$$\frac{dC_A}{dt} = \frac{u}{V}(C_{Ain} - C_A) - k_1 e^{-E_1/RT} C_A - k_3 e^{-E_3/RT} C_A^2 \tag{5.32}$$

$$\frac{dC_B}{dt} = -\frac{u}{V} C_B + k_1 e^{-E_1/RT} C_A - k_2 e^{-E_2/RT} C_B \tag{5.33}$$

$$\begin{aligned} \frac{dT}{dt} =& \frac{1}{\rho c_p}[k_1 e^{-E_1/RT} C_A(-\Delta H_1) + k_2 e^{-E_2/RT} C_B(-\Delta H_2) + k_3 e^{-E_3/RT} C_A^2(-\Delta H_3)] \\ &+ \frac{u}{V}(T_{in} - T) + \frac{Q}{V \rho c_p} \end{aligned} \tag{5.34}$$

where the feed flow rate u is the only controlled variable. The values of the parameters can be found in the work by Hahn and Edgar (Hahn & Edgar, 2001).

The nonlinear model exhibits multiple steady states, of which the upper steady state ($C_{Ass} = 2.49\mathrm{mol/L}$; $C_{Bss} = 1.1\mathrm{mol/L}$; $T_{ss} = 411K$; $u = 800L/h$) is chosen

as the point of operation. The measurable variable is assumed to be the reactor temperature T. Initial conditions are chosen to be $\hat{x}_0 = \begin{bmatrix} 2.5 & 1.09 & 411 \end{bmatrix}^T$ and all process variables were scaled to be dimensionless using the upper steady state as the nominal point. The sampling time for the measurements is 0.02 min.

The remaining filter parameters after scaling are given by

$$\hat{P}_0 = diag\{100, 100, 100\}, \qquad Q = diag\{10^{-2}, 10^{-2}, 10^{-2}\},$$
$$R = diag\{10^{-2}, 10^{-2}\}.$$

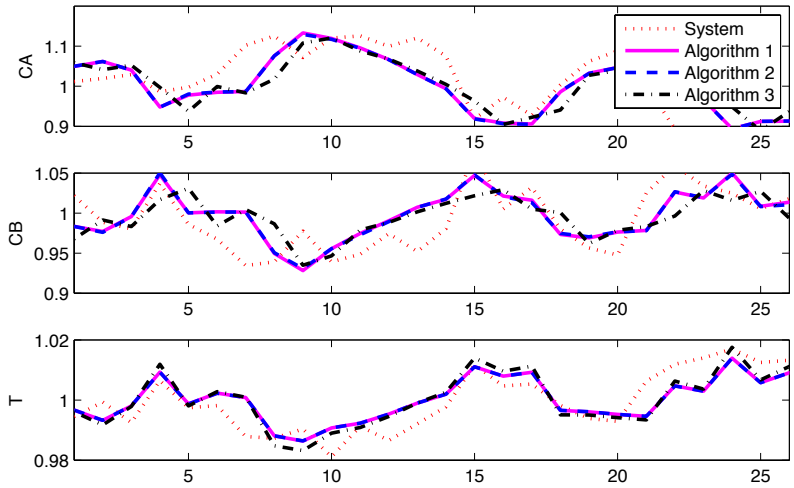


Fig. 17. EKF performance comparison.

The overall MSEs for $\Delta t = 0.02, R = 0.01I$ are shown in Table X and it can be seen that Algorithms 1, 2 and 3 produce relatively small MSEs. The predictions of the states made by these algorithms are also shown in Figure 17.

Compared to Algorithm 1, Algorithm 1.1 also performs reasonably well although its performance is slightly worse than Algorithm 1 for large variations in the operating conditions.

As discussed in Remark 3, Algorithm 2 produces more accurate results than Algorithm 2.1 due to the Euler approximations used in the latter. This can be seen by comparing the MSEs, which are 0.627 for Algorithm 2 and 1.039 for Algorithm 2.1. As shown in the Appendix, Algorithm 1.1 and 2 are identical and generate the same MSEs.

The MSE Results using Algorithms 3, 3.01 and 3.1 are 0.653, 0.675, and 1.537, respectively. Algorithm 3.01 has poorer performance than Algorithm 3 because of the finite differences used for computing the Jacobian instead of solving the sensitivity equation. The difference between Algorithm 3 and 3.1 is significant as Algorithm 3.1 failed to produce reasonably good estimates when the sampling interval is large due to the Euler approximation.

Methods that involve Euler approximations, such as Algorithm 2.1 and 3.1, produce relatively large values of the MSEs as given by 1.039 and 1.537, respectively. The two methods share the formulation for the covariance matrix propagation but differ in their mean predictions.

Table X. MSEs for algorithms ($\Delta t = 0.02, R = 0.01I$)

Algorithms	1	1.1	2	2.1	3	3.01	3.1
MSEs	0.626	0.627	0.627	1.039	0.653	0.675	1.537

Further investigations are also carried out for different measurement noise levels and different sampling interval lengths, and are summarized in Table XI. When measurement noise is relatively small ($\Delta t = 0.02, R = 0.0001I$), methods involving Euler approximations such as Algorithm 2.1 and 3.1 generate acceptable MSEs. However, for large measurement noise($\Delta t = 0.02, R = 1I$), both Algorithm 2.1 and 3.1 exhibit unstable properties due to the Euler Approximation. Similarly, all algorithms

perform comparably well when the sampling time is relatively small ($\Delta t = 0.002$). However unstable filters can result from Algorithm 2.1 or 3.1 when large sampling intervals are used. Therefore approaches such as Algorithm 2.1 or 3.1 should only be considered when measurement noise levels and the sampling time are relatively small.

Table XI. MSEs for algorithms

Algorithm	1	1.1	2	2.1	3	3.01	3.1
$\Delta t = 0.02, R = 0.0001I$	0.250	0.247	0.247	0.339	0.361	0.368	0.420
$\Delta t = 0.02, R = 1I$	0.686	0.686	0.686	6.158	0.689	0.691	11.370
$\Delta t = 0.002, R = 0.01I$	1.516	1.517	1.517	1.508	1.512	1.508	1.507
$\Delta t = 0.2, R = 0.01I$	0.323	0.325	0.325	2.384	0.306	0.306	42.331

Scenarios with input changes are investigated for different measurement noise levels and different sampling times. Table XII provides a summary of MSE results for the case where the input flowrate increases from $800L/h$ to $1200L/h$ at the 1500th sampling interval. This set of results are consistent with the case without input changes or any disturbance. Algorithm 1, 2 and 3 provide reasonably good performance for EKF while Algorithm 2.1 and 3.1, which involve Euler approximations, generate unstable filters.

Table XII. MSEs for algorithms with a 50% input change

Algorithm	1	1.1	2	2.1	3	3.01	3.1
$\Delta t = 0.02, R = 0.01I$	1.297	1.297	1.297	17.090	1.296	1.296	380.76
$\Delta t = 0.02, R = 0.0001I$	0.978	0.986	0.986	7.658	1.095	1.140	24.654
$\Delta t = 0.2, R = 0.01I$	0.567	0.569	0.569	NaN	0.576	0.576	NaN

D. Conclusions

This paper compares different implementations of EKF for a class of continuous-time nonlinear models with discrete-time measurements. The algorithm can be classified by the sequence and methods used for linearization and discretization of nonlinear continuous-time models. The main difference between the methods lies in the methodology for computing the covariance matrix P. The conclusions are that continuously predicting P for EKF results in an accurate implementation. Evaluating P at discrete times can also be applied. In this case, good performance can be expected if P is obtained from integrating the continuous-time equation or if the sensitivity equation is used for computing the Jacobian matrix A. Instead, if a finite difference approach is chosen for computing A, the sampling time of the finite difference scheme needs to be small for acceptable performance of EKF. Approaches involving Euler approximations show good behavior only when the sampling interval is reasonably small and therefore they are not recommended for processes with long sampling time intervals.

CHAPTER VI

CONCLUSION

A. Summary of Contribution

This dissertation provides advanced techniques to be used in the general area of process monitoring through soft sensing of unmeasured variables. Specifically, new techniques for determining the optimal sensor network design and for using soft sensors for fault detection are presented. In addition, a thorough investigation into the implementation of the EKF state estimator reveals that some often used schemes offer inherently poor results.

First, Chapter III detailed a new method for performing fault detection using soft sensors (Serpas *et al.* , Submitted,2011). Although there is a wide variety of fault detection techniques available, most rely on some threshold for the variables of interest. When the variable is under this threshold it is considered normal operation, and when it is above it is declared as a fault. The approach derived and presented in this dissertation allows the calculation of the proper threshold to be used on unmeasured variables calculated by an Extended Kalman Filter. This proper threshold can then be applied using any of the many available fault detection methods that rely on a fault threshold. In addition, this work lays the conceptual foundation for future work extending this approach to other state estimators. The use of the proper threshold with the additional process information in the model provided by the EKF, the performance of fault detection is significantly improved. This enhancement is illustrated through several case studies.

The second contribution, communicated in Chapter IV, is a new generally applicable method for determining the optimal sensor network design for a nonlinear

system (Serpas *et al.* , Submitted,2012). Previous to this work, most of the literature in sensor network design focused on a single sensor or on sensors strictly for steady state linear systems. The approach outlined here is specifically designed for large sensor networks in dynamic nonlinear systems. The approach outlined takes into account information redundancy of multiple sensors by using the determinant of the empirical observability grammian. This problem formulation results in a Mixed Integer Nonlinear Programming problem. In order to solve this nontrival problem, the technique incorporates a recently developed hybrid approach for solving MINLP's. The accuracy and efficiency of the presented method is shown in two case studies, illustrating that this method is indeed feasible for use in full scale industrial application.

Thirdly, a thorough investigation into the implementation of the EKF state estimator is detailed in Chapter V (Serpas *et al.* , 2009). EKF is one of the most commonly implemented state estimator, and is the performance benchmark for future research. Despite the maturity of the EKF in the literature and its important role in the area, it is easily wrongly implemented. The presented work details several of the possibilities for implementation and compares accuracy results.

B. Future Work

Just as diverse as the wide range of topics within the field of soft sensor research is the set of open questions. Although there has been continual advancement in both theory and application of soft sensing within petrochemical process industries, the increasing desire for safer, cleaner, and cheaper processes demands more development in all areas of advanced process monitoring and control.

1. Sensor Network Design

Within the area of sensor network design, there are extensions of the presented work that promise to bring useful results. Using the framework of maximizing the determinant of an information matrix, the problem formulation could be extended to include the state estimate error covariance matrix, P_k, given by equation (2.14). This second matrix would allow the inclusion of information about the reliability of the model and the measurement technology with respect to certain variables. Without the state estimate error covariance matrix the sensor network may be optimal for deterministic systems resulting from perfect system models and perfect sensors, but may dictate placing a sensor on a variable that is actually unreliable due to model or measurement noise. One method for including P_k would be using the model reduction technique of balancing, which "balances" information from the observability and controllability gramians in choosing which states to truncate (Hahn & Edgar, 2002).

In addition to the inclusion of more information in the problem formulation, the presented approach should be augmented by the inclusion of cost terms. There has been much research into the optimization of sensor networks by focusing on the capital and maintenance monetary costs, but much of the time this research does not include the information variability between different process variables. Therefore, from a process management standpoint, the truly optimal sensor network problem formulation must combine the methods of the presented work and those focusing monetary considerations.

Finally, because of the wide variety of available techniques available, in addition to offering combinations mentioned above, future research into comparisons between methods would be very useful for control engineer practitioners in industry.

2. Fault Detection Using Model Based Soft Sensors

The presented work explored the question of how to determine fault boundaries for systems that use soft sensors. The results include a procedure that can be used for Kalman Filter and EKF applications. The next step should be to develop similar techniques for other commonly used state estimators such as Moving Horizon Estimators, Unscented Kalman Filters and Particle Filters. These techniques should also then be compared to the results found from performing fault detection using EKF, as this would add to the ongoing discussion of relative performance of different state estimators.

3. Soft Sensor Design - Arrival Cost for Moving Horizon Estimators

Moving Horizon Estimation is becoming increasingly popular strategy for state estimation due to fact that it allows for directly dealing with system model nonlinearities and state variable constraints. The Moving Horizon Estimator was developed to use an optimization formulation on a fixed in size, but moving, window of data. This method truncates the previous data as the window moves forward in time. This approximation requires the use of an initialization term known as the arrival cost. The arrival cost is responsible for including information about the system at the beginning of the horizon. More specifically, this represents the *a priori* density function. If the system is linear and unconstrained, the Kalman Filter provides the exact *a priori* density in the form of its state estimate and error covariance matrix.

For general nonlinear systems, the exact representation of this density is non-trivial, and is typically approximated in one of many available methods. The main open area of research for MHE is in determining good approximations of the arrival cost. This is critical to the industrial application of MHE because the poorer the

estimation of the arrival cost, the larger the estimation horizon must be in order to provide satisfactory performance. However, the larger the estimation horizon, the longer the state estimation takes at each time step. The goal of arrival cost research is to find a useful compromise in this tradeoff (Ungarala, 2009). There is a need for continued development both of better algorithms for execution, and possibly for the introduction of a new state filtering approach in order to make calculation of arrival cost for MHE more tractable for industrial application.

Moving Horizon Estimation remains one of the most promising solutions to the constrained nonlinear online estimation problem. However, due to high computation costs in the complex optimization problem, live industrial implementation is restricted to low horizon lengths. This requires an accurate estimation of the arrival cost. The best arrival cost calculation seems to be theoretically difficult, especially when considering the inclusion of state constraints.

REFERENCES

Åkesson, B.M., Jørgensen, J.B., Poulsen, N.K., & Jørgensen, S.B. 2008. A generalized autocovariance least-squares method for Kalman filter tuning. *Journal of Process control*, **18**(7), 769–779.

Ali, Y., & Narasimhan, S. 1993. Sensor network design for maximizing reliability of linear processes. *AIChE Journal*, **39**(5), 820–828.

Alspach, D. 1974. A parallel filtering algorithm for linear systems with unknown time varying noise statistics. *Automatic Control, IEEE Transactions on*, **19**(5), 552–556.

Anderson, B.D.O., & Moore, J.B. 1979. *Optimal filtering*. Vol. 11. Prentice-hall Englewood Cliffs, NJ.

Atkinson, A.C., Donev, A.N., & Tobias, R. 2007. *Optimum experimental designs, with SAS*. Vol. 34. Oxford University Press, USA.

Bagajewicz, M., & Sanchez, M. 2000. Cost-optimal design of reliable sensor networks. *Computers & chemical engineering*, **23**(11-12), 1757–1762.

Bagajewicz, M.J. 1997. Design and retrofit of sensor networks in process plants. *AIChE journal*, **43**(9), 2300–2306.

Belanger, P.R. 1974. Estimation of noise covariance matrices for a linear time-varying stochastic process. *Automatica*, **10**(3), 267–275.

Berry, J., Hart, W.E., Phillips, C.A., Uber, J.G., & Watson, J.P. 2006. Sensor

placement in municipal water networks with temporal integer programming models. *Journal of water resources planning and management*, **132**, 218.

Bohlin, T. 1976. Four cases of Identification of changing systems. *System identification: Advances and case studies*, **126**, 441.

Bonami, P., Biegler, L.T., Conn, A.R., Cornujols, G., Grossmann, I.E., Laird, C.D., Lee, J., Lodi, A., Margot, F., & Sawaya, N. 2008. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, **5**(2), 186–204.

Brewer, J., Huang, Z., Singh, A.K., Misra, M., & Hahn, J. 2007. Sensor network design via observability analysis and principal component analysis. *Industrial & Engineering Chemistry Research*, **46**(24), 8026–8032.

Brogan, W.L. 1985. *Modern control theory*. Prentice-Hall, Englewood Cliffs, NJ.

Bunn, Derek W. 1981. Recursive estimation of the observation and process noise covariances in online Kalman filtering. *European Journal of Operational Research*, **6**(3), 302 – 308.

Busch, J., Kühl, P., Schlöder, J.P., Bock, H.G., & Marquardt, W. 2009. State estimation for large-scale wastewater treatment plants. *In: ADCHEM09-international symposium on advanced control of chemical processes, Istanbul*.

Carew, B., & Bélanger, P. 1973. Identification of optimum filter steady-state gain for systems with unknown noise covariances. *Automatic Control, IEEE Transactions on*, **18**(6), 582–587.

Chen, T., Morris, J., & Martin, E. 2005. Particle filters for state and parameter estimation in batch processes. *Journal of Process Control*, **15**(6), 665–673.

Chen, W., Bakshi, B.R., Goel, P.K., & Ungarala, S. 2004. Bayesian estimation via sequential Monte Carlo sampling: unconstrained nonlinear dynamic systems. *Industrial & engineering chemistry research*, **43**(14), 4012–4025.

Chetouani, Y. 2004. Fault detection by using the innovation signal: application to an exothermic reaction. *Chemical Engineering and Processing*, **43**(12), 1579–1585.

Chiang, LH, Russell, EL, & Braatz, RD. 2000. Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometrics and intelligent laboratory systems*, **50**(2), 243–252.

Chu, Y. 2011. *Parameter estimation of complex systems from sparse and noisy data*. Ph.D. dissertation, Texas A&M University, College Station.

Damak, T., Babary, JP, & Nihtila, MT. 1993. *Observer design and sensor location in distributed parameter bioreactors*. Pergamon Press.

de Assis, AJ. 2000. Soft sensors development for on-line bioreactor state estimation. *Computers & Chemical Engineering*, **24**(2-7), 1099–1103.

Ding, SX. 2008. *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Verlag.

Ding, SX, Ding, EL, & Jeinsch, T. 1998. A numerical approach to analysis and design of FDI systems.

Ding, SX, Zhang, P, Frank, PM, & Ding, EL. 2004. Threshold calculation using LMI-technique and its integration in the design of fault detection systems. vol. 1. IEEE.

Ding, W., Wang, J., Rizos, C., & Kinlyside, D. 2007. Improving adaptive Kalman estimation in GPS/INS integration. *Journal of Navigation*, **60**(3), 517.

Emery, A F, & Nenarokomov, Aleksey V. 1998. Optimal experiment design. *Measurement Science and Technology*, **9**(6), 864.

Fortuna, L, Graziani, S, & Xibilia, MG. 2005. Soft sensors for product quality monitoring in debutanizer distillation columns. *Control Engineering Practice*, **13**(4), 499–508.

Fortuna, L, Graziani, S, Rizzo, A, & Xibilia, MG. 2007. *Soft sensors for monitoring and control of industrial processes*. Springer Verlag.

Fraleigh, LM, Guay, M., & Forbes, JF. 2003. Sensor selection for model-based real-time optimization: relating design of experiments and design cost. *Journal of Process Control*, **13**(7), 667–678.

Frank, Paul M. 1990. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results. *Automatica*, **26**(3), 459–474.

Frank, PM, Ding, SX, & Marcu, T. 2000. Model-based fault diagnosis in technical processes. *Transactions-Institute of Measurement and Control*, **22**(1), 45–102.

Georges, D. 1995. The use of observability and controllability gramians or functions for optimal sensor and actuator location in finite-dimensional systems. *Pages 3319–3324 vol.4 of: Decision and Control, 1995., In Proceedings of the 34th IEEE Conference*, vol. 4.

Gertler, J. 1991. Analytical redundancy methods in fault detection and isolation. vol. 1.

Grewal, M.S., & Andrews, A.P. 2001. Kalman filtering: theory and practice using MATLAB.

Hahn, J., & Edgar, T.F. 2001. A gramian based approach to nonlinearity quantification and model classification. *Industrial & engineering chemistry research*, **40**(24), 5724–5731.

Hahn, J, & Edgar, TF. 2002. An improved method for nonlinear model reduction using balancing of empirical gramians. *Computers & Chemical Engineering*, **26**(10), 1379–1397.

Hart, W.E., Laird, C.D., Watson, J.P., & Woodruff, D.L. 2012. *Pyomo - Optimization Modeling in Python*. Springer Optimization and its Applications. Springer.

Haseltine, E.L., & Rawlings, J.B. 2005. Critical evaluation of extended Kalman filtering and moving-horizon estimation. *Industrial & Engineering Chemistry Research*, **44**(8), 2451–2460.

Hermann, R., & Krener, A. 1977. Nonlinear controllability and observability. *Automatic Control, IEEE Transactions on*, **22**(5), 728–740.

Hilborn, CG, & Lainiotis, D.G. 1969. Optimal estimation in the presence of unknown parameters. *Systems Science and Cybernetics, IEEE Transactions on*, **5**(1), 38–43.

Himmelblau, DM. 1978. *Fault detection and diagnosis in chemical and petrochemical processes*. Elsevier Amsterdam.

Hofling, T, & Isermann, R. 1996. Fault detection based on adaptive parity equations and single-parameter tracking. *Control Engineering Practice*, **4**(10), 1361–1369.

Hou, M, & Mller, PC. 1994. Fault detection and isolation observers. *International Journal of Control*, **60**(5), 827–846.

Huang, Reklaitis, G. V., & Venkatasubramanian, Venkat. 2003. A Heuristic Extended Kalman Filter Based Estimator for Fault Identification in a Fluid Catalytic Cracking Unit. *Industrial & Engineering Chemistry Research*, **42**(14), 3361–3371. doi: 10.1021/ie010659t.

Isermann, R. 1984. Process fault detection based on modeling and estimation methods–A survey. *Automatica*, **20**(4), 387–404.

Isermann, R. 1997. Supervision, fault-detection and fault-diagnosis methods – An introduction. *Control Engineering Practice*, **5**(5), 639–652.

Isermann, R. 2006. *Fault-diagnosis systems*. Springer Berlin.

Joshi, S., & Boyd, S. 2009. Sensor selection via convex optimization. *Signal Processing, IEEE Transactions on*, **57**(2), 451–462.

Julier, SJ, & Uhlmann, JK. 1997. A new extension of the Kalman filter to nonlinear systems. vol. 3. Citeseer.

Julier, S.J., & Uhlmann, J.K. 2004. Unscented filtering and nonlinear estimation. *In Proceedings of the IEEE*, **92**(3), 401–422.

Kadlec, P, Gabrys, B, & Strandt, S. 2009. Data-driven Soft Sensors in the process industry. *Computers & Chemical Engineering*, **33**(4), 795–814.

Kadu, S.C., Bhushan, M., & Gudi, R. 2008. Optimal sensor network design for multirate systems. *Journal of Process Control*, **18**(6), 594–609.

Kalman, R.E. 1960. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering*, **82**(Series D), 35–45.

Kandepu, R., Foss, B., & Imsland, L. 2008. Applying the unscented Kalman filter for nonlinear state estimation. *Journal of Process Control*, **18**(7-8), 753–768.

Karasalo, M., & Hu, X. 2011. An optimization approach to adaptive Kalman filtering. *Automatica*.

Kashyap, R. 1970. Maximum likelihood identification of stochastic linear systems. *Automatic Control, IEEE Transactions on*, **15**(1), 25–34.

Katipamula, S, & Brambley, MR. 2005. Methods for fault detection, diagnostics, and prognostics for building systems-a review, Part I. *HVAC&R Research*, **11**(1), 3–25.

Kiefer, JC. 1992. Optimum experimental designs. *Breakthroughs in Statistics: Foundations and basic theory*, **1**, 400.

Kister, H.Z. 1990. *Distillation operations*. Mcgraw-Hill Professional.

Kotecha, P.R., Bhushan, M., Gudi, RD, & Keshari, MK. 2008. A duality based framework for integrating reliability and precision for sensor network design. *Journal of process control*, **18**(2), 189–201.

Kramer, MA, & Mah, RSH. 1994. Model-based monitoring.

Lang, L., Chen, W., Bakshi, B.R., Goel, P.K., & Ungarala, S. 2007. Bayesian estimation via sequential Monte Carlo sampling–Constrained dynamic systems. *Automatica*, **43**(9), 1615–1622.

Legg, S.W., Wang, C., Benavides-Serrano, A.J., & Laird, C.D. 2012. Optimal Gas Detector Placement Under Uncertainty Considering Conditional-Value-At-Risk. *Journal of Loss Prevention in the Process Industries*.

Li, W, Yue, HH, Valle-Cervantes, S, & Qin, SJ. 2000. Recursive PCA for adaptive process monitoring. *Journal of Process Control*, **10**(5), 471–486.

Lin, B., Recke, B., Renaudat, P., Knudsen, J., & Jørgensen, S.B. 2005. Robust statistics for soft sensor development in cement kiln. *matrix*, **1**, 1.

Lin, B, Recke, B, Knudsen, JKH, & Jrgensen, SB. 2007. A systematic approach for soft sensor development. *Computers & Chemical Engineering*, **31**(5-6), 419–425.

Lopez, T., & Alvarez, J. 2004. On the effect of the estimation structure in the functioning of a nonlinear copolymer reactor estimator. *Journal of Process Control*, **14**(1), 99–109.

Lopez-Negrete, R., & Biegler, Lorenz T. 2011. Constrained Particle Filter Approach to Approximate the Arrival Cost in Moving Horizon Estimation. *Journal of Process Control*.

Lopez-Negrete, R., Patwardhan, S.C., & Biegler, L.T. 2009. Approximation of arrival cost in moving horizon estimation using a constrained particle filter. *Computer Aided Chemical Engineering*, **27**, 1299–1304.

Luenberger, D.G. 1964. Observing the state of a linear system. *Military Electronics, IEEE Transactions on*, **8**(2), 74–80.

Luong, M., Maquin, D., Huynh, C.T., & Ragot, J. 1994. Observability, redundancy, reliability and integrated design of measurement systems. *In Proceedings of 2nd IFAC Symposium on Intelligent Components and Instruments for Control Applications, SICICA*, vol. 94. Citeseer.

Madron, F., & Veverka, V. 1992. Optimal selection of measuring points in complex plants by linear models. *AIChE journal*, **38**(2), 227–236.

Mann, A.V., McKenna, S.A., Hart, W.E., & Laird, C.D. 2011. Real-time inversion in large-scale water networks using discrete measurements. *Computers & Chemical Engineering*.

Mariani, S., & Corigliano, A. 2005. Impact induced composite delamination: state and parameter identification via joint and dual extended Kalman filters. *Computer methods in applied mechanics and engineering*, **194**(50), 5242–5272.

Mehra, R. 1970. On the identification of variances and adaptive Kalman filtering. *Automatic Control, IEEE Transactions on*, **15**(2), 175–184.

Mehra, R. 1972. Approaches to adaptive filtering. *Automatic Control, IEEE Transactions on*, **17**(5), 693–698.

Mehra, R. 1976. Optimization of measurement schedules and sensor designs for linear dynamic systems. *Automatic Control, IEEE Transactions on*, **21**(1), 55–64.

Mellefont, DJ, & Sargent, RWH. 1978. Selection of measurements for optimal feedback control. *Industrial & Engineering Chemistry Process Design and Development*, **17**(4), 549–552.

Mller, PC, & Weber, HI. 1972. Analysis and optimization of certain qualities of controllability and observability for linear dynamical systems. *Automatica*, **8**(3), 237–246.

Montgomery, DC, Runger, GC, & Hubele, NF. 2009. *Engineering statistics*. Wiley.

Muske, K.R., & Georgakis, C. 2003. Optimal measurement system design for chemical processes. *AIChE Journal*, **49**(6), 1488–1494.

Musulin, E., Benqlilou, C., Bagajewicz, M.J., & Puigjaner, L. 2005. Instrumentation design based on optimal Kalman filtering. *Journal of Process Control*, **15**(6), 629–638.

Nahas, EP, Henson, MA, & Seborg, DE. 1992. Nonlinear internal model control strategy for neural network models. *Computers & Chemical Engineering*, **16**(12), 1039–1057.

Nguyen, D.Q., & Bagajewicz, M.J. 2008. Design of nonlinear sensor networks for process plants. *Industrial & Engineering Chemistry Research*, **47**(15), 5529–5542.

Odelson, B.J. 2003. *Estimating disturbance covariances from data for improved control performance*. Ph.D. thesis, University of Wisconsin.

Odelson, B.J., Lutz, A., & Rawlings, J.B. 2006a. The autocovariance least-squares method for estimating covariances: application to model-based control of chemical reactors. *Control Systems Technology, IEEE Transactions on*, **14**(3), 532–540.

Odelson, B.J., Rajamani, M.R., & Rawlings, J.B. 2006b. A new autocovariance least-squares method for estimating noise covariances. *Automatica*, **42**(2), 303–308.

Pan, Q., Yang, F., Ye, L., Liang, Y., & Cheng, Y. 2005. Survey of a kind of nonlinear filters-UKF. *Control and Decision*, **20**(5), 481.

Prakash, J., Patwardhan, S.C., & Shah, S.L. 2010. Constrained Nonlinear State Estimation Using Ensemble Kalman Filters. *Industrial & Engineering Chemistry Research*, **49**(5), 2242–2253.

Qin, SJ. 2003. Statistical process monitoring: basics and beyond. *Journal of Chemometrics*, **17**(8-9), 480–502.

Qu, C.C., & Hahn, J. 2009a. Process monitoring and parameter estimation via unscented Kalman filtering. *Journal of Loss Prevention in the Process Industries*, **22**(6), 703–709.

Qu, Cheryl C., & Hahn, Juergen. 2009b. Computation of arrival cost for moving horizon estimation via unscented Kalman filtering. *Journal of Process Control*, **19**(2), 358–363.

Raghuraj, R., Bhushan, M., & Rengaswamy, R. 1999. Locating sensors in complex chemical plants based on fault diagnostic observability criteria. *AIChE journal*, **45**(2), 310–322.

Rajamani, M.R., & Rawlings, J.B. 2009. Estimation of the disturbance structure from data using semidefinite programming and optimal weighting. *Automatica*, **45**(1), 142–148.

Rajamani, M.R., Rawlings, J.B., & Soderstrom, T.A. 2007. Application of a new data-based covariance estimation technique to a nonlinear industrial blending drum. *Submitted for publication*.

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

Rawlings, J. B., & Bakshi, B. R. 2006. Particle filtering and moving horizon estimation. *Computers & Chemical Engineering*, **30**(10-12), 1529–1541.

Robertson, D.G., & Lee, J.H. 1995. A least squares formulation for state estimation. *Journal of Process Control*, **5**(4), 291–299.

Romanenko, A., & Castro, J.A.A.M. 2004. The unscented filter as an alternative to

the EKF for nonlinear state estimation: a simulation case study. *Computers & Chemical Engineering*, **28**(3), 347–355.

Romanenko, A., Santos, L.O., & Afonso, P.A. 2004. Unscented Kalman filtering of a simulated pH system. *Industrial & Engineering Chemistry Research*, **43**(23), 7531–7538.

Scharf, LL, & Demeure, C. 1991. *Statistical Signal Processing: detection, estimation, and time series analysis*. Vol. 63. Addison-Wesley Reading, MA.

Scherpen, J.M.A. 1993. Balancing for nonlinear systems. *Systems & Control Letters*, **21**(2), 143–153.

Schmidt, GT, *et al.* . 1976. Practical aspects of kalman filtering implementation. *AGARD±LS±82, NATO Advisory Group for Aerospace Research and Development, London*.

Serpas, M., Qu, C., & Hahn, J. 2009. Investigation of Different Extended Kalman Filter Implementations. *In: Mary Kay O'Connor Process Safety Center International Symposium: Beyond Regulatory Compliance: Making Safety Second Nature*.

Serpas, M., Chu, Y., & Hahn, J. Submitted,2011. Fault Detection Approach for Systems Involving Soft Sensors. *Journal of Loss Prevention in the Process Industries*.

Serpas, M., Hackebeil, G., Laird, C., & Hahn, J. Submitted,2012. Sensor Location for Nonlinear Dynamic Systems via Controlability Analysis and Max-Det Optimization. *Computers and Chemical Engineering*.

Singh, A.K., & Hahn, J. 2005a. Determining optimal sensor locations for state and parameter estimation for stable nonlinear systems. *Industrial & Engineering Chemistry Research*, **44**(15), 5645–5659.

Singh, A.K., & Hahn, J. 2005b. On the use of empirical gramians for controllability and observability analysis. *Page 140 of: Proceedings of the American Control Conference*, vol. 1.

Singh, A.K., & Hahn, J. 2006. Sensor location for stable nonlinear dynamic systems: Multiple sensor case. *Industrial & Engineering Chemistry Research*, **45**(10), 3615–3623.

Sorenson, HW. 1985. *Kalman filtering: theory and application.* IEEE Press.

Soroush, M. 1998. State and parameter estimations and their applications in process control. *Computers and Chemical Engineering*, **23**(2), 229–245.

Stack, A.J., & Doyle III, F.J. 1997. Application of a control-law nonlinearity measure to the chemical reactor analysis. *AIChE journal*, **43**(2), 425–439.

Stanimirovic, O., Hoefsloot, H.C.J., De Bokx, P.K., & Smilde, A.K. 2008. Variable selection methods as a tool to find sensor locations for distributed parameter systems. *Industrial & Engineering Chemistry Research*, **47**(4), 1184–1191.

Sumana, C., & Venkateswarlu, Ch. 2009. Optimal selection of sensors for state estimation in a reactive distillation process. *Journal of Process Control*, **19**(6), 1024–1035.

Tonomura, O., Nagahara, S., Kano, M., & Hasebe, S. 2008. Sensor Location for Effective Fault Diagnosis in Micro Chemical Plants. *Pages 331–334 of: Proceedingis of Fifth International Conference on Foundations of Computer-Aided Process Operations (FOCAPO).*

Uci ski, D. 2005. *Optimal measurement methods for distributed parameter system identification.* CRC Press, Boca Raton, Florida.

Uci ski, D., & Patan, M. 2010. Sensor network design for the estimation of spatially distributed processes. *International Journal of Applied Mathematics and Computer Science*, **20**(3), 459–481.

Ungarala, S. 2009. Computing arrival cost parameters in moving horizon estimation using sampling based filters. *Journal of Process Control*, **19**(9), 1576–1588.

Ungarala, S. 2011. A direct sampling particle filter from approximate conditional density function supported on constrained state space. *Computers & Chemical Engineering*, **35**(6), 1110–1118.

Vachhani, P., Narasimhan, S., & Rengaswamy, R. 2006. Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation. *Journal of Process Control*, **16**(10), 1075–1086.

Van den Berg, FWJ, Hoefsloot, HCJ, Boelens, HFM, & Smilde, AK. 2000. Selection of optimal sensor position in a tubular reactor using robust degree of observability criteria. *Chemical engineering science*, **55**(4), 827–837.

Van Loan, C. 1978. Computing integrals involving the matrix exponential. *Automatic Control, IEEE Transactions on*, **23**(3), 395–404.

Van Welsenaere, RJ, & Froment, GF. 1970. Parametric sensitivity and runaway in fixed bed catalytic reactors. *Chemical engineering science*, **25**(10), 1503–1516.

Vande Wouwer, A., Point, N., Porteman, S., & Remy, M. 2000. An approach to the selection of optimal sensor locations in distributed parameter systems. *Journal of Process Control*, **10**(4), 291–300.

Vandenberghe, Lieven, Boyd, Stephen, & Wu, Shao-Po. 1998. Determinant Maxi-

mization with Linear Matrix Inequality Constraints. *SIAM J. Matrix Anal. Appl.*, **19**(2), 499–533.

Venkatasubramanian, V, & Chan, K. 1989. A neural network methodology for process fault diagnosis. *AIChE Journal*, **35**(12), 1993–2002.

Venkatasubramanian, V, Rengaswamy, R, Yin, K, & Kavuri, SN. 2003a. A review of process fault detection and diagnosis:: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, **27**(3), 293–311.

Venkatasubramanian, V, Rengaswamy, R, & Kavuri, SN. 2003b. A review of process fault detection and diagnosis:: Part II: Qualitative models and search strategies. *Computers & Chemical Engineering*, **27**(3), 313–326.

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

Waldraff, W., Dochain, D., Bourrel, S., & Magnus, A. 1998. On the use of observability measures for sensor location in tubular reactor. *Journal of Process Control*, **8**(5-6), 497–505.

Walter, E., & Pronzato, L. 1990. Qualitative and quantitative experiment design for phenomenological models - A survey. *Automatica*, **26**(2), 195–213.

Wan, E.A., & Van Der Merwe, R. 2000. The unscented Kalman filter for nonlinear estimation. *Pages 153–158 of: Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000.* IEEE.

Watanabe, K., & Himmelblau, D. M. 1983. Fault diagnosis in nonlinear chemical processes. Part I. Theory. *AIChE Journal*, **29**(2), 243–249.

Wynn, H.P. 1972. Results in the theory and construction of D-optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, 133–147.

Xiong, K., Zhang, HY, & Chan, CW. 2006. Performance evaluation of UKF-based nonlinear filtering. *Automatica*, **42**(2), 261–270.

Yang, Y., & Gao, W. 2006. An optimal adaptive Kalman filter. *Journal of Geodesy*, **80**(4), 177–183.

Zavala, V.M., Laird, C.D., & Biegler, L.T. 2008. A fast moving horizon estimation algorithm based on nonlinear programming sensitivity. *Journal of Process Control*, **18**(9), 876–884.

Zhang, Y., & Jiang, J. 2008. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, **32**(2), 229–252.

Zumoffen, D., & Basualdo, M. 2010. A Systematic Approach for the Design of Optimal Monitoring Systems for Large Scale Processes. *Industrial & Engineering Chemistry Research*, **49**(4), 1749–1761.

APPENDIX A

ALTERNATE DERIVATION FOR VARIANCE OF STATE ESTIMATE

An alternative for the computation of the covariance of $\hat{x}$ shown in Section B is presented in this Appendix. In the body of the work a solution technique performing the calculation relied on numerical techniques to solve the algebraic Riccati Equation. This is very useful as it is a form very familiar to those studying process systems engineering. However, an alternative solution exists that gives the equivalent covariance, but presents it in a final closed form. This is advantageous simply for studying the functional dependence the covariance has on other variables, such as the noise levels Q and R.

Equations (2.3), (2.4), (3.2) and (3.3) can be combined to yield:

$$\hat{x}_k = F\hat{x}_{k-1} + (B - LCB)u_k + LCx_k + Lv_k \tag{A.1}$$

where $F = A - LCA$ for simplicity of notation.

Since the process states and measurements are random variables, perturbed by random disturbances, the state estimate is also a random variable. The mean of the state estimate is given by

$$E[\hat{x}_k] = FE[\hat{x}_{k-1}] + +(B - LCB)u_k + LCE[\hat{x}_k] \tag{A.2}$$

Since u_k is deterministic for all k, its expected value is simply itself. The covariance matrix of the state estimate is given by

$$\begin{aligned} \mathrm{VAR}[\hat{x}_k] = {} & F\mathrm{VAR}[\hat{x}_{k-1}]F^T + F\mathrm{COV}[\hat{x}_{k-1}, x_k]C^T L^T \\ & + LC\mathrm{COV}[\hat{x}_{k-1}, x_k]F^T + LC\mathrm{VAR}[\hat{x}_{k-1}]C^T L^T + L\mathrm{VAR}[\hat{v}_k]L^T \end{aligned} \tag{A.3}$$

where it is assumed that $\text{COV}[\hat{x}_{k-1}, v_k] = 0$ and $\text{COV}[\hat{x}_k, v_k] = 0$. In the equation, the variance of the state is given by

$$\text{VAR}[x_k] = A\text{VAR}[x_{k-1}]A^T + GQG^T \tag{A.4}$$

The covariance of the state and the previous time step's state estimate can be calculated by substituting equation (A.1) and (2.3), i.e.

$$\begin{aligned}\text{COV}[\hat{x}_{k-1}, x_k] &= \text{COV}[F\hat{x}_{k-2} + (B - LCB)u_{k-1} \\ &\quad + LCx_{k-1} + Lv_{k-1}, Ax_{k-1} + Bu_k + Gw_k] \\ &= F\text{COV}[\hat{x}_{k-2}, x_{k-1}]A^T + LC\text{VAR}[x_{k-1}]A^T\end{aligned} \tag{A.5}$$

The equation is simplified as such since the terms $\text{COV}[\hat{x}_{k-2}, w_k]$, $\text{COV}[x_{k-1}, w_k]$, $\text{COV}[v_{k-1}, x_{k-1}]$, and $\text{COV}[v_{k-1}, w_k]$ are all equal to zero by the assumption stated in the text that the states and noise are not correlated.

If the system is assumed to be at steady state (or assuming it is a stationary random process), then $\text{VAR}[x_k] = \text{VAR}[x_{k-1}]$, and $\text{COV}[\hat{x}_{k-1}, x_k] = \text{COV}[\hat{x}_{k-2}, x_{k-1}]$. Using these equalities, equations (A.4) and (A.5) are simplified, using the vector operator.

$$\text{vec}(\text{VAR}[x_k]) = (I_{n^2} - A \otimes A)^{-1}(G \otimes G)\text{vec}(Q) \tag{A.6}$$

$$\text{vec}(\text{COV}[\hat{x}_{k-1}, x_k]) = (I_{n^2} - A \otimes F)^{-1}(G \otimes LC)\text{vec}(\text{VAR}[x_{k-1}]) \tag{A.7}$$

Substituting equation (A.6) and (A.7) into (A.3), and using the steady state formula,

$\mathrm{VAR}[x_k] = \mathrm{VAR}[x_{k-1}]$, gives the following equation:

$$\begin{aligned}\mathrm{VAR}[\hat{x}_k] = (I_{n^2} - (F \otimes F))^{-1} \\ [(LC \otimes F)(I_{n^2} - A \otimes F)^{-1}(A \otimes LC)(I_{n^2} - A \otimes A)^{-1}(G \otimes G)\mathrm{vec}(Q) + \\ (F \otimes LC)(I_{n^2} - A \otimes F)^{-1}(LC \otimes A)(I_{n^2} - A \otimes A)^{-1}(G \otimes G)\mathrm{vec}(Q) + \\ (LC \otimes LC)(I_{n^2} - A \otimes A)^{-1}(G \otimes G)\mathrm{vec}(Q) + (L \otimes L)\mathrm{vec}(R)]\end{aligned} \tag{A.8}$$

It can be concluded from equation (A.8) that the variance of the state estimate for the process shown in equations (2.3) and (2.4) is a linear function of the covariance of the model noise and the measurement noise, Q and R respectively. This is as intuition would predict, however, the dependence upon other system parameters is not intuitive at all.

APPENDIX B

PROOF THAT ALGORITHM 2 IS EQUIVALENT TO ALGORITHM 1.1.

It is adequate to show that P_{k+1}^{-} in both algorithms is identical for proving that Algorithm 2 is equivalent to Algorithm 1.1.

From Algorithm 2 in Table VII, it can be seen that,

$$\begin{aligned} P_{k+1}^{-} &= A_k P_k^{+} A_k^{'} + G Q_k G^{'} \\ &= e^{A(\hat{x}_k)T} P_k^{+} e^{A(\hat{x}_k)^{'}T} + \int_0^T e^{A(\hat{x}_k)\tau} G Q G^{'} e^{A(\hat{x}_k)^{'}\tau} d\tau, \text{where } A(\hat{x}_k) = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}_k}. \end{aligned} \tag{B.1}$$

Also Algorithm 1.1 states that

$$\dot{P} = A(\hat{x}_k)P + PA(\hat{x}_k)^{'} + GQG^{'}, \text{where } A(\hat{x}_k) = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}_k} \tag{B.2}$$

with $P(0) = P_k^{+}$ and $P(T) = P_{k+1}^{-}$.

Noting that $A(\hat{x}_k)$ in equation (B.1) and equation (B.2) is identical, it can be replaced by A as a change of notation.

If it can be shown that

$$P(t) = e^{At} P_k^{+} e^{A^{'}t} + \int_0^t e^{A\tau} G Q G^{'} e^{A^{'}\tau} d\tau \tag{B.3}$$

is the solution of

$$\dot{P} = AP + PA^{'} + GQG^{'}, \text{with } P(0) = P_k^{+}, \tag{B.4}$$

then it can be concluded that Algorithm 2 is identical to Algorithm 1.1.

It can be derived from equation (B.3) that for $t = 0$,

$$P(0) = e^0 P_k^+ e^0 + \int_0^0 e^{A\tau} GQG' e^{A'\tau} d\tau = P_k^+. \tag{B.5}$$

Taking the derivative of equation (B.3), it is derived that

$$\dot{P} = Ae^{At} P_k^+ e^{A't} + e^{At} P_k^+ e^{A't} A' + e^{At} GQG' e^{A't}. \tag{B.6}$$

On the other hand, substituting equation (B.3) into the right side of equation (B.4), the following equations

$$\begin{aligned} AP + PA' + GQG' = {} & Ae^{At} P_k^+ e^{A't} + \int_0^t Ae^{A\tau} GQG' e^{A'\tau} d\tau \\ & + e^{At} P_k^+ e^{A't} A' + \int_0^t e^{A\tau} GQG' e^{A'\tau} A' d\tau + GQG' \end{aligned} \tag{B.7}$$

are obtained.

Examining the second and the fourth terms in the above equation, it is found that

$$\begin{aligned} & \int_0^t Ae^{A\tau} GQG' e^{A'\tau} d\tau + \int_0^t e^{A\tau} GQG' e^{A'\tau} A' d\tau \\ & = \int_0^t d(e^{A\tau}) GQG' e^{A'\tau} + \int_0^t e^{A\tau} GQG' d(e^{A'\tau}) \\ & = \int_0^t d(e^{A\tau} GQG' e^{A'\tau}) \\ & = e^{A\tau} GQG' e^{A'\tau} \Big|_0^t \\ & = e^{At} GQG' e^{A't} - GQG' \end{aligned} \tag{B.8}$$

Therefore equation (B.7) becomes

$$AP + PA' + GQG' = Ae^{At} P_k^+ e^{A't} + e^{At} P_k^+ e^{A't} A' + e^{At} GQG' e^{A't}. \tag{B.9}$$

Comparing equation (B.6) to equation (B.9), it can be shown that

$$\dot{P} = AP + PA^{'} + GQG^{'}. \tag{B.10}$$

Taking equation(B.5) into account, it can be concluded that equation (B.3) is the solution of equation (B.4). Therefore Algorithm 2 is equivalent to Algorithm 1.1.

VITA

Mitchell Roy Serpas received his Bachelors of Science Degree in chemical engineering from Louisiana State University in May 2008. He entered the chemical engineering graduate program at Texas A&M University in August 2008. He received his Doctor of Philosophy Degree in August of 2012. His area of interest is within the broad area of process systems engineering in chemical engineering research and includes advanced process monitoring and control techniques for application to industrial processes. His contribution specifically is concerning the areas of fault detection, sensor network design, and state estimator application.

Dr. Serpas may be reached c/o Dr. Juergen Hahn at Artie McFerrin Department of Chemical Engineering, Texas A&M University, 3122 TAMU, College Station, TX 77843. His email is mitchserpas@gmail.com.

The typist for this thesis was Mitch Serpas.