Use of Multiple Models and Qualitative Knowledge for On-line Moving Horizon Disturbance Estimation and Fault Diagnosis

Edward P.Gatzke, Francis J. Doyle III*

Department of Chemical Engineering

University of Delaware

Newark, DE 19716

Abstract

An integrated fault detection, fault isolation, and parameter estimation technique is presented in this paper. Process model parameters are treated as disturbances that dynamically affect the process outputs. A moving horizon estimation technique minimizes the error between process and model measurements over a finite horizon by calculating model parameter values across the estimation horizon. To implement qualitative process knowledge, this minimization is constrained such that only a limited number of different faults (parameters) may change during a specific horizon window. Multiple linear models are used to capture nonlinear process characteristics such as asymmetric response, variable dynamics, and changing gains. Problems of solution multiplicity and computational time are addressed. Results from a nonlinear chemical reactor simulation are presented.

Keywords: Moving Horizon Estimation, Multiple Linear Models, Mixed Integer Programming, Disturbance Estimation

*Author to whom correspondence should be addressed: fdoyle@udel.edu

1 Introduction

Fault detection is a critical problem for the chemical process industries. In order to ensure both safe operations and quality production, process faults must be detected and isolated. If a fault is considered as a continuous disturbance that disrupts a process, the extent of the disturbance should be estimated in order for corrective actions to be taken. Using model-based approaches, parameter estimation techniques can provide accurate estimates for the disturbance levels of a process. Linear estimation techniques can prove inadequate in cases involving nonlinear systems. On the other hand, nonlinear estimation techniques can quickly become intractable for problems of moderate size. Industrial systems typically have many unmeasured disturbances which can and should be estimated for use in control strategies. Excessive change in an unmeasured disturbance can be considered a system fault. Such faults may not be a catastrophic process event, but rather a change in the system that adversely affects the system properties. A method for quickly and accurately detecting faults and estimating parameter values is desirable to keep processes running safely and on-target. The integrated problem of fault detection, isolation, and estimation for nonlinear systems incorporates concepts from the fields of fault diagnosis, state estimation, and nonlinear modeling.

Fault detection and isolation has become an important topic for chemical engineers attempting to minimize process down time and prevent industrial accidents. A typical approach for fault detection and diagnosis is to compare the process model outputs and actual process values [1, 32, 33]. Such residual methods for isolation can be accomplished by different types of analysis. The various methods include: threshold tests, fuzzy logic, neural networks, and others [14]. A similar method for detection and isolation of process faults is Principal Component Analysis (PCA). PCA methods typically use a low-order multi-dimensional steady state representation of a process for

generation and analysis of residuals [9, 10, 27]. Quantitative rule-based diagnosis and root cause analysis methods have been presented in numerous sources. A successful method for diagnosis is based on digraph representations of qualitative process variable states and their interactions [15, 21]. This method is effectively an efficient method for generating knowledge about a system in the form of expert rules. A more general approach for diagnosis and analysis is the use of a generic rule structure for representation of qualitative process knowledge [8]. Essentially, fault detection methods all use some type of process model and reasoning method for detection and isolation. This is effective for fault detection and isolation on a qualitative level, but in many cases it is desirable to calculate an estimate of the current system state.

If a process fault is treated as a time-dependent continuous parameter, the resulting problem can be considered a traditional state estimation problem. One favored state estimation approach, Kalman filtering, has been used extensively to solve linear estimation problems. Nonlinear estimation techniques have also found application (for an overview, see [20]). In particular, the Extended Kalman Filter (EKF) has been used extensively for nonlinear estimation [6, 5, 24, 30] and is based on higher order approximations of a nonlinear state space process model. Moving horizon estimation methods can produce results similar to those of Kalman filtering. Moving horizon methods solve an on-line optimization problem at every sampling time using a finite set of process data and a model of the system [3, 12, 22, 23, 26, 31]. The advantage of moving horizon methods is the ability to include various types of constraints in the optimization problem. One should keep in mind the similarity of moving horizon estimation methods to moving horizon optimal control methods such as MPC and DMC. Both the control and the estimation problems use receding horizon approaches and solve an optimization problem at each sampling time.

In many situations, a nonlinear state space process model may not be available or convenient for

use. Multiple model approaches hold great promise for nonlinear systems because of the ability to represent a complex nonlinear process using established and tractable linear techniques. In typical multiple model applications, the complex nonlinear system is partitioned into regions where local models are assumed to accurately represent the system. Many different methods for partitioning model spaces and switching between models have been proposed (see [19] for a general discussion of multiple models). Fuzzy logic model selection [17, 25] and Gaussian model selection [2, 13, 16] have both been used, and a multiple model moving horizon approach was proposed in [3]. A multiple model approximation has been applied to fault diagnosis in [1]. These different methods all assume that the behavior will be qualitatively similar to the local approximations of the real nonlinear system.

Fault diagnosis usually implies that qualitative rules expressing knowledge about a process are used for diagnosing a root cause. The state estimation problem can be treated as a moving horizon optimization problem. In this work, we combine the goals of estimation and diagnosis, resulting in a problem that can be formulated and solved using Mixed Integer (MI) optimization methods. Related work on Mixed Integer Quadratic Programming (MIQP) approaches to process control and fault diagnosis have recently appeared [4, 18, 28, 29]. This type of formulation expresses qualitative rules about a system as constraints involving integer variables. The previous work has demonstrated the usefulness of the method on small scale linear systems using a quadratic objective function.

2 Estimation Methodology

The proposed method generates parameter estimates for a process, given process measurements and system models. The method uses multiple linear models to approximate the actual values of the true nonlinear system. At each sampling time, an optimization problem is solved to minimize the error between the past process measurements and the process model for a horizon of limited size. The optimization problem is solved at each sampling time using this limited set of process measurements and past information from process estimates. This problem can become computationally demanding. As a result, efficient solution strategies must be used.

The proposed estimation method uses a Mixed Integer Linear Programming (MILP) formulation. Boolean variables taking the value 0 or 1 are used to express the absence or presence of a fault, respectively. Using a Linear Programming (LP) formulation creates an optimization problem that minimizes either the total absolute error or maximum absolute error between the system measurements and the model predictions. Other estimation techniques use a Quadratic Problem (QP) formulation that minimizes the 2-norm of the measurement-model error. Solving the less computationally demanding LP formulation proposed in the present work allows for real-time solution of complex problems, while still yielding accurate results.

The proposed method also makes use of multiple linear models. The multiple model formulation empowers the estimation method with the ability to represent system nonlinearities, such as asymmetric response, changing system gains, and variable system dynamics. This results in improved parameter estimates as compared to the use of single linear methods. The principle drawbacks of the multiple model approach are the need for multiple accurate system models and the increased number of decision variable in the optimization problem. The following sections

detail the formulation and solution of the horizon-based estimation problem.

2.1 Formulation

Assume that there is an impulse-response model of the process output behavior as a function of the given disturbances. Let $\widehat{y}$ be the estimated model output response to the disturbances Θ. The actual process measurement residuals are given as y. The vectors y and $\widehat{y}$ are composed of the vectors for each of the individual outputs, y_o and $\hat{y}_o$. The index o takes values from 1 to n_o, n_o being the total number of system outputs. The disturbance vector Θ is developed from the concatenation of the individual vectors for separate disturbances, Θ_j. The current formulation solves the following optimization problem for the vector of disturbances Θ:

$$\min_{\Theta} \sum_{i=k-H+1}^{k} || \, Q \, (\, y(i) - \widehat{y}(i) \,) \, ||_1 + || \, R \, \Delta\Theta(i) \, ||_1 \tag{1}$$

subject to the constraints:

$$\Delta\Theta_j(i) = \Theta_j(i) - \Theta_j(i-1) \quad \forall i, j \tag{2}$$

$$\Theta_j(i) = \theta_{j,1}(i) + ... + \theta_{j,n}(i) + ... + \theta_{j,n_j}(i) \quad \forall i \tag{3}$$

$$\widehat{y}_o = \sum_{j=1}^{F} M_{o,j,1} \, \theta_{j,1} + ... + M_{o,j,n} \, \theta_{j,n} + ... + M_{o,j,n_j} \, \theta_{j,n_j} \quad \forall o \tag{4}$$

$$\sum_{i=1}^{k-H+1} \sum_{n=1}^{n_j} \theta_{j,n}(i) \leq P f_j \quad \forall j \tag{5}$$

$$\sum_{j=1}^{F} f_j \leq S \tag{6}$$

$$0 \leq \theta_{j,n}(i) \quad \forall j, n, i \tag{7}$$

$$f_j \in \{0,1\} \tag{8}$$

Equation 1 details the objective function for this problem. In this equation, k is the current time, $y(i)$ is the actual process measurement vector at time i, $\widehat{y}(i)$ is the vector of process model estimates at time i, Q is a diagonal scaling matrix for weighting or normalizing the measurement error, R is a diagonal scaling matrix for weighting and normalizing the change in the parameter estimates, H is the length of the moving estimation horizon, and $\Theta_j(i)$ is the total parameter estimate for disturbance j at time i. The value $\Delta\Theta(i)$ is the change in a parameter from one time step to another defined in Equation 2. In Equation 2, $\Theta(k-H)$ is the value of the parameter estimate $\Theta(k-H+1)$ from the previous horizon window optimization result. For model representation, an impulse response formulation is used to describe the response of the process model estimate, $\widehat{y}(i)$, to changes in the system parameters, Θ. An individual system parameter (or fault) may be described at time i as $\Theta_j(i)$ where j is the index describing distinct faults. In the formulation, there may be multiple models for a single model parameter to model output pairing. The individual parameter estimate is the sum of the contributions from each model for that parameter, as stated in Equation 3. Here, n_j models are used for a given parameter variation. Mixing coefficients are not used to select models based on operating regime. Mixing coefficients would result in a mixed integer nonlinear programming problem. Instead, models are selected to best fit the available data during the optimization, given the constraints. Assume that there are n_o process measurements or model outputs available. The resulting impulse response formulation for a single model output $\widehat{y}_o$ is shown in Equation 4, where $M_{o,j,n}$ are the impulse response coefficients for the model corresponding to output o, parameter j, and model n for that parameter. F is the total number of disturbances modeled in the formulation. The index j represents the index of the possible faults

(or disturbances). The value n_j represents the total number of models used for a single fault or disturbance, j, and can be different for different values of j.

In order to handle asymmetric response to a parameter change, two distinct sets of models can be used for each disturbance. The values for all $\theta_{j,n}(i)$ (and therefore all $\Theta_j(i)$) can be constrained to be semi-positive. This means that for negative changes of a parameter, the formulation will use the negative of the impulse response matrix. This type of formulation assumes that both large positive and large negative parameter changes will not occur in a single parameter over a single window length. Cases where positive and negative changes in a parameter would affect the system over a single horizon calculation would be treated as two separate faults.

The formulation to this point only includes continuous variables and few constraints. Solving this formulation without additional constraints typically yields an underspecified problem that matches the measurement values across the horizon with the estimated model measurement values exactly. One can now make the assumption that only a limited number of disturbances can affect the system during a single horizon. This leads to the use binary decision variables to represent whether or not a fault has occurred in the current horizon window. Equations 5 and 6 are used in the formulation to express the logic constraints, for all fault parameters j, fault models $1...n_j$, and all time values across the horizon, i. The value P is a large number that ensures whenever a disturbance $\theta_{j,n}(i)$ is nonzero, f_j switches from 0 to 1. S is the total number of faults that can occur in a horizon window. In the presented formulation, response to positive and negative changes in a parameter are treated as separate fault events. The variables $\theta_{j,n}(i)$ are constrained to positive values in Equation 7 and Equation 8 limits the values of f_j to 0 or 1.

The size of the problem can now be calculated. Assuming that there are n_j models for each fault, the horizon length is H, and there are n_o system outputs, the total number of variables (for

the 1-norm case) is:

$$HFn_j + Hn_o + HF + F \tag{9}$$

where HFn_j are the number of variables for the parameter values $\theta_{j,n}$, Hn_o are the number of variables used to calculate the one-norm of the measurement-model error, HF is the number of variables used to calculate the one-norm of the change in parameters over the horizon, and F is the number of binary fault variables.

To set up the problem in the general form:

$$\min c^T \, [x_c^T \, x_i^T]^T \tag{10}$$

subject to the constraints:

$$a[x_c^T \, x_i^T]^T \quad \leq \quad b \tag{11}$$

$$lb \leq \quad [x_c^T \, x_i^T]^T \quad \leq ub \tag{12}$$

$$0 \quad \leq \quad x_c \tag{13}$$

$$x_i \quad \in \quad \{0, 1\} \tag{14}$$

The variables x_c and x_i, in the formulation are:

$$x_c \;\; = \;\; \left[\begin{array}{ccc} \Theta^T & |\Delta y|^T & |\Delta \Theta|^T \end{array} \right] \tag{15}$$

$$x_i \;=\; f \tag{16}$$

As stated before, the parameter values, Θ, are all assumed positive. This may be considered to be the extent of a given disturbance, even though the actual parameter may drift negative. The binary variables f are constrained to 0 or 1. Therefore, all the variables have positive values. The value of the objective function, J, can now be evaluated as:

$$J = M_Q \left|\Delta y\right| + M_R \left|\Delta\Theta\right| \tag{17}$$

The impulse response model for the system using multiple models can be written as follows:

$$\widehat{y} = M\Theta \tag{18}$$

where $\hat{y}$, M, θ, and y are given by:

$$\widehat{y} \;=\; \left[\begin{array}{cccccc} \widehat{y}_1^T & \ldots & \widehat{y}_o^T & \ldots & \widehat{y}_{n_o}^T \end{array}\right]^T \tag{19}$$

$$\widehat{y}_o \;=\; \left[\begin{array}{cccccc} \widehat{y}_o(k) & \ldots & \widehat{y}_o(k-i) & \ldots & \widehat{y}_o(k-H+1) \end{array}\right]^T \tag{20}$$

$$M \;=\; \left[\begin{array}{ccccc} M_1 & \ldots & M_j & \ldots & M_F \end{array}\right] \tag{21}$$

$$M_j \;=\; \left[\begin{array}{ccccc} M_{1,j,1} & \ldots & M_{1,j,n} & \ldots & M_{1,j,n_j} \\ \vdots & & \vdots & & \vdots \\ M_{o,j,1} & \ldots & M_{o,j,n} & \ldots & M_{o,j,n_j} \\ \vdots & & \vdots & & \vdots \\ M_{n_o,j,1} & \ldots & M_{n_o,j,n} & \ldots & M_{n_o,j,n_j} \end{array}\right] \tag{22}$$

$$\Theta = \left[\begin{array}{ccccc} \theta_1^T & \dots & \theta_j^T & \dots & \theta_F^T \end{array} \right]^T \tag{23}$$

$$\theta_j^T = \left[\begin{array}{ccccc} \theta_{j,1}^T & \dots & \theta_{j,n}^T & \dots & \theta_{j,n_j}^T \end{array} \right]^T \tag{24}$$

$$\theta_{j,n} = \left[\begin{array}{ccccc} \theta_{j,n}(k) & \dots & \theta_{j,n}(i) & \dots & \theta_{j,n}(k-H+1) \end{array} \right]^T \tag{25}$$

$$y = \left[\begin{array}{ccccc} y_1^T & \dots & y_o^T & \dots & y_{n_o}^T \end{array} \right]^T \tag{26}$$

$$y_o = \left[\begin{array}{ccccc} y_o(k) & \dots & y_o(k-i) & \dots & y_o(k-H+1) \end{array} \right]^T \tag{27}$$

To calculate Δy, the following constraint is used:

$$|y - \hat{y}| \leq \Delta y \tag{28}$$

where the process measurements y are known and the model $\hat{y} = M\Theta$ can be substituted so that the constraint can be written as:

$$\left[\begin{array}{cc} -M & -I \end{array} \right] \left[\begin{array}{c} \Theta \\ \Delta y \end{array} \right] \leq -y \tag{29}$$

$$\left[\begin{array}{cc} M & -I \end{array} \right] \left[\begin{array}{c} \Theta \\ \Delta y \end{array} \right] \leq y \tag{30}$$

The constraint to calculate $\Delta\Theta$ is described by:

$$|\Theta(i) - \Theta(i-1)| \leq \Delta\Theta \tag{31}$$

and can be established by creating a matrix M_Θ such that

$$M_\Theta \theta = \Theta(i) - \Theta(i-1) \tag{32}$$

with the result being:

$$\begin{bmatrix} M_\Theta & -I \end{bmatrix} \begin{bmatrix} \Theta \\ \Delta\Theta \end{bmatrix} \leq 0 \tag{33}$$

$$\begin{bmatrix} -M_\Theta & -I \end{bmatrix} \begin{bmatrix} \Theta \\ \Delta\Theta \end{bmatrix} \leq 0 \tag{34}$$

To express the propositional logic constraints which forces variable f_j to a value of 1 whenever Θ is nonzero, the matrix M_P according to Equation 5 can be found such that:

$$\begin{bmatrix} -M_P & -PI \end{bmatrix} \begin{bmatrix} \Theta \\ f \end{bmatrix} \leq 0 \tag{35}$$

The total number of faults are constrained to be less than S by Equation 6.

The weights for model error, Q, can be used to develop a vector M_Q to appropriately weight $|\Delta \mathbf{y}|$. Similarly, M_R can be found for the parameter weights R penalizing $|\Delta\Theta|$. The complete optimization problem can now be described as:

$$\min[0\, M_Q^T\, M_R^T\, 0]^T\, [\Theta^T\ \Delta y^T\ \Delta\Theta^T\ f^T]^T \tag{36}$$

subject to the constraints:

$$\begin{bmatrix} -M & -I & 0 & 0 \\ M & -I & 0 & 0 \\ M_\Theta & 0 & -I & 0 \\ -M_\Theta & 0 & -I & 0 \\ -M_P & 0 & 0 & -PI \\ 0 & 0 & 0 & [1 \dots 1] \end{bmatrix} [\Theta^T \ \Delta y^T \ \Delta\Theta^T \ f^T]^T \leq \begin{bmatrix} -y \\ y \\ 0 \\ 0 \\ 0 \\ S \end{bmatrix} \tag{37}$$

$$0 \ \leq [\Theta^T \ \Delta y^T \ \Delta\Theta^T]^T \tag{38}$$

$$0 \leq \quad [\, f \,] \quad \leq 1 \tag{39}$$

$$f \quad \in \quad \{0, 1\} \tag{40}$$

2.2 Solution Method

Solving a large scale MILP problem can be computationally difficult. In this application, the optimization computation is expected to be reformulated using new data and solved at every sampling time. The use of multiple models for accommodation of the nonlinear system response increases the problem size. For a moderately sized problem, there may be hundreds of continuous decision variables. Even with increased computational power, the task can be daunting in real time. Improvements to the MILP solution strategy can dramatically decrease the solution time.

In general, when $S = 1$ (the one fault case), the solution for this type of formulation is trivial. In such a case, there are only F possible combinations of feasible integer solutions available to choose from. The formulation can be solved using an general form MILP routine, but having the

knowledge that there are only F possible cases allows one to find a much more direct solution. In each possible fault scenario, only the variables for the single fault are nonzero; the variable for all the other faults are constrained to zero by the propositional logic constraints. Setting up and solving F different LP problems for the single fault cases is computationally trivial. The resulting LP problem solution with the minimal objective function will be the overall MILP solution in the $S = 1$ case.

In the multiple fault case the problem becomes less trivial. There are many combinations of faults to explore and enumeration quickly becomes impossible. Standard MILP solution methods will evaluate a large LP problem involving many variables at each node in a binary branch and bound tree. Intelligently using the properties of branch and bound can serve to speed the computational process.

The proposed solution method is outlined as follows-

- Solve the small single fault problem for each of the F fault cases
- Order the best objective function fits from the F possible solutions
- Evaluate the MILP by modified branch and bound:
  - reduce problem size when possible (number of variables)
  - skip nodes if applicable
  - enumerate cases if useful

A traditional MILP solution strategy would solve a total LP relaxation at the root node, relaxing all integer constraints. The proposed MILP method can start at a node other than the root node,

proceeding by a modified branch and bound procedure. See Figure 1 for an illustration of the modified branch and bound search for a situation where $S = 2$ and $F = 20$.

For example, in the case with a maximum of 2 faults $(S = 2)$, assume that solving the single fault problems for each of the possible faults results in f_1 having the best fit, f_2 having the next best, and so on to f_F. This requires F solutions to small problems. These problems can be considered size 1, where there are $H(n_j + n_o + 1)$ continuous variables and 0 binary variables in each of the problems.

Next, the case with the two best faults ($f_1 = 1$ and $f_2 = 1$) is considered. Addition of the constraints $f_1 = 1$ and $f_2 = 1$ will create an integer feasible 2 fault solution. If $f_1 = 1$ and $f_2 = 1$, all other fault variables are constrained to 0 in this $\sum_{j=1}^{F} f_j \leq 2$ example. Therefore, one may solve this node as a moderately small problem (size 2 with $H(2 * (n_j + 1) + n_o)$ continuous variables). The objective function of this solution becomes the new best integer solution for the problem, which is also an integer feasible upper bound on the overall MILP problem.

The entire enumeration of possible integer solutions assuming $f_1 = 1$ numbers only F. These cases can all be quickly evaluated as moderately small size 2 problems. The best solution of these cases now becomes the best integer feasible upper bound solution. Using the branch and bound technique, the node corresponding to $f_1 = 0$, all other f_i unconstrained must now be evaluated. If the objective function of this node is larger than the integer feasible upper bound, the solution procedure can stop with the current best integer solution as optimal because any other integer feasible solution in this portion of the branch and bound tree ($f_1 = 0$) is guaranteed to have a less optimal solution. The solution of this node is a large LP problem, size $F - 1$, which takes much longer to solve than the smaller scale problems. Additionally, the LP solution procedure at this node can be warm started with a feasible starting point using the results from the original F small

optimization problems of size 1.

If the value of the objective function at this node is worse than the optimal integer solution, the true optimal solution could still lie in the $f_1 = 0$ branch of the tree. There are now $F - 1$ possible integer solution size 2 problems with the constraints $f_1 = 0$ and $f_2 = 1$. These moderately small problems can again be quickly enumerated. A new best integer feasible solution may be found during this enumeration. The node with constraints $f_1 = 0$, $f_2 = 0$, all other f_j unconstrained must now be solved. Again this node is a large LP problem, taking much longer than the smaller problems. The LP solution at this node can also be warm started with a feasible starting point using the results from the original F small optimization problems of size 1. If the value of the objective function at this node is greater than the current optimal integer feasible upper bound solution, the true optimal integer solution has been found.

The procedure can be repeated as needed until a valid optimal integer solution is found. Ultimately, in the worst case, this may require that all possible fault combinations be evaluated. It is expected that such cases will be rare in practice. This modified solution method intelligently modifies the MILP branch and bound solution routine such that large scale problems need not be evaluated. By solving smaller problems corresponding to single fault cases and exploiting the known constraints of the problem, a good integer solution can be found early in the computation. By exploiting the structure of the problem, the LP nodes for computation can be decreased in problem size, greatly increasing computational efficiency.

2.3 Formulation Issues

In some cases, use of multiple models to represent nonlinear systems can lead to biased disturbance estimates. For illustration purposes, consider the following simple example with the forcing

function d representing an exogenous disturbance:

$$\frac{dx}{dt} = -x - (d-1)^2,\ x_o = -1 \tag{41}$$

$$y = x + 1 \tag{42}$$

It is desirable to develop models of the system using step tests. The response to disturbance steps at time 0 from $d = 0$ to $d = 1$ and $d = 0$ to $d = 1.8$, sampled every time unit are shown in the second and third columns of Table 2.

This system exhibits input multiplicities as seen in the steady state locus shown in Figure 2. The scaled transient dynamic character of the response to step changes is identical, no matter the input level. The steady state gain for the system is higher for the lower level input than for the high level input. Using these two models in the current estimation formulation will result in biased parameter estimates. In order to see this, note that the scaled impulse response coefficients are given as M_1 and M_2 in columns four and five of Table 2. The gain for model M_1 is 1 and the gain for model M_2 is 0.2.

Now assume that a step disturbance occurs in the simulation at time $t = 0$. The response of the system to the step change for a level of 0 to 1.7 is seen in column six of Table 2. The best fit for this measurement data (assuming that $\Theta(0) = 0$) could use either model to fit the data. If there is a weight on $\Delta\Theta$, the resulting parameter estimate would be 0.51 using only the high gain model M_1 for the entire horizon length. This example demonstrates that biased estimates are possible in some cases. If the local models accurately capture the transient dynamic differences, the parameter estimates can be more accurate using multiple models. Even in biased estimate

situations, knowledge of the correct fault can be beneficial.

In another case, assume that only one model is used for each fault in a system with F possible disturbances. The impulse response for a single output system is:

$$y = M_1\Theta_1 + M_2\Theta_2 + ... + M_F\Theta_F \tag{43}$$

In some cases, the output response to a parameter change could also be attributed to the combination of other fault responses. Mathematically, this implies that a nonzero vector x can be found such that for a given disturbance model M_i,

$$M_i = \begin{bmatrix} M_1 & ... & M_j & ... & M_F \end{bmatrix} x,\ j \neq i \tag{44}$$

Removing the constraint limiting the maximum number of faults in the estimation formulation (letting $S = F$) can allow this to occur. With $S = F$, if output biases are assumed as faults, any parametric disturbance change could be re-created by the linear combination of the various output biases. In effect, the different faults for such a system are indistinguishable. Depending on the weighting matrices Q and R, either the actual disturbance, the linear combination of the output bias models, or a combination of both will be given as the parameter estimate result. Setting S to a value close to 1 rather than F allows the optimization to correctly identify the fault disturbance or disturbances.

2.4 Detectability Analysis: Degenerate Solutions

In some cases, the measured response to a fault may resemble the response of another fault or combination of faults. It is useful to know which faults are distinguishable from other faults or sets

of faults. As stated previously, one may assume that all measurements may potentially be biased, then the measurement response to a non-bias fault may be represented as either a change in the fault parameter or a combination of changes in the measurement biases. Only S different parameters may change over an optimization window, with these parameters constrained to be positive values. In this way, asymmetric response can be captured in the linear models. The traditional methods of checking the rank of an observability matrix fails in this formulation because the parameters are limited to positive values. A method for finding degenerate fault sets is desirable.

First, using a single model for each fault, assume that M_j is the impulse response matrix for fault j. In the entire system there may be F total faults, $M_j \in [M_1 ... M_F]$. Now given a subset containing N of the F faults, we wish to find out which models in the set $[M_1 ... M_F]$ may be used to represent the subset of models. For notational convenience, we define the subset in question as the first N models of the total set, $[M_1 ... M_N]$ of $[M_1 ... M_F]$. Defining an integer variable f_i which represents the boolean value of whether or not a model set is required to represent the subset, the MILP problem can be formulated as:

$$\min_{[s,z]} \sum_{j=1}^{F} f_j \tag{45}$$

subject to the constraints:

$$[M_1 ... M_N]s - [M_1 ... M_N ... M_F][z_1 ... z_N ... z_F]^T = 0 \tag{46}$$

$$\sum z_j - P f_j \leq 0, \ \forall j \tag{47}$$

$$\sum s \geq B \tag{48}$$

$$\sum_{j=1}^{N} f_j \leq N-1 \tag{49}$$

$$0 \leq s \leq U \tag{50}$$

$$0 \leq z \leq U \tag{51}$$

$$f_j \in [0,1] \tag{52}$$

where the vectors s and z_i are variables used to find any set of parameters that will produce similar output values across the horizon. Equation 46 is used to force s and z to use the models to find an equivalent output value across the horizon. Equation 47 (with P any suitable large value) forces f_j to a value of 1 when any element of z_j is nonzero. Equation 48 forces the elements of s to be a positive number. In Equation 49, the set of z_j is forced to represent a subset that is not the subset itself. The last three constraints, Equations 50, 51, and 52 ensure that the elements of s and z are constrained to be positive numbers and all f_j are binary.

The result of this problem is the minimal set of faults (set f_D) that could possibly be used in an optimization problem to represent a given subset of N models. The total number of faults in the degenerate set is D. The problem can be re-solved with the additional constraint:

$$\sum_{j \in f_D} f_j \leq D-1 \tag{53}$$

The new problem finds the minimal set of faults to represent the subset of models that is not the original set or the recently found set. This iterative procedure can be used to find all of the potential degenerate model sets for a set of models by solving the MILP and adding the new constraint, and then resolving. Eventually, too many constraints will be added (enough subsets will be excluded) and new sets of degenerate models cannot be found. Because of constraints on both the total

number of faults and weightings used in the moving horizon formulation, this analysis result does not mean that degenerate sets are likely to result when used in the moving horizon estimation process.

3 Results and Discussion

A nonlinear Continuous Stirred Tank Reactor (CSTR) model is used to generate process data. In particular, the well-studied Van der Vusse kinetic scheme is considered [7, 11]. This system exhibits many highly nonlinear characteristics, including: input multiplicity, gain sign change, asymmetric response, and both minimum and nonminimum phase behavior. These nonlinear characteristics are very prevalent at the most desirable operating point. See Figure 3 for a comparison of dynamic response to a step change in the feed flow rate. From this figure, one can see that inverse response is observed for changing to a low flow rate and not observed for changes to higher flow rates. As demonstrated from the process values at steady state, the process gain changes from positive to negative, causing input multiplicity. The maximum product concentration is achieved at the operating point where the convergence of changing gains and inverse to non-inverse response exists.

A full description of the Van der Vusse CSTR model may be found in [7] or [11]. A feed stream of feedstock A enters a reactor and reacts to form the desired product, B. The model assumes a first-order reaction for $A \Rightarrow B$ with two competing reactions $B \Rightarrow C$ and $2A \Rightarrow D$. The relevant mass and energy balances are as follows (assuming a well-mixed constant volume reactor) :

$$\frac{dC_A}{dt} = \frac{\dot{V}}{V_R}(C_{AO} - C_A) - k_1(\upsilon)C_A - k_3(\upsilon)C_A^2 \quad (54)$$

$$\frac{dC_B}{dt} = -\frac{\dot{V}}{V_R}C_B + k_1(\upsilon)C_A - k_2(\upsilon)C_B \quad (55)$$

$$\frac{d\upsilon}{dt} = \frac{\dot{V}}{V_R}(\upsilon_O - \upsilon) - \frac{(k_1(\upsilon)C_A\Delta H_{RAB} + k_2(\upsilon)C_B\Delta H_{RBC} + k_3(\upsilon)C_A^2\Delta H_{RAD})}{\rho C_p} \quad (56)$$

$$+\frac{k_w A_R}{\rho C_p V_R}(\upsilon - \upsilon_K) \quad (57)$$

$$\frac{d\upsilon_K}{dt} = \frac{1}{m_K C_{PK}}(F_{KC}C_{PK}(\upsilon_{ko} - \upsilon_k) + k_w A_R(\upsilon - \upsilon_K)) \quad (58)$$

$$k_i = k_{io}e^{(\frac{E_i}{\upsilon+273.15})} \quad (59)$$

Temperature-dependent Arrhenius reaction rates are assumed. The model has four states: concentration of A, concentration of B, reactor temperature, and cooling jacket temperature. With this process, it is desirable to produce as much product B as possible. The model parameters are given in Table 3. All simulation results were performed using MATLAB 5.3 and CPLEX 6.5. Computation time using a Sun Ultra Enterprise 3000 with a 366 MHz processor with 20 binary variables typically is on the order of 500-1000 seconds for each iteration, allowing for one or two faults in the estimation horizon and the general purpose CPLEX MILP optimization routine. Using the proposed modified optimization routine, the same problems can be solved in less than 60 seconds.

It is assumed that the four states of the CSTR are directly measurable. An initial steady state point was selected near the process optima. Step response models were developed for ten different parameter variations using a sampling time of 1 minute and 40 step response coefficients. The step response models were evaluated at $\pm$5% and $\pm$25% parameter variations for a total of 20 different faults, shown in Table 4. These variations represent the expected variation for parameters in the

system. A moving horizon window of size 25 was chosen to account for the system dynamics. R values were all taken as 1 and Q values were set to 0.001. R values can be modified to scale error estimates according to expected variation levels. Q values can be increased to smooth the parameter estimates.

The systems accurately distinguish between parameters in the single fault case ($S = 1$). All twenty faults can be distinguished and accurately estimated. Given a change in a parameter, this estimation method very accurately estimates the parameter value. For a representative example, see the measurements in Figure 4. These measurements are taken from the reactor model after a 10% increase in the feed flow rate at time $t = 5$ minutes. Figure 5 shows the results for estimating this 10% increase in the feed flow at time $t = 25$. The 10% increase affects the system 20 minutes before this estimation result was computed. The two models used to estimate the total parameter are shown. As one can see in Figure 5, both the 5% model and the 25% model are used when estimating a 10% disturbance. The two model estimates can be summed directly because the models were normalized before the optimization took place. In Figure 6, the same fault measurements are shown, but normally distributed noise ($\sigma^2 = 0.01$) is now added to the signals. The resulting estimates are shown in Figure 7. The accuracy of the resulting estimates degrades with the addition of significant noise, but the correct fault is detected. In Figure 8, the estimation results using single models for estimation are shown. As can be seen, the estimates are not as accurate as the multiple model noise-free case, with slight deviation occurring across the horizon. The correct fault is distinguished from the others, but the estimates are slightly in error when not using the multiple linear models.

The evolution of a fault over time is also of interest. This estimation method estimates the value of the parameters over the entire horizon, not just the current estimate or the initial value

estimate. In Figure 9, the actual input values over the estimation horizon for different times are shown. At each sampling time, new measurements are available and the estimation process is accomplished again. In Figure 10, the estimates over the moving horizon for different times are displayed. After the fault is fully evolved, the amount of error in the estimate becomes larger. At this point, the change in the measurements is becoming negligible, so any model error can lead to increased estimation error.

In some cases, disturbances may affect a system as a ramp rather than a step. Figures 11 and 12 show measurements and estimates at time $t = 25$ for a gradual increase in the feed flow rate starting at time $t = 5$ and slowly ramping up to +10% at time $t = 25$. The correct fault and accurate level are detected.

This estimation method works well in multiple fault scenarios as well. In the following example, the input feed flow rate is decreased by 20% at time $k - 25$ ($t = 0$) and the feed concentration is decreased by 10% at time $k - 15$ ($t = 10$). The output measurements for this parameter change are given in Figure 13. The overall parameter estimates for time k ($t = 25$) are shown in Figure 14. Here, the parameter estimates accurately picked the correct faults from the group of 20 and also accurately estimated the parameter values.

Degenerate solutions may be possible in this system. Considering all the sets of faults consisting of a single disturbance, the detectability analysis reveals that at least four other disturbance responses must be used to represent a single fault. For systems with multiple faults, the analysis could be used as needed to determine if degenerate solutions exist. Unique solutions should exist for this example for S taking values 1 or 2.

4 Conclusions

In this paper, we have described a moving horizon method for detecting and estimating parameter changes. The receding horizon formulation allows the use of a finite amount of data in the optimization problem. This method makes use of multiple linear models to capture the behavior of the actual nonlinear system. The problem is formulated as a MILP with an integer constraint on the total number of changing parameters over the horizon.

There are limitations to this method, some of which have been described in this article. The use of multiple linear models can lead to biased estimates in some cases. This will not always occur, but can be seen in some mathematical examples. The results from one problem produce a potential fault set. This set may not be the only set of faults that can accurately portray the existing data if degenerate fault sets are available. This means that multiple causes may exist. Due to optimization weights and a single optimal solution, it should be rare but possible to find different sets of faults yielding the same or similar horizon based estimation results. Computationally, solving a large scale MILP can be difficult. For multiple fault cases, enumerating all possible cases is typically a poor solution. An improved MILP solution method is proposed that can exploit the problem structure and existing information.

In the future, larger problems may be explored. In the proposed method, separate models for positive and negative parameter shifts as well as small and large parameter shifts were used. Use of linear models can allow for the solution of larger scale problems, but the system estimates should be less accurate for nonlinear systems. This may be a reasonable tradeoff in systems where only linear models are available and the process exhibits weak nonlinear character.

Acknowledgments

The authors would like to acknowledge financial support from the Office of Naval Research (grant NOOO-14-96-1-0695) and the University of Delaware. The comments of anonymous reviewers are also gratefully acknowledged.

Appendix: Nomenclature

a	generalized optimization problem constraint matrix
A_r	heat exchange area
b	generalized optimization problem vector
B	arbitrary positive value
c	generalized optimization problem objective coefficients
C_A	concentration of species A in the reactor
C_{AO}	concentration of species A entering the reactor
C_B	concentration of species B in the reactor
C_p	heat capacity of the reaction mixture
C_{PK}	heat capacity of the coolant
D	total number of faults in a degenerate solution
$\Delta\Theta$	variable representing the value of $\Theta(i) - \Theta(i-1)$
Δy	variable representing value of $y - \hat{y}$
ΔH_{Ri}	heat of reaction for reaction i
E_i	activation energy for reaction i

f	variable for the integer values representing separate faults
f_j	single fault for fault j
f_D	set of faults that can produce identical process response
F	total number of faults
F_K	flow rate of coolant into the jacket
H	horizon length
i	index for time
j	index for faults
J	value of the objective function
k_i	reaction rate coefficient for reaction i
k_{io}	Arrhenius pre-exponential factor for reaction i
k	current time
k_w	heat exchange coefficient
LP	Linear Programming problem
m_K	mass of coolant in jacket
M	combined impulse response matrix
M_j	impulse response matrix for a fault j
$M_{o,j,n}$	impulse response matrix relating output o to fault j using model n
M_P	matrix formed for the propositional logic constraint
M_Q	vector formed according to values of Q, the error weight on $\|\Delta\mathbf{y}\|$
M_R	vector formed according to values of R, the error weight on $\|\Delta\Theta\|$
M_Θ	matrix formed for the $\Delta\Theta$ constraint
MILP	a Mixed Integer Linear Programming problem

n	index for models for a fault j
n_j	total number of models for a fault
n_o	total number of process outputs
N	number of faults in the subset used for find a set of equivalent faults
o	index for the outputs
P	large value
Q	weighting matrix for Δy
Q_k	jacket energy transfer from reactor
QP	Quadratic Programming problem
R	weighting matrix for $\Delta\Theta$
ρ	density of reactor contents
s	variable used to in the degenerate fault set formulation, limited to be greater than 0
S	total number of faults allowed to occur over a horizon
t	time
Θ	total estimated parameter variable, limited to be greater than 0
$\Theta_j(i)$	total estimated parameter value for fault j at time i, limited to be greater than 0
$\theta_{j,n}(i)$	individual model parameter value for fault j, model n, at time i, limited to be greater than 0
U	arbitrary upper bound
υ	temperature of the reactor
υ_K	temperature in the cooling jacket
υ_{ko}	temperature of coolant entering the jacket
$\dot{V}$	volumetric flow rate of feedstock into the reactor
V_R	volume of the reactor

x_c	continuous variables in the optimization problem taking values greater than 0
x_i	integer variables in the optimization problem taking value 0 or 1
$y(i)$	process measurement residual vector at time i
$y_o(i)$	process measurement value for output o at time i
$\hat{y}(i)$	process model vector at time i
$\hat{y}_o(i)$	process model value for output o at time i
z	variable used in the degenerate fault set formulation, limited to be greater than 0

References

[1] P. B. Balle, D. Fussel, and O. Hecker. Detection and Isolation of Sensor Faults on Nonlinear Processes Based on Local Linear Models. In *Proc. American Control Conf.*, pages 468–472, Albuquerque, NM, 1997.

[2] P. B. Balle, D. Juricic, A. Rakar, and S. Ernst. Identification of Nonlinear Processes and Model Based Fault Isolation Using Local Linear Models. In *Proc. American Control Conf.*, pages 47–51, Albuquerque, NM, 1997.

[3] A. B. Banerjee, Y. Arkun, B. Ogunnaike, and R. Pearson. Estimation of Nonlinear Systems Using Linear Multiple Models. *AIChE J.*, 43(5):1204–1226, 1997.

[4] A. Bemporad, D. Mignone, and M. Morari. Moving Horizon Estimation for Hybrid Systems and Fault Detection. In *Proc. American Control Conf.*, pages 2471–2475, San Diego, CA, 1999.

[5] C. Chang and J. Chen. Implementation Issues Concerning the EKF-based Fault Diagnosis Techniques. *Chem. Eng. Sci.*, 50(18):2861–2882, 1995.

[6] C. Chang, K. Mah, and C. Tsai. A Simple Design Strategy for Fault Monitoring Systems. *AIChE J.*, 39(7):1146–1163, 1993.

[7] H. Chen, A. Kremling, and F. Allgöwer. Nonlinear Predictive Control of a Benchmark CSTR. In *Proc. of the European Control Conf.*, pages 3247–3252, Rome, Italy, 1995.

[8] L. W. Chen and M. Modarres. Hierarchical Decision Process for Fault Administration. *Comput. Chem. Eng.*, 16(5):425–448, 1992.

[9] R. Dunia and J. Qin. Subspace Approach to Multidimensional Fault Identification and Reconstruction. *AIChE J.*, 44(8):1813–1831, 1998.

[10] R. Dunia, S. J. Qin, T. F. Edgar, and T. J. McAvoy. Identification of Faulty Sensors Using Principal Component Anaylsis. *AIChE J.*, 42(10), 1996.

[11] S. Engell and K.-U. Klatt. Nonlinear Control of a Non-Minimum-Phase CSTR. In *Proc. American Control Conf.*, pages 2941–2945, San Francisco, CA, 1993.

[12] J. M. Flaus and L. Boillereaux. Moving Horizon State Estimation for a Bioprocess Modelled by a Neural Network. *Trans. Inst. Meas. Control*, 19(5):263–270, 1997.

[13] B. A. Foss, T. A. Johansen, and A. V. Sorensen. Nonlinear Predictive Control Using Local Models - Applied to a Batch Process. In *IFAC Symposium on Advanced Control of Chemical Processes*, pages 225–230, Kyoto, Japan, 1994.

[14] P. M. Frank and N. Kiupel. FDI with Computer-Assisted Human Intelligence. In *Proc. American Control Conf.*, pages 913–917, Albuquerque, NM, 1997.

[15] M. Iri, K. Aoki, E. O'Shima, and H. Matsyama. An Algorithm for Diagnosis of System Failures in the Chemical Process. *Comput. Chem. Eng.*, 3:489–493, 1979.

[16] T. A. Johansen and B. A. Foss. Constructing NARMAX Models Using ARMAX Models. *Int. J. Control*, 58(5):1125–1153, 1993.

[17] S. McGinnity and G. Irwin. Nonlinear State Estimation Using Fuzzy Local Linear Models. *Int. J. of Systems Science*, 28(7):643–656, 1997.

[18] D. Mignone, A. Bemporad, and M. Morari. A Framework for Control, Fault Detection, State Estimation, and Verification of Hybrid Systems. In *Proc. American Control Conf.*, pages 134–138, San Diego, CA, 1999.

[19] R. Murray-Smith and T. A. Johansen, editors. *Multiple Model Approaches to Modelling and Control*. Taylor and Francis, London, 1997.

[20] K.R. Muske and T.F. Edgar. *Nonlinear Process Control, M. Henson and D. Seborg*, chapter 6. Nonlinear State Estimation, pages 311–370. Prentice Hall, 1997.

[21] D. S. Nam, C. Han, C. Jeong, and E. S. Yoon. Automatic Construction of Extended Symptom-Fault Associations from the Signed Digraph. *Comput. Chem. Eng.*, 20:5605–5610, 1996.

[22] T. Ohtsuka and H. A. Fujii. Nonlinear Receding-Horizon State Estimation by Real-Time Optimization Technique. *Journal of Guidance, Control, and Dynamics*, 19(4):863–870, 1996.

[23] C.V. Rao and J. Rawlings. Nonlinear Moving Horizon State Estimation. In *International Symposium on Nonlinear Model Predictive Control: Assessment and Future Directions*, pages 146–163, Ascona, Switzerland, 1998.

[24] D. Robertson and J. H. Lee. Integrated State Estimation, Fault Detection, and Diagnosis for Nonlinear Systems. In *Proc. American Control Conf.*, pages 389–392, San Francisco, CA, 1993.

[25] A. Rueda. Approximation of Nonlinear Systems by Dynamic Selection of Linear Models. In *IEEE Canadian Conference on Electrical and Computer Engineering*, pages 270–273, Waterloo, Canada, 1996.

[26] L. P. Russo and R. E. Young. Moving-Horizon State Estimation Applied to an Industrial Polymerization Process. In *Proc. American Control Conf.*, pages 1129–1133, San Diego, CA, 1999.

[27] H. Tong and C. M. Crowe. Detecting Persistent Gross Errors by Sequential Analysis of Principal Components. *AIChE J.*, 43(5):1242–1249, 1997.

[28] M. L. Tyler and M. Morari. Qualitative Modeling Using Propositional Logic. In *AIChE Fall National Meeting, Chicago*, 1996.

[29] M. L. Tyler and M. Morari. Propositional Logic in Control and Monitoring Problems. *Automatica*, 35:565–582, 1999.

[30] K. Watanabe and D. M. Himmelblau. Incipient Fault Diagnosis of Nonlinear Processes with Multiple Causes of Faults. *Chem. Eng. Sci.*, 39(3):491–508, 1984.

[31] E. Yaz and N. Yildizbayrak. Moving Horizon control and Moving Window Estimation Schemes For Discrete Time-varying Systems. *Int. J. Sys. Sci.*, 18(8):1447–1456, 1987.

[32] K. Yin. Minmax Methods for Fault Isolation in the Directional Residual Approach. *IFAC Symposium, On-Line Fault Detection and Supervision in the Process Industies*, 1992.

[33] Q. Zhang, M. Basseville, and A. Benveniste. Fault Detection and Isolation in Nonlinear Dynamic Systems: A Combined Input-Output and Local Approach. *Automatica*, 34(11):1359–1373, 1998.

Figure Captions

Figure 1: Modified branch and bound example for dual fault case with 20 possible faults. Node numbers indicate solution order and node sizes represent the size of the corresponding LP relaxation.

Figure 2: Steady state locus for simple example, $\frac{dx}{dt} = -x - (d-1)^2$, $x_o = -1$, $y = x + 1$, showing input multiplicity.

Figure 3: Response of CSTR model to step changes in the feed flow rate at time 0, with the initial steady state feed flow = 14.9. Input multiplicity and both minimum and nonminimum phase behavior are apparent.

Figure 4: Measurements at time $t = 25$ for 10% increase in feed flow at time $t = 5$ without process noise.

Figure 5: Estimates at time $t = 25$ across the estimation horizon using multiple models and measurements for 10% increase in feed flow at time $t = 5$ without process noise. Squares indicate the total estimate, circles show the contribution from the +5% model, and crosses indicate contribution from the +25% model.

Figure 6: Measurements at time $t = 25$ for 10% increase in feed flow at time $t = 5$ with normally distributed noise ($\sigma^2 = 0.01$).

Figure 7: Estimates at time $t = 25$ across the estimation horizon using multiple models and measurements for 10% increase in feed flow at time $t = 5$ with normally distributed noise ($\sigma^2 = 0.01$). Squares indicate the total estimate, circles show the contribution from the +5% model, and crosses indicate contribution from the +25% model.

Figure 8: Separate estimates at time $t = 25$ using measurements from a 10% increase in feed

flow at time $t = 5$ using single models in both cases.

Figure 9: Actual parameter values over estimation horizon for 10% increase in feed flow at time $t = 5$.

Figure 10: Horizon parameter estimates for 10% increase in feed flow at time $t = 5$ using multiple models with constraint allowing a single fault across estimation horizon.

Figure 11: Measurements at time $t = 25$ for ramped change in in feed flow rate to +10% at time $t = 25$ starting at time $t = 5$ without process noise.

Figure 12: Estimates at time $t = 25$ using measurements for ramped change in in feed flow rate to +10% at time $t = 25$ starting at time $t = 5$ without process noise.

Figure 13: Measurements for dual fault case at time $t = 25$ with -20% input feed flow step at time $t = 0$ and -10% feed concentration step at time $t - 15$.

Figure 14: Estimates at time $t = 25$ using measurements for dual fault case with -20% input feed flow step at time $t = 0$ and -10% feed concentration step at time $t - 15$.

Table 1: Output response for example problem. Models M_1 and M_2 are developed from changing d from a value of 0 to values of 1 and 1.8, respectively.

Table 2: Van der Vusse CSTR model parameters.

Table 3: Potential faults for CSTR case study.

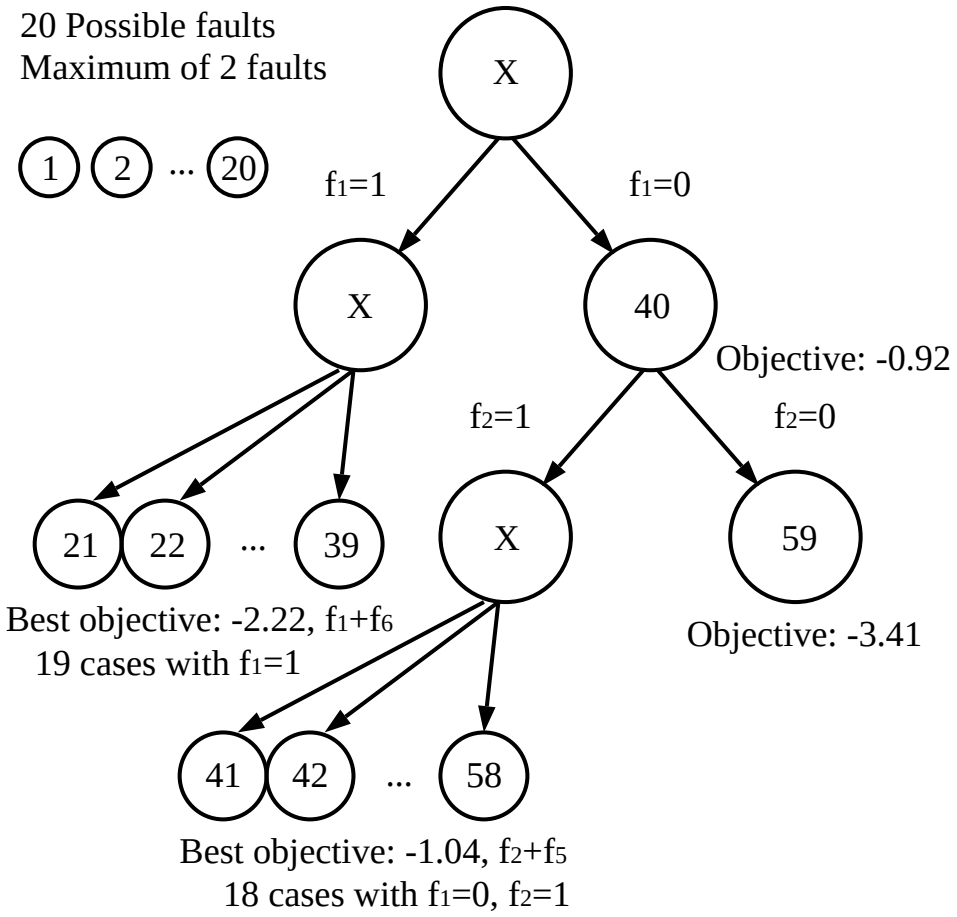

Figure 1: Modified branch and bound example for dual fault case with 20 possible faults. Node numbers indicate solution order and node sizes represent the size of the corresponding LP relaxation.

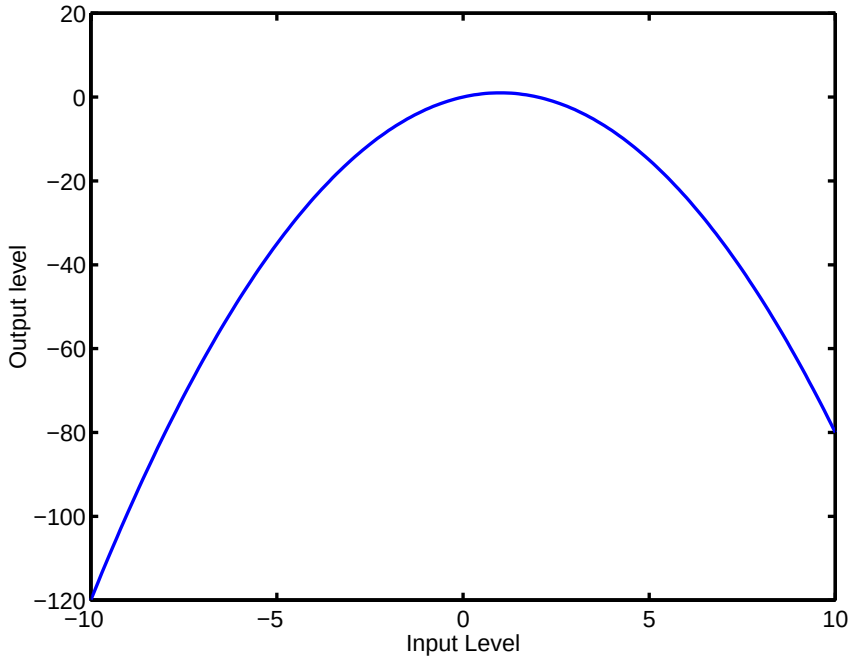

Figure 2: Steady state locus for simple example, $\frac{dx}{dt} = -x - (d-1)^2$, $x_o = -1$, $y = x + 1$, showing input multiplicity.

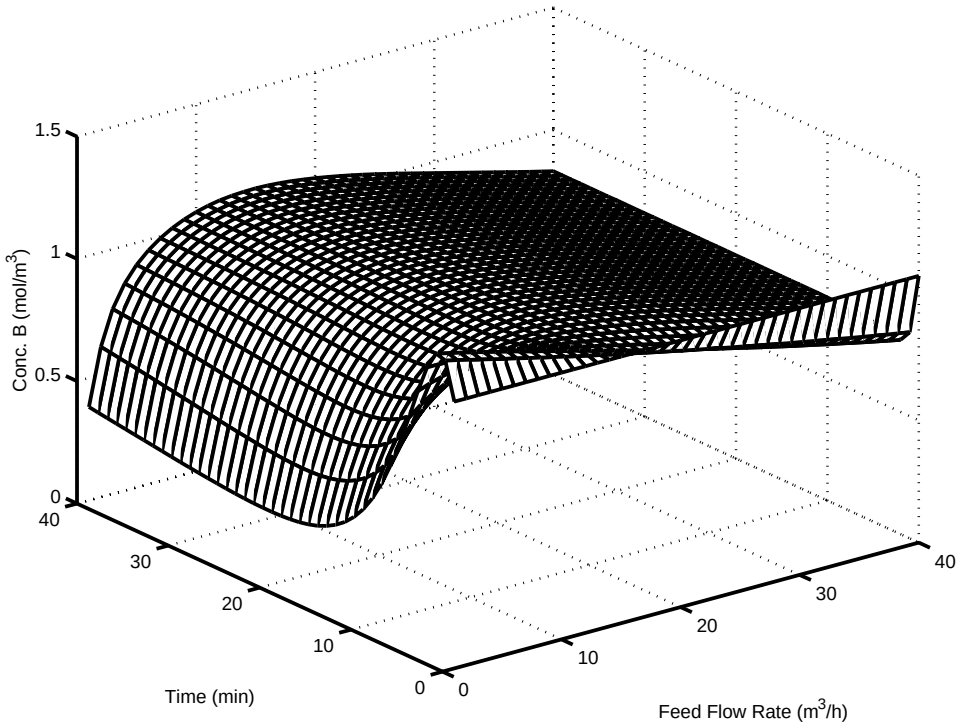

Figure 3: Response of CSTR model to step changes in the feed flow rate at time 0, with the initial steady state feed flow = $14.19\,\frac{l}{h}$. Input multiplicity and both minimum and nonminimum phase behavior are apparent.

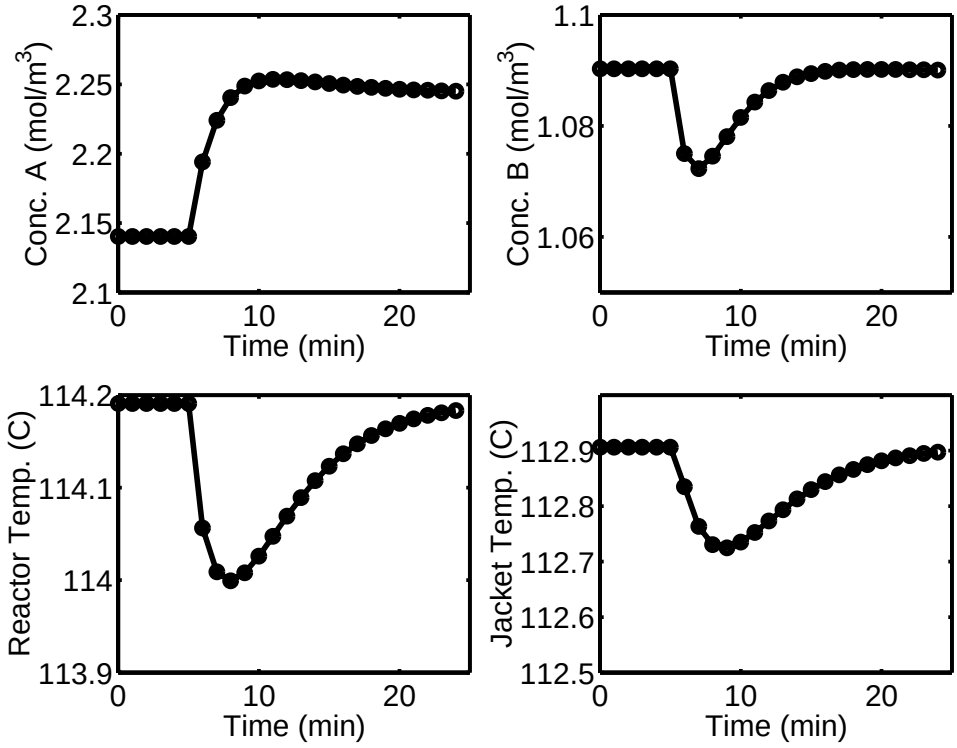

Figure 4: Measurements at time $t = 25$ for 10% increase in feed flow at time $t = 5$ without process noise.

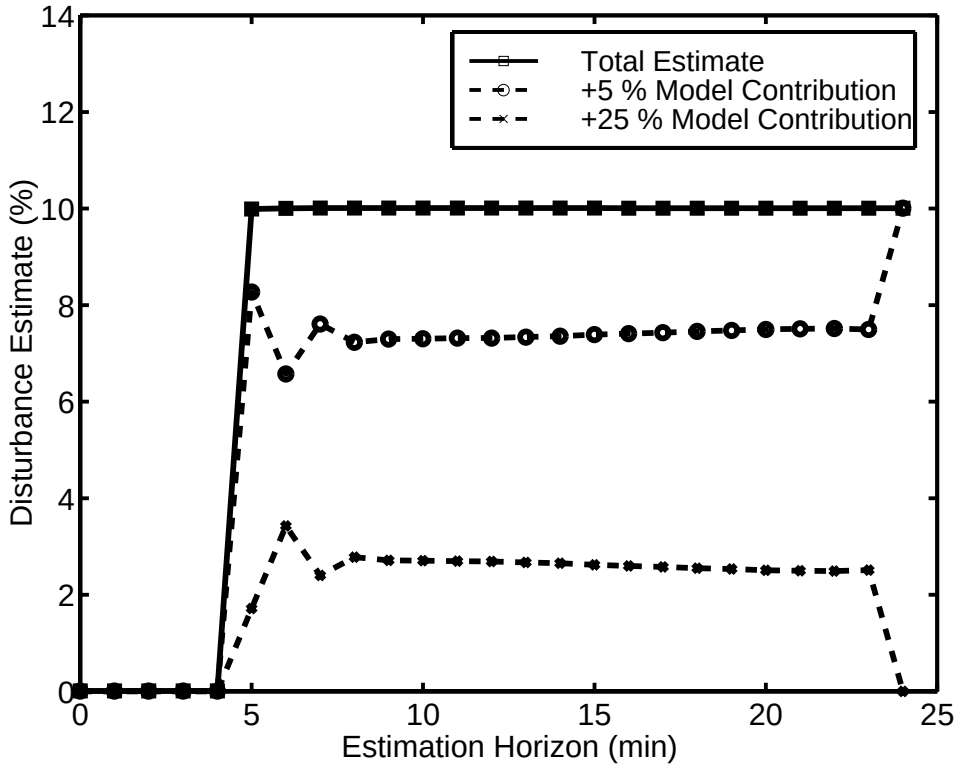

Figure 5: Estimates at time $t = 25$ across the estimation horizon using multiple models and measurements for 10% increase in feed flow at time $t = 5$ without process noise. Squares indicate the total estimate, circles show the contribution from the +5% model, and crosses indicate contribution from the +25% model.

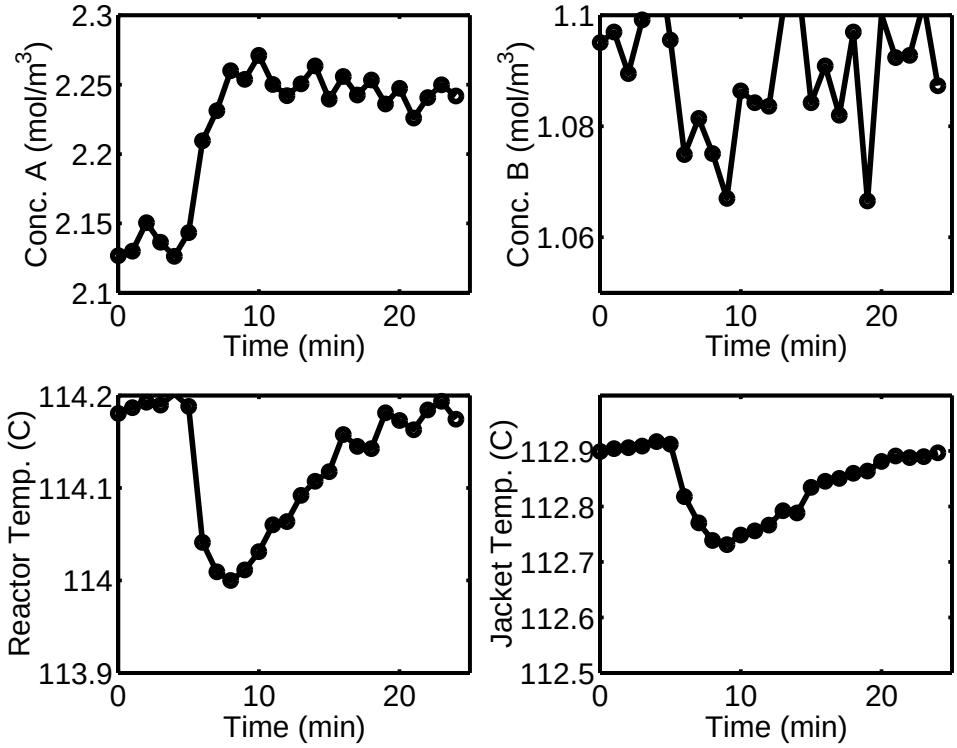

Figure 6: Measurements at time $t = 25$ for 10% increase in feed flow at time $t = 5$ with normally distributed noise ($\sigma^2 = 0.01$).

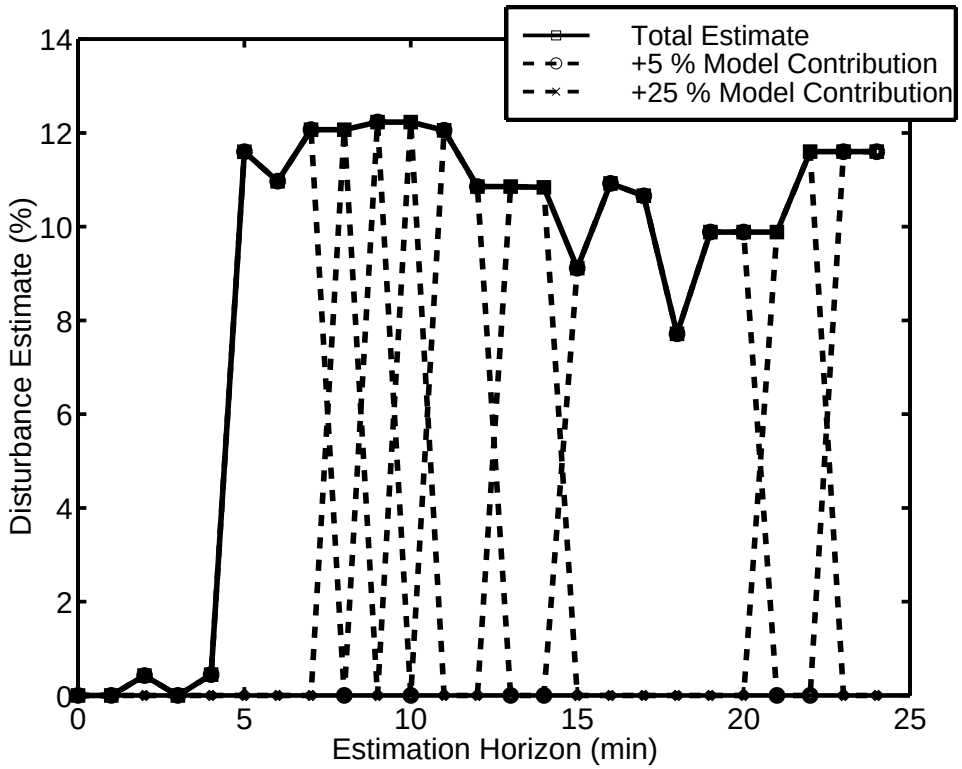

Figure 7: Estimates at time $t = 25$ across the estimation horizon using multiple models and measurements for 10% increase in feed flow at time $t = 5$ with normally distributed noise ($\sigma^2 = 0.01$). Squares indicate the total estimate, circles show the contribution from the +5% model, and crosses indicate contribution from the +25% model.

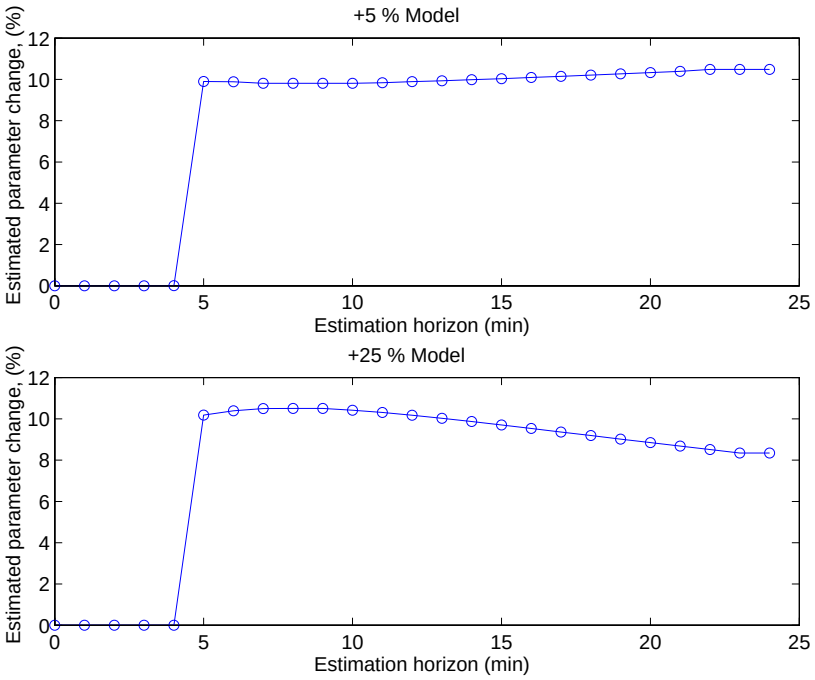

Figure 8: Separate estimates at time $t = 25$ using measurements from a 10% increase in feed flow at time $t = 5$ using single models in both cases.

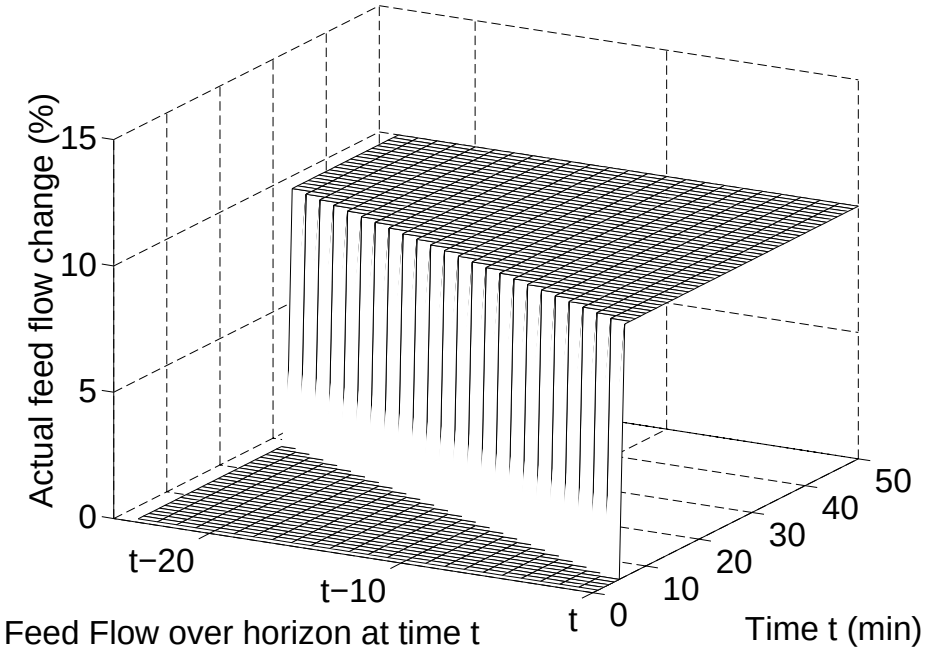

Figure 9: Actual parameter values over estimation horizon for 10% increase in feed flow at time $t = 5$.

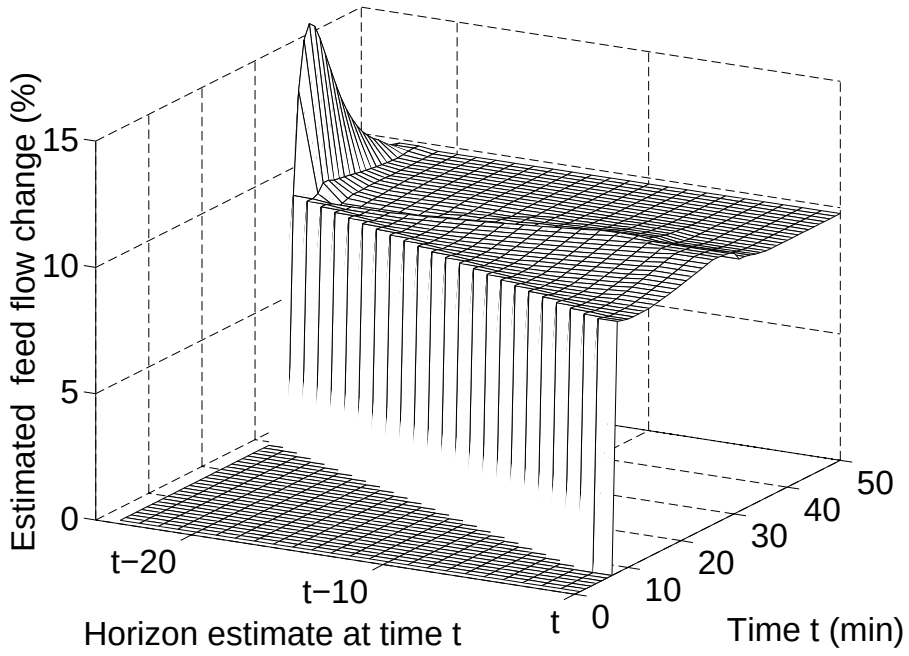

Figure 10: Horizon parameter estimates for 10% increase in feed flow at time $t = 5$ using multiple models with constraint allowing a single fault across estimation horizon.

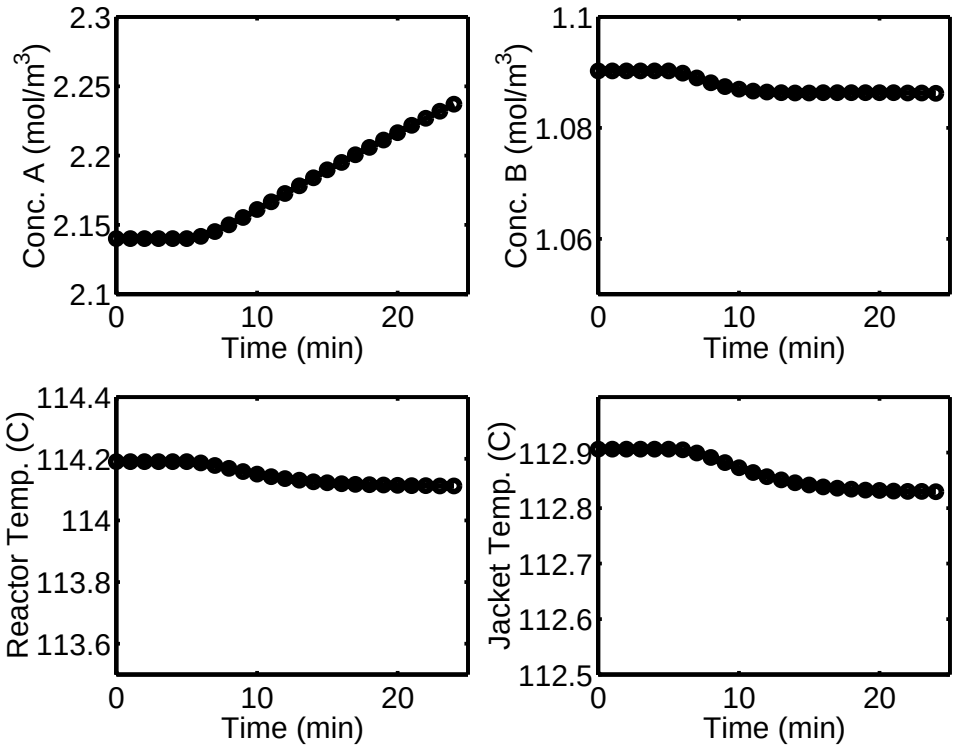

Figure 11: Measurements at time $t = 25$ for ramped change in in feed flow rate to +10% at time $t = 25$ starting at time $t = 5$ without process noise.

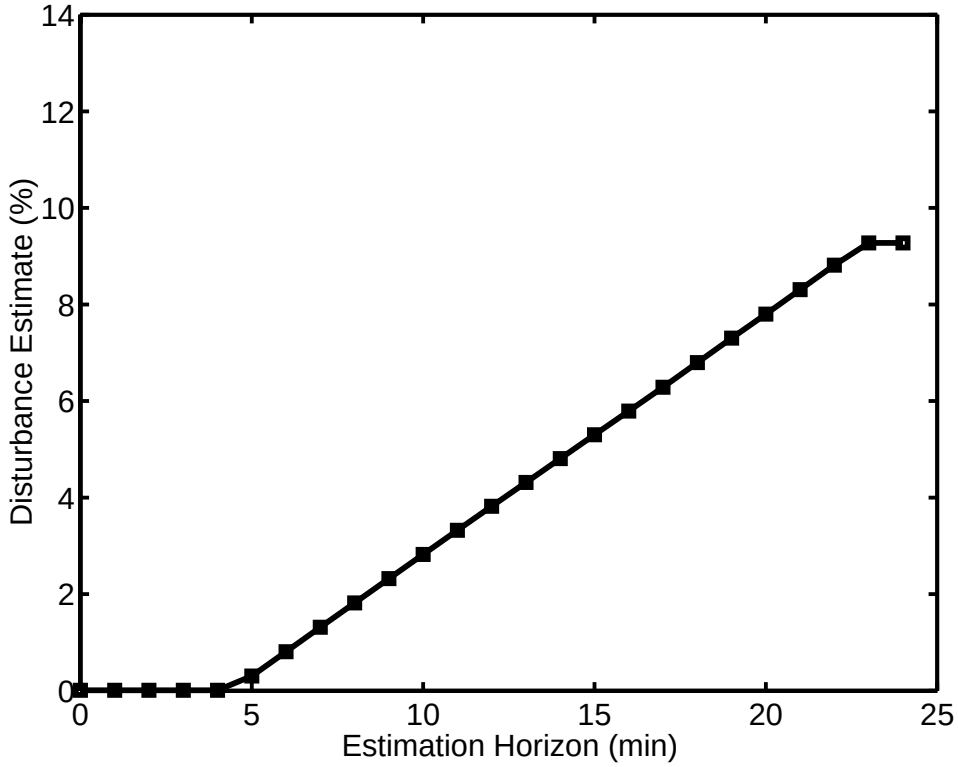

Figure 12: Estimates at time $t = 25$ using measurements for ramped change in in feed flow rate to +10% at time $t = 25$ starting at time $t = 5$ without process noise.

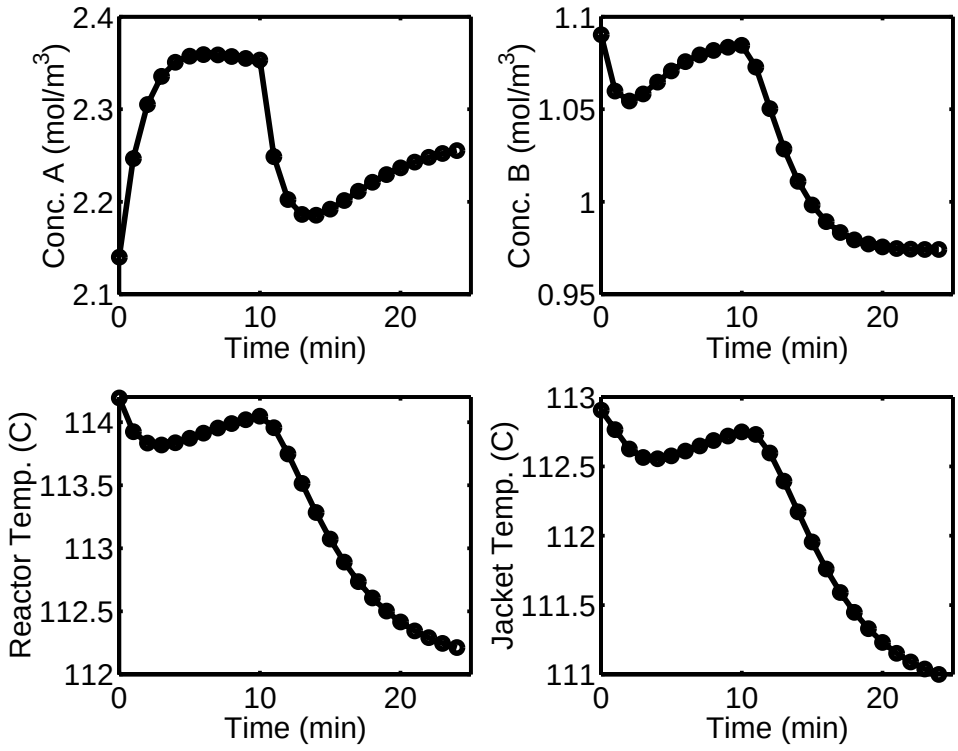

Figure 13: Measurements for dual fault case at time $t = 25$ with -20% input feed flow step at time $t = 0$ and -10% feed concentration step at time $t - 15$.

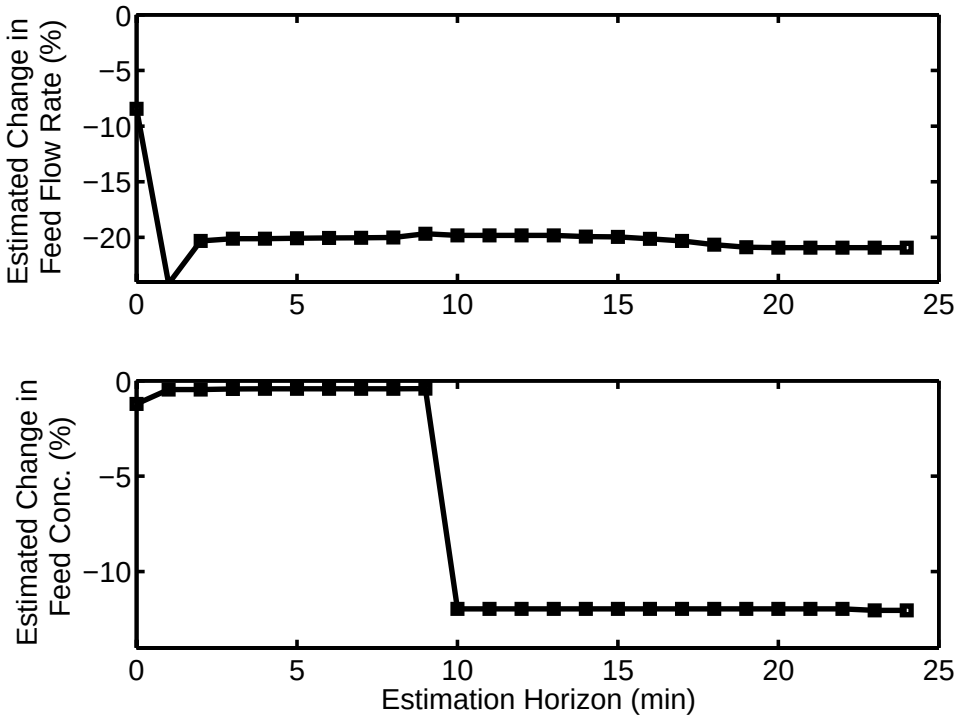

Figure 14: Estimates at time $t = 25$ using measurements for dual fault case with -20% input feed flow step at time $t = 0$ and -10% feed concentration step at time $t - 15$.

t	$y\,(\Delta d = 1)$	$y\,(\Delta d = 1.8)$	M_1	M_2	$y\,(\Delta d = 1.7)$
0	0	0	0	0	0
1	0.63	0.23	0.63	0.13	0.32
2	0.86	0.31	0.23	0.05	0.44
3	0.95	0.34	0.09	0.02	0.48
4	0.98	0.35	0.03	0.01	0.50
5	0.99	0.36	0.01	0	0.51
6	1	0.36	0	0	0.51

Table 2: Output response for example problem. Models M_1 and M_2 are developed from changing d from a value of 0 to values of 1 and 1.8, respectively.

Table 3: Van der Vusse CSTR model parameters.

$k_1o = 1.287 \cdot 10^{12}\, h^{-1}$	$k_2o = 1.287 \cdot 10^{12}\, h^{-1}$	$k_3o = 1.287 \cdot 10^{12}\, \frac{m^3}{mol\, A} h^{-1}$
$E_1 = -9758.3\, K$	$E_2 = -9758.3\, K$	$E_3 = -8560\, K$
$\Delta H_{RAB} = 4.2\, \frac{kJ}{mol\, A}$	$\Delta H_{RBC} = -11\, \frac{kJ}{mol\, B}$	$\Delta H_{RBD} = -41.85\, \frac{kJ}{mol\, A}$
$\rho = 0.9342\, \frac{kg}{l}$	$C_P = 3.01 \frac{kJ}{kg\, K}$	$C_{PK} = 2.0 \frac{kJ}{kg\, K}$
$k_w = 4032 \frac{kJ}{h\, m^2\, K}$	$A_R = 0.215\, m^2$	$m_k = 5.0\, kg$
$V_R = 10\, l$	$F_{KC} = 10.52 \frac{kg}{h}$	$v_{ko} = 60\, C$
	$\dot{V} = 14.10\, \frac{l}{h}$	

Table 4: Potential faults for CSTR case study.

f_1	increase in feed flow rate	f_2	decrease in feed flow rate
f_3	increase in coolant flow rate	f_4	decrease in coolant flow rate
f_5	increase in feed temperature	f_6	decrease in feed temperature
f_7	increase in coolant temperature	f_8	decrease in coolant temperature
f_9	increase in feed concentration	f_{10}	decrease in feed concentration
f_{11}	increase in jacket heat transfer coefficient	f_{12}	decrease in jacket heat transfer coefficient
f_{13}	increase in C_a measurement bias	f_{14}	decrease in C_a measurement bias
f_{15}	increase in C_b measurement bias	f_{16}	decrease in C_b measurement bias
f_{17}	increase in υ measurement bias	f_{18}	decrease in υ measurement bias
f_{19}	increase in υ_k measurement bias	f_{20}	decrease in υ_k measurement bias