Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2007, Article ID 80321, 20 pages
doi:10.1155/2007/80321

Research Article

Fault Detection and Control of Process Systems

Vu Trieu Minh, Nitin Afzulpurkar, and W. M. Wan Muhamad

Received 8 September 2006; Revised 25 December 2006; Accepted 5 February 2007

Recommended by José Manoel Balthazar

This paper develops a stochastic hybrid model-based control system that can determine online the optimal control actions, detect faults quickly in the control process, and reconfigure the controller accordingly using interacting multiple-model (IMM) estimator and generalized predictive control (GPC) algorithm. A fault detection and control system consists of two main parts: the first is the fault detector and the second is the controller reconfiguration. This work deals with three main challenging issues: design of fault model set, estimation of stochastic hybrid multiple models, and stochastic model predictive control of hybrid multiple models. For the first issue, we propose a simple scheme for designing faults for discrete and continuous random variables. For the second issue, we consider and select a fast and reliable fault detection system applied to the stochastic hybrid system. Finally, we develop a stochastic GPC algorithm for hybrid multiple-models controller reconfiguration with soft switching signals based on weighted probabilities. Simulations for the proposed system are illustrated and analyzed.

1. Introduction

Control systems are becoming more and more powerful and sophisticated. Reliability, availability, and safety are primary goals in the operation of process systems. The aim to develop a fast and reliable control system that could detect undesirable changes in the process (referred to as "faults") and isolate the impact of faults has been attracting much attention of researchers. Various methods for fault detection and control of process systems have been studied and developed over recent years [1–8] but there are relatively few successful developments of controller systems that can deal with faults in stochastic

hybrid sense where faults are modeled as multiple-model set with varying variable structure and use of stochastic model predictive control algorithm.

Faults are difficult to foresee and prevent. Traditionally, faults were handled by describing the result behavior of the system and were grouped into a hierarchical structure of fault model [9]. This approach is still widely used in practice: when a failure occurs, the system behavior changes and should be described by a different mode from the one that corresponds to the normal mode. A more appropriate mathematical model for such a system is the so-called stochastic hybrid approach. It differs from the conventional hierarchical structure in that its state may jump as well as it may vary continuously. Apart from the applications to problems involving failures, hybrid systems have found great success in such areas as target tracking and control that involve possible structure changes [10]. Hybrid systems switch among many operating modes, where each mode is governed by its own characteristic dynamic laws. Mode transitions are triggered by variables crossing specific thresholds.

For the fault modeling and verification, design of a model set is the key issue for the application of multiple-model estimator and controller. For simple systems, fault model set can be designed as a fixed structure (FS) or a fixed set of models at all times. The FS has fundamental limitations that only one fixed model set can be represented sufficiently and accurately by all possible failures. Actually, the set of possible system modes is not fixed. It varies and depends on the hybrid state of the system at the previous time. As shown in [11], use of more models in an FS estimation does not improve the performance. In fact, the performance will deteriorate if too many models are used in one fixed model set. Therefore, a variable structure (VS) is considered for modeling faults. The VS overcomes limitations of FS by using a variable set of models determined in real time adaptively. General and representative problems of model-set design, choice, and comparison for multiple-model approach to hybrid estimation are given in [12]. In this paper, a simple scheme for modeling of fault set as discrete and continuous random variables is proposed.

For the fault detection, various methods have been developed in [13–17]. One of the most effective approaches for solving stochastic hybrid systems is based on the use of multiple models (MM): it runs a bank of filters in parallel, each based on a particular model to obtain the model-conditional estimates. MM estimation algorithms appeared in early 1970s when Bar-Shalom and Tse [18] introduced a suboptimal, computationally bounded extension of Kalman filter to cases where measurements were not always available. Then, several multiple-model filtering techniques, which could provide accurate state estimation, have been developed. Major existing approaches for MM estimation are discussed and introduced in [18–23] including the noninteracting multiple model (NIMM), the Gaussian pseudo-Bayesian (GPB1), the second-order Gaussian pseudo-Bayesian (GPB2), and the interacting multiple models (IMM). Among those, we consider and select a fast and reliable algorithm for the fault detection system applied to the above model-set design.

Finally, for the controller reconfiguration (CR), we propose the use of stochastic model predictive control (MPC) algorithm applied to stochastic hybrid multiple models. The problem of determining optimal control laws for hybrid systems has been widely investigated and many results can be found in [24–28]. However, the use of MPC applied to

stochastic hybrid systems is unfavorable since the general MPC algorithms follow deterministic perspective approaches. Thus, we propose use of generalized predictive control (GPC), a stochastic MPC technique developed by Clarke el al. [29, 30]. GPC uses the ideas with controlled autoregressive integrated moving average (CARIMA) plant in adaptive context and self-tuning by recursive estimation. Kinnaert [31] developed GPC from CARIMA model into a more general one in MIMO state-space form. We propose the use of a bank of GPC controllers, each based on a particular model. The optimal control action is the summation of probabilistic weighted outputs of each GPC controller. A similar soft switching mechanism based on weighted probabilities has been developed. Simulations for the proposed controller are illustrated and analyzed. Results show its strong ability for real applications to detect faults in dynamic systems.

The outline of this paper is as follows: Section 2 introduces the stochastic hybrid system and fault modeling design; Section 3 considers the selection of a fault detection system; Section 4 develops a controller reconfiguration integrated with fault detection system; examples and simulations are given after each section to illustrate the main ideas of the section; finally conclusions are given in Section 5.

2. Hybrid system and fault modeling design

An important requirement currently exists for improving the safety and reliability of process systems in ways that reduce their vulnerability to failures. When a failure occurs, the system behavior changes and should be described by a different mode from the one that corresponds to the normal mode of operation. An effective way to model faults for dynamic failures, which state may jump as well as vary continuously in a discrete set of modes, is the so-called a hybrid system.

A simplest continuous time hybrid system is described by the following different linear state update equation:

$$\begin{aligned} \dot{x}(t) &= A\big(t,m(t)\big)x(t) + B\big(t,m(t)\big)u(t) + T\big(t,m(t)\big)\xi(t), \\ z(t) &= C\big(t,m(t)\big)x(t) + \eta\big(t,m(t)\big), \end{aligned} \tag{2.1}$$

and a discrete-time hybrid system is the following:

$$\begin{aligned} x(k+1) &= A\big(k,m(k+1)\big)x(k) + B\big(k,m(k+1)\big)u(k) + T\big(k,m(k+1)\big)\xi(k), \\ z(k) &= C\big(k,m(k)\big)x(k) + \eta\big(k,m(k)\big), \end{aligned} \tag{2.2}$$

where A, B, T, and C are the system matrices, $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the control input, $z \in \mathbb{R}^p$ is the measured output, $\xi \in \mathbb{R}^{n_\xi}$ and $\eta \in \mathbb{R}^{p_\eta}$ are independent noises with means $\overline{\xi}(t)$ and $\overline{\eta}(t)$, and covariances $\Theta(k)$ and $\Xi(k)$. In this equation, $m(t) \in M$ is the system mode, which may jump or stay unchanged, x is the state variable, which varies continuously. The system mode sequence is assumed to be a first-order Markov chain with transition probabilities:

$$\Pi\{m_j(k+1) \mid m_i(k)\} = \pi_{ij}(k), \quad \forall m_i, m_j \in M_k, \tag{2.3}$$

where $\Pi\{\cdot\}$ denotes probability, $m(k)$ is the discrete-valued modal state, that is, the index of the normal or fault modes at time k, $M_k = \{m_1,\ldots,m_N\}$ is the set of all possible system modes at time instant k, $\pi_{ij}(k)$ is the transition probability from mode m_i to mode m_j at time instant k. Obviously, the following relation must be held for any $m_i \in M_k$:

$$\sum_{j=1}^{M_k} \pi_{ij}(k) = \sum_{j=1}^{M_k} \Pi\{m_j(k+1) \mid m_i(k)\} = 1, \quad m_{i,\ldots,N} \in M_k \subset M. \tag{2.4}$$

Faults can be modeled by changing the appropriate matrices A, B, C, or T representing the effectiveness of failures in the systems. They can also be modeled by increasing the process noise covariance Θ or measurement noise covariance Ξ in ξ and η. M_k denotes the set of models used at time instant k and M denotes the total set of models used, that is, M is the union of all $M_k's$:

$$m_i \in M_k = \begin{cases} x(k+1) = A_i(k)x(k) + B_i(k)u(k) + T_i(k)\xi_i(k), \\ z(k) = C_i(k)x(k) + \eta_i(k), \end{cases} \tag{2.5}$$

where the subscript i denotes the fault modeling in model set, $m_i \in M_k = \{m_1, m_2, \ldots, m_N\}$, each m_i corresponds to a node (a fault) occurring in the process at time instant k. In fixed structure, the model set M_k used is fixed over time, that is, $M_k \overline{\Delta} = M$, for all k, to be determined offline based on the initial information about the system faults. Otherwise, we have a variable structure or the model set M_k varies at any time in the total model set M or $M_k \subset M$. Variable structure model overcomes fundamental limitations of fixed structure mode set because the fixed model set used does not always exactly match the true system mode set at any time, or the set of possible modes at any time varies and depends on the previous state of the system.

For faults varying as continuous variables, we can handle them via probabilistic modeling techniques. In these cases, faults can be modeled as discrete modes based on their cumulative distribution function (CDF) and probability density function (PDF). Data of the past operation fault records (fault rate and percentage of fault type) provide the required probability distribution of the mode. More methods on design of model set as continuous random variables can be read in [12]. In this paper, we just propose the simplest method of equal probability to model a continuous random variable into discrete modes. We assume that the CDF $F_s(x)$ of the true continuous variable s is known and we want to reconstruct it into the CDF $F_m(x)$ of discrete modes. In the equal probability method, we propose to group the CDF $F_s(x)$ into $|M|$ discrete modes of equal probabilities, $m_i = 1/|M|$ (preferably an odd number $3,5,7,\ldots$, for symmetric distributions). The design of a continuous variable into discrete modes is shown in Figure 2.1 with $|M| = 5$ and PDF is a normal distribution $f(x;\mu,\sigma) = (1/\sigma\sqrt{2\pi})\exp(-(x-\mu)^2/2\sigma^2)$ with mean $\mu = 0$ and variance $\sigma^2 = 1$.

In Figure 2.1, we group a continuous random variable with a normal distribution into five equal probabilities (discrete modes) with a model set (mode set) of $M = \{m_1, m_2, \ldots, m_{|M|}\}$.

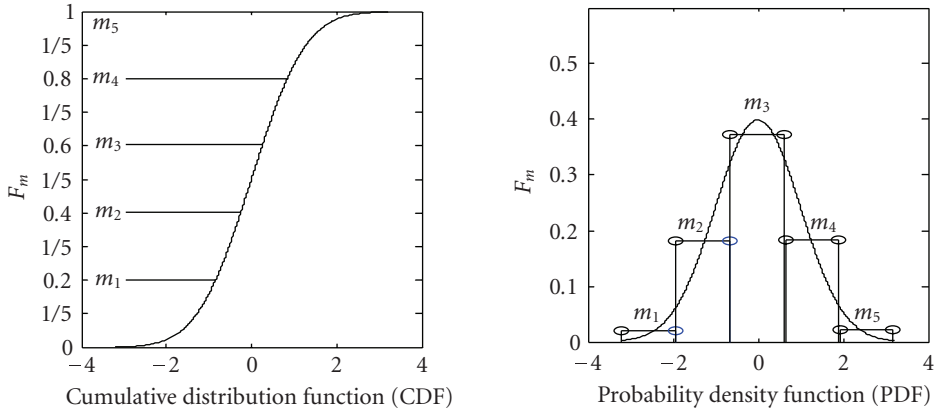


Figure 2.1. Group a normal CDF into equal probabilities.

Example 2.1 (fault model-set design). Consider a continuous process system with the state space model in (2.1):

$$\begin{aligned} \dot{x}(t) &= A_i x(t) + B_i u(t) + T_i \xi_i(t), \\ z(t) &= C_i x(t) + \eta_i(t), \end{aligned} \tag{2.6}$$

where A_i, B_i, T_i, and C_i are system matrices, ξ_i and η_i are independent noises with zero-mean $\overline{\xi} = \overline{\eta} = 0$ and constant covariance $\Theta = \Xi = 0.02^2 I$, $T_i = I$. We assume that at the normal operation mode N, two generic types of faults might take place: one static fault mode S_0 and one varying fault mode V_0. This example is modeled from a chemical process model with four state variables, two inputs and two outputs. For simplicity, we verify only one input.

We have the normal operation mode:

$$A_N = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -0.08 & 0.06 & 0.7 & 0 \\ 0.1 & -0.1 & 0 & 0 \end{bmatrix}, \qquad B_N = \begin{bmatrix} -0.2 \\ 0.03 \\ 2 \\ 1 \end{bmatrix}, \qquad C_N = \begin{bmatrix} 1 & -0.5 & 1 & 1 \\ -1 & 0.6 & 0 & 1 \end{bmatrix}. \tag{2.7}$$

A static failure mode S_0 happens when an interrupted actuator failure, -50%,

$$B_{S_0} = 0.5 B_N = \begin{bmatrix} -0.1 \\ 0.015 \\ 1 \\ 0.5 \end{bmatrix}. \tag{2.8}$$

A varying failure mode V_0 happens when a continuous varying variable appears in A_N,

$$A_{V_0} = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -0.08 & 0.06 & 0.7 & \sin(\omega) \\ 0.1 & -0.1 & 0 & 0 \end{bmatrix}, \tag{2.9}$$

where ω is a continuous varying variable (deg/s).

We assume that at the static mode S_0, two other generic types of static faults might take place: mode S_1 with sensor 1 failure -50% or

$$C_{S_1} = \begin{bmatrix} 0.5z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0.5 & -0.25 & 0.5 & 0.5 \\ -1 & 0.6 & 0 & 1 \end{bmatrix}, \tag{2.10}$$

and mode S_2 with sensor 1 failure $+50\%$ or

$$C_{S_2} = \begin{bmatrix} 1.5z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 1.5 & -0.75 & 1.5 & 1.5 \\ -1 & 0.6 & 0 & 1 \end{bmatrix}. \tag{2.11}$$

We continue to assume that the PDF of the continuous varying variable ω in matrix A_{V_0} is the mixture of three normal distributions:

$$f(\omega) = \frac{1}{3\sqrt{2\pi}} \exp\left(-\frac{(\omega-3)^2}{2}\right) + \frac{1}{3\sqrt{2\pi}} \exp\left(-\frac{(\omega)^2}{2}\right) + \frac{1}{3\sqrt{2\pi}} \exp\left(-\frac{(\omega+3)^2}{2}\right). \tag{2.12}$$

Since the PDF of ω is the combination of three normal curves with three mean values $\overline{\omega}_1 = -3^0/\text{s}$, $\overline{\omega}_0 = 0^0/\text{s}$, and $\overline{\omega}_2 = 3^0/\text{s}$, we can group this continuous varying variable into three discrete models (modes) with A_{V_1}, A_{V_0}, and A_{V_2} corresponding to the above three mean values with equal probabilities of 1/3, 1/3, and 1/3. The model set design via CDF and its reconstruction PDF are shown in Figure 2.2.

Hence, in this example, we have total 7 models (modes) grouped into three varying model sets in Figure 2.3:

$$\begin{aligned} &\text{model set 1: } M_1 = \{m_1 = N_{(A_N,B_N,C_N)}, m_2 = S_{0(A_S,B_N,C_N)}, m_3 = V_{0(A_{V_0},B_N,C_N)}\}, \\ &\text{model set 2: } M_2 = \{m_2 = S_{0(A_S,B_N,C_N)}, m_4 = S_{1(A_S,B_N,C_{S_1})}, m_5 = S_{2(A_S,B_N,C_{S_2})}\}, \\ &\text{model set 3: } M_3 = \{m_3 = V_{0(A_{V_0},B_N,C_N)}, m_6 = V_{1(A_{V_1},B_N,C_N)}, m_7 = V_{2(A_{V_2},B_N,C_N)}\}. \end{aligned} \tag{2.13}$$

We assume that the following Markov transition probability matrix in (2.3) is used for all simulations in the total model set $M = \{m_1, m_2, m_3, m_4, m_5, m_6, m_7\}$:

$$\Pi = \begin{bmatrix} 0.94 & 0.03 & 0.03 & 0 & 0 & 0 & 0 \\ 0.03 & 0.93 & 0 & 0.02 & 0.02 & 0 & 0 \\ 0.03 & 0 & 0.93 & 0 & 0 & 0.02 & 0.02 \\ 0 & 0.05 & 0 & 0.95 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 & 0.95 & 0 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0.95 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0 & 0.95 \end{bmatrix}. \tag{2.14}$$

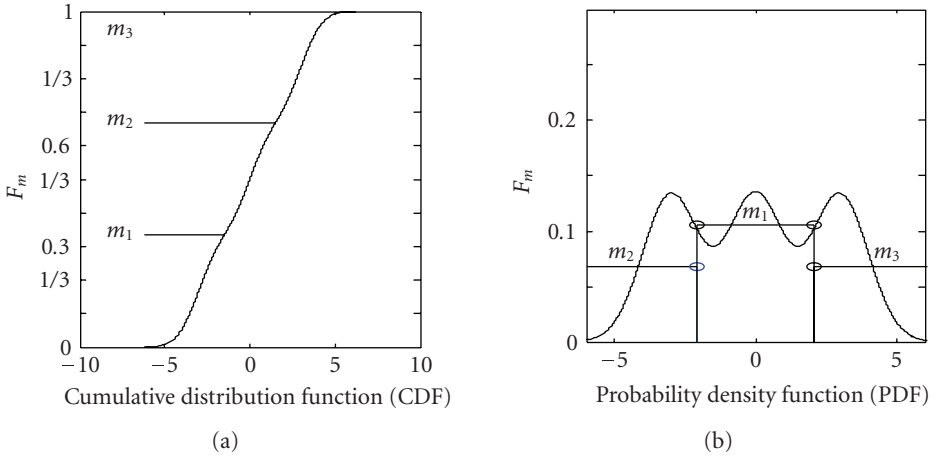


Figure 2.2. Model set design of varying variable ω.

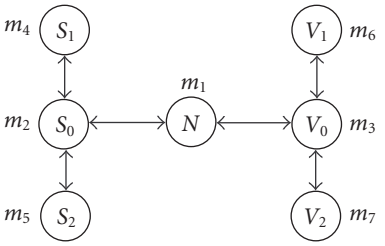


Figure 2.3. Total model set design.

The design of model set now is completed. In the next section, we will consider the selection of a reliable fault detection system applied to this model set.

3. Fault detection system

Fault detection for stochastic hybrid systems has received a great attention in recent years. A variety of different fault detection methods has been developed. For hybrid systems with fixed structure (FS) or variable structure (VS) modeled in mixed logic dynamical (MLD) form or piecewise affine (PWA) systems, the state estimation can be solved by moving horizon estimation (MHE) strategy. MHE has strong ability to incorporate constraints on states and disturbances. Moreover, on the computational side, because MHE algorithms lead to optimization problem of fixed dimension, they are suitable for practical implementation. MHE is applied successfully to constrained linear systems where it can guarantee stability of the estimate when other classical techniques, like Kalman filtering, fail [32]. A number of MHE techniques for fault detection system can be found in [32–35]. However for stochastic hybrid systems where their state can jump as well

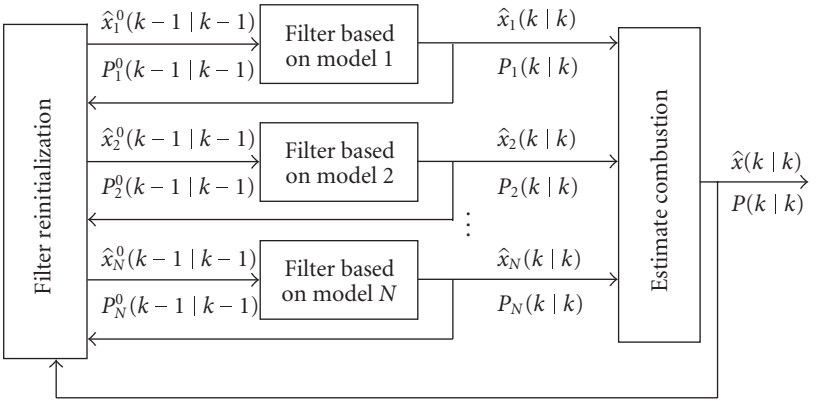


Figure 3.1. Structure of an MM estimator.

as vary continuously and randomly in a model set with the system mode sequence assumed to be a first-order Markov chain in (2.3), a more effective and natural estimation approach is the use of algorithms of multiple-model (MM) estimator. Major existing approaches for MM estimation are discussed and introduced in [18–26]. In this part, we consider and select a reliable fault detection system among the noninteracting multiple models (NIMM), the Gaussian pseudo-Bayesian (GPB1), the second-order Gaussian pseudo-Bayesian (GPB2), and the interaction multiple models (IMM).

From the design of model set (in Section 2), a bank of filters runs in parallel at every time, each based on a particular model, to obtain the model-conditional estimates. The overall state estimate is a probabilistically weighted sum of these model-conditional estimates. The jumps in system modes can be modeled as switching among the assumed models in the set.

Figure 3.1 shows the operation of a recursive multiple-model estimator, where $\hat{x}_i(k \mid k)$ is the estimate of the state $x(k)$ obtained from the filter based on model m_i at time k given the measurement sequence through time k; $\hat{x}_i^0(k-1 \mid k-1)$ is the equivalent reinitialized estimate at time $(k-1)$ as the input to the filter based on model m_i at time k; $\hat{x}(k \mid k)$ is the overall state estimate; $P_i(k \mid k)$, $P_i^0(k-1 \mid k-1)$, and $P(k \mid k)$ are the corresponding covariances.

A simple and straightforward way of filter reinitialization is that each single model-based recursive filter uses its own previous state estimation and state covariance as the input at the current cycle:

$$\begin{aligned} \hat{x}_i^0(k-1 \mid k-1) &= \hat{x}_i(k-1 \mid k-1), \\ P_i^0(k-1 \mid k-1) &= P_i(k-1 \mid k-1). \end{aligned} \tag{3.1}$$

This leads to the so-called noninteracting multiple-model (NIMM) estimator because the filters operate in parallel without interactions with one another, which is reasonable only under the assumption that the system mode does not change.

Another way of reinitialization is to use the previous overstate estimate and covariance for each filter as the required input:

$$\begin{aligned} \hat{x}_i^0(k-1 \mid k-1) &= \hat{x}(k-1 \mid k-1), \\ P_i^0(k-1 \mid k-1) &= P(k-1 \mid k-1). \end{aligned} \tag{3.2}$$

This leads to the first-order generalized pseudo-Bayensian (GPB1) estimator. It belongs to the class of interacting multiple-model estimators since it uses the previous overall state estimate, which carries information from all filters. Clearly, if the transition probability matrix is an identity matrix, this method of reinitialization reduces to the first one.

The GPB1 and GPB2 algorithms were the result of early work by Ackerson and Fu [21] and good overviews are provided in [22], where suboptimal hypothesis pruning techniques are compared. The GPB2 differed from the GPB1 by including knowledge of the previous time step's possible mode transitions, as modeled by a Markov chain. Thus, GPB2 produced slightly smaller tracking errors than GPB1 during nonmaneuvering motion. However in the size of this part, we do not include GPB2 into our simulation test and comparison.

A significantly better way of reinitialization is to use IMM. The IMM was introduced by Zhang and Li in [23]:

$$\begin{aligned} \hat{x}_j^0(k \mid k) &= E\left[x(k) \mid z^k, m_j(k+1)\right] = \sum_{i=1}^{N} \hat{x}_i(k:k) P\{m_i(k) \mid z^k, m_j(k+1)\}, \\ P_j^0(k \mid k) &= \operatorname{cov}\left[\hat{x}_j^0(k:k)\right] = \sum_{i=1}^{N} P\{m_i(k) \mid z^k, m_j(k+1)\} \\ &\qquad \times \{P_i(k \mid k) + \tilde{x}_{ij}^0(k \mid k)\tilde{x}_{ij}^0(k \mid k)'\}, \end{aligned} \tag{3.3}$$

where $\operatorname{cov}[\cdot]$ stands for covariance and $\tilde{x}_{ij}^0(k \mid k) = \hat{x}_i^0(k \mid k) - \hat{x}_j(k \mid k)$. In this paper, we will use this approach for setting up a fault detection system.

For each model in $M_k \in M = \{m_1, \ldots, m_N\}$, we can operate a Kalman filter. The probability of each model matching to the system mode provides the required information for mode's chosen decision. The mode decision can be achieved by comparing it with a fixed threshold probability μ_T. If the mode probabilities $\max_i(\mu_i(k)) \geq \mu_T$, mode at $\mu_i(k)$ has occurred and has taken place at the next cycle. Otherwise, there is no new mode detection. The system maintains the current mode for the next cycle calculation.

Example 3.1 (test and selection of fault detection system). From the model-set design in Example 2.1, model-set modes in (2.6) are discretized with 0.1 second, the threshold value for mode probabilities is chosen as $\mu_T = 0.9$. Now we begin to compare the three estimators of NIMM, GPB1, and IMM to test their ability to detect faults. The seven models are run for a time interval $t = 20$ seconds and for the following sequence: $\{m_1, m_2, m_4, m_2, m_5, m_2, m_1, m_3, m_6, m_3, m_7\}$. Results of simulation are shown in Figure 3.2.

In Figure 3.2, we can see that the GPB1 estimator performs as good as IMM estimator while NIMM estimator fails to detect faults in the model set. Next we continue to test the

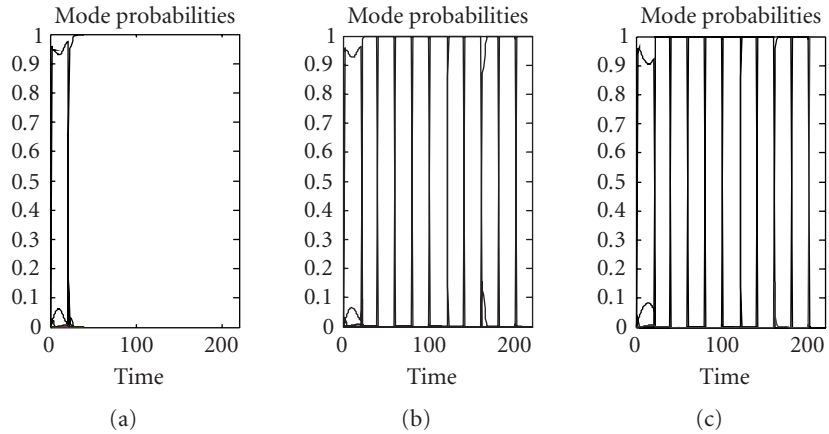


Figure 3.2. Probabilities of estimators (a) NIMM, (b) GPB1, and (c) IMM.

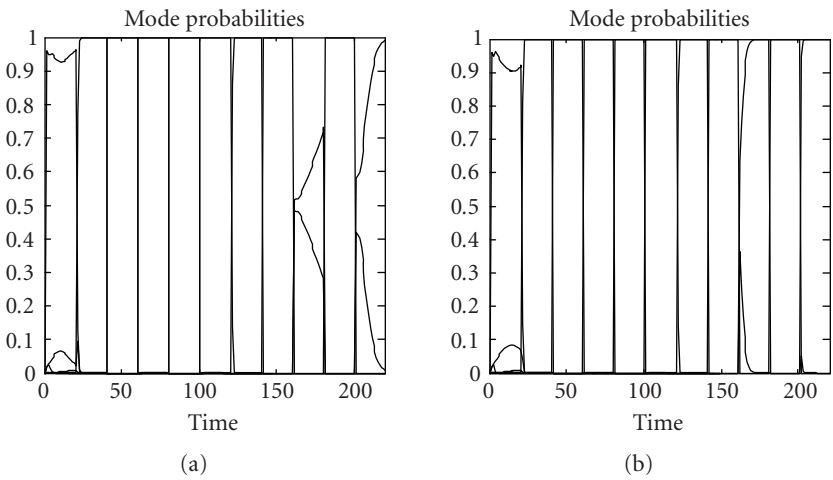


Figure 3.3. Probabilities of estimators (a) GPB1 and (b) IMM.

ability of GPB1 and IMM estimators by narrowing the distances between modes as close as possible until one of methods cannot detect the failures. Now we assume to design new two varying modes of $\{m_6^*, m_7^*\}$ corresponding to a new $A_{V_1}^*$ with $\overline{\omega}_1^* = 0.3^0/\mathrm{s}$ and a new $A_{V_2}^*$ with $\overline{\omega}_2^* = -0.3^0/\mathrm{s}$. With these new parameters, GPB1 fails to detect failures since the distance between modes $\{m_6^*, m_3, m_7^*\}$ is very close, while IMM still proves it is much superior in Figure 3.3.

As a result, we select the IMM for our fault detection system. Now we move to the main part of this paper to set up a controller reconfiguration for the fault detection and control system.

4. Controller reconfiguration

In this section, we develop a new CR which can determine online the optimal control actions and reconfigure the controller accordingly. The problem of determining the optimal control laws for hybrid systems has been widely studied in recent years and many methods have been developed in [24–28]. Optimal quadratic control of piecewise linear and hybrid systems is found in [25, 26]. For complex constrained multivariable control problems, model predictive control (MPC) has become the accepted standard in the process industries [36, 37]. MPC can be applied to multiple models using linear matrix inequalities (LMIs) in [38]. The general MPC algorithms follow deterministic perspective approaches, hence, for stochastic hybrid systems described in (2.1), (2.2), (2.3), and (2.4), there are few MPC ideas applied to control stochastic hybrid systems. Thus, we propose a new controller reconfiguration (CR) using generalized predictive control (GPC) algorithm. We will show how an IMM-based GMC controller can be used as a good fault detection and control system.

Generalized predictive control (GPC) is one of model predictive control (MPC) techniques developed by Clarke et al. [29, 30]. GPC was intended to offer a new adaptive control alternative. GPC uses the ideas with controlled autoregressive integrated moving average (CARIMA) plant in adaptive context and self-tuning by recursive estimation. Kinnaert [31] developed GPC from CARIMA model into a more general form when the models are described in space.

The optimal control problem for the general cost function for GPC controller in (2.1) is

$$\begin{gathered}\min_{U\triangleq\{u_1,u_2,\ldots,u_{t+N_u-1}\}}\left\{J(U,x(t))=x'_{t+N_{y|t}}Px_{t+N_{y|t}}+\sum_{k=0}^{N_y-1}\left[x'_{t+k|t}Qx_{t+k|t}+u'_{t+k|t}Ru_{t+k|t}\right]\right\},\\ \text{subject to } x_{t+k+1|t}=Ax_{t+k|t}+Bu_{t+k}+T\xi_{t+k|t},\\ u_{t+k}=-Kx_{t+k|t},\quad k\geq N_u,\\ x_{t+k}\in\mathbb{X},\quad u_{t+k}\in\mathbb{U},\end{gathered}\tag{4.1}$$

where $Q=Q'>0$, $R=R'\geq 0$ are the weighting matrices for predicted state and input, respectively. Linear feedback gain K and the Lyapunov matrix $P>0$ are the solution of Riccati equation. For simplicity, we assume that the predictive horizon is set equal to the control horizon, that is, $N_u=N_y=N_P$.

By substituting $x_{t+N_p|t}=A^{N_p}x(t)+\sum_{j=0}^{N_p-1}A^jBu_{t+N_p-1-j}+A^{N_p-1}T\xi(t)$, (4.1) can be rewritten as

$$\min_U\left\{\frac{1}{2}U'HU+x'(t)FU+\xi'(t)YU\right\},\quad \text{subject to } GU\leq W+Ex(t),\tag{4.2}$$

where the column vector $U\triangleq[u'_t,\ldots,u'_{t+N_p-1}]'\in\mathbb{R}^U$ is the predictive optimization vector, $H=H'>0$, and H, F, Y, G, W, E are obtained from (4.1) as only the optimizer

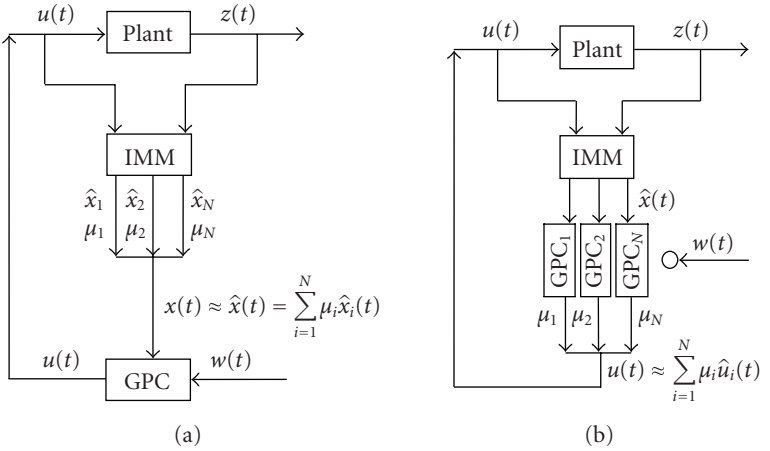


Figure 4.1. Two diagrams of IMM-based GPC controllers.

U is needed. Then, the optimization problem (4.2) is a quadratic program and depends on the current state $x(t)$ and noise $\xi(t)$. The implementation of GPC requires the online solution of a quadratic program at each time step.

For the controller reconfiguration (CR), we can use the output of IMM, the overall state estimate $x(k) \approx \hat{x}(k) = \sum_{i=1}^{N} \mu_i \hat{x}_i(k)$ in (3.3), where N is the number of models in the current model set, as the input for (a) a GPC controller or for (b) a bank of GPC controllers shown in Figure 4.1.

In Figure 4.1(a), IMM provides the overall state estimate $\hat{x}(t)$ and indicates one "most reliable" mode m_i in the mode set. Thus, we can build up a GPC controller corresponding to this "most reliable" mode. The stability of the system is assured if we can find a positive Lyapunov matrix in (4.1).

In Figure 4.1(b), assuming the model probabilities $\mu_i(t)$ are constant during the predictive control horizon, we can derive a new GPC control law that the "true" model $\overline{m}$ is the union of all model modes m_i in the current model set M_k in (2.5) or $\overline{m} = \sum_{i=1}^{N} \mu_i m_i$. Thus, we can achieve a new control law by using a bank of GPC controllers for each model in the model set and have the overall control input $u(t) \approx \sum_{i=1}^{N} \mu_i \hat{u}_i(t)$. The stability of the system is guaranteed if we can find out a common Lyapunov matrix P for all models in the model set.

Lemma 4.1. *The optimal control problem for the general cost function for GPC controller in (4.1) applied to control stochastic hybrid system in (2.1) can guarantee the global and asymptotical stability if there exist positive definite matrices P and θ_i such that $A_i P + P A_i' = -\theta_i$, for all i.*

Proof. For simplicity, we assume that the control input $u(t + N_P) = 0$ after $k \geq N_P$ predictive control horizon so that a common Lyapunov matrix for each model in (4.1) is the solution of Riccati equations $A_i P + P A_i' = -\theta_i$ since the state update equation then becomes $\dot{x}(t) = \sum_{i=1}^{N} \mu_i A_i x(t)$. For a positive Lyapunov function $V(x) = x'(t) P x(t)$, we have

always a negative definite time derivative $\dot{V}(x) < 0$, and the system is stable:

$$\dot{V}(x) = \left(\sum_{i=1}^{N} \mu_i A_i x\right)' Px + x'P\left(\sum_{i=1}^{N} \mu_i A_i x\right) = \sum_{i=1}^{N} \mu_i x'(A_i P + PA_i')x = \sum_{i=1}^{N} \mu_i x'(-\theta_i)x < 0. \tag{4.3}$$

Otherwise, the closed-loop feedback in (4.1) $u_{t+k} = -Kx_{t+k|t}$ for $k \geq N_P$, and we have $\dot{x}(t) = Ax(t) + Bu(t)$ or $\dot{x}(t) = (\sum_{i=1}^{N} \mu_i(t)(A - BK_i))x(t) = (\sum_{i=1}^{N} \mu_i(t)A_{Li})x(t)$ can also satisfy Lemma 4.1 in (4.3). A similar result was found in [38] when we can apply a common Lyapunov matrix to find a robust stabilizing state feedback for uncertain hybrid systems using LMIs.

For the controller reconfiguration (CR), we can apply hard switching or soft switching. For hard switching, we use only one controller implemented at any time—similar scheme in Figure 4.1(a). As indicated in [6], even if each controller globally stabilizes, there can exist a switching sequence that destabilizes the closed-loop dynamics. Now we consider some possible soft switching signals where the outputs of each controller are weighted by a continuous, time-varying, probability vector $\nu_i(t)$ which can guarantee the closed-loop stability, $u(t) = \sum_{i=1}^{N} \nu_i \hat{u}_i(t)$, in which $\sum_{i=1}^{N} \nu_i(t) = 1$, $\nu_i(t) \in [0,1]$ for all i,t.

It is difficult to find out a common Lyapunov matrix for all models in the model set (4.3). Recently, a new type of parameter-dependent Lyapunov function has been introduced in the form that $P_L = \sum_{i=1}^{N} \nu_i P_i$ is a parameter-dependent Lyapunov function for any $A_L = \sum_{i=1}^{N} \nu_i A_{Li}$. That is true since we always have a negative derivative $\dot{V}(x) < 0$ in (4.3) as $\dot{V}(x) = \sum_{i=1}^{N} \nu_i x'(A_i P_i + P_i A_i')x = \sum_{i=1}^{N} \nu_i x'(-\theta)x < 0$. However, parameter-dependent Lyapunov matrices do not insure the stability in switching sequence as indicated in [6].

The existence of a direct common Lyapunov matrix $A_i P + PA_i' = -\theta_i$ can be searched using software for solving LMIs. However we propose another method which can find a common Lyapunov matrix with LMIs from their discrete equations. □

Lemma 4.2. *The optimal control problem for the general cost function for GPC controller in (4.1) applied to control stochastic hybrid system in (2.2) can guarantee the global and asymptotical stability if there exist positive definite matrices P and scalar γ such that*

$$\begin{bmatrix} P & PA_i' & \gamma \\ A_i P & P & 0 \\ \gamma & 0 & \gamma I \end{bmatrix} > 0, \quad \forall i. \tag{4.4}$$

Proof. Suppose there exists a Lyapunov function in (4.1) and the system will be stable if the Lyapunov function is decreasing, that is, $J(x(t+N_P+1)) < J(x(t+N_P))$, or $x(t+N_P+1)'Px(t+N_P+1) - x(t+N_P)'Px(t+N_P) < 0$, or $P - A_i'PA_i > 0$, for all i. By adding a scalar $\gamma > 0$, we have $P - A_i'PA_i - \gamma I > 0$, or $P - (A_i'P)P^{-1}(PA_i) - (\gamma)I\gamma^{-1}(\gamma) > 0$. And using Schur complement, this equation is equivalent to the LMI in Lemma 4.2.

Hence, the indirect common Lyapunov matrix in Lemma 4.2 is the solution to the following LMI:

$$\min_{P>0,\gamma>0} \gamma, \quad \text{subject to} \begin{bmatrix} P & PA_i' & \gamma \\ A_iP & P & 0 \\ \gamma & 0 & \gamma I \end{bmatrix} > 0, \ \forall i. \tag{4.5}$$

The above is CR design proposal for nonoutput tracking GPC controllers. However in reality, the primary control objective is to force the plant outputs to track their set points. What is about the CR design for tracking GPC controllers? In tracking GPC, the state space of the stochastic model in (2.2) now can be changed into a new innovation form [31]:

$$\begin{aligned} \hat{x}(t+1 \mid t) &= \widetilde{A}\hat{x}(t \mid t-1) + \widetilde{B}\Delta u(t) + \widetilde{T}\xi(t), \\ z(t) &= \widetilde{C}\hat{x}(t \mid t-1) + \xi(t), \end{aligned} \tag{4.6}$$

where $\widetilde{A}$, $\widetilde{B}$, $\widetilde{C}$, and $\widetilde{T}$ are fixed matrices from A, B, C, and T in (2.2), $\eta = \xi$, $z(t) \in \mathbb{R}^p$, $\Delta u(t) = u(t) - u(t-1) \in \mathbb{R}^m$, and $\hat{x}(t \mid t-1)$ is an estimate of state $x(t) \in \mathbb{R}^n$ obtained from a Kalman filter. For a moving horizon control, the prediction of $x(t+j \mid t)$ in (4.6) given the information $\{z(t), z(t-1), \ldots, u(t-1), u(t-2), \ldots\}$ is

$$\hat{x}(t+j \mid t) = A^j\hat{x}(t \mid t-1) + \sum_{i=0}^{j-1} A^{j-1-i}B\Delta u(t+i) + A^{j-1}T\xi(t), \tag{4.7}$$

and the prediction of the filtered output is

$$\hat{z}(t+j \mid t) = CA^j\hat{x}(t \mid t-1) + \sum_{i=0}^{j-1} CA^{j-1-i}B\Delta u(t+i) + CA^{j-1}T\xi(t). \tag{4.8}$$

If we form $\widetilde{u}(t) = [\Delta u'(t), \ldots, \Delta u'(t+N_P-1)]$ and $\widetilde{z}(t) = [\hat{z}'(t \mid t, \ldots, \hat{z}'(t+N_P-1 \mid t)]$, we can write the global prediction model for the filtered-out from 1 to N_P prediction horizon as

$$\hat{z}(t) = \begin{bmatrix} CB & \cdots & 0 \\ CAB & \cdots & 0 \\ \vdots & \vdots & \vdots \\ CA^{N_P-1}B & CA^{N_P-2}B \vdots & CB \end{bmatrix} \widetilde{u}(t) + \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_P} \end{bmatrix} \hat{x}(t \mid t-1) + \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N_P-1} \end{bmatrix} T\xi(t). \tag{4.9}$$

For simplicity, we can rewrite (4.9) as

$$\hat{z}(t) = U\widetilde{u}(t) + V\hat{x}(t \mid t-1) + WT\xi(t). \tag{4.10}$$

Consider the new tracking cost function of GPC [29]:

$$\min_{\tilde{u}(t)=[\Delta u'(t),\ldots,\Delta u'(t+N_P-1)]} \left\{ J(\tilde{u}(t),x(t)) = \sum_{j=1}^{N_P} \left[\|z(t+j)-w(t+j)\| + \|\Delta u(t+j-1)\|_\Gamma \right] \right\},$$
$$\text{subject to } x_{t+k} \in \mathbb{X}, \quad z_{t+k} \in \mathbb{Z}, \quad u_{t+k} \in \mathbb{U}, \quad \Delta u_{t+k} \in \Delta\mathbb{U} \tag{4.11}$$

where N_P is the prediction horizon, $w(t+j)$ is the output reference, and Γ is the control weighting matrix, the control law that minimizes this tracking cost function is

$$\tilde{u}(k) = -(U'U+\Gamma)^{-1}\left(V\hat{x}(t \mid t-1) + WT\xi(t) - w(t)\right) \tag{4.12}$$

then the first input $\Delta u(t)$ in $\tilde{u}(t)$ will be implemented into the system. □

Lemma 4.3. *Let (x_c,u_c) be an equilibrium pair and the corresponding equilibrium variable $z(t)=z_c$ at $w(t)=w_c$ assuming that the initial state $x(0)$ is such that a feasible solution of (4.11) exists at time $t=0$. Then the GPC law (4.12) stabilizes the system in* $\lim_{t\to\infty} x(t)=x_c$, $\lim_{t\to\infty} z(t)=w_c$, *and* $\lim_{t\to\infty}\Delta u(t)=0$ *while fulfilling constraints in (4.11).*

Proof. This stability problem follows easily from standard Lyapunov theory. Let $\tilde{u}(0)$ denote the optimal control sequence $\tilde{u}(0)=[\Delta u'(0),\ldots,\Delta u'(N_P-1)]$, let $V(t)\triangleq J(\tilde{u}(0),x(t))$ denote the corresponding value attained by the cost function, and let $\tilde{u}(1)$ be the sequence $\tilde{u}(1)=[\Delta u'(1),\ldots,\Delta u'(N_P-2)]$. Then, $\tilde{u}(1)$ is feasible at time $t+1$, along with the vectors $\Delta u(k \mid t+1)=\Delta u(k+1 \mid t)$, $z(k \mid t+1)=z(k+1 \mid t)$, $k=0,\ldots,N_P-2$, $u(N_P-1 \mid t+1)=u_c$, $z(N_P-1 \mid t+1)=z_c$, because $x(N_P-1 \mid t+1)=x(N_P \mid t)=x_c$. Hence,

$$V(t+1) \le J(\tilde{u}(1),x(t)) = V(t) - \|z(0)-w_c\| - \|\Delta u(0)\|_\Gamma \tag{4.13}$$

and $V(t)$ is reducing. Since $V(t)$ is lower bounded by 0, there exists $V_\infty=\lim_{t\to\infty}V(t)$, which implies that $V(t+1)-V(t)\to 0$. Therefore, each term of the sum

$$\|z(t)-w_c\| + \|\Delta u(t)\|_\Gamma \le V(t)-V(t+1) \tag{4.14}$$

converges to zero as well, and the system is stable.

The tracking cost function of GPC in (4.11) and (4.12) does not require to find out a Lyapunov matrix as in general cost function (4.1) and (4.2) so that the tracking GPC controller can guarantee the system stability for systems which do not have solution for the direct Lyapunov method, and can handle input and output constraints in the optimal control problem.

For tracking GPC controllers, we also propose two CR schemes for hard switcher and soft switcher as in Figure 4.1. For hard switcher, we run a tracking GPC controller corresponding to the "most reliable" mode detected by IMM as in Figure 4.1(a). However for a continuous varying variable system, a better control law is to mix all mode probabilities into a "true" model. We then build a bank of tracking GPC controllers for each model in the model set as in Figure 4.1(b). Assuming the mode probabilities are constant during the control horizon, we can easily derive a new GPC control law in (4.10) by forming

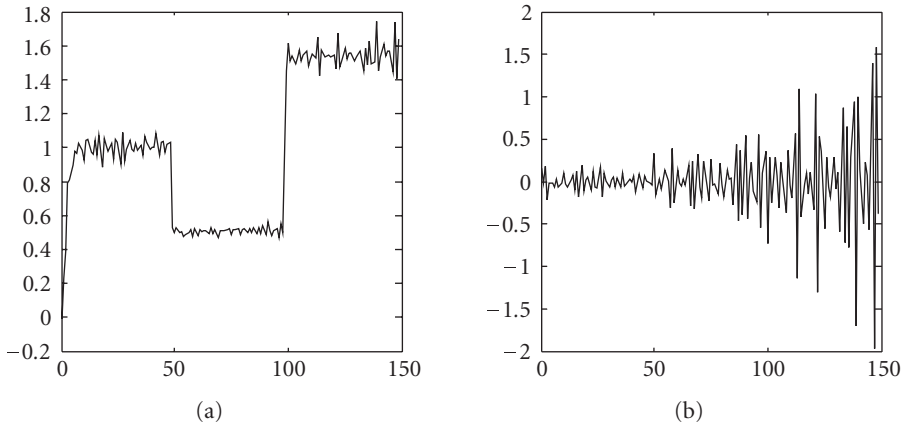


Figure 4.2. Normal GPC controller with sensor errors: (a) output and (b) input.

$U = (\sum_{i=1}^{N} \mu_i U_i)$, $V = (\sum_{i=1}^{N} \mu_i V_i)$, and $W = (\sum_{i=1}^{N} \mu_i W_i)$ matrices that correspond to the "true" model $\overline{m} = (\sum_{i=1}^{N} \mu_i m_i)$, and find out the optimal control action in (4.12). Then the first input $\Delta u(t)$ in $\tilde{u}(t)$ will be implemented into the system. Next, we will run some simulations to test the above proposed fault detection and control system. □

Example 4.4 (controller reconfiguration). The existence of a common Lyapunov matrix in (4.3) can be found by using LMI of Lemma 4.2. For simplicity, we assume that the control input $u(t+N_P) = 0$ after $k \geq N_P$ predictive control horizon so that the solution to the LMI

$$\min_{P>0,\gamma>0} \gamma, \quad \text{subject to} \begin{bmatrix} P & PA_i' & \gamma \\ A_i P & P & 0 \\ \gamma & 0 & \gamma I \end{bmatrix} > 0, \ \forall i, \tag{4.15}$$

can be applied directly to matrices $A_i = \{A_N, A_{V_0}, A_{V_1}, A_{V_2}\}$. We found that a common Lyapunov matrix for all A_i is

$$P = \begin{bmatrix} 6.43 & 1.69 & -1.62 & 0.24 \\ 1.69 & 4.34 & 0.15 & -0.30 \\ -1.62 & 0.15 & 4.14 & -0.10 \\ 0.24 & -0.30 & -0.10 & 3.25 \end{bmatrix}. \tag{4.16}$$

For tracking GPC controller, firstly we run a normal GPC controller with the predictive horizon $N_y = N_u = N_P = 4$, the weighting matrix $\Gamma = 0.1$, and with a reference set point $w = 1$. We assume that the current mode is mode S_0 from time $k = 1 - 50$, mode S_1 with sensor 1 failure -50% from time $k = 51 - 100$, and mode S_2 with sensor 1 failure $+50\%$ from time $k = 101 - 150$. Of course, the normal GPC controller provides wrong outputs (Figure 4.2).

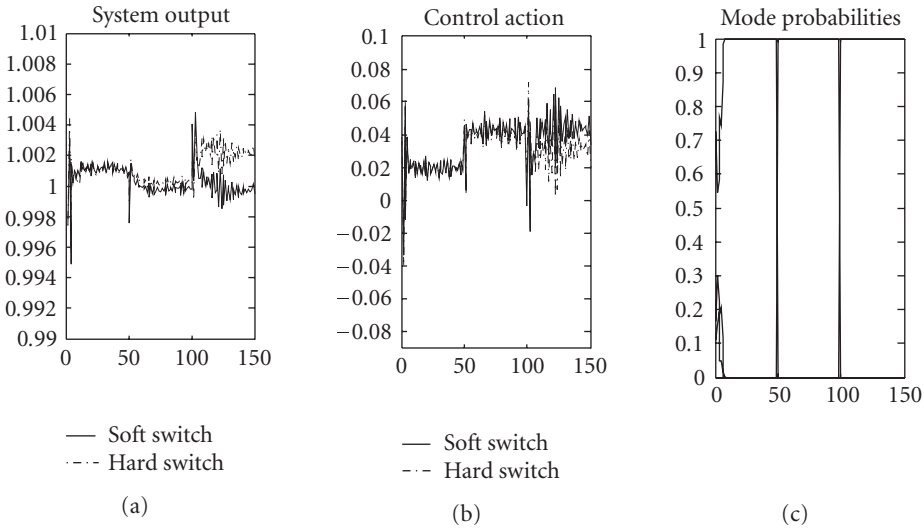


Figure 4.3. IMM-based GPC controllers: (a) output, (b) input, and (c) probabilities (IMM).

Next we run GPC controller simulations using CR system with hard switcher and soft switcher (Figure 4.3). Our new FDMP system still keeps the output at the desired set point since the IMM estimator easily finds out accurate fault modes and activates the CR system online. The soft switcher provides a smoother and smaller offset error in tracking process due to the interaction of mode probabilities that are always mixed into the "true" mode.

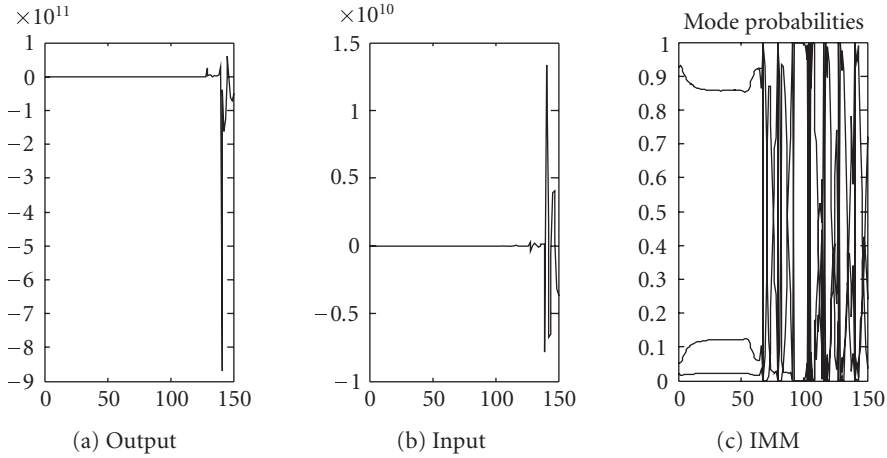
We then test the ability of the system to detect and control continuous varying variable in model set $M_3 = \{m_3, m_6, m_7\}$. Similar results are shown in Figures 3.3 and 4.3 that the IMM-based GPC controller can detect faults online and control well the varying variables with even small mode distances.

Finally, when we continue to narrow the distance between modes as we run the simulation with modes $\{m_6^*, m_3, m_7^*\}$ corresponding to $A_{V_1}^*$ with $\overline{\omega}_1^* = 0.1^0/\mathrm{s}$ and $A_{V_2}^*$ with $\overline{\omega}_2^* = -0.1^0/\mathrm{s}$, the IMM estimator fails to detect faults since the distance between modes becomes too close as shown in Figure 3.3(a), GPB1.

Low magnitude of input signals can also lead to failure of IMM-based GPC controller. If we reduce the reference set point to a very low value at $w = 0.01$, the system becomes uncontrollable (Figure 4.4): when the magnitude of the input signals is very small, the residuals of Kalman filters will be very small, and therefore, the likelihood functions for the modes will approximately be equal. This will lead to unchanging (or very slow changing) mode probabilities which in turn make the IMM estimator incapable to detect failures.

5. Conclusions

Systems subject to dynamic failures can be modeled as a set of variable structures using a variable set of models. The new structure can handle with faults varying continuously



(a) Output (b) Input (c) IMM

Figure 4.4. IMM-based GPC controller with low magnitude of input signals.

as random variables. In that case, faults can be modeled as discrete modes based on their cumulative distribution function.

One of the best methods for a fault detection of stochastic hybrid systems is using IMM algorithm. In our simulations, IMM system proves its higher ability to detect multiple failures of a dynamic process compared with that of GPB1 since the GPB1 estimator runs each elemental filter only once in each cycle while the input to each elemental filter in IMM is a weighted sum of the most recent estimates from all elemental filters.

Our proposed IMM-based GPC controller can provide real-time optimal control performance subject to input and output constrains and detection of failures. Simulations in this study show that the system can maintain the output set points amid failures. One of the main advantages of the GPC algorithm is that the controller can provide soft switching signals based on weighted probabilities of the outputs of different models. The tracking GPC controller does not require finding a common Lyapunov matrix as in the general cost function so that the tracking GPC controller can guarantee the stability of systems which are unstable for the direct Lyapunov method.

The main difficulty of this approach is the choice of modes on the model set as well as the transition probability matrix that assigns probabilities jumping from one mode to another since IMM algorithms are sensitive to the transition probability matrix and distance between modes. Another limitation related to IMM-based GPC controller is the magnitude of the noises and the input. When we change the output set points to small values, the input signals might become very small and this leads to unchanging mode probabilities, or IMM-based GPC controller cannot detect failures. Lastly, this approach does not consider issues of uncertainty in the control system.

References

[1] R. J. Patton, "Fault-tolerant control: the 1997 situation," in *Proceedings of the 3rd IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes (SAFEPROCESS '97)*, pp. 1033–1055, Hull, UK, August 1997.

[2] M. Blanke, Z. R. Izadi, S. A. Bogh, and C. P. Lunau, "Fault-tolerant control systems—a holistic view," *Control Engineering Practice*, vol. 5, no. 10, pp. 693–720, 1997.

[3] M. Bodson and J. E. Groszkiewicz, "Multivariable adaptive algorithms for reconfigurable flight control," *IEEE Transactions on Control Systems Technology*, vol. 5, no. 2, pp. 217–229, 1997.

[4] G.-H. Yang, S.-Y. Zhang, J. Lam, and J. Wang, "Reliable control using redundant controllers," *IEEE Transactions on Automatic Control*, vol. 43, no. 11, pp. 1588–1593, 1998.

[5] Y. M. Zhang and J. Jiang, "Integrated design of reconfigurable fault-tolerant control systems," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 1, pp. 133–136, 2001.

[6] M. S. Branicky, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[7] A. Bemporad, W. P. M. H. Heemels, and B. De Schutter, "On hybrid systems and closed-loop MPC systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 5, pp. 863–869, 2002.

[8] H. Nael, G. Adiwinata, and D. Panagiotis, "Fault-tolerant control of process systems using communication networks," *AIChE Journal*, vol. 51, no. 6, pp. 1665–1682, 2005.

[9] F. Christian, "Understanding fault-tolerant distributed system," *Communications of the ACM*, vol. 34, no. 2, pp. 56–78, 1991.

[10] X.-R. Li, "Hybrid estimation techniques," in *Control and Dynamic Systems*, C. T. Leondes, Ed., vol. 76, pp. 213–287, Academic Press, New York, NY, USA, 1996.

[11] X.-R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *IEEE Transactions on Automatic Control*, vol. 41, no. 4, pp. 478–493, 1996.

[12] X.-R. Li, Z.-L. Zhao, P. Zhang, and C. He, "Model-set design, choice, and comparison for multiple-model approach to hybrid estimation," in *Proceedings of the Workshop on Signal Processing, Communications, Chaos and Systems*, pp. 59–92, Newport, RI, USA, June 2002.

[13] M. Basseville, "Detecting changes in signals and systems—a survey," *Automatica*, vol. 24, no. 3, pp. 309–326, 1988.

[14] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy—a survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.

[15] J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *IEEE Control Systems Magazine*, vol. 8, no. 6, pp. 3–11, 1988.

[16] R. Isermann, "Process fault detection based on modeling and estimation method—a survey," *Automatica*, vol. 20, no. 4, pp. 387–404, 1984.

[17] R. Patton, P. Frank, and R. Clark, *Fault Diagnosis in Dynamic Systems, Theory and Applications*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1989.

[18] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.

[19] A. G. Jaffer and S. C. Gupta, "On estimation of discrete processes under multiplicative and additive noise conditions," *Information Science*, vol. 3, no. 3, pp. 267–276, 1971.

[20] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 14, no. 3, pp. 418–425, 1978.

[21] G. Ackerson and K. Fu, "On state estimation in switching environments," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 10–17, 1970.

[22] J. Tugnait, "Detection and estimation of abruptly changing systems," *Automatica*, vol. 18, no. 5, pp. 607–615, 1982.

[23] Y. Zhang and X.-R. Li, "Detection and diagnosis of sensor and actuator failures using IMM estimator," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1293–1313, 1998.

[24] M. S. Branicky, V. S. Borkar, and S. K. Mitter, "A unified framework for hybrid control: model and optimal control theory," *IEEE Transactions on Automatic Control*, vol. 43, no. 1, pp. 31–45, 1998.

[25] S. Hedlund and A. Rantzer, "Optimal control of hybrid systems," in *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 4, pp. 3972–3976, Phoenix, Ariz, USA, December 1999.

[26] A. Rantzer and M. Johansson, "Piecewise linear quadratic optimal control," *IEEE Transactions on Automatic Control*, vol. 45, no. 4, pp. 629–637, 2000.

[27] P. Riedinger, F. Kratz, C. Iung, and C. Zanne, "Optimal control of hybrid systems," in *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 3, pp. 3059–3064, Phoenix, Ariz, USA, December 1999.

[28] X. Xu and P. Antsaklis, "An approach to switched systems optimal control based on parameterization of the switching instants," in *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, July 2002.

[29] D. Clarke, C. Mohtadi, and P. Tuffs, "Generalized predictive control—part I. The basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, 1987.

[30] D. Clarke, C. Mohtadi, and P. Tuffs, "Generalized predictive control—part II. Extension and Interpretations," *Automatica*, vol. 23, no. 2, pp. 149–160, 1987.

[31] M. Kinnaert, "Adaptive generalized predictive controller for MIMO systems," *International Journal of Control*, vol. 50, no. 1, pp. 161–172, 1989.

[32] G. Ferrari-Trecate, D. Mignone, and M. Morari, "Moving horizon estimation for hybrid systems," in *Proceedings of the American Control Conference (ACC '00)*, pp. 1684–1688, Chicago, Ill, USA, June 2000.

[33] A. Bemporad, D. Mignone, and M. Morari, "Fault detection and state estimation for hybrid systems," in *Proceedings of the American Control Conference (ACC '99)*, San Diego, Calif, USA, June 1999.

[34] H. Michalska and D. Q. Mayne, "Moving horizon observers," in *Proceedings of the 2nd IFAC Symposium on Nonlinear Control Systems Design*, M. Fliess, Ed., Bordeaux, France, June 1992.

[35] K. Muske and J. B. Rawlings, "Nonlinear moving horizon state estimation," in *Methods of Model Based Process Control*, pp. 349–365, Kluwer, Boston, Mass, USA, 1995.

[36] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[37] S. Qin and T. Badgwell, "An overview of industrial model predictive control technology," in *Proceedings of the 5th International Conference on Chemical Process Control*, vol. 93 of *AIChE Symposium Series, no. 316*, pp. 232–256, Tahoe City, Calif, USA, 1997.

[38] V. T. Minh and A. Nitin, "Robust model predictive control for input saturated and soften state constraints," *Asian Journal of Control*, vol. 7, no. 3, pp. 323–329, 2005.

Vu Trieu Minh: The Sirindhorn International Thai-German Graduate School of Engineering (TGGS), King Mongkut's Institute of Technology North Bangkok, 1518 Pibulsongkram, Bangsue, Bangkok 10800, Thailand
Email address: vutrieuminh@kmitnb.ac.th

Nitin Afzulpurkar: Mechatronics, Industrial System Engineering Group (ISE), School of Engineering and Technology, Asian Institute of Technology, Klong Luang, Pathumthani 12120, Thailand
Email address: nitin@ait.ac.th

W. M. Wan Muhamad: Institute of Product Design and Manufacturing, Universiti Kuala Lumpur, Kuala Lumpur 50250, Malaysia
Email address: drwmansor@iprom.unikl.edu.my