Fault–Tolerant Predictive Control

Leonardo Giovanini, Arek Dutka, Andrzej Ordys and Mike Grimble

Industrial Control Centre, University of Strathclyde
50 George Street, Glasgow G1 1QE, UK
leonardo.giovanini@eee.strath.ac.uk, a.dutka@eee.strath.ac.uk
a.ordys@eee.strath.ac.uk, m.grimble@eee.strath.ac.uk

Abstract: The predictive feedback control strategy is used to develop a reconfigurable control system capable of compensating system component faults. The proposed formulation relies on the development of a single step predictor from input-output model that implicitly includes a fault isolation filter. Although no explicit observer is actually involved in the implementation, this predictor can be understood based due to the fact that an input-output model subsumes an implicit observer. Exploiting this idea the resulting input-output model is developing from a fault isolation filter. Simulations of a simple linear system illustrate the properties of the control algorithm in presence of actuator and sensor failures. *Copyright © 2004 IFAC*

Keywords: predictive feedback control, dynamic observer, ARX model, fault detection and isolation.

1. INTRODUCTION

The main goal of modern control law is to achieve high performance with increased safety and reliability in dynamical systems where an unlimited number of faults or failures can happen. This situation generally leads to critical changes in the system. Thus, fault–tolerant control of automated systems has become a high priority. In fact, many complex processes, when a fault happens, the maintenance or the reparation can not be achieved immediately. Hence, to preserve the safety of the operator and the reliability of the process, the presence of faults must be taken into account during the system control design.

During the last decades, different approaches dealing with this problem have been reported. Most of these studies are based on the linear–quadratic control methodology (Looze et. al., 1985; Chen and Joshi, 2002), adaptive control systems (Morse and Ossman, 1990), knowledge–based systems (Chang, Zrida and Bridwell, 1990; Polycarpou and Hemicki, 1995), hybrid systems (Blanke et al., 2000; Lunze and Steffen, 2002; Lunze and Steffen, 2003) and modifying the Youla parameter.

For real–time applications, the current fault–tolerant control design is based on a nominal control law, fault detection and isolation module, fault estimation and fault compensation (Ostroff and Hueschen, 1984). When a fault happens, the task of the fault detection and isolation module gives the type, origin and magnitude of the fault. Then, the parameters of the faulty system are then identified and the reconfiguration algorithm computes the appropriate control law (Morse and Ossman, 1990). Another methods are based on adaptive control and do not require fault diagnosis. The control system is designed such that its coefficients are adjusted to support the required dynamic system performances upon variations of the dynamic parameters caused by faults. These methods also present many advantages for reconfiguration and fault–tolerance.

In this work a fault–tolerant predictive feedback controller is presented. The proposed approach does not require residual evaluation or parameter identification. Its main advantages are that it is no time consuming, very easy to implement and takes into account input bounding constraints. The resulting controller is based on the computation of additional control law, which is added to the nominal one that compensates the fault actuator effect. The predictors employed by the controller are built from an ARX model, which employs the observer information embedded in it. This fact enables us to tackle the problems of controlling process disturbed simultaneously by unmeasured disturbances and system faults.

The paper is organized as follow: In Section 2, the control problem is presented. Section 3 is devoted to the fault estimation problem. A fault isolation and estimation filter is presented. In Section 4 the fault–tolerant predictive feedback is developed. Firstly, fault–tolerant predictors are built. Then, the fault–tolerant predictive feedback controller is developed and the meaning of the design parameters is discussed. Finally, the conclusions are presented in section 6.

2. PROBLEM FORMULATION

Considering the model–based approach, a mathematical model is built and in a non-linear case, a model linearization around an operating point is performed. For fault diagnosis and control purposes, the system is described including actuator and sensor dynamics and faults, which are represented by additive signals. The actuator and sensor dynamics are given

$$\begin{aligned} \dot{x}_A(t) &= A_A x_A(t) + B_A u(t) + f_A(t), \quad u_R(t) = C_A x_A(t), \\ \dot{x}_S(t) &= A_S x_S(t) + B_S y_R(t) + f_S(t), \quad y(t) = C_S x_S(t). \end{aligned}$$

The following state space model describes the overall system dynamic

$$\begin{aligned} \dot{X}(t) &= A_X X(t) + B_U u(t) + B_D d(t) + Ff(t), \\ y(t) &= C_X X(t), \end{aligned} \quad \textbf{(1)}$$

where $X(t)=[x_A(t)\ x(t)\ x_S(t)]^T \in R^{\text{n}}$ is the state vector, $y(t) \in R^{\text{m}}$ are the system outputs, $u(t) \in R^{\text{p}}$ are the control inputs, $u(t) \in R^{\text{q}}$ are disturbances vector, $f(t)=[f_A(t)\ f_S(t)]^T \in R^{\text{p}}$ are the vector of fault magnitudes, and $F \in R^{\text{n}\times\text{p}}$ is the distribution matrix of components faults. The system matrices are defined by

$$A_X = \begin{bmatrix} A_A & 0 & 0 \\ B_u C_A & A & 0 \\ 0 & CB_S & A_S \end{bmatrix}, B_U = \begin{bmatrix} B_u \\ 0 \\ 0 \end{bmatrix}, F = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$
$$C_X = \begin{bmatrix} 0 & 0 & C_S \end{bmatrix}, B_D = \begin{bmatrix} 0 & B_d & 0 \end{bmatrix}^T.$$

Besides, we assume that the plant described by **(1)** is subject to the following assumption: *i*) the number of available actuator effectors can exceed the number of outputs that we want to control ($p > m$), and *ii*) the $\text{rank}(C_X) = m$, $\text{rank}(F) = p$ and $\text{rank}(C_X B_D) = \text{rank}(B_D) = p$. This situation is commonly encountered in most of chemical process and mechanical system, i.e. modern aircraft.

The control objective is design a control law $u(t)$ for the system **(1)**, such that the system output $y_S(t)$ following a given reference trajectory $r(t)$ in presence of faults in control effectors and sensors. In this paper we will focus on the control effectors and sensor freezing type of failures.

3. FAULT–ESTIMATION

The estimation of the fault vector $\hat{f}(k)$ and the non–measurable disturbance $\hat{d}(k)$ can be obtained from the filter for fault isolation (Keller, 1999)

$$\begin{aligned} \hat{X}(k+1) &= [\boldsymbol{A} - K\mathsf{C}]\hat{X}(k) + B_U u(k) + [K\Sigma + \mathsf{W}\Pi]y(k), \\ \hat{y}(k) &= C_X \hat{X}(k), \\ q_r(k) &= \Pi\,(y(k) - \hat{y}(k)) \end{aligned} \quad \textbf{(2)}$$

where $q_r(k) = \left[\hat{f}(t)\hat{d}(k)\right]^T$ are the fault and non-measurable disturbance estimation and the filter matrices are given by

$$\begin{aligned} \boldsymbol{A} &= A_X - \mathsf{W}\Pi C_X, \ \mathsf{K} = K\Sigma + W\,\Pi, \\ W &= [C_X F\ C_X B_D], \mathsf{C} = \Pi C_X. \end{aligned} \quad \textbf{(3)}$$

The matrix W states the effect of faults and disturbances on the system states. The matrices Π and Σ are designed such we obtain an unbiased estimation. Then, Π and Σ must decouple the effect of the of disturbances and fault from the estimation error they satisfy the following relations

$$\begin{aligned} \Pi\,W &= I, \\ \Sigma\,W &= \mathbf{0}. \end{aligned} \quad \textbf{(4)}$$

The filter gain K is designed to obtain the best estimation possible. If there is noise in the system, we can compute K such that we obtain minimum variance estimation. In other case, with no noise in the system, we compute K to obtain the faster observer possible.

Finally, the fault isolation filter **(2)** can be directly implemented using the space–state equations or in a recursive fashion such as ARX model. The coefficients of ARX model are related with the isolation filter through (Giovanini, 2003)

$$\begin{aligned} \tilde{a}_i &= CA^{i\text{-}1}K \qquad i = 1,2,\ldots,p, \\ \tilde{b}_i &= CA^{i\text{-}1}B_U, \end{aligned} \quad \textbf{(5)}$$

where p is the ARX model order that satisfy

$$\boldsymbol{A}^p = (\boldsymbol{A} - KC)^p \approx 0. \quad \textbf{(6)}$$

Then, the output estimator filter can be written as

$$\hat{y}(k) = \sum_{i=1}^{p} \tilde{a}_i y(k-i) + \sum_{i=1}^{p} \tilde{b}_i u(k-i), \quad \textbf{(7)}$$

and the fault isolation filter is given by

$$\hat{q}_r(k) = \sum_{i=0}^{p} \tilde{g}_i y(k-i) + \sum_{i=1}^{p} \tilde{f}_i u(k-i) \quad \textbf{(8)}$$

where

$$\begin{aligned} \tilde{g}_0 = \Pi, \quad \tilde{g}_i &= -\Pi\tilde{a}_i \qquad i = 1,2,\cdots,p, \\ \tilde{f}_i &= -\Pi\tilde{b}_i. \end{aligned}$$

We must see that filter **(7)** estimates the system output including the effects of the faults and the disturbances on the system output. On the other hand, the fault isolation filter **(8)** estimates the origin and the magnitude of the faults. Then, an estimation of the real input applied to the system and real system output are given by

$$\begin{aligned} \hat{u}_R(t) &\approx u(t) + \hat{f}_A(t), \\ \hat{y}_R(t) &\approx y(t) - \hat{f}_S(t). \end{aligned} \quad \textbf{(9)}$$

This information will be used to develop a fault tolerant predictor in the next section.

4. FAULT TOLERANT CONTROL

4.1. Fault–Tolerant Predictor

To built a fault tolerant predictive controller, we need to develop an open–loop predictor that provide an unbiased prediction in spite of the system faults and/or external disturbances.

Firstly, we develop the open–loop predictor from output estimator filter **(7)**, it is given by (Giovanini, 2003)

$$\hat{y}^0(J,k) = y(k) + P_Y(J,z)y(k) + P_U(J,z)u(k) \quad \textbf{(10)}$$

where

$$\begin{aligned} P_U(J,z) &= \sum_{i=1}^{J}\left(\tilde{b}_0^{J-i} - \tilde{b}_0\right)z^{-1} + \sum_{i=1}^{p}\left(\tilde{b}_i^{J} - \tilde{b}_i\right)z^{-i}, \\ P_Y(J,z) &= \sum_{i=1}^{p}\left(\tilde{a}_i^{J} - \tilde{a}_i\right)z^{-i}, \end{aligned} \quad \textbf{(11)}$$

and the coefficients of the predictor are given by

$$\tilde{a}_i^J = \tilde{a}_{i+1}^J + \sum_{l=1}^{J} \tilde{a}_l \tilde{a}_i^{J-l} \qquad J \le p; \quad i = 0,1,\ldots,p$$
$$\tilde{a}_p^J = \sum_{l=1}^{J} \tilde{a}_l \tilde{a}_i^{J-l} \qquad J > p \tag{12.a}$$

$$\tilde{b}_i^J = \tilde{b}_{i+1}^J + \sum_{l=1}^{J} \tilde{a}_l \tilde{b}_i^{J-l} \qquad J \le p; \quad i = 0,1,\ldots,p$$
$$\tilde{b}_p^J = \sum_{l=1}^{J} \tilde{a}_l \tilde{b}_i^{J-l} \qquad J > p \tag{12.b}$$

From equation **(10)**, we can see that the open–loop predictor employed the measured output $y(k)$ to compute the J–step ahead prediction. Therefore, it includes the deviation produced by the sensor fault and the prediction will be biased in spite of developing the predictor from the output estimator filter **(7)**.

$$\hat{y}^0(J,k) = \left(1 + P_y(J,z)\right)\hat{y}_R(k) + P_u(J,z)u(k) \tag{13}$$

Then, we must correct the effect of sensor fault before computing the prediction $\hat{y}^0(J,k)$. To accomplish this correction we employed the fault isolation filter for the sensor ($F_S(z)$), developed from **(8)**, and the measured output. Subtracting the estimation of sensor fault $\hat{f}_S(t)$ from the measure output $y(k)$ (eq. **(9)**) the correction is carried out. This procedure is schematized in the block diagram showed in Figure 1. However, operating with equations **(8)** to **(12)**, it can be reduced into an only one that includes both: the predictor and the fault isolation filter. So, the overall open–loop predictor $P_Y(J,z)$ will be tolerant to sensor faults.

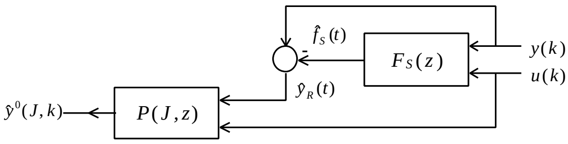

Figure 1: Structure of fault–tolerant sensor open–loop predictor $P_Y(J,z)$.

However, operating with equations **(8)** to **(12)**, it can be reduced into an only one that includes both: the predictor and the fault isolation filter. So, the overall open–loop predictor $P_Y(J,z)$ will be tolerant to sensor faults.

$$P_U^S(J,z) = P_U(J,z) + \sum_{i=1}^{p} \tilde{f}_i P_Y(J,z) z^{-i},$$
$$P_Y^S(J,z) = P_Y(J,z) - \sum_{i=1}^{p} \tilde{g}_i P_Y(J,z) z^{-i}, \tag{14}$$

The Cauchy formula applies to fault isolation sensor filter and open-loop predictor coefficients (eqs **(8)** and **(12)**) gives the coefficients of the fault-tolerant predictor.

The next step is developed an open–loop predictor that computes the effect of actuator faults on the system output. This predictor is used to build the complementary control law that compensates the effect of the actuator faults. To constructed this predictor we assembled the fault isolation filter for the actuator ($F_A(z)$), developed from **(8)**, with the open loop predictor **(10)**. The output of this predictor is the future deviation of the system output, due to the actuator faults, and the input is the estimation of actuator fault ($\hat{f}_A(t)$).

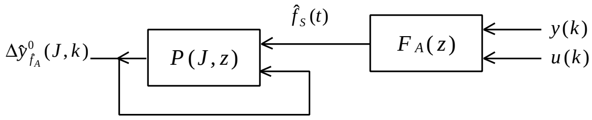

Figure 2: Structure of fault–tolerant actuator open–loop predictor $P_\Delta(J,z)$.

Operating with equations **(8)** to **(12)**, the fault actuator predictor equations can be reduced into an only one ($P_\Delta(J,z)$) that includes both the predictor and the fault isolation filter.

$$P_U^A(J,z) = P_U(J,z) + \sum_{i=1}^{p} \tilde{f}_i P_Y(J,z) z^{-i},$$
$$P_Y^A(J,z) = P_Y(J,z) - \sum_{i=1}^{p} \tilde{g}_i P_Y(J,z) z^{-i}, \tag{15}$$

Again, the coefficients of the fault actuator predictor $P_\Delta(J,z)$ will be given by the Cauchy formula, applied to coefficients of open–loop predictor and fault actuator filter.

4.2. Predictive Feedback Control

Model Predictive Control (MPC) refers to the class of algorithms that uses a model of the system to predict the future behavior of the controlled systems to compute the control actions such that a measure of the closed–loop performance is minimized and all the constrains are and will be fulfilled. The basic formulation implies a. control philosophy similar to an optimal open–loop control law that allows including constrains present in the system.

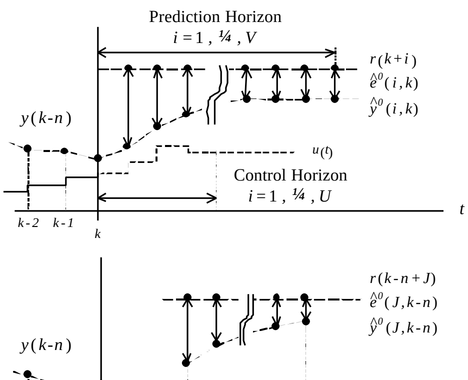

Figure 3: General predictive set–ups

However, as pointed out by (Lee and Yu, 1997) this formulation can give poor closed–loop performance, especially when uncertainties are assumed to be time–invariant in the formulation. This is true even when the underlying system is time–invariant. When the uncertainty is allowed to vary from one time step to next in the prediction, the open loop formulation gives robust, but cautious, control.

A way to solve these problems is introduce a feedback action in the predictive controller (Giovanini, 2003). So the predictive control law is given by

$$\Delta u(z) = \sum_{n=0}^{w} q_n \hat{e}^0(J,z) z^{-n} \tag{16}$$

where $\hat{e}^0(J,z)$ is the Z –transform the J step ahead open–loop predicted error, w is the filter order and q_n $n \in [0,w]$ are the controller parameters. Due to the fact that **(16)** employs only one prediction of the process future behavior, the delay operator z^{-n} $n=0,¼,w$ is applied to the time instant at the prediction is calculated. Hence, the control movement $\Delta u(k)$ is given by

$$\Delta u(k) = \sum_{n=0}^{w} q_n \hat{e}^0(J,k-n), \qquad \textbf{(17)}$$

where $\hat{e}^0(J,k-n)$ is the J step ahead open–loop error computed at time $k-n$. Replacing the open–loop predicted error (assuming a step change in the setpoint) in **(17)** and rearranging gives

$$\begin{aligned} u(k) &= \sum_{n=0}^{w} q_n e(k-n) + u(k-1) \\ &+ \sum_{n=0}^{w} q_n \left[P_Y(J,z) y(k-n) + P_U(J,z) u(k-n) \right] \end{aligned} \qquad \textbf{(18)}$$

Observe that the two first terms of the right size belong to a reduce order controller (Isermann, 1981), while the last term is the contribution of the future deviation from the reference. The last term is a weighing contribution of the future open–loop deviations at time $(k+J-n)\,t_S$

$$\Delta\hat{y}^0(J,k-n) = \hat{y}^0(J,k-n) - \tilde{y}(k-n) \qquad n = 0,1,\cdots,w$$

It states the effect of the past control actions on the future behavior of the system. It has great influence on the closed–loop performance when we have to follow a set point but a negligible one when we have to reject a disturbance, since it has little information about the disturbance. Thus, the control law **(16)** includes a feedback action –based on present and past errors– which improve the closed–loop system response. Now, it is clear that **(16)** uses more information than model predictive control. So, predictive feedback control reduces the effect of disturbances more aggressively than predictive control since it uses more information (Giovanini, 2003). Then, the predictive feedback control combines the advantages of predictive control algorithm (good setpoint tracking, time delay compensation and decoupling control) with the classical use of the feedback information to improve the disturbance rejection.

$$u(z) = \frac{F(z)}{1+F(z)P_U(J,z)} e(z) - \frac{F(z)P_Y(J,z)}{1+F(z)P_U(J,z)} y(z),$$

5. FAULT–TOLERANT PREDICTIVE CONTROL

It is assumed that the nominal system is controlled by a predictive feedback control

$$u(z) = C(z)e(z),$$

where $C(z)$ is the transfer function of the predictive feedback controller (Giovanini, 2003) given by

$$C(z) = \frac{F(z)}{1+F(z)P_U(J,z)}$$

Different techniques have dealt with the fault–tolerant control problem where the goal is to make the faulty system to be close as possible to the nominal one. Many control techniques (Gao and Antsaklis, 1991; Looze et. al., 1985; Blanke et al., 2000; Chen and Joshi, 2002) consist in computing a new control law such that the closed–loop eigenvalues remain unchanged. Even if the control law exists, the closed–loop system may be unstable in some cases. Moreover, these methods depend on the robustness of the FDI system and the ability of the supervision system to determine the fault signature and still not realistic and hardly applicable.

In this paper, another fault–tolerant control method is proposed. Its goal is to compute an additional control law, employing the additional control inputs, able to compensate the effects of the fault effect on the system. Then, the total control law is then compute by

$$\begin{bmatrix} u_P(z) \\ u_S(z) \end{bmatrix} = \begin{bmatrix} C_P(z)e(z) \\ -C_S(z)\Delta\hat{y}_{f_A}(z) \end{bmatrix}, \qquad \textbf{(19)}$$

where $u_p(z) \in R^m$ is the control variable vector with the same size as the system output, $u_S(z) \in R^{p-m}$ is the vector of remaining control inputs, Δy_f is the effect of fault actuator over the system output and, $C_P(z)$ and $C_S(z)$ are the transfer function of predictive feedback controllers.

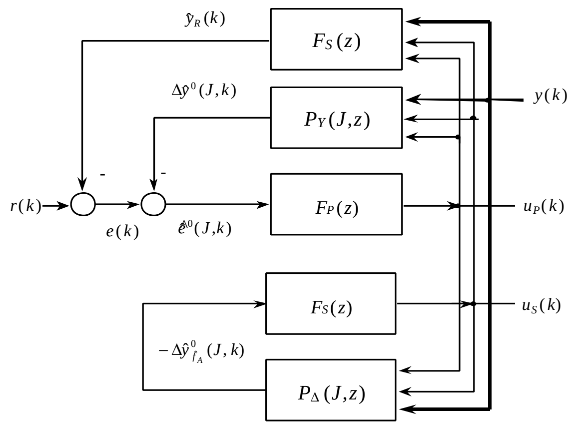

Figure 4: Structure of the fault–tolerant predictive feedback controller.

At this point, we only need to compensate sensor faults in the system output measurements $y(z)$. The correction is again executed employing equation **(9)**. Then, the fault isolation filter for the sensor $F_S(z)$ is included in the controller. The structure of the resulting predictive controller is shown in Figure 4.

6. STABILITY ANALYSIS

The stability of the faulty system given by **(1)** with fault–tolerant control method can be studied by looking at the closed–loop system driven by the new control law composed by the nominal control law $(C_P(z))$ with the additional one $(C_S(z))$

$$y(z) = G_P(z)\left(u_P(z) + \Delta u_P^f(z)\right) + G_S(z)u_S(z)\,. \qquad \textbf{(20)}$$

$$\begin{aligned} y(z) &= G_P(z)C_P(z)e(z) + G_P(z)\Delta u_P^f(z) \\ &- G_S(z)C_S(z)\Delta\hat{y}_{\hat{f}_A}(z) \end{aligned} \, . \qquad \textbf{(21)}$$

Observe that the additional control law $C_S(z)$, develops to compensate the actuator faults, acts like a feedforward control. So, it does not affect the closed–loop stability, which is determined by $C_P(z)$ law as we can see in the next section.

$$y(z)=\frac{G_P(z)C_P(z)}{1+G_P(z)C_P(z)}r(z)+\frac{G_P(z)}{1+G_P(z)C_P(z)}\Delta u_P^f(z)$$
$$-\frac{G_S(z)C_S(z)}{1+G_P(z)C_P(z)}\Delta\hat{y}_{\hat{f}_A}(z)$$

Since the system performances (essentially the stability and the transient behavior) are fixed by the poles of closed–loop equation, $1+C_P(z)G_P(z)$, these performances are still preserved if the new control inputs ($u_S(z)$) remain inside the physical capacities of the actuator.

7. SIMULATION AND RESULTS

Now, to demonstrate the effectiveness of the fault–tolerant predictive feedback controller we consider the following simulation example obtained from the linearization of a continue stirred tank reactor (CSTR) with a jacket, where a first order exothermic reaction is carried out, which was previously used by (Morningred et al, 1992)

In normal operative conditions, the CSTR is controlled by manipulating the coolant flow rate ($q_C(t)$) while the remaining variables remain unchanged. When a fault in the valve of coolant flow rate happen, the reaction can be controlled by manipulating feed flow rate ($q(t)$). Under this operative condition we have to introduce a level control, such that the volume of the reactor and the residence time remain constant.

The linear model employed in the simulations is obtained from the linearization of CSTR non-linear model around the nominal operation point ($Ca=0.05$ $mol\ lt^{-1}$, $T=\ T_j=350\ K$ and $q_C=q=100\ \ lt\ min^{-1}$). The resulting matrices are

$$A=\begin{bmatrix}-1.9998 & -0.0065 & 0\\ 358.7137 & 1.0917 & 0.0002\\ 0 & 0.0006 & -0.0012\end{bmatrix},\quad C=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},$$
$$B_u=\begin{bmatrix}0 & 0.0011\\ -0.1458 & -0.1459\\ 0.5642 & 0\end{bmatrix},\quad B_d=\begin{bmatrix}0.25 & 0\\ 0 & 0\\ 0 & 0.0006\end{bmatrix}. \qquad \textbf{(22)}$$

The disturbances considered in this example are the feed concentration $Ca_0(t)$ and the inlet coolant temperature $T_{CO}(t)$. To complete the description of the system we include the actuators and the concentration sensor dynamic, which are given by

$$A_A=\begin{bmatrix}-20 & 0\\ 0 & -20\end{bmatrix},\quad B_A=C_A=\begin{bmatrix}1&0\\0&1\end{bmatrix},$$
$$A_S=[-0.1],\quad B_A=C_S=[1]. \qquad \textbf{(23)}$$

The dynamics of the temperature sensors are dismissed because they are faster and more reliable than concentration sensor. Different faults are simulated in this system. They correspond to actuator and sensor freeze and loss in the sensitivity of the actuator and sensor. The disturbance that will be estimated is the feed concentration, because it is non-measurable.

Like in Morningred's work (1992), the sampling time period was fixed in 0.1 min, which gives about four sampled-data points in the dominant time constant when the reactor is operating in the high concentration region. Then, the discrete linear model is obtained by Z –transforming the continue model assuming a zero–order-hold device is included.

Firstly, we develop the fault isolation and estimation filter. We check the isolating condition

$$rank([C_X F\ C_X B_D])=p+q=3$$

This condition implies that we can isolate the faults and the disturbance simultaneously with only one filter. Then, we find the fault isolation matrices Π and Σ, and the filter gain K. The fault isolation matrices were computed such that they satisfy **(4)**

$$\Pi=\begin{bmatrix}0.2509 & -0.0134 & 17.7211\\ 0.1278 & 2.2921 & 0.6237\end{bmatrix},$$
$$\Sigma=\begin{bmatrix}0.9971 & -0.0517 & -0.0142\\ -0.0517 & 0.0027 & 0.0007\\ -0.0142 & 0.0007 & 0.0002\end{bmatrix},$$

and the filter gain K was designed such that the filter **(2)** has a deadbeat behavior

$$K=\begin{bmatrix}0.505 & 0.0117 & 0\\ 0.0117 & 0.998 & 0\\ 0 & 0 & 0.618\end{bmatrix}.$$

Then, we built the open–loop predictors $P_Y(J,z)$ and $P_\Delta(J,z)$ using the output estimator and fault isolation filters (eqs **(7)** and **(8)**) and the coefficients of the open–loop predictor (eqs **(12.a)** and **(12.b)**).

Figure 5 shows the results obtained in the simulations. The set point was changed in intervals from 0.1 to 0.125 at 10 min and returns to 0.1 at 45 min. At 25 min the feed stream concentration changes from 1 $mol\,lt^{-1}$ to 1.05 $mol\,lt^{-1}$, and finally the valve that control the coolant flow rate freeze at 47.5 min.

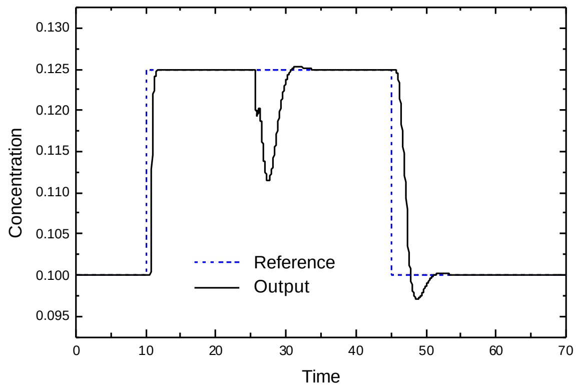

Figure 5: System for faulty conditions

The effect of the disturbance is to drive the principal actuator to saturation (Figure 6). However, the proposed algorithm is able to reconfigure the system, treating the saturation like a fault. Besides, in Figure 6 it possible to appreciate the reconfiguration of the controller when the valve fails.

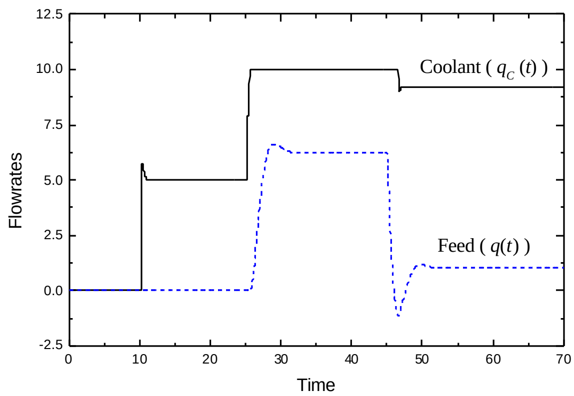

Figure 6: Manipulated variables for the simulation

It can easily be seen that the output of the reconfigured loop matches the output of the nominal loop. Due to these differences, the control performance of the reconfigured control loop is slightly inferior to the nominal control loop. However, the stability of the reconfigured control loop is guaranteed under all operating conditions. It is also interesting to see the differences between the output of the controller (which is the input of the nominal plant) and the output of the second actuator (q) that controls the faulty plant

8. CONCLUSIONS

The method developed in this paper emphasizes the importance of the fault–tolerant control in high-automated systems. The problems inherent to the fault detection and isolation module are avoided using an additional control law, implemented through the additional control inputs, based on the on–line estimation of the fault magnitude. This method is suitable for every type of actuator faults.

Acknowledgements: The authors are grateful for the financial support for this work provided by the Engineering and Physical Science Research Council (EPSRC) grant Industrial Non-linear Control and Applications GR/R04683/01.

REFERENCES

Blanke M., Frei W., Kraus F., Patton R. and Starosweicki M. (2000). "What is fault tolerant?", Proc. of Safeprocess 2000,pp. 40–51.

Chang, G Zrida, J and Birdwell, J "Knowledge–based Distribution System Analysis and Reconfiguration", *IEEE Trans on Power Systems*, 1990, Vol. 5, pp. 744–749.

Chen G. and Joshi M. (2002). "On matching conditions for adaptive state tracking control of systems with actuator failures", *IEEE Trans on Autom Control*, Vol. 47, pp. 743–478.

Gao, Z and Antsaklis, P. "Stability of the Pseudo–inverse Method for Reconfigurable Control Systems", *International Journal of Control*, Vol. 53, 1991, pp. 717–729.

Giovanini, L and Marchetti, J. "Shaping time–domain response with discrete controllers", *Ind. Eng. Chem. Res.*, Vol. 38(12), 1999, pp. 47774789.

Giovanini L. (2003). "Predictive feedback control", *ISA Transaction Journal*, Vol. 42(2), pp. 207–226

Giovanini L. (2004). "Embedded estimators in predictive feedback control", *ISA Transactions Journal*, Vol. 43(3), pp. 343-359.

Keller J. (1999). "Fault isolation filter design for linear stochastic systems", *Automatica*, Vol. 35(10), pp. 1701-1706.

Isermann, R. *Digital Control Systems* Springer–Verlag, 1981.

Lee, J. and Yu, Z. (1997). "Worst-case formulations of model predictive control for systems with bounded parameters", *Automatica*, Vol. 33(5), pp. 763–781.

Looze, D; Weiss, J; Eterno, J and Barett, N. (1985). "An Automatic Redesign Approach for Restructurable Control Systems", *Control Systems Magazine*.

Lunze J. and Steffen T. (2002). "Hybrid reconfigurable control", in *Modelling, Analysis and Design Hybrid Systems*, Engell S., Freshe G. and Schenider E. Eds. Springer-Verlag, pp. 267-284.

Lunze J. and Steffen T. (2003). "Control reconfiguration by means of a virtual actuator", Proc. of Safeprocess 2003, pp. 133-140.

Morningred, W, Paden, B; Seborg, D and Mellichamp, D. (1992). "An Adaptive Nonlinear Predictive Controller", *Chem. Eng. Sci.*, Vol. 47, pp. 969–976.

Morse, W and Ossman, K. (1990). "Model Following reconfigurable flight control system for the AFTI/16", *Journal Guidance*, Vol. 13, pp. 969–976.

Ostroff, A and Hueschen, R. "Investigation of Control Law Reconfigurations to Accommodate a Control Element Failure on Commercial Airplane", *Proc. ACC'84*, 1984, San Diego.

Polycarpou, M and Helmicki, A. (1995). "Automated Fault Detection and Accommodation: A Learning System Approach", *IEEE Trans on Sys, Man and Cybernetic*, Vol. 25(11),