From data to diagnosis and control using generalized orthonormal basis filters. Part II: Model predictive and fault tolerant control

Sachin C. Patwardhan [a], Seema Manuja [b], Shankar Narasimhan [b], Sirish L. Shah [c,*]

[a] *Department of Chemical Engineering, Indian Institute of Technology, Bombay 400076, India*
[b] *Department of Chemical Engineering, Indian Institute of Technology, Madras 600036, India*
[c] *Department of Chemical and Materials Engineering, University of Alberta, Edmonton, Canada T6G 2G6*

Abstract

Given a state space model together with the state noise and measurement noise characteristics, there are well established procedures to design a Kalman filter based model predictive control (MPC) and fault diagnosis scheme. In practice, however, such disturbance models relating the true root cause of the unmeasured disturbances with the states/outputs are difficult to develop. To alleviate this difficulty, we reformulate the MPC scheme proposed by K.R. Muske and J.B. Rawlings [Model predictive control with linear models, AIChE J. 39 (1993) 262–287] and the fault tolerant control scheme (FTCS) proposed by J. Prakash, S.C. Patwardhan, and S. Narasimhan [A supervisory approach to fault tolerant control of linear multivariable systems, Ind. Eng. Chem. Res. 41 (2002) 2270–2281] starting from the innovations form of state space model identified using generalized orthonormal basis function (GOBF) parameterization. The efficacy of the proposed MPC scheme and the on-line FTCS is demonstrated by conducting simulation studies on the benchmark shell control problem (SCP) and experimental studies on a laboratory scale continuous stirred tank heater (CSTH) system. The analysis of the simulation and experimental results reveals that the MPC scheme formulated using the identified observers produces superior regulatory performance when compared to the regulatory performance of conventional MPC controller even in the presence of significant plant model mismatch. The FTCS reformulated using the innovations form of state space model is able to isolate sensor as well as actuator faults occurring sequentially in time. In particular, the proposed FTCS is able to eliminate offset between the true value of the measured variable and the setpoint in the presence of sensor biases. Thus, the simulation and experimental study clearly demonstrate the advantages of formulating MPC and generalized likelihood ratio (GLR) based fault diagnosis schemes using the innovations form of state space model identified from input output data.

Keywords: Innovations form of state space model; Model predictive control; Unmeasured disturbances; Regulatory performance; Likelihood ratio method; Fault tolerant control

1. Introduction

Model predictive control (MPC) schemes developed in late 1970s, such as dynamic matrix control (DMC) or model algorithmic control (MAC), have found wide acceptance in the process industry [1]. The simplistic model used to capture the effect of unmeasured disturbances on the future predictions was a major drawback of these control algorithms. In order to overcome this drawback, Navratil et al. [2] and Ricker [3,4] proposed an alternative formulation of MPC based on a state space model, which represents unmeasured disturbances and measurement noise as stochastic processes. This state space model was used to develop a Kalman filter, which explicitly accounts for the effects of the past unmeasured disturbances on the current state estimates and, in turn, improves the predicted state estimates over the future horizon. The development and analysis of schemes based on the stochastic state space models has been one of the main directions of research in the area of linear model predictive control (MPC) in the last decade [5–9].

Another area of research which is gaining increasing importance in recent years is on-line model based diagnosis of incipient faults [10,11]. Chemical plants usually involve a large number of measurements and actuators, which are used for process monitoring and control. *Soft faults* such as biases or drifts in measurement sensors and actuators are probably the most commonly encountered faults in process operation. If these faults are not detected in time and a corrective action is not taken to counteract their effects, then they can cause degradation in the closed loop performance and can have an impact on the safety and the productivity of the plant. Thus, in order to ensure that plant operation continues without much degradation in control performance until the scheduled maintenance of the measuring instruments/control valves is carried out, it is necessary to develop fault tolerant control schemes (FTCS), which can *accommodate* soft faults in these devices. Recently, Prakash et al. [12] have proposed an on-line fault tolerant control scheme (FTCS), which combines a model based fault detection and identification (FDI) technique with a conventional controller such as MPC. Given a linearized first principles model together with the state noise and the measurement noise characteristics, they have developed procedures for on-line diagnosis and compensation of biases and drifts in sensors, actuators, unmeasured disturbances and model parameters. The FDI component used in their scheme is based on a modified on-line version of the generalized likelihood ratio (GLR) method proposed by Willsky and Jones [13], which uses the innovation sequence generated by the Kalman filter for fault identification. Thus, similar to the state space formulation of MPC schemes, the key component of the GLR based FDI scheme is the Kalman filter, which provides the optimal (or the maximum likelihood) estimates of states in presence of unmeasured disturbances.

Given the state noise and measurement noise characteristics, there are well established procedures to design a Kalman filter based model predictive controller or a GLR based FDI scheme. In most of the Kalman filter based MPC formulations available in the literature or in the FTCS proposed by Prakash et al. [12], it is implicitly assumed that the model relating actual (physical) unmeasured disturbance variables (such as feed composition fluctuations) with the state/output dynamics is available a priori. In practice, however, such disturbance models relating the true root cause of the unmeasured disturbances with the states/outputs are difficult to develop. In addition to the disturbance model, the development of Kalman filter requires knowledge of state and measurement noise characteristics. Since the covariances of state and measurement noise are typically not known in many practical situations, these noise model parameters are treated as tuning parameters in most of the formulations [4,6]. Even in the GPC formulation proposed by Clarke et al. [14], the coefficients of the noise polynomial (or the C polynomial) are often treated as tuning parameters. If the system dimension is small and physical sources of unmeasured disturbances are known, it is relatively easy to select the noise model parameters. However, the task of tuning these parameters can become difficult for a large dimensional system. Moreover, the state estimates generated by the Kalman filter with a *tuned* noise model may be suboptimal if the true nature of the unmeasured disturbances is significantly different. The inaccuracies in the state estimates may not cause serious difficulties in MPC formulation. However, they can cause significant degradation in the performance of the FDI scheme [12].

One remedy to this problem is to develop stochastic models for the unmeasured disturbances directly from the input–output data and use them in MPC/FDI formulations [15–17]. From a practical viewpoint, this can considerably simplify the implementation of a MPC/ FDI scheme based on the stochastic state space models. Conventionally, the noise modeling is viewed only as a way of improving the deterministic part of the model. The present work, however, is based on the premise that the identified stochastic component of the model can be potentially useful in improving predictions [8] and can be used directly in the MPC formulation for improving the regulatory performance or for developing GLR based FDI scheme for isolation of actuator and sensor faults.

The objective of the current work is to demonstrate the development of a model predictive and fault tolerant control scheme using an innovations form of state space model derived purely from data using system identification techniques. The innovations form of state space model developed in the accompanying paper [20] is already in the form of a Kalman predictor and the Kalman gain matrix has been identified directly from the data (see [18,19] for detailed discussion). In the present work, we reformulate the MPC scheme proposed by Muske and Rawlings [5] and the FTCS scheme by Prakash et al. [12] using the identified innovations form of state space model in the accompanying paper [20]. Note that, unlike the MISO models developed conventionally, these identified state space models also account for the correlations between the outputs as stressed by Morari and Lee [8] while discussing identification related issues in their recent review on MPC. In the reformulation of the fault tolerant control scheme using identified models, we limit our considerations to soft faults caused by sensor and actuator biases. If a model of the process is developed from first principles, then it is possible to do root cause analysis of changes in the unmeasured disturbances and small magnitude parametric drifts and use the diagnostic information in a fault tolerant control scheme as demonstrated by Prakash et al. [12]. A key problem that needs to be addressed with identified models is the isolation of such changes with no a priori knowledge of their effect on process dynamics. The current state of the art even in fault detection and identification (FDI)—let alone fault tolerant control—does not adequately address the problem of diagnosing multiplicative faults using identified models. Therefore, the scope of this paper is limited to additive soft faults in sensors and actuators. The efficacy of the proposed MPC scheme and the on-line FTCS is demonstrated by conducting simulation studies on the benchmark shell control problem (SCP) and experimental studies on a laboratory scale continuous stirred tank heater (CSTH) system.

This paper is organized in seven sections. The next section presents development of the state estimator based on the minimal order innovations form of state space model identified in the accompanying paper [20]. The MPC and FTCS schemes are reformulated using this state estimator in Sections 3 and 4, respectively. Section 5 presents the results of the simulation studies carried out on the shell control problem and Section 6 presents the results of experimental verification using the CSTH. The main conclusions reached from the simulation and the experimental studies are discussed in the last section.

2. State space model and Kalman filter

The main component of MPC or FTCS is the model describing deterministic and stochastic contributions to process dynamics, which is used to develop the Kalman filter. The minimal/low order innovations form of state space model developed in the accompanying paper [20] is of the form

$$\mathbf{X}(k+1) = \mathbf{\Phi X}(k) + \mathbf{\Gamma u}(k) + \mathbf{Ke}(k), \tag{2.1}$$

$$\mathbf{y}(k) = \mathbf{CX}(k) + \mathbf{e}(k), \tag{2.2}$$

where $\mathbf{X}(k) \in R^n$, $\mathbf{u}(k) \in R^m$ and $\mathbf{y}(k) \in R^r$ represent state, manipulated input and output vectors, respectively. Here, the innovations (or residuals) $\{\mathbf{e}(k)\}$ are a zero mean Gaussian white noise sequence with covariance matrix $\mathbf{V}_e$ and $\mathbf{K}$ represents the corresponding steady state Kalman gain. The above model can be re-written as

$$\mathbf{X}(k+1) = \mathbf{\Phi X}(k) + \mathbf{\Gamma u}(k) + \mathbf{w}(k), \tag{2.3}$$

$$\mathbf{y}(k) = \mathbf{CX}(k) + \mathbf{v}(k), \tag{2.4}$$

where the vectors $\mathbf{w}(k) \in R^n$ and $\mathbf{v}(k) \in R^r$ are zero mean white noise sequences such that

$$\mathbf{R}_1 = E[\mathbf{w}(k)\mathbf{w}(k)^{\mathrm{T}}] = \mathbf{KV}_e\mathbf{K}^{\mathrm{T}}, \tag{2.5}$$

$$\mathbf{R}_{12} = E[\mathbf{w}(k)\mathbf{v}(k)^{\mathrm{T}}] = \mathbf{KV}_e, \tag{2.6}$$

$$\mathbf{R}_2 = E[\mathbf{v}(k)\mathbf{v}(k)^{\mathrm{T}}] = \mathbf{V}_e. \tag{2.7}$$

Note that, the use of the identification procedure proposed in the accompanying paper [20] implies that the model is open loop stable, controllable and observable (by construction). This model can be used to develop the optimal state predictor as follows:

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}\widehat{\mathbf{X}}(k/k-1), \tag{2.8}$$

$$\widehat{\mathbf{X}}(k+1/k) = \mathbf{\Phi}\widehat{\mathbf{X}}(k/k-1) + \mathbf{\Gamma u}(k) + \mathbf{Ke}(k), \tag{2.9}$$

where $\mathbf{K}$ is the solution of the steady state Riccati equation

$$\mathbf{K} = [\mathbf{\Phi P}_\infty\mathbf{C}^{\mathrm{T}} + \mathbf{R}_{12}][\mathbf{CP}_\infty\mathbf{C}^{\mathrm{T}} + \mathbf{R}_2]^{-1}, \tag{2.10}$$

$$\mathbf{P}_\infty = \mathbf{\Phi P}_\infty\mathbf{\Phi}^{\mathrm{T}} + \mathbf{R}_1 - \mathbf{K}[\mathbf{CP}_\infty\mathbf{C}^{\mathrm{T}} + \mathbf{R}_2]\mathbf{K}^{\mathrm{T}}. \tag{2.11}$$

Here, matrix $\mathbf{P}_\infty$ denotes steady state covariance of error in state estimation. Note that matrix $\mathbf{K}$ appearing in Eq. (2.1) and $\mathbf{K}$ obtained by solving the above Riccati equations are identical matrices (see [18,19]).

3. Model predictive control formulation

In this section, we reformulate model predictive control using the model given by Eqs. (2.3)–(2.7). Note that the deterministic component of the innovations form of state space model equations (2.1)–(2.2) developed in the accompanying paper [20] has been parameterized using generalized orthonormal basis filters (GOBF). As a consequence, we restrict ourselves to predictive control formulation for open loop stable processes. If the process under consideration has unstable or integrating mode(s), it is assumed that these modes are stabilized by feedback prior to the model development exercise.

3.1. Model based prediction of future behavior

In a typical MPC formulation, at each sampling instant, the optimal state estimator/filter is used for predicting future behavior of the plant over a finite future time horizon of length p (defined as the *prediction horizon*) starting from the current time instant k. Let us assume that at any instant k, we are free to choose only q future manipulated input moves

$$\{\mathbf{u}(k/k), \mathbf{u}(k+1/k), \ldots, \mathbf{u}(k+q-1/k)\},$$

with the following constraints on the remaining future input moves:

$$\mathbf{u}(k+q/k) = \mathbf{u}(k+q+1/k) = \cdots = \mathbf{u}(k+p-1/k) = \mathbf{u}(k+q-1/k), \quad (3.1)$$

where q is defined as the *control horizon*. As the expected value of future innovations is zero, the optimal predicted estimates of the state variables can be generated by recursively using the state estimator (2.8) and (2.9) as follows:

$$\widehat{\mathbf{X}}(k+1/k) = \mathbf{\Phi}\widehat{\mathbf{X}}(k/k-1) + \mathbf{\Gamma}\mathbf{u}(k/k) + \mathbf{K}\mathbf{e}(k), \quad (3.2)$$

$$\widehat{\mathbf{X}}(k+j+1/k) = \mathbf{\Phi}\widehat{\mathbf{X}}(k+j/k) + \mathbf{\Gamma}\mathbf{u}(k+j/k) \quad \text{for } j = 1, \ldots, p-1. \quad (3.3)$$

A predictive control formulation based on the above optimal predictions can pose practical difficulties in the presence of mismatch between the plant and the model. Since the model is identified once in the initial phase of the MPC implementation, discrepancies between plant and model parameters can arise over the period of time due to shift in the operating point, changes in the disturbance characteristics, etc. Thus, there is a need to introduce some mechanism to account for the plant model mismatch, which, in turn, introduces integral action in the controller formulation. This can be achieved by augmenting the state space model with artificially introduced input and/or output disturbance variables, which behave as integrated white noise sequences [5,9]. The resulting augmented model can be written as

$$\mathbf{X}(k+1) = \mathbf{\Phi}\mathbf{X}(k) + \mathbf{\Gamma}\mathbf{u}(k) + \mathbf{\Gamma}_\beta\boldsymbol{\beta}(k) + \mathbf{w}(k), \quad (3.4)$$

$$\boldsymbol{\beta}(k+1) = \boldsymbol{\beta}(k) + \mathbf{w}_\beta(k), \quad (3.5)$$

$$\boldsymbol{\eta}(k+1) = \boldsymbol{\eta}(k) + \mathbf{w}_\eta(k), \quad (3.6)$$

$$\mathbf{y}(k) = \mathbf{C}x(k) + \mathbf{C}_\eta\boldsymbol{\eta}(k) + \mathbf{v}(k), \quad (3.7)$$

where $\boldsymbol{\beta} \in \mathbf{R}^s$ and $\boldsymbol{\eta} \in \mathbf{R}^t$ are artificially introduced input and output disturbance vectors while vectors $\mathbf{w}_\beta \in \mathbf{R}^s$ and $\mathbf{w}_\eta \in \mathbf{R}^t$ are zero mean white noise sequences with covariances $\mathbf{Q}_\beta$ and $\mathbf{Q}_\eta$, respectively. The model coefficient matrices $(\mathbf{\Gamma}_\beta, \mathbf{C}_\eta)$ and noise covariances matrices $(\mathbf{Q}_\beta, \mathbf{Q}_\eta)$ are treated as tuning parameters, which can be chosen to achieve the desired closed loop disturbance rejection characteristics. The above set of equations can be combined into an augmented state space model of the form

$$\mathbf{X}_a(k+1) = \mathbf{\Phi}_a\mathbf{X}_a(k) + \mathbf{\Gamma}_a\mathbf{u}(k) + \mathbf{w}_a(k), \quad (3.8)$$

$$\mathbf{y}(k) = \mathbf{C}_a\mathbf{X}_a(k) + \mathbf{v}(k), \quad (3.9)$$

where

$$\mathbf{X}_a(k) = \begin{bmatrix} \mathbf{X}(k) \\ \boldsymbol{\beta}(k) \\ \boldsymbol{\eta}(k) \end{bmatrix}, \quad \mathbf{w}_a(k) = \begin{bmatrix} \mathbf{w}(k) \\ \mathbf{w}_\beta(k) \\ \mathbf{w}_\eta(k) \end{bmatrix},$$

$$\mathbf{\Phi}_a = \begin{bmatrix} \mathbf{\Phi} & \mathbf{\Gamma}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{I}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{I}_\eta \end{bmatrix}, \quad \mathbf{\Gamma}_a = \begin{bmatrix} \mathbf{\Gamma} \\ [\mathbf{0}] \end{bmatrix},$$

$$\mathbf{C}_a = [\,\mathbf{C} \quad [\mathbf{0}] \quad \mathbf{C}_\eta\,],$$

$$\mathbf{R}_{1a} = E[\mathbf{w}_a(k)\mathbf{w}_a(k)^{\mathrm{T}}] = \begin{bmatrix} \mathbf{K}\mathbf{V}_e\mathbf{K}^{\mathrm{T}} & [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{Q}_\beta & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{0}] & \mathbf{Q}_\eta \end{bmatrix},$$

$$\mathbf{R}_{12a} = E\left[\mathbf{w}_a(k)\mathbf{v}(k)^{\mathrm{T}}\right] = \begin{bmatrix} \mathbf{K}\mathbf{V}_e \\ [\mathbf{0}] \end{bmatrix},$$

$$\mathbf{R}_{2a} = E\left[\mathbf{v}(k)\mathbf{v}(k)^{\mathrm{T}}\right] = \mathbf{V}_e.$$

This augmented model can be used for developing a Kalman predictor of the form

$$\mathbf{e}_a(k) = \mathbf{y}(k) - \mathbf{C}_a\widehat{\mathbf{X}}_a(k/k-1) \quad (3.10)$$

$$\widehat{\mathbf{X}}_a(k+1/k) = \mathbf{\Phi}_a\widehat{\mathbf{X}}_a(k/k-1) + \mathbf{\Gamma}_a\mathbf{u}(k) + \mathbf{K}_a\mathbf{e}_a(k), \quad (3.11)$$

where the steady state Kalman gain is obtained by solving the corresponding steady state Riccati equations similar to Eqs. (2.10) and (2.11)

$$\mathbf{K}_a = \left[\mathbf{\Phi}_a\mathbf{P}_{a\infty}\mathbf{C}_a^{\mathrm{T}} + \mathbf{R}_{12a}\right]\left[\mathbf{C}_a\mathbf{P}_{a\infty}\mathbf{C}_a^{\mathrm{T}} + \mathbf{R}_{2a}\right]^{-1}, \quad (3.12)$$

$$\mathbf{P}_{a\infty} = \mathbf{\Phi}_a\mathbf{P}_{a\infty}\mathbf{\Phi}_a^{\mathrm{T}} + \mathbf{R}_{1a} - \mathbf{K}_a\left[\mathbf{C}_a\mathbf{P}_{a\infty}\mathbf{C}_a^{\mathrm{T}} + \mathbf{R}_{2a}\right]\mathbf{K}_a^{\mathrm{T}}. \quad (3.13)$$

In order to maintain the observability of the artificially introduced states, the number of additional states introduced in the augmented model should not exceed the number of measured outputs [9]. Since the state space model (2.3) and (2.4) is observable and stable with no integrating modes, the augmented state space model will be detectable in most cases. The conditions under which the augmented model will be detectable are discussed in [9]. The optimal future state predictions based on measurements up to instant k can be generated using the augmented state space model as follows:

$$\widehat{\mathbf{X}}_a(k+j+1/k) = \mathbf{\Phi}_a\widehat{\mathbf{X}}_a(k+j/k) + \mathbf{\Gamma}_a\mathbf{u}(k+j/k), \quad (3.14)$$

$$\hat{\mathbf{y}}(k+j+1/k) = \mathbf{C}_a\widehat{\mathbf{X}}_a(k+j+1/k) \quad \text{for } j = 1, \ldots, p-1. \quad (3.15)$$

3.2. MPC formulation

In addition to predicting the future output trajectory, at each instant, a filtered future setpoint trajectory is generated using a reference system of the form

$$\mathbf{X}_r(k+j+1/k) = \mathbf{\Phi}_r\mathbf{X}_r(k+j/k) + \Gamma_r[\mathbf{r}(k) - \mathbf{y}(k)], \tag{3.16}$$

$$\mathbf{y}_r(k+j+1/k) = \mathbf{y}(k) + \mathbf{C}_r\mathbf{X}_r(k+j+1/k), \quad \text{for } j = 0,1,\ldots,p-1, \tag{3.17}$$

with initial condition $\mathbf{X}_r(k/k) = \mathbf{0}$ and with steady state gain $\mathbf{C}_r(\mathbf{I} - \mathbf{\Phi}_r)^{-1}\mathbf{\Gamma}_r = \mathbf{I}$. Here, $\mathbf{r}(k) \in R^{r}$ represents the setpoint vector. The coefficient matrices of the reference system are tuning parameters, which can be selected to achieve the desired closed loop tracking performance. Defining the future the prediction error vector $\mathbf{e}_f(k+i/k)$ as

$$\mathbf{e}_f(k+i/k) = \mathbf{y}_r(k+i/k) - \hat{\mathbf{y}}(k+i/k),$$

the model predictive control problem at the sampling instant k is defined as a constrained optimization problem whereby the future manipulated input moves $\mathbf{u}(k/k), \mathbf{u}(k+1/k), \ldots, \mathbf{u}(k+q-1/k)$ are determined by minimizing an objective function

$$\min_{\mathbf{u}(k/k),\ldots,\mathbf{u}(k+q-1/k)} \sum_{i=1}^{p} \mathbf{e}_f(k+i/k)^{\mathrm{T}}\mathbf{W}_E\mathbf{e}_f(k+i/k) + \sum_{i=1}^{q-1}\Big\{\Delta\mathbf{u}(k+i/k)^{\mathrm{T}}\mathbf{W}_U\Delta\mathbf{u}(k+i/k) + \Delta\mathbf{u}_s(k+i/k)^{\mathrm{T}}\mathbf{W}_S\Delta\mathbf{u}_s(k+i/k)\Big\}$$

subject to the following constraints:

$$\mathbf{y}^{\mathrm{L}} \leqslant \hat{\mathbf{y}}(k+i/k) \leqslant \mathbf{y}^{\mathrm{H}} \quad \text{for } i = 1,2,\ldots,p,$$
$$\mathbf{u}^{\mathrm{L}} \leqslant \mathbf{u}(k+i/k) \leqslant \mathbf{u}^{\mathrm{H}} \quad \text{for } i = 0,1,\ldots,q-1,$$
$$\Delta\mathbf{u}^{\mathrm{L}} \leqslant \Delta\mathbf{u}(k+i/k) \leqslant \Delta\mathbf{u}^{\mathrm{H}} \quad \text{for } i = 0,1,\ldots,q-1,$$

where

$$\Delta\mathbf{u}(k+i/k) = \mathbf{u}(k+i/k) - \mathbf{u}(k+i-1/k),$$
$$\Delta\mathbf{u}_s(k+i/k) = \mathbf{u}(k+i/k) - \mathbf{u}_{s,k}.$$

Here, the *steady state* input vector $\mathbf{u}_{s,k}$ can be computed by solving the following optimization problem:

$$\min_{\mathbf{u}_s(k)}[\mathbf{r}(k) - \mathbf{y}_{s,k}]^{\mathrm{T}}\mathbf{Q}_s[\mathbf{r}(k) - \mathbf{y}_{s,k}]$$

subject to

$$(\mathbf{I} - \mathbf{\Phi}_s)\mathbf{X}_{s,k} = \mathbf{\Gamma}\mathbf{u}_{s,k} + \mathbf{\Gamma}_\beta\boldsymbol{\beta}(k/k-1),$$
$$\mathbf{y}_{s,k} = \mathbf{C}\mathbf{X}_{s,k} + \mathbf{C}_\eta\boldsymbol{\eta}(k/k-1),$$
$$\mathbf{u}^{\mathrm{L}} \leqslant \mathbf{u}_{s,k} \leqslant \mathbf{u}^{\mathrm{H}},$$

where $\mathbf{Q}_s$ is a positive definite matrix. The inclusion of the last term in the objective function (i.e., terminal weighting) ensures closed loop stability of MPC controller under nominal conditions [5]. In the above formulation, $\mathbf{W}_E$ represents (positive definite) error weighting matrix and $\mathbf{W}_U$ represents (positive semi-definite) input move weighting matrix, respectively. The desired closed loop performance can be achieved by judiciously selecting the prediction horizon p, control horizon q and the weighting matrices $\mathbf{W}_E$, $\mathbf{W}_U$ and $\mathbf{W}_S$ (refer [5,7] for a detailed discussion). The resulting constrained optimization problem can be further transformed into an equivalent quadratic programming (QP) formulation for improving the on-line computational efficiency. The controller is implemented in a moving horizon framework. Thus, after solving the optimization problem, only the first move $\mathbf{u}_{\mathrm{opt}}(k/k)$ is implemented on the plant, i.e.,

$$\mathbf{u}(k) = \mathbf{u}_{\mathrm{opt}}(k/k),$$

and the optimization problem is reformulated at the next sampling instant based on the updated information from the plant.

4. Fault tolerant control

In this section, we develop a fault tolerant control scheme (FTCS) similar to the FTCS proposed by Prakash et al. [12] using the innovations form of state space model (2.3) and (2.4). Before we describe the fault diagnosis methodology, we would like to clarify that the term *fault* used by us implies a *root cause fault*. A root cause fault may propagate and cause other dependent or output variables to deviate from their nominal values. The effects of a root cause fault represent its signature, which is effectively used to identify it in our fault diagnosis method and not termed as faults in our work. Also, we assume that multiple root cause faults can occur sequentially in time but not at the same time instant.

The most crucial component of an FTCS is the fault diagnosis scheme. While there are several observer based fault isolation approaches available in the literature [10,11], such as parity space or structured residuals, we choose to work with the GLR method as it is based on the Kalman filter. In addition to performing fault isolation, this approach generates maximum likelihood estimates of the fault magnitude. This capability is of vital importance for achieving fault tolerance. On-line parameter estimation is another effective method for fault identification. The key difference between the GLR based FDI method and parameter estimation based FDI method is that in the former approach the fault magnitude (corresponding to the identified fault) is estimated only after a fault occurrence is confirmed, whereas in the latter approach the fault magnitudes of all hypothesized faults are simultaneously estimated regardless of whether these faults are present or not. Due to this, the number of faults that can be handled in the parameter based FDI approach is limited.

It should be noted that the FDI component in this formulation has limited diagnostic capabilities when compared to the scheme proposed by Prakash et al. [12]. If a model of the process is developed from first principles, then it is possible to do root cause analysis of changes in the unmeasured disturbances and small magnitude parametric drifts and use the diagnostic information in a fault tolerant control scheme as demonstrated by Prakash et al. [12]. However, the problem of diagnosing multiplicative faults using identified models is not adequately addressed in the FDI research literature. We have, therefore, limited the scope of this paper to the treatment of soft faults in sensors and actuators. In the following sections, we discuss the reformulation of the FTCS necessary for this purpose.

4.1. Models for normal and faulty behavior

To begin with, note that we differentiate between the true plant input vector, $\mathbf{u}_T(k)$, and the manipulated input vector, $\mathbf{u}(k)$, while dealing with the fault diagnosis problem. Thus, in the context of FDI, the vector $\mathbf{u}(k)$ denotes the computed input signal generated by the control algorithm. Under the fault free situation, we have

$$\mathbf{u}_T(k) = \mathbf{u}(k). \tag{4.1}$$

A model based diagnosis scheme requires models of the process under normal as well as faulty operating conditions. A model for the normal (fault free) behavior of the process can be obtained using the identification scheme described in Part I [20] and is given by Eqs. (2.1)–(2.2). The fault models for sensor and actuator biases are developed as linear additive step changes to the above normal model. If a bias of magnitude $b_{y,i}$ occurs at instant t in the ith sensor, then the measured output is given as

$$\mathbf{y}(k) = \mathbf{CX}(k) + b_{y,i}e_{y,i}\sigma(k-t) + \mathbf{v}(k), \tag{4.2}$$

where $e_{y,i}$ represents the unit vector with only the ith element equal to unity and other elements equal to zero. Also, $\sigma(k-t)$ represents the unit step function such that

$$\sigma(k-t) = \left\{ \begin{array}{ll} 0 & \text{for } k < t, \\ 1 & \text{for } k \geqslant t. \end{array} \right\} \tag{4.3}$$

Similarly, if a bias of magnitude $b_{u,i}$ occurs at instant t in the ith actuator, then the true plant input $\mathbf{u}_T(k)$ is given as

$$\mathbf{u}_T(k) = \mathbf{u}(k) + b_{u,i}e_{u,i}\sigma(k-t). \tag{4.4}$$

4.2. Fault identification using the GLR method

In this subsection, the FDI scheme is explained as it is applied once and for the first time. Modifications necessary to apply it repeatedly on-line are discussed in the next sub-section. The on-line FDI method has to provide the following diagnostic information for implementing the proposed active fault tolerant control scheme

- Detect whether a fault has occurred and the time instant it has occurred,
- Identify the type of fault (and estimate its magnitude).

Under fault-free situation the residual or innovation vector $\mathbf{e}(k)$ is a zero mean Gaussian white noise process with covariance matrix $\mathbf{V}_e$. A sequential strategy is adopted for fault detection, confirmation and subsequent identification. A simple statistical test namely, a fault detection test (FDT) based on innovations [12] is applied at each time instant to estimate the time of occurrence of a fault. The test statistic used for this purpose is given by

$$\epsilon(k) = \mathbf{e}(k)^{\mathrm{T}}\mathbf{V}_e^{-1}\mathbf{e}(k) > \tau. \tag{4.5}$$

The test statistic follows a central chi-square distribution with n degrees of freedom, which can be used to fix the test criterion τ. A fault is suspected to have occurred at a time, say 't', if the test statistic exceeds the test criterion. To further confirm the occurrence of a fault, a fault confirmation test (FCT) is applied over a window of N sampling instants by making use of the innovation sequence over the time interval $[t, t+N]$ using the test statistic

$$\epsilon(N;t) = \sum_{t}^{t+N}\mathbf{e}(k)^{\mathrm{T}}\mathbf{V}_e^{-1}\mathbf{e}(k) > \delta. \tag{4.6}$$

The above test statistic follows a central chi-square distribution with the $n \times (N+1)$ degrees of freedom, which can be used to fix the test criterion δ. Occurrence of a fault at time t is confirmed at $(t+N)$ if the test statistic exceeds the test criterion [12].

Once the FCT confirms the occurrence of fault, a statistical test based on the generalized likelihood ratio (GLR) method [13] is used to identify the fault location and estimate its magnitude. Each isolable fault type has its characteristic innovation trend, which can be captured via a signature matrix. For example, if a fault of magnitude $b_{f,i}$ occurs at time t, the expected values of the innovations at any subsequent time after the occurrence of the fault can be represented as

$$\mathbf{E}[\mathbf{e}(k)] = b_{f,i}\mathbf{G}_f(k;t)e_{f,i}, \quad k \succeq t, \tag{4.7}$$

where subscript $f \in \{y,u\}$ denotes the fault type; $e_{f,i}$ is a unit vector with the ith component equal to one and rest equal to zero, representing the index of sensor or actuator that is faulty. Here, the matrix $\mathbf{G}_f(k;t)$ is referred to as the signature matrix and depends upon the time t at which a fault has occurred and the time k at which the innovations are computed. Similarly, defining the error in state estimation as

$$\delta\mathbf{X}(k) = \widehat{\mathbf{X}}(k/k-1) - \mathbf{X}(k), \tag{4.8}$$

where $\mathbf{X}(k)$ represents the *true value* of the state vector, the expected values of state error after occurrence of a fault can be expressed as

$$E(\delta\widehat{\mathbf{X}}(k)) = b_{f,i}\mathbf{J}_f(k;t)\mathbf{e}_{f,i}. \tag{4.9}$$

It is possible to derive the recurrence relationship for the above signature matrices using the state estimator under fault free conditions (2.8) and (2.9) and the fault models (refer [22,12] for details of derivation). The resulting expressions in the present case are as follows:

- *Sensor bias*

$$\mathbf{G}_y(k;t) = \mathbf{I} - \mathbf{C\Phi J}_y(k-1;t) - \mathbf{CKG}_y(k-1;t), \tag{4.10}$$

$$\mathbf{J}_y(k;t) = \mathbf{\Phi J}_y(k-1;t) + \mathbf{KG}_y(k-1;t). \tag{4.11}$$

- *Actuator bias*

$$\mathbf{G}_u(k;t) = -\mathbf{C\Phi J}_u(k-1;t) + \mathbf{C\Gamma} - \mathbf{CKG}_u(k-1;t), \tag{4.12}$$

$$\mathbf{J}_u(k;t) = \mathbf{\Phi J}_u(k-1;t) - \mathbf{\Gamma} + \mathbf{KG}_u(k-1;t). \tag{4.13}$$

Define scalars

$$T_{f,i} = \frac{d_{f,i}^2}{c_{f,i}}, \tag{4.14}$$

$$d_{f,i} = e_{f,i}^{\mathrm{T}} \sum_{k=t}^{t+N} \mathbf{G}_f^{\mathrm{T}}(k;t)\mathbf{V}_e(k)^{-1}\mathbf{e}(k), \tag{4.15}$$

$$c_{f,i} = e_{f,i}^{\mathrm{T}} \left[\sum_{k=t}^{t+N} \mathbf{G}_f^{\mathrm{T}}(k;t)\mathbf{V}_e(k)^{-1}\mathbf{G}_f(k;t) \right] e_{f,i}. \tag{4.16}$$

The fault that corresponds to maximum value of $T_{f,i}$ is identified as the fault that has occurred at time t. The maximum likelihood estimate of the fault is given by

$$\hat{b}_{f,i} = \frac{d_{f,i}}{c_{f,i}}. \tag{4.17}$$

In order to detect the time of occurrence of a fault quickly and accurately, a low threshold value is used in FDT. This has the disadvantage that the FDT can give rise to many false alarms. The fault confirmatory test (FCT) is used after a time window N based on the entire information (residuals) in this time window, to weed out such false alarms. Moreover, fault identification cannot be performed at the same instant a fault is detected because there is insufficient information for this purpose. The effect of the fault (fault signature) has to manifest itself through a distinct pattern of residuals before diagnosis can be performed. Although, a large value of N causes a delay in fault identification, choosing it to be small can result in incorrect diagnosis. Furthermore, the kind of fault and its impact on the process/control performance is also a criterion on how fast we want the identification. The soft faults that we are considering in this work will not have a disastrous effect if not isolated quickly. On the other hand, hard faults like sensor and actuator failures, leaks in vessels and pipelines need to be identified promptly. Based on these considerations, we have chosen the time window to be about half the settling time of the estimator to a step change, as recommended by Prakash et al. [12].

4.3. On-line fault accommodation

The estimated bias magnitude is used for compensating the biased sensor measurement as given below

- If a bias in the ith sensor is identified, a compensated output is computed as

$$\tilde{b}_{y,i} = \sum_{j=1}^{n_{y,i}} \hat{b}_{y,i,j}, \tag{4.18}$$

$$\mathbf{y}_c(k) = \mathbf{y}(k) - \sum_{i=1}^{r} \tilde{b}_{y,i}\mathbf{e}_{y,i}, \tag{4.19}$$

  where $\tilde{b}_{y,i}$ represents the cumulative estimate of the magnitude of the ith sensor bias, $n_{y,i}$ represents number of times a sensor fault was detected in ith position and $\mathbf{y}_c(k)$ represents the compensated or corrected measurement vector. This compensated value of measurements is used for state estimation as well as controller computations subsequent to detection and identification of sensor fault(s) instead of using the actual measurement $y(k)$.
- If a bias in actuator is identified, the compensated controller output is computed as

$$\tilde{b}_{u,i} = \sum\nolimits_{j=1}^{n_{u,i}} \hat{b}_{u,i,j}, \tag{4.20}$$

$$\mathbf{u}_c(k) = \mathbf{u}(k) + \sum_{i=1}^{m} \tilde{b}_{u,i}\mathbf{e}_{u,i}, \tag{4.21}$$

  where $\tilde{b}_{u,i}$ is the cumulative estimate of the magnitude of the actuator bias obtained using the FDI scheme. This value is used in the state estimator for all instants subsequent to identification of an actuator bias, i.e., Eq. (2.9) is modified as

$$\widehat{\mathbf{X}}(k+1/k) = \mathbf{\Phi}\widehat{\mathbf{X}}(k/k-1) + \mathbf{\Gamma u}_c(k) + \mathbf{Ke}(k), \quad k \geqslant t+N. \tag{4.22}$$

- In addition, each time an occurrence of a fault is confirmed, the state estimates are corrected to account for the delay of N time instants in identifying the fault after it has occurred (see [12]).

Estimating states based on compensated measurements/manipulated inputs ensures that the innovation sequence, after carrying out the on-line compensation, again has zero mean and any other fault that may occur subsequently can be identified. The use of cumulative bias estimates for on-line correction enables the FDI scheme with a certain degree of self-correcting ability and provides a safeguard against false isolation and inaccurate magnitude estimation [12].

4.4. Simulation technique and performance measures

The performance of the FTCS is assessed through stochastic simulations. In the simulation procedure used in this work, a simulation run consisting of N_T trials is conducted for each case being investigated. Each simulation trial is carried out for a time interval approximately equal to 10 times the closed loop setting time. The performance measures used for evaluating control and FDI performances are as follows (see [12]):

- *Integral square error* (ISE): For controlled variable i, the ISE is defined as

$$\text{ISE} = \sum_{k=1}^{L_s} (\mathbf{y}_{t,i}(k) - \mathbf{y}_{r,i}(k))^2, \tag{4.23}$$

where $\mathbf{y}_{t,i}(k)$ represents the true value of the controlled variable, $\mathbf{y}_{r,i}(k)$ its setpoint and L_s is the length of a simulation trial. For each simulation trial, a performance index (PI) is defined as the ratio of ISE under closed loop control with fault compensation over the ISE under closed loop control without fault compensation, that is,

$$(\text{PI})_i = \frac{(\text{ISE})_i(\text{fault tolerant scheme})}{(\text{ISE})_i(\text{conventional control scheme})}.$$

- *Percentage of successful trials* (PST): is the percentage of successful trials in a simulation run, where a successful trial is one in which the introduced fault is correctly isolated at least once during the trial.

$$\text{PST} = \frac{\text{number of successful trials}}{N_T} \times 100.$$

- *False alarm index* (FAI): is the ratio of the average number of false alarms generated per simulation trial to the number of window lengths in a trial:

$$\text{FAI} = \frac{\text{number of false alarms in } N_T \text{ trials}}{(L_s/N) \times N_T}.$$

4.5. Comparison between MPC and fault tolerant control

It is instructive to consider the differences in the mechanism used in the MPC formulation to account for plant-model mismatch versus the Fault tolerant control approach. Both approaches attempt to 'compensate' for the fact that deterministic changes (caused either by changes in unmeasured disturbances and parameters or by biases in sensors and actuators) can occur in an operating process. The approach taken in the MPC formulation is an ad-hoc one, without any attempt to assign a cause for the observed plant model mismatch. A conservative approach is taken by augmenting the system with artificial states, which help to eliminate offset between measured values and set point. This conservative approach results in some control performance giveaway during normal operation. Furthermore, in the presence of sensor biases, the method gives rise to an offset between the true values of controlled outputs and their setpoints.

On the other hand, the fault tolerant control strategy is an active approach which corrects the system as and when required depending on the fault that has occurred. Since there are no permanent augmented states, there is theoretically no limit to the number of faults that can be handled in this approach. Furthermore, this does not lead to any deterioration in control performance during normal operation. The disadvantage of this approach is that in a stochastic process the FDI method can result in false alarms and incorrect fault diagnosis, which may result in performance degradation. It should also be pointed out that in the fault tolerant control procedure we have described, the FDI strategy has been designed to detect and identify sensor and actuator faults and not faults caused by changes in unmeasured disturbances or parameters (which will be handled as in conventional MPC using a step type disturbance prediction and correction).

5. Simulation studies using shell control problem (SCP)

As described in [20], the heavy oil fractionator system in SCP has following configuration:

- *Measured outputs*: Top end point (y_1), side end point (y_2), bottom reflux temperature (y_3).
- *Manipulated inputs*: Top draw (u_1), side draw (u_2), bottom reflux duty (u_3).
- *Unmeasured disturbances*: Internal reflux duty (d_1), upper reflux duty (d_2).

The models used for simulating the plant behavior, unmeasured disturbance dynamics and measurement noise are given in the accompanying paper [20]. The minimal order state space model (2.1)–(2.2) identified using GOBF parameterization (see [20] for details) is used for developing the MPC controllers and a GLR based fault diagnosis scheme in the simulation studies.

5.1. Model predictive control

In this section, we compare the regulatory performance of the proposed MPC scheme based on the identified state space model (referred to as SSMPC in the rest of the text) and the conventional MPC controller (referred to as CMPC in the rest of the text) with DMC type simple disturbance prediction model as described in [21]. In order to carry out the comparative

Table 1
Uncertainties in the steady state gains of shell control problem model

	u_1	u_2	u_3	d_1	d_2
y_1	$4.05 + 2.11\varepsilon_1$	$1.77 + 0.39\varepsilon_2$	$5.88 + 59\varepsilon_3$	$1.2 + 0.12\varepsilon_4$	$1.44 + 0.16\varepsilon_5$
y_2	$5.39 + 3.29\varepsilon_1$	$5.72 + 0.57\varepsilon_2$	$6.9 + 0.89\varepsilon_3$	$1.52 + 0.13\varepsilon_4$	$1.83 + 0.13\varepsilon_5$
y_3	$4.38 + 3.11\varepsilon_1$	$4.42 + 0.73\varepsilon_2$	$7.2 + 1.33\varepsilon_3$	$1.14 + 0.18\varepsilon_4$	$1.26 + 0.18\varepsilon_5$

evaluation, we consider the following specific regulatory control objectives of the shell control problem:

- Reject unmeasured disturbances in upper and intermediate reflux duties due to changes in heat duty requirements in other columns.

In the SCP, the controller is expected to reject the disturbances for the nominal plant as well as in the presence of uncertainties in process gains. The uncertainties in process gain parameters are given in Table 1, where $-1 \leqslant \varepsilon_i \leqslant 1$ $(i = 1, 2, \ldots, 5)$. Since it is impossible to come up with all possible combinations of load disturbances and uncertainty parameters, the SCP comes with five prototype cases which are supposed to represent most relevant uncertainty/disturbance combinations [6]:

- Case 1: No model uncertainty $d_1 = 0.5$, $d_2 = 0.5$.
- Case 2: $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = -1$, $\varepsilon_4 = \varepsilon_5 = 1$, $d_1 = -0.5$, $d_2 = -0.5$.
- Case 3: $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \varepsilon_4 = 1$, $\varepsilon_5 = -1$, $d_1 = -0.5$, $d_2 = -0.5$.
- Case 4: $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \varepsilon_4 = \varepsilon_5 = 1$, $d_1 = 0.5$, $d_2 = -0.5$.
- Case 5: $\varepsilon_1 = -1$, $\varepsilon_2 = 1$, $\varepsilon_3 = \varepsilon_4 = \varepsilon_5 = 0$, $d_1 = -0.5$, $d_2 = -0.5$.

Note that the plant model mismatch in Cases 2–4 is introduced not only in the deterministic component but also in the unmeasured disturbance model.

The anticipated range of variation for unmeasured disturbances is

$$-0.5 \leqslant d_i(k) \leqslant 0.5 \quad \text{for } i = 1, 2.$$

Since we have introduced stochastic disturbances in variables d_1 and d_2 in the present work as discussed in the accompanying paper [12], we modify the step magnitudes of d_1 and d_2 in the above five cases so as to meet the constraints $|d_i(k)| \leqslant 0.5$. Thus, the step magnitude 0.5 is replaced by 0.425 and the step magnitude -0.5 is replaced by -0.425 in the simulation runs. This ensures that the combined step and stochastic disturbances are within the constraint limits (see Fig. 3).

In order to make a fair comparison between performances of the SSMPC and the CMPC, the following measures are taken:

- The low order state space model developed in the accompanying paper is used for formulation of the SSMPC as well as the CMPC controllers. Thus, the deterministic components of the models used for the SSMPC and CMPC formulation are identical. The only difference between the two formulations is the model used for dealing with unmeasured disturbances.
- The tuning parameters for SSMPC and CMPC are identical and are taken from Yu et al. [6]:

$$p = 40, \quad q = 10, \quad \mathbf{W}_S = [\mathbf{0}],$$
$$\mathbf{W}_E = \text{diag}[1 \quad 1 \quad 0],$$
$$\mathbf{W}_U = \text{diag}[1.5 \quad 0.15 \quad 1.5].$$

- Identical realizations of white noise sequences are used for simulating unmeasured disturbance behavior and measurement noise in all the simulation runs.
- The control constraints employed in both SSMPC and CMPC formulations are as follows:

$$-0.5 \leqslant u_i(k + j/k) \leqslant 0.5$$
$$\text{for } i = 1, 2, 3 \text{ and } j = 0, 1, 2, \ldots, q - 1,$$
$$-0.05 \leqslant \Delta u_i(k + j/k) \leqslant 0.05$$
$$\text{for } i = 1, 2, 3 \text{ and } j = 0, 2, \ldots, q - 1,$$
$$-0.5 \leqslant y_1(k + j/k) \leqslant 0.5$$
$$\text{for } j = 1, 2, \ldots, p,$$
$$-0.5 \leqslant y_3(k + j/k) \quad \text{for } j = 1, 2, \ldots, p.$$

In order to compare the performances of the SSMPC and the CMPC controllers, we define following the performance index based on the objective functions in the MPC formulations [23]:

$$J = \sum_{k=1}^{N_s} \Big\{ [\mathbf{r}(k) - \mathbf{y}(k)]^{\mathrm{T}} \mathbf{W}_E [\mathbf{r}(k) - \mathbf{y}(k)] + \Delta\mathbf{u}(k)^{\mathrm{T}} \mathbf{W}_U \Delta\mathbf{u}(k) \Big\}. \tag{5.1}$$

To begin with, we investigate the following three possibilities for introducing artificial states in the augmented model (3.4)–(3.7) required in the proposed MPC formulation:

- *SSMPC-A:* Output bias formulation

$$\boldsymbol{\Gamma}_\beta = [0], \quad \mathbf{Q}_\beta = [0], \quad \mathbf{C}_\eta = I, \quad \mathbf{Q}_\eta = 10^{-4} I.$$

- *SSMPC-B:* Input bias formulation

$$\boldsymbol{\Gamma}_\beta = \boldsymbol{\Gamma}, \quad \mathbf{Q}_\beta = 10^{-4} I, \quad \mathbf{C}_\eta = [0], \quad \mathbf{Q}_\eta = [0].$$

Table 2
Shell control problem: comparison of regulatory performances of SSMPC and CMPC

Controller	max$|y_1|$	max$|y_2|$	J
CMPC	0.43	0.74	11.111
SSMPC-A	0.36	0.75	8.665
SSMPC-B	0.21	0.50	3.151
SSMPC-C	0.13	0.27	0.961

- *SSMPC-C:* Innovation bias formulation (similar to [3])

$$\Gamma_\beta = \mathbf{K}, \quad \mathbf{Q}_\beta = 10^{-4}I, \quad \mathbf{C}_\eta = I, \quad \mathbf{Q}_\eta = 10^{-4}I.$$

The diagonal elements in the tuning matrix $\mathbf{Q}_\beta$ were selected to be commensurate with the diagonal values in the innovation covariance matrix $\mathbf{V}_e$. As shown in Fig. 3, the step changes in d_1 and d_2 are introduced at the 50th sampling instant. Table 2 compares regulatory performances of the CMPC and the three SSMPC formulations for Case 1 (no model uncertainty).

The results for SSMPC-A clearly indicate that the inclusion of unmeasured disturbance model improves the regulatory performance over performance of the CMPC controller. Note that the manner in which additional states are introduced also has a significant influence on the regulatory performance. SSMPC-C was found to generate the best regulatory performance. This could be attributed to the fact that the Kalman gain matrix accounts for the effects of unmeasured disturbances on the state variables. Thus, innovation bias formulation appears to be better suited for dealing with mean shift in the unmeasured disturbances. However, it was observed that the SSMPC-C formulation produces highly oscillatory input behavior and is less robust to model parameter changes. The SSMPC-B formulation, on the other hand performed satisfactorily for all the test cases reported above. The comparison of SSMPC-B and CMPC for Case 1 is presented in Figs. 1–3. As evident from Fig. 1, the SSMPC-B formulation is able to reject the disturbance with much smaller deviations from the setpoints. The corresponding variations in manipulated inputs is also relatively less for the CMPC controller (Fig. 2). For all other cases, SSMPC-B is chosen for further comparison with CMPC.

The comparison of the performances of CMPC and SSMPC-B for the remaining four test cases listed in Table 1 is given in Table 3. (Note that the CMPC controller failed to generate solution within the specified constraints for the Case 3.) It is observed that the performance of SSMPC-B controller for Cases 2, 3 and 5 is clearly superior to the performance of the conventional MPC controller in all aspects. The performances of CMPC and SSMPC-B for the Case 4 are comparable, though CMPC performs slightly better in terms of J value. Thus, the above simulation study indicates that the SSMPC-B formulation based on identified unmeasured disturbance model has a clear advantage over CMPC formulation even in presence of severe plant model mismatch.

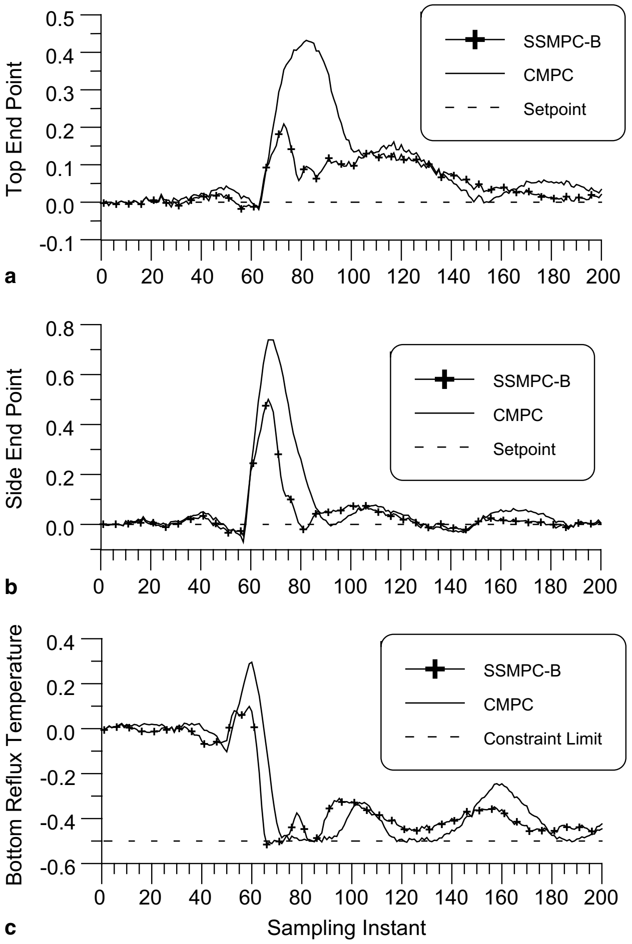

Fig. 1. Shell control problem: disturbance rejection using SSMPC-B and CMPC controllers—comparison of output responses.

5.2. Fault tolerant control

The performance of the FTCS is evaluated using stochastic simulation studies on SCP. The process is controlled using the CMPC controller. While performing comparative study, we consider the following two cases:

- Conventional control: process is controlled using the CMPC controller,
- FTCS: process is controlled using the CMPC controller together with GLR based FDI and on-line fault compensation scheme.

The tuning parameters for the CMPC controller and operating constraints are given in the above section. The output variances when $u(k) = \underline{0}$ (i.e., variance of signal $v_i(k)$ defined by Eq. (11) in the accompanying paper [20])

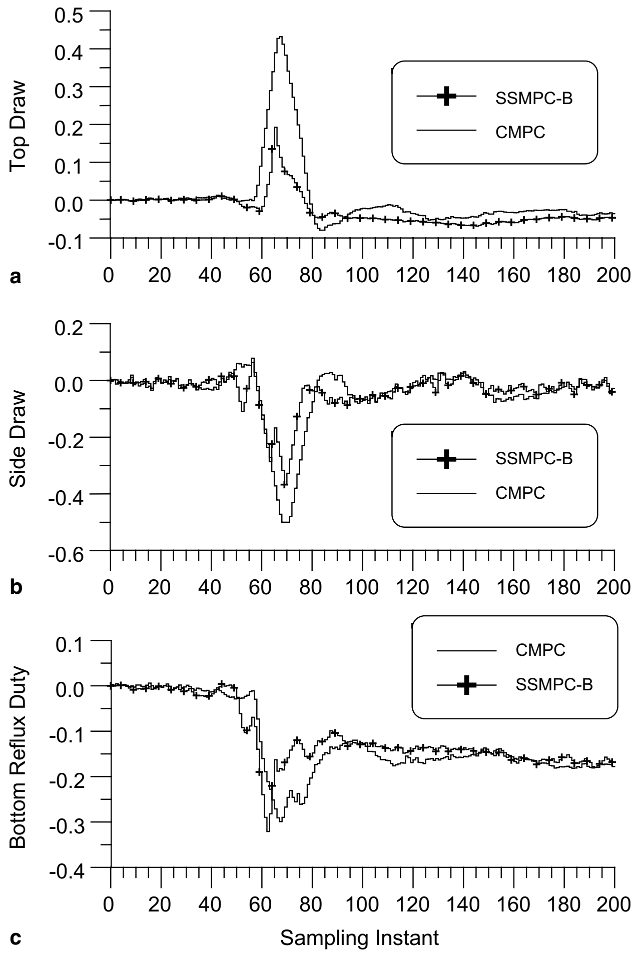

Fig. 2. Shell control problem: disturbance rejection using SSMPC-B and CMPC controllers—comparison of manipulated inputs.

$$\hat{\sigma}_{v_1} = 0.029, \quad \hat{\sigma}_{v_2} = 0.042, \quad \hat{\sigma}_{v_3} = 0.028,$$

are used as the basis for introducing sensor bias magnitudes. Similarly, standard deviations of manipulated inputs in response to random fluctuations in the unmeasured disturbances under fault free conditions are used as basis for selecting input bias magnitudes

$$\hat{\sigma}_{u_1} = 0.0058, \quad \hat{\sigma}_{u_2} = 0.026, \quad \hat{\sigma}_{u_3} = 0.012.$$

The parameters of the stochastic simulation study are given in Table 4.

The performance measures used to analyze the simulation results are given in Section 4.4 (also see [12] for detailed discussion).

The sensor bias is introduced at the first sampling instant in every simulation trial. Table 5 presents simulation results (averaged over a simulation run) for bias in the first sensor.

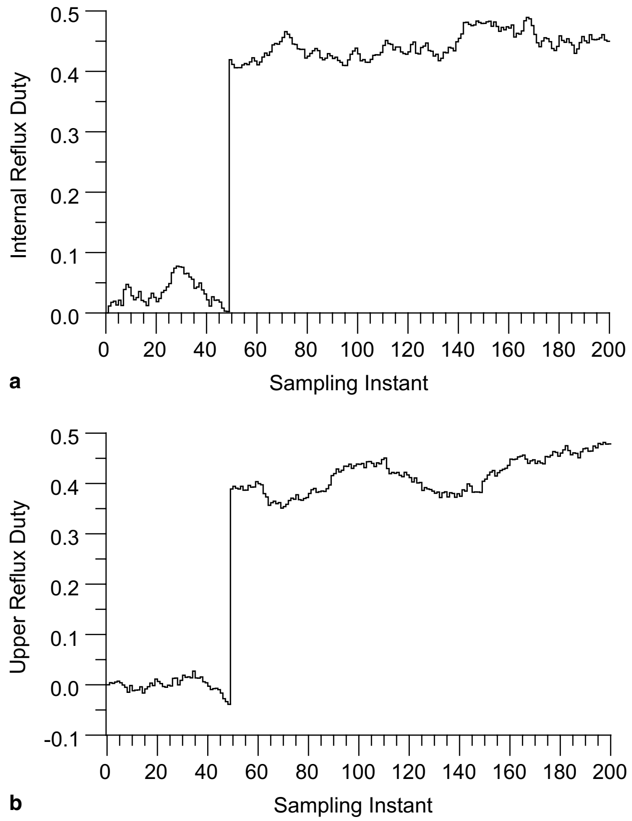

Fig. 3. Shell control problem: disturbance rejection using SSMPC-B and CMPC controllers—unmeasured disturbance profiles.

Table 3
Shell control problem: regulatory performances of SSMPC-B and CMPC

Case	Controller	max$\|y_1\|$	max$\|y_2\|$	J
Case 2	CMPC	0.471	1.211	22.451
	SSMPC-B	0.293	0.805	10.591
Case 3	CMPC	–	–	–
	SSMPC-B	0.308	0.797	9.149
Case 4	CMPC	0.147	0.137	1.104
	SSMPC-B	0.150	0.135	1.356
Case 5	CMPC	0.442	0.799	11.871
	SSMPC-B	0.209	0.490	3.8253

Table 4
Shell control problem: simulation parameters of FTCS

Variable description	Value
Simulation time, L_s	500 sampling instants
Number of trials, N_T	100
Window length, N	50 sampling instants
Level of significance for FDT, α_{FDT}	0.75
Level of significance for FCT, α_{FCT}	0.01

Even for a small magnitude fault, the FDI scheme correctly isolates the fault and false isolations are few. As compensated measurements are used in FTCS by the CMPC after fault has been identified, there is a significant improvement in the closed loop performance of the variable with the biased sensor. This is reflected in the small PI values for y_1 in Table 5.

Table 5
Shell control problem FTCS performance in presence of bias in first sensor

b_1	$\hat{b}_{y,1}$ ($\sigma_{\hat{b}_{y,i}}$)	PST	FAI	$(\mathrm{PI})_{y_1}$ ($\sigma_{(\mathrm{PI})_{y_1}}$)	$(\mathrm{PI})_{y_2}$ ($\sigma_{(\mathrm{PI})_{y_2}}$)	$(\mathrm{PI})_{y_3}$ ($\sigma_{(\mathrm{PI})_{y_3}}$)
0.029	0.026 (0.002)	100	0.002	0.358 (0.069)	1.0003 (0.005)	0.979 (0.068)
0.087	0.078 (0.005)	100	0.008	0.148 (0.023)	1.0040 (0.015)	0.917 (0.117)

Table 6
Shell control problem: FDI performance in the sequential fault case

b_u	b_y	$\hat{b}_u$ ($\sigma_{\hat{b}_u}$)	$\hat{b}_y$ ($\sigma_{\hat{b}_y}$)	PST		FAI
−0.0174	0.042	−0.0141 (0.005)	0.0284 (0.006)	51	98	0.029
−0.0290	0.042	−0.0270 (0.007)	0.0330 (0.007)	99	100	0.026

Table 7
Shell control problem: control loop performance in the sequential fault case

b_u	b_y	$(\mathrm{PI})_{y_1}$ ($\sigma_{(\mathrm{PI})_{y_1}}$)	$(\mathrm{PI})_{y_2}$ ($\sigma_{(\mathrm{PI})_{y_2}}$)	$(\mathrm{PI})_{y_3}$ ($\sigma_{(\mathrm{PI})_{y_3}}$)
−0.0174	0.042	1.021 (0.111)	0.653 (0.148)	1.022 (0.118)
−0.029	0.042	1.003 (0.026)	0.646 (0.124)	1.069 (0.116)

In order to simulate faults occurring sequentially in time, a bias in one of the actuators was introduced at first sampling instant followed by bias introduced in the second sensor at 250th sampling instant in each simulation trial. The summary of results averaged over simulation run for sequential fault case are presented in Tables 6 and 7. Table 6 shows that small magnitude actuator biases are more difficult to detect and the number of successful trials in which the actuator fault is correctly isolated is relatively low. The FDI performance, however, improves with increase in the magnitude of actuator bias. Note that the simulation results in the present case are qualitatively similar to the results obtained by Prakash et al. [12] using a first principles model for actuator and sensor bias cases. Also, the simulation results presented above are typical results and representative of simulation results obtained for faults in other sensors and actuators. Thus, the FTCS reformulated using the identified innovations form of state space model is able to isolate faults in sensors as well as actuators.

Robustness (or insensitivity) to the unmeasured disturbances is an important factor while developing any FDI scheme. As evident from the above results, the proposed FTCS is robust to the stationary unmeasured disturbances, which have already been captured by the identified noise model. However, the performance of the FDI component can deteriorate in the presence of step jumps (mean shifts) in the unmeasured disturbances. In order to investigate the response of the FTCS scheme to abrupt changes in the unmeasured disturbances, we have carried out simulation studies by introducing a step jump at $k = 1$ in the direction of Case 1 above, i.e.,

$$[\,d_1 \quad d_2\,]^{\mathrm{T}} = \alpha[\,1 \quad 1\,].$$

Since step jumps in disturbances directly affect the state dynamics, it may be expected that these changes will be classified as input biases. The results of stochastic simulations are presented in Tables 8 and 9. Note that these results are typical and the simulation results obtained for step jumps in other directions are qualitatively similar.

It is observed that FAI increases as the magnitude of the disturbance changes increases. This can be explained by the fact that the hypotheses set does not include step changes in unmeasured disturbances as faults and thus every detection and isolation is classified as a false alarm. For small magnitude disturbances ($\alpha = 0.05$), it was found that no fault is detected for a long time after the introduction of the step change. However, for the large magnitude disturbances ($\alpha = 0.1, -0.15$), fault detection is immediate and, as can be expected, the fault is classified as an actuator bias. Subsequently, sensor faults of small magnitude get isolated till the innovation sequence becomes a white noise. The increase in FAI, however, does not directly indicate the performance of the FTCS. Unless the step change in unmeasured disturbance is classified as a sensor bias of significant magnitude, the FTCS performance will be comparable to a

Table 9
Shell control problem: controller performance in presence of step jumps in unmeasured disturbances

α	$(\mathrm{PI})_{y_1}$ ($\sigma_{(\mathrm{PI})_{y_1}}$)	$(\mathrm{PI})_{y_2}$ ($\sigma_{(\mathrm{PI})_{y_2}}$)	$(\mathrm{PI})_{y_3}$ ($\sigma_{(\mathrm{PI})_{y_3}}$)
0.05	0.9604 (0.0877)	0.9678 (0.0789)	1.0342 (0.0696)
0.10	0.8770 (0.1306)	0.9548 (0.1102)	1.0780 (0.0967)
−0.15	0.8287 (0.1053)	0.9570 (0.1100)	1.0882 (0.0943)

Table 8
Shell control problem: FDI performance in presence of step jumps in unmeasured disturbances

α	FAI	$\hat{b}_{u1}$ ($\sigma_{\hat{b}_{u_1}}$)	$\hat{b}_{y_1}$ ($\sigma_{\hat{b}_{y_2}}$)	$\hat{b}_{y_2}$ ($\sigma_{\hat{b}_{y_2}}$)	$\hat{b}_{y_3}$ ($\sigma_{\hat{b}_{y_3}}$)
0.05	0.306	0.0189 (0.0069)	0.0041 (0.0067)	0.0031 (0.0070)	−0.0205 (0.0083)
0.10	0.543	0.0484 (0.0083)	0.0159 (0.0103)	0.0050 (0.0103)	−0.0396 (0.0091)
−0.15	0.660	−0.0790 (0.0074)	−0.0265 (0.0088)	−0.0058 (0.0134)	0.0614 (0.0110)

conventional controller. It should be noted that the estimated sensor bias magnitudes are of relatively small. In fact, the estimated magnitudes of bias in the first two measured variables are less than one standard deviation of the noise in corresponding variables, and only in the case of the third measured variable is the estimated bias magnitude significant (roughly three times the standard deviation of noise, for the largest step change in disturbance). Thus, the controller performance indices for all outputs are close to unity (Table 9) and there is no significant degradation in the closed loop performance due to these false isolations.

If a model of the process is developed from the first principles or if the *disturbance coupling matrix* relating the physical disturbance variables with state dynamics is known a priori, then it is possible to make the FDI scheme (and FTCS) robust to sudden changes in the unmeasured disturbances as demonstrated by Prakash et al. [12]. However, when the disturbances are unmeasured, the disturbance coupling matrix cannot be estimated directly from the input–output data. The development of a robust FDI scheme when the disturbance coupling matrix is unknown has, to the best of our knowledge, not been satisfactorily solved. Some possible approaches to solve this problem have been discussed in the recent book by Chen and Patton [24]. Also, Manuja et al. [25,26] have recently proposed a method of constructing an approximation to this matrix, which provides a systematic approach to dealing with step jumps in unmeasured disturbances when working with identified models.

6. Experimental studies using continuous stirred tank heater (CSTH)

In this section, we present results of experimental validation carried out on a computer-interfaced, pilot scale continuous stirred tank heater. This laboratory scale experimental setup consists of a stirred tank heater in which a hot water stream and a cold water stream are continuously mixed and heated as shown in Fig. 4. The heat necessary to raise the temperature of the water in the tank is supplied by a steam coil, which remains completely immersed in water during the experimental runs. Sensors are available to measure the water level and temperature in the tank and the hot water flow rates. The cold water flow, steam flow and hot water flow rates can be manipulated through the control computer using real-time MATLAB and SIMULINK. The level loop is assumed to be closed a priori as shown in Fig. 4. The level is controlled using the following PI controller:

$$g_{c1}(s) = 3 + \frac{0.1}{s}, \tag{6.1}$$

which manipulates the inlet cold water flow rate. The hot water inflow (F_H) is treated as an unmeasured disturbance while carrying out system identification. Note that the hot water inflow (F_H) is measured and can be changed from the control computer by changing the valve position of control valve CV-3 on the hot water line (see Fig. 4). A PI controller

$$g_{c2}(s) = 0.1\left(1 + \frac{1}{7.5s}\right), \tag{6.2}$$

is used to control the hot water flow to the system as shown in Fig. 4. The unmeasured disturbance signal is generated by changing the setpoint to the hot water flow loop, which is fed as a piecewise constant input signal. The setpoint to the hot water flow control loop was generated as follows:

$$\text{Disturbance model: } \delta F_{H,sp}(k) = \frac{0.3}{1 - 0.95z^{-1}} v(k), \tag{6.3}$$

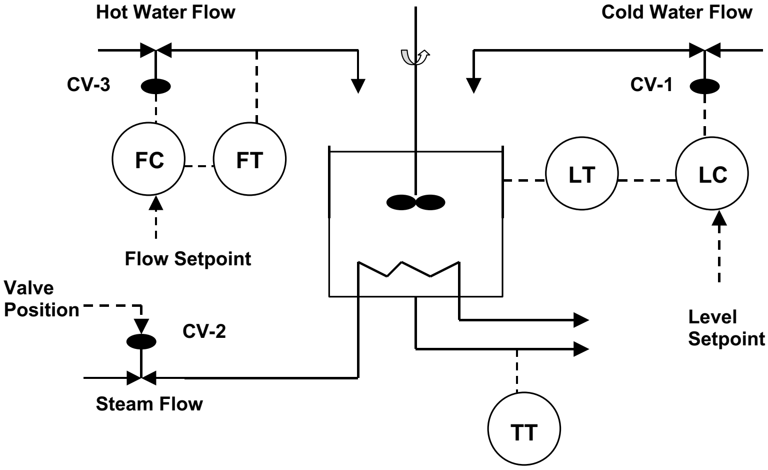

Fig. 4. Stirred tank heater: schematic diagram.

where $\delta F_{H,s}(k) = F_{H,s}(k) - F_{H,sp}$ (in mA) represents setpoint to the hot water flow loop and $v(k)$ represents zero mean Gaussian white noise sequence with $\sigma_v = 1$. The steady state operating setpoint for hot water flow is fixed at $F_{H,sp} = 0.1111$ l/s (7 mA). The sampling time is chosen as 1 s.

The minimal order state space model for this system (2.1)–(2.2) identified using GOBF parameterization (see ([20]) for details) is used for developing the MPC controllers and a GLR based fault diagnosis scheme in the simulation studies. Note that the model development exercise was carried using data generated in the month of July 2001 while the experimental verification of MPC and FTCS schemes were conducted in the month of October 2001. As a consequence, the operating points were different which is reflected in the average steady state steam valve position at the desired steady state operating point

$$\bar{h} = 20.52\ \text{cm}\ (12\ \text{mA}), \quad \bar{T} = 45.2\ ^\circ\text{C}\ (11\ \text{mA}),$$
$$\bar{S}_v = 36.25\%\ (9.8\ \text{mA}) \quad (\text{in July}),$$
$$\bar{S}_v = 48.75\%\ (11.8\ \text{mA}) \quad (\text{in October}).$$

The shift in the operating point also introduced some plant model mismatch in terms of process gains and unmeasured disturbance characteristics. Despite the fact that we have not considered such drifts in unmeasured disturbances and parameter variations as faults, we point out that a controller with integral action is capable of rejecting such faults to a limited extent. Thus, such faults of low severity will not impair the performance of our fault tolerant scheme. This is vindicated by the experimental results which we present below.

6.1. Model predictive control

As one of the main objectives of this work is to compare the disturbance rejection capabilities of CMPC and SSMPC formulations, we have deliberately introduced large magnitude unmeasured disturbances into the system. The effect of these disturbances on the measured outputs when manipulated inputs are held constant at their respective steady state operating points is shown in Fig. 5. As evident from this figure, these disturbances induce large magnitude oscillations in both the measured outputs (±1 cm in level and ±3 °C in temperature) in the absence of any control action. In the regulatory control problem, the CMPC and SSMPC controllers are expected to reject these stationary disturbances as well as step changes in the hot water flow rate.

In order to make a fair comparison of the regulatory performances of the unconstrained CMPC and SSMPC formulations, these controllers were developed using identical prediction horizon (p), control horizon (q), error weighting matrix ($\mathbf{W}_E$) and input move weighting matrix ($\mathbf{W}_U$). These tuning parameters are as follows:

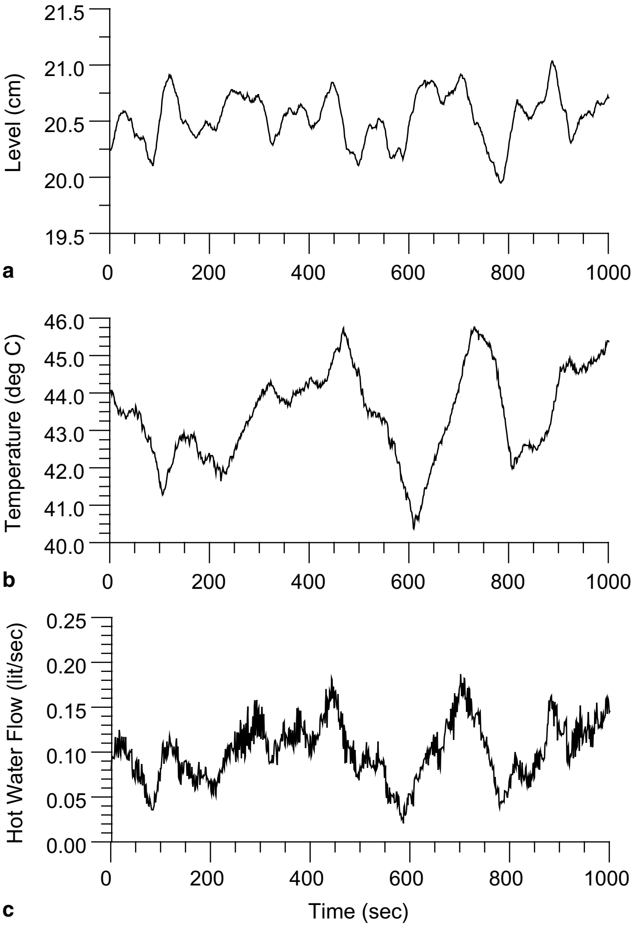

Fig. 5. CSTH: Output variations due to unmeasured disturbances when manipulated input are held constant.

$$p = 150, \quad q = 10, \quad \mathbf{W}_S = [\mathbf{0}],$$
$$\mathbf{W}_E = \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{W}_U = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.6 \end{bmatrix}.$$

The minimal order innovations form of state space model of the form Eqs. (2.1) and (2.2) developed in the accompanying paper [20] was used to develop the SSMPC and the CMPC controllers. Note that the deterministic components of the models used for the development of the SSMPC and the CMPC controller are identical. The main difference between the two controllers was the stochastic component of the prediction model. While tuning the SSMPC controller, two artificial states in the augmented state space model were introduced by choosing the following tuning parameters:

$$\mathbf{\Gamma}_\beta = \mathbf{\Gamma}, \quad \mathbf{Q}_\beta = \begin{bmatrix} 10^{-5} & 0 \\ 0 & 10^{-5} \end{bmatrix},$$
$$\mathbf{C}_\eta = [0], \quad \mathbf{Q}_\eta = [0],$$

where [0] represents the null matrix. Furthermore, identical stochastic realizations were used to generate the unmeasured disturbance (represented by the setpoint to hot water flow control loop), for testing regulatory performance of both the controllers. No constraints were imposed either on the manipulated inputs or the predicted outputs in the SSMPC or the CMPC formulations.

In addition to the random fluctuations of the hot water flow setpoint, a step jump of magnitude 3 mA (0.0963 l/s, 87% of the initial mean flow) was introduced in the hot water flow setpoint at 500th sampling instant in each case. The resulting hot water flow behavior is given in Fig. 7(c) for the experimental run with CMPC and in Fig. 9(c) for experimental run with SSMPC. Figs. 6 and 7 show the output and manipulated input responses generated using CMPC while Figs. 8 and 9 present the output and manipulated input responses generated using SSMPC. Table 10 presents comparison of regulatory performances based on maximum deviation from the setpoint, ISE values and performance index defined by Eq. (5.1).

From Figs. 6 and 8, it may be observed that both the formulations are able to achieve considerable reduction in the output variabilities around the initial and the final steady states (±0.2 cm in level and ±1 °C in temperature) when compared to the open loop response (see Fig. 5). From Table 10, it is also clear that the SSMPC formulation performs better than the CMPC formulation in terms of deviations from the setpoints and ISE values while rejecting the step change in the unmeasured disturbances. However, relatively smaller variability of the outputs in the case of SSMPC is achieved at the cost of larger variability of the manipulated inputs. This is reflected in the fact that the J values for SSMPC and CMPC are comparable. (Note that all the on-line computations have been carried out using models expressed in mA units. Thus, it is convenient to compare values of J reported in Table 10 in terms of $(\mathrm{mA})^2$, as mA units offer a common basis to express all inputs and outputs.)

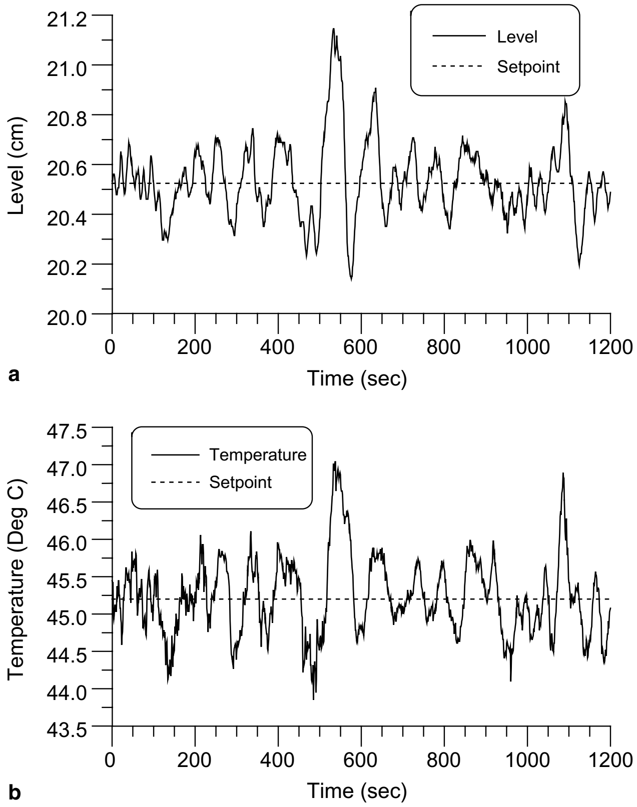

Fig. 6. CSTH: Rejection of step jump in unmeasured disturbance using CMPC controller: output responses.

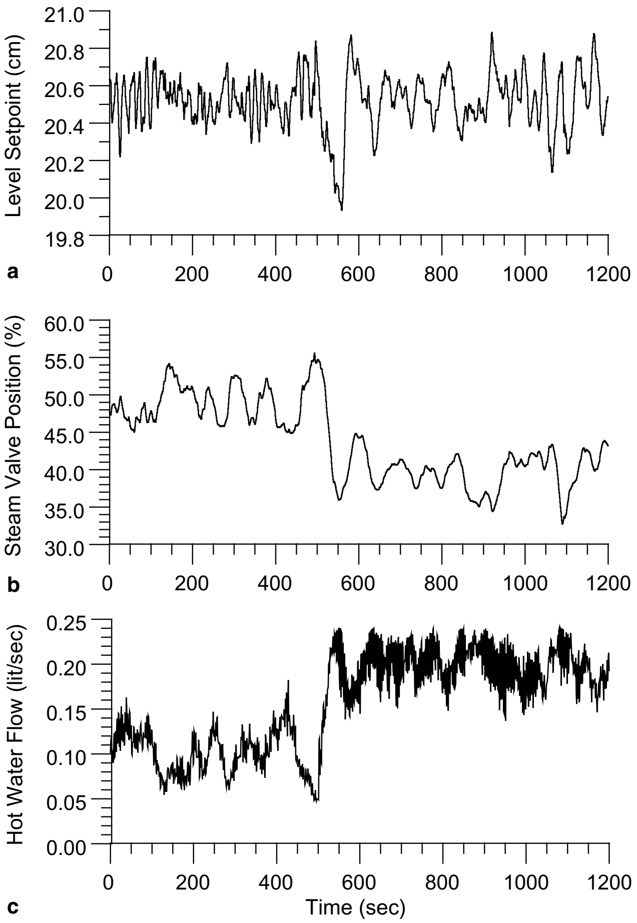

Fig. 7. CSTH: Rejection of step jump in unmeasured disturbance using CMPC controller: manipulated input and unmeasured disturbance profiles.

6.2. Fault tolerant control

The on-line FDI scheme is developed by hypothesizing that biases (step jumps) can occur in any one of the two sensors (temperature and level) and any one of the two manipulated inputs. The parameters used in the on-line fault diagnosis scheme are as follows:

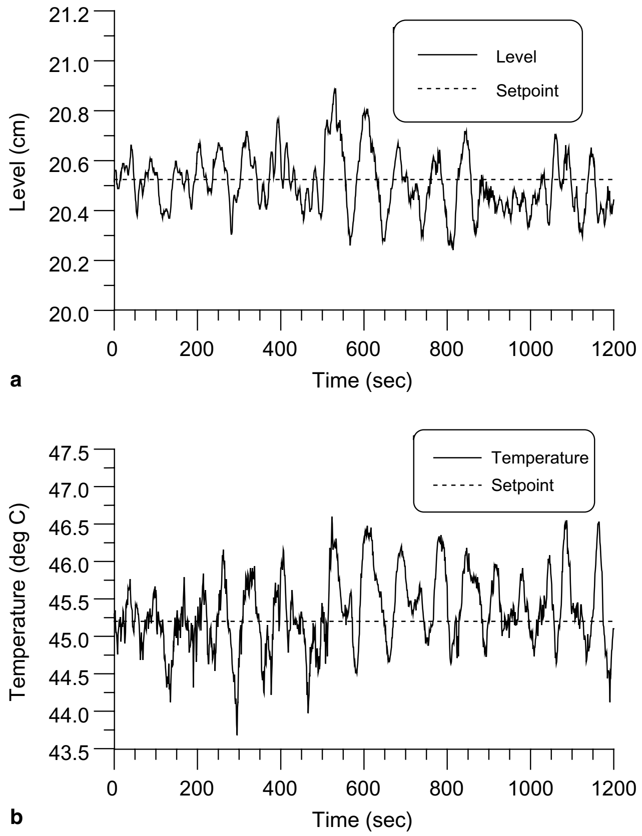

Fig. 8. CSTH: Rejection of step jump in unmeasured disturbance using SSMPC controller—output responses.

Window length N	100 sampling instants
Level of significance for FDT, α_{FDT}	0.2
Level of significance for FCT, α_{FCT}	0.01

The controller used is the unconstrained conventional MPC controller described in the above section. The efficacy of the proposed fault tolerant control scheme was tested for the following three cases

- *Sensor bias*: Bias in the temperature measurement,
- *Actuator bias*: Bias in the steam valve position,
- *Sequential fault*: Bias in temperature measurement followed by a bias in the steam valve position.

Note that fault identification is carried out in presence of continuously changing unmeasured disturbance in hot water flow. Characteristics of this disturbance are given in the accompanying paper [20] (see Fig. 5(c)).

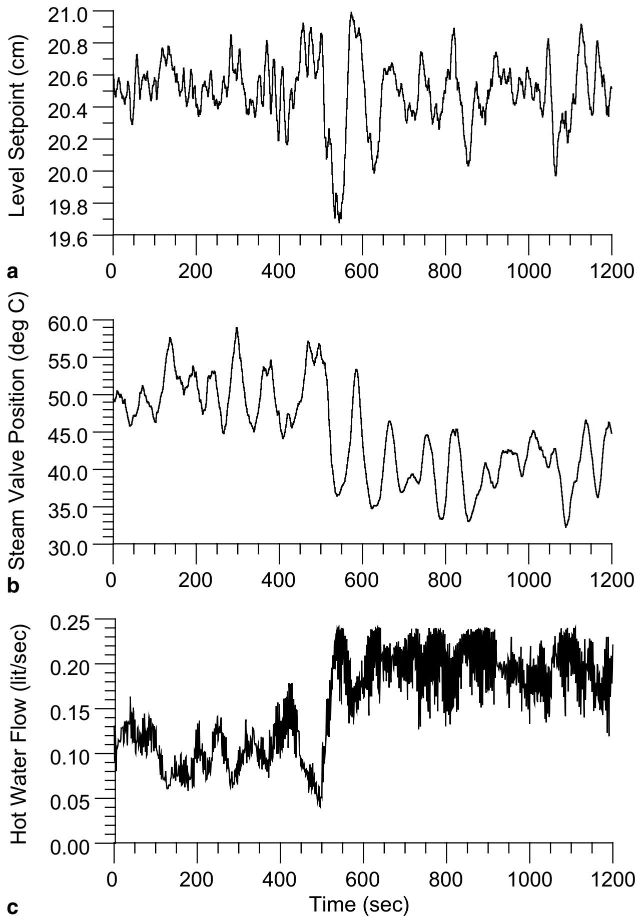

Fig. 9. CSTH: Rejection of step jump in unmeasured disturbance using SSMPC controller—manipulated input and unmeasured disturbance profiles.

6.2.1. Sensor bias

A persistent bias of magnitude 0.4 mA (equivalent to 2.5 °C) was introduced in the sampled temperature measurement as a step jump at sampling instant 500. The occurrence of fault is detected within one sampling instant by FDT and confirmed at 601st sampling instant by FCT. The FDI scheme correctly isolated the bias in temperature sensor and the maximum likelihood estimate of bias was obtained as 0.4058 mA (equivalent to 2.55 °C). The fault accommodation started at 601st instant using the estimated fault position and magnitude. The results of closed loop performance with and without fault accommodation in the presence of fault are shown

Table 10
CSTH: Regulatory performances of CMPC and SSMPC

Controller	$\max\|y_1(k)\|$ (cm)	$\max\|y_2(k)\|$ (°C)	ISE(y_1) (cm^2)	ISE(y_2) (°C^2)	J (mA^2)
CMPC	0.6207	1.8408	12.4780	114.928	10.24
SSMPC	0.3653	1.5216	9.4608	110.176	10.72