2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

Time-Varying Fault Tolerant Control of Nonlinear MIMO Systems

Meriç Çetin*[1], Selami Beyhan[2] and Serdar İplikçi[2]

Abstract— **In this study, a recently proposed discretization based observer is designed for the estimation of unknown time-varying fault variable. Using the estimated fault, a model predictive control based fault tolerant controller is proposed and applied for the real-time control of nonlinear multi-input multi-output (MIMO) systems. The main contributions are given as follows. i) An unknown parameter of the nonlinear grey model is estimated to compensate the uncertainty of the nonlinear system. ii) Second, an unknown process fault function of the nonlinear system is estimated and then the model inaccuracy is compensated using the estimated fault function. Fault and unknown parameter estimations are used to improve control performance of model predictive fault tolerant controller. In experiments, time-varying parameter and fault variables are accurately estimated and hence precise tracking performances are provided in real-time.**

I. INTRODUCTION

Process control systems usually incorporated with undesired dynamics such as noises, external disturbances, process faults and time-varying uncertain parameters. They cause inefficient control performances and inaccurate monitoring of the state. Using observers is one of the solutions to get the estimate values of the undesired dynamics such that using the estimated signals, the faulty dynamics can be corrected. In order to control the systems under time-varying faults, the fault can be detected and isolated via fault tolerant controllers. The fault tolerant controllers (FTC) are mainly constructed based on observer and controller designs. In the literature, various linear and nonlinear observers can be seen which are developed for different purposes such as real-time monitoring [1], [2], [3], fault-detection and isolation [4], [5], observer-based control applications [6], [7]. More recently, a discretization-based nonlinear observer has been developed and successively applied to state estimation and control [8], [9], [10], [11] and also to adaptive PID tuning [12].

On the other hand, model predictive control (MPC) approach is one of the most used controllers as a FTC [13]. Fault tolerant MPCs improve the process control performance when a system is subject to unknown time-varying faults or parameters. In order to get an acceptable control performance in fault-tolerant MPC framework, exact model of the post-failure nonlinear system is required. In addition, abrupt parameter changes caused by failure should be synchronously predicted from input/output measurements in an adaptive estimation mechanism. Therefore, a two-step observer-based controller: (i)time-varying dynamic function estimations, ii)real-time adaptation of the nonlinear system to this estimations, may be considered for the solution of such problems. In literature, unknown disturbance observers and fault tolerant controllers have been designed and implemented in the real-time applications [14], [15], [16], [17].

The main purpose of this paper is to design an fault tolerant controller under time-varying unknown system faults and parameters for real-time applications. In this sense, model predictive controller and discretization-based gradient observer (DBGO) are designed together as the fault tolerant control of nonlinear MIMO systems. In the first case, DBGO is used to estimate an unknown time-varying parameter of the system. In the second case, an unknown time-varying system fault function is estimated using designed nonlinear fault estimator. In order to implement the proposed fault tolerant controller, three-tank liquid-level system is used such that the faults are obtained by changing the openness of a tap between of two tanks. The change in the openness of the tap directly varies the mathematical model of the system. In that case, MPC controller can not control the liquid-level of the system itself. Using the DBGO based fault tolerant MPC, the unknown parameter and fault are estimated and desired level positions are accurately tracked in real-time application.

The rest of the paper is organized as follows. A brief introduction to the model predictive control is given in the next section. Then, the DBGO is introduced and in the following section, the real-time experiments related to uncertainties and faults are conducted. Finally, conclusions are drawn in the last section.

II. BRIEF INTRODUCTION TO NONLINEAR MPC

Consider a n*th* order continuous-time nonlinear multi-input multi-output (MIMO) system of the form

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{g}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}), \\ \mathbf{y} &= \mathbf{h}(\mathbf{x},\mathbf{u},\boldsymbol{\theta}) \\ \mathbf{u} &\in U, \mathbf{x} \in X, \forall t \geq 0 \end{aligned} \tag{1}$$

where $\mathbf{x}(t) \in \Re^N$, $\mathbf{u}(t) \in \Re^R$, $\mathbf{y}(t) \in \Re^Q$ and $\boldsymbol{\theta}$ denote the vectors of state, input, output measurements and parameters, respectively. It is assumed that the nonlinear system is controllable, $\mathbf{g}(.)$, $\mathbf{h}(.)$ functions are known and continuously differentiable with respect to the control inputs, the state variables and the system parameters. In general, when a continuous-time controller is designed for the system (1), the produced control input is applied to the system and fourth-order Runge-Kutta (RK) integration routine is used to evaluate the states of the system. In this paper, based on the discretized nonlinear system given in the next section,

*Corresponding author

[1] Pamukkale University, Department of Computer Engineering, 20070, Denizli, TURKEY mcetin@pau.edu.tr

[2] Department of Electrical and Electronics Engineering, 20070, Denizli, TURKEY sbeyhan@pau.edu.tr,iplikci@pau.edu.tr

978-1-5386-7891-6/18/$31.00 ©2018 IEEE

a conventional discrete-time nonlinear model predictive controller (NMPC) is designed. For the given nonlinear system (1), the discrete-time NMPC strategy has been formulated as a constrained optimization problem,

$$\begin{aligned}
&\min_{\mathbf{u}} F(x[n], u[n]) \\
&\textit{subject to} \\
&\mathrm{X}_i = \{x_i \in \Re \mid \quad x_{i_{min}} \leq x_i \leq x_{i_{max}}, i = 1, \ldots, N\} \\
&\mathrm{U}_r = \{u_r \in \Re \mid \quad u_{r_{min}} \leq u_r \leq u_{r_{max}}, r = 1, \ldots, R\} \\
&\Delta\mathrm{U}_r = \{\Delta u_r \in \Re \mid \quad |\Delta u_r| \leq \Delta u_{r_{max}}, r = 1, \ldots, R\} \\
&\mathrm{Y}_q = \{y_q \in \Re \mid \quad y_{q_{min}} \leq y_q \leq y_{q_{max}}, q = 1, \ldots, Q\}
\end{aligned} \tag{2}$$

with

$$\begin{aligned}
F(\mathbf{u}[n+1]) = &\frac{1}{2}\sum_{p=1}^{\kappa}\sum_{j=1}^{Q}(\tilde{y}_j[n+p] - \hat{y}_j[n+p])^2 + \\
&\frac{1}{2}\lambda_r\sum_{r=1}^{R}(u_r[n+1] - u_r[n])^2
\end{aligned} \tag{3}$$

where λ_r is the penalty term that prevents the excessive change in the *rth* input, κ defines the moving prediction horizon. $[\hat{y}_1[n+1], \ldots, \hat{y}_q[n+\kappa]]$ is the κ-step ahead prediction vector for the *qth* output of the nonlinear system and $\tilde{\mathbf{y}}$ is the reference signal vector at the discrete-time index *n*. The minimization of (3) is achieved by using the Levenberg-Marquardt (LM) algorithm such that the goal of the NMPC problem is computing the optimal control signal ($\mathbf{u}^*$) that takes the system to the desired region at every sampling time. When the first input values in the sequence of the candidate control vector ($\mathbf{u}[n+1] = [u_1[n], \ldots, u_R[n]]^{\mathrm{T}}$) are applied to the system κ times, the discretized model of the nonlinear system produces the future predictions of the system. Using this standard receding horizon procedure, closed-loop performance in the presence of unmeasured disturbances and modeling errors are enhanced. Based on the future predictions, $\mathbf{u}^*[n+1]$ is updated to minimize the objective function F by adding a correction term. When $\delta\mathbf{u}[n+1]$ is the correction term, an optimal control action ($\mathbf{u}^*[n{+}1] = \mathbf{u}[n{+}1]{+}\delta\mathbf{u}[n{+}1]$) to be applied to the nonlinear system. $\delta\mathbf{u}[n+1]$ is calculated by minimizing the objective function F while $F(\mathbf{u}[n+1] + \delta\mathbf{u}[n+1]) < F(\mathbf{u}[n+1])$. The second-order Taylor approximation is used to minimize the F with respect to $\delta\mathbf{u}[n+1]$ as follows:

$$\begin{aligned}
F(\mathbf{u}[n+1] + \delta\mathbf{u}[n+1]) \cong F(\mathbf{u}[n+1]) + \frac{\partial F(\mathbf{u}[n+1])}{\partial(\mathbf{u}[n+1])}\delta\mathbf{u}[n+1] \\
+ \frac{1}{2}\frac{\partial^2 F(\mathbf{u}[n+1])}{\partial\mathbf{u}^2[n+1]}(\delta\mathbf{u}[n+1])^2
\end{aligned} \tag{4}$$

The approximate objective function must be derivated with respect to $\delta\mathbf{u}[n+1]$ and then equated to zero ($\delta\mathbf{u}[n+1] = -\eta\frac{\frac{\partial F}{\partial\mathbf{u}[n+1]}}{\frac{\partial^2 F}{\partial\mathbf{u}^2[n+1]}} = 0$). $0 < \eta < 1$ is an adjustable parameter in order for the constraints on the input signals, input-slews and output signals to stay within their allowable limits. Because of the computational complexity of the second-order derivatives in (4), Jacobian approximations can be used instead of first and second-order derivatives. Using this approximation, derivatives in (4) can be written as: $\frac{\partial F}{\partial\mathbf{u}[n]} = 2\mathbf{J}^T\hat{\mathbf{e}}$, $\frac{\partial^2 F}{\partial\mathbf{u}^2[n]} \cong 2\mathbf{J}^T\mathbf{J}$. Substituting $\Delta\mathbf{u}[n+1] = \mathbf{u}[n+1] - \mathbf{u}[n]$, $\hat{\mathbf{e}}$ is the prediction error vector between desired and measured output as follows:

$$\hat{\mathbf{e}} = \begin{bmatrix} \hat{e}_1[n+1] \\ \vdots \\ \hat{e}_Q[n+\kappa] \\ \sqrt{\lambda_1}\Delta\mathbf{u}[n+1] \\ \vdots \\ \sqrt{\lambda_{\mathrm{R}}}\Delta\mathbf{u}[n+1] \end{bmatrix} = \begin{bmatrix} \tilde{y}_1[n+1] - \hat{y}_1[n+1] \\ \vdots \\ \tilde{y}_Q[n+\kappa] - \hat{y}_Q[n+\kappa] \\ \sqrt{\lambda_1}\Delta\mathbf{u}[n+1] \\ \vdots \\ \sqrt{\lambda_{\mathrm{R}}}\Delta\mathbf{u}[n+1] \end{bmatrix}. \tag{5}$$

Since the reference signal is in constant form, $\mathbf{J}$ matrix is defined as

$$\mathbf{J} = -\begin{bmatrix} \frac{\partial\hat{y}_1[n+1]}{\partial u_1[n+1]} & \frac{\partial\hat{y}_1[n+1]}{\partial u_2[n+1]} & \cdots & \frac{\partial\hat{y}_1[n+1]}{\partial u_{\mathrm{R}}[n+1]} \\ & & \vdots & \\ \frac{\partial\hat{y}_Q[n+\kappa]}{\partial u_1[n+1]} & \frac{\partial\hat{y}_Q[n+\kappa]}{\partial u_2[n+1]} & \cdots & \frac{\partial\hat{y}_Q[n+\kappa]}{\partial u_{\mathrm{R}}[n+1]} \\ \sqrt{\lambda_1} & \sqrt{\lambda_1} & \cdots & \sqrt{\lambda_1} \\ & & \vdots & \\ \sqrt{\lambda_{\mathrm{R}}} & \sqrt{\lambda_{\mathrm{R}}} & \cdots & \sqrt{\lambda_{\mathrm{R}}}. \end{bmatrix}. \tag{6}$$

After these changes regarding with the correction term, finally, the update rule for the optimal control action is given as follows:

$$\delta\mathbf{u}[n+1] = -(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})^{-1}\mathbf{J}^T\hat{\mathbf{e}}. \tag{7}$$

$$\mathbf{u}[n+1] \Longleftarrow \mathbf{u}[n] + \delta\mathbf{u}[n+1] \tag{8}$$

where $\mathbf{I}_{R\times R}$ is an identity matrix and μ provides switching the optimization behavior between the Steepest Descent and the Gauss Newton method.

III. Nonlinear Observer Design

The continuous-time nonlinear system (1) is sampled in T_s intervals at the discrete-time index n. The state and measurement equations can be written at the $(n+1)th$ sampling index as

$$\begin{aligned}
\hat{\mathbf{x}}[n+1] &= \hat{\mathbf{g}}(\hat{\mathbf{x}}[n], \mathbf{u}[n], \boldsymbol{\theta}) \\
&= \hat{\mathbf{x}}[n] + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4), \\
\hat{\mathbf{y}}[n+1] &= \mathbf{h}(\hat{\mathbf{x}}[n], \mathbf{u}[n], \boldsymbol{\theta}).
\end{aligned} \tag{9}$$

with

$$\begin{aligned}
\mathbf{k}_1 &= T_s\hat{\mathbf{g}}(\hat{\mathbf{x}}[n], \mathbf{u}[n], \boldsymbol{\theta}) \\
\mathbf{k}_2 &= T_s\hat{\mathbf{g}}(\hat{\mathbf{x}}[n] + 0.5\mathbf{k}_1, \mathbf{u}[n], \boldsymbol{\theta}) \\
\mathbf{k}_3 &= T_s\hat{\mathbf{g}}(\hat{\mathbf{x}}[n] + 0.5\mathbf{k}_2, \mathbf{u}[n], \boldsymbol{\theta}) \\
\mathbf{k}_4 &= T_s\hat{\mathbf{g}}(\hat{\mathbf{x}}[n] + \mathbf{k}_3, \mathbf{u}[n], \boldsymbol{\theta}).
\end{aligned} \tag{10}$$

where $\hat{\mathbf{g}}$ is named as the discrete-time model of the continuous-time process. There are several studies associated with the discretization-based methods in the literature [8], [9], [12], [10], [11]. In this paper, the discrete-time model of the nonlinear system is used to derive the future control actions in the NMPC framework and then, to predict the unknown system parameters or unknown faults.

A. Parameter Estimation

In this study, the iterative equations for the state and parameter estimation are derived based on the gradients for the error square minimization. In order to estimate the uncertain parameters of the $\boldsymbol{\theta}$ vector in (9), input and output signals are assumed known at n*th* index values. First, $\hat{\mathbf{x}}[n+1]$ is predicted by discrete model of the system then, the vector of prediction errors ($\hat{\mathbf{e}}$) related to the parameters is updated as follows.

$$\hat{\mathbf{e}} = \begin{bmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \vdots \\ \hat{e}_N \end{bmatrix} = \begin{bmatrix} x_1[n+1] - \hat{x}_1[n+1] \\ x_2[n+1] - \hat{x}_2[n+1] \\ \vdots \\ x_N[n+1] - \hat{x}_N[n+1] \end{bmatrix}_{N\times 1} . \tag{11}$$

This error vector is used in $\mathbf{J}_{\boldsymbol{\theta}}$ jacobian vector for the parameter estimation as

$$\mathbf{J}_{\boldsymbol{\theta}} = \begin{bmatrix} \frac{\partial \hat{e}_1}{\partial \boldsymbol{\theta}} \\ \frac{\partial \hat{e}_2}{\partial \boldsymbol{\theta}} \\ \vdots \\ \frac{\partial \hat{e}_N}{\partial \boldsymbol{\theta}} \end{bmatrix} = - \begin{bmatrix} \frac{\partial \hat{x}_1[n+1]}{\partial \boldsymbol{\theta}} \\ \frac{\partial \hat{x}_2[n+1]}{\partial \boldsymbol{\theta}} \\ \vdots \\ \frac{\partial \hat{x}_N[n+1]}{\partial \boldsymbol{\theta}} \end{bmatrix}_{N\times P} \tag{12}$$

where P indicates the number of unknown parameters. The update rule for the parameter estimation process is given by

$$\boldsymbol{\theta}[n+1] \Longleftarrow \boldsymbol{\theta}[n] + \Delta\boldsymbol{\theta}[n] \tag{13}$$

$$\Delta\boldsymbol{\theta}[n] = -(\mathbf{J}_{\boldsymbol{\theta}}^T \mathbf{J}_{\boldsymbol{\theta}} + \mu \mathbf{I})^{-1} \mathbf{J}_{\boldsymbol{\theta}}^T \hat{\mathbf{e}}. \tag{14}$$

Note that $\mathbf{I}$ is a proper identity matrix. Now, the problem is to calculate the partial derivatives $\frac{\partial \hat{x}_1[n+1]}{\partial \boldsymbol{\theta}}, \ldots, \frac{\partial \hat{x}_N[n+1]}{\partial \boldsymbol{\theta}}$ in $\mathbf{J}_{\boldsymbol{\theta}}$ definition:

$$\frac{\partial \hat{x}_i[n+1]}{\partial \boldsymbol{\theta}} = \frac{1}{6}\left(\frac{\partial \mathbf{K}_1}{\partial \boldsymbol{\theta}} + 2\frac{\partial \mathbf{K}_2}{\partial \boldsymbol{\theta}} + 2\frac{\partial \mathbf{K}_3}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{K}_4}{\partial \boldsymbol{\theta}}\right). \tag{15}$$

Using chain rule, the partial derivative components in (15) is obtained by discretization-based model of the nonlinear system as follows.

$$\begin{aligned} \frac{\partial \mathbf{K}_1}{\partial \boldsymbol{\theta}} &= T_s \frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}}\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n] \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_2}{\partial \boldsymbol{\theta}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_1}{\partial \boldsymbol{\theta}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+0.5\mathbf{K}_1 \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_3}{\partial \boldsymbol{\theta}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_2}{\partial \boldsymbol{\theta}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+0.5\mathbf{K}_2 \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_4}{\partial \boldsymbol{\theta}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_3}{\partial \boldsymbol{\theta}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+\mathbf{K}_3 \\ \mathbf{u}[n]=\mathbf{u}[n]}}. \end{aligned} \tag{16}$$

B. Fault Estimation

A nonlinear system may be affected by some unexpected factors such as actuator, sensor and plant faults. In this section, the dynamics of the nonlinear discrete-time system given in (9) with additive faults is presented as

$$\begin{aligned} \hat{\mathbf{x}}[n+1] &= \hat{\mathbf{g}}(\hat{\mathbf{x}}[n], \mathbf{u}[n], \boldsymbol{\theta}) + \mathbf{f}(\hat{\mathbf{x}}[n], \mathbf{u}[n]), \\ \hat{\mathbf{y}}[n+1] &= \mathbf{h}(\hat{\mathbf{x}}[n], \mathbf{u}[n], \boldsymbol{\theta}), \end{aligned} \tag{17}$$

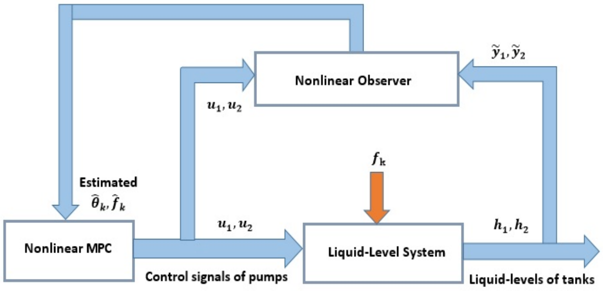


Fig. 1. Adaptive unscented Kalman filter-based nonlinear optimal controller block diagram.

where $\mathbf{f}(.) = (f_1, f_2, ..., f_\varrho) \in \Re^\varrho$ is the fault function vector of the process. The fault vector is considered as an unknown input of the system and can be a constant or varying signal depending on the time or state. In order to estimate the unknown fault function using DBGO, the prediction error vector ($\hat{\mathbf{e}}$) is derived with respect to the fault vector. The obtained $\mathbf{J_f}$ jacobian vector includes the partial derivatives related to the faults using chain rule as in following steps:

$$\mathbf{J_f} = \frac{\partial \hat{e}[n+1]}{\partial \mathbf{f}} = \left[\frac{\partial^T \mathbf{h}}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}[n+1]} \frac{\partial \hat{\mathbf{x}}[n+1]}{\partial \mathbf{f}[n+1]}\right], \tag{18}$$

where

$$\frac{\partial \hat{x}_i[n+1]}{\partial f_\varrho[n+1]} = \frac{1}{6}\left(\frac{\partial \mathbf{K}_1}{\partial \mathbf{f}} + 2\frac{\partial \mathbf{K}_2}{\partial \mathbf{f}} + 2\frac{\partial \mathbf{K}_3}{\partial \mathbf{f}} + \frac{\partial \mathbf{K}_4}{\partial \mathbf{f}}\right) \tag{19}$$

and accordingly,

$$\begin{aligned} \frac{\partial \mathbf{K}_1}{\partial \mathbf{f}} &= T_s \frac{\partial \mathbf{g}}{\partial \mathbf{f}}\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n] \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_2}{\partial \mathbf{f}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \mathbf{f}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_1}{\partial \mathbf{f}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+0.5\mathbf{K}_1 \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_3}{\partial \mathbf{f}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \mathbf{f}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_2}{\partial \mathbf{f}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+0.5\mathbf{K}_2 \\ \mathbf{u}[n]=\mathbf{u}[n]}}, \\ \frac{\partial \mathbf{K}_4}{\partial \mathbf{f}} &= T_s \left[\frac{\partial \mathbf{g}}{\partial \mathbf{f}} + \frac{1}{2}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{\partial \mathbf{K}_3}{\partial \mathbf{f}}\right]\Big|_{\substack{\mathbf{x}[n]=\mathbf{x}[n]+\mathbf{K}_3 \\ \mathbf{u}[n]=\mathbf{u}[n]}}. \end{aligned} \tag{20}$$

The update rule for the dynamic fault estimation process is given as follows ($\mathbf{I}$ is an proper identity matrix):

$$\Delta\mathbf{f}[n+1] = -(\mathbf{J_f}^T \mathbf{J_f} + \mu \mathbf{I})^{-1} \mathbf{J_f}^T \hat{\mathbf{e}}. \tag{21}$$

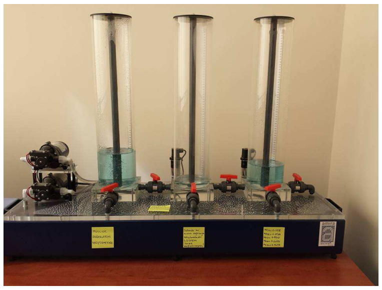
$$\mathbf{f}[n+1] \Longleftarrow \mathbf{f}[n] + \Delta\mathbf{f}[n+1] \tag{22}$$

Comprehensive stability proof for DBGO was shown in [11]. A block diagram of designed time-varying fault-tolerant control of three-tank liquid-level system is added in Figure 1.

IV. Real-Time Experimental Results

A. Three-Tank Liquid Level System

The three-tank liquid-level system, which is a nonlinear MIMO experimental system [18], is illustrated in Figure 2. Liquid-level system consists of three interconnected cylindrical tanks, which are coupled by two connecting pipes, two



(a) Liquid-level control system

Fig. 2. Three-tank liquid-level control system and schematic structure.

pumps and six valves. The pumps feed water from left tank to right tank. Valve positions are controlled and measured by electrical signals which allow the precise positioning and pumps are controlled by analogue signals in range from −10 V to 10 V. The dynamic model of the nonlinear system is given as

$$\begin{aligned}\dot{x}_1(t) =&\frac{1}{\mathrm{A}}[u_1(t) - \mathrm{Q}_{13}(t) - \mathrm{Q}_{10}(t)],\\ \dot{x}_2(t) =&\frac{1}{\mathrm{A}}[u_2(t) + \mathrm{Q}_{32}(t) - \mathrm{Q}_{20}(t)],\\ \dot{x}_3(t) =&\frac{1}{\mathrm{A}}[\mathrm{Q}_{13}(t) - \mathrm{Q}_{32}(t) - \mathrm{Q}_{30}(t)],\end{aligned} \tag{23}$$

with

$$\begin{aligned}Q_{13}(t) =&\mathrm{az}_{13}\mathrm{S}_n\mathrm{sign}(x_1(t) - x_3(t))\sqrt{2\mathrm{g}|x_1(t) - x_3(t)|},\\ Q_{32}(t) =&\mathrm{az}_{32}\mathrm{S}_n\mathrm{sign}(x_3(t) - x_2(t))\sqrt{2\mathrm{g}|x_3(t) - x_2(t)|},\\ Q_{10}(t) =&\mathrm{az}_{10}\mathrm{S}_n\sqrt{2\mathrm{g}x_1(t)},\\ Q_{20}(t) =&\mathrm{az}_{20}\mathrm{S}_n\sqrt{2\mathrm{g}x_2(t)},\\ Q_{30}(t) =&\mathrm{az}_{30}\mathrm{S}_n\sqrt{2\mathrm{g}x_3(t)},\end{aligned} \tag{24}$$

where u_r is the supplying flow rate of pump_r as the *rth* input and x_i is the liquid-level of tank_i as the *ith* output of the controlled plant. Q_{ij} $(i, j = 0 \ldots 3)$ represents the water flow between the tanks. The limits of the control signals are $\mathrm{u}_{r_{\min}} = 0$ m^3/s and $\mathrm{u}_{r_{\max}} = 0.0001$ m^3/s. In addition, the limits of the output signals are $\mathrm{y}_{q_{\min}} = 0$ m and $\mathrm{y}_{q_{\max}} = 0.6$ m. The numerical values of the liquid-level system parameters are given in Table I.

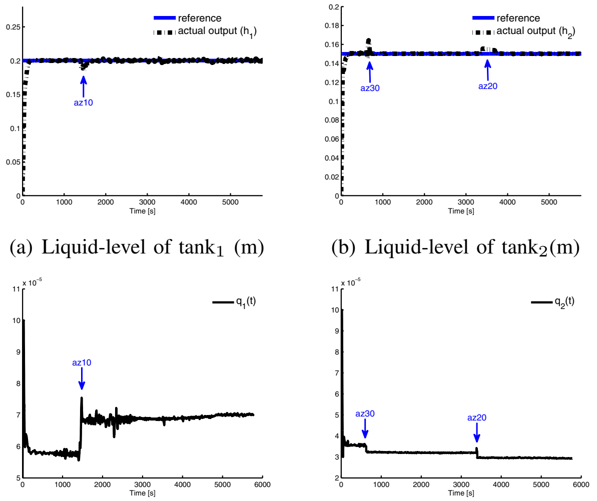
TABLE I

THE PARAMETERS OF THREE-TANK LIQUID-LEVEL CONTROL SYSTEM.

Parameter description	Value
az_{13}: outflow coefficient between tank_1 - tank_3	0.0280
az_{32}: outflow coefficient between tank_3 - tank_2	0.2569
az_{10}: outflow coefficient between tank_1 - tank_0	unknown
az_{20}: outflow coefficient between tank_2 - tank_0	unknown
az_{30}: outflow coefficient between tank_3 - tank_0	unknown
A: section of cylinders	0.0154 [m^2]
S_n: section of connection pipe n	5x10^{-5} [m^2]
g: gravitation coefficient	9.81[m/s^2]

B. Parameter Estimation Based Control Results

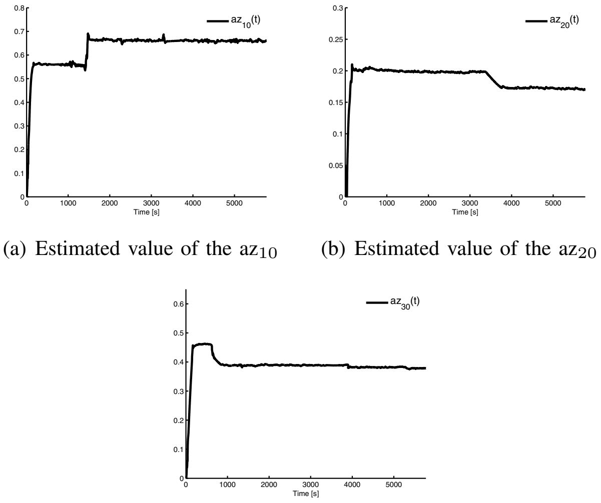
In this section, it is assumed that az_{10}, az_{20}, az_{30} parameters are not known and a DBGO is designed to estimate those parameters. Using the estimated parameters, a model based MPC is designed where the real-time estimation and tracking results are shown in Figure 3. The main issue is to find optimal values of the parameters corresponding to the valves during the control action. The unknown parameters (az_{10}, az_{20} and az_{30}) are initially set to zero. Then the parameters are updated at every iteration by designed observer in adaptive control mechanism. When the candidate control signal is applied to the system κ times, the discrete model produces the future predictions of the nonlinear system then the unknown parameters are updated to minimize the objective function. For an acceptable control performance, the



(a) Liquid-level of tank_1 (m) (b) Liquid-level of tank_2(m)

(c) Control signal of pump_1(m^3/s) (d) Control signal of pump_2(m^3/s)

Fig. 3. Real-time parameter estimation based control results of liquid-level system.



(a) Estimated value of the az_{10} (b) Estimated value of the az_{20}

(c) Estimated value of the az_{30}

Fig. 4. Real-time parameter estimations.

design parameters of the observer based control mechanism are determined as optimum values through grid−search as follows: i) $\kappa = 20$, ii) $\lambda = 0.001$, iii) $\mu = 10$.

Using the designed controller, the liquid-levels of the system are accurately tracked to the desired references. The references are selected constant to indicate the effect of the parameter variation on the control which are shown in Figure 3(a) and 3(b), respectively. The input signals have been generated to achieve a satisfactory control performance at every sampling interval seen in Figure 3(c) and 3(d), respectively. As shown in Figure 4(a), Figure 4(b) and Figure 4(c), unknown time-varying system parameters are converged to the fixed values during the control process. The parameters az_{10}, az_{20} and az_{30} are changed at $1500th$, $3200th$ and $800th$ seconds, respectively. The effects of variations those parameters are individually indicated on the input-output measurements and Table II. At the change of parameters, the observer detects the variations of input-output dynamics. Then the parameters are converged to the true parameter values where this adaptation is evaluated by the control mechanism at the same time. Finally, it is seen that when the unknown parameters are close to optimal values it means that the control performance of adaptive controller is acceptable.

TABLE II

THE NUMERICAL VALUES OF THE VALVES

Parameters	az_{10}	az_{20}	az_{30}
Initial	≈ 0	≈ 0	≈ 0
First Est.	0.559	0.204	0.461
Second Est.	0.657	0.170	0.387

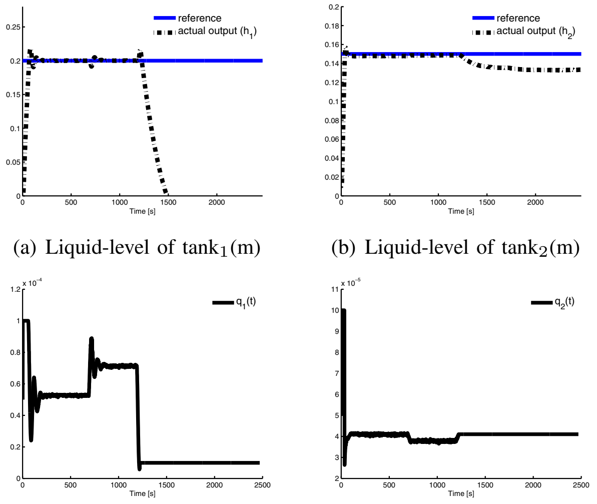
C. Fault Estimation Based Control Results

During the real-time control process, systems can be exposed to some undesired dynamics such as external disturbances, noises and faults. In order to obtain an acceptable control performance, a model based nonlinear observer can be designed to estimate such undesired dynamics. In this section, the effect of the unknown faults is investigated for liquid-level system. Therefore, an additive unknown fault is applied to the first and third tank of the liquid-level system. The dynamics of the first and third tanks are turned into the following form with additive unknown fault $f_1(t)$.

$$\begin{aligned}\dot{x}_1(t) =& \frac{1}{\mathrm{A}}[u_1(t) - \mathrm{Q}_{13}(t) - \mathrm{Q}_{10}(t) + f_1(t)],\\ \dot{x}_3(t) =& \frac{1}{\mathrm{A}}[\mathrm{Q}_{13}(t) - \mathrm{Q}_{32}(t) - \mathrm{Q}_{30}(t) + f_1(t)],\end{aligned} \tag{25}$$

The fault $f_1(t)$ function is constructed by changing the az_{13} parameter. The effect of this parameter is seen in the dynamics of $tank_1$ and $tank_3$ at the same time. Therefore, the variation of az_{13} causes a fault function as $f_1(t) = \mathrm{az}_{13}\mathrm{S}_n\mathrm{sgn}(x_1(t) - x_3(t))\sqrt{2\mathrm{g}|x_1(t) - x_3(t)|}$. Thus, the main difference is here that not only the parameter is estimated instead the fault function is estimated where $f_1(t)$ is a function of x_1, x_3 and az_{13}, respectively.

Figure 5 shows the real-time control results of the liquid-level system with additive unknown fault function is presented in dynamics (25). It is seen that the experimental results have not satisfactory control performance in the absence of a fault tolerant control method. Figure 5(a) and 5(b) show the outputs of the liquid-level system. The actual outputs of the tanks are different from the reference trajectories. The control signals are insufficient to achieve the satisfactory control performance due to the imposed a severe fault at $1200th$ second as illustrated in Figure 5(c) and 5(d), respectively. $tank_1$ is out of order and the level of h_1 decreases dangerously and reaches its physical limitation (see Fig. 5(a)). The fault influence on $tank_2$ and control signals can be observed in Figure 5.



(a) Liquid-level of tank_1(m) (b) Liquid-level of tank_2(m)

(c) Control signal of $\mathrm{pump}_1(\mathrm{m}^3/\mathrm{s})$ (d) Control signal of $\mathrm{pump}_2(\mathrm{m}^3/\mathrm{s})$

Fig. 5. Real-time control results of faulty situation.

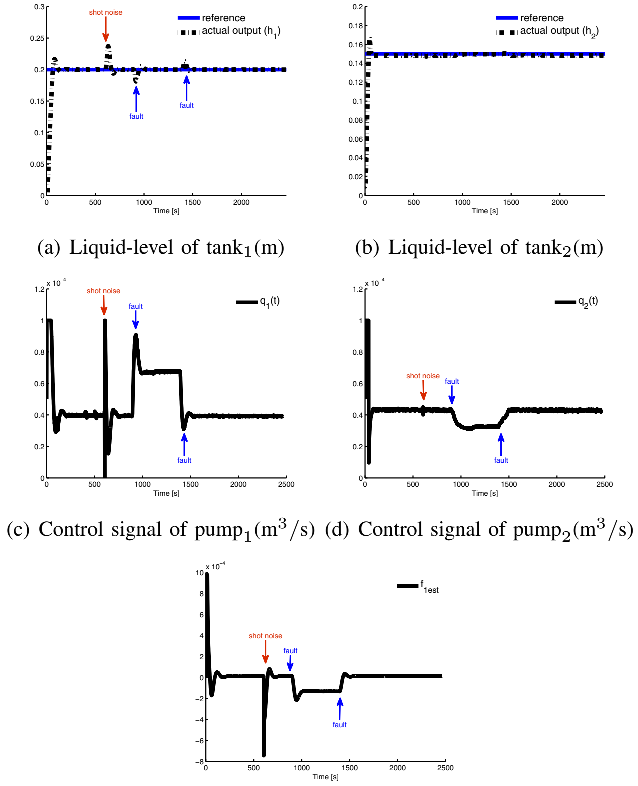
In the next experiment, unknown fault function is estimated using the DBGO. Then, estimated fault function is corrected in system dynamics as

$$\begin{aligned}\dot{x}_1(t) =& \frac{1}{\mathrm{A}}[u_1(t) - \mathrm{Q}_{13}(t) - \mathrm{Q}_{10}(t) + f_1(t) - \hat{f}_1(t)].\\ \dot{x}_3(t) =& \frac{1}{\mathrm{A}}[\mathrm{Q}_{13}(t) - \mathrm{Q}_{32}(t) - \mathrm{Q}_{30}(t) + f_1(t) - \hat{f}_1(t)],\end{aligned} \tag{26}$$

In Figure 6, fault tolerant control results are demonstrated using DBGO based MPC. The design parameters of the control method are determined as optimum values through grid−search as follows: i) $\kappa = 10$, ii) $\lambda = 0.001$, iii) $\mu = 3 \times 10^4$. The detailed explanation about the choice of design parameters is given in [12]. In the experiments, an additive unknown fault to the tank_1 and tank_3 is applied at instant of 600th,$1000th$ and $1500th$ seconds, respectively.

As shown in Figure 6(a) and 6(b), the outputs of the liquid-level system track the desired reference trajectories accurately although the level dynamics are affected by undesired fault. The control signals have been produced to achieve a satisfactory level of control performance at every sampling interval given in Figure 6(c) and 6(d), respectively.

The estimated fault function is presented in Figure 6(e). As seen from the Figures, when the fault function affects the level dynamics, the control signals are adapted to their optimum values. Then, undesired dynamics are compensated and acceptable control performance is obtained by designed fault tolerant controller.



(a) Liquid-level of tank_1(m)

(b) Liquid-level of tank_2(m)

(c) Control signal of $\text{pump}_1(\text{m}^3/\text{s})$

(d) Control signal of $\text{pump}_2(\text{m}^3/\text{s})$

(e) Estimated value of the $\hat{f}_1(t)$

Fig. 6. Real-time fault tolerant control results of liquid-level system.

V. Conclusion

In this paper, model predictive fault tolerant controller is proposed for the control of nonlinear MIMO systems. First, possible uncertain parameters and unknown fault are defined on the three tank system so that the effect of the uncertainties are considered in the control. It must be noted that the estimation of the unknown parameter and fault are similar tasks for the designed fault estimator since its estimation dynamics are based on the gradient of the system function w.r.t. to the estimated parameter. On the other hand, the lack of knowledge for fault and unknown parameter have very different effects on the tracking results. In the experiments, the uncertain parameters and unknown fault are accurately estimated and used in real-time control such that accurate observer based tracking results are obtained. It is concluded that the designed fault tolerant model predictive controller can be used for the control of uncertain systems with a satisfactory level of performance. The future work is planned on the fault tolerant control of unknown systems under sinusoidal disturbances.

VI. Acknowledgement

This paper is supported by Pamukkale University Scientific Research Projects Council under grand number 2018KRM002-035.

References

[1] F. Chen and M. Dunnigan, "Comparative study of a sliding-mode observer and kalman filters for full state estimation in an induction machine," in *Electric Power Applications, IEE Proceedings-*, vol. 149. IET, 2002, pp. 53–64.

[2] A. Mesbah, A. E. Huesman, H. J. Kramer, and P. M. Van den Hof, "A comparison of nonlinear observers for output feedback model-based control of seeded batch crystallization processes," *Journal of Process Control*, vol. 21, no. 4, pp. 652–666, 2011.

[3] S. Beyhan, "Adaptive fuzzy terminal sliding-mode observer with experimental applications," *International Journal of Fuzzy Systems*, vol. 18, p. 585, 2015.

[4] P. M. Frank and X. Ding, "Survey of robust residual generation and evaluation methods in observer-based fault detection systems," *Journal of process control*, vol. 7, no. 6, pp. 403–424, 1997.

[5] C. Edwards, S. K. Spurgeon, and R. J. Patton, "Sliding mode observers for fault detection and isolation," *Automatica*, vol. 36, no. 4, pp. 541–553, 2000.

[6] Y. Feng, J. Zheng, X. Yu, and N. V. Truong, "Hybrid terminal sliding-mode observer design method for a permanent-magnet synchronous motor control system," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 9, pp. 3424–3431, 2009.

[7] M. Ghanes and G. Zheng, "On sensorless induction motor drives: sliding-mode observer and output feedback controller," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 9, pp. 3404–3413, 2009.

[8] S. Iplikci, "Runge-Kutta model-based adaptive predictive control mechanism for non-linear processes," *Transactions of the Institute of Measurement and Control*, vol. 35, no. 2, pp. 166–180, 2013.

[9] S. Beyhan, "Runge-Kutta model-based nonlinear observer for synchronization and control of chaotic systems," *ISA Transactions*, vol. 52, no. 4, pp. 501–509, 2013.

[10] S. Iplikci and B. Bahtiyar, "A field-programmable gate array implementation of a real-time nonlinear Runge–Kutta model predictive control," *Transactions of the Institute of Measurement and Control*, 2015.

[11] M. Cetin, S. Beyhan, and S. Iplikci, "Soft sensor applications of rk-based nonlinear observers and experimental comparisons," *Intelligent Automation & Soft Computing*, vol. 23, no. 01, pp. 109–116, 2017.

[12] M. Cetin and S. Iplikci, "A novel auto-tuning PID control mechanism for nonlinear systems," *ISA transactions*, vol. 58, pp. 292–308, 2015.

[13] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Springer Science & Business Media, 2012, vol. 3.

[14] C. Edwards and C. P. Tan, "Sensor fault tolerant control using sliding mode observers," *Control Engineering Practice*, vol. 14, no. 8, pp. 897–908, 2006.

[15] L. Kiltz, C. Join, M. Mboup, and J. Rudolph, "Fault-tolerant control based on algebraic derivative estimation applied on a magnetically supported plate," *Control Engineering Practice*, vol. 26, pp. 107–115, 2014.

[16] M. Mahmoud, A. Memon, and P. Shi, "Observer-based fault-tolerant control for a class of nonlinear networked control systems," *International Journal of Control*, vol. 87, no. 8, pp. 1707–1715, 2014.

[17] S. Yin, H. Luo, and S. X. Ding, "Real-time implementation of fault-tolerant control systems with performance optimization," *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 5, pp. 2402–2411, 2014.

[18] *Amira, DTS 200 Laboratory Setup Three-Tank-System.*, Amira GmbH, Duisburg, 2002.