

Actuator Fault Diagnosis System Design and Verification for Ball on Plate System

Matej Oravec, Anna Jadlovská

Department of the Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice
Letná 9, Košice, Slovak Republic
matej.oravec@tuke.sk, anna.jadlovaska@tuke.sk

Abstract: The paper deals with the fault diagnosis methodology design for fault detection, localization, estimation and accommodation in a predictive control algorithm. The result of the proposed methodology realization is control and fault diagnosis system, which has a capability of actuators fault tolerance. The fault diagnosis system design is based on the group of unknown input observers. The fault diagnosis system and predictive control algorithm with fault accommodation are implemented in MATLAB. The designed algorithms, which are presented in this paper, are verified by simulations using the simulation model of the Ball on Plate system.

Keywords: fault diagnosis; fault-tolerant system; modeling; predictive control

1 Introduction

In recent years, control systems must ensure a very good steady-state and dynamic performance of controlled dynamic systems [1]. For these purposes, various types of control algorithms are designed using different approaches.

At present, the optimal control is very popular in various applications. In [2] is an optimal approach used for traffic lights optimization. In [1], authors present servo motor control system based on iterative feedback tuning that minimizes an objective function. Also, in [3], the iterative procedure is used for optimal control design of permanent magnet linear motor. Authors in [3] present speed controllers design for a class of small turbojet engines, which can be applied in a discrete digital control environment. In [18], the predictive control algorithms are listed which are based on the technique that solves the optimization problem at each sampling period. In this paper, the main focus is on the predictive control algorithm design using the discrete transfer function of the dynamic system.

Also, the risk of the dynamic system faults increases with its growing complexity means an undesirable influence on dynamic system control. The actuators and sensors are the most affected parts by faults. The fault tolerance capability of control algorithms is usually used to fulfil the given control goal, despite the actuator or sensor faults occurrence. The important assumption for control with fault tolerance is the Fault Detection and Diagnosis (FDD) system design, which is used for fault detection, localization, and estimation. The effectiveness of the diagnosis systems depends on the quality of used mathematical models [8].

In this paper, the predictive control algorithm is enhanced by the ability of fault tolerance [11], [12]. The information about faults is obtained from FDD system, which is based on the group of Unknown Inputs Observers (UIO) [9]. Also, the design of algorithms for the faults detection, localization, and estimation are shortly listed.

The Ball on Plate system has been widely used for the control algorithms design and verification of their performance because it is an unstable, nonlinear system [5], [6], [7], [18]. Application of the fault-tolerant control to Ball on Plate system is not common, but in [13] stress is put to solve this task. Also, this paper is dedicated to verification of the fault-tolerant control of Ball on Plate laboratory model, which is situated in the *Laboratory of Modern Control Techniques of Physical Systems* of the DCAI, FEEI at the Technical university of Košice. The paper follows previous research activities of the *Center of Modern Control Techniques and Industrial Informatics* (CMCTII, web: <http://kyb.fei.tuke.sk>) [21].

The main goal of this paper is to design of methodology that connects partial tasks as modelling, control or faults diagnosis algorithms design leading to creation of the complex control and FDD system. All steps of proposed methodology are described in next sections with their application to Ball on Plate system [22].

2 Design of Actuators Faults Diagnosis System

In general, faults detection and diagnosis system design consists of some steps, which are common for each dynamic system [9]. In this article, the methodology of the FDD system design for the actuators faults diagnosis is stated and implemented using the Ball on Plate model [22]. The presented methodology consists of these steps:

- 1) nominal system modeling,
- 2) design of nominal system control algorithm,
- 3) faulty system modeling and design of algorithms for actuator faults detection, localization and estimation,
- 4) fault-tolerant capability implementation to control algorithm.

Nominal model of Ball on Plate system is designed according to system analysis of laboratory model (<http://kyb.fei.tuke.sk/laben/modely/gnk.php>).

Model can be divided into the two subsystems, as the Servomotors subsystem and the B&P subsystem (Figure 1). The description of physical variables and parameters of the model are stated in Table 1.

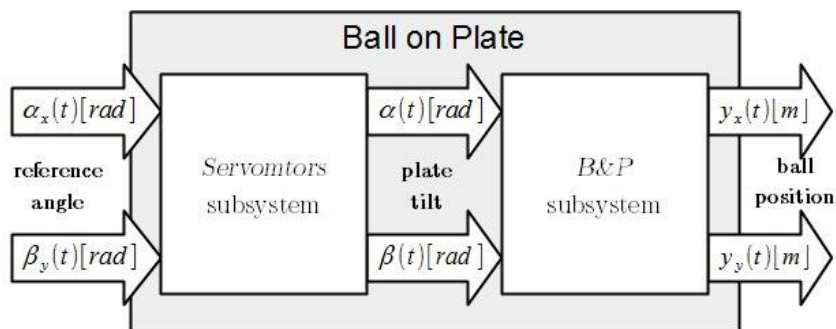


Figure 1

System decomposition of the Ball on Plate system

In the past, the Ball on Plate laboratory model was used for various control algorithms verification in CMCTII [13], [14], [15]. Also, the predictive control verification, which was designed using pseudo-state space form, was presented in [16].

Table 1

Variables and parameters of the Ball on Plate system

Description	Label	Units
ball position – axis x	$y_x(t)$	[m]
ball position – axis y	$y_y(t)$	[m]
plate tilt – axis x	$\alpha(t)$	[rad]
reference angle – axis x	$\alpha_x(t)$	[rad]
plate tilt – axis y	$\beta(t)$	[rad]
reference angle – axis y	$\beta_y(t)$	[rad]
servomotor gain – axis x	K_x	-
time constant of servomotor – axis x	T_x	[s]
servomotor gain – axis y	K_y	-
time constant of servomotor – axis y	T_y	[s]

This section shortly presents the algorithm design of the mentioned predictive control with actuator fault tolerant capability. However, early actuators fault detection, localization and estimation is very important for successful functionality of the control system. The Fault Detection and Diagnosis system (FDD) is implemented for this purpose, which is based on the group of the unknown inputs observers using Kalman filtration principles. The information from the FDD is

used for fault accommodation in control law. All algorithms are used in structure with interconnections between control and FDD system. The general scheme of control and FDD system is illustrated in Figure 2.

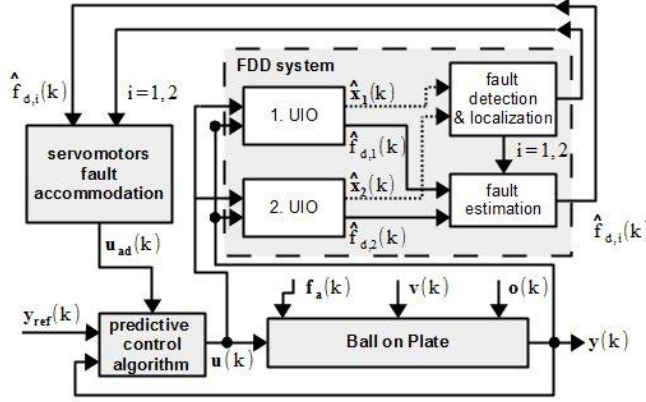


Figure 2

General scheme of the control & FDD system with fault accommodation in predictive control law

2.1 Ball and Plate System Modeling

As it was mentioned, the Ball on Plate system can be divided into two subsystems (Figure 1, Table 1). Mathematical model of the B&P subsystem can be designed as the two nonlinear differential equations:

$$\ddot{y}_x(t) = \frac{5}{7} g \sin \alpha(t) \quad (1)$$

$$\ddot{y}_y(t) = \frac{5}{7} g \sin \beta(t) \quad (2)$$

The subsystem of Servomotors can be expressed using two 1st order linear differential equations:

$$\dot{\alpha}(t) = \frac{K_x}{T_x} \cdot (\alpha_x(t) - \alpha(t)) \quad (3)$$

$$\dot{\beta}(t) = \frac{K_y}{T_y} \cdot (\beta_y(t) - \beta(t)) \quad (4)$$

where coefficients K_x , K_y , T_x , T_y are obtained by experimental identification [13].

The mathematical model of the system expressed by the differential equations (1) – (4) can be arranged into the canonical form using substitution:

- for axis x , where $y_x(t) = x_1(t)$, $\dot{y}_x(t) = x_2(t)$, $\alpha(t) = x_3(t)$, $\alpha_x(t) = u_x(t)$:

$$\dot{x}_1(t) = x_2(t) \quad (5)$$

$$\dot{x}_2(t) = \frac{5}{7}g \sin x_3(t) \quad (6)$$

$$\dot{x}_3(t) = \frac{1}{T_x}(K_x u_x(t) - x_3(t)) \quad (7)$$

- for axis y, where $y_y(t) = x_4(t)$, $\dot{y}_y(t) = x_5(t)$, $\beta(t) = x_6(t)$, $\beta_y(t) = u_y(t)$:

$$\dot{x}_4(t) = x_5(t) \quad (8)$$

$$\dot{x}_5(t) = \frac{5}{7}g \sin x_6(t) \quad (9)$$

$$\dot{x}_6(t) = \frac{1}{T_y}(K_y u_y(t) - x_6(t)) \quad (10)$$

The mathematical model in canonical state form (5) - (10) is used for the implementation as a simulation model in a *Matlab* environment. The simulation model was verified with the real laboratory model and the results [13], [14].

The nonlinear differential equations of the mathematical model in canonical form are linearized around the operating point:

$$x_x^{op} = [x_1^{op} \quad x_2^{op} \quad x_3^{op}], \quad x_y^{op} = [x_4^{op} \quad x_5^{op} \quad x_6^{op}] \quad (11)$$

The result of the linearization can be arranged into the two linear mathematical models in state-space form:

$$\underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix}}_{\dot{x}_x(t)} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{5}{7}g \\ 0 & 0 & -\frac{1}{T_x} \end{bmatrix}}_{A_x} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}}_{x_x(t)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \frac{K_x}{T_x} \end{bmatrix}}_{B_x} u_x(t) \quad (12)$$

$$y_x(t) = \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{C_x} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}}_{x_x(t)} \quad (13)$$

$$\underbrace{\begin{bmatrix} \dot{x}_4(t) \\ \dot{x}_5(t) \\ \dot{x}_6(t) \end{bmatrix}}_{\dot{x}_y(t)} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{5}{7}g \\ 0 & 0 & -\frac{1}{T_y} \end{bmatrix}}_{A_y} \underbrace{\begin{bmatrix} x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}}_{x_y(t)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \frac{K_y}{T_y} \end{bmatrix}}_{B_y} u_y(t) \quad (14)$$

$$y_y(t) = \underbrace{\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}}_{c_y^T} \underbrace{\begin{bmatrix} x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}}_{x_y(t)} \quad (15)$$

The state-space form (12)-(15) of the mathematical model is discretized using *c2d* function in *Matlab* environment and using a *ss2tf* function it can be expressed as the two transfer function:

$$F_x(z^{-1}) = \frac{B_x(z^{-1})}{A_x(z^{-1})} = \frac{b_{x,1}z^{-1} + b_{x,2}z^{-2} + b_{x,3}z^{-3}}{1 + a_{x,1}z^{-1} + a_{x,2}z^{-2} + a_{x,3}z^{-3}} \quad (16)$$

$$F_y(z^{-1}) = \frac{B_y(z^{-1})}{A_y(z^{-1})} = \frac{b_{y,1}z^{-1} + b_{y,2}z^{-2} + b_{y,3}z^{-3}}{1 + a_{y,1}z^{-1} + a_{y,2}z^{-2} + a_{y,3}z^{-3}} \quad (17)$$

where sampling period $T_s = 0.05$ s.

The mathematical model in the transfer functions form (16), (17) is used in the next subsection for the predictive control algorithm design. Also the state space form equations (12)-(15) are used for the faulty model derivation in Subsection 2.3.

2.2 Predictive Control Algorithm Design

For control of the Ball on Plate system a predictive control algorithm is used, which minimizes the cost function:

$$J_{PC} = \sum_{i=1}^{N_p} \mathbf{Q}[\hat{\mathbf{y}}(k+i) - \mathbf{y}_{ref}(k+i)]^2 + \sum_{i=1}^{N_u} \mathbf{R}[\mathbf{u}(k+i-1)]^2 \quad (18)$$

where N_p , N_u represents prediction horizon and control horizon, $\hat{\mathbf{y}}(k)$ is the vector of predicted values of the system output, $\mathbf{y}_{ref}(k)$ represents reference trajectory, $\mathbf{u}(k)$ is control input and \mathbf{Q} , \mathbf{R} are weight matrices [17], [19].

The predicted values $\hat{\mathbf{y}}(k)$ are obtained using pseudo-state space form:

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{A}_0 \mathbf{X}(k) + \mathbf{B}_0 \mathbf{U}_0(k), \\ y(k) &= \mathbf{C}_0 \mathbf{X}(k), \end{aligned} \quad (19)$$

which can be derived from the system discrete transfer function (16) or (17) as follow:

$$\underbrace{\begin{pmatrix} \mathbf{X}(k+1) \\ y(k-n+2) \\ \vdots \\ y(k) \\ y(k+1) \end{pmatrix}}_{\mathbf{X}(k+1)} = \underbrace{\begin{pmatrix} 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & & & 1 \\ -a_n & -a_{n-1} & \cdots & -a_1 \end{pmatrix}}_{\mathbf{A}_0} \underbrace{\begin{pmatrix} \mathbf{X}(k) \\ y(k-n+1) \\ \vdots \\ y(k-1) \\ y(k) \end{pmatrix}}_{\mathbf{X}(k)} + \underbrace{\begin{pmatrix} 0 & 0 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & & & 0 \\ b_n & b_{n-1} & \cdots & b_1 \end{pmatrix}}_{\mathbf{B}_0} \underbrace{\begin{pmatrix} u(k-q+1) \\ \vdots \\ u(k-1) \\ u(k) \end{pmatrix}}_{\mathbf{U}_0(k)} \quad (20)$$

$$y(k) = \underbrace{(0 \dots 0 \ 1)}_{\mathbf{C}_0} \underbrace{\begin{pmatrix} y(k-n+1) \\ \vdots \\ y(k) \end{pmatrix}}_{\mathbf{x}(k)} \quad (21)$$

The system output prediction $\hat{\mathbf{y}}(k) = [y(k+1), \dots, y(k+N_p)]$ can be computed using pseudo-state space (13) in form:

$$\begin{pmatrix} y(k+1) \\ \vdots \\ y(k+N_p) \end{pmatrix} = \begin{pmatrix} \mathbf{C}_0 \mathbf{A}_0 \\ \vdots \\ \mathbf{C}_0 \mathbf{A}_0^{N_p} \end{pmatrix} \begin{pmatrix} y(k-n+1) \\ \vdots \\ y(k) \end{pmatrix} + \bar{\mathbf{G}} \begin{pmatrix} u(k-q+1) \\ \vdots \\ u(k+N_p-1) \end{pmatrix} \quad (22)$$

where matrix $\bar{\mathbf{G}}$ consists of:

$$\bar{\mathbf{G}} = \begin{pmatrix} \mathbf{C}_0 \mathbf{B}_0 & \mathbf{0} \\ \vdots & \vdots \\ \mathbf{C}_0 \mathbf{B}_i & \mathbf{0} \\ \vdots & \vdots \\ \mathbf{C}_0 \mathbf{B}_{N_p-1} \end{pmatrix}, \mathbf{B}_0 = \begin{pmatrix} 0 & 0 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & & & 0 \\ b_n & b_{n-1} & \dots & b_1 \end{pmatrix} \quad (23)$$

$$\mathbf{B}_i = (\mathbf{A}_0 \mathbf{B}_{i-1} \ 0 \dots 0) + (0 \dots 0 \ \mathbf{B}_0), \quad i = 1, \dots, N_p - 1 \quad (24)$$

Finally, the system output prediction $\hat{\mathbf{y}}(k)$, according to [18], can be expressed:

$$\hat{\mathbf{y}}(k) = \mathbf{y}_0(k) + \mathbf{G}\mathbf{u}(k) \quad (25)$$

where \mathbf{y}_0 represents system free response:

$$\mathbf{y}_0(k) = \begin{pmatrix} \mathbf{C}_0 \mathbf{A}_0 \\ \vdots \\ \mathbf{C}_0 \mathbf{A}_0^{N_p} \end{pmatrix} \begin{pmatrix} y(k-n+1) \\ \vdots \\ y(k) \end{pmatrix} + \bar{\mathbf{G}}_{(:,1:q-1)} \begin{pmatrix} u(k-q+1) \\ \vdots \\ u(k-1) \end{pmatrix} \quad (26)$$

and matrix $\bar{\mathbf{G}}_{(:,1:q-1)}$ consists of all rows and $q-1$ columns of the $\bar{\mathbf{G}}$ matrix (the convention “:” is used according to the *MATLAB* environment).

Next columns of matrix $\bar{\mathbf{G}}$ are used for the computation of the system forced response $\mathbf{G}\mathbf{u}(k)$, where:

$$\mathbf{G} = \bar{\mathbf{G}}_{(:,q:q+N_p-1)} \quad (27)$$

The matrices of the system output prediction can be arranged into the Hessian \mathbf{H} and gradient \mathbf{g} :

$$\mathbf{H} = (\mathbf{G}^T \mathbf{Q}^T \mathbf{Q} \mathbf{G} + \mathbf{R}^T \mathbf{R}) \quad (28)$$

$$\mathbf{g}^T = (\mathbf{y}_0 - \mathbf{y}_{ref}(k))^T \mathbf{Q}^T \mathbf{Q} \mathbf{G}$$

Control law has form:

$$\mathbf{u}_{opt}(k) = -\mathbf{H}^{-1} \mathbf{g} \quad (29)$$

where the optimal control sequence \mathbf{u}_{opt} is computed with respect to criteria:

$$\frac{\partial J_{PC}}{\partial \mathbf{u}(k)} \stackrel{!}{=} 0 \quad (30)$$

Also, physical constraints of the controlled system, which are composed to inequality $\mathbf{U}_{con}\mathbf{u}_{opt} \leq \mathbf{v}_{con}$, are assumed and they can be expressed in form:

$$\mathbf{U}_{con} = \begin{pmatrix} \mathbf{I}_D \\ -\mathbf{I}_D \end{pmatrix}, \mathbf{v}_{con} = \begin{pmatrix} \mathbf{1}u_{max} \\ -\mathbf{1}u_{min} \end{pmatrix} \quad (31)$$

and $\mathbf{1}$ is a unit vector and \mathbf{I}_D is a unit matrix [19].

The optimal control sequence $\mathbf{u}_{opt}(k)$ can be computed in *Matlab* environment using the *quadprog* function (function of *Optimization toolbox*) by formula:

$$\min_{\mathbf{u}} \left(\frac{1}{2} \mathbf{u} \mathbf{H} \mathbf{u} + \mathbf{g}^T \mathbf{u} \right) \quad (32)$$

The designed predictive control algorithm is implemented as *m-file* called *ioGPCcon* to the *Matlab* environment with respect to control structure, which is shown in Figure 3. The predictive control algorithm is illustrated as a flowchart in Figure 4.

For computation of the Hessian \mathbf{H} , matrices of the free response y_0 and matrix of the forced response \mathbf{G} is also a created function - $[H, y0_AC, y0_G, G, Ucon, vcon] = \text{paramGPCc}(Bx, Ax, Q, R, Np, Nu)$.

The predictive control algorithm is used twice for the control of the servo motors in x or y direction independently.

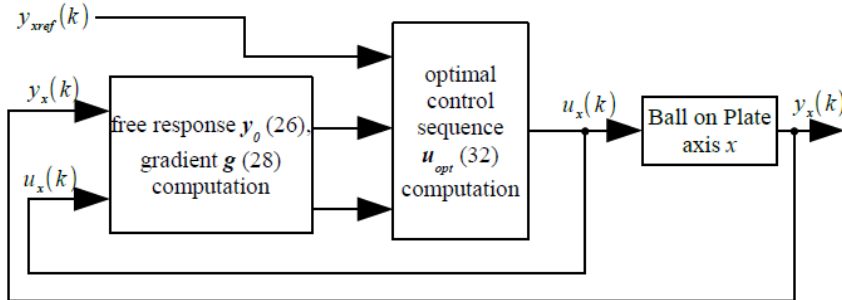


Figure 3

The predictive control of the dynamic system implemented to the control structure

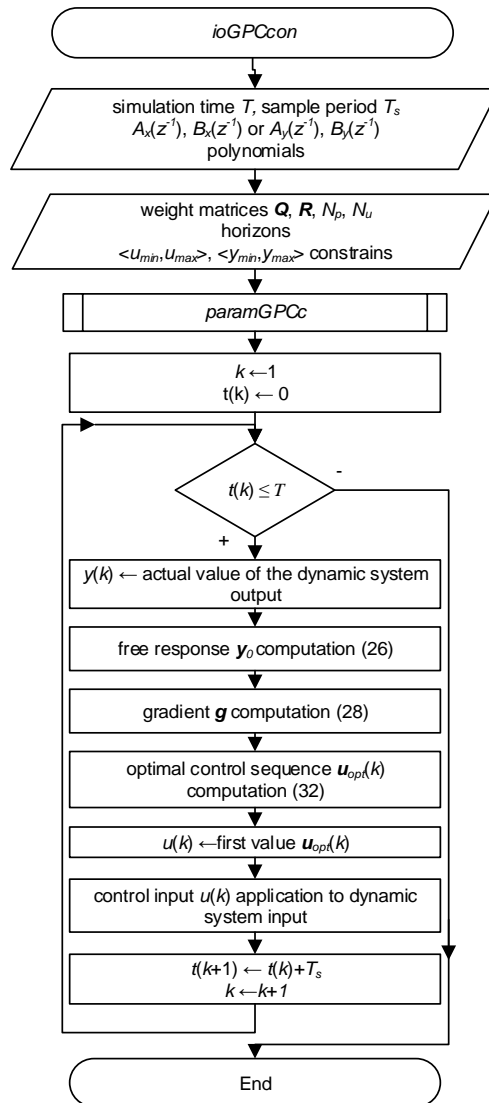


Figure 4

The predictive control of the dynamic system implemented to the control structure

The designed predictive control of the nominal system was verified by a simulation model of Ball on Plate for the circle trajectory tracking. The control inputs and ball position time responses are illustrated in Figure 5.

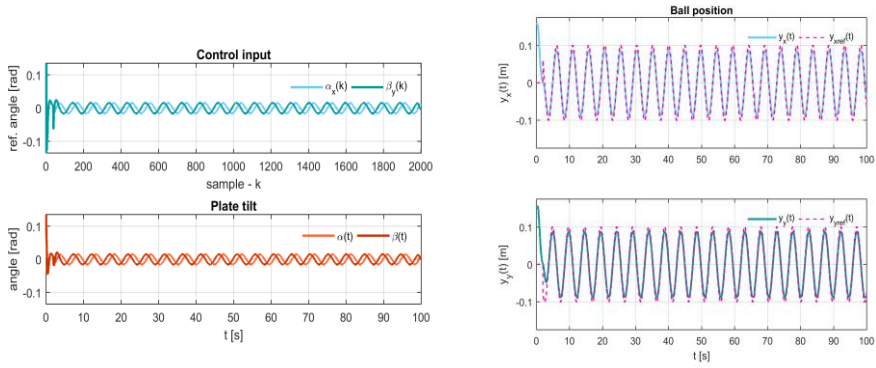


Figure 5

Time responses of nominal Ball on Plate system outputs and control inputs

After successful verification of the designed predictive control algorithm of the nominal system, the fault detection and diagnosis system design is the next step according to the presented methodology.

2.3 Actuators Fault Detection and Diagnosis System Design

The fault detection and diagnosis system is based on the group of observers with unknown inputs (UIO). For their design, it is important to rearrange the state space models (12) - (15) of the Ball on Plate system and also specify the faulty models of the system. The state space models (12) - (15) can be expressed as MIMO model of the system:

$$\underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \\ \dot{x}_5(t) \\ \dot{x}_6(t) \end{bmatrix}}_{\mathbf{x}(t)} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{5}{7}g & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{T_x} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{5}{7}g \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{T_y} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}}_{\mathbf{x}(t)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{K_x}{T_x} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{K_y}{T_y} \end{bmatrix}}_{\mathbf{B}} \mathbf{u}(t) \quad (29)$$

$$\underbrace{\begin{bmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \\ y_4(t) \\ y_5(t) \\ y_6(t) \end{bmatrix}}_{\mathbf{y}(t)} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{C}^T} \mathbf{x}(t) \quad (30)$$

which can be also discretized using sampling period $T_s = 0.05$ s:

$$\mathbf{x}(k+1) = \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & 0 & 0 \\ 0 & 0 & a_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} & a_{46} \\ 0 & 0 & 0 & 0 & a_{55} & a_{56} \\ 0 & 0 & 0 & 0 & 0 & a_{66} \end{bmatrix}}_{\mathbf{A}_d} \mathbf{x}(k) + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ b_{31} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & b_{62} \end{bmatrix}}_{\mathbf{B}_d} \mathbf{u}(k) \quad (31)$$

$$\mathbf{y}(k) = \mathbf{C}^T \mathbf{x}(k) \quad (32)$$

In the MIMO model of the system (33) - (36), the angle of the plate tilt should be possible to measure in real system by sensor (3-axis accelerometer or gyroscope).

The faulty model of the Ball on Plate system is assumed in general state space form:

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) + \mathbf{F}_d \mathbf{f}_d(k) + \mathbf{v}(k) \quad (33)$$

$$\mathbf{y}(k) = \mathbf{C}^T \mathbf{x}(k) + \mathbf{o}(k)$$

where $\mathbf{f}_d(k)$ is vector of faults magnitudes, $\mathbf{v}(k)$, $\mathbf{o}(k)$ are system and measurement noise. Also, it is assumed that faulty matrix $\mathbf{F}_d = \mathbf{B}_d$.

The faulty model of the system can be expressed as follows:

$$\mathbf{x}(k+1) = \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) + \mathbf{F}_{d,i} f_{d,i}(k) + \mathbf{F}_{d,i}^* \mathbf{f}_d^*(k) + \mathbf{v}(k) \quad (34)$$

$$\mathbf{y}(k) = \mathbf{C}^T \mathbf{x}(k) + \mathbf{o}(k)$$

where $\mathbf{F}_{d,i}$ is the i -th column of faulty matrix \mathbf{F}_d , $f_{d,i}$ is the i -th actuator fault magnitude, $\mathbf{F}_{d,i}^*$, \mathbf{f}_d^* are composed from the faulty matrix \mathbf{F}_d and vector \mathbf{f}_d without i -th column [8].

In the FDD system two faulty models of Ball on Plate system are used.

2.3.1 Unknown Input Observer Design for Actuator Fault Detection, Localization and Estimation

For actuator faults detection, localization, and estimation of Ball on Plate system two UIO (Figure 2) are used. According to [9], these assumptions have to be satisfied for UIO design:

- number of estimated faults (rank of matrix $\mathbf{F}_{d,i}$) is lower than number of system outputs (rank of matrix \mathbf{C}^T)
- $\text{rank}(\mathbf{C}^T \mathbf{F}_{d,i}) = \text{rank}(\mathbf{F}_{d,i})$

In general, the UIO has form:

$$\mathbf{x}_e(k+1) = \mathbf{F}_e \mathbf{x}_e(k) + \mathbf{T}_e \mathbf{B}_d \mathbf{u}(k) + \mathbf{L}(k) \mathbf{C}^* \mathbf{x}(k) \quad (35)$$

$$\hat{\mathbf{x}}(k) = \mathbf{x}_e(k) + \mathbf{C}^* \mathbf{H}_e \mathbf{x}(k)$$

where \mathbf{x}_e is extended state vector of the UIO and \mathbf{F}_e , \mathbf{T}_e , \mathbf{H}_e matrices are expressed:

$$\mathbf{H}_e = \mathbf{F}_{d,i} \left(\mathbf{C}^* \mathbf{F}_{d,i} \right)^+ \quad (36)$$

$$\mathbf{T}_e = \mathbf{I} - \mathbf{H}_e \mathbf{C}^* \quad (37)$$

$$\mathbf{A}_1 = \mathbf{T}_e \mathbf{A}_d \quad (38)$$

and $\mathbf{C}^* = \mathbf{C}^T$ in case of actuator fault.

The gain \mathbf{L} of UIO is iterative computed using Kalman principles as follow:

$$\mathbf{L}(k) = \mathbf{L}_1(k) + \mathbf{L}_2(k) \quad (39)$$

$$\mathbf{L}_1(k) = \mathbf{A}_1 \mathbf{P}_1(k | k-1) \mathbf{C}^{*T} \left(\mathbf{C}^* \mathbf{P}_1(k | k-1) \mathbf{C}^{*T} + \mathbf{R}_o \right)^{-1} \quad (40)$$

$$\mathbf{L}_2(k) = \mathbf{F}_e(k) \mathbf{H}_e \quad (41)$$

$$\mathbf{F}_e(k) = \mathbf{A}_1 - \mathbf{L}_1(k) \mathbf{C}^* \quad (42)$$

where the covariance matrix $\mathbf{P}_1(k|k-1)$ is computed:

$$\mathbf{P}_1(k | k-1) = \mathbf{P}_e(k | k-1) - \mathbf{L}_1(k-1) \mathbf{C}^* \mathbf{P}_e(k | k-1) \mathbf{A}_1^T \quad (43)$$

$$\mathbf{P}_e(k+1 | k) = \mathbf{A}_1 \mathbf{P}_1(k | k-1) \mathbf{A}_1^T + \mathbf{T}_e \mathbf{Q}_v \mathbf{T}_e^T + \mathbf{H}_e \mathbf{R}_o \mathbf{H}_e^T \quad (44)$$

where \mathbf{Q}_v , \mathbf{R}_o are system and measurement covariance matrices [10].

The algorithm of state estimation by UIO is expressed by in flowchart in Figure 6.

The estimated states are used for the residuals generation in form:

$$\mathbf{r}(k) = \mathbf{y}(k) - \mathbf{C}^T \hat{\mathbf{x}}(k) \quad (45)$$

Fault detection is based on GLR test using a residuals generator (49) and it can be expressed as follows:

$$J(k) = \max_{k-M_d+1 \leq i \leq k} \frac{1}{2} \left(\sum_{i=k-M_d+1}^k (\mathbf{r}(i) - \boldsymbol{\mu}_0) \right)^T \mathbf{Q}_z^{-1} \left(\sum_{i=k-M_d+1}^k (\mathbf{r}(i) - \boldsymbol{\mu}_0) \right) \quad (46)$$

where $\boldsymbol{\mu}_0$ is the mean value and \mathbf{Q}_z is matrix of variances, which are determined using the residuals obtained from the nominal system. The fault detection function $J(k)$ is used for the generation of the fault symptoms:

$$s = \begin{cases} 0, & \text{if } J(k) < J_{th} \\ 1, & \text{if } J(k) \geq J_{th} \end{cases} \quad (47)$$

where J_{th} is threshold value.

The generated symptoms s_l ($l=1,2$) are used for the fault localization (Table 2).

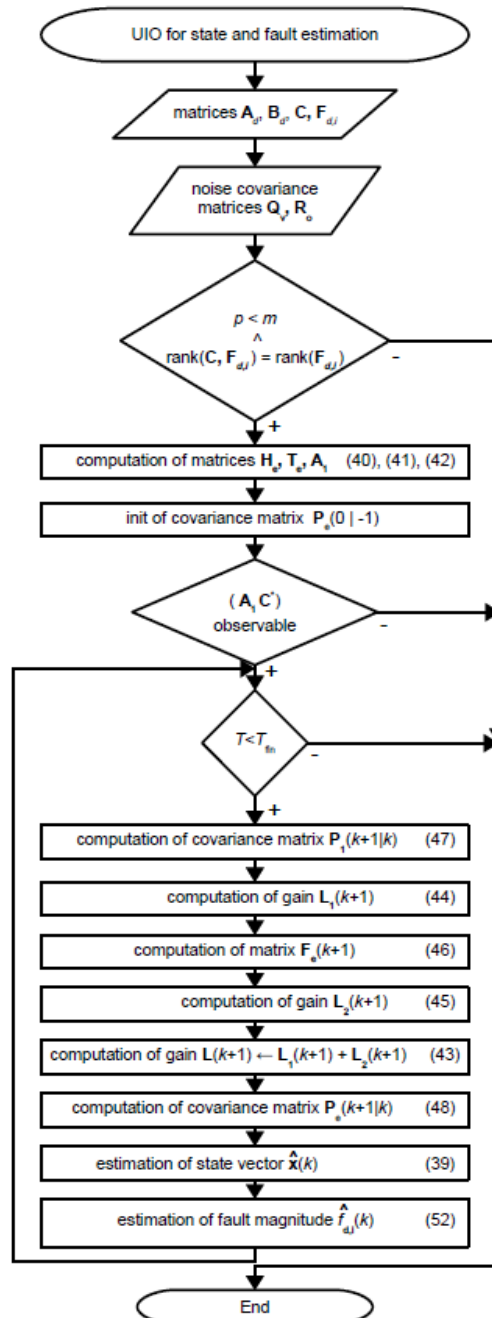


Figure 6
Flowchart of the UIO algorithm

Table 2
Localization of actuators faults using symptoms generation

	Fault free	Fault – servo of axis x	Fault – servo of axis y	Fault – both servos
s ₁	0	0	1	1
s ₂	0	1	0	1

After the successful fault detection and localization the fault magnitude can be estimated:

$$\hat{f}_{d,i}(k) = \mathbf{VS}^{-1}\mathbf{U}^T \hat{\mathbf{x}}(k) \quad (48)$$

where matrices \mathbf{U} , \mathbf{S} , \mathbf{V} are computed using SVD decomposition of matrix $\mathbf{F}_{d,i}$:

$$\mathbf{F}_{d,i} = \mathbf{U} \begin{bmatrix} \mathbf{S} \\ \mathbf{0} \end{bmatrix} \mathbf{V}^T \quad (49)$$

The results of the fault detection, localization and estimation are used in fault accommodation in the predictive control algorithm [9], [11].

2.3.2 Fault Accommodation in Predictive Control Algorithm

If the fault of the some actuator occurs during the control of the Ball on Plate system, the control inputs are affected. The influence of the fault can be eliminated using fault accommodation in control law [11].

According to [11], both control laws are extended by the additive inputs, which are determined using the results of fault detection and localization (Figure 2) [22]. Finally, the control laws in general have form:

$$\mathbf{u}(k) = \mathbf{u}_f(k) + \mathbf{u}_{ad}(k) \quad (50)$$

where:

$$\mathbf{u}(k) = \begin{bmatrix} u_x(k) \\ u_y(k) \end{bmatrix}, \quad \mathbf{u}_f = \begin{bmatrix} u_{x,f}(k) \\ u_{y,f}(k) \end{bmatrix}, \quad \mathbf{u}_{ad} = \begin{bmatrix} u_{x,ad}(k) \\ u_{y,ad}(k) \end{bmatrix} = \begin{bmatrix} -\hat{f}_{d,1}(k) \\ -\hat{f}_{d,2}(k) \end{bmatrix} \quad (51)$$

and also have to be updated constraints values:

$$\mathbf{U}_{con} = \begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix}, \quad \mathbf{v}_{con} = \begin{pmatrix} \mathbf{u}_{\max} + \mathbf{I}\hat{f}_{d,i}(k) \\ -\mathbf{u}_{\min} - \mathbf{I}\hat{f}_{d,i}(k) \end{pmatrix} \quad (52)$$

3 Simulation Verification of Actuators Faults Diagnosis System

According to FDD system design methodology presented in Section 2, it was implemented and tested in MATLAB environment using the nonlinear simulation model of the Ball on Plate system. Also, the effect of the fault accommodation in the predictive control algorithm was verified and evaluated by simulations, which are shortly introduced in this section. In the first case the multiplicative fault of the servomotor (loss of effectiveness) was assumed in a axis x direction, which occurred in $T_f = 25$ s. The effect was reflected mainly in time response of the ball position (Figure 7) and also in the second fault detection function J_2 (Figure 8). The estimated magnitude of the actuator fault (Figure 8) was used for fault accommodation in the control algorithm (Figure 9).

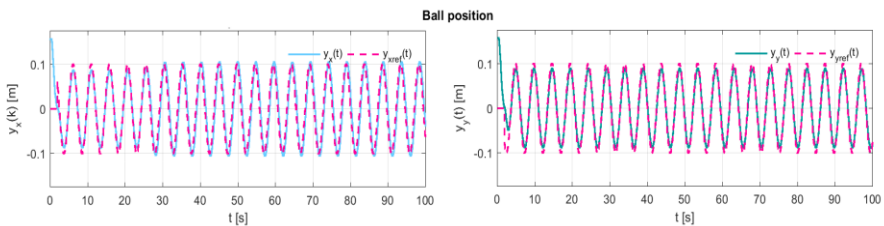


Figure 7

Time responses of B&P system outputs with servomotor multiplicative fault (axis x)

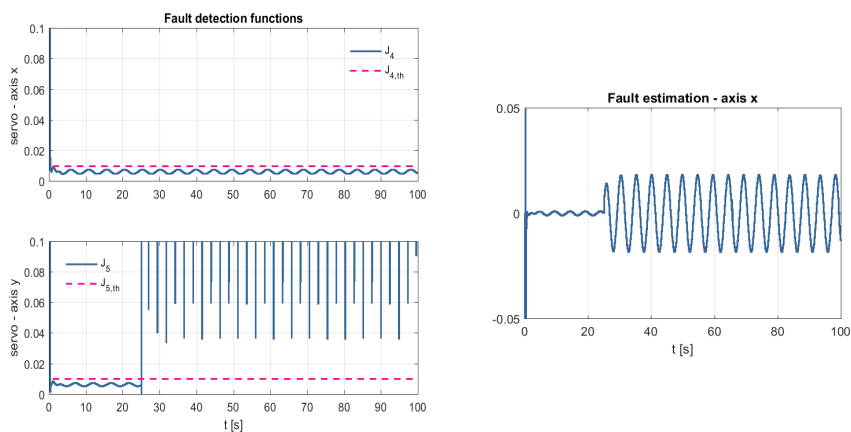


Figure 8

Detection functions and fault estimation generated by FDD system - servomotor fault (axis x)

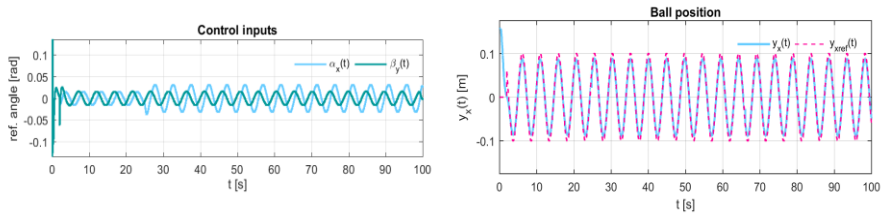


Figure 9

Time responses of Ball on Plate system output and control inputs with fault accommodation for servomotor of axis x

In the second case, the presence of multiplicative fault of the servomotor in a axis y direction (Figure 10, Figure 11) was simulated. The fault occurred in $T_f = 30$ s and its effect was eliminated by the fault accommodation using estimated magnitude (Figure 12).

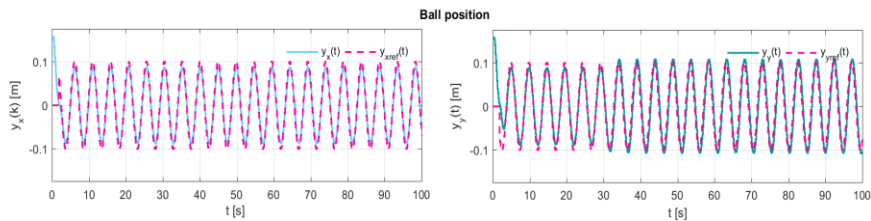


Figure 10

Time responses of B&P system outputs with servomotor multiplicative fault (axis y)

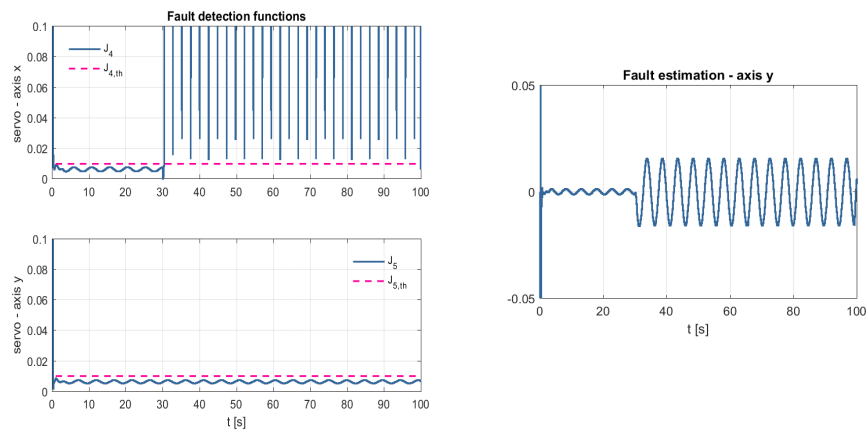


Figure 11

Detection functions and fault estimation generated by FDD system - servomotor fault (axis y)

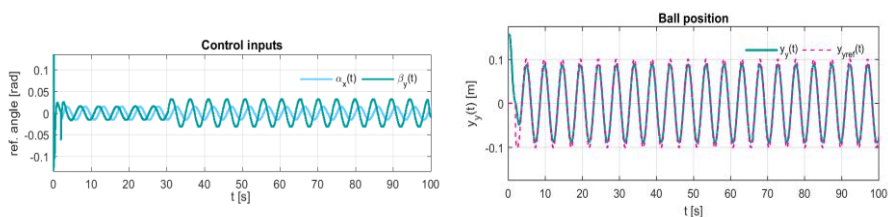


Figure 12

Time responses of Ball on Plate system output and control inputs with fault accommodation for servomotor of axis y

The results of simulations shows that the system output in both cases had higher amplitudes in axis x (Figure 7) or axis y (Figure 10) after faults occurring ($T_f=25$ s in case one or $T_f=30$ s in case two). When the fault tolerant capability was used in the predictive control algorithm, the system output responses (Figure 9, Figure 12) were very similar with no-fault case (Figure 5). The fault accommodation effect in the predictive control was also evaluated using the quantitative method. The norm of control errors in both directions were evaluated using formulas:

$$\|e_x\| = \sqrt{\sum_{k=0}^{T_{sim}/T_s} e_x^2(k)} = \sqrt{\sum_{k=0}^{T_{sim}/T_s} (y_{x,ref}(k) - y_x(k))} \tag{53}$$

$$\|e_y\| = \sqrt{\sum_{k=0}^{T_{sim}/T_s} e_y^2(k)} = \sqrt{\sum_{k=0}^{T_{sim}/T_s} (y_{y,ref}(k) - y_y(k))} \tag{54}$$

The results of the control errors computation are presented in the Table 3.

If the fault occurred, the fault tolerant predictive control of Ball on Plate system has better results than without this capability, this illustrated by laboratory model simulation outputs (Figure 7 - Figure 12) and also evaluated by norm of control error (Table 3).

Table 3
Norm of predictive control error

	nominal system	fault of servomotor - x axis		fault of servomotor - y axis	
		without fault accomodation	with fault accomodation	without fault accomodation	with fault accomodation
$\ e_x\ $	0,7942	1,1318	0,7738	0,7942	0,7942
$\ e_y\ $	0,8542	0,8542	0,8542	1,117	0,8398

The purpose of the the realized simulations was to confirm the accuracy of the proposed methodology and also capability of the designed predictive control algorithm to fault tolerant control.

Conclusions

The designed methodology for control and fault detection and diagnosis system design was presented in this paper. The steps of the presented methodology includes the modeling of the nominal system, the predictive control and the

diagnosis system design with capability of actuators faults tolerance. Finally, the results of the designed control and diagnosis systems according to the methodology were verified by simulation in the MATLAB environment. The time responses and quantitative evaluation of the simulations results shows that the FDD system is able to detect, localize, and estimate actuator fault magnitude. Also, the fault accommodation in the predictive control algorithm almost completely eliminates the effect of the actuator fault using the information about fault from the diagnosis system. The problem of the predictive control of Ball on Plate system with fault tolerance capability is not solved very often and this paper demonstrates it. In case of the Ball on Plate system, the fault accommodation is limited only to one actuator fault occurrence, but the faults of both servomotors concurrently is unlikely.

The issues mentioned in this article were also solved within the tasks of the project *University Science Park Technicom for innovative applications with knowledge technology support* (ITMS code 313011D232). The next step of the research should be oriented toward verification of the designed FDD system in a real laboratory model of Ball on Plate.

Acknowledgement

This work has been supported by grant KEGA Implementation of research results in the area of modeling and simulation of cyber-physical systems into the teaching process – development of modern university textbooks – 072TUKE - 4/2018 (100%).

References

- [1] S. Preitl *et. al.*, “Iterative feedback and learning control, servo systems applications,” IFAC Proceedings Volumes, Vol. 40, Issue 8, pp. 16-27, 2007, ISSN 1474-6670
- [2] R. G. Alvarez, Z. Johanyák, T. Kovács, “Surrogate model based optimization of traffic lights cycles and green period ratios using microscopic simulation and fuzzy rule interpolation,” *International Journal of Artificial Intelligence*, Vol. 16, Issue 1, pp. 20-40, 2018, ISSN 0974-0635
- [3] K. K. Tan, S. Zhao and J. Xu, “Online automatic tuning of a proportional integral derivative controller based on an iterative learning control approach,” *IET Control Theory & Applications*, Vol. 1, No. 1, pp. 90-96, 2007, ISSN 1751-8644
- [4] R. Andoga, L. Fozo, R. Kovács, K. Beneda, T. Moravec, M. Schreiner, “Robust Control of Small Turbojet Engines,” *Machines*, Vol. 7, Issue 1, 2019, ISSN 2075-1702

-
- [5] B. Heeseung, L. Young, "Embedded Model Predictive Control for Enhancing Tracking Performance of a Ball and Plate System," IEEE Access, Vol. 7, pp. 39652-39659, 2019, ISSN 2169-3536
- [6] B. Heeseung, L. Young, "Implementation of a Ball and Plate Control System Using Sliding Mode Control," IEEE Access, Vol. 6, pp. 32401-32408, 2018, ISSN 2169-3536
- [7] U. Musa et. al., "Position and Trajectory Tracking Control for the Ball and Plate System using Mixed Sensitivity Problem," Covenant Journal of Informatics & Communication Technology. Vol. 6, No. 1, 2018, ISSN 2354-3507
- [8] D. Theilliol, H. Noura, J.-C. Ponsart, "Fault diagnosis and accommodation of a three-tank system based on analytical redundancy," ISA transactions, Vol. 41, Issue 3, pp. 365-382, 2002, ISSN 0019-0578
- [9] P. F. Odgaard, J. Stoustrup, "Unknown input observer based scheme for detecting faults in a wind turbine converter," IFAC Proceedings Volumes, Vol. 42, Issue 8, pp. 161-166, 2009
- [10] E. F. Camacho, T. Alamo, D. Muñoz de la Peña, "Fault-tolerant model predictive control," Emerging Technologies and Factory Automation (ETFa), IEEE Conference Bilbao, Spain, pp. 1-8, 2010, ISSN: 1946-0740
- [11] T. Miksch, A. Gambier, E. Badreddin, "Real-time implementation of fault-tolerant control using model predictive control," International Federation of Automatic Control Proceedings, Vol. 41, Issue 2, pp. 11136-11141, 2008, ISBN 978-3-902661-00-5
- [12] A. R. Kodakkadan, S. Olaru, V. Reppa, "Experimental fault tolerant control of a ball-on-plate system," 21st International Conference -System Theory, Control and Computing (ICSTCC), IEEE pp. 577-582, 2017, ISBN 978-1-5386-3843-9
- [13] M. Oravec, A. Jadlovská, "Optimal State Control of the Mechatronical Laboratory Model B&P_Kyb," Electrical Engineering and Informatics 5: Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, TU, pp. 9-16, 2014, ISBN 978-80-553-1704-5
- [14] M. Oravec, A. Jadlovská, "Model predictive control of a ball and plate laboratory model," Symposium of Applied Machine Intelligence and Informatics (SAMII), IEEE 13th International Symposium, pp. 165-170, 2015, ISBN 978-1-4799-8220-2

- [15] M. Oravec, A. Jadlovská, “Intelligent positioning plate predictive control and concept of diagnosis system design,” *Journal of Manufacturing and Industrial Engineering*, Vol. 15, Issues 1-2, pp. 1-9, 2017, ISSN 1339-2972
- [16] A. W. Ordys, D. W. Clarke. “A state-space description for GPC controllers”, *International Journal of Systems Science*, Vol. 24, Issue 9, pp. 1727-1744, 1993, ISSN 1464-5319
- [17] K. Belda, J. Böhm, “Adaptive Generalized Predictive Control for Mechatronic Systems”, *WSEAS Transactions on Systems*, Athens, Greece, Vol. 5, Issue 8, pp. 1830-1837, 2006
- [18] Š. Jajčišin, A. Jadlovská, “Predictive Control Algorithms Verification on the Laboratory Helicopter Model,” *Acta Polytechnica Hungarica (Journal of Applied Sciences)*, Vol. 9, No. 4, 2012, ISSN 1785-8860
- [19] Š. Jajčišin, A. Jadlovská, “Design of predictive control algorithms using nonlinear models of physical systems,” (in Slovak), Košice: elfa, 2013, pp. 139, ISBN 978-80-8086-229-9
- [20] S. X. Ding, “Model-based fault diagnosis techniques: design schemes, algorithms and tools,” Springer Science & Business Media, 2008, ISBN 978-3-540-76303-1
- [21] J. Jadlovský, A. Jadlovská, S. Jadlovská, J. Čerkala, M. Kopčík, J. Čabala, M. Oravec, M. Varga, D. Vošček, “Research Activities of the Center of Modern Control Techniques and Industrial Informatics,” In: SAMI 2016: IEEE 14th International Symposium on Applied Machine Intelligence and Informatics: proceedings, 2016, pp. 279-285, ISBN 978-1-4673-8739-2
- [22] M. Oravec, “Methodology design for fault diagnosis of model of physical systems and its application into the distributed control system,” Dissertation Thesis. CMCT&II DCAI FEEI TU Košice, Slovakia, 2018, pp. 163 (in Slovak)