Iterative Data-Driven Tuning of Controllers for Nonlinear Systems with Constraints

Mircea-Bogdan Rădac, *Member, IEEE*, Radu-Emil Precup, *Senior Member, IEEE*, Emil M. Petriu, *Fellow, IEEE*, and Stefan Preitl, *Senior Member, IEEE*

***Abstract*—This paper presents a new iterative data-driven algorithm (IDDA) for the experiment-based tuning of controllers for nonlinear systems. The proposed IDDA solves the optimization problems for nonlinear processes while using linear controllers accounting for operational constraints and employing a quadratic penalty function approach. The search algorithm employs first-order gradient information obtained from Neural Network-based process models in order to reduce the number of experiments needed to run on real-world processes. A data-driven controller tuning for the angular position control of a nonlinear aerodynamic system is used as an experimental case study to validate the proposed IDDA.**

***Index Terms*—constrained optimization; iterative data-driven algorithm; iterative feedback tuning; iterative learning control; neural networks; penalty functions.**

I. Introduction

DATA-DRIVEN optimization techniques for controller design and tuning needing only limited measurement information to model complex processes have frequently been reported in the literature [1]–[12].

Different performance indices can be aggregated into conveniently defined cost functions (CFs). The minimization of such CFs in the framework of constrained optimization problems (OPs) can fulfill different objectives like reference trajectory tracking (including model reference tracking), control signal penalty, disturbance rejection, etc. These specific features of data-driven optimization techniques provide efficient control and monitoring solutions for many complex industrial applications.

Iterative Feedback Tuning (IFT) is a well known data-driven technique for iterative data-driven controller tuning using experiment-based updates of the controller parameters [1]. IFT needs only few experiments conducted on the real-world control system (CS) to estimate the CF gradients used in the iterative solving of the OPs.

Neural network (NN) and fuzzy modeling techniques have also been used for data-driven control, optimization, and monitoring applications [13]–[21].

Reinforcement learning, approximate dynamic programming, model-free adaptive control and their combinations using NNs, as well as supervised and unsupervised learning are other representative data-driven controller tuning techniques [12], [22]–[26]. It should be noted that all these were associated with appropriately defined OPs [27]–[30].

Building upon the recent results on Iterative Learning Control (ILC) reported in [11], this paper proposes a novel iterative data-driven algorithm (IDDA) for more efficiently solving optimal control problems while accounting for operational constraints on the control signal. The algorithm employs an experiment-based quadratic penalty function approach.

The proposed IDDA has the following characteristics, which are advantageous for the control of nonlinear systems:

- as it uses experiments conducted on the real-world CS, it can compensate for process nonlinearities and uncertainties;
- the reduced number of experiments that are required allows for a cost-effective implementation.

The paper is organized as follows: Section II formulates the iterative controller tuning problem for nonlinear processes in the framework of optimal control. Section III discusses the NN-based estimation of the gradients needed by the search algorithm. Section IV presents the model-free constrained optimal control problem and offers a formulation of IDDA using quadratic penalty functions. Section V presents the case study of an angular position controller for a laboratory nonlinear aerodynamic system, used to experimentally validate the proposed IDDA. Conclusions are highlighted in Section VI.

Manuscript received July 16, 2013; revised October 7, 2013; accepted December 4, 2013. This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS – UEFISCDI, project number PN-II-ID-PCE-2011-3-0109, and by a grant from the NSERC of Canada.

M.-B. Rădac, R.-E. Precup (corresponding author), and S. Preitl are with the Department of Automation and Applied Informatics, Politehnica University of Timisoara, 300223 Timisoara, Romania (e-mail: mircea.radac@aut.upt.ro; radu.precup@aut.upt.ro; stefan.preitl@aut.upt.ro).

E. M. Petriu is with the School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada (e-mail: petriu@eecs.uottawa.ca).

II. Problem Setting

Single Input-Single Output (SISO) discrete-time CS can be described by the nonlinear process and controller equations

$$\begin{aligned} &y(k) = P(y(k-1),..., y(k-n_{y1}), u(k-1),..., u(k-n_{u1})) + v(k), \\ &u(k) = C(\boldsymbol{\rho}, u(k-1),..., u(k-n_{u2}), y(k),..., y(k-n_{y2}), r(k), \\ &..., r(k-n_r)), \end{aligned} \quad (1)$$

where y is the process output, u is the control signal, r is the reference input, v is the zero-mean stochastic disturbance acting on the output and it can account for a large class of disturbances, and $\boldsymbol{\rho}$, $\boldsymbol{\rho} \in \mathbf{R}^{n_\rho}$, is the parameter vector of the controller. The nonlinear functions P and C make the model (1) belong to the class of nonlinear autoregressive exogenous (NARX) models treated in [31].

Several assumptions are formulated in relation with (1). The closed-loop CS is stable and the nonlinear operators P, C are smooth functions of their arguments. The nominal trajectory of the CS is denoted as $\{r_n(k), u_n(k), y_n(k)\}$, $k = 0...N$, where N is the experiment length.

A typical objective in iterative controller tuning is to search for the controller parameters that solve an OP starting with the initial solution $\boldsymbol{\rho}_0$

$$\begin{aligned} &\boldsymbol{\rho}^* = \arg \min_{\boldsymbol{\rho} \in D_S} J(\boldsymbol{\rho}), \\ &J(\boldsymbol{\rho}) = \frac{0.5}{N} E\{\sum_{k=0}^{N} [(y(k) - y^d(k))^2 + \lambda u^2(k)]\}, \end{aligned} \quad (2)$$

subject to system dynamics (1) and to operational constraints, where D_S is the stability domain of those parameter vectors $\boldsymbol{\rho}$ which ensure a stable CS [32]. The constraints can usually be formulated as inequalities imposed to $u(k)$ and $y(k)$, and to their rates with respect to time, $\Delta u(k) = u(k) - u(k-1)$ and $\Delta y(k) = y(k) - y(k-1)$, and they depend on the specific applications [33]–[42]. The formulation of the CF in (2) targets the trajectory tracking of the desired system output y^d while the control effort is also penalized by the weighting parameter $\lambda \geq 0$, and the expectation $E\{...\}$ is taken with respect to the stochastic disturbance v. The usual approach to solve the OP (2) in the unconstrained case is to employ the recursive stochastic search algorithm

$$\boldsymbol{\rho}_{j+1} = \boldsymbol{\rho}_j - \gamma_j \mathbf{R}_j^{-1} est\left\{ \left. \frac{\partial J}{\partial \boldsymbol{\rho}} \right|_{\boldsymbol{\rho} = \boldsymbol{\rho}_j} \right\}, \quad (3)$$

with the search information provided by the estimate of the gradient of the CF J with respect to the controller parameters and using, for example, second-order information as a Gauss-Newton approximation of the Hessian of the CF given in the matrix $\mathbf{R}_j$. The subscript j, $j \in \mathbf{N}$, is the current iteration number, and $\gamma_j > 0$ is the step size [1].

The main feature of IFT [1] is that the gradient information can be obtained from special experiments conducted on the closed-loop CS. These experiments avoid the use of the process model but, at the same time, they require special operating regimes that are different from the nominal ones. The experiments generate the gradients of y and u with respect to the controller parameters, namely $\partial y / \partial \boldsymbol{\rho}$ and $\partial u / \partial \boldsymbol{\rho}$, which are next used to compute both the gradient of J and the matrix $\mathbf{R}_j$. Although the linearity is assumed, the nonlinear-based procedure is also feasible according to [31]. The gradients can be estimated, as shown in [43], not by finite difference approximations for modifications of $\boldsymbol{\rho}$ but by using modified reference trajectories for small changes in the vicinity of the nominal trajectories $\delta r(k) = r(k) - r_n(k)$, $\delta u(k) = u(k) - u_n(k)$ and $\delta y(k) = y(k) - y_n(k)$. The procedure used in [31] is based on identification of linear time-varying models with least squares criterion with forgetting factor which is different from our NN-based approach.

The advantage of this approach is twofold. First, the closed-loop CS is not changed for the special purpose of obtaining the gradient estimate. Second, the experiments are carried out in the close vicinity of the nominal trajectories.

Two issues have been addressed in the literature in this context, viz. the number of gradient experiments which can be expensive for an increasing number of parameters, and the constrained approach [44]. This paper will show that the nonlinear tuning accounting for operational constraints gives good results, and it is also efficient as it requires a relatively small number of iterations and experiments.

III. Neural Network-Based Data-Driven Estimation of Gradients

A. Gradient Estimation Using Neural Networks

NNs, which are universal approximators with arbitrary accuracy for dynamic nonlinear systems, will be used to provide the gradient information need by the search algorithm. Each time the gradient information is necessary, the nonlinear map from the reference input to the process output and the nonlinear map from the reference input to the control signal can be identified using data collected under the normal experiment in which the CF is evaluated. Let these maps from r to y and from r to u be

$$\begin{aligned} &y(k) = M_{ry}(y(k-1),...y(k-n_y), r(k-1),...r(k-n_{ry})), \\ &u(k) = M_{ru}(u(k-1),...u(k-n_u), r(k-1),...r(k-n_{ru})). \end{aligned} \quad (4)$$

The variables $\partial y / \partial \rho_h$ and $\partial u / \partial \rho_h$ can then be estimated by finite difference approximations as

$$\frac{\partial \hat{y}(k)}{\partial \rho_h} = \frac{\bar{y}(k, r_n + \mu_h \delta r_h) - \bar{y}(k, r_n)}{\mu_h \delta \rho_h},$$
$$\frac{\partial \hat{u}(k)}{\partial \rho_h} = \frac{\bar{u}(k, r_n + \mu_h \delta r_h) - \bar{u}(k, r_n)}{\mu_h \delta \rho_h}, h = 1 \ldots n_\rho, k = 0 \ldots N, \tag{5}$$

where $\delta\rho_h = 1$ is considered, and the numerators are equivalent to carrying out two simulations: one with nominal controller parameter vector $\boldsymbol{\rho}$ and another one with the h^{th} controller parameter disturbed with the term $\mu_h \delta\rho_h$. The scalars μ_h are chosen to account for only small changes around the nominal reference input trajectory $\{r_n(k)\}$ where the analysis holds. The variables $\bar{y}$ and $\bar{u}$ are obtained by filtering the nominal and the disturbed reference trajectories through the nonlinear functions M_{ry} and M_{ru}, respectively.

Our approach has the following advantages:

- It is applicable to both linear and nonlinear systems, and the risk of non-desired controller parameter changes is mitigated until a descent direction is computed in order to be used in the search algorithm.
- The closed-loop operation of the CS is kept because equations (4) and (5) indicate that the gradients with respect to the controller parameter changes are obtained by changing the reference trajectory.
- The simulation with the disturbed reference input has to be conducted in the vicinity of the nominal trajectory for which the NN is trained, and this allows for using simple NN architectures with few neurons (parameters).
- The numerical differentiation issues that occur in noisy environments are mitigated because the obtained trajectories are not affected by the noisy data involved in NN training.

B. NN Training Using Iterative Learning Control

We will use NN batch learning in order to ensure a smooth operation of the learning system in terms of the controller tuning. Adaptive learning can also be employed for repetitive control actions [45] such as the ones in our paper. An ILC-based approach is developed with this respect.

We are using a feed-forward NN architecture consisting of one hidden layer with a hyperbolic tangent activation function and a single linear neuron. The input-output map is:

$$\hat{y}(k+1) = \mathbf{W}^T(k)\boldsymbol{\sigma}(\mathbf{V}(k), \mathbf{x}(k)), \tag{6}$$

where $\mathbf{W}^T = [w_0 \quad w_1 \quad \ldots \quad w_H] \in \mathbf{R}^{H+1}$ is the vector of output layer weights, $\boldsymbol{\sigma}^T = [1 \quad \sigma_1(\mathbf{V}_1^T\mathbf{x}) \quad \ldots \quad \sigma_H(\mathbf{V}_H^T\mathbf{x})]$ is the vector of hidden layer neurons outputs with the hyperbolic activation tangent activation functions $\sigma_m(x) = \tanh(x), m = 1 \ldots H$, and the superscript T indicates the matrix transposition. The first term in $\boldsymbol{\sigma}$ corresponds to the bias of the output neuron. Each hidden layer neuron is parameterized by its vector of weights $(\mathbf{V}^m)^T = [v_m^0 \quad v_m^1 \quad \ldots \quad v_m^{nu}] \in \mathbf{R}^{nu+1}$, $m = 1 \ldots H$, which multiplies the input vector $\mathbf{x}^T = [x_0 \quad x_1 \quad \ldots \quad x_{nu}]$. Each vector $\mathbf{V}^m$ includes the weight v_m^0 of the bias of m^{th} neuron. Here $nu+1$ is the number of inputs to the network, and H is the number of hidden layer neurons. The time domain index is $k = 0 \ldots N$.

The NN is treated as a nonlinear multi input-multi output dynamical system considered in the iteration domain

$$\mathbf{W}_{j+1} = \mathbf{W}_j + \mathbf{u}_j^w,$$
$$\mathbf{V}_{j+1}^i = \mathbf{V}_j^i + \mathbf{u}_j^{v^i}, i = 1 \ldots H, \tag{7}$$
$$\mathbf{Y}_j(k+1) = \mathbf{W}_j^T \boldsymbol{\sigma}(\mathbf{V}_j^i, \mathbf{x}(k)), k = 0 \ldots N,$$

where

$$\mathbf{u}_j^w = [u_j^{w0} \quad \ldots \quad u_j^{wH}]^T \in \mathbf{R}^{H+1},$$
$$\mathbf{u}_j^{v^i} = [u_j^{v^{i0}} \quad \ldots \quad u_j^{v^{inu}}]^T \in \mathbf{R}^{nu+1}, \tag{8}$$
$$\mathbf{Y}_j = [y_j(1) \quad \ldots \quad y_j(N+1)]^T \in \mathbf{R}^{N+1},$$
$$\mathbf{X}_j = [\mathbf{x}_j^T(0) \quad \ldots \quad \mathbf{x}_j^T(N)]^T \in \mathbf{R}^{(N+1)(nu+1)},$$

where j is the iteration index, $\mathbf{u}_j^w, \mathbf{u}_j^{v^i}$ are the input vectors, and the weight vectors $\mathbf{W}_j, \mathbf{V}_j^i$ previously defined are viewed as the state vectors of the dynamical system. The vector $\mathbf{X}_j$ can be regarded as a trial-repetitive time-series disturbance input, but it can be also regarded as a time-varying parameter vector of the nonlinear system (7). The vector $\mathbf{Y}_j$ is the output of the nonlinear dynamical system (7).

Using the ILC framework, the dynamical system (7) is transformed into a static map from the inputs to the outputs. ILC usually focuses on the minimization of the tracking error between the actual output and a desired output using a proper input. The desired output vector in our case is $\mathbf{Y}_d = [y_d(1) \ldots y_d(N+1)]^T \in \mathbf{R}^{N+1}$, with $y_d(k)$ – the desired process outputs for $k = 1..N+1$. Therefore, the batch training of the NN can be regarded as a supervised learning approach where the purpose is to minimize the tracking error $\mathbf{E}_j = \mathbf{Y}_j - \mathbf{Y}_d$ referred to also as training error. However, the input at each iteration can be derived in the framework of norm-optimal ILC as the solution to the OP

$$(\mathbf{u}_j^{w^*}, \mathbf{u}_j^{v^{i^*}}) = \arg \min_{\mathbf{u}_j^w, \mathbf{u}_j^{v^i}} \left\| \mathbf{E}_{j+1}^T \, \mathbf{R} \, \mathbf{E}_{j+1} + \mathbf{U}_j^T \, \mathbf{Q} \, \mathbf{U}_j \right\|_2^2, \tag{9}$$

where $\mathbf{U}_j = [(\mathbf{u}_j^w)^T \quad (\mathbf{u}_j^{v^1})^T \quad \ldots \quad (\mathbf{u}_j^{v^H})^T]^T \in \mathbf{R}^{H+1+H(nu+1)}$ is the stacked vector of inputs, $\mathbf{R} = \mathbf{R}^T \succ 0$ and $\mathbf{Q} = \mathbf{Q}^T \succ 0$ of proper dimensions are symmetric positive definite diagonal matrices, $\mathbf{E}_{j+1} = \mathbf{Y}_{j+1} - \mathbf{Y}_d$ is the tracking error at iteration

$j+1$, and $\|\bullet\|$ is the Euclidean norm of the vector $\bullet$. The penalty on $\mathbf{U}_j$ in (9) is used to prevent over-fitting.

The typical approach of nonlinear least squares is applied in order to obtain the analytical solution to the OP (9). The linearization of $y_{j+1}(k+1)=\mathbf{W}_{j+1}{}^T\boldsymbol{\sigma}(\mathbf{V}_{j+1}^i,\mathbf{x}(k)),\ k=0...N$, is carried out around $\mathbf{W}_j,\mathbf{V}_j^i$ for small variations of $\mathbf{u}_j^w,\mathbf{u}_j^{v^i}$ by considering the output as a nonlinear function of the weight vectors $y_{j+1}(k+1)=f(\mathbf{W}_{j+1},\mathbf{V}_{j+1}^i,\mathbf{x}(k)), k=0...N$, and the input vector $\mathbf{x}(k)$ as a parameter vector. The Taylor series expansion yields

$$\begin{aligned}&y_{j+1}(k+1)=\mathbf{W}_j{}^T\boldsymbol{\sigma}(\mathbf{V}_j^i,\mathbf{x}(k))+\\&[1\quad \tanh(\mathbf{V}_j^{1^T}\mathbf{x}(k))\quad ...\quad \tanh(\mathbf{V}_j^{H^T}\mathbf{x}(k))]\mathbf{u}_j^w\\&+w_j^1\frac{4}{(e^{\mathbf{V}_j^{1^T}\mathbf{x}(k)}+e^{-\mathbf{V}_j^{1^T}\mathbf{x}(k)})^2}\mathbf{x}^T(k)\mathbf{u}_j^{v^1}+...\\&+w_j^H\frac{4}{(e^{\mathbf{V}_j^{H^T}\mathbf{x}(k)}+e^{-\mathbf{V}_j^{H^T}\mathbf{x}(k)})^2}\mathbf{x}^T(k)\mathbf{u}_j^{v^H}+h.o.t.\end{aligned}\tag{10}$$

Since $y_j(k+1)=\mathbf{W}_j{}^T\boldsymbol{\sigma}(\mathbf{V}_j^i,\mathbf{x}(k))$, introducing the notations $g_i(k)=4/(e^{\mathbf{V}_j^{i^T}\mathbf{x}(k)}+e^{-\mathbf{V}_j^{i^T}\mathbf{x}(k)})^2$ and $\boldsymbol{\sigma}_j(k)=[1\quad \tanh(\mathbf{V}_j^{1^T}\mathbf{x}(k))\quad ...\quad \tanh(\mathbf{V}_j^{H^T}\mathbf{x}(k))]^T$, and neglecting the higher order terms in (10), the result is

$$\begin{aligned}&y_{j+1}(k+1)=y_j(k+1)+\boldsymbol{\sigma}_j{}^T(\mathbf{x}(k))\mathbf{u}_j^w\\&+w_j^1g_1(k)\mathbf{x}^T(k)\mathbf{u}_j^{v^1}+...+w_j^Hg_H(k)\mathbf{x}^T(k)\mathbf{u}_j^{v^H}.\end{aligned}\tag{11}$$

Then, by stacking the $N+1$ outputs over the time argument k we obtain

$$\begin{aligned}&\mathbf{Y}_{j+1}=\mathbf{Y}_j+\boldsymbol{\Psi}_j\mathbf{U}_j,\boldsymbol{\Psi}_j\in\mathbf{R}^{(N+1)\times(H+1+H(nu+1))},\\&\boldsymbol{\Psi}_j=\begin{bmatrix}\boldsymbol{\sigma}_j{}^T(\mathbf{x}(0)) & w_j^1g_1(0)\mathbf{x}^T(0) & ... & w_j^Hg_H(0)\mathbf{x}^T(0)\\ \boldsymbol{\sigma}_j{}^T(\mathbf{x}(1)) & w_j^1g_1(1)\mathbf{x}^T(1) & ... & w_j^Hg_H(1)\mathbf{x}^T(1)\\ \vdots & \vdots & \vdots & \vdots\\ \boldsymbol{\sigma}_j{}^T(\mathbf{x}(N)) & w_j^1g_1(N)\mathbf{x}^T(N) & ... & w_j^Hg_H(N)\mathbf{x}^T(N)\end{bmatrix}.\end{aligned}\tag{12}$$

Since $\mathbf{E}_{j+1}=\mathbf{Y}_{j+1}-\mathbf{Y}_d=\mathbf{Y}_j+\boldsymbol{\Psi}_j\mathbf{U}_j-\mathbf{Y}_d=\mathbf{E}_j+\boldsymbol{\Psi}_j\mathbf{U}_j$, the OP (9) can be rewritten as

$$\begin{aligned}&\mathbf{U}_j^*=\arg\min_{\mathbf{U}_j}\left\|\mathbf{U}_j^T\ \mathbf{X}\ \mathbf{U}_j+2\ \mathbf{Z}\ \mathbf{U}_j+\mathbf{E}_j^T\ \mathbf{R}\ \mathbf{E}_j\right\|_2^2,\\&\mathbf{X}=\boldsymbol{\Psi}_j{}^T\ \mathbf{R}\ \boldsymbol{\Psi}_j+\mathbf{Q},\ \mathbf{Z}=\mathbf{E}_j^T\ \mathbf{R}\ \boldsymbol{\Psi}_j.\end{aligned}\tag{13}$$

Using the matrix derivation rules with respect to vectors and noting that $\mathbf{X}$ is symmetric as $\mathbf{R}$ and $\mathbf{Q}$ are symmetric, it follows that the analytic solution to the quadratic OP (13) is

$$\mathbf{U}_j^*=-(\mathbf{X}^T)^{-1}\mathbf{Z}^T=-(\boldsymbol{\Psi}_j{}^T\ \mathbf{R}\boldsymbol{\Psi}_j+\mathbf{Q})^{-1}\boldsymbol{\Psi}_j{}^T\ \mathbf{R}\mathbf{E}_j=-\mathbf{K}_j\ \mathbf{E}_j.\tag{14}$$

The matrix $\mathbf{K}_j$ can be obtained easily as the matrices $\boldsymbol{\Psi}_j,\mathbf{E}_j$ can be computed at the current iteration. The optimal vector $\mathbf{U}_j^*$ contains the increments of the NN weights. Using the partitioning $\mathbf{K}_j=[\mathbf{K}_j^{w^T}\quad \mathbf{K}_j^{v^{1^T}}\quad ...\quad \mathbf{K}_j^{v^{H^T}}]^T$ with the dimensions $\mathbf{K}_j^w\in\mathbf{R}^{(H+1)\times(N+1)},\mathbf{K}_j^{v^i}\in\mathbf{R}^{(nu+1)\times(N+1)}$, the first two equations in (7) using the optimal inputs solving (9), are transformed into

$$\begin{aligned}&\mathbf{W}_{j+1}=\mathbf{W}_j-\mathbf{K}_j^w\mathbf{E}_j,\\&\mathbf{V}_{j+1}^i=\mathbf{V}_j^i-\mathbf{K}_j^{v^i}\mathbf{E}_j.\end{aligned}\tag{15}$$

The update laws (15) are of ILC type [46], depending on the error at the current iteration. Thus, solving (9) at each iteration the NN training has been expressed as an ILC learning scheme. The norm-optimal ILC formulation is more general since the CF also incorporates the penalty on the weights, and it allows for a degree of freedom in the learning.

IV. Constrained Optimization Algorithm Using Penalty Functions

The OP that ensures the reference trajectory tracking with control signal constraints and with control signal rate constraints is defined as

$$\begin{aligned}&\boldsymbol{\rho}^*=\arg\min_{\boldsymbol{\rho}\in D_S}J(\boldsymbol{\rho}),\ J(\boldsymbol{\rho})=\frac{0.5}{N}\sum_{k=1}^{N}[r(k)-y(k,\boldsymbol{\rho})]^2,\\&\text{subject to } u_{\min}(k)\le u(k,\boldsymbol{\rho})\le u_{\max}(k), k=1...N,\\&\Delta u_{\min}(k)\le\Delta u(k,\boldsymbol{\rho})\le\Delta u_{\max}(k), k=1...N,\end{aligned}\tag{16}$$

and it aims the minimization of the expected mean squared control error $e(k,\boldsymbol{\rho})=r(k)-y(k,\boldsymbol{\rho})$.

This type of problems can conveniently be solved in the deterministic case using the interior point barrier algorithm. As shown in [47], the constrained OP is transformed into an unconstrained OP using penalty functions. We propose the following augmented CF which accounts for inequality constraints on the control signal saturation and on the control signal rate:

$$\begin{aligned}
&\widetilde{J}_{p_j}(\boldsymbol{\rho}) = J(\boldsymbol{\rho}) + p_j \phi(\boldsymbol{\rho}), \\
&\phi(\boldsymbol{\rho}) = 0.5\sum_{m=1}^{c}\{[\max\{0,-q_m(\boldsymbol{\rho})\}]^2\}, \\
&\mathbf{q}(\boldsymbol{\rho}) = [u_{\max}(1) - u(1,\boldsymbol{\rho}) \quad \ldots \quad u_{\max}(N) - u(N,\boldsymbol{\rho}) \\
&u(1,\boldsymbol{\rho}) - u_{\min}(1) \quad \ldots \quad u(N,\boldsymbol{\rho}) - u_{\min}(N) \\
&\Delta u_{\max}(1) - \Delta u(1,\boldsymbol{\rho}) \quad \ldots \quad \Delta u_{\max}(N) - \Delta u(N,\boldsymbol{\rho}) \\
&\Delta u(1,\boldsymbol{\rho}) - \Delta u_{\min}(1) \quad \ldots \quad \Delta u(N,\boldsymbol{\rho}) - \Delta u_{\min}(N)]^T \in \mathbf{R}^{4N},
\end{aligned} \tag{17}$$

where the positive and strictly increasing sequence of penalty parameters $\{p_j\}_{j\geq 0}$, $p_j \to \infty$, guarantees that the minimum of the sequence of augmented CFs $\{\widetilde{J}_{p_j}(\boldsymbol{\rho})\}_{j\geq 0}$ will converge to the solution to the constrained OP (16), m, $m = 1...c$, is the constraint index, $q_m(\boldsymbol{\rho}) > 0$ is the m^{th} constraint. The OP (17) is solved using a stochastic approximation algorithm which makes use of the experimentally obtained gradient of $\widetilde{J}_{p_j}(\boldsymbol{\rho})$.

The quadratic penalty function $\phi(\boldsymbol{\rho})$ in (17) uses the maximum function which in this case is non-differentiable only at zero. Given that $\phi(\boldsymbol{\rho})$ is Lipschitz and non-differentiable at a set of points of zero Lebesgue measure, the algorithm visits the zero-measure set with probability zero when a normal distribution for the noise is assumed [47]. Therefore, using

$$\frac{\partial[\max\{0,-q_m(\boldsymbol{\rho})\}]^2}{\partial \rho_h} = -2\max\{0,-q_m(\boldsymbol{\rho})\}\frac{\partial q_m(\boldsymbol{\rho})}{\partial \rho_h}, \tag{18}$$

the expression of the gradient of the CF $\widetilde{J}_{p_j}(\boldsymbol{\rho})$ given in (17) at the current iteration *j* with respect to the parameter ρ_h is

$$\frac{\partial \widetilde{J}_{p_j}(\boldsymbol{\rho})}{\partial \rho_h} = \frac{\partial J(\boldsymbol{\rho})}{\partial \rho_h} - p_j \sum_{m=1}^{c}\{\max\{0,-q_m(\boldsymbol{\rho})\}\frac{\partial q_m(\boldsymbol{\rho})}{\partial \rho_h}\}. \tag{19}$$

The first term in (19) corresponding to the gradient of the original CF requires the knowledge of the gradient $\partial y(k)/\partial \boldsymbol{\rho}$, and the second term in (19) requires the gradients of $u(k)$ and $\Delta u(k)$ with respect to $\boldsymbol{\rho}$. All these variables can be estimated using the NN-based mechanism given in (5). The derivative of the control signal rate with respect to the parameter vector $\boldsymbol{\rho}$ is estimated using the finite differences approximation approach for the sampling period δt

$$\frac{\partial \Delta\hat{u}(k)}{\partial \rho_h} = \frac{1}{\delta t}[\frac{\partial \hat{u}(k)}{\partial \rho_h} - \frac{\partial \hat{u}(k-1)}{\partial \rho_h}], h = 1...n_\rho, k = 1...N. \tag{20}$$

The proposed IDDA algorithm consists of the following steps:

Step S1. Start with the initial $\boldsymbol{\rho}$ referred to as $\boldsymbol{\rho}_0$. Choose the upper and lower bounds for the control signal, the upper and lower bounds for the control input rate and the nominal reference input signal. Choose the tolerance tol_N for stopping the stochastic search algorithm. Set the iteration number for $\boldsymbol{\rho}$ and $\{p_j\}_{j\geq 0}$ to $j = 0$. Choose the sequence $\{p_j\}_{j\geq 0}$ and γ_0.

Step S2. Conduct the normal experiment with the current $\boldsymbol{\rho}_j$ for the nominal reference input. Evaluate the objective function $\widetilde{J}(\boldsymbol{\rho}_j)$. Train the models M_{ry}^j and M_{ru}^j at the current iteration using proposed ILC approach.

Step S3. Calculate the disturbed reference trajectories $\{\delta r_h(k)\}$ to be used in (5) and use M_{ry}^j and M_{ru}^j to estimate $\partial\hat{y}(k)/\partial\boldsymbol{\rho}$, $\partial\hat{u}(k)/\partial\boldsymbol{\rho}$ and $\partial\Delta\hat{u}(k)/\partial\boldsymbol{\rho}$ using (5) and (20). Evaluate the gradient of the CF using (19).

Step S4. Calculate the next controller parameter vector $\boldsymbol{\rho}_{j+1}$

$$\boldsymbol{\rho}_{j+1} = \boldsymbol{\rho}_j - \gamma_j est\{\left.\frac{\partial \widetilde{J}}{\partial \boldsymbol{\rho}}\right|_{\boldsymbol{\rho}=\boldsymbol{\rho}_j}\}. \tag{21}$$

Step S5. If the gradient search has converged in terms of the maximum allowed decrease of the CF, $\widetilde{J}(\boldsymbol{\rho}_j) - \widetilde{J}(\boldsymbol{\rho}_{j-1}) < tol_N$, stop the algorithm. Otherwise set $j = j+1$ and jump to S2.

V. Experimental Case Study

The case study deals with the angular positioning of the vertical motion of a twin-rotor aero-dynamical system experimental setup [48]. A rigid beam supports at one end a horizontal rotor which produces vertical motion and at the other end a vertical rotor causing horizontal motion. The horizontal position is considered fixed in this case study. The nonlinear equations that describe the vertical motion are [49]

$$\begin{aligned}
&J_v\dot{\Omega}_v = l_m F_v(\omega_v) - \Omega_v k_v + g[(A-B)\cos\alpha_v - C\sin\alpha_v], \\
&\dot{\alpha}_v = \Omega_v, \\
&I_v\dot{\omega}_v = M(U_v) - M_r(\omega_v),
\end{aligned} \tag{22}$$

where $U_v\,(\%) = u$ is the control signal represented by the PWM duty-cycle corresponding to the input voltage range of the DC motor, $-24\text{ V} \leq u \leq 24\text{ V}$, $\omega_v\,(rad/s)$ is the angular speed of the rotor, $\alpha_v\,(rad) = y$ is the process output corresponding to the pitch angle of the beam which supports the main and the tail rotor, $\Omega_v\,(rad/s)$ is the angular velocity of the beam. The expressions of the other parameters and variables related to (22) are given in [49], and the parameter values are [48], [50]

$$\begin{aligned}
&J_v = 0.02421\,kg\,m^2,\; I_v = 4.5\cdot 10^{-5}\,kg\,m^2, \\
&k_v = 0.0127\,\; kg\,m^2/s,\; B-A = 0.05\,\; rad\,kg\,m, \\
&l_m = 0.2\,m,\; C = 0.0936\,\; rad\,kg\,m.
\end{aligned} \tag{23}$$

The nonlinear model (22) is not used in the tuning process except for obtaining an initial controller which can also be obtained by model-free approaches using the Ziegler-Nichols's tuning method or Virtual Reference Feedback Tuning [2].

Next we use a linear PID controller with the transfer function $H(q^{-1}) = (\rho_1 + \rho_2 q^{-1} + \rho_3 q^{-2})/(1-q^{-1})$ and with the parameter vector $\boldsymbol{\rho} = [\rho_1 \quad \rho_2 \quad \rho_3]^T$ for this SISO CS. The initial $\boldsymbol{\rho}$ is $\boldsymbol{\rho}_0 = [0.01185 \quad 0.00080 \quad -0.00025]^T$. The controller is tuned to solve the OP (2) with $\lambda = 0$ and with the two sets of constraints, $-0.1 \le u(k) \le 0.22$ and $-0.04 \le \Delta u(k) \le 0.04$. The sampling period is set to 0.1 s and the experiment length is 90 s, i.e., $N = 900$; therefore, $4 \cdot 900 = 3600$ inequality constraints are generated for the control signal and for the control signal rate. The desired trajectory specified as a reference input to the CS is a step of amplitude 0.2 rad (approximately 11.45°) for 45 s and next zero for the remaining 45 s.

The NN architecture used for the gradient estimation consists of one hidden layer with six neurons and one output layer with one neuron. A hyperbolic tangent function is employed as the hidden layer activation function, and a linear function is employed as the output neuron activation function. This NARX architecture uses the last two outputs and the last two inputs in order to obtain the output prediction. The same simple architecture is used for both M_{ry} and M_{ru}. The inputs of the two NNs are $\mathbf{x}_{ry}^T(k) = [1 \quad y(k) \quad y(k-1) \quad r(k) \quad r(k-1)]$ for M_{ry} and $\mathbf{x}_{ru}^T(k) = [1 \quad u(k) \quad u(k-1) \quad r(k) \quad r(k-1)]$ for M_{ru}. The outputs of the NNs are process output $y(k)$ and the control signal $u(k)$ from (4).

The training of the two NARX architectures is carried out in the ILC framework using the guidelines from Section III.B. Each neuron in the hidden layer has five parameters, i.e., four weights and one bias. The output layer has seven weights including the bias. We trained the weight vectors $\mathbf{W} \in \mathbf{R}^{7\times1}$ and $\mathbf{V}_i \in \mathbf{R}^{5\times1}, i = 1...6$. The initial values of the hidden neurons parameters are chosen from a normal distribution centered at zero with variance 1. Because of the special structure of the NN which is linear in the output weights vector **W**, a least squares initialization of **W** was performed.

The NN-based identification is carried out on the nominal trajectories of the closed-loop CS for the initial controller parameters $\boldsymbol{\rho}_0$. Only the results concerning the identified map M_{ry} are given here. For an experiment of 90 s, 898 samples are used for training. For the norm-optimal ILC problem, the weighting matrices were chosen as $\mathbf{R} = \mathbf{I}_{898}$ and $\mathbf{Q} = 0.0005 \cdot \mathbf{I}_{37}$, where $\mathbf{I}_\zeta$ is the general notation for the ζ^{th} order identity matrix. The training error throughout the iterations of our algorithm together with the initial error and final error after ILC-based scheme are shown in Fig. 1. Fig. 1 illustrates the identification of M_{ry} for the first IFT iteration with the initial controller in the loop.

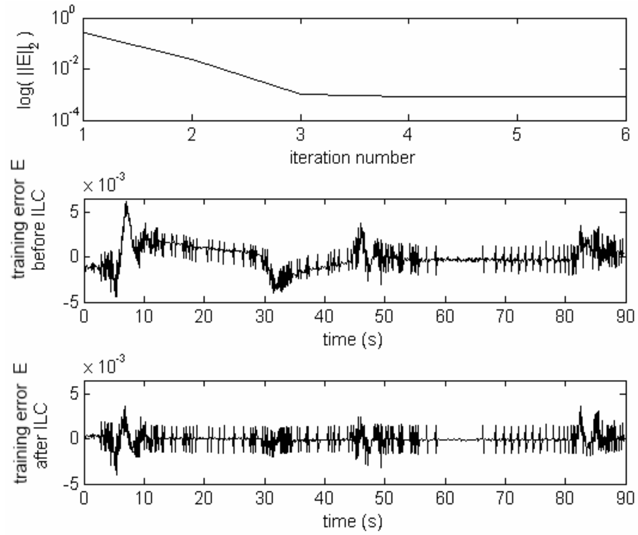

Fig. 1. Network training error for M_{ry} at the first iteration of the algorithm.

The initial training error norm results after the least squares initialization of the output weights vector **W**. The results with the training of the neural network using the ILC framework show a decrease of the training error of about three orders of magnitude in four iterations. Thus the learning in the ILC framework is feasible for the current NN architecture.

The IDDA is run for 20 iterations. The step size sequence in (3) is chosen as $\gamma_j = \gamma_0 / j^{0.65}$, $\gamma_0 = 0.003$, and the sequence $\{p_j\}_{j\ge0}$ as $p_j = 0.007 \cdot j^{0.7}$. The matrix $\mathbf{R}_j$ is set as the identity matrix. The final values of the controller parameters after optimization are $\boldsymbol{\rho}_{20} = [0.01480 \quad 0.00360 \quad 0.00250]^T$.

The evolution of the CF throughout the iterations is presented in Fig. 2 along with the evolutions of the penalty function and of the controller parameters. The evolutions of the pitch position, control signal and control signal rate over 20 iterations of our IDDA are given in Fig. 3.

Fig. 3 illustrates that the penalty function tends to get large and is weighted more in the CF due to the sequence $\{p_j\}_{j\ge0}$. However, the gradient of the CF accounts for the constraint violation and drives the controller parameters towards the minimization of the CF. In the long run, the penalty function is weighted more and the search continues until both the control error is minimized and the constraints are fulfilled.

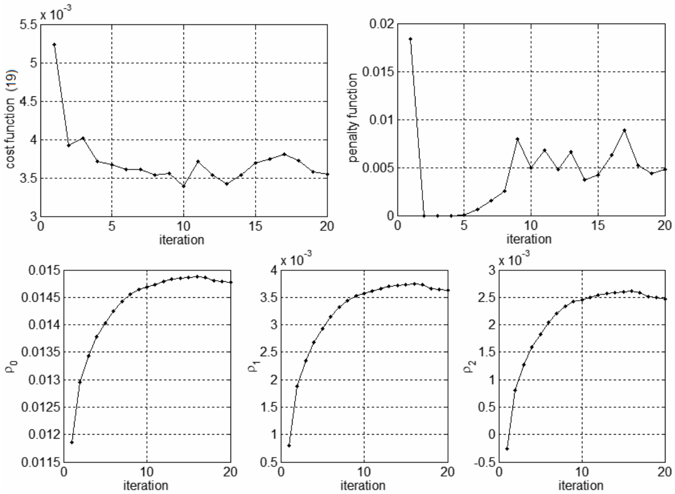

Fig. 2. Cost function, penalty function and controller parameters versus iteration number.

Fig. 3 clearly shows the improvement of the CS behavior after only 20 iterations that correspond to 20 experiments. The improvements are visible even at the second iteration. At the final iteration, the control signal violates the lower bound constraint only mildly. The control signal rate increases from one iteration to another until it reaches the upper and lower bounds constraints and violates them only mildly. The behavior has to be correlated with the strong nonlinearity of the process and with the slightly chaotic behavior shown in the experiments which is due to the static frictions in the axis of the moving mechanical parts. In the up-lifting motion the aerodynamic thrust has to compensate for the gravitation effect whereas for the down-lifting motion the gravity helps. Other discussions can be formulated for different nonlinear processes [52–58].

VI. Conclusion

The proposed IDDA for controller tuning has the following advantages: (i) other integral-type constraints can be added to the OP without additional experiments and without an increase in complexity, (ii) nonlinear controllers can easily be incorporated in this framework enhancing thus its generality. (iii) the model needs to be valid only around the nominal trajectory where the gradients are generated, and not within a wide operating range, (iv) it conveniently deals with numerical differentiation issues in noisy environments.

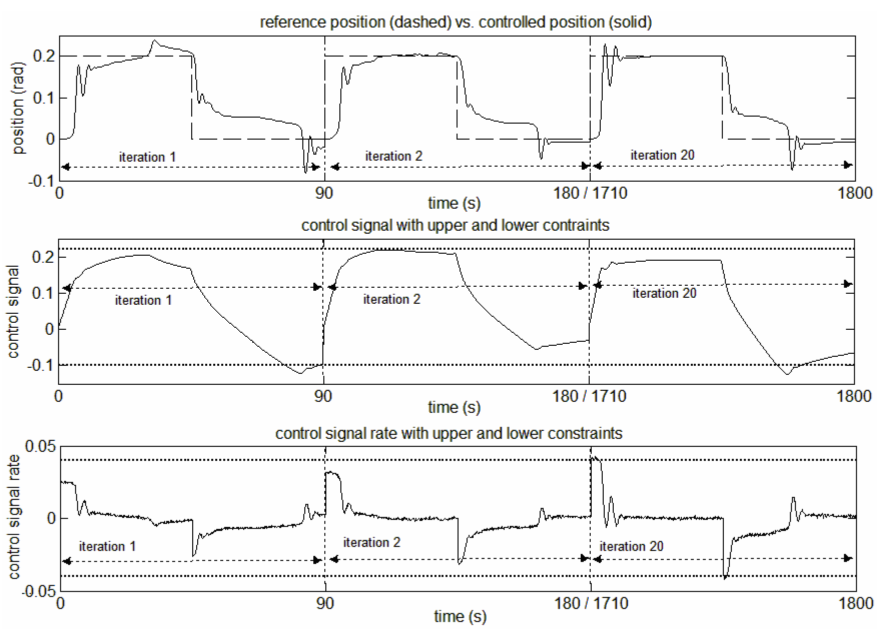

Fig. 3. Controlled output, control signal and control signal rate throughout 20 iterations of IDDA. The constraints are illustrated with dotted lines.

Experiments have demonstrated the validity of the proposed NN-based estimation of CF gradients. The batch training in an ILC framework was also demonstrated. Other NN architectures, such as the resource-allocating networks including radial basis activation functions, with or without dynamic hidden unit allocation, can also be used [51]. The optimization approach using quadratic penalty functions ensures the operation in a stochastic environment. Additive noise applied to the reference input can be used in order to provide sufficient excitation for the NN-based identification.

Future research will focus on the development of appropriate tools for stability analysis. The stability can be indirectly dealt with by ensuring that the steps of the search algorithm are small enough and always conducted in the negative direction of the gradient and also including a penalty on the control energy in the original cost function. Using the numerical optimization approach, the convergence to the global optimum will further be addressed. A comparison of the proposed algorithm's performance relative to other known numerical solvers for nonlinear constrained optimization also needs to be carried out in the future.

References

[1] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, "Iterative feedback tuning: theory and applications," *IEEE Control Syst. Mag.*, vol. 18, pp. 26–41, Aug. 1998.

[2] M. C. Campi, A. Lecchini, and S. M. Savaresi, "Virtual reference feedback tuning: a direct method for the design of feedback controllers," *Automatica*, vol. 38, pp. 1337–1346, Aug. 2002.

[3] D. Wang, "Robust data-driven modeling approach for real-time final product quality prediction in batch process operation," *IEEE Trans. Ind. Informat.*, vol. 7, pp. 371–377, May 2011.

[4] M.-B. Rădac, R.-E. Precup, E. M. Petriu, and S. Preitl, "Application of IFT and SPSA to servo system control," *IEEE Trans. Neural Netw.*, vol. 22, pp. 2363–2375, Dec. 2011.

[5] S. Yin, S. X. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *J. Process Control*, vol. 22, pp. 1567–1581, Oct. 2012.

[6] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.

[7] S. Yin, S. X. Ding, A. Haghani, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *Int. J. Syst. Science*, vol. 44, pp. 1366–1376, Jul. 2013.

[8] H. Saxen, C. Gao, and Z. Gao, "Data-driven time discrete models for dynamic prediction of the hot metal silicon content in the blast furnace - a review," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 2213–2225, Nov. 2013.

[9] S. Formentin and A. Karimi, "A data-driven approach to mixed-sensitivity control with application to an active suspension system," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 2293–2300, Nov. 2013.

[10] R. Chi, Z. Hou, S. Jin, D. Wang, and J. Hao, "A data-driven iterative feedback tuning approach of ALINEA for freeway traffic ramp metering with PARAMICS simulations," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 2310–2317, Nov. 2013.

[11] M.-B. Rădac, R.-E. Precup, E. M. Petriu, S. Preitl, and C.-A. Dragoş, "Data-driven reference trajectory tracking algorithm and experimental validation," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 2327–2336, Nov. 2013.

[12] S. Yin, H. Luo, and S. Ding, "Real-time implementation of fault-tolerant control systems with performance optimization," *IEEE Trans. Ind. Electron.*, vol. 61, pp. 2402–2411, May 2014.

[13] Z. Petres, P. Baranyi, P. Korondi, and H. Hashimoto, "Trajectory tracking by TP model transformation: Case study of a benchmark problem," *IEEE Trans. Ind. Electron.*, vol. 54, pp. 1654–1663, Jun. 2007.

[14] J. Vaščák, "Approaches in adaptation of fuzzy cognitive maps for navigation purposes," in *Proc. 8th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2008)*, Herľany, Slovakia, 2008, pp. 31–36.

[15] J. Vaščák, "Adaptation of fuzzy cognitive maps by migration algorithms," *Kybernetes*, vol. 41, pp. 429-443, Mar. 2012.

[16] O. Linda and M. Manic, "Monotone centroid flow algorithm for type reduction of general type-2 fuzzy sets," *IEEE Trans. Fuzzy Syst.*, vol. 20, pp. 805–819, Oct. 2012.

[17] R.-J. Lian, "Enhanced adaptive self-organizing fuzzy sliding-mode controller for active suspension systems," *IEEE Trans. Ind. Electron.*, vol. 59, 958–968, Aug. 2012.

[18] R.-E. Precup, R.-C. David, E. M. Petriu, S. Preitl, and M.-B. Rădac, "Fuzzy control systems with reduced parametric sensitivity based on simulated annealing," *IEEE Trans. Ind. Electron.*, vol. 59, pp. 3049–3061, Aug. 2012.

[19] E. Kayacan, O. Cigdem, and O. Kaynak, "Sliding mode control approach for online learning as applied to type-2 fuzzy neural networks and its experimental evaluation," *IEEE Trans. Ind. Electron.*, vol. 59, pp. 3510–3520, Sep. 2012.

[20] Z. C. Johanyák and O. Papp, "A hybrid algorithm for parameter tuning in fuzzy model identification," *Acta Polyt. Hung.*, vol. 9, pp. 153–165, Dec. 2012.

[21] D. Flieller, N. K. Nguyen, P. Wira, G. Sturtzer, D. Abdeslam, and J. Merckle, "A self-learning solution for torque ripple reduction for non-sinusoidal permanent magnet motor drives based on artificial neural networks," *IEEE Trans. Ind. Electron.*, vol. 61, pp. 655–666, Feb. 2014.

[22] S. Mohagheghi, Y. del Valle, G. K. Venayagamoorthy, and R. G. Harley, "A proportional-integrator type adaptive critic design-based neurocontroller for a static compensator in a multimachine power system," *IEEE Trans. Ind. Electron.*, vol. 54, pp. 86–96, Feb. 2007.

[23] C.-F. Juang and C.-H. Hsu, "Reinforcement ant optimized fuzzy controller for mobile-robot wall-following control," *IEEE Trans. Ind. Electron.*, vol. 56, pp. 3931–3940, Oct. 2009.

[24] A. Heydari and S. N. Balakrishnan, "Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 24, pp. 145–157, Jan. 2013.

[25] H. Xu and S. Jagannathan, "Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 24, pp. 471–484, Mar. 2013.

[26] D. Liu and Q. Wei, "Finite-approximation-error-based optimal control approach for discrete-time nonlinear systems," *IEEE Trans. Cybern.*, vol. 43, pp. 779–789, Apr. 2013.

[27] S. Oblak, I. Škrjanc, and S. Blažič, "Fault detection for nonlinear systems with uncertain parameters based on the interval fuzzy model," *Eng. Appl. Artif. Intell.*, vol. 20, pp. 503–510, Jun. 2007.

[28] Z. C. Johanyák and O. Papp, "Benchmark based comparison of two fuzzy rule base optimization methods," in *Applied Computational Intelligence in Engineering and Information Technology*, R-E. Precup, S. Kovács, S. Preitl, and E. M. Petriu, Eds. Berlin, Heidelberg: Springer-Verlag, Topics in Intelligent Engineering and Informatics, vol. 1, 2012, pp. 83–94.

[29] W. Sun, Z. Zhao, and H. Gao, "Saturated adaptive robust control for active suspension systems," *IEEE Trans. Ind. Electron.*, vol. 60, pp. 3889–3896, Sep. 2013.

[30] P. Baranyi, Y. Yam, and P. Varlaki, *TP Model Transformation in Polytopic Model-Based Control*. Boca Raton, FL: Taylor & Francis, 2013.

[31] J. Sjöberg, P.-O. Gutman, M. Agarwal, and M. Bax, "Nonlinear controller tuning based on a sequence of identifications of linearized timevarying models," *Control Eng. Pract.*, vol. 17, pp. 311–321, Feb. 2009.

[32] M.-B. Rădac, R.-E. Precup, E. M. Petriu, B.-S. Cerveneak, C.-A. Dragoş, and S. Preitl, "Stable iterative correlation-based tuning algorithm for servo systems," in *Proc. 38th Annual Conference of IEEE Industrial Electronics Society (IECON 2012)*, Montreal, QC, Canada, 2012, pp. 2500–2505.

[33] T. Orlowska-Kowalska and K. Szabat, "Damping of torsional vibrations in two-mass system using adaptive sliding neuro-fuzzy approach," *IEEE Trans. Ind. Informat.*, vol. 4, pp. 47–57, Feb. 2008.

[34] W. Sun, H. Gao, and O. Kaynak, "Finite frequency H_∞ control for vehicle active suspension systems," *IEEE Trans. Contr. Syst. Technol.*, vol. 19, pp. 416–422, Mar. 2011.

[35] X. Yang, X. Cao, and H. Gao, "Sampled-data control for relative position holding of spacecraft rendezvous with thrust nonlinearity," *IEEE Trans. Ind. Electron.*, vol. 59, pp. 1146–1153, Feb. 2012.

[36] M. Bošnak, D. Matko, and S. Blažič, "Quadrocopter control using an on-board video system with off-board processing," *Robot. Auton. Syst.*, vol. 60, pp. 657–667, Apr. 2012.

[37] C. Buccella, C. Cecati, and H. Latafat, "Digital control of power converters - A survey," *IEEE Trans. Ind. Informat.*, vol. 8, pp. 437–447, Aug. 2012.

[38] R. C. Luo and C. C. Lai, "Enriched indoor map construction based on multisensor fusion approach for intelligent service robot," *IEEE Trans. Ind. Electron.*, vol. 59, pp. 3135–3145, Aug. 2012.

[39] K. Szabat and T. Orlowska-Kowalska, "Application of the Kalman filters to the high-performance drive system with elastic coupling," *IEEE Trans. Ind. Electron.*, vol. 59, pp. 4226–4235, Nov. 2012.

[40] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems - A survey," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 403–416, Feb. 2013.

[41] S. X. Ding, P. Zhang, S. Yin, and E. L. Ding, "An integrated design framework of fault-tolerant wireless networked control systems for industrial automatic control applications," *IEEE Trans. Ind. Informat.*, vol. 9, pp. 462–471, Feb. 2013.

[42] S. John and J. O. Pedro, "Neural network-based adaptive feedback linearization control of antilock braking system," *Int. J. Artif. Intell.*, vol. 10, pp. 21–40, Mar. 2013.

[43] J. Sjöberg, F. De Bruyne, M. Agarwal, B. D. O. Anderson, M. Gevers, F. J. Kraus, and N. Linard, "Iterative controller optimization for nonlinear systems," *Control Eng. Pract.*, vol. 11, pp. 1079–1086, Sep. 2003.
[44] U. S. Park, Y. Yamada, and Y. Nakabo, "Force control with safety constraints via iterative feedback tuning," in *Proc. IEEE International Conference on Robotics and Automation (ICRA '09)*, Kobe, Japan, 2009, pp. 3670–3675.
[45] S.-L. Dai, C. Wang, and F. Luo, "Identification and learning control of ocean surface ship using neural networks," IEEE *Trans. Ind. Informat.*, vol. 8, pp. 801–810, Nov. 2012.
[46] D. H. Owens, C. T. Freeman, and V. D. Thanh, "Norm-optimal iterative learning control with intermediate point weighting: theory, algorithms, and experimental evaluation," *IEEE Trans. Control Syst. Technol.*, vol. 21, pp. 999–1007, Mar. 2013.
[47] I.-J. Wang and J. C. Spall, "Stochastic optimization with inequality constraints using simultaneous perturbations and penalty functions," *Int. J. Control*, vol. 81, pp. 1232–1238, Aug. 2008.
[48] *Two Rotor Aerodynamical System, User's Manual*. Krakow, Poland: Inteco Ltd., 2007.
[49] H. Nguyen, "Modelling, simulation and calibration of twin rotor system," M.Sc. thesis, Polyt. Univ. Catalonia, Barcelona, Spain, 2007.
[50] M.-B. Rădac, R.-C. Roman, R.-E. Precup, E. M. Petriu, C.-A. Dragoş, and S. Preitl, "Data-based tuning of linear controllers for MIMO twin rotor systems," in *Proc. IEEE Region 8 EuroCon 2013 Conference*, Zagreb, Croatia, 2013, pp. 1915–1920.
[51] M. L. Corradini, V. Fossi, A. Giantomassi, G. Ippoliti, S. Longhi, and G. Orlando, "Minimal resource allocating networks for discrete time sliding mode control of robotic manipulators," *IEEE Trans. Ind. Informat.*, vol. 8, pp. 733–745, Apr. 2012.
[52] A. Manenti, A. Abba, A. Merati, S. M. Savaresi, and A. Geraci, "A new BMS architecture based on cell redundancy," *IEEE Trans. Ind. Electron.*, vol. 58, pp. 4314–4322, Sep. 2011.
[53] L. G. Guzmán, A. S. Gómez, C. J. Ardila, and D. Jabbaro, "Adaptation of the GRASP algorithm to solve a multiobjective problem using the Pareto concept," *Int. J. Artif. Intell.*, vol. 11, pp. 222–236, Oct. 2013.
[54] K. Asawarungsaengkul, T. Rattanamanee, and T. Wuttipornpun, "A multi-size compartment vehicle routing problem for multi-product distribution: models and solution procedures," *Int. J. Artif. Intell.*, vol. 11, pp. 237–256, Oct. 2013.
[55] K. Lamár and J. Neszveda, "Average probability of failure of aperiodically operated devices," *Acta Polyt. Hung*., vol. 10, pp. 153–167, Dec. 2013.
[56] M. Monfared, S. Golestan, and J. M. Guerrero, "Analysis, design, and experimental verification of a synchronous reference frame voltage control for single-phase inverters," *IEEE Trans. Ind. Electron.*, vol. 61, pp. 258–269, Jan. 2014.
[57] S. I. Han and J. M. Lee, "Fuzzy echo state neural networks and funnel dynamic surface control for prescribed performance of a nonlinear dynamic system," *IEEE Trans. Ind. Electron.*, vol. 61, pp. 1099–1112, Feb. 2014.
[58] Y. Maeda and M. Iwasaki, "Mode switching feedback compensation considering rolling friction characteristics for fast and precise positioning," *IEEE Trans. Ind. Electron.*, vol. 61, pp. 1123–1132, Feb. 2014.

Mircea-Bogdan Rădac (M'12) received the Dipl.Ing. degree in systems and computer engineering and the Ph.D. degree in systems engineering from the "Politehnica" University of Timisoara (PUT), Timisoara, Romania, in 2008 and 2011, respectively. He has been working towards the Ph.D. degree at PUT since 2008.

He is currently with the Politehnica University of Timisoara, Timisoara, Romania, where he became an Assistant Lecturer with the Department of Automation and Applied Informatics in 2012. He is the co-author of more than 30 papers published in scientific journals, refereed conference proceedings, and contributions to books. His current research interests include control structures and algorithms with focus on iterative methods in control design and optimization.

Dr. Rădac is a Member of the Romanian Society of Control Engineering and Technical Informatics.

Radu-Emil Precup (M'03–SM'07) received the Dipl.Ing. (with honors) degree in automation and computers from the "Traian Vuia" Polytechnic Institute of Timisoara, Timisoara, Romania, the Dipl. degree in mathematics from the West University of Timisoara, Timisoara, and the Ph.D. degree in automatic systems from the "Politehnica" University of Timisoara, Timisoara, Romania, in 1987, 1993, and 1996, respectively.

He is currently with the Politehnica University of Timisoara, Timisoara, Romania, where he became a Professor with the Department of Automation and Applied Informatics in 2000. He is an Honorary Professor with the Óbuda University, Budapest, Hungary. He is the author or coauthor of more than 200 papers published in scientific journals, refereed conference proceedings, and contributions to books. His current research interests include intelligent control systems, data-based control, and nature-inspired algorithms for optimization.

Prof. Precup is a Member of the Subcommittee on Computational Intelligence as part of the IEEE Industrial Electronics Society.

Emil M. Petriu (M'86–SM'88–F'01) received Dipl.Eng. and Dr.Eng. degrees in eElectrical engineering from the Polytechnic Institute of Timisoara, Timisoara, Romania.

He is a is a University Research Chair Professor with the School of Electrical Engineering and Computer Science, at the University of Ottawa, Ottawa, ON, Canada. His research interests include multisensor systems, soft computing, biology-inspired robot sensing, and human-computer symbiosis.

Prof. Petriu is a Fellow of the Canadian Academy of Engineering and the Engineering Institute of Canada. He was a co-recipient of the 2003 IEEE Donald G. Fink Prize Paper Award.

Stefan Preitl (M'03–SM'07) received the Dipl.Ing. (with honors) degree in electrical engineering and the Ph.D. degree in measurement techniques from the "Traian Vuia" Polytechnic Institute of Timisoara, Timisoara, Romania, in 1966 and 1983, respectively.

He is currently with the Politehnica University of Timisoara, Timisoara, where he became a Professor in the Department of Automation and Applied Informatics in 1992. He is an Honorary Professor with the Óbuda University, Budapest, Hungary. He is the author or co-author of more than 200 papers published in various scientific journals, refereed conference proceedings, and books in the field of automatic control. His current research interests include conventional and advanced structures and algorithms for automatic control applied to power or servo systems, control systems of electrical drives, methodical aspects of teaching, and development of computer-assisted education.

Prof. Preitl is a Member of the International Federation of Automatic Control Technical Committee on Control Design.