Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Model based control of a water tank system ⋆

Juri Belikov * **Eduard Petlenkov** **

* *Institute of Cybernetics, Tallinn University of Technology, Akadeemia tee 21, 12618, Tallinn, Estonia, (e-mail: jbelikov@cc.ioc.ee)*
** *Department of Computer Control, Tallinn University of Technology, Ehitajate tee 5, 19086, Tallinn, Estonia, (e-mail: petlenkov@dcc.ttu.ee)*

Abstract: Neural network with a specific restricted connectivity structure is used to identify a model of a real-life process. Parameters of the identified model are used to design a controller based on dynamic feedback linearization. The designed neural network based controller is verified on mathematical model within MATLAB/Simulink environment and applied to the real-time control of a plant. The static error is eliminated retuning input signal in the steady-state mode. Liquid level tank system was chosen as a case study to illustrate the applicability of the proposed approach. Experimental results have shown a good performance of the proposed technique. The designed controller is capable of tracking the desired water level for all set points with high degree of accuracy and without significant over/undershoot.

Keywords: nonlinear control systems, input-output linearization, water tank system.

1. INTRODUCTION

Present paper explores the abilities of the so-called Neural Networks based Simplified Additive Auto Regressive eXogenous models to be used to control of water tank type systems. NN-based SANARX is a subclass of a more general ANARX model class Kotta et al. [2006] and inherits all the advantages of its parent class. To be more specific, models of this type are always linearizable by dynamic output feedback. In other words, for a model employing SANARX structure one may always write down equations of the linearizing feedback. The latter means that once coefficients of the model are identified, one just needs to substitute their values into equations describing controller. The main idea of feedback linearization technique consists in modifying the system structure by appropriate feedback, so that the input-output (i/o) equation of the closed-loop system becomes linear. After that it is possible to apply all the standard control methods for linear systems to meet the required goals.

While the problem of liquid level control in a tank is not new, it still has not lost its actuality. Level regulators are used in industry to maintain a constant fluid pressure, or a constant fluid supply to a process, or in waste storage Dunn [2005]. The common examples of possible industrial applications include chemical industry and food processing Kern and Manness [1997] as well as different irrigation systems like dams, etc. Through the years various techniques have been used to tackle the problem. Recent analytic solutions employ tensor product based methods Precup et al. [2010] and decoupling control Wang et al. [2009]. In many cases the problem is approached by means of PI Sundaravadivu et al. [2011], PID Kern and Manness [1997], and fractional-order PID Tepljakov et al. [2013] controllers. Recently, methods based on computational intelligence have started to gain popularity and are applied either solely or in combination with some classical techniques Liang [2008]. While PID-type controllers are still popular choice in many industrial applications, they cannot guarantee that system will work with the same level of accuracy in the entire operating range. Furthermore, over/underregulation as well as changes in the environment are the common problems which one may spot in many other applications, and have to be taken into account during the design stage. Though pure analytic or numeric techniques have known limitations, their combinations with advanced methods of computational intelligence may lead to generic solutions with a broader application range.

The present contribution may be seen in application of the classical control technique (linearization via dynamic output feedback) and neural networks based modeling to control a water level in a tank system. In addition, a method for compensating steady-state error is proposed. In the paper we describe all design steps: starting from collecting the i/o data of a process and finishing with implementation and test of the synthesized controller on a real prototype. One of the most complicated parts of the overall design procedure was solved using feedback linearization technique. Like any other analytic method the linearization by dynamic output feedback provides one with equations of a controller, whose coefficients are taken directly from the identified model. In other words, selecting NN-SANARX model, one merges together numeric parts of the modeling and control synthesis. Therefore, the research, presented in the paper, can be seen as a preliminary step towards real industrial application.

⋆ This work was partially supported by the Estonian Science Foundation Grant no. 8738.

2. PRELIMINARIES

Hereinafter, if $\xi : \mathbb{Z} \to \mathbb{R}$ and $k \in \mathbb{N}$, then $\xi^{[k]}$ stands for kth-step forward time shift of ξ and is defined by $\xi^{[k]} := \xi(t+k)$. Similarly for backward shift. Note that $\xi^{[0]} := \xi(t)$. Moreover, to simplify exposition of the material, in this paper we restrict our attention to the case of single-input single-output (SISO) systems. The nonlinear control systems are typically represented either by the higher order i/o difference equation

$$y^{[n]} = \varphi\left(y, y^{[1]}, \ldots, y^{[n-1]}, u, u^{[1]}, \ldots, u^{[n-1]}\right), \qquad (1)$$

or by the state equations

$$\begin{aligned} x^{[1]} &= f(x,u) \\ y &= h(x), \end{aligned} \qquad (2)$$

where $x : \mathbb{Z} \to \mathcal{X} \subset \mathbb{R}^n$ is the state vector, $u : \mathbb{Z} \to \mathcal{U} \subset \mathbb{R}$ is the input signal, $y : \mathbb{Z} \to \mathcal{Y} \subset \mathbb{R}$ is the output signal; $\varphi : \mathcal{Y}^n \times \mathcal{U}^n \to \mathbb{R}$, $f : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ and $h : \mathcal{X} \to \mathcal{Y}$ are real analytic functions.

The system, represented by (1), is known in the literature as a discrete-time Nonlinear AutoRegressive eXogenous (NARX) model. This model can be used to identify a wide class of complex processes with a high degree of accuracy, see Leontaritis and Billings [1985]. However, from the control point of view, models of the form (1) have several drawbacks. The most important for the practise is linearizability by dynamic output feedback. In general, this property does not always hold for models with NARX structure, see Pothin et al. [2000] for details. To overcome this problem, so-called Additive NARX (ANARX) structure was proposed. It is a subclass of the NARX model having separated time instances Kotta et al. [2006]

$$y^{[n]} = f_1\left(y^{[n-1]}, u^{[n-1]}\right) + \cdots + f_n(y,u). \qquad (3)$$

Model (3) can always be linearized using the following dynamic output feedback

$$\begin{aligned} \eta_1^{[1]} &= \eta_2 - f_1(y,u) \\ &\vdots \\ \eta_{n-2}^{[1]} &= \eta_{n-1} + f_{n-1}(y,u) \\ \eta_{n-1}^{[1]} &= v - f_n(y,u) \\ y &= x_1, \end{aligned}$$

where $v : \mathbb{Z} \to \mathcal{V} \subset \mathbb{R}$ is a reference signal (desired output). In addition, ANARX model (3) can be directly represented via state equation. The latter can be used, for example, to design state controller Vassiljeva et al. [2010].

3. NEURAL NETWORKS BASED SIMPLIFIED ANARX MODEL

In order to perform analysis and design of the appropriate controller for the process, one is usually interested in mathematical equations rather than in the black-box description. In fact, one can derive the model from the first principles, relying on the Newton equations. However, most likely in many cases such an approach will result in a quite complex model. Thus, one may start from the measured data and identify relations between variables. One of the most popular approaches consists in employing Neural Networks (NN) formalism. Thus, the theory for ANARX models was adopted to the case of neural networks in Kotta et al. [2006], Petlenkov et al. [2006]. To be more specific, NN-ANARX model can be represented as

$$y = \sum_{i=1}^{n} C_i \phi_i \left(W_i \left[y^{[-i]} \;\; u^{[-i]}\right]^{\mathrm{T}}\right), \qquad (4)$$

where i stands to the sublayer, ϕ_i is an activation function, C_i and W_i are $1 \times l_i$ and $l_i \times 2$ dimensional matrices of the output and input synaptic weights, respectively. In addition, l_i is the number of hidden neurons. A schematic diagram of the neural network, representing ANARX structure, is depicted in Fig. 1.

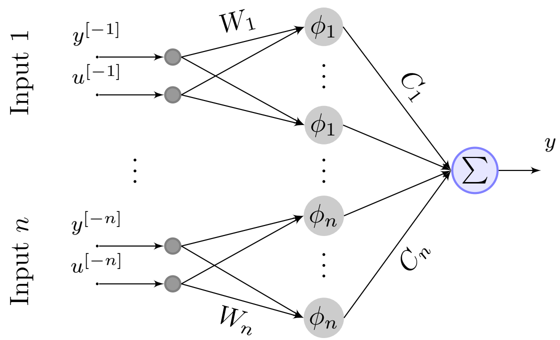

Fig. 1. Neural network representing ANARX structure

Note that the application of the feedback linearization algorithm to ANARX model leads to the finite discrete-time linear closed-loop system described either by equation $y^{[n]} = v$ or transfer function $Y(z)/V(z) = 1/z^n$. However, in real-life poles of the transfer function corresponding to such a model can be too *fast* resulting in an undesirable behavior of the control system (for example, significant overshootings, high control signals, etc.). Therefore, an algorithm based on a more general reference model can be used instead. In this case the closed-loop system can be described as

$$y + a_1 y^{[-1]} + \cdots + a_n y^{[-n]} = b_1 v^{[-1]} + \cdots + b_n v^{[-n]}, \qquad (5)$$

where $a_1, \ldots, a_n \in \mathbb{R}$ and $b_1, \ldots, b_n \in \mathbb{R}$ are parameters of the reference model. Observe that (5) is a linear discrete-time reference model that can be used to predefine dynamics of the closed-loop system. Thus, the issue with undesirable behavior can be solved selecting appropriate zeros and poles. As a result, the dynamic output feedback can be written using parameters of the neural network as

$$\eta_1 = a_1 y - b_1 v + C_1 \phi_1 \left(W_1 \left[y \;\; u\right]^{\mathrm{T}}\right) \qquad (6)$$

and

$$\begin{aligned} \eta_1^{[1]} &= \eta_2 + b_2 v - a_2 y - C_2 \phi_2 \left(W_2 \left[y \;\; u\right]^{\mathrm{T}}\right) \\ &\vdots \\ \eta_{n-2}^{[1]} &= \eta_{n-1} + b_{n-1} v - a_{n-1} y \\ &\quad - C_{n-1} \phi_{n-1} \left(W_{n-1} \left[y \;\; u\right]^{\mathrm{T}}\right) \\ \eta_{n-1}^{[1]} &= b_n v - a_n y - C_n \phi_n \left(W_n \left[y \;\; u\right]^{\mathrm{T}}\right). \end{aligned} \qquad (7)$$

In the following proposition we will formulate the condition of equivalence between the closed-loop system and the reference model.

Proposition 1. Application of (6) and (7) to (4) yields (5).

Proof. Apply the dynamic output feedback (6) and (7) to NN-ANARX model (4)

$$C_1\phi_1\left(W_1\left[y^{[-1]}\ u^{[-1]}\right]^{\mathrm{T}}\right)+a_1y^{[-1]}-b_1v^{[-1]}=$$
$$=b_nv^{[-n]}-a_ny^{[-n]}-C_n\phi_n\left(W_n\left[y^{[-n]}\ u^{[-n]}\right]^{\mathrm{T}}\right)+\cdots+$$
$$b_2v^{[-2]}-a_2y^{[-2]}-C_2\phi_2\left(W_2\left[y^{[-2]}\ u^{[-2]}\right]^{\mathrm{T}}\right) \quad (8)$$

and regroup terms as

$$\sum_{i=1}^{n}C_i\phi_i\left(W_i\left[y^{[-i]}\ u^{[-i]}\right]^{\mathrm{T}}\right)+a_1y^{[-1]}+\cdots$$
$$\cdots+a_ny^{[-n]}=b_1v^{[-1]}+b_2v^{[-2]}+\cdots+b_nv^{[-n]}. \quad (9)$$

Now, one can easily see that the sum in the left-hand side of (9) is exactly the right-hand side of (4). Therefore, (9) can be rewritten as

$$y+a_1y^{[-1]}+\cdots+a_ny^{[-n]}=b_1v^{[-1]}+\cdots+b_nv^{[-n]}. \quad (10)$$

It remains to stress that (10) describes both closed-loop and reference model. ■

Remark 1. The control technique presented in Petlenkov [2007] is a special case of (6)-(7) for $b_1=\ldots=b_{n-1}=a_1=\ldots=a_n=0$ and $b_n=1$.

In order to simplify the calculation of the control signal in (6), we assume like in Petlenkov [2007] that ϕ_1 is a linear function, resulting in a simplified structure of the neural network known as an NN-based Simplified ANARX (NN-SANARX) model. The latter yields

$$u=T_2^{-1}(\eta_1-(T_1+a_1)y+b_1v), \quad (11)$$

where T_1 and T_2 are the first and second elements of the vector C_1W_1, respectively. Note that T_2 cannot be equal to zero. The overall structure of the corresponding control system is represented in Fig. 2.

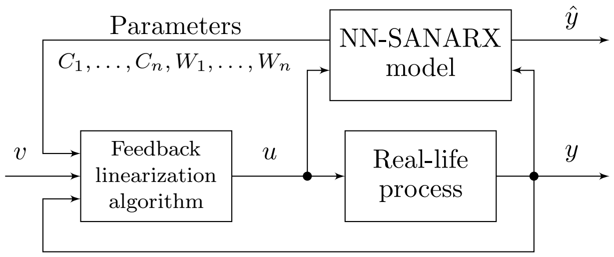

Fig. 2. Control system

From Fig. 2, one can see that the application of the feedback linearization algorithm (6)-(7) to NN-SANARX model and process yields two different equations

$$\hat{y}+a_1y^{[-1]}+\cdots+a_ny^{[-n]}=b_1v^{[-1]}+\cdots+b_nv^{[-n]}, \quad (12)$$

and

$$y+a_1y^{[-1]}+\cdots+a_ny^{[-n]}=$$
$$b_1v^{[-1]}+\cdots+b_nv^{[-n]}+\varepsilon, \quad (13)$$

respectively. Subtracting (12) from (13), we get $y=\hat{y}+\varepsilon$, where ε is an error caused by imperfectness of an NN-based model describing the process. Since we are interested in analysis of a steady-state error e_{ss} after the transient process is complete, equation $y=\hat{y}+e_{ss}$ is analyzed instead. Obviously, the value of the error will depend not only on the identified model, but on the accuracy of calculations as well. In real-life applications one cannot expect absolute accuracy due to various problems like rounding, limited memory size, etc. In any case one is interested in making error as small as possible in the limit as time goes to infinity, i.e. $e_{ss}\to\min$ as $t\to\infty$. Now, we need to understand how to detect in real-time that the output of the process has reached steady-state. In case of linear systems the answer can be found very easily from the limit theorems. However, for nonlinear systems this approach is not applicable. This is the reason why we decided to use an alternative approach schematically depicted in Fig. 3.

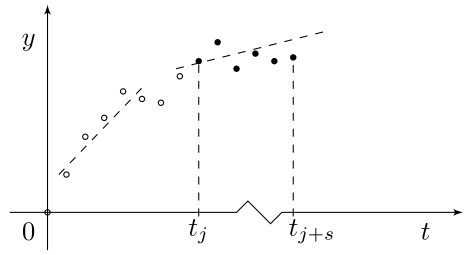

Fig. 3. Steady-state detection: slope based approach

The main idea of this approach consists in the analysis of a slope of a line fitting a certain amount of data. To be more precise, we are given with a set of data samples $\{y_j,\ldots,y_{j+s}\}$ obtained from measurements. First, we need to derive equation of the best-fitting line $p(t)=\alpha t+\beta$ to the given set of points. Coefficients α and β can be found using, for instance, least squares fitting procedure. We are interested only in the coefficient α that defines the slope (between the line $p(t)$ and time axis O_t) of the line that can be calculated as $\gamma=\arctan(\alpha)\cdot 180/\pi$. We analyze only the last s data samples of incoming measurements. Thus, index j is not fixed and increases with time. This yields the sequence of angles as $\Gamma=\{\gamma_j\}_{j=1}^{\infty}$. In fact, this sequence is bounded either by the simulation or by the working time; however *a priori* the time is unknown. Note that Γ has to converge to zero as the transient process is complete and the steady-state is reached.

Remark 2. If the computational capabilities are the case in the application, then one may use a *simplified* version of the approach proposed above. Namely, instead of finding equation for the line $p(t)$, it is possible to use numerical differentiation to calculate the slope as $\alpha=(t_{i+s}-t_i)/h$ for $h\ll 1$. However, this approach is less robust to rounding errors that may appear in the output signal.

Once the steady-state is detected, we can set $e_{ss}:=v-y$ and use this value in our algorithm to *calibrate* the input signal by adding e_{ss} to the last equation in (7). This is possible due to the specific structure of the controller. One may easily check that application of (6) and (7) to the process yields (13) with $|\varepsilon-e_{ss}|\to 0$. It is important to mention that if the improved version of the algorithm will not be able to detect the stead-state, then the algorithm automatically reduces to the classical version, i.e. to the version without calibration of the input signal with e_{ss} equals to zero.

4. MULTI-TANK SYSTEM: MATHEMATICAL MODEL

The model of a Multi-Tank system is borrowed from the manual, provided by INTECO Sp. z o. o [2013], and depicted in Fig. 4.

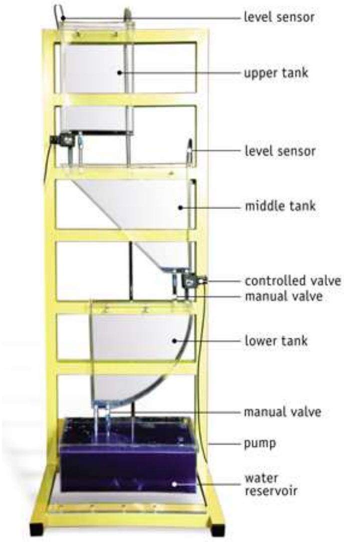

Fig. 4. Model of the Multi-Tank system

Since we are interested in control of the water level in the first tank, the differential equations, describing dynamics of the system, can be derived, assuming the laminar outflow rate of an *ideal fluid* from a tank, by means of mass balance as

$$\begin{aligned} \dot{x}_1 &= \frac{1}{aw}(u - c_1 x_1^{\alpha_1}) \\ y &= x_1. \end{aligned} \tag{14}$$

In (14) u is the inflow to the upper tank, x_1 is the fluid level in the tank, w is the width, a is the length, c_1 is the resistance of the output orifice, and α_1 is the flow coefficient. The numerical values for the parameters will be provided further in Section 5. The plant is designed to operate with an external PC-based digital controller. The computer communicates with the level sensor, valve and pump by a dedicated i/o board and the power interface. The i/o board is controlled by the real-time software which operates in Simulink using MATLAB Real-Time Windows Target environment.

5. LIQUID LEVEL CONTROL OF WATER TANK SYSTEM: PRACTICAL RESULTS

All the experiments, described in this section, were performed on the equipment available in the laboratory at the Department of Computer Control, Tallinn University of Technology, see ALab [2014] for more details. The physical parameters of the plant have the following numerical values $w = 0.035$m, $a = 0.25$m, $\alpha_1 = 0.4628$, and the maximal inflow provided by the pump is $1.0284 \cdot 10^{-4}\mathrm{m}^3/\mathrm{s}$. In addition, the resistance of the output orifice of the first tank was determined experimentally $c_1 = 2.0687 \cdot 10^{-4}\mathrm{m}^2/\mathrm{s}$, using MATLAB routine provided with the installation package. Next, we describe the identification procedure based on the neural networks approach.

The identification data was collected from the real plant with sampling time 0.5s. The input signal was normalized into the unit interval $[0, 1]$ to simplify the training procedure of a neural network. The collected i/o data was used to train NN-SANARX structure by means of gradient descent with adaptive learning rate backpropagation algorithm. The network shown in Fig. 1 was trained with two sublayers, corresponding to the second order ($n = 2$) of the model, and 3 neurons on each sublayer, i.e. $l_1 = l_2 = 3$. The pure linear activation function was chosen on the first and output sublayers as well as hyperbolic tangent sigmoid activation function (tansig) on the second sublayer, reflecting nonlinearities of the process. The identified model has the following structure

$$\begin{aligned} \hat{y} = T_1 y^{[-1]} + T_2 u^{[-1]} + \\ + C_2 \mathrm{tansig}\left(W_2 \left[y^{[-2]} \;\; u^{[-2]}\right]^{\mathrm{T}}\right). \end{aligned} \tag{15}$$

Next, using (7), (11) and parameters of the identified model (15), we get dynamics of the controller as follows

$$\begin{aligned} u &= T_2^{-1}(\eta_1 - T_1 y) \\ \eta_1^{[1]} &= v - C_2 \mathrm{tansig}\left(W_2 \left[y \;\; u\right]^{\mathrm{T}}\right). \end{aligned} \tag{16}$$

Note that we have intentionally chosen the parameters of the reference model to be $b_1 = a_1 = a_2 = 0$ and $b_n = 1$, since the transient process has relatively slow nature. The reference signal v was chosen as a piecewise constant function described in Table 1.

Table 1. Set points

Value [m]	Time interval [s]
0.20	$0 \leq t < 180$
0.05	$180 \leq t < 270$
0.10	$270 \leq t < 360$
0.15	$360 \leq t < 450$

The quality of the control algorithm is depicted in Fig. 5.

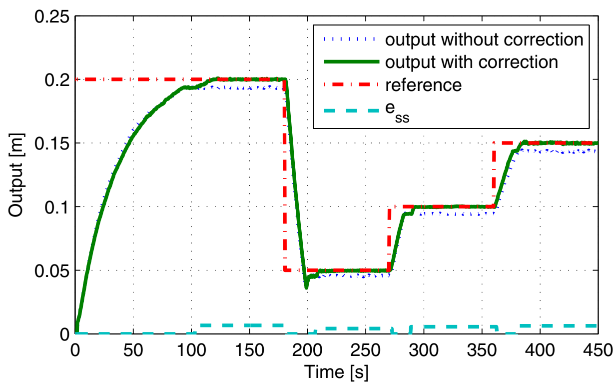

Fig. 5. Comparison of control systems with and without input signal correction

It can be seen from Fig. 5 that the control system is capable of tracking the reference signal v and react correctly to the changes in a set point. In addition, the algorithm based on the analysis of the slope of the fitting line was able to recognize that the steady-state is reached by the system as shown in Fig. 6.

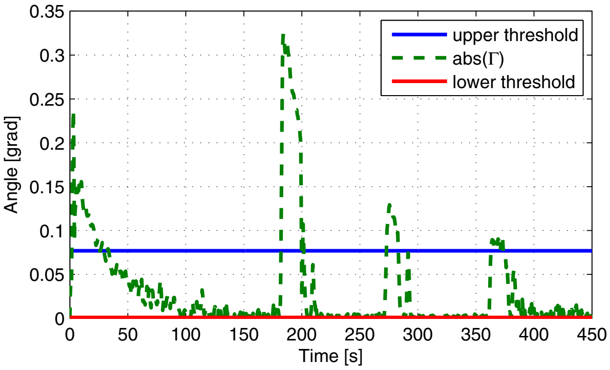

Fig. 6. Detection of a steady-state via slope analysis

The lower and upper thresholds were found to be 0.01 and 0.077 grad, respectively. Small values of the angles are due to the unequal scaling of the time and angle axes. In fact, one may not care about it, since this does not affect the algorithm. Really, bringing axes to a common denominator is the same as multiplication of one of the axes by some positive constant. The latter that does not change the steady-state detection time. The timely detection allowed to eliminate the static error from the loop calibrating input signal. Finally, it is important to stress that the control system works with the same accuracy on the whole region of set points.

5.1 *Comparison with existing approaches*

Next, we provide a brief comparison of the proposed above technique with widely used approaches such as:

- **Anti-Windup PID** control.
- **Relay**. This is a usual ON-OFF regulator.
- **Analytic**. This approach is based on the so-called exact state feedback linearization technique.

In addition, we discuss the strong and weak points of each technique. The simulation results are depicted in Fig. 7.

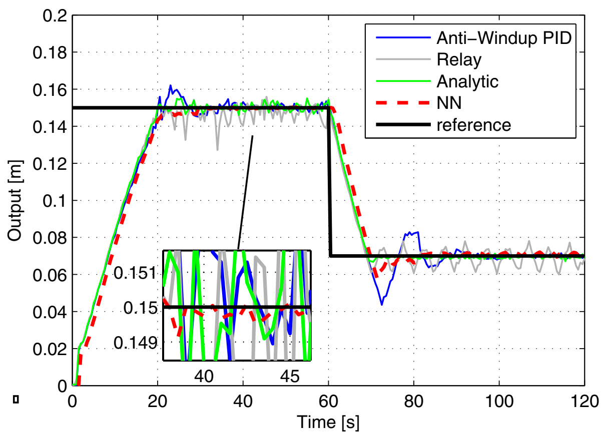

Fig. 7. Comparison results: outputs

It can be seen that all control methods are capable of tracking the reference signal. In addition, we present in Fig. 8 the performance of the control signal for each particular approach.

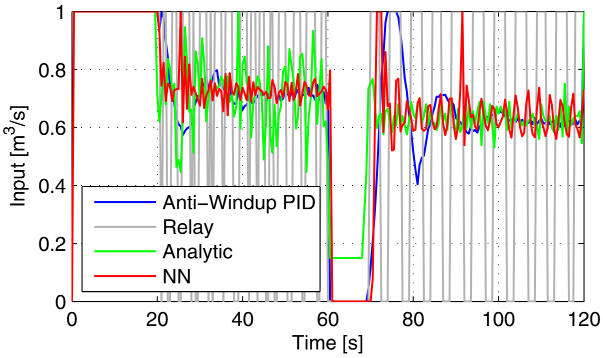

Fig. 8. Comparison results: control signals

To evaluate the quality of each control algorithm we rely on several statistical tools. The results are presented in Table 2. One may see that NN provides satisfactory results due to the fact that the output of the system has quite small oscillations compared to other techniques.

Table 2. Statistical measure of performance in steady-state

Method	AW PID	Relay	Analytic	NN
MSE	$3.15 \cdot 10^{-5}$	$2.47 \cdot 10^{-5}$	$2.6 \cdot 10^{-6}$	$4.73 \cdot 10^{-6}$
SSE	0.2681	0.2103	0.0221	0.0402
$\sum \lvert v - y \rvert$	26.5788	34.2955	10.3010	11.0672

The evaluation of each method is summarized in Table 3.

Table 3. Overview of the control methods

Criteria	AW PID	Relay	Analytic	NN
complexity	medium	low	high	medium
versatility	medium	high	low	high
robustness	high	medium	medium	high
model	–	–	required	–
quality of u	medium	low	medium	medium

Hence, it follows that:

- *Anti-Windup PID*: (−) strongly depends on the working point and the quality of the corresponding linear model; (+) naturally eliminates steady-state error.
- *Relay*: (−) poor and not effective performance of the control signal (heavily exploits an actuator); (+) the simplest among other considered techniques.
- *Analytic*: (−) requires mathematical model of the process, yielding dependence on the quality of the identified parameters like α_1 and c_1; (+) in the presence of small measurement noise provides the most efficient control performance.
- *NN*: (−) relies on the heuristic identification techniques; (+) works in the whole range of set points with the same accuracy.

6. DISCUSSION AND FUTURE WORK

It is well known that the majority of model based control techniques suffer from a common drawback. Namely, the quality of the overall control system significantly depends on the accuracy of the derived model. In this paper, classical dynamic output feedback linearization algorithm with

NN-based SANARX model and static error compensation technique was applied to control the liquid level in the real tank system. The controller is designed incorporating a reference model, which makes possible to predefine the desired dynamics of the designed control system. The choice of NN-SANARX model class has allowed to merge numeric parts of system modeling and control synthesis that reduced computational complexity and in turn simplified implementation. Real-time application was performed in MATLAB/Simulink without using any specific toolbox functions, indicating that the proposed algorithm is implementation friendly. The proposed static error compensation technique allows to automatically tune the controller in the whole range of set points.

The paper presents all steps required to design the controller starting from collecting the i/o data of a process and finishing with implementation and test of the synthesized controller on the prototype of the real plant. It can be seen that the approach guarantees accurate tracking with small over/underregulation and stable performance for the entire range of the set points. We want to emphasize that the proposed input signal calibration technique is not sensitive to short-time disturbances. Finally, discrete-time integrator can not be effectively used to compensate the static error because of fluctuating liquid level in real tank system. The proposed technique provides more effective solution for tuning the system in a steady-state.

Next, we discuss several limitations appearing within the proposed approach. First, it is always necessary to assume at least the second order of the identified model due to the controller structure (6)-(7). In fact, this is a natural assumption that has to be made in majority of the applications. Second, from (11) it follows that one has to take care of $T_2 \neq 0$. Finally, it is necessary to define 2 thresholds to tune the steady-state detection algorithm for elimination of a static error. Our experiments have clearly shown that it can be done for each particular system after reaching the first steady-state and switching to the next set point. The thresholds can be detected using the slope based approach proposed in the paper. The current version of the algorithm is insensitive to small deviations in the set point or disturbances that do not exceed the threshold. This is done to make it more robust. However, it can be improved, for example, by decreasing the level of noise in measurements. To conclude, all experiments discussed in the paper were performed on the real plant with fluctuating water level.

The idea from Petlenkov [2007] can be used to simultaneous control of liquid level in several interconnected tanks. Moreover, the results of Belikov et al. [2013] allows us to rely on statistical methods in searching for a good NN model.

REFERENCES

ALab. Alpha Control Laboratory, Tallinn University of Technology, 2014. URL `http://a-lab.ee/`.

J. Belikov, E. Petlenkov, K. Vassiljeva, and S. Nõmm. Computational intelligence methods based design of closed-loop system. In *Neural Information Processing*, Lecture Notes in Computer Science, pages 215–224. Springer-Verlag Berlin Heidelberg, 2013.

W. C. Dunn. *Fundamentals of Industrial Instrumentation and Process Control*. McGraw-Hill Professional, 2005.

INTECO Sp. z o. o. INTECO, 2013. URL `http://www.inteco.com.pl`.

C. Kern and M. Manness. PID controller tuning for mixed continuous/discrete event processes using dynamic simulation. In *Dynamic Modeling Control Applications for Industry Workshop*, pages 37–43, Vancouver, BC, Canada, May 1997.

Ü. Kotta, F. N. Chowdhury, and S. Nõmm. On realizability of neural networks-based input-output models in the classical state-space form. *Automatica*, 42(7):1211–1216, 2006.

I. J. Leontaritis and S. A. Billings. Input-output parametric models for non-linear systems Part I: deterministic non-linear systems. *International Journal of Control*, 41 (2):303–328, 1985.

L. Liang. The application of fuzzy PID controller in coupled-tank liquid-level control system. In *International Conference on Electronics, Communications and Control*, pages 2894–2897, Ningbo, China, September 2008.

E. Petlenkov. NN-ANARX structure based dynamic output feedback linearization for control of nonlinear MIMO systems. In *The 15th Mediterranean Conference on Control and Automation*, pages 1–6, Athena, Greece, June 2007.

E. Petlenkov, S. Nõmm, and Ü. Kotta. Neural networks based ANARX structure for identification and model based control. In *The 9th International Conference on Control Automation Robotics and Vision*, pages 2284–2288, Singapore, December 2006.

R. Pothin, Ü. Kotta, and C. H. Moog. Output feedback linearization of nonlinear discrete time systems. In *The IFAC Conference on Control Systems Design*, pages 181–186, Bratislava, Slovak Republic, 2000.

R. Precup, L.-T. Dioanca, E. M. Petriu, M. Radac, S. Preitl, and C. Dragos. Tensor product-based real-time control of the liquid levels in a three tank system. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 768–773, Montreal, ON, Canada, October 2010.

K. Sundaravadivu, V. Jeyakumar, and K. Saravanan. Design of fractional order PI controller for liquid level control of spherical tank modeled as fractional order system. In *IEEE International Conference on Control System, Computing and Engineering*, pages 522–525, Penang, Malaysia, November 2011.

A. Tepljakov, E. Petlenkov, J. Belikov, and M. Halás. Design and implementation of fractional-order PID controllers for a fluid tank system. In *The 32nd American Control Conference*, pages 1780–1785, Washington, USA, June 2013.

K. Vassiljeva, E. Petlenkov, and J. Belikov. State-space control of nonlinear systems identified by ANARX and neural network based SANARX models. In *The World Congress on Computational Intelligence*, pages 3816–3823, Barcelona, Spain, July 2010.

Q. Wang, L. Xing, and X. Shi. Decoupling control of three-tank liquid level systems based on feedforward compensation. In *Chinese Control and Decision Conference*, pages 5863–5866, Guilin, China, June 2009.