# Analysis of Adaptation Law of the Robust Evolving Cloud-based Controller

Goran Andonovski, Sašo Blažič
Faculty of Electrical Engineering
University of Ljubljana, Slovenia
goran.andonovski@fe.uni-lj.si,
saso.blazic@fe.uni-lj.si

Plamen Angelov
School of Coputing and Communications
Lancaster University, United Kingdom
p.angelov@lancaster.ac.uk

Igor Škrjanc
Faculty of Electrical Engineering
University of Ljubljana, Slovenia
igor.skrjanc@fe.uni-lj.si

*Abstract*—**In this paper we propose a performance analysis of the robust evolving cloud-based controller (RECCo) according to the different initial scenarios. RECCo is fuzzy rule-based (FRB) controller with non-parametric antecedent part and PID-type consequent part. Moreover, the controller structure (the fuzzy rules and the membership function) are created based on the on-line streaming data. The advantage of the RECCo controller is that do not require any a priory knowledge of the controlled system. The algorithm starts with zero fuzzy rules (zero data clouds) and evolving during the process control. The PID parameters of the controller are initialed with zeros and are adapted in on-line manner. According to the zero initialization of the parameters the new adaptation law is proposed in this article to solve the problems in the starting phase of the process control. Several initial scenarios were theoretically propagated and experimentally tested on model of heat-exchanger plant.**

## I. INTRODUCTION

Fuzzy system theory and fuzzy sets were firstly introduced by Prof. Lotfi. A Zadeh in [1]. Since then an impressive growth of fuzzy systems from both the theoretical and applied points of view is evident. The primary expectations were that the fuzzy sets would find the applications in non-technical systems (linguistic, economy, etc.), but the reality is that the fuzzy systems are widely used in control applications ( [2], [3], [4], [5], [6]).

Controlling the complex and nonlinear processes is represent by dividing the problem into several locally linear problems/solutions. Local linearizion of the system provides satisfactory results but only for a certain range of operating conditions. In therms of fuzzy theory, the nonlinear system is represent with more local problems modeled with individual rules (each rule represent a local model) and furthermore, they are combined (according to the fuzzy membership functions) to describe the global behavior of the nonlinear system. Fuzzy controllers represent an approach for solving the nonlinearity process control problem.

The classical nonlinear control systems based on the Takagi-Sugeno (TS) fuzzy model [7] is one of the most popular and widely used for different applications ( [8], [9], [10], [11]). It is made up of a IF-THEN fuzzy rules representing local linear input/output relations of the nonlinear system. The first part (IF) of the conditional is termed the antecedent, and the second part (THEN) is the consequent. Another fuzzy rule based (FRB) system was introduced by E. Mambdani in [12], [13]. The main difference between the TS FRB systems and Mamdani, is the way of how the consequent part is defined. In TS fuzzy systems the antecedent part is fuzzy and the consequent part is functional while in the Mamdani FRB systems both parts are fuzzy. The challenge in this type of FRB systems is setting the parameters of the fuzzy set (at least two parameters per fuzzy set). In [14] the authors have analyzed that the fuzzy rules and membership functions should be predefined to map numerical data into linguistic terms and to make fuzzy reasoning work. But there is no guarantee that the predefined knowledge will produce permanent solutions in case of changing environment and changing condition. Due to this problem there were developed several approaches for on-line adaptation of the fuzzy parameters ( [15], [16]).

New FRB system ANYA was introduced in 2001 by Angelov [17]. This approach does not require explicit predefinition of the antecedent part and allows the controller structure (the membership function, fuzzy rules, fuzzy set, etc.) to be created based on on-line streaming data. Simplified antecedent part in this case is non-parametric and it is formed by grouping the data samples with similar properties into data clouds. The clouds have no specific shape, parameters or boundaries. ANYA uses the relative density measure of the current data sample according to the existing clouds to determine the membership to the particular cloud. The density measures takes into account the distances to all previous data samples and can be calculated recursively. The consequent part of the ANYA type FRB system can still be same as in TS or Mamdani FRB system.

In [18], [19], [20] a new control algorithm RECCo was introduced, which is based on ANYA FRB system. RECCo (Robust Evolving Cloud-based Controller) starts with zero fuzzy rules (zero data clouds) and with empty controller's parameters. During the process control the structure of RECCo evolves and the parameters of the controller are adapted. The control algorithm do not requires any a priory knowledge of the controlled system.

In this paper we present a performance analysis of the RECCo algorithm according to the different initial situations. The idea and the goal is to test the performance with both theoretical and practical implementation.

The rest of this article is organized as follows. Section II describes the robust evolving cloud-based controller and his structure. The adaptation law in section III is presented. Finally in section IV different simulation scenarios are analyzed and presented as a proof of concept for the proposed assumptions. The main conclusions are summarized in section V.

## II. THE STRUCTURE OF THE RECCO CONTROLLER

A special feature of the RECCo is that no a priori information about the controlled process is needed. Theoretically, the controller could be initialized from the first data sample received. But of course, any existing information can be used to suitably initialize the controller parameters. After this, for every incoming sample the controller gains are adapted and if the certain conditions are satisfied, a new cloud is created.

In this section we present the robust cloud-based controller structure with non-parametric antecedents. As we already mentioned, this method applies the concept of fuzzy data clouds and relative data density to define antecedents. Each data element is associated on-line to one of the existing clouds (if current data is close enough according to a chosen similarity measure) or a new fuzzy rule (cloud) is created. The concept does not employ membership functions in the classical sense. Degree of fulfillment is based on the distances between samples and the corresponding cloud relative density. At this point we have to mention that already two different similarity measures were used: Euclidean [21] [19] and Mahalanobis [20] distances. In this paper a simpler Euclidean distance is used according to the fact that both methods produced satisfying results.

The structure of ANYA was initially introduced in [21]. The authors proposed simplified FRB system of the following form:

$$\mathcal{R}^i : \text{IF} \quad (\mathbf{x} \sim X^i) \quad \text{THEN} \quad (u^i) \tag{1}$$

where the operator $\sim$ denotes the fuzzy membership expressed linguistically as '*is associated with*'. The number of the rules is defined by the number of the data clouds $c$ ($i = 1, 2, \ldots, c$), and usually changes with time. $X^i$ denotes the $i$-th cloud in the data space where $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ is the controller's input and $u^i$ denotes its output in the $i$-th fuzzy domain. The contribution of a particular controller output in a certain fuzzy domain to the actual control output is given by the normalized relative density:

$$\lambda_k^i = \frac{\gamma_k^i}{\sum\limits_{j=1}^{c} \gamma_k^j} \quad i = 1, \ldots, c \tag{2}$$

where $\gamma_k^i$ is the local density of the $i$-th cloud for the current data $\mathbf{x}_k$.

When a new data sample arrives, we compute $c$ separate densities $\gamma_k^j$ that define how "close" a new sample is relative to the existing clouds. According to the calculated densities, the current data sample is finally associated with the cloud with maximal density and all the parameters of this particular cloud are updated.

In [19] and [18] the local density of the $i$-th cloud is defined by Cauchy kernel as follows:

$$\gamma_k^i = \frac{1}{1 + \frac{\sum_{j=1}^{k-1}(d_{kj}^i)^2}{k-1}} \tag{3}$$

where $k$ is the current time instant and $\sum_{j=1}^{k-1}(d_{kj}^i)^2$ is the sum of the square of Euclidean distances ($d_{kj}^i = \|\mathbf{x}_k - \mathbf{x}_j\|^2$) between the new data $\mathbf{x}_k$ and all the data points of the $i$-th cloud. Furthermore, (3) can be recursively written as follows:

$$\gamma_k^i = \frac{1}{1 + \|\mathbf{x}_k - \mu_k^i\|^2 + \sigma_k^i - \|\mu_k^i\|^2} \tag{4}$$

where $\mu_k^i$ is the mean value of the cloud's data points and $\sigma_k^i$ is the variance. Both of them can be recursively calculated using following equations for mean value and variance, respectively:

$$\mu_k^i = \frac{k-1}{k}\mu_{k-1}^i + \frac{1}{k}\mathbf{x}_k \tag{5}$$

$$\sigma_k^i = \frac{k-1}{k}\sigma_{k-1}^i + \frac{1}{k}\|\mathbf{x}_k\|^2 \tag{6}$$

Initial condition for the mean value is $\mu_1^i = \mathbf{x}_1$ and for the variance is $\sigma_1^i = \|\mathbf{x}_1\|^2$.

Once we classified the current data sample to one of the clouds and updated its properties we can do the defuzzification of the FRB system. As we said above, the ANYA FRB system can work with both Mamdani and TS consequents. Therefore, if we consider the weighted average for the defuzzification, the output of the controller becomes:

$$u_k = \sum_{i=1}^{c} \lambda_k^i u^i = \frac{\sum\limits_{i=1}^{c} \gamma_k^i u^i}{\sum\limits_{i=1}^{c} \gamma_k^i} \tag{7}$$

where $u^i$ denotes the $i$-th rule consequent and normalized relative density (2) is used.

Depending on the system dynamics we defined first order linear reference-model with corresponding time constant $\tau$. We have to note here that larger time constant usually produce better results especially in terms of robustness at the cost of a lower speed. In our case our reference model is defined as:

$$y_{k+1}^r = a_r y_k^r + (1 - a_r)r_k \quad 0 < a_r < 1 \tag{8}$$

where $r_k$ is the reference signal set by the operator. Parameter $a_r$ is the pole of the first order discrete reference model and defines the transient dynamics. For systems with fast sampling, $a_r$ can be approximated by $(1 - \frac{T_s}{\tau})$ where $T_s$ defines the sampling period. In all our experiments we used $\tau = 40$ and $T_s = 2s$ which follows to $a_r = 0.95$. The values of the time constant and the sampling time are convenient for the dynamics of the studied heat-exchanger plant.

In this approach, the PID-based rule consequents of the following form are proposed:

$$u_k^i = P_k^i \varepsilon_k + I_k^i \Sigma_k^\varepsilon + D_k^i \Delta_k^\varepsilon + R_k^i \tag{9}$$

where $\varepsilon_k = y_k^r - y_{k-1}$ is the tracking error (the difference between the reference model and the process output). $P_k^i, I_k^i, D_k^i$ are controller gains and $R_k^i$ is compensation of the operating point for each cloud $i = 1, \dots, c$. Discrete-time integral ($\Sigma_k^\varepsilon$) and discrete derivative ($\Delta_k^\varepsilon$) of the tracking error $\varepsilon_k$ are recursively computed as:

$$\Sigma_k^\varepsilon = \sum_{\kappa=0}^{k-1} \varepsilon_\kappa = \Sigma_{k-1}^\varepsilon + \varepsilon_{k-1} \tag{10}$$

$$\Delta_k^\varepsilon = \varepsilon_k - \varepsilon_{k-1} \tag{11}$$

In this paper the input data is defined as follows:

$$\mathbf{x} = \left[ \begin{array}{cc} \frac{\varepsilon_k}{\Delta \varepsilon}, & \frac{y_k^r - r_{min}}{\Delta r} \end{array} \right]^T \tag{12}$$

where $\Delta r = r_{max} - r_{min}$ and $\Delta r$ depends on the operating system area and $\Delta \varepsilon = \frac{\Delta r}{2}$. In this case we are mostly interested in the region where we expect the majority of the data samples.

## III. ADAPTATION OF THE CONTROLLER

In this section we introduce the adaptation scheme of the RECCo controller's parameters and furthermore, we expose different initial scenarios and analyze the possible problematic situations. An improved adaptation of the RECCo controller is proposed, where the absolute value of error is used. New adaptation method proposed is used only in the starting phase of the RECCo evolving system and afterwards, the adaptation continues as originally proposed in [18] and [19].

In the starting phase the controller gains are adapted as follows:

$$\Delta P_k^i = \alpha_P \, G_{sign} \lambda_k^i \frac{|e_k \varepsilon_k|}{1 + r_k^2}$$

$$\Delta I_k^i = \alpha_I \, G_{sign} \lambda_k^i \frac{|e_k \Delta_k^\varepsilon|}{1 + r_k^2}$$

$$\Delta D_k^i = \alpha_D \, G_{sign} \lambda_k^i \frac{|e_k \Delta_k^\varepsilon|}{1 + r_k^2} \tag{13}$$

$$\Delta R_k^i = \alpha_R \, G_{sign} \lambda_k^i \frac{\varepsilon_k}{1 + r_k^2}$$

$$i = 1, \dots, c$$

where $r_k$ is the reference signal, $e_k = r_k - y_{k-1}$ is the difference between the reference signal and the process output and $\lambda_k^i$ is a normalized relative density. The constants $\alpha_P, \alpha_I, \alpha_D, \alpha_R$ are the adaptation gains, and the $G_{sign} = \pm 1$ represent the constant sign of the monotonic plant. Furthermore, the controller gains vector is defined as $\theta_k^i = \left[ P_k^i, I_k^i, D_k^i, R_k^i \right]^T$ and the adaptation of parameters is done only for the current active cloud while the others are kept constant. Adaptation is defined as follows:

$$\theta_k^i = \theta_{k-1}^i + \Delta \theta_k^i \tag{14}$$

The absolute values in (13) prevent undesirable transient response of the system in the starting phase of the adaptation

process. We firstly need to note that the first cloud's parameters are initialized with zero values ($\theta_k^1 = [0,0,0,0]^T$). Second, according to the parameter projection mechanism (which will be discussed later) the controller parameters are defined in the range $[0, \infty]$. Taking into account the zero initialization and the parameter projection we need to avoid negative values of the part $\Delta \theta_k^i$ in (14) in the starting phase. There is no sense of controlling the process with zero values of the controller. In the starting phase we need to ensure that the part $\Delta \theta_k^i$ in (14) will be positive.

Analyzing the sign of the elements that are part of (13) we consider that some of them are always positive. Those are constants $\alpha_P, \alpha_I, \alpha_D, \alpha_R$, normalized relative density $\lambda_k^i$, and the summation $1 + r_k^2$. In our case we also consider a positive monotonic plant ($G_{sign} = 1$). Remaining elements from the (13) are $e_k \varepsilon_k$ and $e_k \Delta_k^\varepsilon$. Only these two products could be negative (the multiplier and the multiplicand have different sing). As we said above in the starting phase we want to avoid such kind of situations. We can also notice that in (13) the parameter $\Delta R_k^i$ do not consist absolute value operator. This is because the parameter present the compensation in operating point that might be positive or negative.

The following scenarios describe all the possible situations that could appear in the starting phase of the process control. These situations (also presented in Fig. 1) analyze the different initial values of the process output ($y_0$) in comparison to the reference value ($r_0$) and moreover, propagate the possible ways of how the process output might react:

1) IF $(y_0 > r_0 \quad$ and $\quad y_k > y_k^r)$ THEN
   $(e_k < 0 \quad$ and $\quad \varepsilon_k < 0 \quad$ and $\quad \Delta_k^\varepsilon < 0)$
   The product $e_k \varepsilon_k > 0$
   The product $e_k \Delta_k^\varepsilon > 0$

2) IF $(y_0 > r_0 \quad$ and $\quad y_k < y_k^r)$ THEN
   $(e_k < 0 \quad$ and $\quad \varepsilon_k > 0 \quad$ and $\quad \Delta_k^\varepsilon > 0)$
   The product $e_k \varepsilon_k < 0$
   The product $e_k \Delta_k^\varepsilon < 0$

3) IF $(y_0 < r_0 \quad$ and $\quad y_k < y_k^r)$ THEN:
   $(e_k > 0 \quad$ and $\quad \varepsilon_k > 0 \quad$ and $\quad \Delta_k^\varepsilon > 0)$
   The product $e_k \varepsilon_k > 0$
   The product $e_k \Delta_k^\varepsilon > 0$

4) IF $(y_0 < r_0 \quad$ and $\quad y_k > y_k^r)$ THEN:
   $(e_k > 0 \quad$ and $\quad \varepsilon_k < 0 \quad$ and $\quad \Delta_k^\varepsilon < 0)$
   The product $e_k \varepsilon_k < 0$
   The product $e_k \Delta_k^\varepsilon < 0$

From the scenarios from 1) to 4) we can notice that only in the second and forth scenario we have a undesirable situation where products $e_k \varepsilon_k$ and $e_k \Delta_k^\varepsilon$ are negative. To avoid this in the starting phase we simply use the absolute values of the mention products (13). In case when the products are positive, using the absolute values do not change the performance of the adaptation law. Our assumption here is that the performance of the process control will be improved in case when the absolute

values are used in the starting phase.

The another question then arises, "When the starting phase finishes (omit the absolute values in (13))?" and when to continue the adaptation without calculating the absolute values? We consider that the period of five time constants ($5\tau$) is enough time to deal with the problem describe above. The time constant is defined in section II according to the chosen linear reference model.

When dealing with adaptive control algorithms one needs to have in mind the potential instability of the system that can occur [22]. There exist many known approaches that make adaptive laws more robust [23], [24]. In [18] and [19] several supervisory mechanisms were included in the adaptive law to prevent parameter drift and instability. In this article we will employ the same ones (dead zone $d_{dead}$ in the adaptive law, parameter projection $[\underline{\theta}, \overline{\theta}]$, leakage $\sigma_L$ in the adaptive law and interruption of adaptation $[u_{min}, u_{max}]$). The general idea behind the dead zone in the adaptive law is that the adaptation is simply switched off if the absolute value of the error is small [19]. All the supervisory mechanisms of the adaptive law are the same as were proposed in [18] and [19].

## IV. SIMULATION STUDY

In the section III we presented four different scenarios that might appear during the starting phase of the process control and furthermore, we propagated the possible trajectories of the output signal. In this section we present more detail analysis of two scenarios (the second and the third). Our goal is to confirm the assumptions exposed in section III which foresees that using the adaptation law with absolute values will improve the performance of the process control in starting phase.

For the purpose of testing our assumptions we chose the model of a heat exchanger (HE) plant. The basic idea of controlling the HE is to control the output temperature (output signal $y_k$) in the second water circuit with the water flow (control signal $u_k$) in the first circuit.

All the simulations shown in this section are started with no a priori knowledge about the controlled process and the RECCo controller is started with zero fuzzy rules. Fuzzy membership functions and fuzzy rules (clouds) are evolved during the process control. Also the parameter settings are the same for all experiments, and only the initial value of the output signal $y_0$ is different, because our goal is to show that the algorithm successfully works for different starting scenarios.

The parameters in RECCo algorithm are separated into tree subgroups (process, evolving and adaptive parameters). In the group of process parameters are the time constant reference model $\tau = 40\,\mathrm{s}$, sampling time of the system $T_s = 2\,\mathrm{s}$, and minimal and maximal value of the reference signal ($r_{min} = 10$ and $r_{min} = 50$). In the second group are evolving parameters which define "why", "when" and "how many" clouds are going to be created. Due to the normalized data space the parameter $\gamma_{max}$ can be fixed to 0.93. The minimal number of data samples between the last created cloud and the new cloud is defined with parameter $n_{add} = 20$, and moreover, the

maximal number of clouds that could be created is defined with $c_{max} = 100$. The third group of parameters define the adaptive law of the RECCo controller and are described in section III. The adaptation gains $\alpha_P, \alpha_I, \alpha_D$, and $\alpha_R$ are equal to 0.1, the dead zone is $d_{dead} = 0.05$, parameter projection is $[\underline{\theta}, \overline{\theta}] = [0, \infty]$, and the leakage is $\sigma_L = 10^{-6}$. The output of the RECCo controller (actuator's interval) is defined in the range $[0, 20]$.

As we already said, in this section we present analysis of the second and third scenario described in section III. For both scenarios we provided experiments where first, the absolute value in (13) is omitted and second the absolute value in starting phase is used. The Fig. 2 presents the situation when the initial value of the output signal (output temperature) is bigger than the reference value ($y_0 = 40 > r_k = 30$). In Fig. 2 the first row present the results without absolute values in adaptation law, and the second row present the results where the new adaptation law is used (13). In top left and bottom left plots in Fig. 2 are shown the reference $r_k$, the model-reference $y_k^r$, and the output signal $y_k$. It is obvious that using the adapting law as (13) provides better results and the RECCo algorithm very quickly evolve and adapt his structure to control the system. In this case we confirmed our assumptions that new adaptation will provide better results. The second pair of plots (in the middle of Fig. 2) show the control signals $u_k$. In the third pair of plots are presented tracking error $\varepsilon_k$ and process error $e_k$.

The third scenario from the section III describe the situation when the initial output value of the controlled system is lower than the desired reference ($y_0 = 20 < r_k = 30$). In this case we assumed that there will be no big difference between the two approaches (without and with absolute value). In Fig. 3 are shown the results of the experiment. First row present the results without and second row with absolute value in the adaptation law (13). In first pair of plots the reference $r_k$, the model-reference $y_k^r$, and the output signal $y_k$ are shown. In this case it is intuitively not so obvious which performance is better. We measured the rise time $T_r$ (black dashed line) and settling time $T_s$ (green dashed line) due to compare the results. For both quality and performance measures the new adaptation law using the absolute values provides better results. The middle plots in Fig. 3 the control signals are shown. We can notice that in this case it is not very big difference. The last plots show the tracking error $\varepsilon_k$ and the process error $e_k$.

## V. CONCLUSION

The RECCo controller is adaptive and evolving algorithm which starts with zero fuzzy rules (zero data clouds) and with zero initial values of the controller parameters. Furthermore, the adaptation law is error driven and downward limited so we want to avoid situations where in the starting phase we have negative error. In this paper we proposed an solution and improvement in the adaptation law were in the starting phase an absolute value of the tracking and process error is used. Through the theoretical and experimental analysis in this paper we showed that using this approach we improve the efficiency
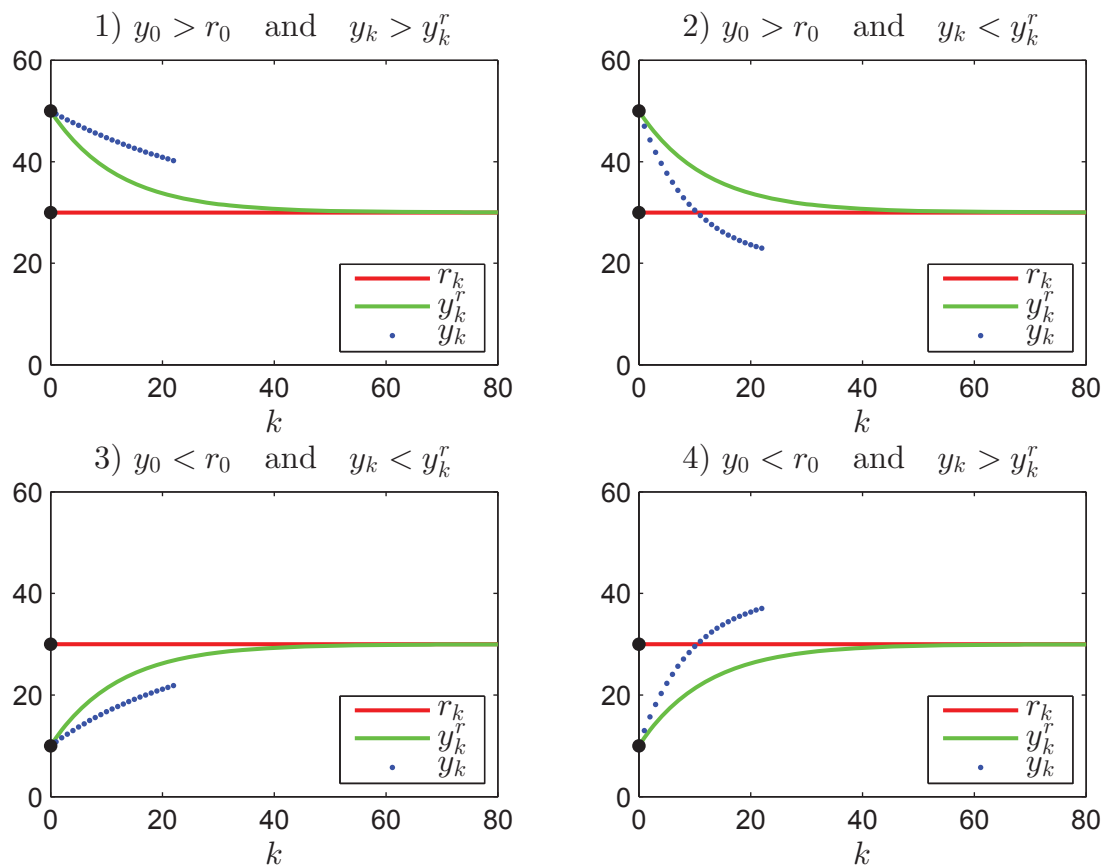
Fig. 1. Four different scenarios for analyzing the starting phase of the process control. In each figure a reference signal $r_k$, model-reference $y_k^r$, and output signal $y_k$ are shown.

of the RECCo controller. These are the real life scenarios when the initial value of the output signal is bigger than the desired reference.

## REFERENCES

[1] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. SMC-3, no. 1, pp. 28–44, Jan 1973.

[2] Y. Bai, H. Zhuang, and Z. S. Roth, "Fuzzy logic control to suppress noises and coupling effects in a laser tracking system," *Control Systems Technology, IEEE Transactions on*, vol. 13, no. 1, pp. 113–121, Jan 2005.

[3] I. Baturone, F. Moreno-Velo, S. Sanchez-Solano, and A. Ollero, "Automatic design of fuzzy controllers for car-like autonomous robots," *Fuzzy Systems, IEEE Transactions on*, vol. 12, no. 4, pp. 447–465, Aug 2004.

[4] P. Bonissone, V. Badami, K. Chiang, P. Khedkar, K. Marcelle, and M. Schutten, "Industrial applications of fuzzy logic at general electric," *Proceedings of the IEEE*, vol. 83, no. 3, pp. 450–465, Mar 1995.

[5] Z. Bingul, G. Cook, A. Strauss, and K. Rashid, "Application of fuzzy logic to spatial thermal control in fusion welding," in *Industry Applications Conference, 1999. Thirty-Fourth IAS Annual Meeting. Conference Record of the 1999 IEEE*, vol. 1, 1999, pp. 627–634 vol.1.

[6] R. Boukezzoula, S. Galichet, and L. Foulloy, "Observer-based fuzzy adaptive control for a class of nonlinear systems: real-time implementation for a robot wrist," *Control Systems Technology, IEEE Transactions on*, vol. 12, no. 3, pp. 340–351, May 2004.

[7] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. SMC-15, no. 1, pp. 116–132, Jan 1985.

[8] G. Feng, "A survey on analysis and design of model-based fuzzy control systems," *Fuzzy Systems, IEEE Transactions on*, vol. 14, no. 5, pp. 676–697, Oct 2006.

[9] Y. Hou, J. Zurada, W. Karwowski, W. Marras, and K. Davis, "Identification of key variables using fuzzy average with fuzzy cluster distribution," *Fuzzy Systems, IEEE Transactions on*, vol. 15, no. 4, pp. 673–685, Aug 2007.

[10] J.-H. Kim, C.-H. Hyun, E. Kim, and M. Park, "Adaptive synchronization of uncertain chaotic systems based on t ndash;s fuzzy model," *Fuzzy Systems, IEEE Transactions on*, vol. 15, no. 3, pp. 359–369, June 2007.

[11] B. Chen, X. Liu, and S. Tong, "Adaptive fuzzy output tracking control of mimo nonlinear uncertain systems," *Fuzzy Systems, IEEE Transactions on*, vol. 15, no. 2, pp. 287–300, April 2007.

[12] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *Int. J. Hum.-Comput. Stud.*, vol. 51, no. 2, pp. 135–147, Aug. 1999. [Online]. Available: http://dx.doi.org/10.1006/ijhc.1973.0303

[13] E. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," *Electrical Engineers, Proceedings of the Institution of*, vol. 121, no. 12, pp. 1585–1588, December 1974.

[14] T.-P. Hong and C.-Y. Lee, "Induction of fuzzy rules and membership functions from training examples," *Fuzzy Sets and Systems*, vol. 84, no. 1, pp. 33 – 47, 1996. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0165011495003053

[15] W.-Y. Wang, Y.-H. Chien, and I.-H. Li, "An on-line robust and adaptive t-s fuzzy-neural controller for more general unknown systems," *Inter-*
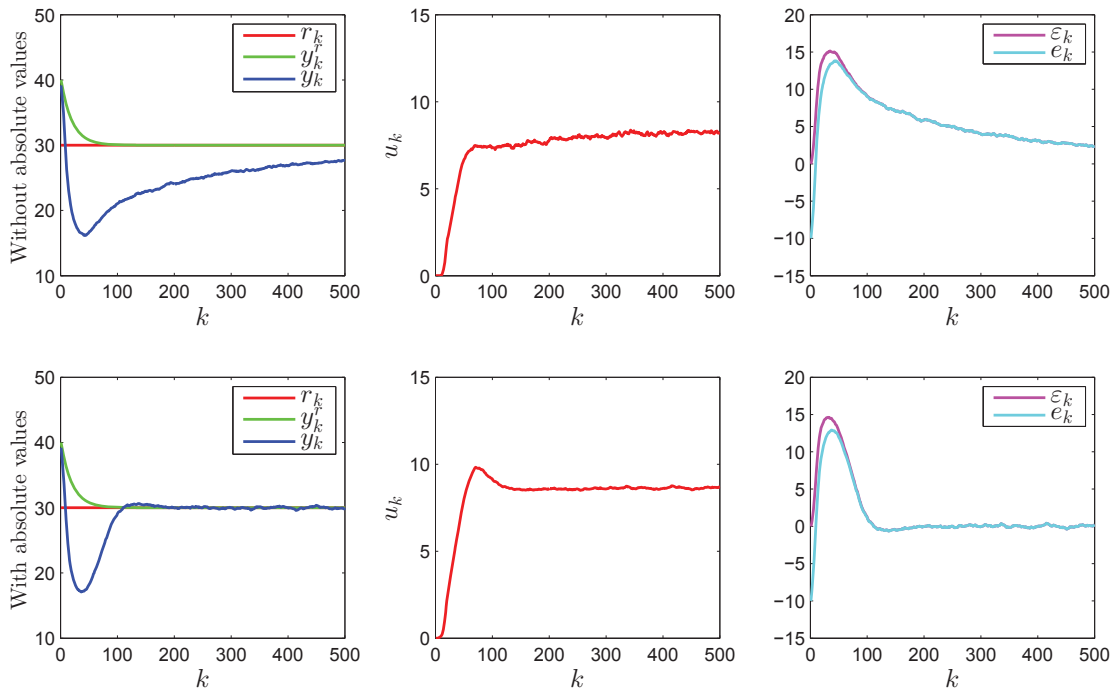
Fig. 2. Compassion between the process control performances where calculating the adaptive law is implemented without (first row) and with (second row) absolute value in (13). Initial value of output signal is $y_k = 40$ and the reference is $r_k = 30$.
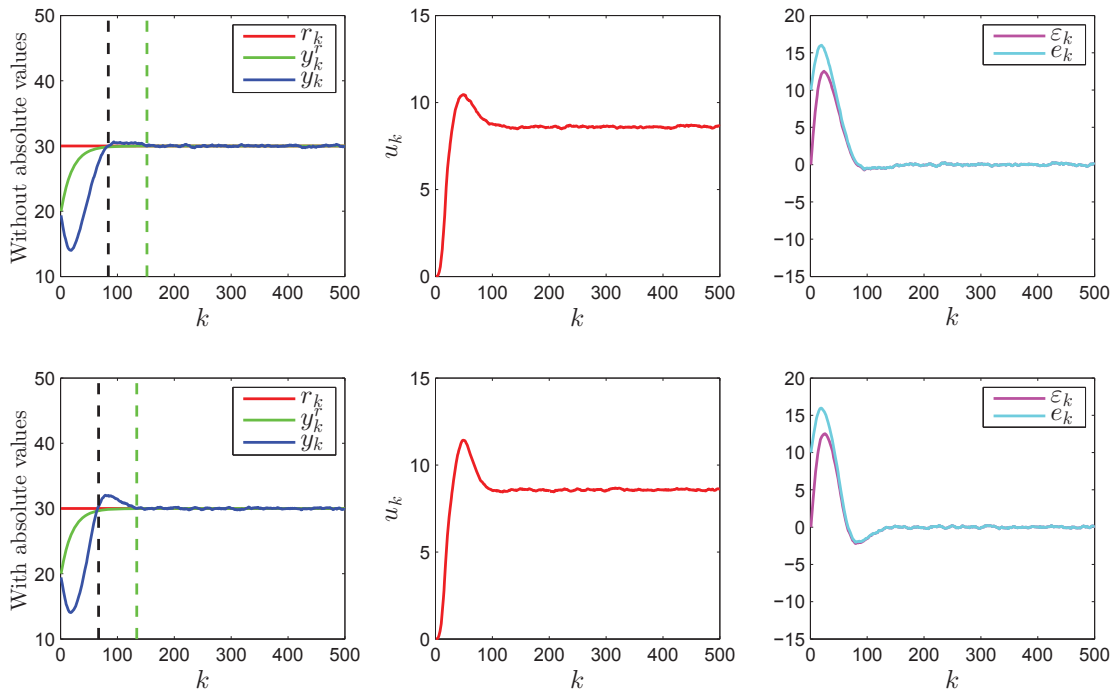


Fig. 3. Compassion between the process control performances where calculating the adaptive law is implemented without (first row) and with (second row) absolute value in (13.) Initial value of output signal is $y_k = 20$ and the reference is $r_k = 30$.

*national Journal of Fuzzy Systems*, vol. 10, pp. 33–43, 2008.

[16] I. Rojas, H. Pomares, J. Gonzalez, L. Herrera, A. Guillen, F. Rojas, and O. Valenzuela, "Adaptive fuzzy controller: Application to the control of the temperature of a dynamic room in real time," *Fuzzy Sets and Systems*, vol. 157, no. 16, pp. 2241 – 2258, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165011406001011

[17] P. Angelov, R. Buswell, J. A. Wright, and D. Loveday, "Evolving rules-based control," in *EUNITE Symposium*, December 2001, pp. 36–41.

[18] P. Angelov, I. Škrjanc, and S. Blažič, "Robust evolving cloud-based controller for a hydraulic plant," in *Evolving and Adaptive Intelligent Systems (EAIS), 2013 IEEE Conference on*, April 2013, pp. 1–8.

[19] I. Škrjanc, S. Blažič, and P. Angelov, "Robust evolving cloud-based pid control adjusted by gradient learning method," in *Evolving and Adaptive Intelligent Systems (EAIS), 2014 IEEE Conference on*, June 2014, pp. 1–8.

[20] S. Blažič, D. Dovžan, and I. Škrjanc, "Cloud-based identification of an evolving system with supervisory mechanisms," in *Intelligent Control (ISIC), 2014 IEEE International Symposium on*, Oct 2014, pp. 1906–1911.

[21] P. Angelov and R. Yager, "Simplified fuzzy rule-based systems using non-parametric antecedents and relative data density," in *Evolving and Adaptive Intelligent Systems (EAIS), 2011 IEEE Workshop on*, April 2011, pp. 62–69.

[22] C. Rohrs, L. Valavani, M. Athans, and G. Stein, "Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics," *Automatic Control, IEEE Transactions on*, vol. 30, no. 9, pp. 881–889, Sep 1985.

[23] S. Blažič, I. Škrjanc, and D. Matko, "A new fuzzy adaptive law with leakage," in *Evolving and Adaptive Intelligent Systems (EAIS), 2012 IEEE Conference on*, May 2012, pp. 47–50.

[24] ——, "Globally stable direct fuzzy model reference adaptive control," *Fuzzy Sets and Systems*, vol. 139, no. 1, pp. 3 – 33, 2003.