Adaptive Control of Non-linear System Using Neural Network

Ivan TAUFER *(CZ)* *ivan.taufer@upce.cz*

Oldřich DRÁBEK (CZ) odrich.drabek@upce.cz

BIOGRAPHICAL NOTES

prof. Ing. Ivan Taufer, DrSc. Graduated from Moscow power institute (1963), faculty of electronics. He obtained at the same institution (1972) Ph.D. degree, specialization Technical cybernetics and D.Sc. degree (1990), specialization Control of technical systems. He was established as associated professor (1980) in specialization Automatic control systems, professor (1998) in specialization Automation of machines and technological processes. He was employed 2 years in Tesla Rožnov as developer, 10 years in Přerovské chemické závody subsequently as production engineer, head of computer centre and head of Automatic control department. He is employed at University of Pardubice from 1974 currently at Department of Process Control, Faculty of Electrical Engineering and Informatics. Nowadays his research area is application of artificial neural networks in modelling and control of complex nonlinear systems. He is member of Academy of Engineering Sciences of Russian Federation, International Informatization Academy of RF, International Academy of Electrotechnical Sciences of RF and emeritus professor at Moscow Power Institute.

prof. Ing. Oldřich Drábek, CSc. Was educated as an electrician by fy. Telegrafia Pardubice (1945), then he attended the Higher Electrotechnical School in Prague (1949). Graduated from ČVUT - Electrotechnical Faculty in Prague in 1953. He worked in Tesla Pardubice as development-engineer in area of the radar – servomechanisms and other military apparatus. From 1966 he was employed as a senior lecturer on VŠCHT - Pardubice at department of Automatization of chemical technologies. In 1970 he obtained Ph.D. degree and six years latter he was established as an associated professor. In those time, his research area was the experimental identification and control systems. He is author or co–author of eight textbooks and many of the research reports, drawn up within the framework of the cooperation with industry. In 1991, he was established as a professor of Automatic Control Systems. From 1993 he is interested in modelling and control systems, using artificial neural networks.

KEY WORDS

PS controller, adaptive control, neural network, perceptron

ABSTRACT

The paper deals with one of possible methods for control of non-linear dynamical system, namely with the adaptive neural PS controller (ANPSC). Its principle is in on-line refinement of the model system and then in the adaptive adjustment of the controller parameters. The model of the plant is realized with a two-layer feedforward neural network.

The control part of the ANPSC has the structure of the classical PS controller, and it is also created in the form of the neural network (simple perceptron). The simulation calculations were performed in MATLAB.

INTRODUCTION

This work was motivated by an attempt at spreading information about the results of simulation of control of a non-linear dynamical system by means of adaptive neural PS controller (ANPSC), analysis of the controlled system, goals of the control, choice of learning coefficient of the neural network representing the model of the plant, choice of learning coefficient of the neural network of controller, and comparison of the quality of control process by means of this type of controller with a classical PS controller created also in the form of neural network (hereinafter classical PS controller).

It is presumed that the reader is acquainted with classical methods of identification and control of systems as well as with the methods based on utilization of artificial neural networks (ANN), which are applicable to the same purpose. If not, then he/she can find the necessary information about these principles of identification and control in [1], [2], [3], [4], [5], [6], [7], [8], [9], [10].

It is well known that adaptive control [2], [9], [11], [12], [13] is always realized on the basis of knowledge of mathematical model of the controlled process. If the properties of this process change in the course of control, then it is necessary to on-line readjust also the parameters of the controlling system. The algorithms of this control are well developed for the cases of the model being presented in the form of a differential or difference equation. However, construction of such model necessitates the possibility of linearization in the surrounding of the working point [2].

The present development of artificial neural networks inter alia as means for direct modeling of non-linear systems offers the possibility of application of these networks also in schemes of adaptive control. The controller itself can then be realized as a neural network. One of the ways of realization of this type of control is described below.

ANALYSIS OF CONTROLLED SYSTEM

Let us presume that the controlled system is described by difference equation [14]

$$y_s\left(k\right) + y_s^2\left(k-1\right) - A\left(k\right)y_s\left(k-1\right) = B\left(k\right)u\left(k-1\right) \quad (1)$$

where u(k) is the input signal and ys(k) is the output signal of the controlled system. The nominal values of parameters A and B (when they are not changed in the course of control): $A = 1{,}2$ and $B = 0{,}92$. The properties of system (1), which is presented in [3] without any closer description, are most simply determined by experiment, namely on the basis of analysis of results of simulation calculations of responses of the system to a sequence of input impulses:

$$u_i\left(k\right) = \begin{cases} u_i & for\ k \in \left(0;15\right) \\ 0 & for\ k \in \left(15;30\right) \end{cases} \quad (2)$$

The responses to the impulses with the magnitude $u_i = 0{,}05;\ 0{,}273;\ 0{,}65;\ 0{,}87;\ 1{,}087;\ 1{,}15$ are given in Fig. 1.

It can be seen that:

■ *The character of responses of system depends on the magnitude of input signal u_i (Fig. 1), the results of analysis of this experiment can be summarized as follows:*
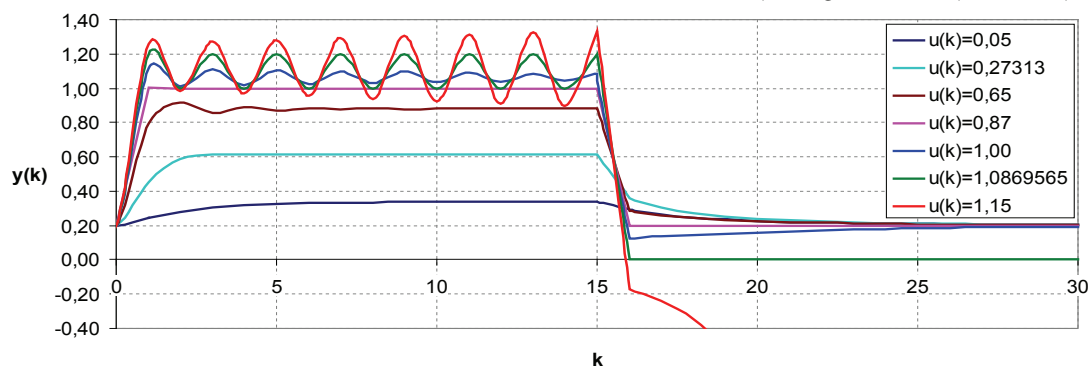*-the course of output signal is non-periodic up to*

Fig. 1 *Impulse responses*

$u \leq 0,2731$,

-in the range $u \in (0,2731; 0,87)$ the courses of responses are oscillating, dampened, with an increase and subsequent decrease in the oscillations,

- the oscillation maximum lies at $u = 0,65$,

-at the value $u = 0,87$ the output signal stabilizes after the first step both at the ascent of impulse and at its descent,

-in the range of $u \in (0,87; 1,087)$the course of responses is oscillating, dampened, with growing oscillations,

-at the value of $u = 1,087$ the system is at its stability limit.

■ At $u > 1,087$ the system is unstable.

■ When the impulse decreases to zero value, the output signal of the system (in the case of the preceding impulse magnitude being $u < 1,087$) stabilizes at the value $y_s = 0,2$, i.e. the difference equation (1) is in this range defined for the initial conditions $u(0) = 0$; $y_s(0) = 0,2$ (see Fig. 1).

■ When the impulse decreases from the value $u = 1,087$ to zero, also the value of y decreases to zero.

■ In the case of impulse decrease from the value $u > 1,087$ to zero, when the system is unstable, the solution is unstable too.

■ The static characteristic obtained from stabilized values of y is given in Fig. 2.

The above-given calculations show that the system is non-linear in stabilized states as well as in dynamical properties, and its working range is in the interval of the input quantity $u \in (0,2; 1)$ and output quantity $y_s \in (0,2; 1)$. The anomalies described in items 4) and 5) lead to the conclusion that the value of action signal $u(k)$ in the case of control must not decrease below zero value.
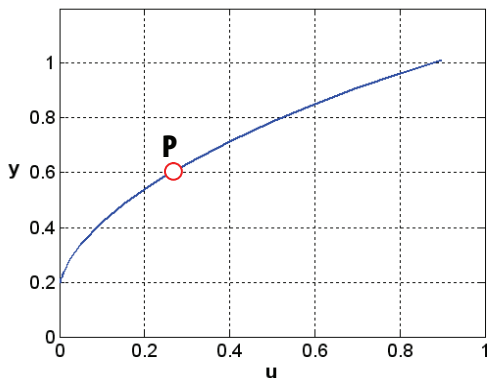

Fig. 2 *Static characteristic*

GOALS OF CONTROL

The below-presented simulation calculations for the system control using ANPSC pursue the following goals:

■ *After starting the control algorithm, an "as fast as possible" adjusting is required for the controlled quantity $y_s(k)$ to its requested value into the starting working point **P** without any distinct overshoot of the regulated quantity. The working point has coordinates of the centre of working range of the system (see Fig. 2).*

■ *On moving from the working point **P** to another working point, we want in principle the same as in the preceding point, being aware of the fact that the term "the fastest" is relative. We want the quality of control process to be verified in the whole working range of the system for the change of required value of regulated quantity in the form of step function $r(k) = 0.6; 1; 0.6; 0.2; 0.6$ with the length of one step equal to 25 steps.*

■ *In the context of the fact that the changes ΔA and ΔB of the parameters $A(k)$ and $B(k)$ of the system can have the form of impulses, and the failures in the action quantity $v_u(k)$ and in the regulated quantity $v_{ys}(k)$ can have the form of jumps (see equations (3a,b) and (3c,d)) we are interested in the velocity and efficiency of elimination of the influence of these failures upon the regulated quantity.*

■ *The action quantity must fulfil the condition $u(k) \geq 0$.*

$$A(k) = \begin{cases} 1.2 & \text{for } k \in (0; 125) \\ 1.32 & \text{for } k \in (126; 150) \\ 1.2 & \text{for } k \in (151; 300) \end{cases} \tag{3a}$$

$$B(k) = \begin{cases} 0.92 & \text{for } k \in (0; 175) \\ 1.012 & \text{for } k \in (176; 200) \\ 0.92 & \text{for } k \in (201; 300) \end{cases} \tag{3b}$$

and

$$v_u(k) = 0.1 \text{ for } k \in (225; 300) \tag{3c, d}$$

$$v_{ys}(k) = -0.1 \text{ for } k \in (250; 300)$$

Hence, in principle, the problem lies in assigning the "task of monitoring", which is typical of the control realized by means of a servo, and the "task of stabilization", which ensures maintaining of the regulated quantity at the required value during all sorts of failures affecting the system.

In the simplest case, the successfulness of solution can be evaluated on the basis of merely visual assessing of courses of quantities of the control process using verbal expressions such as "better - worse", "satisfactory - unsatisfactory" etc., or analytically by means of the mean-quadratic criterion (see below, chapter 3), or by determination of the value of control step at which the required value of regulated quantity is achieved (see below, chapter 4).

The above-defined goals of control show that the task will lie in suggesting such control system whose properties fall within the area of robust control realized, e.g., by the adaptive control algorithm mentioned. This algorithm enables adaptation of controller to various transfer properties of the controlled system and contributes to more effective elimination of all sorts of failures affecting the system etc.

DESCRIPTION OF BLOCK SCHEME OF ANPSC
The regulation circuit with ANPSC is formed by mins of two partial feedback circuits (see Fig. 3) and one common neural controller NC.

The first feedback circuit consists of:

■ *Increment controller NC realized by a perceptron (e.g., see [3] or [9]) with the weights q_i, $i = 0, 1$, which generally have the character of coefficients of classical PS controller.*

■ *Non-linear controlled system (S), in our case in the form of equation (1).*

The second feedback circuit consists of:

■ *The above-mentioned neural controller NC.*

■ *Model of controlled system in the form of feedforward neural network (FFNN) with the input u, output y_m and the connection weights γ_i.*
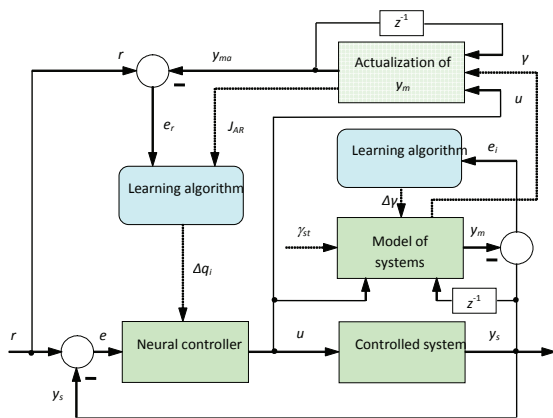

Fig. 3 *Block scheme of ANPSC*

■ *Learning algorithm of FFNN. Before starting the control algorithm of ANPSC, the FFNN is trained off-line by means of BP - algorithm. The result of training is the values of weights γ_i of FFNN calculated from the equation.*

$$\gamma_i(k) = \gamma_i(k-1) + \Delta\gamma_i(k) \qquad (4)$$

The changes of weights $\Delta\gamma_i(k)$ of FFNN are calculated by the method of Back-propagation (BP - algorithm), which starts from the relationship.

$$\Delta\gamma_i(k) = -\alpha\frac{\partial E_s(k)}{\partial\gamma_i(k)} \qquad (5)$$

In this equation, α is the learning coefficient of FFNN, and Es(k) is quadratic function of deviation.

$$E_S = \sum_{k=1}^{N} e(k)^2 = \sum_{k=1}^{N}(y_s(k) - y_m(k))^2 \qquad (6)$$

■ *The "actualized" FFNN (AFFNN) whose output quantity y_{ma} is a function of momentary values of connection weights γ_i of FFNN calculated during its on-line training step, realized within the framework of control of the regulated system.*

■ *Learning algorithm of NC which in the course of control ensures continuous refinement of the starting values q_{ist} of ANPSC. The result of learning the NC is the values q_i of weights of NC calculated from the equation.*

$$q_i(k) = q_i(k-1) + \Delta q_i(k) \qquad (7)$$

The weight changes $\Delta qi(k)$ of NC are calculated from equations that are analogous to equations (5) and (6), i.e.:

$$\Delta q_i(k) = -\beta\frac{\partial E_r(k)}{\partial q_i(k)} \qquad (8)$$

In this equation, β is the learning coefficient of NC, and $E_r(k)$ is quadratic function of deviation.

$$E_r = \sum_{k=1}^{N} e_r(k)^2 = \sum_{k=1}^{N}(r(k) - y_{ma}(k))^2 \qquad (9)$$

The control circuit is described in [2] more briefly and in the context with classical adaptive control with description of adaptive controller as well as with system of difference equations. The BP - algorithm used for control of ANSPC is developed in [5], and the derivation of learning algorithm of NC is available in [13].

Briefly speaking, the design procedure and putting of ANPSC into operation consists in the following steps:

■ *Adjustment of starting values of parameters q_{ist} of controller NC, which are used further as starting values of ANPSC.*

■ *Construction of model of controlled system in the form of FFNN and its training off-line.*

■ *Continuous refining of the values q_{ist} by learning NC with simultaneous on-line training of FFNN.*

ADJUSTMENT OF STARTING VALUES q_{ist} AND CONTROL BY MEANS OF NEURAL CONTROLLER NC ITSELF

The starting values of parameters qist are adjusted in the first feedback circuit (see chapter 3). It is presumed that NC is in the form of incremental PS controller:

$$u(k) = q_0 e(k) + q_1 e(k-1) + u(k-1) \qquad (10)$$

with parameters q_i, $i = 0, 1$, where $e(k)$ is the control error and $u(k)$ is the output signal of the controller. In spite of the fact that in our case the controlled system is non-linear, its dynamical properties in the surroundings of origin (see Fig. 2), which are analogous to the properties of a linear system of the 1st order, justify application of equation (3). The controller parameters are adjusted tentatively, using the method of "trial and error".

For the purpose of adjustment of NC parameters, the value $\beta = 0$ is set in the second partial control circuit (i.e., this circuit is put out of operation), and then we work with the first control loop consisting of controller (10) and system (1). Thereafter, for the required value of controlled variable $r(k)$ (adjustment of system regime into the working point) we find the values of parameters q_0 and q_1 which approximately correspond with the requirement stated in item 1) of the goals formulated in chapter 2, e.g.:

$$q_0 = 0,45 \qquad q_1 = -0,05 \qquad (11a, b)$$

The proportional component of controller is equal to parameter q_0, the increment of control action - this is an incremental controller - is $\Delta u = q_0 - q_1$. The course of control response during controlling the system (1) by the NC controller with the coefficients given in equation (11a,b) is presented in Fig. 4 for the first ten regulation steps. The controlled variable reaches the set point $y_s(k) \approx r(k) = 0,6$ in the 7th step, with the overshoot $\Delta y_{smax} < 2\%$ in the 3rd step.
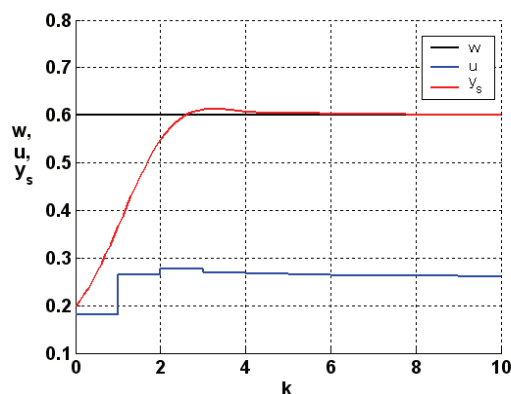

Fig. 4 *To the control with Neural Network Controller*

Further, the parameters q_0 and q_1 (11a,b) are referred to as the starting values q_{ist}, $i = 0, 1$ for the adaptive controller ANPSC.
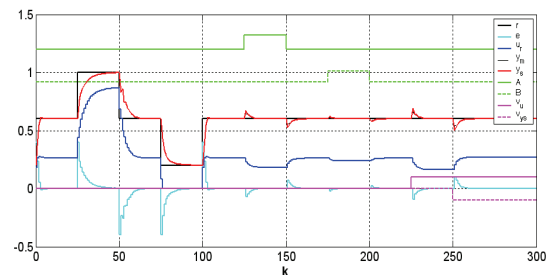

Fig. 5 *System control whith application of PS Controller*

Figure 5 presents the course of control response for the set point tracking of controlled variable in the form of stepped function $r(k)$, defined in chapter 2, for changes in parameters A, B and disturbances v_u and v_{ys} - see equations (3a,b) and (3c,d). It can be seen that deviations towards higher (lower) required values are connected with slowing down (speeding up) of the control response, which is in agreement with the course of static characteristic in Fig. 3. The oscillating course of quantities of the control loop

at $r(k) = 0,2$ is removed by programmed restriction of control action to $u(k)_{min} = 0$, as it was required in "goals of control" in chapter 2. The courses of responses to impulse changes in parameters A and B indicate a relatively fast elimination of the influence of these failures upon the course of regulation process. The same also applies to the elimination of influence of jump failures in the action quantity ν_u and in the controlled variable ν_{ys}.

Mere visual inspection shows that the required goals of control were not fully fulfilled. On the basis of analysis of simulation calculations results (see Tab. 1) it can be stated that:

Jump w	Reaching of required value	Regulation step
0,6 - 1,0	95%	8
	100%	22
1,0 - 0,6	95%	9
	100%	24
0,6 - 0,2	95%	11
	100%	25
0,2 - 0,6	95%	3
	100%	16

Tab. 1 *Successfulness of regulation by means of classical PS controller during changes of required value*

■ *On moving from the working point $r(k) = 0,6$ to the set point $r(k) = 1$, the controlled variable attains 95% of set point in the 8th step, while stabilized value is not attained: beginning from the 22nd step y(k) oscillates.*
■ *On changing from $r(k) = 1$ to $r(k) = 0,6$; 95% of the set point is attained in the 9th step and stabilization in the 24th step.*
■ *When moving from the working point $r(k) = 0,6$ to the required value $r(k) = 0,2$; 95% of the set poin is attained in the 11th step, while in the 25th step the set point is not fully attained. This is due by the necessary restriction of the action range to zero value.*
■ *When changing from $r(k) = 0,2$ to the working point $r(k) = 0,6$ the system behaves like during adjustment of the controller parameters in the first phase of control. The overshoot is in the 3rd step, 95% of set point is attained in the 5th step and stabilization in the 16th step.* The results of elimination of failures are given in Tab. 2.

Failure	Attaining of requested value	Regulation step
$A\uparrow$	95%	6
	100%	15
$A\downarrow$	95%	6
	100%	17
$B\uparrow$	95%	3
	100%	14
$B\downarrow$	95%	4
	100%	15
υ_u	95%	7
	100%	19
υ_{ys}	95%	6
	100%	18

Tab. 2 *Successfulness of regulation with classical PS controller during failures*

From the given facts it follows that the classical PS controller without problems eliminates the influence of disturbances in roughly the same steps of control: the controlled variable attains 95% of requested value in the 7th to 8th step, while complete stabilization is attained in the 14th to 19th step.

Another possibility of evaluation of successfulness of control is the calculation of the mean-quadratic criterion of the deviations between the set point and actual value at the output of system (control error) according to the relationship.

$$E_a = \sqrt{\frac{\sum_{k=1}^{N}\left(r(k) - y_s(k)\right)^2}{N - 1}} \tag{12}$$

In the problem dealt with here, the value of this criterion calculated for the whole region of control is equal to $E_{a0} = 0,067902$, for the region of changes of set point $E_{a1} = 0,10396$, and for the region of disturbances $E_{a2} = 0,016830$.

Thus there arises a possibility to "improve" the course of control process by application of ANPSC. For this purpose, however, it is necessary to deal with construction and training of the feedforward neural network as the model of system.

MODEL IN THE FORM OF FFNN AND ITS TRAINING OFF-LINE

A more detailed treatise of topology design of FFNN, its training and testing is available, e.g., [7]. In our case the suggested FFNN has the following topology (without claim that the structure is optimal):

- *Number of source nodes: 2.*
- *Number of hidden layers: 1.*
- *Number of neurons in a hidden layer: 3.*
- *Number of neurons at the output: 1.*
- *Activation function of hidden neurons: unipolar sigmoid with steepness s = 1.*
- *Activation function of output neuron: linear with steepness K = 1.*

The starting point for construction of training set of FFNN is the stepped function $u(k)$ = 0,2; 0,4; 0,6; 0,8; 1; 0,8; 0,6; 0,4; 0,2; 0 with the number of steps in each step equal to 30. The training is realized by means of the BP - algorithm.

The results of training can be seen in the courses of the variables $u(k)$, $ys(k)$ and $y_{ma}(k)$ in Fig. 6 for the following values of parameters of BP - algorithm:

- *The learning coefficient determined experimentally and ensuring the minimum value of criterion function and its most rapid convergence: α = 0,3.*
- *The maximum number of training periods enabling visual monitoring of fulfilment of the abovementioned condition: p_{emax} = 10 000.*
- *The starting connection weights w(j) (between the hidden layer and output neuron) and v(i,j) (between the input nodes and hidden layer): chosen at random - see Tab. 3.*

j	0	1	2	3
w(j)	0,1	0,1	-0,5	0,2
i/j	0	1	2	3
υ(i,j) 0		-0,5	0,4	-0,3
1		0,1	-0,2	0,3
2		0,25	-0,35	0,45

Tab. 3 *Starting values of connection weights*

From Fig. 6 it can be seen that in this way constructed and trained FFNN as a model of system (1) very well approximates its dynamical behaviour. The character of response $y_s(k)$ of the system and $y_m(k)$ of FFNN to the stepped function $u(k)$ is in accordance with the character of responses to the jump functions in Fig. 1, i.e. increasing value of the input signal is connected with increasing amplitude of the dampened

oscillations of the corresponding responses, practically applying $y_s(k) \approx y_m(k)$. The partial responses to individual steps of function $u(k)$ have overshoots of periodic, dampened character, this feature being more marked for the responses approaching the maximum of the stepped function.
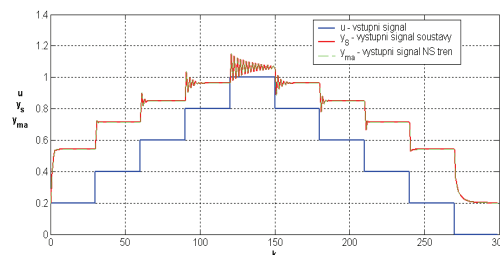

Fig. 6 *Course of training*

The obtained values of connection weights are given in Tab. 4.

j	0	1	2	3
w(j)	0,0428	2,5639	-2,5727	0,0255
i/j	0	1	2	3
υ(i,j) 0		-0,698	1,499	0,028
1		-0,269	-2,163	-3,129
2		-0,237	0,664	0,604

Tab. 4 *Resulting values of connection weights for a = 0,3*

CONTROL WITH APPLICATION OF ANPSC

The above-obtained values of connection weights of FFNN and starting values of parameters q_{ist} of neuron controller are starting parameters of the algorithm of ANPSC.

After starting ANPSC there proceeds on-line, gradually in individual steps - briefly speaking - refinement of the starting values of weights γ_{st} of FFNN, actualization of the output quantity y_{ma} by means of the block AFFNN, and subsequent training of NC involving readjustment of parameters q_0 and q_1 of controller so as to ensure the minimum value of the control error e (6) and quadratic deviation e_r (9).

The aim of the simulation calculations then is (like in the case of off-line training of FFNN looking for the optimum value of learning coefficient of FFNN α ensuring the minimum value of the criterion function E_a (6)) to find such value of learning coefficient β of NC that will ensure the required course of the control process.

For this purpose, a series of simulation calculations were carried out for $\beta = 0{,}1$ to $2{,}8$ and the same changes of the set point $r(k) = 0{,}6$; $1{,}0$; $0{,}6$; $0{,}2$; $0{,}6$, failures of parameters of controlled system A and B, and failures of the action quantity and regulated quantity $v_u(k)$, $v_{ys}(k)$, like in the case of control of the system by means of classical PS controller (3a, b, c, d). The obtained results presented as the minimum numbers of control steps needed for stabilization of the regulated quantity to 95% and/or 100% of the required value of the regulated quantity for the individual jump changes, and the corresponding values of learning coefficients of NC β_{opt} are given in Tab. 5.

jump	β_{opt}	0,1	0,3	0,4	0,7	1,1	1,4
0,6 - 1,0	95%				*4*		
	100%			*22*			
1,0 - 0,6	95%					*4*	
	100%						*11*
0,6 - 0,2	95%	*11*					
	100%	*25*					
0,2 - 0,6	95%		*4*				
	100%		*5*				

Tab. 5 *Limit values of regulation steps with ANPSC*

From the table it is clear that the optimum value of learning coefficient β_{opt} changes depending on the change of the required value. This phenomenon can be explained by the non-linearity of the controlled system (1). In the case of changes of the required value in the range of 0,6 - 1,0 - 0,6, where the amplification of system is smaller than that in the working point P = 0,6, the learning value β_{opt} of NC is high; on the other hand, in the case of the changes in the range of 0,6 - 0,2 - 0,6, where the amplification of the system is higher than that in the working point **P**, the learning value β_{opt} of NC is smaller.

Comparison with the results of control by means of classical PS controller (see Tab. 1) shows that, apart from the jump 0,6 - 0,2 (with artificially introduced restriction of the action quantity to $u \geq 0$), the control process was accelerated, i.e. its quality was improved. The same can be documented also using the mean-quadratic criterion (12), see Fig. 7. These courses were calculated for the whole region of simulated control (course of E_{a0}), as well as for the region of changes of set point (course of E_{a1}) and

for the region of disturbances (course of E_{a2}). Circle denotes here the minimum values, which were obtained for the whole region of control as well as for the region of changes of set point at the value of learning coefficient $\beta \doteq 2$. The course in the region of disturbances is almost unchangeable and equal to the value attained in the control with classical PS controller. This can be explained by the fact that both the changes of system parameters and disturbances are introduced into the system under the regime of the working point in which the coefficients of classical PS controller were determined.
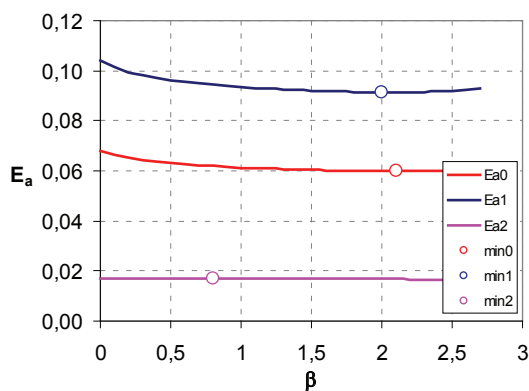

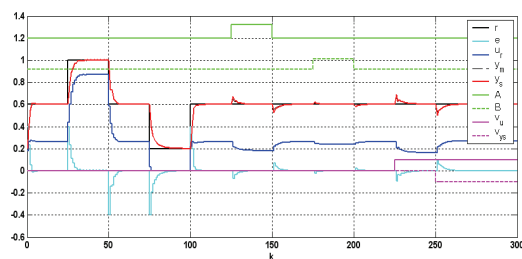Fig. 7 *Course of mean-quadratic deviation Ea after change of learning coefficient of NC*

Fig. 8 *Regulation process in ANPSC regime for β=1,0*

From what has been said it follows that an unambiguous determination of learning coefficient of NC controller for changing conditions of control is impossible. Therefore, its choice will depend not only on a more detailed analysis of behaviour of the control loop but also on the control system designer's experience.

The control course for a chosen value of learning coefficient of NC $\beta = 1{,}0$ is presented in Fig. 8. Even merely visual inspection shows a distinct improvement in the control quality as compared with the

control by means of classical PS controller. For illustration, Fig. 9 also presents the courses of adaptation of parameters of NC controller. The resulting adapted values of weights (parameters) of neural controller are $q_0 = 0{,}6070$ and $q_1 = -0{,}1917$.

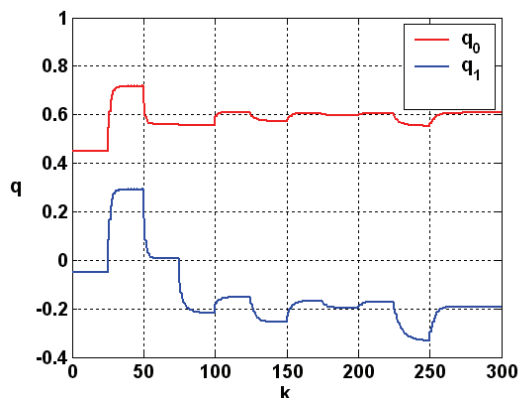Table 6 gives the resulting adapted values of weights of FFNN.

Fig. 9 *Learning of neural controller*

j	0	1	2	3
w(j)	*0,0393*	*2,5592*	*-2,5771*	*0,0187*
i/j	0	1	2	3
$v(i,j)$	0	*-0,6980*	*1,5005*	*0,0279*
	1	*-0,2691*	*-2,1623*	*-3,1275*
	2	*-0,2372*	*0,6593*	*0,6106*

Tab. 6 *Resulting values of adapted connection weights*

CONCLUSION

The quality improvement of control process with application of ANPSC, as compared with the results of control by means of NC controller, is obvious, but it cannot be considered astonishing. The obtained result only confirms the known fact that a simple classical PS controller (in the form of either the difference equation or neural network) is able to successfully control not only linear system but (with certain limitations) also simpler non-linear systems. The above-presented system (1) belongs to such category. However, on the other hand it is not a mere cliché to claim that it is sometimes suitable - for control of more complex non-linear systems - to adopt also more complex types of controllers, such as ANPSC, predictive controller, or hybrid controller. That is why it is useful to become well acquainted

with the principles and properties of these controllers so as to be able to adopt them if necessary. Of course, application of a particular controller also depends on the specification of the control task, i.e. of the aims to be pursued by the control.

REFERENCES

[1] DRÁBEK, O., MACHÁČEK, J.: Experimental Identification. (In Czech). Pardubice: VŠCHT, 1987

[2] MIKLEŠ, J.; FIKAR, M. Modeling, Identification and Control of Processes 2. Identification and Optimum Control. (In Slovak). Bratislava: Vydavateľstvo STU Bratislava, 2004. ISBN 80-227-2134-4

[3] DRÁBEK, O., MACHÁČEK, J.: Adaptive Control. (In Czech). Pardubice : VŠCHT, 1992

[4] SINČÁK, P., ANDREJKOVÁ, G.: Neural Networks. Engineering Approach. (In Slovak) Vol 1 and 2. Košice: Elfa s.r.o., 1996. p. 107 and p. 63. ISBN 80-88786-38-X a ISBN 80-88786-42-8

[5] FAUSETT, L.V. Fundamentals of Neural Network: Architectures, Algorithm and Applications. New Persey: Prentice Hall, 1994

[6] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (6). CHEMagazin, 6 (XVI), 2006, pp. 31-33. ISSN 1210-7409

[7] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (7). CHEMagazin, 3 (XVII), 2007, pp. 38-41. ISSN 1210-7409

[8] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (8). CHEMagazin, 4 (XVII), 2007, pp. 28-30. ISSN 1210-7409

[9] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (9). CHEMagazin, 6 (XVII), 2007, pp. 32-34. ISSN 1210-7409

[10] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (12). CHEMagazin, 5 (XVIII), 2008, pp. 28-30. ISSN 1210-7409

[11] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In

Czech). (13). CHEMagazin, 6 (XVIII), 2008, pp. 28-30. ISSN 1210-7409

[12] DRÁBEK, O., MACHÁČEK, J.: Adaptive Control. (In Czech) Pardubice : VŠCHT Pardubice, 1992

[13] TAUFER, I., DRÁBEK, O., SEIDL, P.: Artificial Neural Networks - Theory and Applications (In Czech). (14). CHEMagazin, 1 (XIX), 2009, pp. 26-28. ISSN 1210-7409

[14] TAN,Y. Architecture for Adaptive Neural Control. Neural Control Journal A. vol. 34, No 4, 1993

[15] KOZÁK, Š., KAJAN, S.: Adaptive Control of Nonlinear Dynamical Systems Using Neural Networks. In Proceeding of the 2nd Slovak Conference on Artificial Neural Networks SCANN 98, November 10 - 12, 1998, pp. 72-76