NEURAL NEAR-TIME-OPTIMAL CONTROL OF A SYSTEM WITH QUAZISTATIONARY SWITCHING LINE

STOYANOV VALENTIN, MIHAILOV NIKOLAY

University of Rousse, Bulgaria

Abstract: This paper deals with the time-optimal and near-time-optimal control of a system with quazistationary switching line. The control law is realized by approximation of the optimal switching line by artificial neural network. The neural approximation of the optimal switching line is compared with the linear and fuzzy approximation of the switching line and with the exact switching curve. Some simulations of the time-optimal and near-time-optimal control systems are done.

Key words: Neural network, optimal quazistationary switching line, time-optimal control, near-time - optimal control

1. INTRODUCTION

The changing of position during two operations executed by a certain class of the industrial devices is “lost” time from the technological efficiency point of view. Thus, the time-optimal control understood as a transfer of any initial state to any target state for minimum time is an important, economical problem.

The theory of optimal control has given a considerable place to synthesis of time-optimal control of a system with quazistationary switching line [2]. The quazistationary line is a multitude of points in the plane, which are time unchanging but depend on reference mode. The position of the switching line is changed with changing the reference parameters and mode.

The exact realization of the optimal switching line is a complicated technical problem. That is why the control law is realized by more simple approximation function or by fuzzy or neural approximation of the optimal switching line [3, 4].

The purpose of this paper is to realize a near-time-optimal control of a system with quazistationary switching line applying neural approximation of the optimal switching line.

2. OPTIMAL QUAZISTATIONARY SWITCHING LINE

The schematic diagram of a time-optimal closed-loop system is shown in the fig.1. The dynamics of the controlled object is given by

$$\left| \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u, \end{aligned} \right. \qquad (1)$$

where x_1, x_2 is position and velocity of the object and the control function is

$$|u| \le U_{max} . \tag{2}$$

The reference is described with the following equation

$$y_0(t) = \lambda_0 + \lambda_1 t + \lambda_2{}^2 t^2 , \tag{3}$$

where λ_0, λ_1 and λ_2 are constants.

The motion is admissible if the second derivative of the reference is limited

$$|2\lambda_2| \le U_{max} . \tag{4}$$

The solution of (1) determines the optimal quazistationary switching line under reference (3) and the limitations (2), (4)

$$V_0(x_1, x_2) = x_1 + \frac{x_2{}^2 \operatorname{sign} x_2}{2(U_{max} - 2\lambda_2 \operatorname{sign} x_2)} . \tag{5}$$

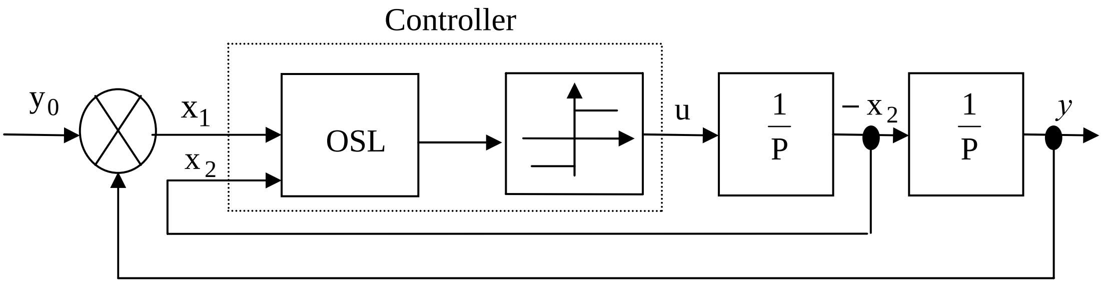

Fig.1. Schematic diagram of a time-optimal (near-time-optimal) control system

The optimal function u is

$$u = U_{max} \operatorname{sign} V_0(x_1, x_2) . \tag{6}$$

The obtained optimal quazistationary switching line (OQSL), fig.2, is a non-linear function of state variable. The exact realization of OQSL requires non-linear functional transducers and precise multiplier links. The realization of OQSL is additionally complicated because it depends on parameter λ_2 of the reference. For that reason, after determining the OQSL, arises the problem to synthesize near-time-optimal control. The control law is realized by approximation of OQSL with linear function or using the approximation feature of fuzzy multitudes and neural networks.

3. NEURAL APPROXIMATION OF THE OPTIMAL SWITCHING LINE

The control law is realized by approximation of the optimal switching line by artificial neural network (ANN).

The input parameters for the first layer of the ANN are the state variables x_1, x_2 and the parameter λ_2 of the input signal (3). The last layer of single neuron provides the

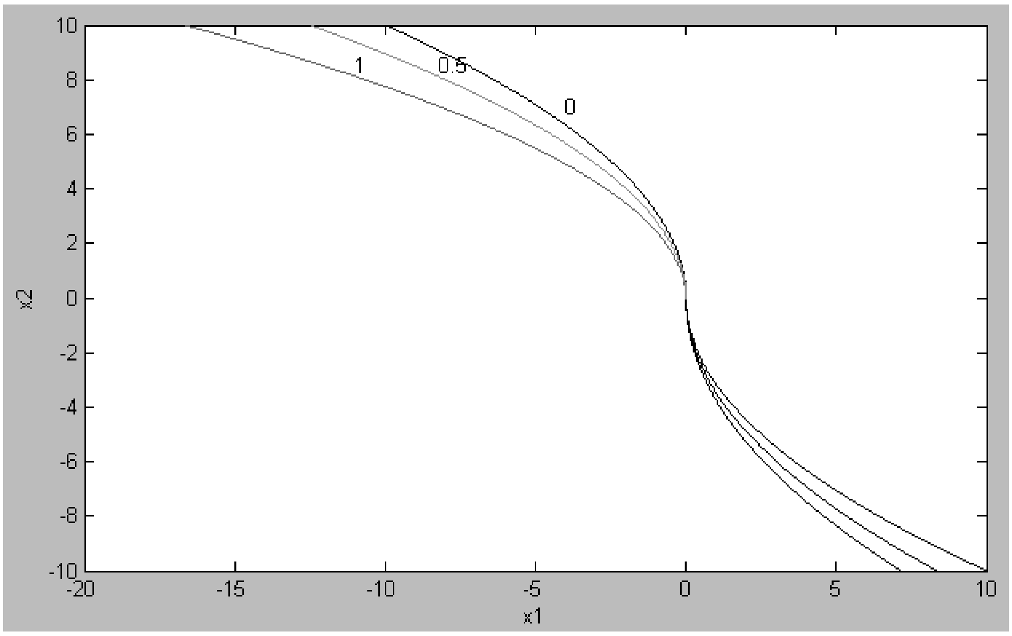

Fig.2. Optimal quazistationary switching line for $\lambda_2 = 0, \lambda_2 = 0{,}5$ *and* $\lambda_2 = 1$

single output parameter – control function u.

The neurons were positioned in three layers. The selected structure of the neural network is shown in fig.3. The ANN was realized by Neural Network Toolbox of Matlab [1].

he following symbols for the weight and bias are used:

$LW\{1,1\}10 \times 3$ - input weight {layer number, input number} neuron number $\times$ input parameter number;

$LW\{2,1\}5 \times 10$ - layer weight {layer number, input layer number} neuron number $\times$ input parameter number;

$LW\{3,1\}1 \times 5$ - layer weight {layer number, input layer number} neuron number $\times$ input parameter number;

$B\{1\}10 \times 1$ - bias {layer} neuron number $\times$ 1(column vector);

$B\{2\}5 \times 1$ - bias {layer} neuron number $\times$ 1(column vector);

$B\{3\}1 \times 1$ - bias {layer} neuron number $\times$ 1(column vector).

Setting values for $x_1 \in [-17,10]$, $x_2 \in [-10,10]$ and $\lambda_2 = 0;\ 0{,}5;\ 1$ the control function u was determined according to corresponding optimal switching curve, fig.2. Learning data of 2231 vectors representing system state (x_1, x_2) and adequate control u were generated. In the neighborhood of the origin and on the decision curve the density of learning data was greater than in the remaining region.

In order to obtain the most appropriate topology, a research was conducted out concerning the number of neurons, training algorithm and the step function for the neurons layers. The success of the training was determined by the average mean square error

$$E = \sum_{i=1}^{N} \left[u - g(x_1, x_2, \lambda_2)\right]^2 \rightarrow \min , \tag{7}$$

where u is adequate control determined from fig.2,

$g(x_1, x_2, \lambda_2)$ - control function generated by ANN.

The optimal number of neurons in hidden layers is 10 and 5 neurons with tan-sigmoid (tansig) activation function and 1 neuron with tan-sigmoid activation function in the output layer.

The training of the neural network was done by back-propagation algorithm. The software used for training of the ANN has 11 versions of the back-propagation algorithm. Only the Levenberg-Marquardt algorithm (trainlm) produced acceptable approximation. After every data set training, ANN weights were adjusted. The training was done for 500 epochs. The mean square error was $E = 3{,}27.10^{-16}$.

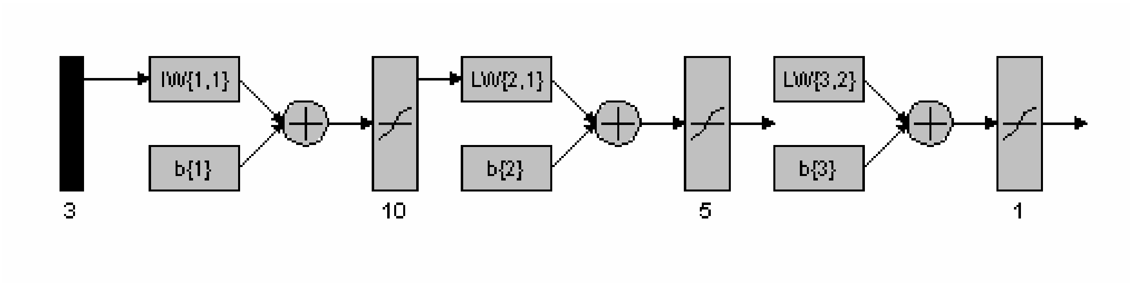

Fig.3. Matlab realization of ANN structure

4. SIMULATION RESUTLS

Simulation of the feedback system was executed in MATLAB/SIMULINK. The reference parameters were $\lambda_0 = -10, \lambda_1 = 0, \lambda_2 = \text{var}$ and. The current system state was given to the input layer of the neural network and on the net output we obtain optimal control, fig.3.

The results of experimental simulation in a form of the phase trajectories for the different initial states and different λ_2 are demonstrated in figures 4, 5, 6, 7. In the fig.4 we have trajectory with two switching operations generated by $u = \{-1, +1, -1\}$ and in the fig.5 the corresponding process in the time area is shown. In the fig.6 is shown the trajectory with one switching operation generated by $u = \{+1, -1\}$ and in the fig.7 is shown the corresponding process in the time area.

In Table 1 are shown the times of target reaching in the time optimal system – System1, in the near-time optimal system with linear approximation of the optimal switching line – System 2, in the near-time-optimal system with fuzzy approximation of the OQSL – System 3 and in the near-time-optimal system with neural approximation of the OQSL – System 4.

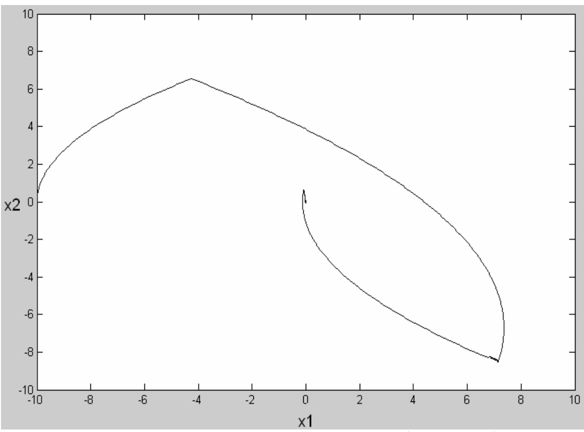

Fig. 4. Trajectory for initial state $(-10, 0)$ and

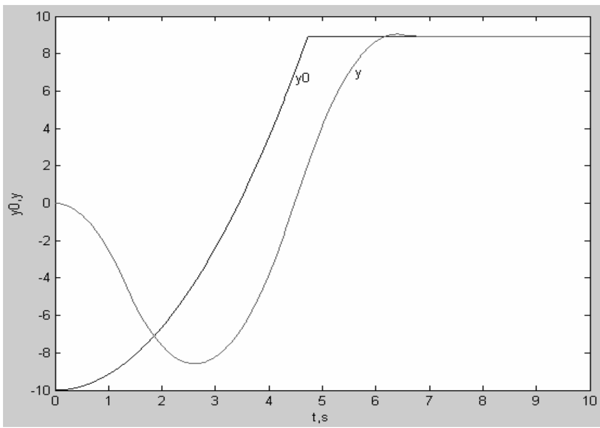

Fig.5. Process in the time area

$\lambda_2 = 0{,}92$

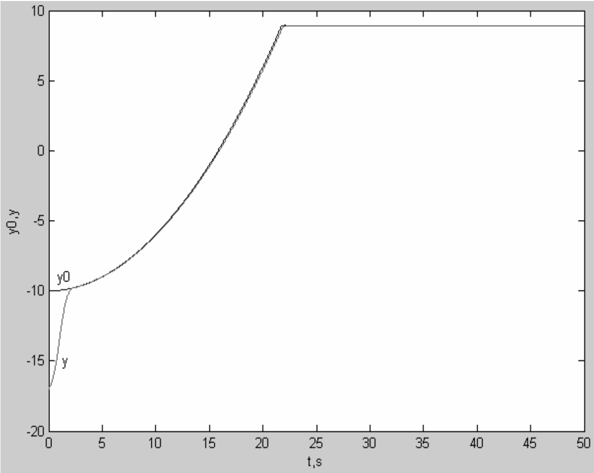

Fig. 6. Trajectory for initial state $(7, -1)$ and $\lambda_2 = 0{,}2$

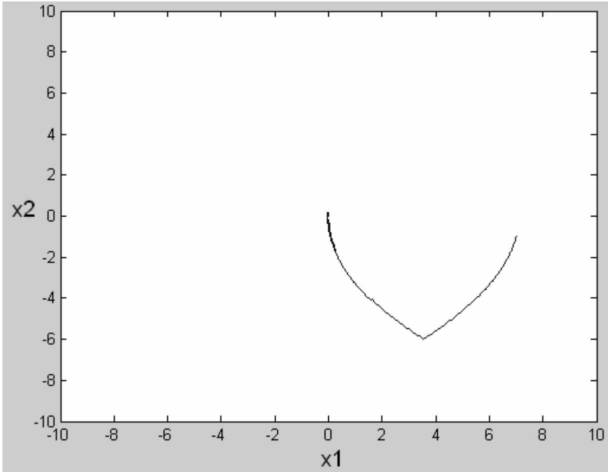

Fig.7. Process in the time area

Table 1. Motion time on phase trajectory

Initial state (x_1, x_2), λ_2	System 1, T_{opt}, s	System 2, T_2, s	δ_2	System 3, T_3, s	δ_3	System 4, T_4, s	δ_4
$(-10, 0)$, $\lambda_2 = 0{,}92$	6	6,5	8,33	6,15	2,5	6	0
$(7, -1)$, $\lambda_2 = 0{,}2$	21,68	22,64	4,43	22,5	3,78	21,69	0,05
$(-10, 0)$, $\lambda_2 = 0$	2,33	3	28,76	2,5	7,3	2,36	1,28
$(7, -1)$, $\lambda_2 = 0{,}5$	9	9,9	10	12,5	38,88	9,04	0,44

For the systems 2, 3 and 4 the relative variation of time for motion on the phase trajectory are determined as follows

$$\delta_i = \frac{\left|T_{opt} - T_i\right|}{T_{opt}}.100\% . \tag{8}$$

5. CONCLUSIONS

The idea of the neural network applying for the near-time-optimal control of quazistationary system is quite suitable. The system synthesized in such a way ensures the properties of the control process very close to time-optimal process under reference changing.

Results obtained from numerical simulation show that the transfer time in the near-time-optimal system with neural approximation of the switching line is commensurable with transfer time in the time-optimal control system and better than transfer time in the systems with linear and fuzzy approximation of the optimal switching line.

The presented neural network is characterized by facility of learning and good generalization capability.

REFERENCES

[1]. Diakonov V. P., Abramenkova I. V., Kruglov V.V. *MATLAB 5.3.1.* Moscow, 2001.

[2]. Gunchev L. A. *Optimal control.* “Technics”, Sofia, 1987.
[3] Hejmo Wl., Majewska K. *On an Almost Time-Optimal Neural Control of a Non-linear and Discontinuous Object.* Proceeding of the 14th International Conference on Systems Science, Vol. I, Wroclaw, Poland, 2001, p.361-368.
[4]. Tsankova D. *Neural Near-Time-Optimal Control for Second Order Non-linear Oscillators.* Automatics and information, 2, 1997, p.20-23.

Valentin Stoyanov *– Associate Professor, Dr., Faculty of Electrical Engineering, University of Rousse, 8 Studentska Str., 7017 Rousse, BULGARIA. e-mail: vstojanov@ru.acad.bg*

Nikolay Mihailov *– Associate Professor, Dr., Faculty of Electrical Engineering, University of Rousse, 8 Studentska Str., 7017 Rousse, BULGARIA. e-mail: mihailov@ru.acad.bg*