

Learning Algorithms for Servomechanism Time Suboptimal Control

M. Alexik

Department of Technical Cybernetics,

University of Zilina, Univerzitna 8215/1, 010 26 Zilina, Slovakia

mikulas.alexik@fri.uniza.sk

ABSTRACT

This paper describes three strategies for realisation of time sub optimal learning algorithm applied for position servomechanism control. This servomechanism was realised in laboratory and its control was realised in real time. The necessity of learning algorithm usage results from demand of time sub optimal control of position servomechanism even its loads is changed in large range. Instantaneous value of moment of inertia is not known, so it is not possible to use deterministic time optimal control with switching curved line. Author derived three different learning strategies for “recovery” time sub optimal trajectory. The effectiveness (algorithm learning time) is different for every strategy. Strategies of time sub optimal switching curved line finding are based on sliding mode control. It is combined with: 1.) progressive search of suitable slope of switching line, 2) real time continuous identification of servo mechanism parameters and computing of switching curved line, 3) off line computing of servo mechanism inverse neurons model with switching curved line computing followed by real time classification with time suboptimal control.

KEY WORDS

Sliding mode, hardware in loop simulation, neural nets.

1. Introduction

The learning control systems have an advantage against the systems with classical control algorithms in case when the inner or outer control conditions change. Classical strategy can utilize learned information from previous control processes or situations, which they have stored in memory and after the successful situation recognition they can acquire the optimal results in a shorter time. Better advance for learning can be successfully applied also on the sliding mode control with the help of the artificial neural networks. The learning system should work with a memory, which stores the previous adaptation results. In the learning process, following the adaptation results, the system will choose the best one. Then the aim is to minimize the loss:

$$Q(x, \Omega, \omega_r) = \min_{\omega} Q(x, \Omega, \omega) \quad (1)$$

where x is a system expression (state), Ω is a teacher information and ω is a control rule. The general loss of

optimisation criteria, in the learning system after the learning process, is always less than in the adaptive system. The best strategy, from author point of view, is combination of continuous identification of servomechanism parameters with switching curved line computation in real time from neuro nets model.

The time needed for the system learning is specified by speed of solving the equation (1) and markedly depends on the amount of priory information about the controlled system. The advantage of the learning system against the optimal controllers is that its design does not require the whole priory information about the environment or controlled system.

The paper is organized as follows. Section 2 describes the problem of the optimal control, section 3 describes sliding mode control, section 4 describes learning controller and section 5 describes real time simulation experiments and practical results. The paper ends with conclusion and outlook in section 6.

2. Time Optimal Control Problem.

The tasks of t-optimal control belonged among the first problems, which were solved in the theory of the automatic control and the system optimization. Only the formulation of minimum principle allowed the common view on questions of the time-optimal control of the linear systems with the limitation of controlled variable. The properties of the optimal trajectories are often used in non-linear systems, in time-suboptimal servomechanism of the robots and in the adaptive and learning algorithms. The learning controller is designed for the laboratory carriage model, powered by DC - motor. The aim is to find t-optimal control of its position. The picture of the model is on the Figure 1. The transfer function of this system can be reduced to the form

$$S(p) = \frac{K}{s(Ts + 1)} \quad (2)$$

where K is gain and T is time constant of carriage model. The laboratory carriage model can be loaded with 0 to 6 different weights (1 weight = 0.6 kg). Then transfer function (2) has 7 different gains K and time constant T , which depends also from friction. Real friction is non linear model (2) suppose linear coulomb friction.

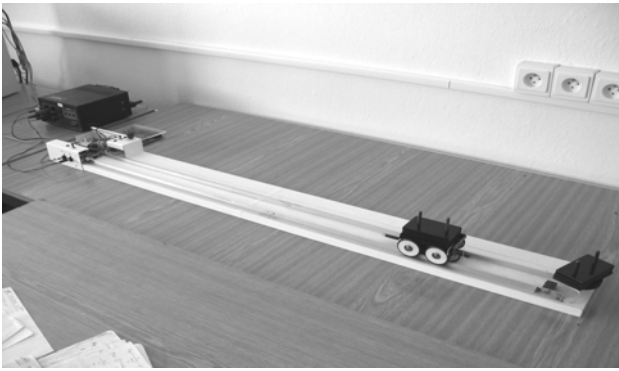


Fig. 1 The laboratory carriage model

Carriage model is a system with the 2nd-level delay for which it is possible to derive time optimal control by control loop on Fig. 4 and responses on Fig.2. Trajectory under the time axis represents the control process in the phase space $[x_1, x_2] = [e(t), e'(t)]$. There is one switch-point on the phase trajectory.

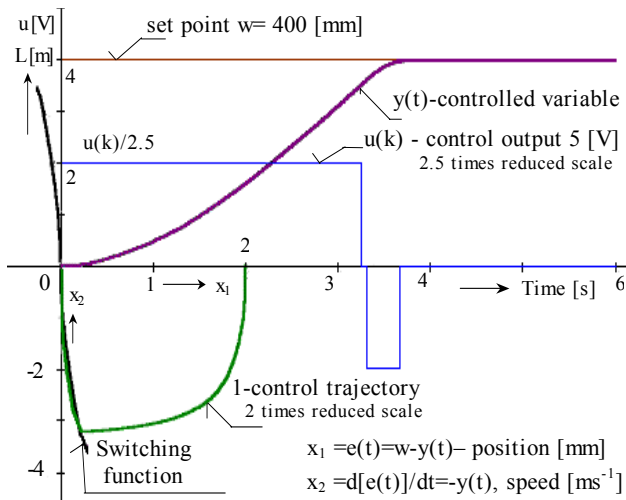


Fig. 2. Response and state space trajectory of the second-order time-optimal controlled system.

If controller knows “K” and “T” exactly although weights on carriage are changing, than it can compute switching function (4), so the control process becomes t-optimal.

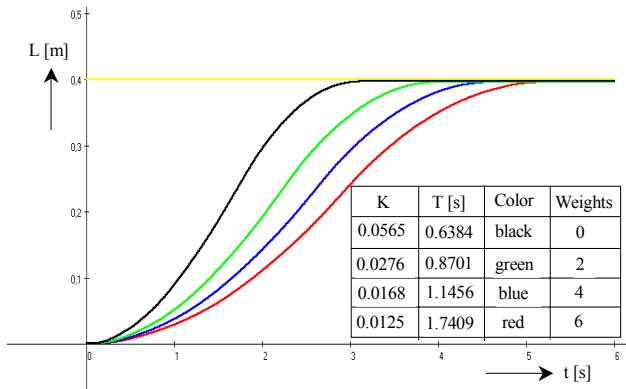


Fig. 3 Responses and parameters for some weights

3. Sliding Mode Control.

The sliding mode control (SMC) is very popular and commonly used. The advantage is its really simple design, invariance and robustness. The relay control (bang-bang) belongs to the first applications [2], [4], when the actual signal is bounded. Therefore the t-optimal control acquires only minimal or maximal actual values. The SMC controller is very simple. The actual value is appointed according to the place in the phase state. The phase state is divided by the switching surface $s(x)$:

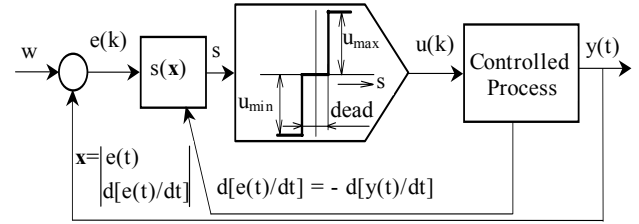


Fig. 4 Time optimal control loop block scheme.

The most frequent form of the switching surface is linear function [3],[5]:

$$s(\mathbf{x}) = C\mathbf{x} = 0 \quad (3)$$

Usually this switching function does not satisfy the t-optimal control requirement and the control can reach the sliding mode. The proper t-optimal switching function is non-linear. The first aim of learning algorithm is to find this switching function. Switching function is derived from step response (4) of controlled process (2) and is described by (5). Then controlled variable from Fig. 4 is described by (6).

$$y(t) = K[t - T + T \exp(-t/T)]U_i \quad (4)$$

$$s(\mathbf{x}) = U_i K T \ln\left(\frac{1 - x_2(t)}{K U_i}\right) + T x_2(t) + x_1(t)$$

$$\text{if } x_2 < 0 \text{ then } U_i = u_{\max} \quad (5)$$

$$\text{if } x_2 \geq 0 \text{ then } U_i = u_{\min}$$

$$x_1(t) = e(t) = w(t) - y(t); \quad x_2(t) = d[e(t)/dt]$$

$$\text{if } (s \geq \text{dead}) \text{ then } u(k) = u_{\max} \quad (6)$$

$$\text{if } (\text{abs}(s) < \text{dead}) \text{ then } u(k) = 0$$

$$\text{if } (s < -\text{dead}) \text{ then } u(k) = u_{\min}$$

Because weights on carriage can be changed we don't know instantaneous value for time constant T and gain K of controlled process, so we cannot compute switching function and realize t-optimal control. Learning controller in three ways can solve this problem. Real time measurement of carriage position in every sampling interval (5 [ms] to 20 [ms]) and their filtration is very important step in all control strategy but it is not described in detail. Next section describes three ways for realisation of learning controller based on t-optimal control.

4. Learning Controller.

Why do we need learning controller? When the carriage-loads are changed this means that parameters of controlled process are changed. There are six various carriage-loads and therefore it is a system with seven different parameter couples. So, when the controller is set for one system and the switching function is found, the function is saved in memory of learning controller for case of a repeated regulation of this system. The learning controller could control the system t-optimally even if system parameters would change. The fundamentals for learning algorithm formulation were published in [7]. Some outputs can be seen on Fig. 6 to Fig. 9 and in section with simulation experiments.

The strategies of time sub optimal switching function finding are based on sliding mode control. It is combined with: 1.) progressive search of suitable slope of switching line, 2) real time continuous identification of servo mechanism parameters and computing of switching curved line, 3) off line computing of servo mechanism inverse neurons model with switching curved line computing, than real time classification with time suboptimal control. The idea of learning controller for these strategies is common and is illustrated on Fig. 5.

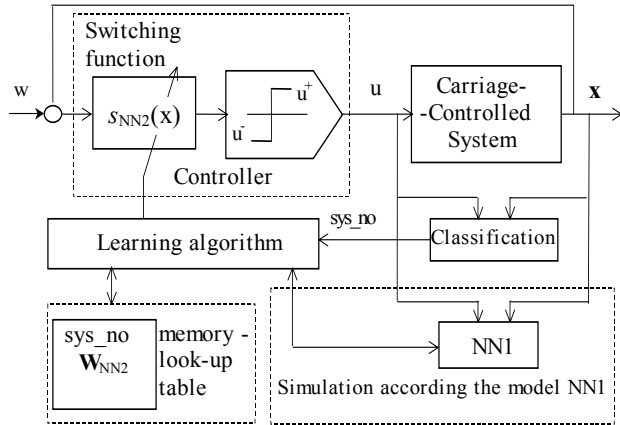


Fig. 5 Block scheme of the learning controller

Blocks in this figure are as follows:

The block of classification is responsible for system detection. It generates number of the system, according to the system parameters. Because of ability to use the saved results for correct system, it is necessary for the controller to classify the current system. The classifications, which are used in this paper, are based on the parameter identification or on ART network [1].

The block of controller is responsible for actual value. In this block we firstly describe strategy: progressive search of suitable slope of switching line.

During progressive generation of switching line slope C_p , by adaptive sliding mode algorithm, the points from switching curve for several value of set point are saved in memory. With such proceeding, more points for switching curve can be found and the parameters of switching curve

function can be calculated or interpolated. The optimal step response for selected set points as well as points from switching curve is selected from all generated step responses according to response with minimal settling time without overshoot. On the Fig. 6 and Fig. 7 is illustrated process for switching curved line points searching. As it can be seen it is needed 5 to 10 step responses for finding points from switching curved line for one pair of parameters $[K, T]$ of model (2). So, this learning strategy cannot be realized in real time, but it is first step for problem solving.

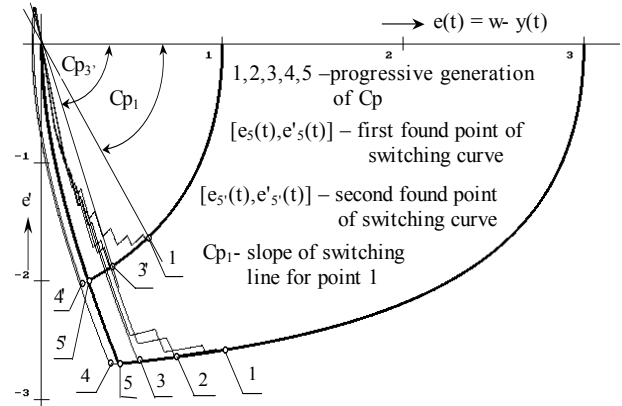


Fig. 6 State space trajectories during learning process by searching of points from switching curved line

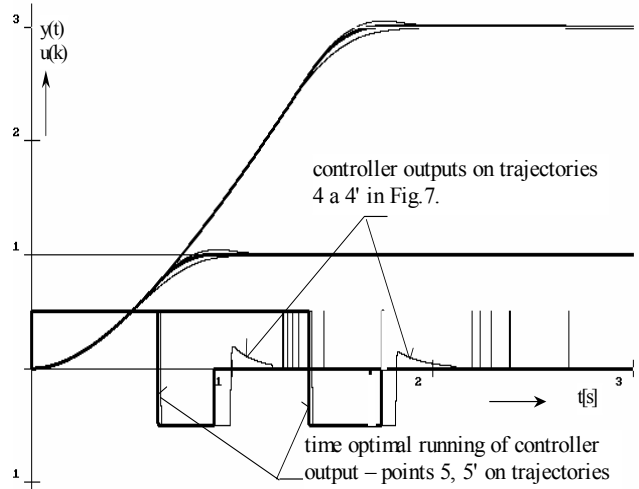


Fig. 7 Step responses during learning process by searching of points from switching curved line.

For control strategy with identification of controlled parameters and follow-up computation of switching curved line (4) in real time a new way for parameters computation from step response is needed (6) of controlled process (2). The on line continuous identification cannot be used. Parameters K , and T have to be computed before instant of time when controlled variable begins switching. From step response (4) can be derived dotted parameters estimation of transfer function (2) in the form (7).

$$K = [x_2(t/2)]^2 / \{U_{\max} [2x_2(t/2) - x_2(t)]\} \quad (7)$$

$$T = -t / \{\ln[1 - x_2(t)/K]\}$$

The weak point of calculation is that we need to know immediate derivative values of controlled variable. As can be seen from Fig. 8 derivation values is change only in 9-15 levels (sensor with 1 increment on 0.28 mm was used). The loop responses on Fig. 8 have assumed that controlled variable parameters and also switching curved line are known therefore identification was not necessary.

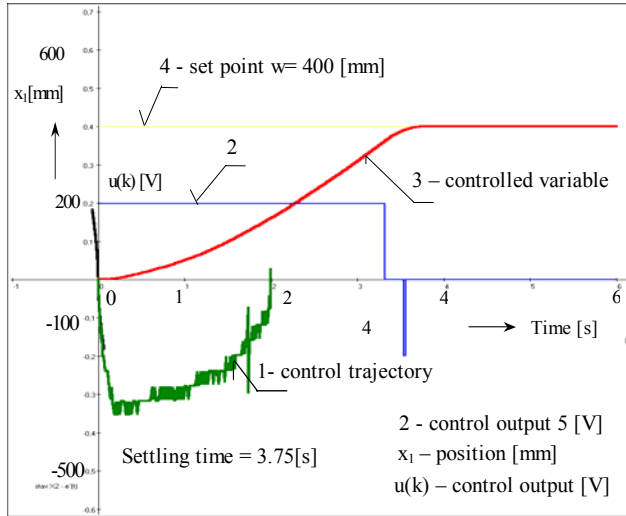


Fig. 8 Loop response, state trajectory and controller output of time suboptimal control for carriage system.

Although filtered signal (100 [ms] filtered time constant) was used on Fig 9, derivative values were still too corrupt with noise, and then derivative values of controlled variable has not be computed precisely, so settling time was not time optimal and controlled variable also were switching only to one polarity.

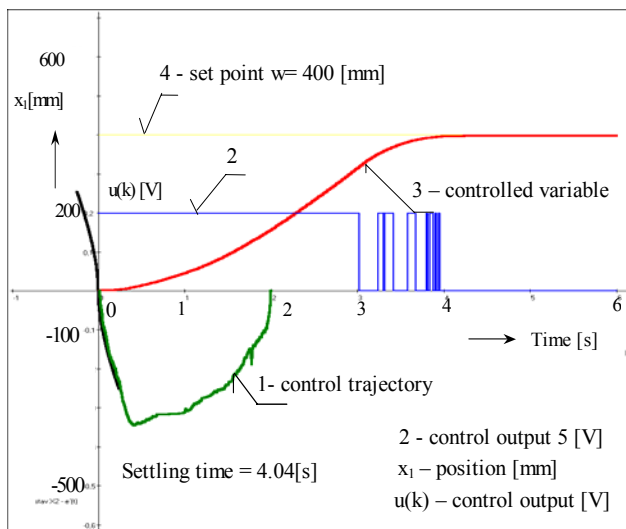


Fig. 9 Loop response, state trajectory and controller output for time suboptimal control.

In this strategy real time identification is used for classification and also for t- suboptimal control and it is nod needed special block for simulation and learning controller.

The third strategy it uses the feed-forward neural net NN2 [5] for approximation of the switching function. At the beginning the NN2 approximates the linear switching function. The NN2 is trained according to the simulated phase points. Later the NN2 is adjusted to approximate the non-linear t-optimal switching function. The switching function with NN2 can be:

$$s_{NN2}(\mathbf{x}) = x_n(k) - f_{NN2}(x_1(k), x_2(k), \dots, x_{n-1}(k)) = 0 \quad (8)$$

where n is the system order. The complexity of the net NN2 depends on controlled system order. In case of the second-order system (2) the NN2 will have one input and one output. It should have at least 10 neurons with the non-linear activation functions.

When system parameter are changing, the learning algorithm sets the NN2 (its weight's matrix W_{NN2}) according to the classified system (sys_no).

The block of simulation contains the discrete linear neural model (feed-forward NN1) of system in the form:

$$y(k) = f_{NN1}(y(k+1), y(k+2), \dots, y(k+n), u(k), u(k+1), \dots, u(k+n-1)) \quad (9)$$

The number of NN1 inputs depends on the system order. In case of the second-order system (2), the NN1 will have at least 6 inputs and one output. Therefore the system is linear and the NN1 should have a couple of linear neurons. This model is used for a simulation. The simulation generates the points of the t-optimal phase trajectory. According to these points the neural net NN2 is trained. After that, the NN2 approximates the t-optimal switching function.

The block of learning algorithm is responsible for task cooperation, memory management, neural nets training and simulation.

Both nets (NN1, NN2) are trained with Levenberg-Marquardt method [5]. This method is faster then the common back-propagation.

The learning controller, described above, is very effective, because it is able to find t-optimal control in two learning steps for every single system parameter change. In the first step of the learning, the control process goes according to the a priori defined switching function in the NN1. In the second step of the learning, the control process is t-optimal.

5. Simulation Experiments.

The best results have been achieved with combination of real time measurement of controlled variable and exact derivation values computation from state estimator. In the state estimator model of controlled process with 2 weights were assumed in all situations Block scheme of t-optimal control loop with state estimator can be seen in Fig. 10. All responses for control loop with state estimator can be seen on Fig. 11.

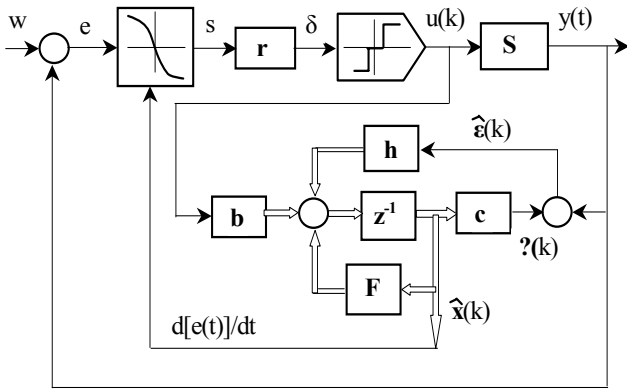


Fig.10 Block scheme of t-optimal control loop with state estimator.

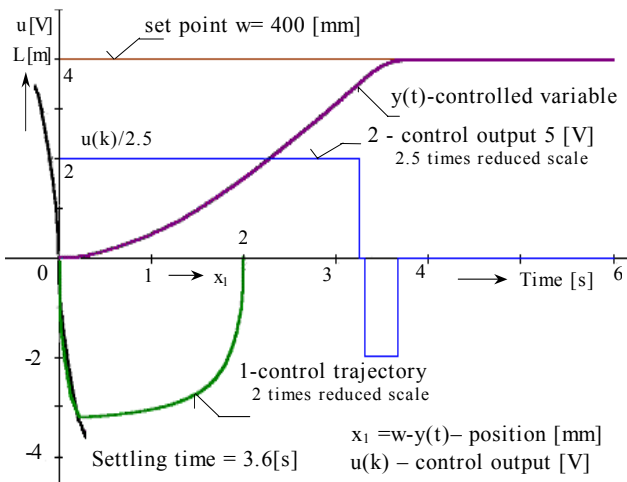


Fig.11 Loop response, state trajectory and controller output for control loop with state estimator.

Next simulation experiments are for control loop with neuron nets. The simulation in the phase space can be used to find the switching function for the t-optimal control, because the part of the t-optimal phase trajectory is coincident with the switching function. The simulation runs on the system model (NN1). The switching function passes over the phase points, which are the simulation results. The approximation of the switching function with the NN2 can be improved, if the number of simulated phase points is higher. The t-optimal control has a special property for the actual value. If the control has to be optimal, the actual signal has to take only extreme values.

The main task of the simulation is to simulate an inverse t-optimal control process. This process begins in the desired state (the system output and the desired value are identical) and then the maximal or minimal actual signal starts to switch. The phase points of this simulated process are saved for the NN2 training. The process of simulation for the second-order system in the phase space is shown in Figure 13. After successful simulation, there are two curves of phase points, which are used to approximate the t-optimal switching function via the neural network NN2.

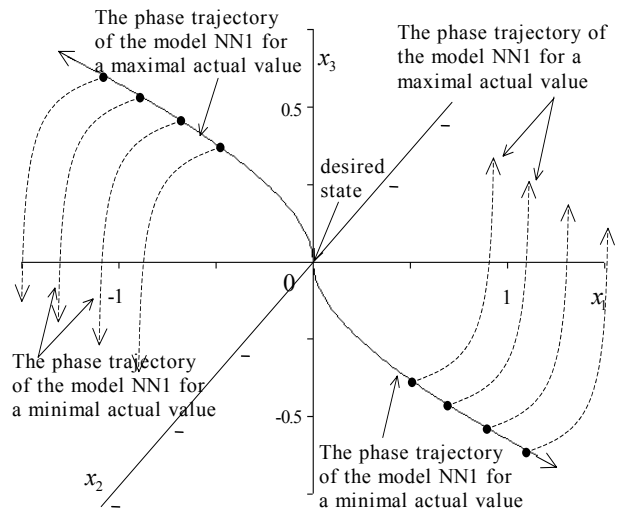


Fig. 12 The simulation in the phase space for II.-order system

Evidently the control time in second step is shorter than the time in first learning step. The desired control values were not identical. As the Figure 14 shows, the controller will work even with a noise in the measured signal of the carriage position.

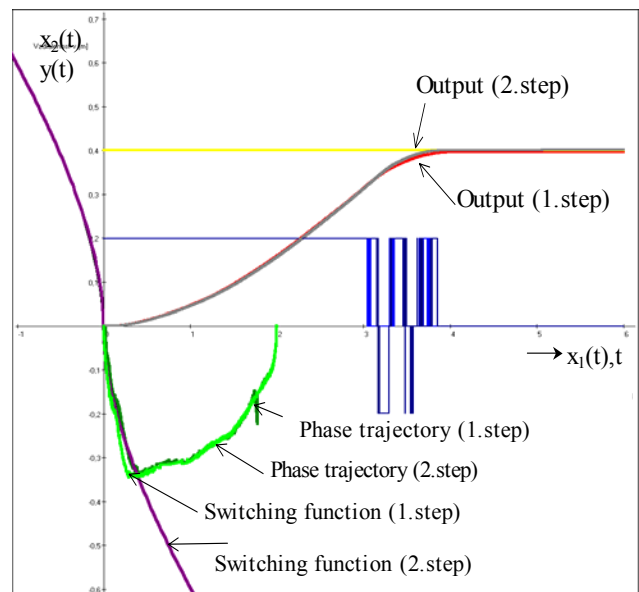


Fig. 13 Real time simulation experiment for time suboptimal control loop with neuro net model.

In case of higher-order system, the simulation will be more complicated. For instance, the 3-order system has to be simulated in a 3D phase space and the final t-optimal switching function can be represented as a 3D surface. The simulation consists of two actual combinations:
 maximal => minimal actual value
 minimal => maximal actual value
 All simulated phase points describe the shape of 3D switching surface. The simulation of the higher-order systems takes a lot of time, because the amount of

simulated points increases with the phase space dimension rapidly. Choosing the rational precision of the approximation or disregarding the insignificant system orders can solve that problem. For simulation experiment (discrete simulation only) with third order system, which will be shown during presentation, was realized 5000 points of phase trajectories. Neural nets NS2 were created from 3 layers and 13 neurons (6 input layers, 6 hidden layers 1 output layer).

The last simulation experiments is comparison of two learning strategies, which can be seen on figure 14. There are the output signals, phase trajectories and switching functions for both strategies: First is computation of switching function during real time control from identification and second is computation of switching function after off line learned neuro nets.

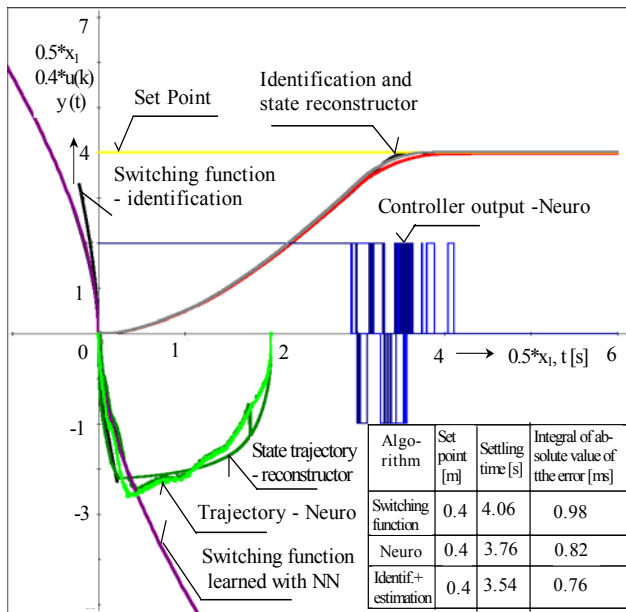


Fig. 14 Comparison real time simulation experiments

6. Conclusions

In many practical applications especially in servomechanism, the t-optimal control problem is usually solved for desired system and then applied with the specific control rules. For class of second order systems with single input and single output sliding mode control are used. If the system is not stationary or there is a possibility of the system parameters change, the classical sliding mode control cannot be used and the learning controller based on sliding mode control could then assign the optimal control requirement. The paper describes three learning algorithms. The first is based on classical sliding mode control, the second on sliding mode control combined with the neural networks and the third is based on continuous computation of controlled process parameters and follow-up real time computation of switching curved line in every sampling interval,

combined with state estimator. The first algorithm is the clearer one, but learns very slowly, because we have to measure 5 to 9 loop responses for one computation of switching function. The second algorithm, which uses neural networks, learns more quickly and to understand it fully it is crucial to know how the first one works. Both learning algorithms described in the paper set the t-optimal switching surface for second order-controlled system. The combined algorithm with NN can do so even for third order-controlled system. The third algorithm is the best one from effectiveness point of view but cannot be used for problems where switching curved line is not known as a function.

The only a priori condition is the existence of the initial stable control, for example the sliding mode control based on switching curve and switching line. Described algorithms were tested on laboratory equipment by real time simulation experiment.

References

- [1] Carpenter, G., Grossberg, S.: ART: Stable Self-organization of Neural Recognition Codes in Response to Arbitrary Lists of Input Patterns, *8th Annual Conference of the Cognitive Science Society*, 45-62, 1986.
- [2] Cypkin, J. Z.: *Teorija relejnych sistem avtomatičeskogo regulirovanija*, Gostechizdat, Moskva, 1955.
- [3] Ferrara, A., Giacomini, L.: First and Second Order Sliding Mode Control for a Class of Single-input Nonlinear Systems with Nonmatched Uncertainties, *IMA Journal of Mathematic Control and Information*, 18, 253-268, 2001.
- [4] Liu, T. S., Lee S. W.: A Repetitive Learning Method Based on Sliding Mode for Robot Control, *Journal of Dynamic Systems, Measurement and Control*, ASME, 2000.
- [5] Efe, M. Ö., Kaynak, O., Wilamowski, B. M., Yu, X.: A Robust On-line Learning Algorithm for Intelligent Control Systems, *International Journal of Adaptive Control and Signal Processing*, 17, 489-500, 2003.
- [6] Alexík M., Vittek J.: Adaptive Sliding Mode Control of Position Servosystem. *Preprints of 1st IFAC Workshop on: New Trends in Design of Control Systems*. 7.-9. 9. 1994, Smolenice. pp. 278 - 283.
- [7] Alexík M.: Learning Algorithm for II. Order Systems and Analytic PID Algorithm for III. Order Systems. (in Slovak). In: *Proceedings of the 4th International Scientific-Technical Conference Process Control 2000*, June 11-14, 2000, Kouty nad Desnou, Czech Republic, pp. RIP213. ISBN 80-7194-271-5.