INŽENÝRSKÁ MECHANIKA 2005
NÁRODNÍ KONFERENCE
s mezinárodní účastí
Svratka, Česká republika, 9. - 12. května 2005

SOME NOTES ON CONTROL OF ASYNCHRONOUS ELECTROMOTOR BY IMPROVED CARLA METHOD

M. Turek[1], T. Březina[2], J. Pulchart[3]

Summary: *Modifications of reinforcement learning algorithm, so called continuous action reinforcement learning automaton (CARLA), are presented in this contribution. Automaton learning algorithm is based on rewarding that gradually evolves the set of probability densities. This set is consequently used for action set determination. Modifications consist of improving learning parameters based on learned values. Thereby higher values of probability density near the best action are reached and therefore the variance of chosen actions is lower than original. The influence of modifications is proved by simulation study describing learning and behavior of asynchronous electromotor scalar control. Standard PSD controller is used whose parameter values represent actions of three independent automata. The goal of on line learning process is to minimize the mean square of control error. Here described modifications of algorithm allow the improvement of quality of revolutions control with preserving basic algorithm characteristics.*

1 Introduction

Despite the progress in development of control systems, the general problem of setting their parameters still remains unsolved. It is possible to calculate the parameters easily, when mathematical model of controlled system is known. In the other cases the analogy with similar system or experts practice are used. When this approach fails the appropriate method of artificial intelligence can be used. One of these methods is CARLA (e. g. [3]). Its function was successfully proved by practical applications (e. g. [2]) and results prove assumption that better results can be reached by CARLA method.

The appropriate learning parameters [1] improve the behavior of CARLA method. The problem is that different characteristics are important in different phases of learning. First the high speed of learning, later the precision of learned value is needed. These two demands contradict each other when CARLA method is used. But both of them can be achieved by its modification.

[1] Milan Turek, Ing., VUT v Brně, FSI ÚAI, Technická 2, 61669 Brno, ČR
e-mail: yturek00@stud.fme.vutbr.cz
[2] Tomáš Březina, RNDr., Ing., CSc., VUT v Brně, FSI ÚAI, Technická 2, 61669 Brno, ČR
e-mail: brezina@fme.vutbr.cz
[3] Jaroslav Pulchart, Ing., VUT v Brně, FSI ÚAI, Technická 2, 61669 Brno, ČR
e-mail: pulchart@kn.vutbr.cz

2 Approach

CARLA method [1, 2, 3] consists of six steps:

- random selection of action,
- action application to the system,
- time delay,
- calculation of action cost,
- calculation of performance and
- modification of probability density.

Actions are selected randomly from continuous interval. Selection is based on learned probability density. The action applied is to system after selection. It can mean setting the value of parameter. Following time delay should be as short as possible, but must be long enough so that influence of action to system can be seen.

Next the cost of action which represents as quality criteria is calculated. CARLA method minimizes the cost. The performance is calculated based on the cost. It is value expressing improvement reached by applied action. Finally the probability density is updated so that successful action and actions from its neighborhood are selected more often. This step represents learning method.

It seems that big improvement of learning speed can be achieved by decreasing time delay. But practical improvement is minimal. Minimum of time delay is function of system. Because this, it can’t be easily decreased. Much bigger improvement of learning speed can be done by changing the way of modifying probability density.

Probability density in iteration k is described by function *f(x,k)*, which to every action from continuous interval $<x_{min};x_{max}>$ assigns probability of its selection. Probability of actions selection which don’t fall to this interval equals to zero. The way of modifying probability density by CARLA method is following:

$$f(x,k+1)=\begin{cases}\alpha(k)\cdot\left[f(x,k)+\beta(k)\cdot H(x(k))\right], & pokud\ x\in\langle x_{\min};x_{\max}\rangle\\ 0, & jinak\end{cases} \qquad (1)$$

$\alpha(k)$ normalizes probability density, i.e. guarantees compliance of condition $\int f(x,k+1)\,dx=1$, $\beta(k)$ is reinforcement and $H(x(k))$ is Gaussian function centered at last selected action $x(k)$.

$$H(x(k))=\lambda\cdot\exp\left(-\frac{(x-x(k))^2}{\sigma^2}\right) \qquad (2)$$

$$\lambda=\frac{g_h}{x_{\max}-x_{\min}} \qquad (3)$$

$$\sigma^2=2\cdot\left(g_w\cdot(x_{\max}-x_{\min})\right)^2 \qquad (4)$$

Coefficients g_h and g_w affect learning speed and precision of learned value [1]. By their change during learning it is possible to achieve first increased speed of learning and then bigger precision of learned value.

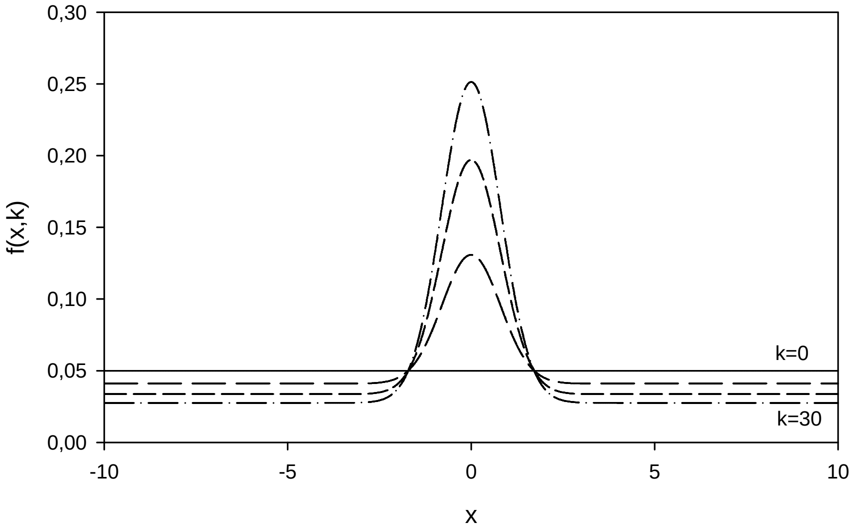

Figure 1: Ideal learning progress

3 Implementation

As was said above high learning speed is desired. So the initial values of coefficients g_h and g_w are chosen to maximize learning speed. When optimal action is approximately found, their values are modified to increase the precision. As can be seen on figure 1, the value of probability density function is increasing in neighborhood of optimal action during learning progress. The values are decreasing to zero in the rest of interval. So it is possible to use the size of interval $<x^{\kappa}_{min};x^{\kappa}_{max}>$, where value or probability density is not near zero, as measure of learning stage. The interval is defined as follows:

$$x^{\kappa}_{\min} = \min\left\{ x;\ f\left(x,k\right) \geq \frac{\kappa}{x_{\max} - x_{\min}} \right\} \tag{5}$$

$$x^{\kappa}_{\max} = \max\left\{ x;\ f\left(x,k\right) \geq \frac{\kappa}{x_{\max} - x_{\min}} \right\} \tag{6}$$

Value of κ constant was determined by experiments. Its recommended value is 10^{-4}.

Because values of coefficients g_h and g_w should be proportional to size of interval $<x^{\kappa}_{min};x^{\kappa}_{max}>$ it is possible instead of changing their values equivalent method and change equations (3) and (4) by the following way:

$$\lambda = \frac{g_h}{x^{\kappa}_{\max} - x^{\kappa}_{\min}} \tag{7}$$

$$\sigma^2 = 2 \cdot \left(g_w \cdot \left(x^{\kappa}_{\max} - x^{\kappa}_{\min} \right) \right)^2 \tag{8}$$

This modification theoretically leads to zero variance of selected actions. But in practice the limitation is induced by limited memory size.

It is not possible to save probability density function *f(x,k)* exactly for whole interval $<x_{min};x_{max}>$ because infinite memory would by needed, so limited number of samples is used. Values between samples are computed by (linear) interpolation.

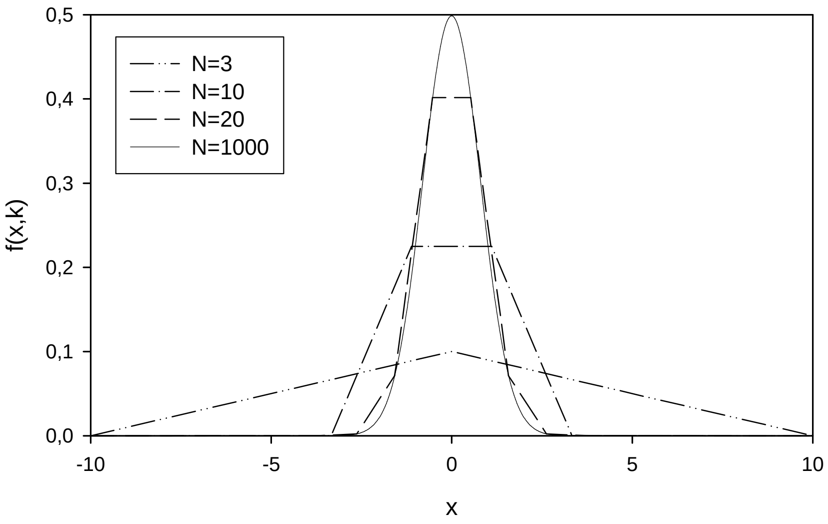

Figure 2: Influence of number of samples (N) to probability density function ($g_h, g_w = konst$)

Minimal variance of selected actions is proportional to the distance between two adjacent samples used to save the probability density function. Decreasing the distance decreases the minimal reachable variance.

Decreasing the distance of two adjacent samples can be achieved by three different ways:

- increasing number of samples
- decreasing size of interval $<x_{min};x_{max}>$
- change of position of samples

Too high number of samples is not suitable because it leads not only to increasing of memory requirements, but as well to increasing of computing time. Decreasing size of interval $<x_{min};x_{max}>$ increases probability so that optimal action needs not lie in it. So the best way for decreasing the distance of two adjacent samples is the change of their position. One way of their positioning looks as follows.

Probability of selecting actions from big part of interval $<x_{min};x_{max}>$ equals approximately to zero after learning. So in this part of interval samples are not needed. Two samples are left on borders of interval and the rest of samples is arranged in appropriate way at the part of interval, where probability of action selection is not near to zero. For example as follows:

$$\begin{aligned} x_1 &= x_{\min} \\ x_N &= x_{\max} \\ x_i &= x_{\min}^{\kappa} + (i-2) \cdot \frac{x_{\max}^{\kappa} - x_{\min}^{\kappa}}{i-3} \quad \forall i \in \{2, \ldots, N-1\} \end{aligned} \tag{9}$$

x_i is position of i-th sample. x^{κ}_{min}, x^{κ}_{max} are defined by (5) and (6). Experimentally determined value of coefficient κ for this method of positioning is 10^{-6}.

4 Results

Function of modified CARLA method was proved by simulating control of asynchronous motor. Motor was controlled by discrete regulator and CARLA method was used to setting of regulator parameters.

The classic discrete PSD regulator was used for control:

$$u(k) = K_p \cdot e(k) + K_s \cdot T \cdot \sum_{i=1}^{k} e(i) + K_d \cdot \frac{e(k) - e(k-1)}{T} \tag{10}$$

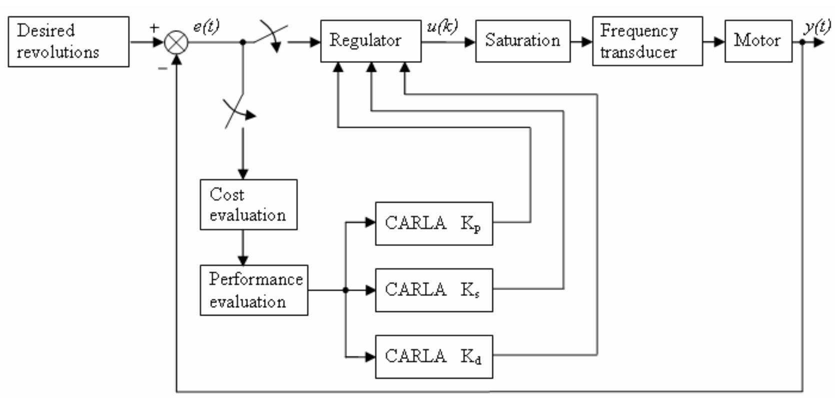

Figure 3: Schema of simulated system

Assumption that modified CARLA method can lead to better results than unmodified one, was confirmed (see fig. 4). The modification eliminated oscillations of revolutions induced by too high variance of selected values of K_s coefficient.

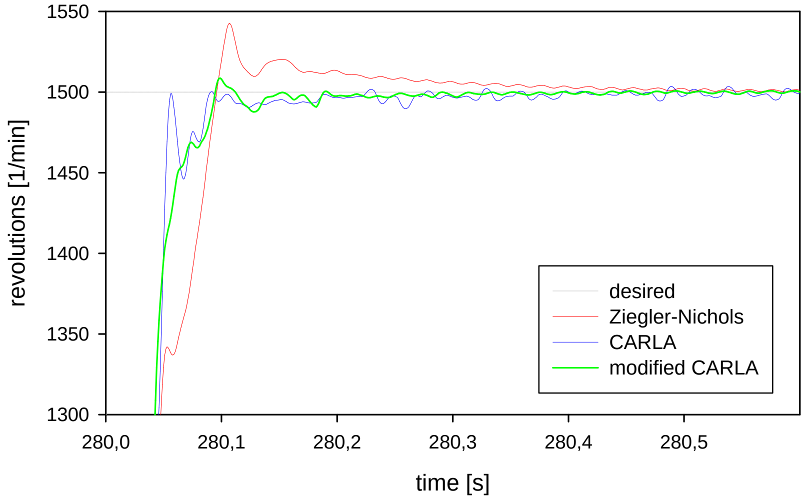

Figure 4: Behavior of simulated system

Modification decreased the variance of selected values of K_s coefficient (see table 1). This leads to improvement of control. Variance of remaining coefficients stayed unchanged, because its decreasing does not lead to control improvement.

Table 1: Variance of selected values of regulator parameters

	K_p	K_s	K_d
CARLA	3,66E-05	1,23E-05	1,61E-07
Modified CARLA	8,73E-05	1,84E-08	1,24E-07

High decrease of selected variance values for K_s parameter with simultaneous negligible variance change of selected values for remaining parameters leads to conclusion, that quality of control is most dependent on summative part of regulator.

This conclusion is proved by means of selected values. Mean value of parameter K_s remained practically unchanged. Mean values of other two parameters was adapted to decreased variance of selected values of K_s parameter.

Table 2: Mean value of selected parameters

	K_p	K_s	K_d
CARLA	0,085088191	0,54974466	0,001625666
Modified CARLA	0,037049342	0,549577577	0,000998907

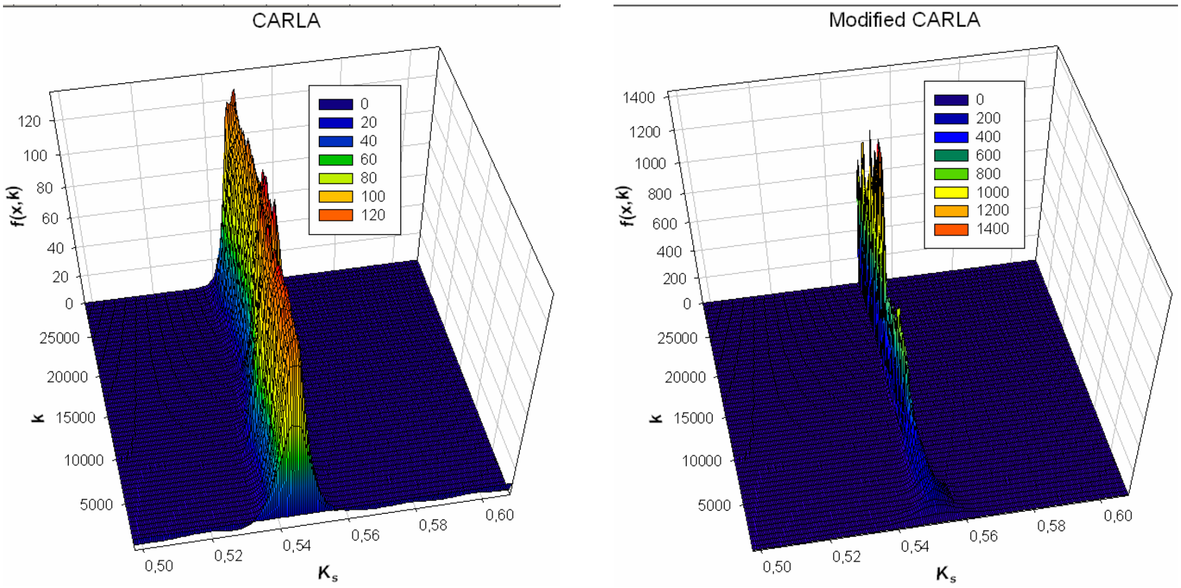

Figure 5: Progress of learning of probability density

5 Conclusion

As verified, earlier CARLA method is capable to improve behavior of controlled system. In spite of parameters learned by CARLA method should be optimal, it is still possible to improve them. The modification above is capable to learn more precise parameters without negative impact to method behavior. Also stops decreasing of variance of selected parameters when it is not leading to improvement and by this prevents overlearning. This modification is not the only possible, but it leads to improvement without negative impact to some of method attributes (speed of learning, noise resistance, …).

6 Acknowledgement

This work was done with the support of research project MSM 0021630518 "Simulation modeling of mechatronics systems".

7 Literature

[1] Březina, T., Turek, M.: Use of Continuous action reinforcement learning automata for asynchronous electromotor control. Engineering mechanics 2004, 53-54, CDROM.

[2] Howell, M. N., Best, M. C.: On-line PID tuning for engine idle-speed control using continuous action reinforcement learning automata. Control Engineering Practice 8, 2000, 147-154.

[3] Howell, M. N., Frost, G. P., Gordon, T. J.,Wu, Q. H.: Continuous action reinforcement learning applied to vehicle suspension control. Mechatronics, 7(3), 1997, 263-276.

[4] Howell, M. N., Gordon, T. J.: Continuous action reinforcement learning automata and their application to adaptive digital filter design, Engineering Applications of Artificial Intelligence, 14(5), 2001, 549-561.

[5] Najim, K., Poznak, A.S.: Learning Automata - Theory and Applications. Pergamon Press, Oxford, 1994.

[6] Narendra, K.S., Thathachar, M.A.L.: Learning Automata: An Introduction, Prentice Hall, London, 1989.

[7] Ong, Ch.M. (1998) Dynamic Simulation of Electric Machinery, Prentice Hall, New Jersey.

[8] Wu, Q. H.: Learning coordinated control of power systems using interconnected learning automata. International Journal of Electrical Power and Energy Systems, 17(2), 1995, 91-99.