USING PSEUDO-RANDOM SIGNAL FOR SPEEDING UP OF IDENTIFICATION

Petr Malounek
Doctoral Degrese Programme (1), FEEC BUT
E-mail: xmalou01@stud.feec.vutbr.cz

Supervised by: Petr Pivoňka
E-mail: pivonka@feec.vutbr.cz

ABSTRACT

This paper describes possibilities using pseudorandom signal added to desired signal to speed-up convergence of identification. Two types of identification are used, recursive least square identification and identification based on neural networks with Levenberg-Marquardt learning method. Linear Quadratic controller is used for process control.

1. INTRODUCTION

Fast convergence and good parameters approximation of the controlled process is needed for using self tuning controllers as one type of adaptive systems. Main state scheme is shown in Figure 1[5].Parameters of adaptive LQ controller are computed from model estimation. Enough excitation of controlled process is needed for good approximation. If is used the signal, e.g. ramp, the excitation of the process is insufficiently and the approximation of parameters is wrong and computed parameters of LQ controller are wrong too. The destabilization of control loop can be achieved. The small pseudo-random signal is tested in this paper for ensuring fast convergence of the model parameters. Pseudo-random signal is added to the excitation signal. ARX model is estimated by recursive least mean square (RLS) identification and identification based on neural network with Levenberg-Marquardt (LM) learning.

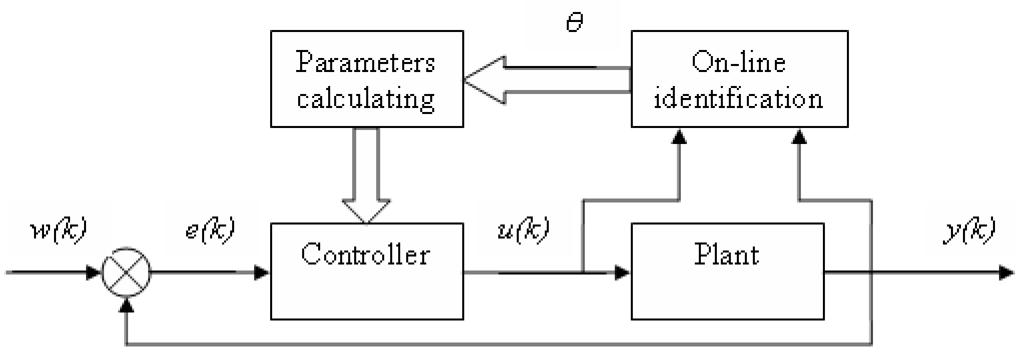

Figure 1: Main scheme of self tuning controller

2. ON-LINE IDENTIFICATION

On-line identification is one of the most important component of the self tuning controller. On-line identification is running all of time when control algorithm is calculated. Controller can adapt its parameters thanks to on-line. The parameters of controller are computed from the estimate model of the process in any steps of control algorithm. The ARX model or NARX model is used in this case. The transfer function of ARX model *n* order is

$$F(z) = \frac{b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}} \tag{1}$$

estimated parameters are

$$\Theta^T = \begin{bmatrix} a_1 & a_2 & \cdots & a_n & b_1 & b_2 & \cdots & b_n \end{bmatrix} \tag{2}$$

The vector form output of the model is

$$\hat{y}(k) = \Theta^T(k)\varphi(k) + \varepsilon \tag{3}$$

where

$$\varphi^T(k) = [-y(k-1) \quad -y(k-2) \quad \cdots \quad -y(k-n) \quad u(k-1) \quad u(k-2) \quad \cdots \quad u(k-n)] \tag{4}$$

is the vector of inputs and outputs from the process and ε is the immeasurable noise[1][3]. Estimation of the parameters is given by recursive least mean square method and Levenberg-Marquardt method. The second order of model is used.

2.1. RECURSIVE LEAST MEAN SQUARE IDENTIFICATION

This method is widely used method. It is easy to implement. The vector of estimated parameters is computed at every step by equation

$$\Theta(k) = \Theta(k-1) + K(k-1)\left[y(k) - \varphi^T(k)\Theta(k-1)\right] \tag{5}$$

Model parameters are actualized by computing prediction error

$$y(k) - \varphi^T(k)\Theta(k-1) \tag{6}$$

where *y(k)* is process output. Vector *K(k)* is weight which accelerates or slows down attribution of prediction error to estimation parameters

$$K(k) = \frac{P(k-1)\varphi(k)}{\lambda + \varphi^T(k)P(k-1)\varphi(k)} \tag{7}$$

where λ is forgetting factor and *P(k)* is a covariance matrix[3]

$$P(k) = \frac{P(k-1) - K(k)\varphi^T(k)P(k-1)}{\lambda} \tag{8}$$

2.2. IDENTIFICATION BASED ON NEURAL NETWORKS WITH LEVENBERG-MARQUARDT LEARNING METHOD

Levenberg-Marquardt method originates from Gauss-Newton method and accelerates this method in iterative parameters estimation. This method is numerical solution of minimization sum of squares generally nonlinear functions. Method is working on the principle of finding global minimum between n process inputs and model outputs. Inputs and outputs from process are accumulated to matrix

$$X(k) = [\varphi(k) \quad \varphi(k-1) \quad \cdots \quad \varphi(k-n)] \tag{9}$$

Number of rows is given by model order. Number of columns is optional parameter. It is given by number of states which are remembered from process. Iterative algorithm is given by equation

$$\Theta(i \mid k) = \Theta(i \mid k-1) - \left[J^T(i \mid k) J(i \mid k) + \lambda I\right]^{-1} J(i \mid k) E(i \mid k) \tag{10}$$

where k is step, i is iteration in step, λ is optional parameter, which influence evolution of prediction error and $E(i|k)$ is error vector

$$E(i \mid k) = T^T(k) - X^T(k)\Theta(k) \tag{11}$$

Where T is vector of last values of process outputs with n rows

$$T(k) = [y(k) \quad y(k-1) \quad \cdots \quad y(k-n)] \tag{12}$$

Jacobian matrix J is evaluated in each iteration[6]

$$J(i \mid k) = \frac{\partial E(i \mid k)}{\partial \Theta(i \mid k)} = \frac{\partial\left(T^T(k) - X^T(k)\Theta(i \mid k)\right)}{\partial \Theta(i \mid k)} = -X^T(k) \tag{13}$$

3. ADAPTIVE CONTROLLER

Linear quadratic (LQ) controller is state controller with feedback proportional gains from process states in the main form. Quadratic criterion without end state is

$$J = \sum_{k=0}^{N} (x^T(k) Q x(k) + u^T(k) R u(k)) \tag{14}$$

where matrix R scales control energy, matrix Q scales errors of system states. Inputs and outputs of process are known in this case. Model estimation is computed from them. LQ controller computed from input-output model only is known as pseudo state. The LQ controller solution is based on iterations of Riccati equation. Control signal is given by

$$u(k) = -K_{LQ} x(k) \tag{15}$$

K_{LQ} matrix is solved[2][5]

$$K_{LQ} = \left[R + B^T P_{LQ} B\right]^{-1} B^T P_{LQ} A \tag{16}$$

$$P_{LQ} = Q + K_{LQ}^T R K_{LQ} + (A - BK_{LQ})^T P (A - BK_{LQ}) \tag{17}$$

4. SIMULATION RESULTS

The verification possibilities to speed-up convergence of parameters were done in Matlab/Simulink. Process transfer function was

$$F(s) = \frac{0.7}{(10s+1)(s+1)} \tag{18}$$

Process was excited desired signal with or without added pseudorandom signal at the start only. Pseudorandom signal was cut off from desired signal in an instant when the LQ controller was switched on. Only the identification proceeded at this time. LQ controller was started up at 30 second. All algorithms were programmed as S-functions. The quantizers were added on the input and output of the S-functions. Quantizers presented possibilities of converters, 14-bits converters were used in this case. Ramp function was used as the reference trajectory. Controller had to trace this trajectory. Sampling time was $T_{vz}=0,1s$.

Figure 2 and Figure 3 show process behavior and control signal of systems with LQ controller with identification by RLS and L-M methods. Both figures are with added pseudorandom signal to the desired signal at the start. Result difference between RLS identification with and without added pseudorandom signal is in the first step of control signal only when the LQ controller is switch on. First step is smaller in the case using RLS with added pseudo-random signal at 30s. Close loop was unstable in the case using L-M method without pseudorandom signal when the LQ controller was started. It is given by the fact that the convergence speed of L-M algorithm to real values is a smaller than RLS identification method.

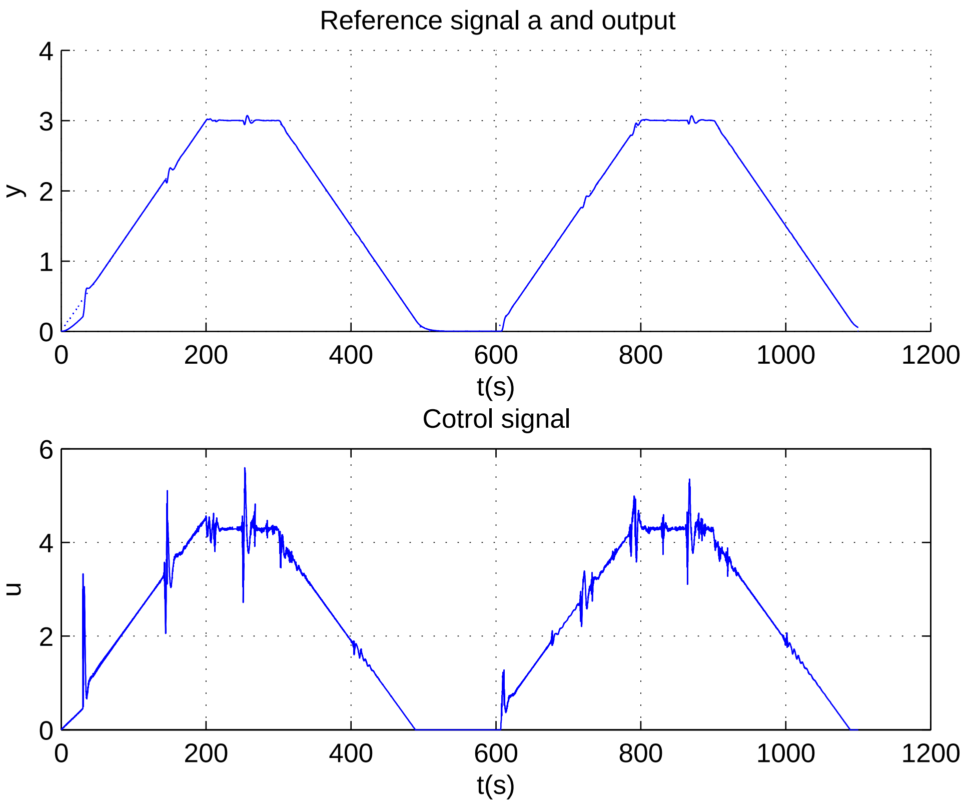

Figure 2: RLS identification method with added pseudorandom signal

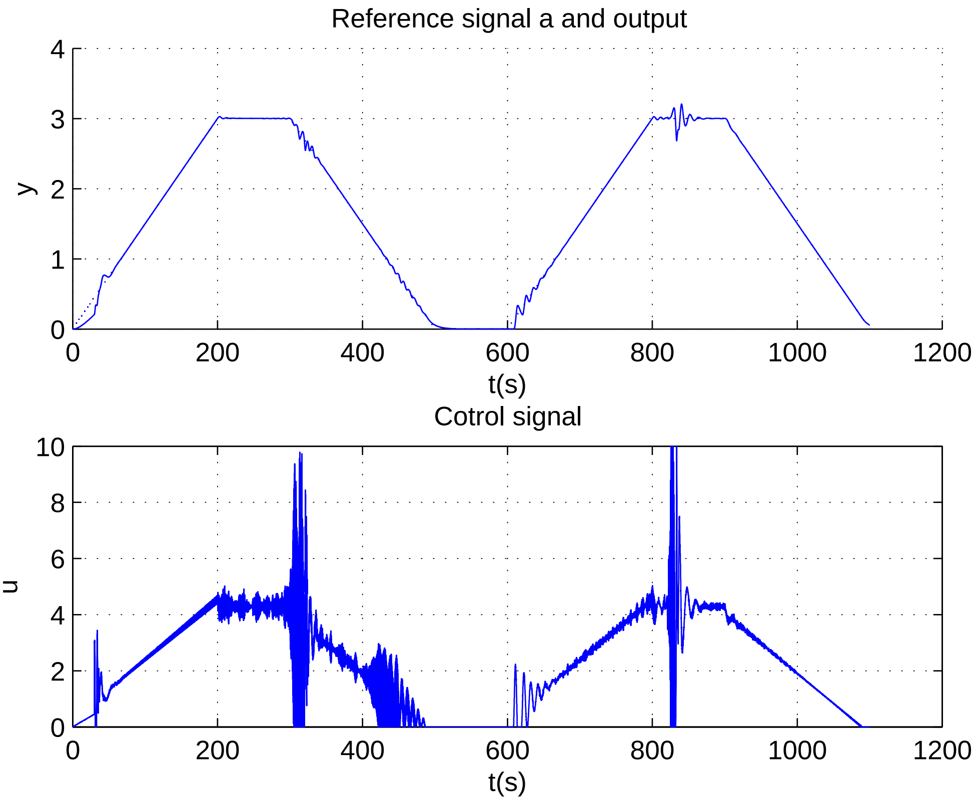

Figure 3: L-M learning method with added pseudorandom signal

5. CONCLUSION

This paper discuses possibility using pseudorandom signal added to desired signal to speed up convergence of ARX model parameters. As it can be seen from simulation results is addition pseudorandom signal to desire signal one of possibilities to speed up convergence. It can be seen when the identification based on neural networks is used.

REFERENCES

[1] Pivoňka, P.: *Optimalizace regulátorů*. VUT, Brno, e-text 2005.

[2] Lorenc, V.: *Materiály do cvičení Optimalizace regulátorů*. VUT, Brno, e-text 2007.

[3] Ljung, L., Söderström T.: *Theory and Practice of Recursive Identification*. 1983. ISBN 0-262-12095-X.

[4] Filatov, N., Unbehauen, H.: *Adaptive Dual Control*: *Theory and Applications*. 2004. ISBN 3-540-21373-2

[5] Bobál, V., Böhm, J., Prokop, R., Fessl, J.: *Praktické aspekty samočinně se nastavujících regulátorů*: *algoritmy a implementace*. 1999. ISBN 80-214-1299-2

[6] Veleba, V. Pivoňka, P.: *Adaptive controller with Identification Based on Neural Networks for Systems with Rapid Sampling Rates*. WSEAS Transactions on Systems, 2005, vol. 4. issue 4, pp. 385 – 388, ISSN 1109-2777