Adaptive Controllers by Using Neural Network Based Identification for Short Sampling Period

Petr Pivoňka, Václav Veleba

***Abstract*—**The use of short sampling period in adaptive control has not been described properly when controlling the real process by adaptive controller. The new approach to analysis of on-line identification methods based on one-step-ahead prediction clears up their sensitivity to disturbances in control loop. On one hand faster disturbance rejection due to short sampling period can be an advantage but on the other hand it brings us some practical problems. Particularly, quantization error and finite numerical precision of industrial controller must be considered in the real process control. We concentrate our attention on dealing with adverse effects that work on real-time identification of process, especially quantization. It is shown; that a neural network applied to on-line identification process produces more stable solution in the rapid sampling domain.

***Keywords*—**Adaptive Controllers, Neural Networks for Identification, Comparison of Identifications methods, Rapid Sampling Domain.

I. Introduction

THE correct choice of the sampling period is a top-priority task in adaptive control. It is important to keep in mind, that long sampling period results problem with aliasing. On the other hands, rapid sampling causes problem with numerical stability. The most advantages of fast sampling are faster disturbances cancellation and smaller overshoot in the control process.

When we use the classical identification method with a rapid sampling rate for a real time identification of a real dynamic plant, this method fails, though simulation (even with simulated disturbances) behaves differently. This fact is caused by existence of quantization in an A/D converter. The quantization effect, the real noise and other nonlinearities of the plant make on-line identification more complex than could be expected. We will show that a possible solution of this problem is using of an identification method based on neural networks.

Manuscript received October 30, 2006: Revised version received March 9, 2007. This work was supported in part by the Czech Science Foundation GAČR project No. 102/06/1132 Soft Computing in Control and by the Czech Ministry of Education in the frame of MSM MSM0021630529 Intelligent Systems in Automation.

P. Pivoňka is with the Brno University of Technology, Faculty of Electrical Engineering and Communication, Dept. of Automation, Kolejní 4, 612 66 Brno, Czech Republic (corresponding author to provide phone: +420 541 141 172; fax: +420 541 141 123; (e-mail: pivonka@feec.vutbr.cz).

V. Veleba, was with the Brno University of Technology, Faculty of Electrical Engineering and Communication, Dept. of Automation. He is now with the B&R Company, Brno, Czech Republic (e-mail: VaclavVeleba@br-automation.com.

II. On-Line Identification

The basic idea of on-line identification is to compare the output of estimated system with the output of model during some time. The model is describable as a parameter vector. The aim is to adjust parameter until the model output is similar to the observed system output. The classical Recursive Least mean Square (RLS) identification method and gradient method compares only actual model output to system output, while the identification method based on neural network-approaches compares outputs over some interval of time defined by length of a training set.

A. Linear Regression

The predicted output can be expressed as a linear function of vector $\theta(k)$; that is

$$\hat{y}(k)=\varphi^{\mathrm{T}}(k)\theta(k) \tag{1}$$

where $\varphi^{\mathrm{T}}(k)$ is the vector of measured variables. We use a discrete time shift operator model ARX expressed in form

$$\hat{y}(k)=\sum_{i=1}^{m} b_i u(k-i)-\sum_{j=1}^{n} a_j y(k-j) \tag{2}$$

where b_i and a_j are in the vector $\theta(k)$ parameters.

$$\theta(k)=[b_1(k) \quad \cdots \quad b_m(k) \quad a_1(k) \quad \cdots \quad a_n(k)]^{\mathrm{T}} \tag{3}$$

In accordance with (1) we write

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

B. Classical RLS Identification

RLS is a widely used method. It is often used in case that data comes continuously in time (e.g. on-line estimation). In each sampling period vector $\theta(k)$ is updated by

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

It is interesting to note that the model $\theta(k+1)$ is updated through a prediction error that has a very small value even if inaccurate vector $\theta(k)$ is used. This problem cause that RLS is sensitive to disturbances.

The posterior information of the model errors is incorporated in covariance matrix $P(k)$ that is updated too

$$P(k+1)=P(k)-K(k+1)\varphi^{T}(k+1)P(k) \tag{6}$$

Vector of correction $K(k+1)$ is computed by applying

covariance matrix

$$K(k+1)=P(k)\varphi(k+1)\left[1+\varphi^T(k+1)P(k)\varphi(k+1)\right]^{-1} \quad (7)$$

C. Simple Gradient Identification

Simple gradient identification is an older method, which become more popular by expansions of neural network techniques. It is suitable especially for fluently perturbed system identification. It has the worst quality for identification of unknown processes (from described methods), but its advantage is simplicity and small time-consuming computation.

$$\theta(k+1)=\theta(k)+\eta(\theta(k)-\theta(k-1))+\mu\left(y(k+1)-\varphi^T(k+1)\theta(k)\right)\theta(k) \quad (8)$$

You can note similarly to the RLS method that the model is updated by the same principle. That affects the similar problems like the RLS method. Parameter η is momentum constant and parameter μ is learning-rate constant.

D. Identification Based on Neural Network with Levenberg-Marquardt Training Method

The Levenberg-Marquardt iterative algorithm, gives a numerical solution to the problem of minimizing a sum of squares of generally nonlinear functions. We can consider a real dynamic system to be nonlinear because it contains nonlinear saturation, A/D (D/A) converters with constrained inputs (outputs) and quantization. The L-M identification works in accordance to the principle of searching of global minima of an error between the plant last outputs and model outputs through entire a states buffer

$$X(k)=\left[\varphi(k) \quad \varphi(k-1) \quad \ldots \quad \varphi(k-p)\right] \quad (9)$$

The states buffer (training set) contains a certain number of last states of the plant, where p is a length of buffer. It is desirable to set the length of buffer that the buffer contains a time period invariant to the sampling rate.

The minimization algorithm iterate certain number of iterations *i* at each identification step *k*

$$\theta(i|k+1)=\theta(i|k)-\left[J^T(i|k)J(i|k)+\lambda I\right]^{-1}J^T(i|k)E(i|k) \quad (10)$$

where $E(i|k)$ is a vector of errors (11) between model output and estimated system output $T(k)$ in (12):

$$E(k)=T^T(k)-X^T(k)\theta(k) \quad (11)$$

$$T(k)=\left[y(k) \quad y(k-1) \quad \ldots \quad y(k-p)\right] \quad (12)$$

The Jacobian matrix $J(i|k)$ represents the best linear approximation to a differentiable vector-valued function near a given point and is evaluated at each of iteration.

$$J(k)=\frac{\partial E(k)}{\partial\theta(k)}=\frac{\partial\left(T^T(k)-X^T(k)\theta(k)\right)}{\partial\theta(k)}=-X^T(k) \quad (13)$$

The (non-negative) damping factor λ is adjusted at each of iteration by evaluation of a quadratic error.

III. Quantization Effect

The quantization effect is more known for example in instrumentation theory or signal processing theory than in control theory. Furthermore, in control theory the phenomenon has been usually disregarded. It is due to the fact that the conditions used in process control allow the quantization effect to be ignored. Nowadays, when the sampling period is demanded to be very short and the requirements for the control precision are higher than before, the quantization effect plays considerable role in the practical control.

A. Quantization Error

The process control of continuous time system and the control of sampled continuous time system are two different fields. It happens that the controller design is created without precise knowledge of sampling, shaping and quantization effect.

The A/D and D/A converters are necessary parts of each real-time system [2]. The basic feature of the converters is to convert continues signal to discrete values and back (see Fig. 1).

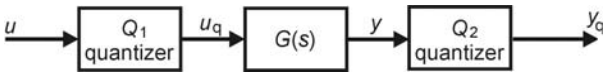

Fig. 1 the real model with A/D and D/A converters represented as quantizer

$$y=G(s)u_q \quad y_q=Q_2(y) \quad u_q=Q_1(u)$$

$$y_q=Q_2\{G(s)Q_1(u)\} \quad (14)$$

The quantization error *e* is limited to quantization band ≡ 1 LSB. The quantization range Q_{RANGE} and the quantization resolution Q_{RES} are basic parameters for definition of the quantization band. For example $2^{Q_{\text{RES}}}=2^8=256$ number of codes is given for Q_{RES}. Next, for bipolar converters ±10 V the quantization band is Q_{BAND} = 10/256 = 39.1 mV≈ ≈ 0.04 V. Therefore the value in finite word-length precision is numerically rounding off to the three valid places divisible by ≈ 0.04 V.

The quantization error may be modeled as deterministic or stochastic signal in linear analysis. In deterministic model, the error is modeled as constant having the size of quantization errors and with the resolution in the arithmetic calculation. In the stochastic model, the error introduced by rounding or quantization is than described as additive white noise with rectangular distribution [1]. Next paper [3] deals with quantization analysis and shows cases where after linearization the round off quantization error is uncorrelated with quantizer input.

Let us consider the modeling of quantizer. The model can be built from quantization effect description to show the disturbance properties of quantization effect. The model can be seen in Fig. 2, where the linear part of value u_L is disturbed by non-linear part represented as quantization error e. This point of view is very simple, given from description of

quantization effect and it gives us the beginning point for explanation of quantization effect.

It can be written that

$$u_q = u_L + e \quad u_q = f(u) \tag{15}$$

where $f(\cdot)$ is exact non-linear function. The idea to derive presented equation explains answer to the question how the quantization error arises. It is shown that quantization error is dependent on quantizer input signal.

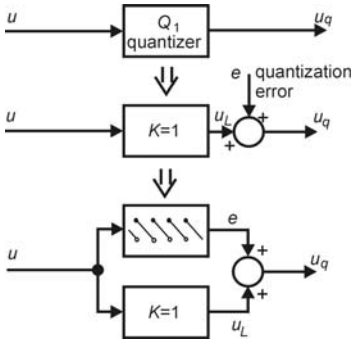

Fig. 2 principal model of quantization effect

This dependence is negligible as long as the sampling period is not too short and the numerical precision of quantizer error added to output is insignificant. In our case where the process control needs short sampling period, it is clearly shown that quantization error e is not independent from quantizer input u and hence cannot be treated as the independent additive noise [4], [5]. Next the quantization error cannot be treated as the Gaussian or even white noise because it is directly derived from quantizer input. It means that the noise is deterministic and it can be predicted. For example the quantization error is bigger when the amplitude of quantizer input is smaller.

B. The Limitation of Sampling Rates

The limitation of sampling rates in identification based on prediction error method. The correct setting of a sampling period in case of identification is described in [6]. The authors advise, without detailed analysis, setting of the sampling period empirically by the bandwidth w_B of a close loop. The domain, where is the sampling faster than the recommended one, is called Rapid Sampling.

A typical adaptive controller works with an unchangeable sampling period (the operating system doesn't allow a change of cycle time without a new initialization of the system), thus, when we demand high-adaptive algorithm, we get to rapid sampling domain easily. It will be shown; the boundaries of the domain are fuzzy and depending on disturbances in control loop.

We have cleared up that the sampling period is set, so the performance of identification will consider to the relative time constant. In this case it holds that with rising relative time constant of plant identification became more difficult.

We define the relative time constant as follows:

$$T_{\mathrm{REL}} = \frac{T_{\mathrm{G}}}{T_{\mathrm{S}}} \tag{16}$$

where T_G is a global time constant of plant. T_S is a sample time.

Prediction error methods (PEM) updates the model of a plant by prediction error $e(k,\theta)$:

$$e(k,\theta) = y(k) - \varphi^T(k)\theta(k) \tag{17}$$

where $y(k)$ is an actual output of plant and part $\varphi^{\mathrm{T}}(k)\theta(k)$ is a predicted output by model $\theta(k)$ with values of last plant inputs and outputs.

We define quadratic prediction error P_{Ei}

$$P_{E_i} = T_S \sum_{k=0}^{\frac{T_t}{T_S}} \left(e(k,\theta_i)\right)^2 \tag{18}$$

T_t is computation interval by lasting simulation; for (19) we used $T_t = 100\,\mathrm{s}$. Prediction error $e(k,\theta)$ falls exponentially to noise level by rising T_{REL} (16). The main disadvantage of PEM methods is obvious form Fig. 3. The figure shows relation between quadratic prediction error P_{Ei} and relative time constant. The predicted system had transfer function (19). His predictors θ_i (the parameters of predictor model F_i) was discrete equivalents of systems (20): Predictor 1 – $F_1(\mathrm{s})$; Predictor 2 – $F_2(\mathrm{s})$; Predictor 3 – $F_3(\mathrm{s})$ – only for comparison very bad estimation of (19).

$$F(s) = \frac{2}{(10s+1)(s+1)^2} \tag{19}$$

$$F_1(s) = \frac{1}{(s+1)^3}; \quad F_2(s) = \frac{10}{s(s+1)^2}; \quad F_3(s) = \frac{100}{s^3} \tag{20}$$

The theoretic boundary of T_{REL} is given by levels of quantization noise of A/D converters (10-bit and 16 bit one is shown in the Fig. 3).

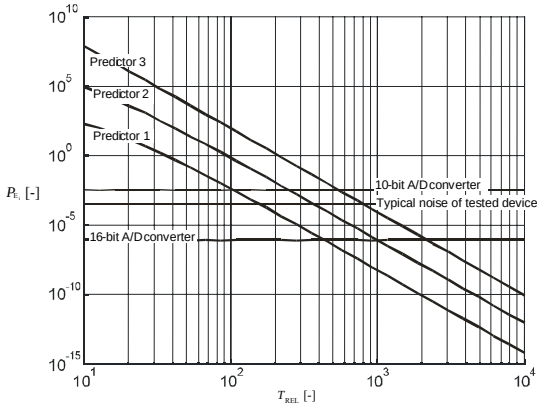

Fig. 3 the relation between prediction error P_{Ei} and relative time constant for different predictors $F_i(s)$

IV. The Influence of Rapid Sampling and Quantization

In this section is the influence of rapid sampling and quantization on the applicability of identification methods described. In the last section, we explained that there is an upper bound of a sampling rate (or relative time constant) at prediction error methods. Now, we will show, that the identification method based on a neural networks approach with Levenberg-Marquardt training algorithm provides better estimate of a model dynamics than the gradient method and RLS method. This is caused by an accumulation of prediction error by training set (13).

The validation of the trained model by one-step prediction error is insufficient (see last section), therefore we validate similarity between plant and its model by cumulating difference between output of plant and free model. In each step we compute free prediction error:

$$e_{\mathrm{V}}(k,\theta) = y(k) - \varphi^{T}(k,\theta)\theta(k) \tag{21}$$

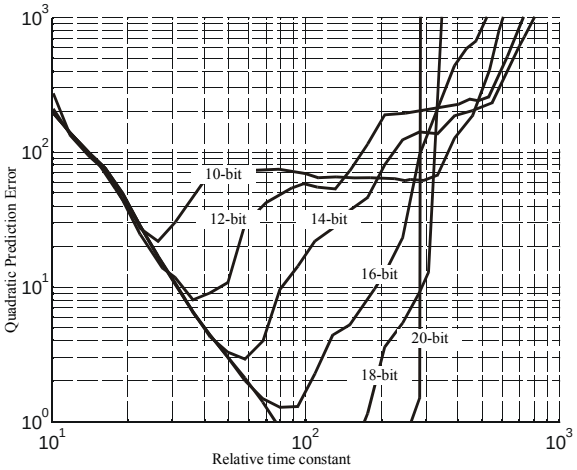

Fig. 4 the influence of sampling period and quantization (from 10-bit to 20-bit converter) on the performance of a RLS algorithm identification

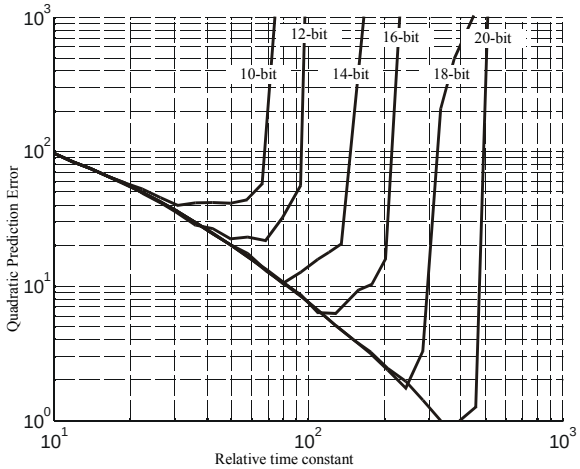

Fig. 5 the influence of sampling period and quantization (from 10-bit to 20-bit converter) on the performance of a Simple Gradient Algorithm identification

Now, the content of the regression vector $\varphi^{\mathrm{T}}(k,\theta)$ depend on the model $\theta(k)$ which is adjusted in each step by a tested method. The criterion of similarity between plant and model:

$$V_F = T_S \sum_{k=0}^{\frac{T_t}{T_S}} \left(e_{\mathrm{V}}(k,\theta)\right)^2 \tag{22}$$

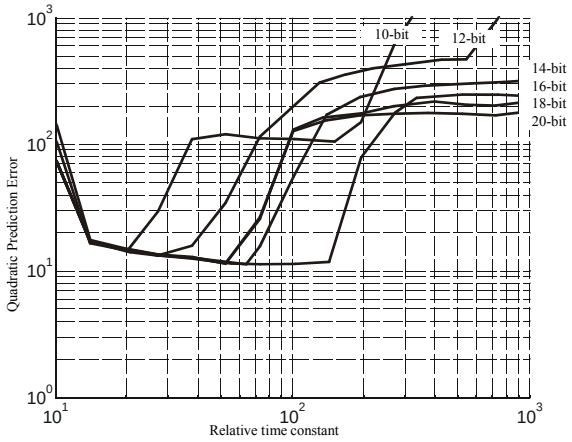

Fig. 6 the influence of sampling period and quantization (from 10-bit to 20-bit converter) on the performance of an Identification based on neural network with Levenberg-Marquardt training method

Each identification method estimates parameters of plant (19). The gain of plant is perturbed by 50%.

The values of (22) are shown in next Fig. 4, Fig. 5 and Fig. 6 for each identification method and depend on sampling rates and quantization. Note that the identification based on a neural network gives less accurate solution, but it produces the most stable solution in the rapid sampling domain. This feature is more obvious in real system control (see Fig. 6).

V. Adaptive Control

Application that on-line parameter identification can be put to is in adaptive control. The idea of adaptive controllers (or self-tuning controllers) is to combine an on-line identification with on-line control law synthesis. Many of control law synthesis approaches are based on two methods – pole placement and inversion of dynamic. Both of the methods are numerically sensitive to the bad-estimated model of a plant.

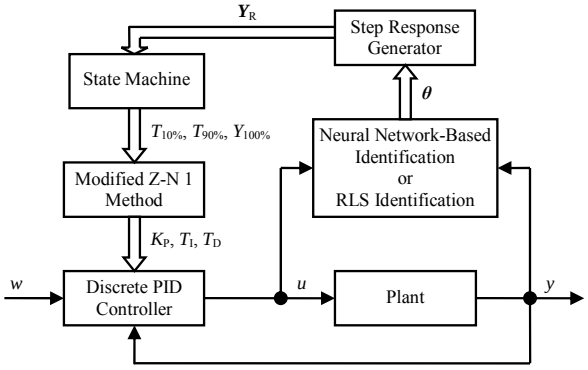

Fig. 7 the architecture of the adaptive heuristic controller based on modified Ziegler-Nichols open loop method

The requirement for correctly computed vector θ is not often fulfilled during controlling of a real system with a higher order. Therefore, we use simple heuristic synthesis based on modified Z-N 1 method. The basis architecture of the adaptive controller we discussed is shown in Fig. 7.

$$L = 0.8T_{10\%} \quad R = \frac{Y_{100\%}}{T_{90\%} - L} \; ; \tag{23}$$

$$K_P = \frac{0.8}{RL}; \quad T_{\mathrm{I}} = 3L; \quad T_{\mathrm{D}} = 0.5L; \tag{24}$$

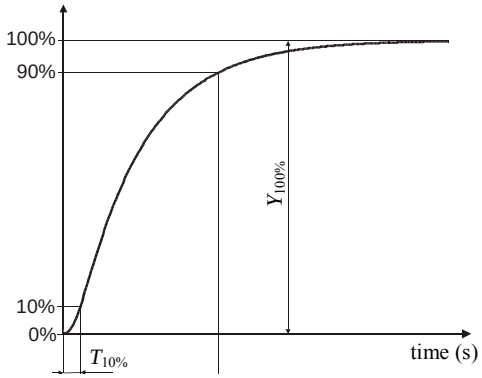

Fig. 8 the characteristic points used for a tuning of the adaptive heuristic controller based on modified Ziegler-Nichols open loop method

The step response generator generates the sequence Y_{R} of a step response of the estimated model θ. Then, the state machine finds characteristic points $Y_{10\%}$, $Y_{90\%}$ and $Y_{100\%}$ in the sequence Y_{R} (Fig. 8). These values are used to design the PID discrete controller according modification [10] – (23), (24).

A. Real Process Control Results

The comparison of a controller that uses RLS identification method with a controller that uses identification based on neural network with Levenberg-Marquardt training method is shown. The real process control proves the advantages of the second identification method. The transfer function of controlled dynamic system was

$$F(s) \approx \frac{1}{(10s+1)(s+1)^2} \tag{25}$$

with interval of linearity <-6, 7> (V).

Fig. 9 and Fig. 10 shows the both methods of identification applied in an adaptive control. Both controllers work with the same settings. The sampling period was set to T_S = 0.1 s. The short sampling period is used in order to reduce an overshot and mainly for a disturbance cancellation.

VI. CONCLUSION

This paper discus influences affecting the process of identification at rapid sampling domain. We compared three methods of on-line identification: the recursive least square method, the gradient identification method and the identification method based on neural network approach with Levenberg-Marquardt minimization. On the basis of section 5 we applied the neural estimator for an adaptive control.

The real process control shows the advantage of using identification based on neural networks in the real application against the classical identification methods. The identification based on a neural network gives less accurate solution, but it produces the most stable solution in the rapid sampling domain.

It was shown that:

Quantization deeply affects a performance of identification.

Neural networks based identification enables plants with greater T_{REL} to be used in adaptive control process (with shorter sampling period).

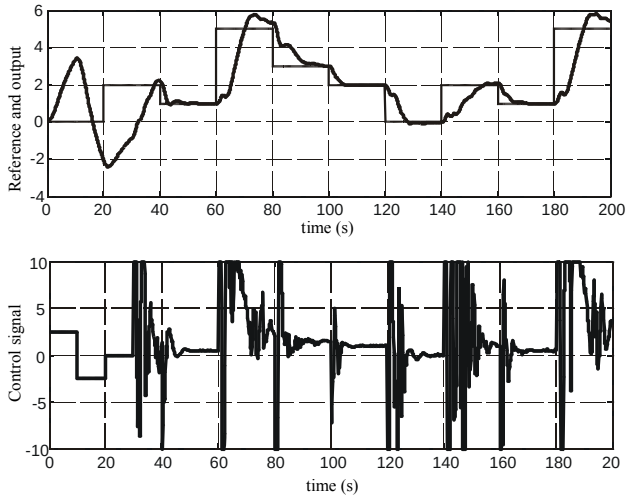

Fig. 9 real process control; RLS identification method (12–bit A/D and D/A converter)

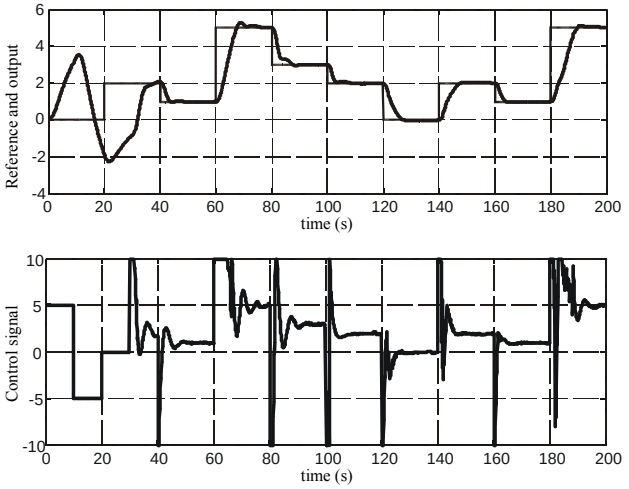

Fig. 10 real process control – identification method based on neural network (12–bit A/D and D/A converter)

On-line control law synthesis with step response generator provides stable coefficients of discrete PID controller.

Application that on-line parameter identification can be put to is in adaptive control. The idea of adaptive controllers (or

self-tuning controllers) is to combine an on-line identification with on-line control law synthesis. The basis architecture for the adaptive controller is discussed in [9]. We used some variant of this architecture.

REFERENCES

[1] K., J. Åström, and B. Wittenmark, *Computer-Controlled Systems, Theory and Design.* Prentice Hall, Inc., 1997.

[2] K. Fowler, "Analog to Digital Conversion in Real-Time Systems, Part 7," *IEEE Instrumentation & Measurement*, vol. 6, 2003, pp. 58–64.

[3] D. Williamson, *Digital Control and Implementation: Finite Wordlength Considerations.* Prentice Hall, Inc., 1991.

[4] K. Švancara, "Adaptive Optimal Controller with Identification Based on Neural Network," Ph.D. dissertation, Dept. of Aut. and Meas., Brno University of Technology, Czech Republic, 2004.

[5] P. Pivoňka and K. Švancara, "The Use of Short Sampling Period in Adaptive Control," *WSEAS Transactions on Systems*, vol. 10, No. 4, 2005, pp. 1627–1636.

[6] R. H. Middleton and G. C. Goodwin, *Digital Control and Estimation: A United Approach*. Prentice Hall, Inc., 1990.

[7] L. Ljung, *System Identification: Theory for User*. Prentice Hall, Inc., 1987.

[8] T. Söderström and P. Stoica, *System Identification*. Prentice Hall, Inc., 1989.

[9] A. Cichocki and R. Unbehauen, *Neural networks for optimization and signal processing*. John Wiley & Sons, 1994.

[10] K. J. Åström and T. Hägglund, "New Tuning Methods for PID Controllers," *In Proc. 3rd European Control Conference ECC95*, Rome, 1995, pp. 2456–2462.