Simulation of Adaptive LQ Control of Nonlinear Process

Jiri Vojtesek, Petr Dostal

Department of Process Control, Faculty of Applied Informatics,
Tomas Bata University in Zlin,
nam. T. G. Masaryka 5555, Zlin, 760 01, Czech Republic,
vojtesek@fai.utb.cz, dostalp@fai.utb.cz

Abstract. The contribution is focused on the adaptive control of the nonlinear system represented by the continuous stirred-tank reactor with the spiral cooling in the jacket. The mathematical model of this reactor is described by two nonlinear ordinary differential equations which were solved numerically. The adaptive controller is based on the choice of the external linear delta model of the originally nonlinear process, parameters of which are identified recursively and parameters of the controller are recomputed too. The structure of the controller was constructed with the use polynomial synthesis together with linear-quadratic approach. The resulted controller fulfills basic control requirements and it can be used for system with negative control properties. All experiments were made by simulations in the MATLAB environment.

Keywords: Adaptive control, External linear model, Recursive identification, Polynomial synthesis, Linear quadratic theory, Continuous stirred-tank reactor, Nonlinear system.

1. Introduction

The use of the computer simulation not only in the control engineering grows rapidly nowadays with the increasing speed of the computers and low prices of the hardware. Furthermore, the simulation is very often used at present as it has many advantages over an experiment on a real system, which is not feasible and can be dangerous, time and money demanding. A modelling of the system usually precedes the simulation [1]. The mathematical model is a kind of abstract representation of the process which uses input, state or output variables, relations between these variables collected in the set of mathematical equations [1] and [2]. Some simulation and modelling examples can be found also in [3] and [4].

It is known, that almost all processes in the nature have a nonlinear behaviour [1], [5] and our goal is to cope with this nonlinearity. Typical examples of nonlinear systems are chemical reactors. A chemical reactor is a vessel or pipe which is used for the production of chemicals used in chemical, biochemical, drug and other industries through a specific reaction inside [6]. The controlled system here is represented by a Continuous Stirred-Tank Reactor (CSTR) as a typical member of a group of nonlinear systems used not only in the chemical industry. The mathematical model of the plant is described by the set of two nonlinear Ordinary Differential Equations (ODE) [7].

The thorough analysis of the system usually precedes the controller design. Steady-state and dynamic analyses as a typical simulation tools gives overview of system's behavior especially for chemical reactors [5], [8], [9] etc. The methods used here was a Simple iteration method [10] and a Runge-Kutta's standard method for the numerical solving of set of ODE. Big advantage of both methods is that both are easily programmable or even build-in functions in popular mathematical software, such as MATLAB [11], Mathematica etc.

The idea of an adaptive control [12] comes from the nature where every organism even humans try to "adapt" for the current environment. Transformed to the control theory, the controller also adapts parameters, structure etc. to the actual state of the controlled plant or desired course of the output signal [13]. The adaptive approach here is based on the approximation of the nonlinear system by the appropriate linear model, parameters of which are estimated online.

The structure of the controller uses the polynomial synthesis [14] with Linear Quadratic (LQ) theory [15]. Resulted controller fulfills basic requests for the control loop such as stability, reference signal tracking and disturbance attenuation – [14] and [16].

Although there could be found a lot of contributions dealing with the simulation of control, the goal of this contribution is to describe the procedure from the steady-state and dynamic analyses to the design of the hybrid adaptive controller for temperature control inside the CSTR as a typical member of the nonlinear processes. This method could be applied to similar nonlinear processes which are described also by the mathematical model.

The contribution is divided into six main parts. The second section after this introduction describes the mathematical model of the controlled plant (CSTR) and the results of steady-state and dynamic analyses. Then, the third part describes theoretical background to the adaptive control with recursive identification and the LQ approach and polynomial synthesis of the controller. The fourth part is dedicated to various simulation experiments of the proposed controller on the mathematical model. After that, the last part before conclusion presents usability of the controller to the real plant followed by the final conclusion and future work.

All experiments were done by the simulations on the mathematical software MATLAB, version 7.0.3.

2. Model of the Plant

The model under the consideration is a Continuous Stirred-tank Reactor (CSTR) with the spiral cooling in the jacket – see Figure 1.

The mathematical description of the process uses material and heat balances inside the reactor. Full description of the process is, of course, very complex because of number variables. There must be introduced some simplifications to reduce this complexity – we expect that reactant is perfectly mixed and reacts to the final product with the concentration $c_A(t)$. Furthermore we also assume that volume, heat capacities and densities are constant during the control.
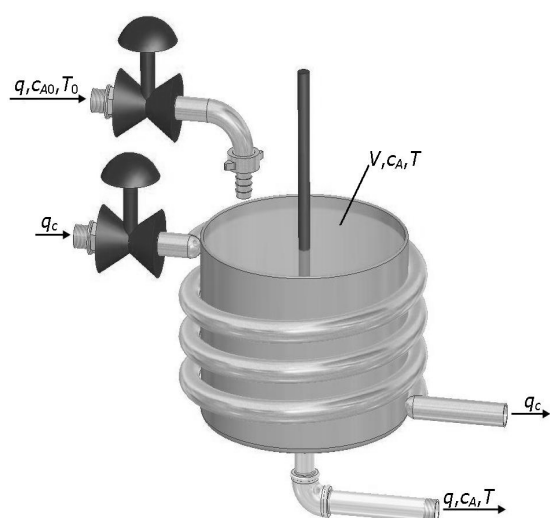

Figure 1. Continuous Stirred-Tank Reactor (CSTR)

The mathematical model in the form of Ordinary Differential Equations (ODE) with all these simplifications has then form [7]:

$$\frac{dT}{dt} = a_1 \cdot (T_0 - T) + a_2 \cdot k_1 \cdot c_A + \dots$$

$$\dots + a_3 \cdot q_c \cdot \left(1 - e^{\frac{a_4}{q_c}}\right) \cdot (T_0 - T) \quad (1)$$

$$\frac{dc_A}{dt} = a_1 \cdot (c_{A0} - c_A) - k_1 \cdot c_A$$

where $a_{1\text{-}4}$ are constants computed as

$$a_1 = \frac{q}{V}; \ a_2 = \frac{-\Delta H}{\rho \cdot c_p};$$

$$a_3 = \frac{\rho_c \cdot c_{pc}}{\rho \cdot c_p \cdot V}; \ a_4 = \frac{-h_a}{\rho_c \cdot c_{pc}} \quad (2)$$

The variable t in the previous equations denotes time, T is used for temperature of the reactant, V is volume of the reactor, c_A represents concentration of the product, q and q_c are volumetric flow rates of the reactant and cooling respectively. Indexes $(\cdot)_0$ denote input values of the variables, $(\cdot)_c$ is used for variables related to the cooling and variables without index are connected with the reactant.

The main nonlinearity of this model can be found in the reaction rate, k_1, as a nonlinear function of the reactant temperature, T, which also state variable in (1). This reaction rate is computed via Arrhenius law, e.g.

$$k_1 = k_0 \cdot e^{\frac{-E}{R \cdot T}} \quad (3)$$

where k_0 is the reaction rate constant, E denotes an activation energy and R is a gas constant.

The mathematical model in (1) together with constants (2) and variable k_1 in (3) defines a *nonlinear system with lumped parameters*. There could be several input variables but from the practical point of view, only volumetric flow rates of the reactant, q, and the cooling, q_c, were chosen as input variables. The output variables are the temperature of the reactant, T, and the concentration of the product, c_A. The fixed parameters of are shown in Table 1 [7]:

Table 1. Fixed parameters of the reactor

Name of the variable	Value
Reactant's flow rate	$q = 0.1 \; m^3.min^{-1}$
Reactor's volume	$V = 0.1 \; m^3$
Reaction rate constant	$k_0 = 7.2 \cdot 10^{10} \; min^{-1}$
Activation energy to R	$E/R = 1 \cdot 10^4 \; K$
Reactant's feed temp.	$T_0 = 350 \; K$
Reaction heat	$\Delta H = -2 \cdot 10^5 cal.mol^{-1}$
Specific heat of reactant	$c_p = 1 \; cal.g^{-1}.K^{-1}$
Specific heat of cooling	$c_{pc} = 1 \; cal.g^{-1}.K^{-1}$
Density of the reactant	$\rho = 1 \cdot 10^3 \; kg. \; m^3$
Density of the cooling	$\rho_c = 1 \cdot 10^3 \; kg. \; m^3$
Feed concentration	$c_{A0} = 1 \; kmol. \; m^3$
Heat transfer coef.	$h_a = 7 \cdot 10^5 cal.min^{-1}.K^{-1}$

Analyses inside the reactor are the next step after the developing of the mathematical model. There were used steady-state and dynamic analysis to obtain information about the type and behaviour of the system.

Steady-state Analysis

Steady-state analysis for stable systems involves computing values of state variables in time $t \rightarrow \infty$, when changes of these variables are equal to the zero. That means that all equations which consist of derivations with respect to the time in (1) have these derivations equal to the zero, e.g. $d(\cdot)/dt = 0$. The mathematical model (1) is then transformed to the set of nonlinear equations:

$$T^s = \frac{a_1 \cdot T_0 + a_2 \cdot k_1 \cdot c_A^s + a_3 \cdot q_c \cdot T_0 \cdot \left(1 - e^{\frac{a_4}{q_c}}\right)}{a_1 + a_3 \cdot q_c \cdot \left(1 - e^{\frac{a_4}{q_c}}\right)} \quad (4)$$

$$c_A^s = \frac{a_1 \cdot c_{A0}}{a_1 + k_1}$$

The simple iteration method [10] was used for solving of this set of equations (4).

The heat balance of the reactor shows interesting thing about this reactor – it has 3 steady-states. The Q_r in Figure 2 is used for heat of the reactant and Q_c denotes heat of the coolant. It holds, that $Q_r = Q_c$ in the steady state which means that this type of reactor has two stable (S_1 and S_2) and one unstable (N_1) steady-states.
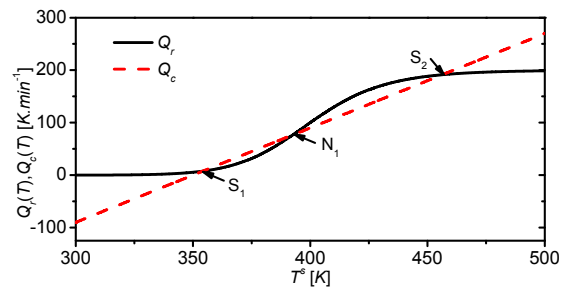

Figure 2. Heat balance inside the reactor

Values of the state variables in these points are

$$S_1: \quad T^s = 354.23 \; K \qquad c_A^s = 0.9620 \; kmol.m^{-3}$$
$$N_1: \quad T^s = 392.45 \; K \qquad c_A^s = 0.6180 \; kmol.m^{-3} \quad (5)$$
$$S_2: \quad T^s = 456.25 \; K \qquad c_A^s = 0.0439 \; kmol.m^{-3}$$

Although it seems that the second steady-state S_2 has better efficiency (95.6% of the reactant reacts), the steady-state temperature of the reactant is very high which in some case means that this point is only theoretical and not practically feasible. That is why the first steady-state S_1 was used in this work.

The steady-state analysis here was done for different input volumetric flow rates q and q_c and results are shown in the following figures where T^s denotes steady-state value of the reactant's temperature and c_A^s is steady-state value of the product's concentration.
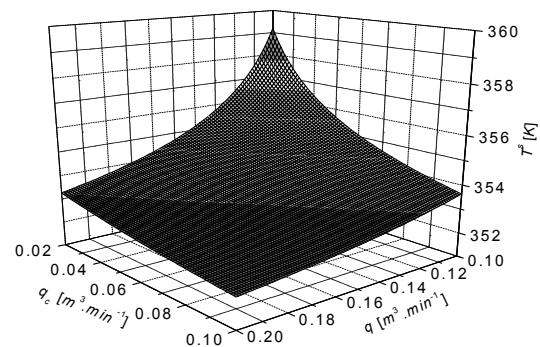

Figure 3. Steady-state values of the temperature T^s for different volumetric flow rates q and q_c

Graphs have shown nonlinearities in both product's concentration and temperature. The steady-state analysis usually results in the optimal working point. From the practical and mainly cost point of view is good to choose volumetric flow rates as low as possible. The working point is then characterized by the pair of volumetric flow rates $q_c = 0.08 \; m^3.min^{-1}$ and $q = 0.10 \; m^3.min^{-1}$.
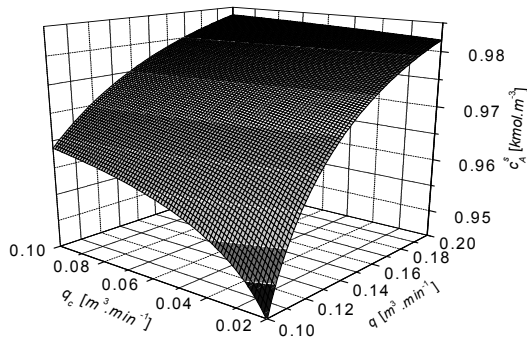
Figure 4. Steady-state values of the concentration c_A^s for different volumetric flow rates q and q_c

Steady-state values of state variables T and are c_A for this working point are $T^s = 354.26\ K$ and $c_A^s = 0.9620\ kmol.m^{-3}$.

Dynamic Analysis

The dynamic analysis is usually the next step after the steady-state analysis. The dynamic analysis observes behavior of the system after the step change of the input variable. The mathematical interpretation of this analysis is the numerical solving of the set of ODE in equations (1). The standard Runge-Kutta's method [10] was used in this work. This numerical method belongs to the class of high-order methods, it can be used for computation of the initial values or for the final result too and they are easily programmable. It can be also found as a build-in function in various mathematical simulation software, such as Matlab (functions ode23, ode43 etc.) [11], Mathematica (function NDSolve) etc.

As there were chosen two volumetric flow rates of the reactant and the cooling as input variables, several changes of both recomputed to % were examined in the following simulations. Input variables are then:

$$u_1(t) = \frac{q_c(t) - q_c^s}{q_c^s} \cdot 100\ [\%]$$

$$u_2(t) = \frac{q(t) - q^s}{q^s} \cdot 100\ [\%] \tag{6}$$

Results of dynamic analyses for several step changes of the input variables u_1 and u_2 are shown in following Figure 5 – Figure 8. Steady-state values of the state variables T^s and c_A^s are used as a initial conditions for the dynamic which also means that the course of output variables starts from these values.
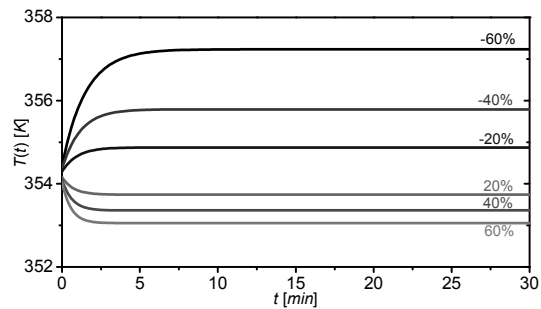

Figure 5. Time responses of the product's temperature, $T(t)$ to the step change of the volumetric flow rate of the coolant, u_1
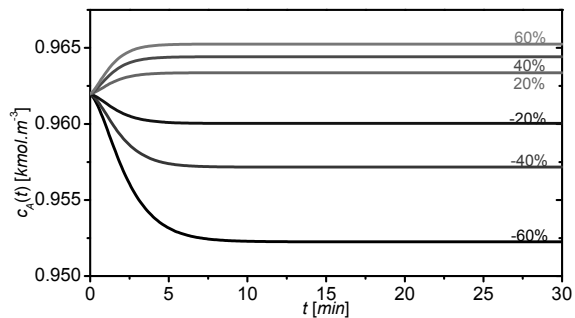

Figure 6. Time responses of the product's concentration, $c_A(t)$ to the step change of the volumetric flow rate of the coolant, u_1
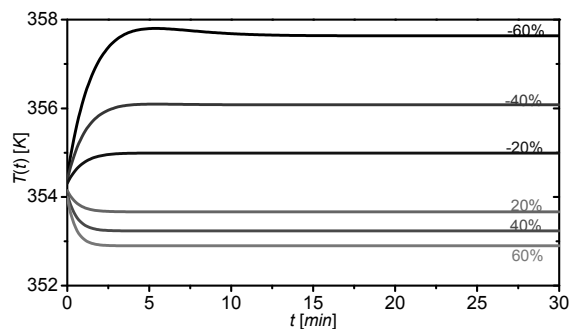

Figure 7. Time responses of the product's temperature, $T(t)$ to the step change of the volumetric flow rate of the reactant, u_2
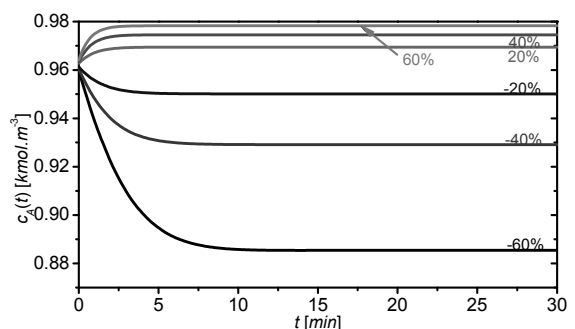

Figure 8. Time responses of the product's concentration, $c_A(t)$ to the step change of the volumetric flow rate of the reactant, u_2

Step responses for both input variables u_1 and u_2 show dynamical behaviour of the examined system. All outputs of the nonlinear system

could be from the control point of view described by the second order transfer function with relative order one in the polynomial form:

$$G(s) = \frac{b(s)}{a(s)} = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} \qquad (7)$$

3. Control of the Plant

The adaptive control [12] was used here as a control strategy. The basic idea of adaptive control is that parameters or the structure of the controller are adapted to parameters of the controlled plant according to the selected criterion [13]. The adaptive approach in this work is based on choosing an External Linear Model (ELM) (7) of the original nonlinear system whose parameters are recursively identified during the control. Parameters of the resulted controller are recomputed in every step from the estimated parameters of the ELM.

External Linear Model

In our case, the ELM has form of the Equation (7). It means that variables must be identified in continuous-time (CT) which is problem especially from the technical point of view. The other type of ELM is discrete-time (DT) model where input and output variables are read in the defined time intervals and parameters are estimated in these intervals too. Disadvantages of this model can be found on the choice of the sampling period – there is no general tool or advice to the right choice of this interval. Somewhere between these two types of ELM are the delta (δ-) models that belongs to the DT models but its parameters are close to the continuous ones for very small sampling period as it proofed in [17].

The CT model (7) could be also described in the general form and the use of input and output variables as:

$$a(\sigma)y(t) = b(\sigma)u(t) \qquad (8)$$

Where a and b are polynomials from (7) and σ is the differentiation operator. If we want to use δ–model we must introduce a new complex variable γ computed as [18]:

$$\gamma = \frac{z-1}{\beta \cdot T_v \cdot z + (1-\beta) \cdot T_v} \qquad (9)$$

and we can obtain infinitely many models for optional parameter β from the interval $0 \leq \beta \leq 1$

and a sampling period T_v. A forward δ-model was used in this work. The γ operator is then

$$\beta = 0 \Rightarrow \gamma = \frac{z-1}{T_v} \qquad (10)$$

The continuous model (8) is then rewritten to the form

$$a^\delta(\delta)y(t') = b^\delta(\delta)u(t') \qquad (11)$$

where polynomials a^δ and b^δ are discrete polynomials and their coefficients are different from those of the CT model $a(s)$ and $b(s)$. Time t' is discrete time and with the new substitution $t' = k - n$ for $k \geq n$ the δ-model for this concrete transfer function would be:

$$\delta^2 y(k-n) = b_1^\delta \delta u(k-n) + b_0^\delta u(k-n) - \\ - a_1^\delta \delta y(k-n) - a_0^\delta y(k-n) \qquad (12)$$

It means that the regression vector φ_δ is then

$$\varphi_\delta(k-1) = \left[-y_\delta(k-1), -y_\delta(k-2), \ldots \\ \ldots, u_\delta(k-1), u_\delta(k-2) \right]^T \qquad (13)$$

where y_δ and u_δ denotes the recomputed output and input variables to the δ-model and

$$y_\delta(k) = \frac{y(k) - 2y(k-1) + y(k-2)}{T_v^2}$$

$$y_\delta(k-1) = \frac{y(k-1) - y(k-2)}{T_v}$$

$$y_\delta(k-2) = y(k-2) \qquad (14)$$

$$u_\delta(k-1) = \frac{u(k-1) - u(k-2)}{T_v}$$

$$u_\delta(k-2) = u(k-2)$$

The computed vector of parameters θ_δ is generally

$$\theta_\delta(k) = \left[a_1^\delta, a_0^\delta, b_1^\delta, b_0^\delta \right]^T \qquad (15)$$

and parameters of this vector are computed by the on-line identification.

Finally, the differential equation (12) has the vector form:

$$y_\delta(k) = \theta_\delta^T(k) \cdot \varphi_\delta(k-1) + e(k) \qquad (16)$$

where $e(k)$ is a general random immeasurable component.

Recursive Identification

The Recursive Least-Squares (RLS) method is used for the on-line estimation of the vector of parameters θ_δ in (15). This method is well-known, easily programmable and widely used for the parameter estimation. It is usually modified with exponential or directional forgetting because parameters of the identified system can vary during the control which is typical for nonlinear systems and the use of some forgetting factor could result in better output response.

As an example, the RLS method with changing exponential forgetting used here is described by the set of equations:

$$\varepsilon(k) = y(k) - \varphi_\delta^T(k) \cdot \hat{\theta}_\delta(k-1)$$

$$\xi(k) = \left[1 + \varphi_\delta^T(k) \cdot \mathbf{P}(k-1) \cdot \varphi_\delta(k) \right]^{-1}$$

$$\mathbf{L}(k) = \xi(k) \cdot \mathbf{P}(k-1) \cdot \varphi_\delta^T(k)$$

$$\mathbf{P}(k) = \frac{1}{\lambda_1(k-1)} \Big[\mathbf{P}(k-1) - \dots \qquad (17)$$

$$\dots - \frac{\mathbf{P}(k-1) \cdot \varphi_\delta(k) \cdot \varphi_\delta^T(k) \cdot \mathbf{P}(k-1)}{\lambda_1(k-1) + \varphi_\delta^T(k) \cdot \mathbf{P}(k-1) \cdot \varphi_\delta(k)} \Big]$$

$$\hat{\theta}_\delta(k) = \hat{\theta}_\delta(k-1) + \mathbf{L}(k)\varepsilon(k)$$

where the changing forgetting factor λ_1 is computed from the equation

$$\lambda_1(k) = 1 - K \cdot \xi(k) \cdot \varepsilon^2(k) \qquad (18)$$

and K is small number, in our case $K = 0.001$.

Control Synthesis

The last theoretical step is the design of the controller. The polynomial approach [14] and [16] was used in this work. This method is based on the input-output model of the controlled system or its transfer function. It can be classified as an algebraic method and it is based on algebraic operations in the ring of polynomials. One of the biggest advantages of the polynomial method compared to the conventional method is that it provides not only relations for computing of the controller's parameters but the structure of the controller too. This structure fulfils general requirements for control systems and input signals (reference signal and disturbance) and it can be used for controlling of the systems with negative properties from the control point of view, such as non-minimum phase systems or unstable

systems. Another advantage is that the resulted relations are easily programmable.

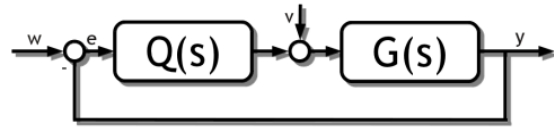The configuration with one degree-of-freedom (1DOF) was used here – see Figure 9.

Figure 9. 1DOF control configuration

The block G represents the transfer function of the plant (7), w is the reference signal (the wanted value), e stands for the control error ($e = w - y$), v is a disturbance, u is used for the control variable and y denotes the controlled output. Block Q is a transfer function of the controller which ensures three basic control conditions – (I.) stability, (II.) asymptotic tracking of the reference signal and (III.) load disturbance attenuation and it can be described by the polynomials in s-plain as

$$Q(s) = \frac{q(s)}{s \cdot \tilde{p}(s)} \qquad (19)$$

where degrees of the polynomials are computed from

$$\deg q(s) = \deg a(s), \ \deg \tilde{p}(s) \geq \deg a(s) - 1 \ (20)$$

and parameters of the polynomials $\tilde{p}(s)$ and $q(s)$ are computed from a Diophantine eq. [14]:

$$a(s) \cdot s \cdot \tilde{p}(s) + b(s) \cdot q(s) = d(s) \qquad (21)$$

Polynomials $a(s)$ and $b(s)$ are known from the recursive identification and the polynomial $d(s)$ on the right side of (21) is an optional stable polynomial. Roots of this polynomial are called poles of the closed-loop and their position affects quality of the control. One method for designing of this polynomial is a Pole-placement method [19]. Disadvantage of this method is the uncertainty – there is no rule for the choice of this root. The second method used in this work uses Linear Quadratic (LQ) approach [15] which is based on the minimization of the cost function

$$J_{LQ} = \int_0^\infty \left\{ \mu_{LQ} \cdot e^2(t) + \varphi_{LQ} \cdot \dot{u}^2(t) \right\} dt \qquad (22)$$

where $\varphi_{LQ} > 0$ and $\mu_{LQ} \geq 0$ are weighting coefficients, $e(t)$ is control error and $\dot{u}(t)$ denotes difference of the input variable.

The polynomial $d(s)$ is then divided into two polynomials $n(s)$ and $g(s)$, i.e. $d(s) = g(s) \cdot n(s)$, where polynomial $n(s)$ is connected to the controlled system via the spectral factorization of the polynomial $a(s)$ and $g(s)$ comes from the solution of the equation (22) again with the use of the spectral factorization. Both factorizations are shown in the following equation:

$$\left(a \cdot f\right)^* \cdot \varphi_{LQ} \cdot a \cdot f + b^* \cdot \mu_{LQ} \cdot b = g^* \cdot g \quad (23)$$
$$n^* \cdot n = a^* \cdot a$$

The polynomial $f(s)$ for control variable $u(t)$ and disturbance $v(t)$ from the ring of step functions is equal to s, i.e. $f(s) = s$. The resulted controller is strictly proper and the degree of $d(s)$ together with degrees of the controller's polynomials $q(s)$ and $\tilde{p}(s)$ from (20) are then

$$\deg d = \deg\left(g \cdot n\right) = 2\deg a + 1 = 5$$
$$\deg q = \deg a = 2; \quad \deg \tilde{p} \geq \deg a - 1 = 2 \quad (24)$$

which means that the transfer function of the controller (19) is

$$\tilde{Q}(s) = \frac{q_2 s^2 + q_1 s + q_0}{s \cdot \left(s^2 + p_1 s + p_0\right)} \quad (25)$$

Finally, parameters of $n(s)$ and $g(s)$ are practically computed from Equation (23) as

$$g_0 = \sqrt{\mu_{LQ} b_0^2}, \, g_1 = \sqrt{2g_0 g_2 + \varphi_{LQ} a_0^2 + \mu b_1^2},$$
$$g_2 = \sqrt{2g_1 g_3 + \varphi_{LQ}\left(a_1^2 - 2a_0\right)}, g_3 = \sqrt{\varphi_{LQ}}, \quad (26)$$
$$n_0 = \sqrt{a_0^2}, n_1 = \sqrt{2n_0 + a_1^2 - 2a_0}$$

We can say, that the resulting controller is "hybrid" because the polynomial synthesis is made for continuous-time but recursive identification runs on the δ-model, which belongs to the class of discrete-time models. This simplification can be introduced with the assumption that the parameters of the δ-model are close to the continuous one, i.e. $a \approx a^\delta$ and $b \approx b^\delta$.

4. Simulation Results

The usability of the proposed controller from the previous part was examined by simulations on the mathematical model of the reactor from the chapter 2. All simulations were done on the mathematical software MATLAB. Common values for all simulations are the sampling

period $T_v = 0.3$ min, the initial vector of parameters used for identification $\theta_\delta^T(0) = [0.1 \ 0.1 \ 0.1 \ 0.1]$ and the initial covariance matrix $\mathbf{P}_{ii}(0) = 1 \cdot 10^7$ for $i = 1,..,4$. The input variable was limited in the interval $u(t) = \pm 80\%$ due to physical reasons. The change of the input volumetric flow rate of the reactant u_2 was chosen as a control input and the temperature of the reactant as a controlled output, i.e. $y(t) = T(t)$.

The first simulation analysis examines behavior of the controlled output for the changing weighting factor φ_{LQ} from (22) - $\varphi_{LQ} = 0.1$, 1 and 3. On the other hand, the second parameter was $\mu_{LQ} = 1$ for all simulations.
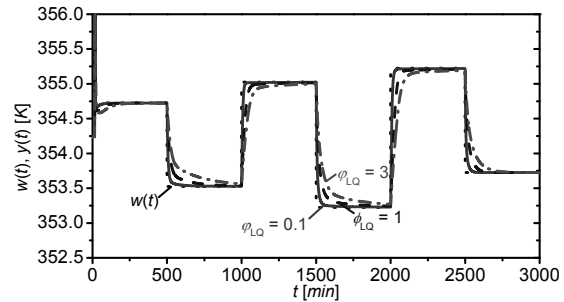

Figure 10. The course of the output variable $y(t)$ and the reference signal $w(t)$ for various values of the factor φ_{LQ}
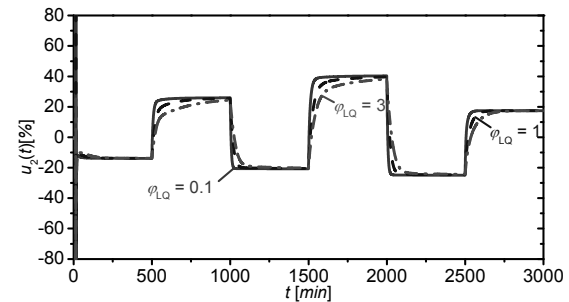

Figure 11. The course of the input variable $u(t)$ for various values of the factor φ_{LQ}

Results presented in Figure 10 and Figure 11 clearly show effect of the weighting factor φ_{LQ} – decreasing value of this parameter results in the quicker output response but also very quick changes of the input variable. This change could cause some problems from the practical point of view where this value represents for example twist of the valve in the input pipe to the reactor and quick and often changes could destroy the valve. Anyway, the controlled plant produced very good control results except very beginning of the control. This inaccurate course was caused by the recursive identification which did not have enough information about the system behaviour at the beginning – note

that all simulations starts from the general values of $\theta_\delta^T(0)$ and $P_{ii}(0)$ and controller needs some time for adapting to the system.
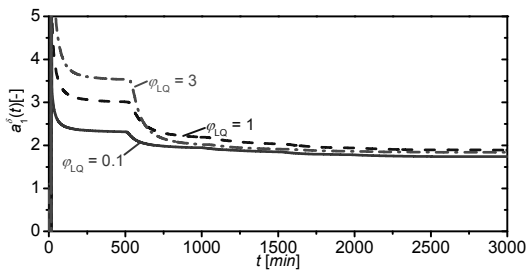

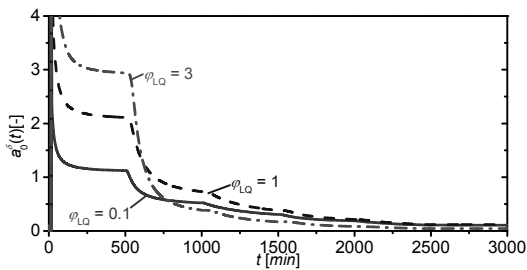**Figure 12.** The course of the identified parameter a_1^δ during the control

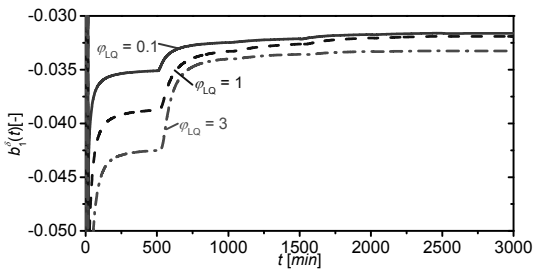**Figure 13.** The course of the identified parameter a_0^δ during the control

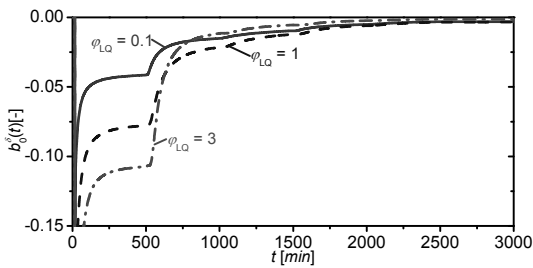**Figure 14.** The course of the identified parameter b_1^δ during the control

Figure 15. The course of the identified parameter b_0^δ during the control

This is illustrated also in Figure 12 – Figure 15 where courses of identified parameters are displayed. This adaptation phase takes about 50 *min* and the course after this initial time is again very smooth which mirrors in the course of the output variable too. Results also show fulfillment of first two control conditions – the stability of the control loop and the reference signal tracking. The third condition,

disturbance attenuation, was tested in the second analysis for three disturbances – two of them were inserted on the input to the system (change of the input reactant temperature T_0 and input concentration c_{A0}) and one disturbance on the output from the reactor (change of the temperature T), i.e. $v_1(t) = -8\%$ step change of the input concentration c_{A0} for time $t = <400; 1000>$ *min*, $v_2(t) = +1\ K$ step change of the input reactant's temperature T_0 for time $t = <600; 1000>$ *min*, $v_2(t) = -0.5\ K$ step change of the output product's temperature T for time $t = <800; 1000>$ *min*.

The weighting factor was $\varphi_{LQ} = 0.25$ and one step change of the reference signal, $w(t)$, was done before disturbance injection due to inaccurate results caused by the recursive identification at the beginning of the control.
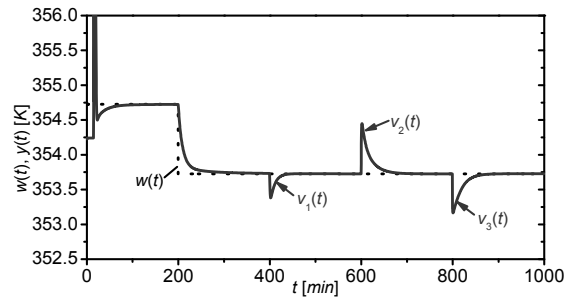

Figure 16. The course of the output variable $y(t)$ and the reference signal $w(t)$ for the simulation with three disturbances
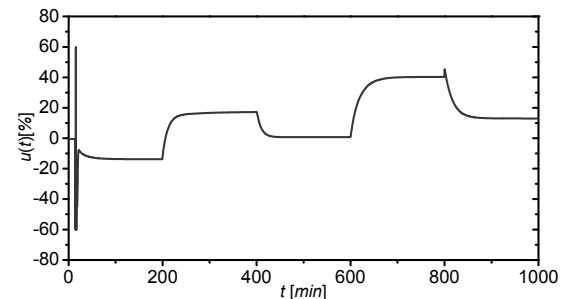

Figure 17. The course of the input variable $u(t)$ for the simulation with three disturbances

It is clear from Figures 16 and 17 that proposed controller copes with these disturbances sufficiently even though the reactor is affected by all three disturbances from time 800 min to the end of the simulation.

5. Verification of the Simulation

As it is written at the beginning, all results presented here are results of computer simulations. The problem with the simulation is that it is always question of the reliability if the

proposed mathematical model provides sufficient description of the real system or it is too simple. This could be proofed only by the verification on the real system.

The advantage of the control strategy presented here is that not only simulation of the mathematical model's behaviour, but also the computation of controller's parameters are easily programmable. Both blocks in Figure 9 representing the controlled plant (block *G*) and the feedback controller (*Q*) form different parts of the M-file in MATLAB which makes the controller more universal. The part of the simulation program which represents mathematical model of the system (*G*) could be replaced by the MATLAB's Real-Time toolbox routines which reads output from the controlled system, generally *y*(*t*), and send signals to the system according computed output from the controller *u*(*t*).

One example can be found in our previous article [19], where we use the same controller for conductivity control inside the real model of CSTR as a part of the Multifunctional Process Control Teaching equipment PCT40 from Armfield – see schematic representation of PCT40 in Figure 18.
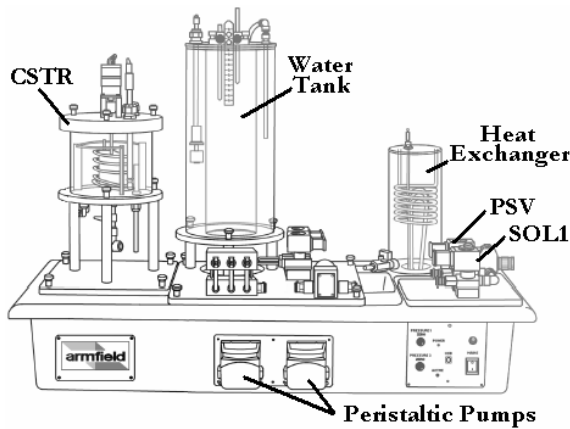

Figure 18. Multifunctional process control teaching system PCT40

This system is connected to the computer via two technological MF624 multifunction I/O cards from Humusoft. We can use 9 inputs and 17 outputs at the same time in the MATLAB through the Real-Time toolbox.

The control experiment uses the same 1DOF control scheme and LQ controller presented here and the sample results for various values of the weighting factor φ_{LQ} = 0.005, 0.001 and 0.01 are shown in following graphs.
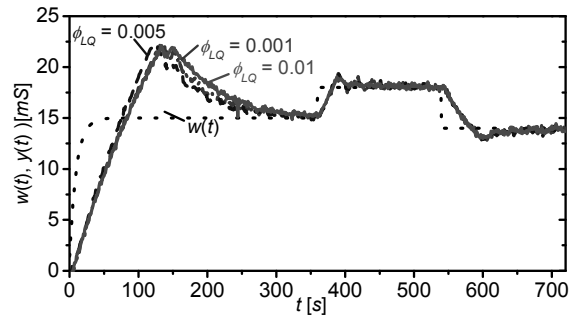

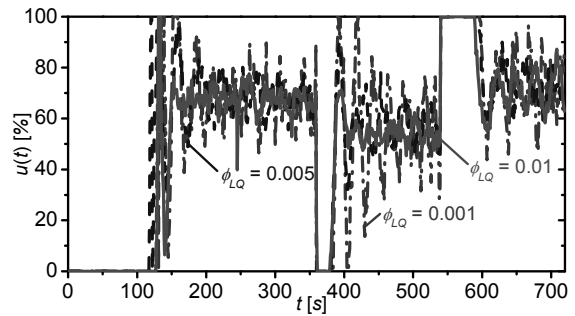**Figure 19.** Course of the output variable *y*(*t*) for 1DOF controller and LQ controller

Figure 20. Course of the input variable *u*(*t*) for 1DOF controller and LQ controller

Results in Figure 19 and Figure 20 have shown usability of this control strategy for different nonlinear systems. The only difference is in the settings of the controller. The setting, or let us say tuning, parameter here is weighting factor φ_{LQ} which is different for each control strategy and must be found by the experiments.

6. Conclusions and Future Work

The article presents simulation results of hybrid adaptive controller which uses polynomial synthesis together with delta-models and LQ approach. Results satisfy basic control requirements – the stability, the reference signal tracking and the disturbance attenuation. The controller could be also tuned via the choice of the weighting factor φ_{LQ} while the increasing value of this parameter results in the quicker output response. The only problem can be found at the beginning of the control when the recursive identification needs some adaptation time due to incomplete a priori information about the system. On the other hand, the controller has no problems at the rest of the control after this initial time even if it is affected by various disturbances.

Although results are products of the computer simulation, the proposed hybrid adaptive controller could be used also for other similar

nonlinear processes as it was shown in the last part before conclusion which presents results of the conductivity control inside the CSTR with the use of the same adaptive LQ controller.

The future work is related to the simulation experiments of other let us say "modern" control techniques such as predictive, robust or nonlinear control and verifications by the application of these controllers on the real process.

REFERENCES

1. INGHAM, J., I. J. DUNN, E. HEINZLE, J. E. PRENOSIL, **Chemical Engineering Dynamics. An Introduction to Modelling and Computer Simulation**. Second, Completely Revised Edition, VCH Verlagsgesellshaft, Weinheim, 2000.

2. LUYBEN, W. L., **Process Modelling, Simulation and Control for Chemical Engineers.** McGraw-Hill, New York, 1989.

3. LEBRUN, M., D. VASILIU, N. VASILIU, **Numerical Simulation of the Fluid Control Systems by AMESim**. Studies in Informatics and Control, vol. 18(2), 2009, pp. 111-118.

4. CHORUKOVA, E., I. SIMEONOV, **Neural and Hybrid Modelling of Biotechnological Process**, Studies in Informatics and Control, vol. 17(3), 2008, pp. 305-31.

5. CORRIOU, J.-P., **Process Control. Theory and Applications.** Springer-Verlag London. 2004.

6. SCHMIDT, L. D., **The Engineering of Chemical Reactions**. Oxford University Press, 1997.

7. GAO, R., A. O'DYWER, E. COYLE, **A Non-linear PID Controller for CSTR Using Local Model Networks**. Proceedings of the 4th World Congress on Intelligent Control and Automation. Shanghai, China, 2002, pp. 3278-3282.

8. GAZDOS, F., L. MACKU, **Control-Oriented Simulation Analysis of a Semi-Batch Reactor**, International Review of Automatic Control, vol. 2(5), 2009, pp. 584-591. ISSN 1974-6059.

9. MACKU, L., D. SAMEK, **Two Step, PID and Model Predictive Control using Artificial Neural Network Applied on**

Semi-batch Reactor, WSEAS Transactions on Systems, vol. 9(10), 2010, ISSN 1109-2777.

10. JOHNSTON, R. L., **Numerical Methods.** John Wiley & Sons, 1982.

11. MATHEWS, J. H., K. K. FINK, **Numerical Methods Using Matlab**, Prentice-Hall, 2004.

12. ASTROM, K. J., WITTENMARK, B., **Adaptive Control**, 2nd edition, Prentice Hall, 1994.

13. BOBAL, V., J. BOHM, J. FESSL, J. MACHACEK, **Digital Self-tuning Controllers. Algorithms, Implementation and Applications**, Springer, 2005.

14. KUCERA, V., **Diophantine Equations in Control – A Survey**. Automatica, vol. 29, 1993, ISSN 1361-1375.

15. HUNT, K. J., V. KUCERA, M. SEBEK, **Optimal Regulation using Measurement Feedback. A Polynomial Approach**. IEEE Transactions on Automation Control, vol. 37(5), pp. 682-685.

16. KUCERA, V., **Analysis and Design of Discrete Linear Control Systems**. Prentice-Hall, London 1991.

17. STERICKER, D. L., N. K. SINHA, **Identification of Continuous-time Systems from Samples of Input-Output Data using the δ-operator**. Control-Theory and Advanced Technology, vol. 9, 1993, pp. 113-125.

18. MUKHOPADHYAY, S., A. G. PATRA, G. P. RAO, **New Class of Discrete-time Models for Continuous-time Systems**. International Journal of Control, vol. 55, pp. 1161-1187.

19. VOJTESEK, J., P. DOSTAL, **Adaptive Control of Continuous-Stirred Tank Reactor in Two Stable Steady-States**, in Proceedings of the IFAC Workshop Adaptation and Learning in Control and Signal Processing (ALCOSP 2010), Antalya, Turkey, 2010.

20. VOJTESEK, J., P. DOSTAL, **Adaptive LQ Approach Used in Conductivity Control inside Continuous-Stirred Tank Reactor**, in Proceedings of the 17th IFAC World Congress, Seoul, 2008, pp. 12929-12934.