Polynomial Approach to Digital Self-Tuning Control: Industrially Motivated Application

RADEK MATUŠŮ, ROMAN PROKOP
Department of Automation and Control Engineering
Faculty of Applied Informatics
Tomas Bata University in Zlín
nám. T. G. Masaryka 5555, 76001 Zlín
CZECH REPUBLIC
{rmatusu; prokop}@fai.utb.cz

Abstract: This contribution is focused mainly on presenting the software implementation of selected digital self-tuning control algorithms into the environment of Matlab and Pascal. The applied methods comprise a polynomial approach to discrete-time control design and recursive least-squares identification algorithm LDDIF. The motivation and basic conditions of the application have been based on real technical assignment of a manufacturer of aluminium-based rolled products and packaging materials.

Key-Words: Self-Tuning Controllers, Digital Control, Polynomial Approach, Software Implementation

1 Introduction
Real control of industrial processes is almost always burdened with various disturbances, changes in process parameters or dynamics due to varying operational conditions, plant properties, etc. In such situations, an acceptable a priori mathematical model does not have to be known. In spite of it, such processes have to be controlled.

A possible solution to this task represents an area of control theory known as adaptive control or more specifically usage of self-tuning controllers [1]-[9]. The basic idea consists in modification of control law according to the changing plant parameters obtained via recursive identification. Its advantage is some kind of “intelligent” behaviour, but on the other hand these regulators are quite complex and not easily applicable.

This contribution deals mainly with software implementation of selected digital self-tuning control algorithms into the Matlab and Pascal environment for the purpose of possible industrial utilization. The work was motivated by cooperation with a manufacturer of aluminium-based rolled products and packaging materials. His project has supposed primarily the application of discrete-time adaptive compensator to control of a metal smelting furnace. Other requirements were the plant model with “a2b3” structure and final implementation in Borland Pascal (because of integration into the existing system). However the paper presents not only derived relations applicable to Pascal environment but also program for simulative purposes and testing created under Matlab and some preliminary simulation results. Even though the task was motivated by the specific problem, the contribution tries to present it in more or less generally applicable way.

The previous versions of this work have been presented at conferences [10], [11] and as the chapter in the book [12].

2 Polynomial Synthesis in Discrete-Time Domain
The basic principle of applied self-tuning control scenario consists in consecutive identification of the controlled process using a recursive algorithm (see the following section) and application of obtained plant parameters in computing the control law. The control design itself has been based on algebraic approach and pole placement [13]-[16].

In spite of the existence of more complex control configurations, only the very basic single-input single-output (SISO) control loop with one degree of freedom has been assumed. This classical feedback connection in a discrete-time sense is shown in fig. 1.

![Fig. 1: Discrete-time feedback control loop](image)

The signals \( w(k) \), \( e(k) \), \( u(k) \) and \( y(k) \) from fig. 1 represent reference value, tracking (control) error, actuating (manipulated) signal and controlled (output) variable, respectively, and blocks \( C(z^{-1}) \) and \( G(z^{-1}) \).
mean discrete-time transfer functions of a controller and controlled system.

According to project requirements a controlled plant is supposed to has an “a_2b_3” structure, i.e. its transfer function is:

\[ G(z^{-1}) = \frac{b(z^{-1})}{a(z^{-1})} = \frac{b_1z^{-1} + b_2z^{-2} + b_3z^{-3}}{1 + a_1z^{-1} + a_2z^{-2}} \] (1)

A suitable controller which ensures stabilization of the whole control loop (fig. 1) and asymptotic tracking of stepwise reference variable can be obtained by solution of Diophantine equation [13], [14]:

\[ a(z^{-1})f(z^{-1})p(z^{-1}) + b(z^{-1})q(z^{-1}) = m(z^{-1}) \] (2)

where \( a(z^{-1}), b(z^{-1}) \) are from the controlled system (1), and \( p(z^{-1}), q(z^{-1}) \) from discrete-time-controller:

\[ C(z^{-1}) = \frac{q(z^{-1})}{f(z^{-1})p(z^{-1})} = \frac{q_0 + q_1z^{-1} + q_2z^{-2}}{f(z^{-1})(p_0 + p_1z^{-1} + p_2z^{-2})} \] (3)

and where \( f(z^{-1}) \) is the denominator of image of stepwise reference signal:

\[ W(z^{-1}) = \frac{b(z^{-1})}{f(z^{-1})} = \frac{b(z^{-1})}{1 - z^{-1}} \] (4)

Moreover, right-hand polynomial \( m(z^{-1}) \) from (2) is a stable polynomial of appropriate order. Thus the equation (2) takes here the specific form:

\[
\begin{align*}
(1 + a_1z^{-1} + a_2z^{-2})(1 - z^{-1})(p_0 + p_1z^{-1} + p_2z^{-2}) + \\
\cdots (b_1z^{-1} + b_2z^{-2} + b_3z^{-3})(q_0 + q_1z^{-1} + q_2z^{-2}) = \\
m_0 + m_1z^{-1} + m_2z^{-2} + m_3z^{-3} + m_4z^{-4} + m_5z^{-5}
\end{align*}
\] (5)

The aim is to calculate coefficients of \( p(z^{-1}), q(z^{-1}) \) to get the controller (3). A simple method for finding the particular solution of Diophantine equation (5) grounds in the comparison of coefficients with the same power and consequent transformation of (5) into the set of six equations with six unknowns. This set can be written in a matrix form as follows:

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
a_1 & -1 & 1 & 0 & b_1 & 0 \\
a_2 - a_1 & -1 & 1 & b_2 & b_1 & 0 \\
-a_2 & a_2 - a_1 & a_1 - 1 & b_3 & b_2 & b_1 \\
0 & -a_2 & a_2 - a_1 & 0 & b_3 & b_2 \\
0 & 0 & -a_2 & 0 & 0 & b_3
\end{bmatrix}
\begin{bmatrix}
p_0 \\
p_1 \\
p_2 \\
q_0 \\
q_1 \\
q_2
\end{bmatrix}
= \begin{bmatrix}
m_0 \\
m_1 \\
m_2 \\
m_3 \\
m_4 \\
m_5
\end{bmatrix}
\] (6)

Solving the equation system (6) would be an easy task in many software packages. However, the final implementation of control algorithm in Borland Pascal environment was required by assignment and so the analytical solution of (6) had to be derived in order to be easily programmable. Thus, the utilizable controller parameters are computed according to:

\[
\begin{align*}
g_0 &= x_{13}/x_{12} \\
g_1 &= x_{14}/x_{12} \\
g_2 &= x_{15}/x_{12} \\
p_0 &= m_0 \\
p_1 &= x_2 - b_1q_0 \\
p_2 &= (q_2b_3 - m_5)/a_2
\end{align*}
\] (7)

where auxiliary variables are:

\[
\begin{align*}
x_1 &= (a_2 - a_1)b_2 + b_2a_2 \\
x_2 &= m_0 + (1 - a_1)m_0 \\
x_3 &= [a_1m_4 + a_2^2x_2 + (a_2 - a_1)m_1]/x_1 \\
x_4 &= (a_2b_2)/x_1 \\
x_5 &= (a_2b_3)/x_1 \\
x_6 &= m_0(a_2 - a_1) + x_2a_2 - x_2 \\
x_7 &= -b_1a_1 + b_3 + b_2 \\
x_8 &= m_2 - x_6 + m_5/a_2 \\
x_9 &= -b_1a_2 + b_3a_1 + b_1 \\
x_{10} &= (b_1a_2)/a_2 - b_3/a_2 + b_1 \\
x_{11} &= m_1 + m_1a_2 - h_2a_2 + b_2a_1 + (m_1a_1)/a_2 - m_5/a_2 \\
x_{12} &= x_4b_4x_{10} + x_2x_{12} + x_3x_3(b_1/a_2) - h_2x_9 - \\
\cdots & (b_3/a_2)b_2x_4 - x_{10}x_5x_7 \\
x_{13} &= x_4b_1x_{10} + x_2b_2 + x_1x_5(b_1/a_2) - b_1x_{11} - \\
\cdots & (b_3/a_2)b_2x_4 - x_{10}x_5x_7 \\
x_{14} &= x_4b_4x_{10} + x_3x_{11} + x_3x_5(b_1/a_2) - x_5x_9 - \\
\cdots & (b_3/a_2)x_{11}x_4 - x_{10}x_3x_7 \\
x_{15} &= x_2b_1x_{11} + x_2b_2x_3 + x_9x_3x_9 - x_3b_1x_9 - \\
\cdots & x_3b_1x_4 - x_3x_7
\end{align*}
\] (8)

The coefficients of \( m(z^{-1}) \) can be used for controller tuning and thus for influencing the closed-loop control behaviour. The suitable choice of the roots of the closed-loop characteristic polynomial \( m(z^{-1}) \) is known as pole placement problem. Anyway, this case of fifth order \( m(z^{-1}) \) can be easily “degraded” to the lower order ones by equalling the appropriate coefficients to zero. The special events are represented by dead-beat control for \( m(z^{-1}) = 1 \) or by linear quadratic (LQ) control for \( m(z^{-1}) \) given by means of minimizing the LQ criterion [8], [9].
Finally, the calculated parameters (7) are applied to programmable control law which corresponds to the controller (3) and which generates the control signal $u(k)$. It can be formulated as:

$$u(k) = \left( p_0 - p_1 \right) u(k-1) + \cdots + \left( p_1 - p_2 \right) u(k-2) + \cdots + p_2 u(k-3) + q_0 e(k) + \cdots + q_2 e(k-1) + q_3 e(k-2) \right) / p_0 \quad (9)$$

3 Recursive Identification Algorithm

A LDDIF routine has been used as plant parameters identification technique for combination with algebraic synthesis from the previous section in order to obtain self-tuning controller. It is recursive least-squares algorithm with exponential and directional forgetting [17]. Moreover, the corrections influencing the covariance matrix $P(k)$ of the estimated parameters by adding some multiple of identity matrix, which have been suggested in [18], are implemented to improve the tracking performance. The algorithm can be described by equations [19]:

$$\varepsilon(k) = y(k) - \Phi(k)^T \theta(k-1)$$

$$r(k) = \Phi(k)^T P(k-1) \Phi(k)$$

$$\kappa(k) = P(k-1) \Phi(k) / 1 + r(k)$$

$$\beta(k) = \begin{cases} \varphi - \frac{1 - \varphi}{r(k)} & \text{if } r(k) > 0 \\ 1 & \text{if } r(k) > 0 \end{cases}$$

$$P(k) = P(k-1) - \frac{P(k-1) \Phi(k) \Phi(k)^T P(k-1)}{\beta(k)^{-1} + r(k)} + \delta I$$

$$\theta(k) = \theta(k-1) + \kappa(k) \varepsilon(k)$$

where $\Phi(k) = [\gamma y(k-1) - y(k-2) u(k-1) u(k-2) u(k-3)]$ is observation vector, $\theta(k) = [a_1(k) a_2(k) b_1(k) b_2(k) b_3(k)]$ is vector of parameters and $\varphi$ is exponential forgetting. The initial values for the algorithm are usually preset to $\varphi = 0.985$, $P(0) = 10^6 I$ and $\delta = 0.01$.

The main complication from the implementation viewpoint has been the arduousness in working with matrices.

4 Software Implementation

As it was outlined before, Borland Pascal had to be supposed for final application under real industrial conditions because of easy implementation into the existing system. However, several preliminary tests, algorithm verifications and simulations were done in Matlab environment due to better convenience for these testing purposes. As a result, a simple program has been created. Its main window is shown in fig. 2.

![Fig. 2: Main window of the preliminary simulation program in Matlab](image)

Initial supportive control and identification experiments for sampling time $T = 45s$ have led to parameters of controlled system (1):

$$a_1 = -1.04$$

$$a_2 = 0.139$$

$$b_1 = -0.327$$

$$b_2 = 1.079$$

$$b_3 = 0.763$$

The closed-loop characteristic polynomial has been supposed as:

$$m(z^{-1}) = 1 - 2.45z^{-1} + 2.22z^{-2} - 0.907z^{-3} + \cdots$$

$$-0.1601z^{-4} - 0.008925z^{-5}$$

which means that the poles of the closed loop transfer function (fig. 1) have been placed to:

$$r_1 = 0.85$$

$$r_2 = 0.7$$

$$r_3 = 0.5$$

$$r_4 = 0.3$$

$$r_5 = 0.1$$

Simulation result of control behaviour is depicted in fig. 3. The huge overshoot in the beginning of the process is caused by not completed identification stage. The parameters of the controlled system were assumed to be
unknown and preset to random starting values (as demonstrated in fig. 2). The progress in identification of these parameters during control is shown in fig. 4 with zoomed x-axis. As can be seen the plant parameters was properly identified after several initial steps and thanks to this the control response from fig. 3 is much better at the middle step change of reference signal.

![Fig. 3: Control of plant using discrete-time self-tuning controller – simulation](image1)

![Fig. 4: Development of the identified parameters](image2)

5 Conclusions

The contribution has been focused on preliminary software implementation of digital self-tuning controllers into the Matlab (for simulative and testing purposes) and Pascal (for real application) environment. The motivation to this task as well as basic conditions and restrictions have been based on technical assignment of a manufacturer of aluminium-based products related to control of a metal smelting furnace. The applied techniques have comprised a polynomial approach to discrete-time control design and recursive least-squares identification algorithm LDDIF.

Acknowledgements:

The work was supported by the Ministry of Education, Youth and Sports of the Czech Republic under the Research Plan No. MSM 7088352102 and by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089. This assistance is very gratefully acknowledged. Moreover, the authors would like to thank Jaromír Kovář for his intense and enthusiastic co-operation on the project connected with content of this contribution.

References:


