PPSA: A Tool for Suboptimal Control of Time Delay Systems – Revision and Open Tasks

Libor Pekař and Pavel Navrátil

Faculty of Applied Informatics, Tomas Bata University in Zlín, Czech Republic
{pekar, pnavratil}@fai.utb.cz

Abstract. During the development of algebraic controller design in a special ring for time delay systems (TDSs) a problem of a suitable free controller parameters setting appeared. The first author of this contribution recently suggested a natural idea of placing the dominant characteristic numbers (poles) and zeros of the infinite-dimensional feedback control system on the basis of the desired overshoot for a simple finite-dimensional matching model and shifting of the rest of the spectrum. However, the original procedure called the Pole-Placement Shifting based controller tuning Algorithm (PPSA) was not developed and described entirely well. The aim of this paper is to revise the idea of the PPSA and suggest a possible ways how to improve or extend the algorithm. A concise illustrative example is attached to clarify the procedure for the reader as well.

Keywords. Time delay systems, pole placement controller tuning, optimization, direct-search algorithms, evolutionary algorithms, SOMA, Nelder-Mead algorithm, Gradient Sampling Algorithm, model matching.

Introduction

Time delay systems (TDSs) constitute a huge class of processes and systems that are affected by any form of delay or latency, either in the input-output relation (as it is known in classical engineering problems) or inside the system dynamics (in this case notions of internal or state delays are introduced). The latter models and processes those are much more involved for analysis and control can be found in many theoretical and practical applications covering various fields of human activity, such as technology, informatics, biology, economy, etc., see e.g. [1-4].

A typical feature of TDSs is their infinite spectrum, due to transcendental nature of the characteristic equation, i.e. they have an infinite number of solution modes and corresponding system poles. This unpleasant attribute makes them difficult to analyze and design a control law as well. Linear time-invariant TDSs can be modeled and described by transfer functions by means of the Laplace trans-

form. In most cases, roots of the transfer function denominator coincide with system poles.

The ring of quasipolynomial meromorphic functions (R_{MS}), originally developed and introduced in [5] and revised and extended in [6], represents a possible tool for description and control design of TDSs. However, in many cases, namely, for unstable TDSs, the control algorithm must deal with also infinitely many feedback characteristic poles the positions of which depend on the selectable controller parameters. The use of pole-placement (pole-assignment, root-locus) tuning algorithms can be a possible way how to solve the setting problem, see e.g. [7-9]. However, these algorithms deal with poles only ignoring closed-loop zeros and/or they have been derived for state-space controllers.

The idea of the Pole-Placement Shifting based controller tuning Algorithm (PPSA) provides slightly different approach [10]. It is based on the analysis of a simple finite-dimensional model where the relative maximum overshoot, relative dumping and relative time-to-overshoot of the reference-to-output step response are calculated and serve as a control performance indicators. Then, according to the selected values, the desired positions of dominant (i.e. the rightmost) poles and zeros are calculated, and poles and zeros of the infinite-dimensional feedback system are shifted to the prescribed positions while the rest of the spectrum is pushed to the left (i.e. to the "stable" region). In some sense, it represents a matching problem. The initial solution (i.e. controller parameter setting) is obtained using the Quasi-Continuous Shifting Algorithm (QCSA) [7], [8] which is followed by the use of an advanced numerical optimization algorithm. The method was independently developed in [9]; however, there are some essential differences – the reader is referred e.g. to [10] for details.

However, the original algorithm was described neither precisely nor in details and it contains some shortcomings and errors. Thus, the aim of this contribution is to revise and consolidate the PPSA and raise some open tasks how to improve and accelerate the algorithm. In this connection, the reader is kindly asked to participate on the solution of these problems in the future if he or she is interested in them.

To make the procedure clearer (to the reader) a short illustrative example on the control of an unstable time delay system by means of Matlab-Simulink environment is provided.

Time Delay Systems – Introductory Description

Since the reader is supposed to be a non-expert in system and control theory and the description and control design of TDSs is not the primary topic of this contribution, only a very concise overview of TDS models is provided such that all necessary information are given him or her.

A possible formulation of a TDS model (either a plant or a delayed control feedback loop) can be done using the transfer function in a complex variable s as the direct consequence of the use of the Laplace transform as follows

$$G(s) = \frac{b(s)}{a(s)} \tag{1}$$

where $a(s)$, $b(s)$ are quasipolynomials of a general form

$$x(s) = s^n + \sum_{i=0}^{n} \sum_{j=1}^{h_i} x_{ij} s^i \exp(-s\eta_{ij}), \ \eta_{ij} \geq 0, x_{ij} \in \mathbb{R} \tag{2}$$

where η_{ij} express delays and $\mathbb{R}$ means the set of real numbers. If delays are included only in the numerator $b(s)$, they influence the input-output relation; in the contrary, the system contain internal delays and equation $a(s) = 0$ has infinitely many solution. These solution values constitute (in overwhelming majority of cases) system poles, more precisely, poles $s_i, i = 1,2,...$ are singularities of $G(s)$ satisfying

$$\lim_{s \to s_i} G(s) = \pm\infty; \exists n_0, \forall n \geq n_0 : \lim_{s \to s_i} (s - s_0)^n G(s) < \infty \tag{3}$$

Zeros have the same meaning as in (3) yet for $1/G(s)$ instead of $G(s)$, i.e. they coincide with the roots of $b(s)$ (in most cases).

Problem Formulation

Now consider that $G(s)$ means the control feedback transfer function. Some control design approaches yield this function with the denominator containing delays along with free real controller parameters from the set $\mathbf{K} = \{k_1, k_2, ..., k_r\} \neq \emptyset \in \mathbb{R}^n$. This results in the infinite-dimensional (delayed) control feedback. Naturally, the numerator can own delays (and controller parameters) as well.

The idea of the PPSA is to match some number of the rightmost (i.e. the dominant) poles and zeros of $G(s)$ with all poles and zeros of a finite-dimensional model $G_m(s)$. Thus the selected poles and zeros of $G(s)$ are quasi-continuously shifted to the desired positions by small steps and the rest of both spectra (of poles and zeros) try to push to the left (i.e. to the stable complex semiplane) as far as possible. The shifting can be done e.g. using the QCSA or via an advanced algorithm, [11-13], minimizing a suitable cost function reflecting the distance of

dominant poles from prescribed positions and the spectral abscissa (i.e. the value of the real part of the rightmost pole/zero). By doing this, the values of $\mathbf{K}$ are being adjusted and hence the controller parameters are being tuned.

A crucial problem is to choose a suitable number of prescribed poles and zeros, i.e. degrees of the numerator, $N(s)$, and denominator, $D(s)$, of $G_m(s)$. Let us denote the numerator and the denominator as $N(s, \mathbf{K}_N)$ and $D(s, \mathbf{K}_D)$, respectively, where $\mathbf{K}_N$ and $\mathbf{K}_D$ mean free real parameters of the numerator and denominator, respectively, with $r_N = |\mathbf{K}_N| \geq 0$, $r_D = |\mathbf{K}_D| > 0$. It is initially assumed that equations $N(s_i, \mathbf{K}_N) = 0$, $D(s_j, \mathbf{K}_D) = 0$ are independent for arbitrary yet fixed s_i, s_j with $i = 1, 2, \ldots n_N \leq r_N$, $j = 1, 2, \ldots n_D \leq r_D$, that is

$$
\begin{aligned}
&\text{rank}\left[\frac{\partial}{\partial k_{N,l}} N(s_i, \mathbf{K}_N) \right]_{\substack{i=1,2,\ldots n_N \\ l=1,2,\ldots r_N}} = n_N \\
&\text{rank}\left[\frac{\partial}{\partial k_{D,l}} D(s_j, \mathbf{K}_D) \right]_{\substack{j=1,2,\ldots n_D \\ l=1,2,\ldots r_D}} = n_D
\end{aligned}
\tag{4}
$$

Then the following conditions must hold: As indicated above, the number of prescribed poles, n_D, and zeros, n_N, must be less or equal to the number of corresponding free parameters to obtain a solvable matching problem. Moreover, if one needs to enable shifting the rest of the spectrum to the left, some parameters might not be bounded with desired position of roots, hence

$$
0 \leq n_D < r_D, 0 < n_N < r_N
\tag{5}
$$

where $\Delta n_D = r_D - n_D, \Delta n_N = r_N - n_N$ serve for adjusting the rightmost real parts of the rest of spectra. Naturally, the number of all desired solutions can not exceed the number of all free parameters, which gives rise to

$$
n_D + n_N < r
\tag{6}
$$

In addition, the model has to be strictly proper, i.e.

$$
n_D < n_N
\tag{7}
$$

Conditions (5)-(7) ought to be taken into account when designing the finite-dimensional model.

PPSA Strategies

Three possible revised modifications of the PPSA follows. A thorough algorithm description is consequently supported by its vague explanation and discussion in all three cases. Let is use these notations in the algorithms: $\mathbf{K} = \mathbf{K}_N \cup \mathbf{K}_D$ where numerator coefficients of $G(s)$ read $\mathbf{K}_N = \mathbf{K}_{N\backslash D} \cup \mathbf{K}_{\overline{ND}}$ with $r_{\overline{ND}} = \left|\mathbf{K}_{\overline{ND}}\right| = \left|\mathbf{K}_N \cap \mathbf{K}_D\right|$, $r_{N\backslash D} = \left|\mathbf{K}_{N\backslash D}\right| = \left|\mathbf{K}_N \backslash \mathbf{K}_{\overline{ND}}\right|$, whereas denominator ones analogously are $\mathbf{K}_D = \mathbf{K}_{D\backslash N} \cup \mathbf{K}_{\overline{ND}}$ with $r_{D\backslash N} = \left|\mathbf{K}_{D\backslash N}\right| = \left|\mathbf{K}_D \backslash \mathbf{K}_{\overline{ND}}\right|$. Simply, $r_N = r_{N\backslash D} + r_{\overline{ND}}$, $r_D = r_{D\backslash N} + r_{\overline{ND}}$.

Algorithm 1 (PPSA strategy 1: "Poles First Independently")
Input. Closed-loop reference-to-output transfer function $G(s)$ with $r_{N\backslash D} > 0$.
Step 1. Set $n_D = r_D - 1$, thus $\Delta n_D = r_D - n_D = 1$. (Or just select $n_D < r_D$ as high as desirable).
Step 2. Verify that there can exist a non-negative number n_N satisfying

$$0 \le n_N < \min\{n_D, r_{N\backslash D}\} \tag{8}$$

If (8) holds, fix n_N and go to Step 3; otherwise, set $n_D = n_D + 1$. If $n_D < \min\{r_D, r_{N\backslash D} + 1\}$, i.e. $n_D < r_D$ and $n_D \le r_{N\backslash D}$, go to Step 2, else terminate the procedure (a solution does not exist).
Step 3. Choose a simple matching model of a stable finite-dimensional system with the numerator of degree n_N, the denominator of degree n_D and the unit static gain governed by the transfer function $G_m(s)$. The model can be prescribed e.g. according to the desired dynamic behavior of the feedback loop. Its poles and zeros are referred as "prescribed" below.
Step 4. Set a part of the spectrum of poles via the number n_D of coefficients from the set $\mathbf{K}_D$ into the prescribed positions while the rest of denominator parameters are chosen arbitrarily. If these poles are dominant, initialize the counter of currently shifted poles as $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D + 1$ where $n_{sp,m} = n_D$ and $n_{sp,opt} = 1$. If not, then $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D$, $n_{sp,m} = n_D$, $n_{sp,opt} = 0$.
Step 5. Check that (4) holds for the number n_{sp} of the rightmost poles and $\mathbf{K}_D$. If not, go to Step 4 and reset the initial assignment; otherwise, shift the number $n_{sp,m}$ of the rightmost feedback system poles towards the prescribed locations (i.e., keep in the close proximity of them), e.g. using the QCSA, whereas the number $n_{sp,opt}$ of poles is pushed to the left. If necessary, increase $n_{sp,opt} \Rightarrow n_{sp}$. If $n_{sp} = r_D$ and/or the shifting is no more successful, go to Step 6.

Step 6. If all $n_{sp,m}$ poles are dominant, go to Step 7. Otherwise, select a suitable cost function $\Phi_P(\mathbf{K}_D)$ reflecting the distance of dominant poles of $G(s)$ from prescribed positions and the spectral abscissa. Minimize $\Phi_P(\mathbf{K}_D)$ starting with results from Step 5 (using e.g. an advanced iterative algorithm, [11-13]). Fix $\mathbf{K}_D$.

Step 7. Place a part of the spectrum of zeros of $G(s)$ using the number of n_N coefficients from the set $\mathbf{K}_{N\setminus D}$ into the prescribed positions and the remaining parameters in $\mathbf{K}_{N\setminus D}$ are chosen arbitrarily. If these zeros are dominant, initialize the counter of currently shifted zeros as $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N + 1$ where $n_{sz,m} = n_N$ and $n_{sz,opt} = 1$; otherwise, set $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N$, $n_{sz,m} = n_N$, $n_{sz,opt} = 0$.

Step 8. Check that (4) holds for the number n_{sz} of the rightmost zeros of $G(s)$ and for current values of $\mathbf{K}_{N\setminus D}$. If it is approved, $n_{sz,m}$ zeros are to be incessantly moved to the prescribed positions whereas $n_{sz,opt}$ zeros are pushed to the left. If necessary, increase $n_{sz,opt} \Rightarrow n_{sz}$. If $n_{sz} = r_{N\setminus D}$ and/or the shifting is no more successful, go to Step 9.

Step 9. If all $n_{sz,m}$ zeros are dominant, the algorithm is finished. Otherwise, select a suitable cost function $\Phi_Z(\mathbf{K}_{N\setminus D})$ reflecting the distance of dominant zeros of $G(s)$ from prescribed positions and the spectral abscissa. Minimize $\Phi_Z(\mathbf{K}_{N\setminus D})$ with initial setting of $\mathbf{K}_{N\setminus D}$ obtained from Step 8.

Output. The vector of controller parameters $\mathbf{K} = \mathbf{K}_{N\setminus D} \cup \mathbf{K}_D$, positions of the rightmost poles and zeros and the spectral abscissae.

The above presented strategy of the PPSA places the feedback poles to the desired positions first, and consequently, transfer function numerator parameters not included in the numerator serve as tuning tool for inserting zeros to the desired loci. Thus, zeros are placed independently from poles by means of $\mathbf{K}_{N\setminus D}$. In both the cases, the rest of the spectrum is pushed to the left as far as possible to minimize the spectral abscissa. If this quasi-continuous shifting is not successful, a trade-off between the zeros/poles matching task and the spectral abscissa is optimized. Note that condition (8) stem from (5) and (7) while (6) always holds for this strategy.

In fact, the QCSA or a shifting technique presented in [14] enables to shift a conjugate pair of roots along the real axis using a single controller parameter, i.e. it is possible to write $n_{sp,m} + n_{sp,opt,R} + n_{sp,opt,C} \le r_D$ and $n_{sz,m} + n_{sz,opt,R} + n_{sz,opt,C} \le r_{N\setminus D}$ where a subscript *R* denotes real roots whereas *C* means complex conjugate pairs.

If $r_{D\setminus N} > 0$, it is possible to apply the strategy reversely, i.e. to set zeros first and, afterwards, to place poles. However, the presented variant prefers poles since they affect the system dynamics more significantly.

Let us present now another (a simpler) strategy combining both, the poles and zeros matching, under one procedure.

Algorithm 2 (PPSA strategy 2: "Poles and Zeros Together")

Input. Closed-loop reference-to-output transfer function $G(s)$.

Step 1. Set $n_D = r_D - 1$, or just select $n_D < r_D$ as high as desirable.

Step 2. Verify that there exists a non-negative number n_N satisfying

$$0 \le n_N < \min(n_D, r - n_D, r_N) \tag{9}$$

If (9) holds, fix n_N and go to Step 3; otherwise, set $n_D = n_D - 1$. If $r_D > n_D \ge \max\{r - n_D, r_N\}$, go to Step 2; contrariwise, a solution does not exist.

Step 3. Choose a simple model $G_m(s)$ of a stable finite-dimensional system with the numerator of degree n_N, the denominator of degree n_D, the unit static gain and prescribed (desired) zeros and poles.

Step 4. Set finite subsets of both the spectra, poles and zeros, via the number n_D of coefficients from the set $\mathbf{K}_D$ and by means the number n_N of coefficients from the set $\mathbf{K}_N$, respectively, into the prescribed positions of $G_m(s)$ while the rest of parameters from $\mathbf{K}$ are chosen arbitrarily. If all these poles are dominant, initialize the counter of currently shifted poles as $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D + 1$ where $n_{sp,m} = n_D$ and $n_{sp,opt} = 1$; otherwise, $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D$, $n_{sp,m} = n_D$, $n_{sp,opt} = 0$. Similarly for zeros, if they are the rightmost ones, set $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N + 1$, $n_{sz,m} = n_N$, $n_{sz,opt} = 1$; in the contrary, $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N$, $n_{sz,m} = n_N$, $n_{sz,opt} = 0$.

Step 5. Check that (4) holds for the number n_{sp} of the rightmost poles and $\mathbf{K}_D$, and for n_{sz} dominant zeros along with $\mathbf{K}_N$. If not, go to Step 4 and reset the initial assignment; otherwise, shift mutually the number $n_{sp,m}$ and $n_{sz,m}$ rightmost feedback system poles and zeros, respectively, towards the prescribed locations the number $n_{sp,opt}$ and $n_{sz,opt}$ of poles and zeros, respectively, is pushed to the left along the real axis. If necessary, increase $n_{sp,opt} \Rightarrow n_{sp}$ and/or $n_{sz,opt} \Rightarrow n_{sz}$. If $n_{sp,m} + n_{sp,opt,R} + n_{sp,opt,C} \le r_D$ and $n_{sz,m} + n_{sz,opt,R} + n_{sz,opt,C} \le r_N$ and $n_{sz,m} + n_{sp,m} + n_{sp,opt,C} + n_{sz,opt,R} + n_{sz,opt,C} \le r$, or the shifting is no more successful, go to Step 6.

Step 6. If all $n_{sp,m}$ poles and $n_{sp,z}$ zeros are dominant, the procedure is finished. Otherwise, select a suitable cost function $\Phi(\mathbf{K})$ reflecting the distance of dominant poles and zeros of $G(s)$ from prescribed positions and spectral abscissae of both the spectra. Minimize $\Phi(\mathbf{K})$ starting with results from Step 5.

Output. The vector of controller parameters $\mathbf{K}$, positions of the rightmost poles and zeros and the spectral abscissae.

The methodology is useful in case $r_{N\backslash D} = 0$ (and/or $r_{D\backslash N} = 0$). Roughly speaking to summarize it, poles and zeros are moved simultaneously over a common set $\mathbf{K}$ of adjustable parameters, therefore their positions are not independent to each other.

A trade-off between Algorithm 1 and Algorithm 2 can be done by a procedure when only a subset $\mathbf{K}_{\overline{ND},D} \subset \mathbf{K}_{\overline{ND}}$ is dedicated to poles while a subset $\mathbf{K}_{\overline{ND},N} \subset \mathbf{K}_{\overline{ND}}$ is given to zeros to be modified, where $\mathbf{K}_{\overline{ND},D} \cap \mathbf{K}_{\overline{ND},N} = \varnothing$. Hence, these disjunctive sets provide a certain kind of independency.

The last conceivable strategy consists in the accurate setting of a part of the spectrum of zeros, which results in that some parameters from $\mathbf{K}_N$ are dependent to others, and consequently, find the optimal setting of independent parameters by strategies from Algorithm 1 or Algorithm 2. This idea, however, does not guarantee the dominancy of the placed zeros.

Due to the limited space, these two strategies mentioned above will be a topic of any of our future papers.

Illustrative Example

A very concise demonstrative example follows to provide the reader with the idea of control of TDS and the PPSA.

In [15] a mathematical model of a skater on the swaying bow, which represents an unstable TDS system, was introduced, and a corresponding controller designed in the R_{MS} ring was derived in [16]. The eventual reference-to-output transfer function reads

$$
\begin{aligned}
G(s) &= \frac{bb_Q(s)}{(s + m_0)^4 m_Q(s)} \exp(-(\tau + \vartheta)s) \\
b_Q(s) &= b(q_3 s^3 + q_2 s^2 + q_1 s + q_0)(s + m_0)^4 + p_0 m_0^4 s^2 (s^2 - a \exp(-\vartheta s)) \qquad (10) \\
m_Q(s) &= s^2 (s^2 - a \exp(-\vartheta s))(s^3 + p_2 s^2 + p_1 s + p_0) \\
&\quad + b \exp(-(\tau + \vartheta)s)(q_3 s^3 + q_2 s^2 + q_1 s + q_0)
\end{aligned}
$$

where delays $\tau, \vartheta \geq 0$ stand for the skater's and servo latencies, respectively, b, a are real plant parameters. Note that the spectral assignment for the polynomial factor $(s + m_0)^4, m_0 > 0$ is trivial, then the goal is to find unknown parameters of $m_Q(s)$. To cancel the impact of the quadruple real pole $s_1 = -m_0$ to the feedback dynamics, it must hold that $m_0 >> -\alpha(\mathbf{K})$ where $\alpha(\mathbf{K})$ expresses the spectral abscissa of the quasipolynomial factor. Hence, we have $\mathbf{K} = \{p_2, p_1, p_0, q_3, q_2, q_1, q_0\}$ with $\mathbf{K}_{\overline{ND}} = \{q_3, q_2, q_1, q_0, p_0\}$, $\mathbf{K}_{N\backslash D} = \varnothing$, $\mathbf{K}_{D\backslash N} = \{p_2, p_1\}$, $\mathbf{K}_N = \mathbf{K}_{\overline{ND}}$, $\mathbf{K}_D = \mathbf{K}$, that is $r = r_D = 7, r_{\overline{ND}} = r_N = 5$,

$r_{N\backslash D} = 0, r_{D\backslash N} = 2$. Let us follow Algorithm 2 which is suitable in this case since $r_{N\backslash D} = 0$ and hence Algorithm 1 can not be used.

We attempt to set $n_D = 2$, then the conditions (9) reads $0 \le n_N < 2$; therefore, let $n_D = 1$ and consider the model

$$G_m(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} = k \frac{s - z_1}{(s - s_1)(s - \bar{s}_1)} \tag{11}$$

According to the desired dynamic properties, we prescribe a zero $z_1 = -0.18$ and a complex conjugate pair of poles $s_1 = -0.1 + 0.2\mathrm{j}$. Since the initially place roots are not dominant with abscissas for poles and zeros as $\alpha_P(\mathbf{K}) = 0.8959$ and $\alpha_Z(\mathbf{K}) = -0.1373$, respectively, set $n_{sp} = 2, n_{sz} = 1$ and perform Steps 5-6 of the PPSA by means of the QCSA.

In Figs. 1 and 2 distances of the rightmost poles pair σ and the zero ζ from the prescribed ones are displayed, and the evolution of $\mathbf{K}$ during the quasi-continuous shifting is provided in Fig. 3.

Further, the SOMA is used to minimize the cost function $\Phi(\mathbf{K}) = |\sigma_1 - s_1| + |\zeta_1 - z_1| + 0.01\alpha_{r,P}(\mathbf{K}) + 0.01\alpha_{r,Z}(\mathbf{K})$ where $\alpha_{r,P}(\mathbf{K})$, $\alpha_{r,Z}(\mathbf{K})$ mean the spectral abscissa of the rest of poles and zeros, respectively. It is worth noting that the optimization yields only a slightly improvement giving the eventual spectra and the parameters set as in (12). However, final poles and zeros positions are quite far from the desired ones, which proves the fact about TDS that the desired spectrum can not be chosen arbitrarily in general.
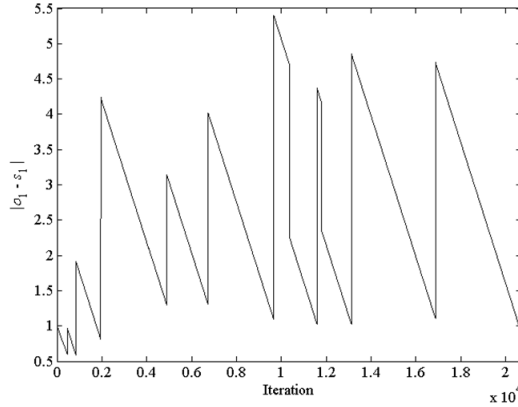
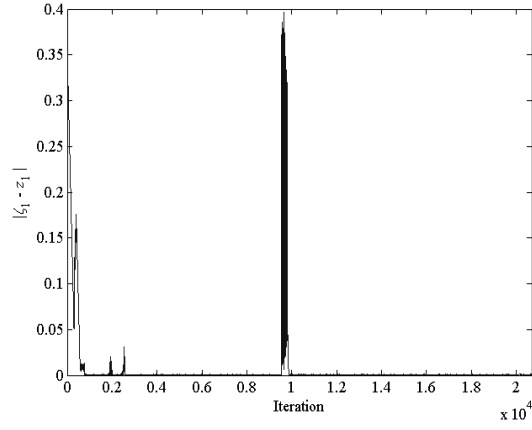**Fig. 1.** Evolution of $|\sigma_1 - s_1|$ using the PPSA with QCSA

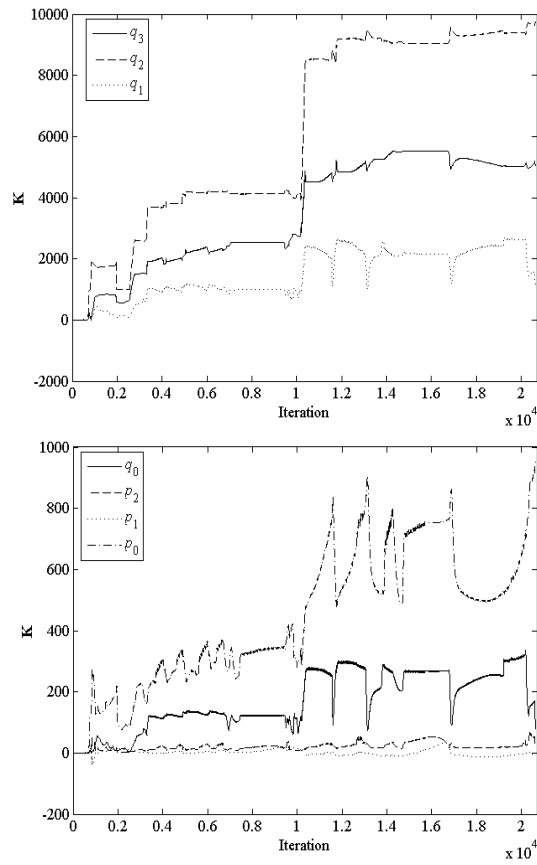**Fig. 2.** Evolution of $\left|\zeta_1 - z_1\right|$ using the PPSA with QCSA

Fig. 3. Evolution of **K** using the PPSA with QCSA

$$\Omega_{P,opt} = \{-0.1158 \pm 0.0674j, -0.1161 \pm 5.1163j, -0.1211 \pm 1.2103j, \ldots\}$$
$$\Omega_{Z,opt} = \{-0.1801, -0.2247 \pm 0.1032j, -0.7607, -2.817 \pm 8.1939j, ,\ldots\} \qquad (12)$$
$$\mathbf{K}_{opt} = [5235.169, 9829.219, 1060.87, 78.2405, 30.9684, 1.763, 947.517]^T$$

Discussion

Let us now present some ideas how to modify, extend or improve the PPSA, regarding computation acceleration, shifting strategies, model selection etc.

Considering these aspects in the chronological order according to the running of Algorithm 1 or Algorithm 2, we can start with the selection of a finite-dimensional matching model. In the example above, it is supposed that the feedback dynamics is primarily given by positions of the rightmost poles and zeros where the model is found from the desired maximum overshoot, time-to-overshoot and the relative dumping. Naturally, other strategies how to prescribe the model (with corresponding roots) can be adopted. Moreover, the dominancy of the roots can be evaluated in a different way, e.g. in [14], the method based on the "weights" of modes of the impulse response was presented.

The initial shifting, convergence and the speed of the PPSA may be improved by the use of other "approaching" strategies, e.g. only roots of the same type (real, complex) are approaching to each other, or by thorough consideration that a complex conjugate pair means two separate roots instead of one (as it used here).

Last but not least another optimization procedures can be utilized in, e.g. the well-known and efficient NM algorithm [13] or some of many modern evolutionary or genetic algorithms. In fact, computationally the most time-consuming operation is the finding of the spectrum; hence the aim is to minimize the number of these spectral evaluations. For instance, it would be desirable to parallelize an existing spectrum-searching procedure and to utilize distributed computations on graphical cards, e.g. Compute Unified Device Architecture (CUDA) or Open Computing Language (OpenCL).

Conclusion

It is always difficult to tackle optimal or suboptimal control design or controller tuning for TDS. The presented paper has summarized and revised the basic principles of the PPSA which is based on quasi-continuous feedback poles and zeros shifting to the described dominant ones according to a selected finite-dimensional feedback model. The semi-finite result from the shifting has been then improved by an optimization procedure. Two possible PPSA strategies have been introduced and discussed, and the explanation has been supported by an illustrative example. In the future research, the other possible strategies will be analyzed and, more-

over, the two presented ideas will be tested, compared and enhanced by tools discussed in this paper. Hence, the reader is kindly asked to participate on the future research, with the accent to provide us with the computational and programming support, to benchmark and verify the discussed ideas.

References

1. Hale, J.K., Verduyn Lunel S.M.: Introduction to Functional Differential Equations. Applied Mathematical Sciences, vol. 99. Springer, New York (1993)
2. Kolmanovskii, V.B., Myshkis A.: Introduction to the Theory and Applications of Functional Differential Equations. Cluwer Academy, Dordrecht, Netherlands (1999)
3. Niculescu S.I.: Delay effects on stability. Lecture Notes in Control and Information Sciences, vol. 269. Springer, Berlin (2001)
4. Richard J.P.: Time-Delay Systems: An Overview of Some Recent Advances and Open Problems. Automatica 39, 1667-1694 (2003)
5. Zítek, P., Kučera, V.: Algebraic Design of Anisochronic Controllers for Time Delay Systems. Int. J. Control 76, 1654-1665 (2003)
6. Pekař, L.: A Ring for Description and Control of Time-Delay Systems. WSEAS Trans. on Systems 11. Special Issue on Modelling, Identification, Stability, Control and Applications, 571-585 (2012)
7. Michiels, W., Engelborghs, K., Vansevevant, P., Roose, D.: Continuous Pole Placement for Delay Equations. Automatica 38, 747-761 (2002)
8. Michiels, W., Vyhlídal, T.: An Eigenvalue Based Approach for the Stabilization of Linear Time-Delay Systems of Neutral Type. Automatica 41, 991-998 (2005)
9. Michiels, W., Vyhlídal, T., Zítek, P.: Control Design for Time-Delay Systems Based on Quasi-Direct Pole Placement. J. Process Control 20, 337-343 (2010)
10. Pekař, L.: On a Controller Parameterization for Infinite-dimensional Feedback Systems Based on the Desired Overshoot. WSEAS Trans. Systems 12, 325-335 (2013)
11. Zelinka, I.: SOMA-Self Organizing Migrating Algorithm. In: Onwobolu, G.C, Babu, B.V. (eds.) New Optimization Techniques in Engineering. pp. 167-217, Springer, Berlin, 2004.
12. Vanbiervliet, T., Verheyden, K., Michiels, W., Vandewalle, S.: A Nonsmooth Optimization Approach for the Stabilization of Time-Delay Systems. ESIAM: Control, Optim. Calculus of Variations 14, 478-493 (2008)
13. Nelder, J.A., Mead, R.: A Simplex Method for Function Minimization. The Computer J. 7, 308–313 (1965)
14. Vyhlídal, T.: Analysis and Synthesis of Time Delay System Spectrum. Ph.D. thesis. Faculty of Mechanical Engineering, Czech Technical University in Prague, Prague (2003)
15. Zítek, P., Kučera, V., Vyhlídal, T.: Meromorphic Observer-Based Pole Assignment in Time Delay Systems. Kybernetika 44, 633-648 (2008)
16. Pekař, L., Prokop, R.: Algebraic Optimal Control in RMS Ring: A Case Study. Int. J. Math. Computers in Simul. 7, 59-68 (2013)