MONTE-CARLO TECHNIQUE FOR STABILIZATION OF LINEAR DISCRETE-TIME SYSTEMS VIA LOW-ORDER CONTROLLERS

Ya. Petrikevich, B. Polyak

Institute of Control Sciences of
the Russian Academy of Sciences,
Moscow, Russia

ABSTRACT. A novel randomized approach to fixed-order controller design is proposed for discrete-time SISO plants. The algorithm generates stable discrete-time polynomials, and for each of them we find the nearest polynomial in the subspace of characteristic polynomials corresponding to low-order controllers. If this polynomial is stable, the stabilizing controller is found. Otherwise we proceed to generate stable polynomials. If we achieved no success, several "most promising" candidates are found and we try to improve them locally by means of an iterative method of nonsmooth optimization. Numerical simulations demonstrated high efficiency of the approach.

1. INTRODUCTION. Stabilization of SISO plants by low-order controllers is known to be one of the challenging problems in control theory [1]. Its importance stems from the fact that such controllers are easy to adjust, and most of the real-world controllers that are presently exploited in industry are low-order ones; they are basically PI- or PID-controllers having two or three free parameters. On the other hand, few control parameters may be insufficient, and the order of the controller has to be increased thus leading to a more complicated structure.

In general, fixed-order controller design is hard, since it reduces to finding a stable polynomial in an affine family, which is known to be NP-hard [2]. Moreover, at present there are no satisfactory "yes or no" methods for the existence of stabilizing controllers for a given plant; in particular, this is the case even with PID-controllers. Various straightforward randomized methods proposed so far (e.g., see [3] for the most recent results in this area) demonstrate weak performance because they work directly in the coefficient space, while the domain of stability is typically very small.

To circumvent some of the difficulties of this sort, we propose a novel randomized approach to fixed-order controller design. It is based on random generation of stable polynomials and finding for each of them the closest element in the set of closed-loop characteristic polynomials of the system. This affine set is specified by the fixed structure of the controller.

Although the method that we propose equally applies to both continuous and discrete time systems, for ease of exposition we consider the discrete-time case in this paper, because an existing efficient mechanism of generating discrete stable polynomials can be exploited without serious modifications.

2. DESIGN METHODOLOGY. In this section, we formulate the problem, give the overall description of the method, and discuss each of its main components in more detail.

2.1. Overall description of the method. We consider SISO plants specified by the scalar transfer function

$$G(z) = \frac{n_G(z)}{d_G(z)}$$

with known polynomials $n_G(z)$, $d_G(z)$ and controllers of the form

$$C(z) = \frac{n_C(z)}{d_C(z)},$$

where the degrees of polynomials $n_C(z)$, $d_C(z)$ are fixed, thus defining the structure of a controller. To make the dependence on the controller coefficients explicit, we denote the whole set of the coefficients of $n_C(z)$, $d_C(z)$ by $q \in \mathbb{R}^{\ell}$. Then the characteristic polynomial of the closed-loop system is given by

$$p(z,q) = n_G(z)n_C(z) + d_C(z)d_C(z).$$

The collection of all polynomials such that q varies in $\mathbb{R}^{\ell}$, is denoted by $\mathcal{P}$, which is seen to be an affine polynomial family in the vector q. It is assumed that the leading coefficient of $p(z, q)$ is non-zero for all values of q, i.e., the family $\mathcal{P}$ of polynomials has invariant degree n.

We seek for a value $q = q^*$ such that $p(z, q^*)$ is stable. To this end, we propose first to generate randomly a stable polynomial $p^j(z)$ of degree n; a possible way to do this is described in Section 2.2 below. The next step is to project this sampled polynomial onto the set $\mathcal{P}$, i.e., to obtain a polynomial $p(z, q) \in \mathcal{P}$ which minimizes the distance to its stable prototype $p^j(z)$. If this projection (denote it by $p(z, q^j)$) is stable, the point q^j provides a stabilizing controller. Otherwise, we keep generating polynomials $p^j(z)$ until a stabilizing q^j is found or the user-specified number N of samples is exceeded.

Noteworthy, it may be reasonable to generate all N samples, no matter if a stabilizing controller is found or not. The reason is that this process is numerically cheap (see Section 2.2), while the outcome might be the whole collection of stabilizing controllers, so that various performance indices can further be optimized over this collection.

If the sampling-projecting process terminates without obtaining a stable $p(z, q)$, we choose $N_{\text{cand}} \ll N$ candidate polynomials among the $p(z, q^j)$, $j = 1,\ldots, N$, to be used for further tuning. By tuning we mean an optimization procedure (to be described in Section 2.4 below) which is aimed at shifting the roots of a candidate polynomial towards the unit disk iteratively in q.

The candidate polynomials can be chosen in different ways from the available information. For instance, let d_j denote the distance between the $p(z, q_j)$ and its stable prototype $p^j(z)$. Then, the polynomials having the least values of d_j can be selected as candidates. Alternatively, let $\sigma_j = \max_k |z_k|$ be the degree of instability of the (unstable) projected $p(z, q^j)$. Then the candidate polynomials are those having the least degrees of instability. A combination of both criteria might as well be accepted. The motivation for such a choice for the candidates is that they might be close (in the q-space) to stable characteristic polynomials in $\mathcal{P}$.

It is hardly possible to evaluate the probability of obtaining a stable projection (if it exists); nor there is a guarantee that the tuning procedure will lead to a stabilizing point. However, numerical tests of the proposed methodology over a number of problems involving fixed-order controllers testify to its validity and high efficiency.

Now we consider each of the components of the method in more detail.

2.2. Generation of stable polynomials. We restrict our attention to monic n-th order polynomials of the form

$$p(z) = z^n + a_{n-1}z^{n-1} + ... + a_1 z + a_0. \tag{1}$$

A polynomial is said to be (*discrete*) *stable* (Schur stable) if all its roots z_k belong to the open unit disk $C_1 \doteq \{z \in \mathbb{C} : |z| < 1\}$ on the complex plane.

The lemma below is one of the cornerstones of the approach proposed (see [4, 5]).

Lemma 1. *Any Schur stable monic polynomial* (1) *can be obtained by the recursive procedure*

$$p_0(z) = 1,\ \ p_{k+1}(z) = zp_k(z) + t_k z^k p_k(z^{-1}),\ \ |t_k| \le 1,\ \ k = 0,...,n-1. \tag{2}$$

In the control literature, the numbers t_k are referred to as the *Fam–Meditch parameters*; the lemma states that sweeping the unit cube $T = [-1, 1]^n$ in the space of these parameters yields all stable monic polynomials of degree n. Moreover, all stable polynomials of *all* degrees less than or equal to n are generated by means of this recursive procedure. Sometimes relations (2) are called the Levinson–Durbin recursion (see [6]).

Notably, this procedure does not lean on sampling any roots or coefficients of a desired stable $p(z)$, but rather produces its coefficients from the Fam–Meditch (FM) parameters; the resulting stable polynomials will be referred to as FM-polynomials.

In the implementation of the approach in this paper, the FM parameters t_k are generated randomly uniformly on [–1, 1].

2.3. Projecting onto the set of closed-loop polynomials. This simple technical step is briefly described below.

We identify the n-th-degree FM polynomial

$$p^j(z) = z^n + a_{n-1}z^{n-1} + ... + a_1 z + a_0$$

with the n-dimensional vector $p^j = (a_{n-1}, \ldots, a_1, a_0)^{\mathrm{T}}$ of its coefficients. Similarly, let p_q denote the corresponding vector for a closed-loop polynomial $p_q(z) \in \mathcal{P}$. We have

$$p_q(z) = p_0(z) + \sum_{i=1}^{\ell} q_i p_i(z) \tag{3}$$

with certain constituent polynomials $p_i(z)$, $i = 0, \ldots, \ell$ (having different degrees). This relation can be re-written in the form $p_q = Aq + p_0$, where the rectangular matrix $A \in \mathbb{R}^{nxl}$ is composed from the coefficient vectors p_i with the properly added zero components to have the same lengths. Then the projection problem of finding $\min_q \| p_q - p^j \|$ is formulated as finding

$$\arg\min_q \| Aq + p_0 - p^j \|.$$

If the Euclidean norm is used, the solution is given in closed form as the solution of the least squares problem:

$$q = (A^{\mathrm{T}} A)^{-1} A^{\mathrm{T}} (p^j - p_0).$$

Use of other norms such as ℓ_1 or ℓ_∞ require solving an appropriate linear program.

2.4. Local iterative tuning. For a detailed description of the tuning algorithm under consideration, a reader is referred to [7], where it was proposed for the continuous-time case. Here we present just its main idea as applied to Schur stability.

Given an affine polynomial family (3) of invariant degree n such that $p(z, 0)$ is unstable, we seek for a $q = q^*$ such that $p(z, q^*)$ is stable. The algorithm works in the q-space and moves the roots of $p(z, q)$ iteratively towards the boundary of the stability region (the interior of the unit circle).

The following result on the perturbed roots of a polynomial constitutes the basis for this algorithm; it can be easily obtained by expanding $p(z, q)$ into a Taylor series.

Lemma 2 [7]. *Let $p(z, q)$ be a polynomial in $z \in \mathbb{C}$, which depends on the vector of real parameters $q = (q_1,\ldots, q_l)^{\mathrm{T}}$, and* $\deg p(z, q) = n = \mathrm{const}(q)$. *Assume that $p(z, q)$ is differentiable with respect to q at the point $q = 0$ and denote*

$$\pi_i(z) = \left.\frac{\partial p(z,q)}{\partial q_i}\right|_{q=0}, \quad i = 1,\ldots,\ell.$$

Let $z_k = z_k(0)$ denote a simple zero of the polynomial $p_0(z) = p(z, 0)$. Then for sufficiently small q there exists a zero $z_k(q)$ of the polynomial $p(z, q)$ such that

$$z_k(q) = z_k + (w^k, q) + o(q),$$

where $w^k = -\pi^k / r_k,\ r_k = p_0'(z)\big|_{z=z_k}$, *and* $\pi^k = \left(\pi_1(z_k),\ldots,\pi_\ell(z_k)\right)^{\mathrm{T}}$.

For the affine linear dependence (3) on q we have $\pi_i(z) = p_i(z)$, and the vectors w^k are computed in closed form.

We start our constructions by distinguishing the *critical* roots among the roots of $p(z, 0)$, namely, those having maximal moduli. More precisely, let z^w be the "worst" root, i.e., $|z^w| \geq |z_k|$ for all roots of $p(z, 0)$. We specify a small constant $\varepsilon > 0$ and consider the roots z_k^c such that $|z_k^c| \geq |z^w| - \varepsilon$. These critical roots $z_k^c(q)$, $k = 1,\ldots, m$, are then linearized in the neighborhood of $q = 0$ using Lemma 2, and their linear approximations

$$\tilde{z}_k^c \doteq z_k^c(0) + (w^k, q), \quad k = 1,\ldots,m,$$

are considered; it is these linearized roots that will be shifted at every current iteration.

Namely, we specify a small $\delta > 0$ and seek for a smallest q that makes all of $\tilde{z}_k^c$ stable with degree of stability $1 - \delta$:

$$\min \|q\| \ \text{ s.t. } \ |z_k^c(0) + (w^k, q)| < 1 - \delta. \tag{4}$$

If a solution $\tilde{q}$ of this problem (which is seen to be a low-dimensional convex program in q) exists, it defines the direction at the current step. Finally, to guarantee that *all* roots of the "renewed" polynomial $p(z, q)$ are shifted towards the unit disk, we find

$$\alpha_{\min} = \arg\min_{\alpha>0} \max_k |z_k|$$

by one-dimensional search, where z_k are the roots of $p(z, \alpha\tilde{q})$, adopt (as the step size and take $q = \alpha_{\min}\tilde{q}$. As a result, the degree of instability of the renewed polynomial is decreased. The next iteration is performed with the resulting $p(z, q)$ after the change of variables $q := q - \alpha_{\min}\tilde{q}$.

In the context of the approach in this paper, this algorithm should be repeatedly run N_{cand} times for different unstable initial points q^j, which are provided by the candidate controllers as described in Section 2.1. As it was said, these "meaningful" initial conditions are expected to be located close to the stability domain, hence contributing to a faster convergence of the algorithm.

Several comments are due. First, there might be several "worst" roots and, moreover, some of them may have multiplicity greater than one. In that case, a refined linearization formulas can be derived. Second, the constraints in the optimization problem (4) may turn out to be inconsistent. In that case, the optimization step can be performed from a different point in a small neighborhood of the current point. Third, other optimization schemes can be devised, e.g., the one where direction at every step is chosen as a direction of common local decrease or the one that maximizes the degree of stability:

$$\max \delta$$
$$|z_k^c(0) + (w^k, q)| < 1 - \delta, \quad k = \overline{1, m},$$
$$\| q \| \le \varepsilon,$$

where $\varepsilon > 0$ is a small number.

It should be noted that this method is approximate and does not provably lead to a stable solution even if it exists. However it has demonstrated very stable performance over a range of test examples.

3. SIMULATION. We demonstrate the efficiency of the approach by a simple illustrative example of stabilization via PI-controllers. In that case, only two control parameters q_1, q_2 are involved that make the method easy to visualize on the plane.

In the first example we consider the plant with one unstable pole, specified by the transfer function

$$G(z) = \frac{2z^2 - 1.5}{z^4 - 0.5z^3 - 0.98z^2 + 0.048z + 0.144},$$

and with PI-controller of the form

$$C(z) = \frac{q_1 + q_2 z}{z}$$

in the feedback loop. The 5th-order closed-loop characteristic polynomial has the form

$$p(z, q) = p_0(z) + q_1 p_1(z) + q_2 p_2(z). \tag{5}$$

For benchmark purposes, we determined explicitly the boundary of the stability domain for $p(z, q)$ in the two-dimensional q-space using the classical D-decomposition technique originally proposed in [8]. The result is given in Fig. 1.

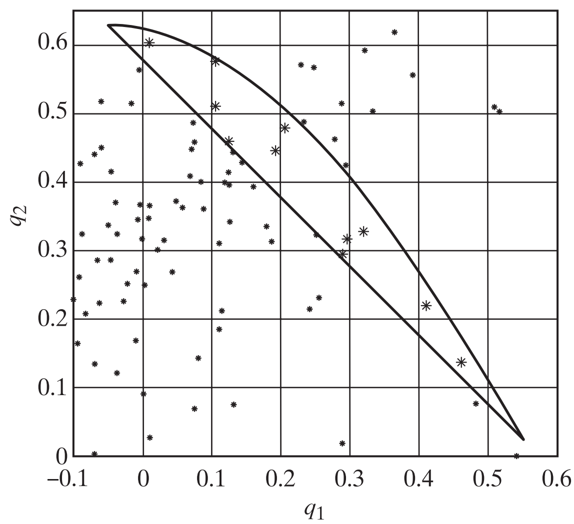

Fig. 1. Stability domain and the projected samples in the $\{q_1, q_2\}$-plane

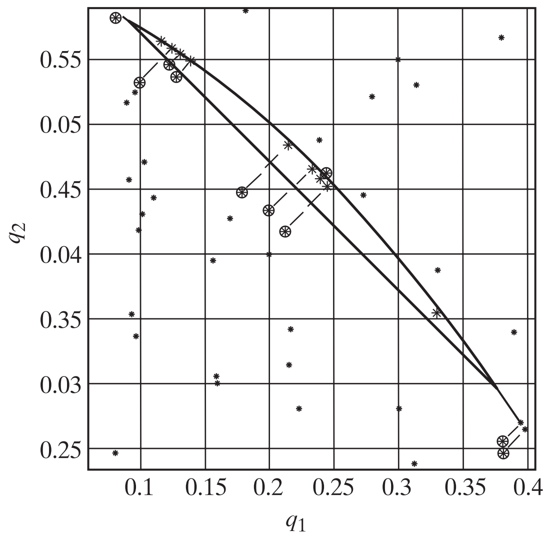

Fig. 2. Local optimization of the candidate controllers

Next, N = 500 was specified as the maximum number of sample Schur polynomials of degree 5 to be generated by the FM algorithm. As early as at the 39th attempt, the projected $p(z,q^j)$ was detected to be stable with $q^j = (0.1039, 0.5774)$ being a stabilizing controller. For illustration, we plotted the stability domain and the points q^j associated with all $N_{\text{stab}} = 11$ stable projections found during the sampling; these are marked by asterisks in the figure.

Then we slightly changed the coefficients of the plant:

$$G(z) = \frac{2z^2 - 1.57}{z^4 - 0.5z^3 - 0.98z^2 + 0.048z + 0.144}.$$

In this case the stability domain of family (5) is also nonempty, but now it is much smaller than in the previous example (see Fig. 2). As a result, all $N = 500$ randomly sampled Schur polynomials have unstable projections (see the associated points in the space of the controller coefficients, none of which fall in the stability domain).

Among these, we selected $N_{\text{cand}} = 10$ candidate polynomials having the least degrees of instability σ_j and applied the iterative method of Section 2.4 (with parameter $\delta = 0.001$) trying to locally optimize them. In other words, we ran the algorithm N_{cand} times choosing the respective q^j, $j = 1,\dots,10$, as the initial point for iterations. Optimization terminated successfully for all of them after 1 to 4 steps, thus leading to stabilizing controllers. The corresponding N_{cand} trajectories of the algorithm are also shown in Fig. 2.

Numerical simulations have been performed for higher-order plants having several unstable poles, higher-order controllers (for instance, PID-controllers), etc. The method showed excellent performance in all the examples.

4. CONCLUSIONS. In this paper, we proposed a simple-to-implement randomized technique for fixed order controller design. Although it does not provably lead to a solution, the method has demonstrated high efficiency over a range of test problems; it is believed to be practically useful in control applications.

Among natural extensions of the approach is its an immediate modification to the continuous-time case, with the associated methods of random generation of Hurwitz stable polynomials.

The technique can be extended to the problem of maximizing the degree of stability of the closed-loop system, robust statements of the problem (where the plant coefficients are not known exactly), simultaneous stabilization, etc. In these cases, only the iterative algorithm of Section 2.4 should be properly modified.

One of the interesting directions for further research would be development and deeper analysis of alternative efficient methods for generating stable polynomials and extensions of the overall approach to the MIMO case would be fulfilled.

REFERENCES

1. *Åstro¨m K.J., Hägglund T.* Advanced PID Control. Instrumentation, Systems and Automation Society: Res. Triangle Park (N.C.), 2006.
2. *Blondel V., Tsitsiklis J.N.* A survey of computational complexity results in systems and control // Automatica. 2000. Vol. 36. P. 1249–1274.
3. *Tempo R., Calafiore G., Dabbene F.* Randomized Algorithms for Analysis and Control of Uncertain Systems. L.: Springer, 2004.
4. *Fam A., Meditch J.* A canonical parameter space for linear systems design // IEEE Trans. Autom. Control. 1978. Vol. 23(3). P. 454–458.
5. *Prakash M.N., Fam A.T.* A geometric root distribution criterion // Ibid. 1982. Vol. 27(2). P. 494–496.
6. *Jury E.I.* Inners and Stability of Dynamic Systems. N.Y.: Wiley, 1974.
7. *Polyak B.T., Shcherbakov P.S.* A new approach to robustness and stabilization of control systems via perturbation theory // Proc. XIV World Congr. IFACC. 1999. P. 13–18.
8. *Neimark Yu.I.* Stability of Linearized Systems Leningrad: LKVVIA, 1949 (in Russian).