KYBERNETIKA — VOLUME 12 (1976), NUMBER 4

New Algorithm for Polynomial Spectral Factorization with Quadratic Convergence II

ZDENĚK VOSTRÝ

In this paper new efficient algorithm for the numerical spectral factorization of polynomials arising continuous optimality problem is derived.

INTRODUCTION

It is known that two cases of the polynomial spectral factorization are used in applications: the polynomial "discrete" spectral factorization $\varphi(\zeta)\,\varphi(\zeta^{-1})$ of $b(\zeta)$ $b(\zeta^{-1})$ mentioned in Part I [1], and the polynomial "continuous" spectral factorization $\varphi(s)\,\varphi(-s)$ of $b(s)\,b(-s)$ such that $b(s)\,b(-s) = \varphi(s)\,\varphi(-s)$ and all roots of the polynomial $\varphi(s)$ have nonpositive real parts. It is supposed that $b(s) = b_0 + b_1 s + \ldots$ $\ldots + b_k s^k$, $\varphi(s) = \varphi_0 + \varphi_1 s + \ldots + \varphi_k s^k$ are polynomials with real coefficients. This spectral factorization is used in quadratic continuous optimality problems.

There are known three numerical methods for the computation of the spectral factorization $\varphi(s)\,\varphi(-s)$ of $a(s^2) = b(s)\,b(-s)$:

(i) computation of the roots of $a(s^2)$ and their suitable selection,

(ii) mapping the "continuous" variable s into the "discrete" variable ζ by

$$s = \frac{1 - \zeta}{1 + \zeta}$$

and solving the resulting discrete factorization problem and map the solution back into the continuous plane [3].

(iii) Newton-Raphson method.

First method is very cumbersome.

Second method requires more operations than the "discrete" factorization problem.

Third method is very useful and in this paper a new computational approach will be derived.

In the same way as in Part I we obtain the iteration formula

(1) $$\varphi^{(i)}\bar{\varphi}^{(i+1)} + \bar{\varphi}^{(i)}\varphi^{(i+1)} = a + \varphi^{(i)}\bar{\varphi}^{(i)}, \quad i = 0, 1, 2, \ldots,$$

where i is number of iteration,

$$\varphi^{(i)} = \varphi^{(i)}(s) = \varphi_0^{(i)} + \varphi_1^{(i)}s + \ldots + \varphi_k^{(i)}s^k,$$

$$\bar{\varphi}^{(i)}(s) = \varphi^{(i)}(-s),$$

$$a = a(s^2) = a_0 + a_1 s^2 + \ldots + a_k s^{2k}, \quad a_0, a_1, \ldots a_k \quad \text{real}.$$

It is known that the polynomial $a(s^2)$ can be factorized as $\varphi(s)\,\varphi(-s) = a(s^2)$ with $\varphi_0, \varphi_1, \ldots, \varphi_k$ real if and only if $a(-\omega^2) \geqq 0$ for all real ω. We shall consider only factorizable polynomials $a(s^2)$.

We say that a polynomial φ with real coefficients is stable if all its roots have negative real parts. If φ is a stable polynomial and $\varphi\bar{\varphi} = a$ then $(-\varphi)(-\bar{\varphi}) = a$ and $-\varphi$ is a stable polynomial, too. As the stable spectral factor of $a(s^2)$ we define a polynomial φ such that $\varphi\bar{\varphi} = a$ and the coefficients of the polynomial φ are positive.

Properties of the sequence $\varphi^{(0)}, \varphi^{(1)}, \varphi^{(2)}$ generated by (1).

Theorem 1. For any polynomial $\varphi^{(i)}$ the next inequality holds

(2) $$\varphi^{(i+1)}\bar{\varphi}^{(i+1)} \geqq a \quad \text{for} \quad s = j\omega, \omega \in (-\infty, +\infty), \quad i = 1, 2, \ldots$$

Proof. It is evident that $(\varphi^{(i+1)} - \varphi^{(i)})(\bar{\varphi}^{(i+1)} - \varphi^{(i)}) \geqq 0$ and hence the inequality (2) follows on using (1).

Theorem 2. Let $\varphi^{(0)} \neq 0$ in the closed right half-plane (CRHP) i.e. $\varphi^{(0)}$ is a stable polynomial, the sequence $\varphi^{(0)}, \varphi^{(1)}, \ldots$ has the properties:

(i) $\varphi^{(i)} \neq 0$ in the CRHP implies $\varphi^{(i+1)} \neq 0$ in the CRHP (if $\varphi^{(i)}$ is stable then $\varphi^{(i+1)}$ is stable, too).

(ii) $\varphi^{(0)}, \varphi^{(1)}, \varphi^{(2)} \ldots$ converges to a φ such that $\varphi\bar{\varphi} = a$ and $\varphi \neq 0$ in the open right half plane.

(iii) For real $r \in \langle 0, \infty \rangle$ the following inequality

$$\tfrac{1}{2}\varphi^{(i)}(r) < \varphi^{(i+1)}(r) \leqq \varphi^{(i)}(r), \quad i = 1, 2, \ldots$$

holds.

(iv) The convergence is quadratic in nature.

Proof.

(i) Divide the equation (1) by $\varphi^{(i)}\bar{\varphi}^{(i)}$ then

$$\frac{\varphi^{(i+1)}}{\varphi^{(i)}} + \frac{\bar{\varphi}^{(i+1)}}{\bar{\varphi}^{(i)}} = \frac{a}{\varphi^{(i)}\bar{\varphi}^{(i)}} + 1 .$$

The function

$$\frac{\varphi^{(i+1)}(s)}{\varphi^{(i)}(s)}$$

is analytic in the CRHP, which is bounded by the imaginary axis and the right half circle with infinite radius.

Consider such a stable $\varphi^{(0)}$ for which $\varphi_0^{(0)} = \sqrt{a_0}$, $\varphi_k^{(0)} = \sqrt{|a_k|}$ then $\varphi_0^{(0)} = \varphi_0^{(i)}$, $\varphi_k^{(0)} = \varphi_k^{(i)}$ from (1) and

$$\lim_{s \to \infty} \frac{a}{\varphi^{(i)}\bar{\varphi}^{(i)}} = 1 .$$

Using (2)

$$\mathcal{R}e \frac{\varphi^{(i+1)}(j\omega)}{\varphi^{(i)}(j\omega)} = \frac{1}{2}\left(\frac{a(-\omega^2)}{\varphi^{(i)}(j\omega)\varphi^{(i)}(-j\omega)} + 1\right) \geqq \frac{1}{2}, \quad i = 1, 2, \ldots$$

The function

$$\frac{\varphi^{(i+1)}(s)}{\varphi^{(i)}(s)}$$

is harmonic in the CRHP and hence

$$\mathcal{R}e \frac{\varphi^{(i+1)}(s)}{\varphi^{(i)}(s)} \geqq \frac{1}{2}$$

and $\varphi^{(i+1)}(s) \neq 0$ in the CRHP, i.e. the $\varphi^{(i+1)}$ is a stable polynomial.

(iii) Choosing $s = r$, $r \geqq 0$ real, then from the above inequality it follows

$$\frac{1}{2} \leqq \frac{\varphi^{(i+1)}(r)}{\varphi^{(i)}(r)} \lesseqgtr 1 , \quad \text{for} \quad i = 1, 2, \ldots \tag{3}$$

(ii) By (3) the sequence $\varphi^{(1)}(r)$, $\varphi^{(2)}(r), \ldots$, $r \geqq 0$ is nonincreasing and bounded and hence it converges to φ. From (i) it follows that the root of φ have non-nonpositive real parts.

Consider the substitution

(4) $$\varphi^{(i+1)} = \tfrac{1}{2}(\varphi^{(i)} + x^{(i)})$$

where $x^{(i)}$ is a polynomial,
then from (1) the polynomial $x^{(i)}$ is given as

(5) $$\varphi^{(i)}\bar{x}^{(i)} + \bar{\varphi}^{(i)}x^{(i)} = 2a\,.$$

By solving (5) and (4) for a suitable initial polynomial $\varphi^{(0)}$ and for $i = 1, 2, \ldots$ we obtain the sequence $\varphi^{(0)}, \varphi^{(1)}, \varphi^{(2)}, \ldots$ This polynomial sequence converges to the spectral factor φ of the polynomial a as proved above. The basic problem in this spectral factorization algorithm is to solve the symmetric polynomial equation

(6) $$\varphi\bar{x} + \bar{\varphi}x = 2a\,.$$

This equation is symmteric with respect to the substitution of $-s$ for s.

Properties of equation (6)

Denote ∂x the degree of a polynomial x. In our case $2\,\partial\varphi = \partial a$ and, moreover, we require $\partial x = \partial\varphi$.

If φ is a stable polynomial (it implies that the roots of φ do not lie on the imaginary axis) then the equation (6) has only one solution with $\partial x = \partial\varphi$.

If $\varphi(\mathrm{j}\omega) = 0$ and $a(-\omega^2) = 0$ for some real ω then the equation (6) have many solutions with $\partial x = \partial\varphi$.

Example 1. Let

$$\varphi = s^3 + s^2 + s + 1 = (s^2 + 1)(s + 1)$$
$$a = -s^6 - s^4 + s^2 + 1 = (s^2 + 1)(-s^2 + 1)$$

then the polynomial $(s^2 + 1)$ can be canceled out of the equation (6) and hence

$$(s + 1)\,\bar{x} + (-s + 1)\,x = 2(-s^2 + 1)(s^2 + 1)$$

gives the solutions

$$x = 1 + \alpha s + \alpha s^2 + s^3$$

where α is any real number.

This example ilustrates the situation in our algorithm when $\varphi^{(i)}$ is a very good aproximation of φ. From the numerical point of view, this is a troublesome case.

The polynomial equation (6) could be solved by the Euclid algorithm as it is shown in [2] but with some complications due to the requirement $\partial x = \partial \bar{x}$. Further we shall construct high efficiency algorithm for solving (6) with some useful properties. In particular, the number of operations is reduced to one quarter in comparison with the Euclid algorithm.

Solving the symmetric polynomial equation (6)

Write the symmetric polynomial equation (6) in the following matrix form

$$(7)\qquad \begin{bmatrix} \varphi_0 & 0 & 0 & 0 & 0 & & & & \\ \varphi_2 & \varphi_1 & \varphi_0 & 0 & 0 & & & & \\ \varphi_4 & \varphi_3 & \varphi_2 & \varphi_1 & \varphi_0 & & & & \\ \cdot & \cdot & \varphi_4 & \varphi_3 & \varphi_2 & & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ & & & & \varphi_k & \varphi_{k-1} & \varphi_{k-2} & \varphi_{k-3} & \varphi_{k-4} \\ & & & & 0 & 0 & \varphi_k & \varphi_{k-1} & \varphi_{k-2} \\ & & & & & & 0 & 0 & \varphi_k \end{bmatrix} \begin{bmatrix} x_0 \\ -x_1 \\ x_2 \\ -x_3 \\ \cdot \\ \cdot \\ (-1)^k x_{k-2} \\ -(-1)^k x_{k-1} \\ (-1)^k x_k \end{bmatrix} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ \cdot \\ \cdot \\ a_{k-2} \\ a_{k-1} \\ a_k \end{bmatrix}.$$

It is evident that $\varphi_0 = \sqrt{a_0}$, $\varphi_k = \sqrt{(-1)^k a_k}$ imply

$$x_0 = \varphi_0\,, \quad x_k = \varphi_k$$

Denote

$$(8)\qquad \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \\ \cdot \\ \cdot \\ a'_{k-1} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \cdot \\ \cdot \\ a_{k-1} \end{bmatrix} - \begin{bmatrix} \varphi_0 \\ \varphi_2 \\ \varphi_4 \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} x_0 - \begin{bmatrix} \cdot \\ \cdot \\ \\ \varphi_{k-4} \\ \varphi_{k-2} \\ \varphi_k \end{bmatrix} (-1)^k x_k$$

then for unknown $x_1, x_2, \ldots, x_{k-1}$ the next equation holds

$$(9)\qquad \begin{bmatrix} \varphi_1 & \varphi_0 & & & & & & \\ \varphi_3 & \varphi_2 & \varphi_1 & \varphi_0 & & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ & & & & \varphi_k & \varphi_{k-1} & \varphi_{k-2} & \varphi_{k-3} \\ & & & & & \cdot & \varphi_k & \varphi_{k-1} \end{bmatrix} \begin{bmatrix} -x_1 \\ x_2 \\ \cdot \\ \cdot \\ (-1)^k x_{k-2} \\ -(-1)^k x_{k-1} \end{bmatrix} = \begin{bmatrix} a'_1 \\ a'_2 \\ \cdot \\ \cdot \\ a'_{k-2} \\ a'_{k-1} \end{bmatrix}.$$

In shortland matrix notation

$$(10)\qquad \Phi X = A'\,.$$

The matrix Φ is the so called Hurwitz matrix.

To find the coefficients of the polynomial x from the matrix equation (9) we use column elementary transformations on Φ instead of row elementary transformations. Introduce a substitution $X = TY$ in (10), where Y is a $k-1$ vector and T is a $(k-1) \times (k-1)$ column elementary transformations matrix. such that ΦT matrix will be low triangular.

From the special form of the Φ matrix it follows that the above column elementary transformations are given by the Routh stability test of the polynomial φ.

Example.

Let $\varphi = \varphi_0 + \varphi_1 s + \varphi_2 s^2 + \varphi_3 s^3 + \varphi_4 s^4$, then

$$\Phi = \begin{bmatrix} \varphi_1 & \varphi_0 & 0 \\ \varphi_3 & \varphi_2 & \varphi_1 \\ 0 & \varphi_4 & \varphi_3 \end{bmatrix}.$$

Now we construct the T matrix in the form $T = T_1 T_2$, where

$$T_1 = \begin{bmatrix} 1 & P_1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad P_1 = \frac{-\varphi_0}{\varphi_1}$$

$$\Phi T_1 = \begin{bmatrix} \varphi_1 & 0 & 0 \\ \varphi_3 & \varphi_2^1 & \varphi_1^1 \\ 0 & \varphi_4^1 & \varphi_3^1 \end{bmatrix} \quad \begin{aligned} \varphi_1^1 &= \varphi_1 \\ \varphi_3^1 &= \varphi_3 \\ \varphi_2^1 &= \varphi_2 + \varphi_3 P_1 \\ \varphi_4^1 &= \varphi_4 \end{aligned}$$

$$T_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & P_2 \\ 0 & 0 & 1 \end{bmatrix} \quad P_2 = \frac{-\varphi_1^1}{\varphi_2^1}$$

$$\Phi T_1 T_2 = \begin{bmatrix} \varphi_1 & 0 & 0 \\ \varphi_3 & \varphi_2^1 & 0 \\ 0 & \varphi_4^1 & \varphi_3^2 \end{bmatrix} \quad \varphi_3^2 = \varphi_3^1 + \varphi_4^1 P_2 .$$

The Routh test of stability is in this case

$$\begin{array}{cccccl} \varphi_0 & \varphi_1 & \varphi_2 & \varphi_3 & \varphi_4 & \\ \varphi_1 & 0 & \varphi_3 & 0 & 0 & | P_1 \quad P_1 = \dfrac{-\varphi_0}{\varphi_1} \\ \hline 0 & \varphi_1^1 & \varphi_2^1 & \varphi_3^1 & \varphi_4^1 & \\ & \varphi_2^1 & 0 & \varphi_4^1 & 0 & | P_2 \quad P_2 = \dfrac{-\varphi_1^1}{\varphi_2^1} . \\ \hline & & \varphi_2^2 & \varphi_3^2 & \varphi_4^2 & \end{array}$$

This example shows that we need only four operations (divisions or multiplications) to obtain ΦT.

The equation $\Phi TY = A'$ gives Y after simple computation. The vector X is given by $X = TY$ as follows

$$T_2 Y = \begin{bmatrix} y_1 \\ y_2 + P_2 y_3 \\ y_3 \end{bmatrix} = \begin{bmatrix} y_1^1 \\ y_2^1 \\ y_3^1 \end{bmatrix}$$

$$T_1 T_1 Y = \begin{bmatrix} y_1^1 + P_1 y_2^1 \\ y_2^1 \\ y_3^1 \end{bmatrix} = \begin{bmatrix} -x_1 \\ x_2 \\ -x_3 \end{bmatrix}.$$

Computing TY requires in this case only two operations. The Routh test of stability is in general

(11)
$$\begin{array}{llll}
\varphi_0 \; \varphi_1 \; \varphi_2 \; \varphi_3 \; \cdots & & & \\
\varphi_1 \; 0 \; \varphi_3 \; 0 \; \cdots & | P_1 & P_1 = \dfrac{-\varphi_0}{\varphi_1} \\
\hline
\varphi_1^1 \; \varphi_2^1 \; \varphi_3^1 \; \varphi_4^1 \; \cdots & & \\
\varphi_2^1 \; 0 \; \varphi_4^1 \; 0 \; \cdots & | P_2 & P_2 = \dfrac{-\varphi_1^1}{\varphi_2^1} \\
\hline
\cdots & & \\
\cdots & & \\
0 \; \varphi_{k-3}^{k-3} \; \varphi_{k-2}^{k-3} \; \varphi_{k-1}^{k-3} \; \varphi_k^{k-3} & & \\
\varphi_{k-2}^{k-3} \; 0 \; \varphi_k^{k-3} \; 0 & | P_{k-2} & P_{k-2} = \dfrac{-\varphi_{k-3}^{k-3}}{\varphi_{k-2}^{k-3}} \\
\hline
\varphi_{k-2}^{k-2} \; \varphi_{k-1}^{k-2} \; \varphi_k^{k-2} & &
\end{array}$$

It is known that the polynomial φ is stable if $\varphi_0, \varphi_1^1, \varphi_2^2, \ldots, \varphi_{k-2}^{k-2}$ and $\varphi_{k-1}^{k-2}, \varphi_k^{k-2}$ are positive numbers.

The equations $\Phi TY = A'$ can be written as

(12)
$$\begin{bmatrix} \varphi_1 & 0 & & & & \\ \varphi_3 & \varphi_2^1 & 0 & & & \\ \varphi_5 & \varphi_4^1 & \varphi_3^2 & 0 & & \\ \cdots & & & & & \\ \cdots & & & & & \\ & & 0 & \varphi_k^{k-5} & \varphi_{k-1}^{k-4} & \varphi_{k-2}^{k-3} \\ & & & 0 & \varphi_k^{k-3} & \varphi_{k-1}^{k-2} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdot \\ \cdot \\ y_{k-2} \\ y_{k-1} \end{bmatrix} = \begin{bmatrix} a_1' \\ a_2' \\ a_3' \\ \cdot \\ \cdot \\ a_{k-2}' \\ a_{k-1}' \end{bmatrix}.$$

Hence y_1, y_2, y_{k-1} are given by simple computations. The substitution $X = TY$ can be computed by using the following scheme

$$
(13) \qquad
\begin{array}{llllllll}
y_1 & \cdots\cdots\cdots & & y_{k-4} & y_{k-3} & y_{k-2} & y_{k-1} & \\
0 & \cdots\cdots\cdots & & 0 & 0 & y_{k-1} & 0 & \big| \; P_{k-2} \\
\hline
y_1^1 & \cdots\cdots\cdots & & y_{k-4}^1 & y_{k-3}^1 & y_{k-2}^1 & y_{k-1}^1 & \\
0 & \cdots\cdots\cdots & & 0 & y_{k-2}^1 & 0 & 0 & \big| \; P_{k-3} \\
\hline
y_1^2 & \cdots\cdots\cdots & & y_{k-4}^2 & y_{k-3}^2 & y_{k-2}^2 & y_{k-1}^2 & \\
0 & \cdots\cdots & 0 & y_{k-3}^2 & 0 & y_{k-1}^2 & 0 & \big| \; P_{k-4} \\
\hline
y_1^3 & \cdots\cdots & y_{k-5}^3 & y_{k-4}^3 & y_{k-3}^3 & y_{k-2}^3 & y_{k-1}^3 & \\
0 & \cdots\; 0 & y_{k-4}^3 & 0 & y_{k-2}^3 & 0 & 0 & \big| \; P_{k-5} \\
\hline
y_1^4 & \cdots\cdots & y_{k-5}^4 & y_{k-4}^4 & y_{k-3}^4 & y_{k-2}^4 & y_{k-1}^4 & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \big| \; P_1 \\
\hline
y_1^{k-2} & y_2^{k-2} & \cdots\cdots & \cdots\cdots & \cdots & y_{k-2}^{k-2} & y_{k-1}^{k-2} \; . &
\end{array}
$$

The last row of this scheme gives $-x_1, x_2, -x_3, \ldots, -(-1)^k x_{k-1}$.

In the first and the second steps of the above scheme only the coefficients y_{k-2}^1 and y_{k-3}^2 are computed, the others remain the same. In the third and the fourth step only the coefficients y_{k-2}^3, y_{k-4}^3 and y_{k-3}^4, y_{k-5}^4 are computed, the others remain the same etc.

This simple algorithm gives very good results. The symmetric polynomial equation (6) is solved in such a way that the stability of the polynomial φ is tested during the solution process.

This fact is very useful because in our spectral factorization algorithm (see (4), (5)) the stability of $\varphi^{(i)}$ is numerically tested in every iteration.

The number of operations for solving (6) by the above method is $\frac{3}{4}k^2 + \frac{1}{2}k - 3$ for k even and $\frac{3}{4}k^2 + \frac{3}{2}k - \frac{5}{4}$ for k odd, where k is the degree of φ.

Choice of the starting polynomial

Consider a polynomial $g(s) = g_0 + g_1 s + \ldots + g_k s^k$, $g_k \neq 0$. It is known that a huge range of values g_i, $i = 0, 1, \ldots, k$ is inconvenient from the numerical point of view. A common remedy is to use a substitution $g = \lambda p$, for real λ, in $g(s)$ such that the polynomial $h(p) = g(\lambda p)$ has its coefficients in a substantially smaller range. This substitution cannot be used in the "discrete" spectral factorization problem because it changes the "discrete" stability boundary (the unite circle centred at the origin). In the "continuous" spectral factorization problem this substitution can be used because it does not change the "continuous" stability boundary (imaginary axis) and $\lambda > 0$ implies that $h(p)$ is a stable polynomial if and only if $g(p)$ is a stable polynomial.

From numerical experiments it seems that the suitable choice of λ is $\lambda = (g_0/|g_k|)^{1/k}$. For this λ, the product of all roots of $k(p)$ is equal to 1.

Recommended procedure

Given the polynomial $G(s^2) = G_0 + G_1 s^2 + \ldots + G_k s^{2k}$, then we compute the polynomial

$$A(p^2) = \frac{1}{G_0} G(\lambda p^2) = A_0 + A_1 p^2 + \ldots + A_2 p^{2k},$$

where

$$\lambda = \left(\frac{G_0}{|G_k|}\right)^{1/k}$$

and spectral factorization of $A(p^2) = \varphi(p)\,\varphi(-p)$. The spectral factorization of $G(s^2) = g(s)\,g(-s)$ is given as

$$g(s) = \sqrt{(G_0)}\,\varphi(\mu s),$$

where

$$\mu = \frac{1}{\sqrt{\lambda}}.$$

As the starting polynomial for the factorization of $A(p)$ we choose

$$\varphi^{(0)} = (1 + p)^k .$$

In this case $\varphi_0^{(i)} = \varphi_k^{(i)} = x_0^{(i)} = x_k^{(i)} = A_0 = (-1)^k A_k = 1$. Therefore, k operations for solving (7) are saved and only $\varphi_1^{(i)}, \varphi_2^{(i)}, \ldots, \varphi_{k-1}^{(i)}$ are computed in every iteration.

Stop rule

The basic problem in the above iterative algorithm is to stop the iteration process in such a way that the result is stable in spite of numerical errors and has a maximal reachable accuracy.

The first condition can be guaranteed very simply because the computation of each iteration is based on the stability test of the previous iteration.

The second condition may be satisfied to some extend by testing of the monotonicity of the sequence $\varphi^{(1)}(1), \varphi^{(2)}(1), \ldots$, (see (3)) and inequality $\varphi^{(i)}(1)\bar{\varphi}^{(i)}(1) \geqq a(1)$.

The result of the iteration process is chosen in the following way

(i) if $(\varphi^{(n)}(1) > \varphi^{(n-1)}(1)$ or $\varphi^{(n)}(1)\bar{\varphi}^{(n)}(1) < a(1)$ then (if $\varphi^{(n)}$ is stable then $\varphi = \varphi^{(n)}$ else $\varphi = \varphi^{(n-1)}$),

(ii) if $n > 30$ or $\|\varphi^{(n)}\bar{\varphi}^{(n)} - a\| < 10^{-14}\|a\|$ then $\varphi = \varphi^{(n)}$ $(\|a\| = \max_{0 \leqq l \leqq k} |a_l|)$,

(iii) if during the computation of $\varphi^{(n)}$ the stability test of $\varphi^{(n-1)}$ does not hold then $\varphi = \varphi^{(n-2)}$.

Short description of the algorithm (factorization $G(p^2) = g(p)\, g(-p)$).

(1) If the degree of $G(p^2)$ equals 2, $k = 1$, then $g_0 = \sqrt{G_0}$, $g_1 = \sqrt{-G_1}$. Go to 14.

(2) If the degree of $G(p^2)$ equals 4, $k = 2$, then $g_0 = \sqrt{G_0}$, $g_2 = \sqrt{G_2}$, $g_1 = \sqrt{(2g_0g_1 - G_1)}$. Go to 14.

(3) Generate a polynomial $A(p^2)$ from the given polynomial $G(p^2)$ such that

$$A(p^2) = \frac{1}{G_0}\, G(\lambda p^2)\,, \quad \lambda = \left(\frac{G_0}{|G_k|}\right)^{1/k}.$$

(4) $\varphi^{(0)} = (1 + p)^k \,.\, A'$ by (8).

(5) $i = 0$.

(6) $i = i + 1$. Routh test of stability for the polynomial $\varphi^{(i-1)}$ (see (9)).

(7) If $\varphi^{(i-1)}$ is not stable then $\varphi = \varphi^{(i-2)}$. Go to 12.

(8) Compute Y from $\Phi TY = A'$ (see (10)).

(9) $X = TY$ (by scheme (11)).

(10) $$\varphi^{(i)} = \tfrac{1}{2}(\varphi^{(i-1)} + x^{(i-1)})\,.$$

(11) If the stop rule (i) or (ii) is not satisfied then go to 6.

(12) $$\mu = \frac{1}{\sqrt{\lambda}}\,.$$

(13) $$g(p) = \sqrt{(G_0)}\, \varphi(\mu p)\,.$$

(14) END.

Numerical examples

Computer IBM 370, 16 decimal digits, program in the PL/I language)

1. $\partial b = 4$, $b\bar{b} = a$, b – accurate spectral factor of a.

b	24	50	35	10	1
a	576	−820	237	−30	1

after CPU time 0·11 s and $n = 5$ (n – number of iterations) a polynomial b is written as

e	e_0	e_1	e_2	e_3	e_4
$\varphi - b$	0	7·1 E–15	7·1 E–15	8·9 E–16	0
$\varphi\bar{\varphi} - a$	0	4·0 E–13	–1·7 E–13	3·6 E–15	0

2. $\partial b = 6$

b	1	11	43	83	73	25	1
a	1	–23	169	–1159	1265	–479	1

after CPU time 0·23 s and $n = 8$

$\varphi - b$	0	1·8 E–15	7 E–15	1·1 E–14	7·1 E–15	0	0
$\varphi\bar{\varphi} - a$	0	2·5 E–14	5·7 E–14	0	1·3 E–12	0	0

3. $\partial b = 6$

b	1	36	251	485	251	36	1
a	1	–794	28 583	–111 813	28 583	–794	1

after CPU time 0·30 s and $n = 11$

$\varphi - b$	0	3·6 E–15	2·8 E–14	5·7 E–14	2·8 E–14	3·6 E–15	0
$\varphi\bar{\varphi} - a$	0	2·3 E–13	–5·4 E–12	2·9 E–11	–5·4 E–12	2·3 E–13	0

4. $\partial b = 4$, $b = (s^2 + 1)^2$

b	1	0	2	0	1
a	1	4	6	4	1

after CPU time 0·50 s and $n = 30$

$\varphi - a$	0	–5·7 E–5	7·3 E–9	–5·7 E–5	0
$\varphi\bar{\varphi} - a$	0	–3·6 E–8	1·8 E–8	–1·8 E–8	0

5. $\partial b = 4$

$(s^2 + 0{\cdot}2s + 1)^2$	CPU = 0·30 s	$\varphi - b$	0	–1·3 E–11	–5·3 E–13	–1·3 E–11	0
	$n = 17$	$\varphi\bar{\varphi} - a$	0	–5·1 E–15	–1·2 E–14	–6·9 E–15	0
$(s^2 + 0{\cdot}002s + 1)^2$	CPU = 0·41 s	$\varphi - b$	0	–2·3 E–8	–9·2 E–11	–2·3 E–8	0
	$n = 24$	$\varphi\bar{\varphi} - a$	0	–6·7 E–16	–2·4 E–15	2·2 E–15	0
$(s^2 + 0{\cdot}0002\,s + 1)^2$	CPU = 0·49 s	$\varphi - b$	0	–1·4 E–5	–5·6 E–9	–1·4 E–5	0
	$n = 29$	$\varphi\bar{\varphi} - a$	0	–1·6 E–14	–3·3 E–14	–1·7 E–14	0
$(s^2 + 0{\cdot}00002s + 1)^2$	CPU = 0·52 s	$\varphi - b$	0	–1·4 E–4	–1·6 E–8	–1·4 E–4	0
	$n = 31$	$\varphi\bar{\varphi} - a$	0	–3·7 E–10	–7·3 E–10	–3·7 E–10	0

Examples four and five are troublesome cases because the roots of b lie on or close to the imaginary axis, see Example 1.

CONCLUSION

From many numerical examples it follows that the accuracy of the results depends only on the roots nearest to the imaginary axis and on their multiplicity. If the polynomial b has not troublesome roots the polynomial φ is correct to fifteen decimal digits.

(Received November 13, 1975.)

REFERENCES

[1] Z. Vostrý: New algorithm for polynomial spectral factorization with quadratic convergence I. Kybernetika *11* (1975), 6, 415—422.

[2] V. Kučera: Algebraic theory of discrete optimal control for single-variable systems I. Kybernetika *9* (1973), 2, 94—107.

[3] J. Ježek: Polynomial factorization algorithm. Scientific report ČSAV 1971.

Ing. Zdeněk Vostrý, CSc., Ústav teorie informace a automatizace ČSAV (Institute of Information Theory and Automation — Czechoslovak Academy of Sciences), Pod vodárenskou věží 4, 180 76 Praha 8. Czechoslovakia.