Jednoczesne metody znajdowania wartości szczególnych oraz zer wielomianów ortogonalnych

On Simultaneous Rootfinding Methods for Singular Values and Orthogonal Polynomials

Iwona Wróbel[1]

Treść: Rozważamy zastosowania pewnych metod wyznaczania miejsc zerowych w problemie obliczania wartości szczególnych macierzy dwudiagonalnych.
Proponujemy algorytmy będące modyfikacjami metod klasycznych: Weierstrassa, Abertha i Bairstowa obliczania wszystkich pierwiastków wielomianu. Wykorzystywane są własności rozpatrywanych macierzy zarówno w konstrukcji samego algorytmu jak i odpowiednim doborze wartości początkowych oraz w wyborze warunku zakończenia obliczeń. Rozważane zmodyfikowane metody mogą być również stosowane do wyznaczania pierwiastków wielomianów ortogonalnych.

Słowa kluczowe: metody jednoczesnego wyznaczania pierwiastków, wartości szczególne macierzy, wartości własne macierzy, wielomiany ortogonalne.

Abstract: We consider applications of certain rootfinding methods for the bidiagonal singular value problem. The problem of computing singular values of a bidiagonal n-by-n matrix is equivalent to computing eigenvalues of a symmetric tridiagonal $2n$-by-$2n$ matrix.
The algorithms we propose are modifications of the classical Weierstrass, Aberth and Bairstow methods for computing all roots of a polynomial. We make use of the properties of the matrix, both in algorithms themselves and in the choice of the initial approximation and the stopping criterion. We also apply these modified methods to finding roots of orthogonal polynomials.

2000 Mathematics Subject Classification: 65F15, 15A18.

Key words: Simultaneous rootfinding methods, singular values, eigenvalues, orthogonal polynomials.

1. Introduction

In this paper we present several modifications of existing rootfinding algorithms. They may be used for solving several kinds od problems such as finding singular values of matrices, eigenvalues of certain matrices, roots of some orthogonal polynomials.
The algorithms that we consider here are the Weierstrass method (see e.g. [16, 14, 11]), the Aberth method ([1]) and the simultaneous version of the Bairstow method ([10, 13]).
We will start by presenting the details of these problems. Then we will describe the algorithms and their modifications. These modifications can be applied to every simultaneous rootfinding algorithm.

2. Singular values and rootfinding methods

In this section we will describe how rootfinding methods can be used to compute the singular values of any matrix.

Let $A \in \mathbb{R}^{n\times n}$ have the singular values $\sigma_1 \geqslant \cdots \geqslant \sigma_n$. Recall that they are the square roots of the eigenvalues of the matrix A^TA. However, methods based on forming A^TA and computing its eigenvalues may lead to significant loss of accuracy, especially in small singular values.
On the contrary, the problem of computing singular values of bidiagonal matrices is very well conditioned in the sense of the relative error. By a bidiagonal matrix we mean a matrix with nonzero entries located only on the main diagonal and on the superdiagonal. Demmel and Kahan [4] (see also [2, p.90]) proved that all singular values of bidiagonal matrices may be computed with high relative accuracy.
The method we propose here is a variation of the approach first outlined by Golub and Kahan in [7]. They used the fact that any matrix $A \in \mathbb{R}^{n\times n}$ can be decomposed as

$$Q^TAP = B = \begin{pmatrix} c_1 & b_2 & & & \\ & c_2 & b_3 & & \\ & & \ddots & \ddots & \\ & & & c_{n-1} & b_n \\ & & & & c_n \end{pmatrix},$$

[1]Faculty of Mathematics and Information Science, Warsaw University of Technology, Pl. Politechniki 1, 00-661 Warsaw, POLAND, e-mail: wrubelki@wp.pl

where Q and P are orthogonal, which guarantees that the singular values of A and B are the same. This reduction can be obtained by the Lanczos process or using a finite sequence of Householder transformations, for details see e.g. [2, pp.81-82]. The latter algorithm is backward stable in the following sense. It can be shown that the computed $\widetilde{B}$ is the exact result obtained for a matrix $A+E$, where

$$\|E\|_F \leqslant cn^2\epsilon_M\|A\|_F,$$

where ϵ_M is machine precision and c is a constant of order unity, see [2, p.83].
Once we have B, we form a symmetric matrix

$$C = \begin{pmatrix} 0 & B^T \\ B & 0 \end{pmatrix} \in \mathbb{R}^{2n\times 2n}, \tag{1}$$

whose eigen values are $\pm\sigma_k$, $k=1,2,\ldots,n$. Permuting rows and columns of C leads to a symmetric tridiagonal matrix with zeros on the main diagonal,

$$T = P^TCP = \begin{pmatrix} 0 & c_1 & & & & \\ c_1 & 0 & b_2 & & & \\ & b_2 & 0 & c_2 & & \\ & & c_2 & 0 & \ddots & \\ & & & \ddots & \ddots & c_n \\ & & & & c_n & 0 \end{pmatrix}, \tag{2}$$

where P is a permutation matrix. Since T and C have the same spectrum, computation of the singular values of A is equivalent to computation of the eigenvalues of T.
As T is tridiagonal, $\det(T-zI)$ can be evaluated easily and in a stable way, so methods that compute eigenvalues as roots of the characteristic polynomial find a reasonabl applica tion here.
To compute the spectrum of T, Golub and Kahan used a method based on Sturm sequences, and some variations of QR algorithm, applied to the bidiagonal matrix. We will consider here simultaneous rootfinding methods, the Weierstrass, Aberth and Bairstow methods.

3. Simultaneous rootfinding methods

Let $f(z)$ be a polynomial

$$f(z) = a_m z^m + a_{m-1}z^{m-1} + \ldots + a_1 z + a_0, \tag{3}$$

where, $a_k \in \mathbb{C}$, for $k = 0,1,...,m$.
E. Durand [6] and I. O. Kerner [11] independently have proposed an iterative method for finding all zeros of $f(z)$ simultaneously. It was a rediscovery of a scheme presented by K. Weierstrass in [16]. This is the reason why this method appears in literature as either the Durand-Kerner method or the Weierstrass method.
Assume that $\alpha_1, \alpha_2, \ldots, \alpha_m$ are simple roots of $f(z)$ and $z_1^{(0)}, z_2^{(0)}, \ldots, z_m^{(0)} \in \mathbb{C}$ are distinct initial approximations to these zeros. The k+1st iterate of the Weierstrass method is computed according to the formula

$$z_i^{(k+1)} = z_i^{(k)} - \frac{f(z_i^{(k)})}{a_m \prod_{j=1,j\neq i}^{m}(z_i^{(k)} - z_j^{(k)})}, \tag{4}$$

for $i = 1,2,...,m$.
Observe that

$$a_m \prod_{j=1,j\neq i}^{m}(\alpha_i - \alpha_j) = f'(\alpha_i)$$

and

$$a_m \prod_{j=1,j\neq i}^{m}(z_i^{(k)} - z_j^{(k)}) \approx f'(z_i^{(k)})$$

for $z_i^{(k)}$ close to α_i. Hence the Weierstrass method can be interpreted as Newton's method for every root. Hence it should converge quadratically, provided that the roots are simple. Indeed, it is so, which was first shown by Kerner.
The Aberth method (introduced in [1]) uses the following formula to compute k+1st approximation

$$z_i^{(k+1)} = z_i^{(k)} - \frac{f(z_i^{(k)})}{f'(z_i^{(k)}) - \sum_{j=1,j\neq i}^{m}\frac{f(z_i^{(k)})}{(z_i^{(k)}-z_j^{(k)})}},$$

for $i = 1,2,....,m...$
It is a simultaneous version of the Halley method. Analogously, for simple roots, its order of convergence is 3. The argument was given by O. Aberth in [1].
To modify these methods to suit better computing singular values we take into account the fact that the eigenvalues of T are $\pm\sigma_k$, $k=1,2,\ldots,n$, i.e. they are opposite numbers. Hence we can compute the consecutive approximations of only half of them. The classical Weierstrass formula (4) may be modified to obtain

$$z_i^{(k+1)} = z_i^{(k)} - \frac{f(z_i^{(k)})}{2z_i^{(k)}\prod_{j=1,j\neq i}^{n}(z_i^{(k)} - z_j^{(k)})(z_i^{(k)} + z_j^{(k)})}, \tag{5}$$

for $i = 1,2,...,n$.
The complexity of the classical and modified algorithms counted as a number of complex multiplications is $2n^2$ and $4n^2$, respectively. We omitted here the cost of computing $f(z_i^{(k)})$.
Similar modification of the classical Aberth formula leads to the following iteration

$$z_i^{(k+1)} = z_i^{(k)} - \frac{f(z_i^{(k)})}{f'(z_i^{(k)}) - f(z_i^{(k)})\left(2\sum_{j=1,j\neq i}^{n}\frac{z_i^{(k)}}{(z_i^{(k)}-z_j^{(k)})(z_i^{(k)}+z_j^{(k)})} + \frac{1}{2z_i^{(k)}}\right)} \tag{6}$$

for $i = 1,2,...,n$.

The complexity of the classical and modified methods is respectively $4n^2$ and $2n(n+1)$, plus the cost of computing $f(z_i^{(k)})$ and $f'(z_i^{(k)})$.

In the Bairstow method we seek the quadratic factors $(z^2 - rz - q)$ of the polynomial $f(z)$. In our problems we have only opposite roots, so we may assume that all factors are of the form $(z^2 - q)$. We will use this assumption from now on and the algorithms will be presented only in this setting.

Let $(z^2 - r^{(0)}z - q^{(0)})$ be the approximation of the quadratic factor of $f(z)$.

Then

$$f(z) = (z^2 - q^{(0)})f_1(z) + Rz + S,$$

and

$$f_1(z) = (z^2 - q^{(0)})f_2(z) + R_1 z + S_1.$$

The Bairstow method can be written as

$$\begin{pmatrix} 0 \\ q^{(k+1)} \end{pmatrix} = \begin{pmatrix} 0 \\ q^{(k)} \end{pmatrix} - \begin{pmatrix} S_1 & R_1 \\ R_1 q^{(k)} & S_1 \end{pmatrix}^{-1} \begin{pmatrix} R \\ S \end{pmatrix},$$

for k = 0,1,...

Here we assumed that the initial and computed approximations of the coefficient of the middle term of all quadratic factors are zero.

To avoid deflation this process may be changed to compute simultaneously consecutive approximations of all quadratic factors of $f(z)$. This approach was proposed by D. C. Handscomb in [10], then further developed by W. S. Luk in [13].

Here we will present a simpler version of their algorithm, which we modified to suit better the problem we are interested in, i.e. polynomials with opposite roots.

First we need a method of suppression of computed quadratic factors of $f(z)$. Assume that a factor $(z^2 - \hat{q})$ has already been found. Then we want to carry on the Bairstow method on the deflated polynomial $\hat{f}(z) = f(z)/(z^2 - \hat{q})$ without explicitly constructing $\hat{f}(z)$. It is possible, because we only need the coefficients of the linear reminders of $\hat{f}(z)$ and $\hat{f}_1(z)$, where

$$\hat{f}(z) = (z^2 - q)\hat{f}_1(z) + \hat{R}z + \hat{S}$$

and

$$\hat{f}_1(z) = (z^2 - q)\hat{f}_2(z) + \hat{R}_1 z + \hat{S}_1.$$

D. C. Handscomb (see [10]) found that the relation between $R,\ S,\ R_1,\ S_1$ and $\hat{R},\ \hat{S},\ \hat{R}_1,\ \hat{S}_1$, is the following

$$\begin{aligned} l &= q - \hat{q}, \\ \hat{R} &= R\,l^3, \\ \hat{S} &= S\,l^3, \\ \hat{R}_1 &= (R_1 l - R)\,l^2, \\ \hat{S}_1 &= (S_1 l - S)\,l^2. \end{aligned}$$

We stated these formulae in simplified form, suited for our problem. They are much simpler than original ones (compare [13] or [10]).

The simultaneous version of the Bairstow method works as follows. Choose $q_i^{(0)},\ i = 1, 2, \ldots, n,$ the initial approximations to the coefficients q of the quadratic factors. Then apply one step of the Bairstow method to each of them, treating all other as computed factors and perform suppression as described above.

The whole algorithm:

Choose initial approximations $q_i^{(0)},\ i = 1, 2, \ldots, n,$
for $k = 1, 2, \ldots$ until convergence do
  for $i = 1, \ldots, n$ do
    find $R,\ S,\ R_1,\ S_1$ using $q_i^{(k)}$
    for $j = 1, \ldots, n,\ j \neq i$ do
      $l := q - \hat{q}$
      $\hat{R} := R\,l^3$
      $\hat{S} := S\,l^3$
      $\hat{R}_1 := (R_1 l - R)\,l^2$
      $\hat{S}_1 := (S_1 l - S)\,l^2$
    end
    find $q_i^{(k+1)}$ by the Bairstow method
  end
end

4. Zeros of orthogonal polynomials

The above modification can also be used for computing the zeros of certain orthogonal polynomials.

Any set of orthogonal polynomials $\{p_0(x), p_1(x), \ldots, p_m(x)\}$ satisfies a three term recurrence relation

$$p_k(x) = (p_k x + q_k)p_{k-1}(x) - r_k p_{k-2}(x), \qquad k = 1, 2, \ldots, m, \tag{7}$$

with $p_{-1}(x) \equiv 0,\ p_0(x) \equiv 1$, where $p_k > 0,\ r_k > 0$.

It can be shown (see [17] and [9]), that the zeros of $p_m(x)$ are the eigenvalues of

$$T = P^T C P = \begin{pmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_{m-1} \\ & & & \beta_{m-1} & \alpha_m \end{pmatrix},$$

where

$$\alpha_i = -\frac{q_i}{p_i}, \qquad \beta_i = \left(\frac{r_{i+1}}{p_i p_{i+1}}\right)^{1/2}.$$

Zeros of many orthogonal polynomials are opposite numbers, hence our modifications of rootfinding methods can be applied here.

5. Numerical tests

This paragraph contains the results of the tests performed in Matlab, with machine precision $\epsilon_M \approx 2.2 \cdot 10^{-16}$.

We implemented the modified Weierstrass method and compared the results it gave either with the exact solution or with the results given by the built-in Matlab function "svd". In all tests we computed the componentwise relative

error

$$\max_{k=1,\ldots,n} \frac{|\sigma_k - \tilde{\sigma}_k|}{\sigma_k}, \tag{8}$$

where σ_k and , $\tilde{\sigma}_k$, $k = 1, \ldots, n$ are the respective exact (or given by "svd") and computed singular values of a matrix $B \in \mathbb{R}^{n\times n}$.

The situation where we know the exact solution is quite rare. However, there is one class of matrices that is very useful for tests, namely the Clement matrices, see [3]. These are matrices of the form

$$T_m = \begin{pmatrix} 0 & t_1 & & & \\ t_1 & 0 & t_2 & & \\ & t_2 & 0 & \ddots & \\ & & \ddots & \ddots & t_{m-1} \\ & & & t_{m-1} & 0 \end{pmatrix}, \tag{9}$$

where $t_k = \sqrt{k(m-k)}$ for $k = 1, \ldots, m$. The spectrum of T_m consists of opposite integer numbers and possibly zero, if m is odd, i.e. $\sigma(T_m) = \{\pm(m-1), \pm(m-3), \ldots, \pm 1\}$ or $\sigma(T_m) = \{\pm(m-1), \pm(m-3), \ldots, 0\}$.

For our tests it is convenient to assume that T_{2n} is a matrix of the form (2), obtained (via bidiagonalization, forming a matrix C (see (1)) and permuting its rows and columns), from some unknown matrix $A \in \mathbb{R}^{n\times n}$, whose singular values are the eigenvalues of T_{2n}. Note that $\sigma(T_{2n}) = \{\pm(2n-1), \pm(2n-3), \ldots, \pm 1\}$ and, as T_{2n} is symmetric, its condition number is equal to $2n-1$, so T_{2n} is well-conditioned if the dimension is not too big.

Figure 1 presents the results of tests of the Weierstrass method for Clement matrices. The value of the logarithm of the relative error (8) at the final step is plotted against the half of the dimension of the Clement matrix. As initial approximations we took equally distributed points from the interval $[\|T_{2n}\|_\infty/(n+1); \|T_{2n}\|_\infty]$ and numbers opposite tothem. As a termination criterion we used Gill's criterion

$$\text{STOP if} \quad \|s^{(k+1)} - s^{(k)}\|_\infty < \delta \|s^{(k+1)}\|_\infty + \epsilon_M, \tag{10}$$

where $s^{(k)}$ is a vector of approximations at the kth step, δ is a tolerance and ϵ_M is a machine precision. In all tests with Clement matrices $\delta = 1e{-}12$.

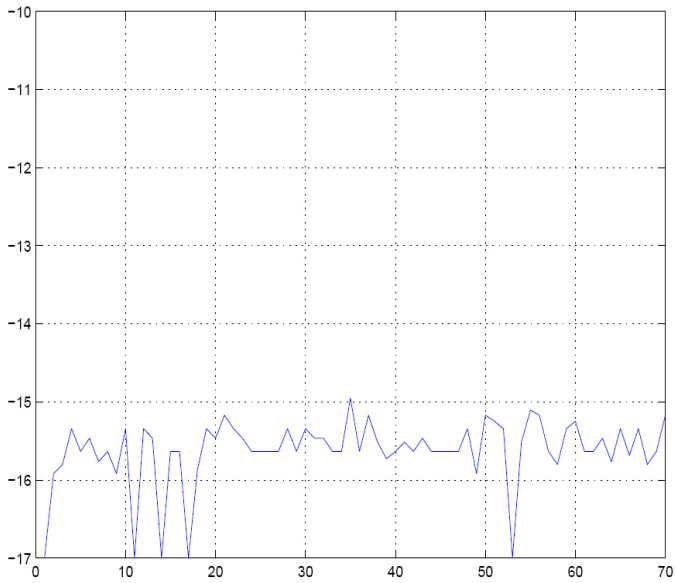

Figure 1: The logarithm of the relative error (8) of the Weierstrass method for Clement matrices.

We can see that the results are very satisfactory. However, the situation can change significantly if we change initial approximations. Figure 2 presents the results for analogous tests with different initial guess, namely equally distributed points from the interval $[\|T_{2n}\|_\infty/n; \|T_{2n}\|_\infty]$ and numbers opposite to them.

We tried many different starting approximations and it turned out that very often the method breaks down for some n. The values of n for which the method does not work strongly depend on the initial guess. Moreover, large matrices also present a problem, for $n > 150$ it is hard to find initial guess for which the method works.

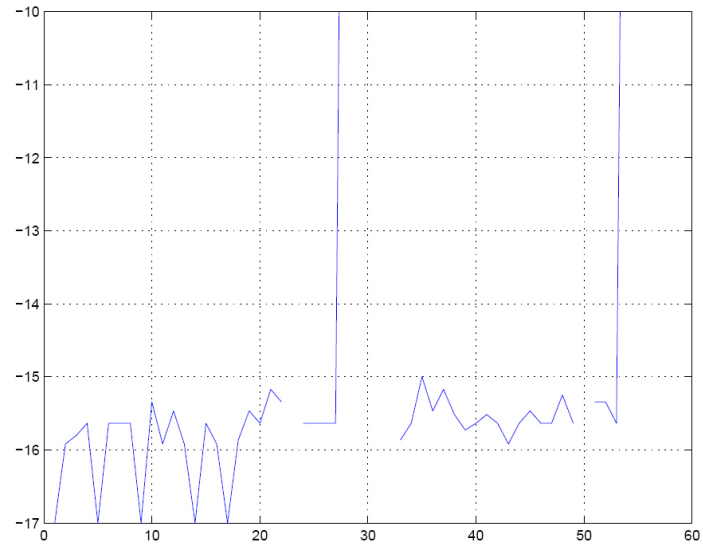

Figure 2: The logarithm of the relative error (8) of the Weierstrass method for Clement matrices.

These breakdowns are mainly due to the overflow, so some method of scaling may help to improve the behaviour of this algorithm.

On the other hand, the method is very fast, as predicted. The number of iterations required to obtain desired accuracy grows slowly with n, for $n = 150$ in our test it was

8.
The same tests we performed for the Aberth method. The results and conclusions are similar. This method is also very sensitive to the choice of initial guess. And, if it works, it is usually slightly faster than the Weierstrass method, which is not surprising, as its order of convergence is also greater.
In the case of the Bairstow method the situation is different. The method works fine and is not too sensitive for the choice of initial guess. However, for larger matrices it converges very slowly or not at all, which shows already for 40×40 matrices. This may be partially caused by the fact, that the inner loop in the algorithm is performed for all $j = 1, \ldots, n,\ j \neq i$, i.e. all factors are suppressed, except the current updated one. If, instead of all, we suppress only a couple of factors around the current one, it may in some cases improve the convergence.
The next group of tests was performed on matrices of the form

$$B = \begin{pmatrix} 1 & b & & \\ & 1 & b & \\ & & 1 & b \\ & & & 1 \end{pmatrix}, \qquad b \in \mathbb{R}. \tag{11}$$

This time we computed the singular values of B by forming an appropriate matrix $T \in \mathbb{R}^{8\times 8}$ (see (2)) and, as the exact results are not known, we compared the results given by the modified Weierstrass algorithm with the ones given by the Matlab function "svd".
For larger values of b one of the singular values of B is significantly smaller than the others, which results in loss of accuracy. This is already visible for $b \geqslant 10$. Hence iterative refinement is recommended, for example by means of Newton's method applied to the smallest singular value.
The Tables 1, 2 and 3 present the results for matrices of the above form (11) for different values of the parameter b, respectively, for the modified Weierstrass, Aberth and Bairstow methods. As b increases, the condition number grows and results become less accurate. The last two columns contain the values of the relative componentwise error

$$\max_{k=1,\ldots,4} \frac{|\sigma_k(svd) - \tilde{\sigma}_k|}{\sigma_k(svd)},$$

and the absolute componentwise error

$$\max_{k=1,\ldots,4} |\sigma_k(svd) - \tilde{\sigma}_k|,$$

where $\sigma_k(svd)$ and , $\tilde{\sigma}_k,\ k = 1, \ldots, 4$ are the singular values computed by the "svd" function and the Weierstrass or the Aberth algorithm, respectively. The tolerance δ (see (10)) in these tests was $1e{-}13$, the initial approximations were chosen as before.

Table 1: Results for the modified Weierstrass method for the matrix (11) for different values of b.

b	cond(B)	iterations	relative error	absolute error
0.01	1.0163	31	$1.1192e{-}16$	0
0.1	1.1757	14	$2.1483e{-}16$	0
10	$1.0850e{+}04$	19	$6.5709e{-}16$	$6.5052e{-}19$
100	$1.0072e{+}08$	32	$2.1177e{-}16$	$1.4210e{-}14$
1000	$1.0007e{+}12$	43	$2.3524e{-}04$	$1.1368e{-}13$

In Table 2 we see that the Aberth method did not work for $b > 8.99$. This is only for this particular initial guess. For some initial approximations this method worked well: the results were very close to these given by the function "svd" (error of order $1e{-}16$). The problem is to find such initial guess, as it very strongly depends on the matrix. We were not able to find a general formula that would work for a family of matrices.

Table 2: Results for the modified Aberth method for the matrix (11) for different values of b.

b	cond(B)	iterations	relative error	absolute error
0.01	1.0163	10	$1.1193e{-}16$	$1.1102e{-}16$
0.1	1.1757	10	$2.1483e{-}16$	$2.2204e{-}16$
7	$2.7149e{+}03$	18	$3.0370e{-}16$	$1.7764e{-}15$
8.98	$7.1304e{+}03$	68	$3.9201e{-}16$	$3.5527e{-}15$
$\geqslant 8.99$		no convergence		

Table 3: Results for the modified Bairstow method for the matrix (11) for different values of b.

b	cond(B)	iterations	relative error	absolute error
0.01	1.0163	10	$1.9644e{-}11$	$3.9290e{-}11$
0.1	1.1757	6	$8.5244e{-}14$	$1.7112e{-}13$
10	$1.0850e{+}04$	7	$5.4864e{-}16$	$9.5659e{-}15$
100	$1.0072e{+}08$	8	$2.1918e{-}13$	$3.7966e{-}11$
1000	$1.0007e{+}12$	>9	$2.2341e{-}11$	$3.8696e{-}08$
10000	$1.0000e{+}16$	13	$3.9409e{-}09$	$6.8258e{-}05$

In the test for $b = 1000$ in Table 3 errors of the results given by the Bairstow method were the same for any number of iterations between 9 and 1000.
Next group of tests is involves orthogonal polynomials. These tests were performed for Chebyshev polynomials, whose zeros are known, which makes it possible to verify the results.
Table 4 presents results for the Chebyshev polynomials. The first column contains the degree of the polynomial, the second and the third contain the relative error and number of iterations for the Weierstrass method, the last two columns present the same data for the Aberth method. The condition numbers of corresponding matrices are quite modest, they do not exceed 100. We used analogous initial approximations as in the case of Clement matrices, and the same stopping criterion with the tolerance $1e{-}13$.

Table 4: The relative errors of the modified Weierstrass (W) and Aberth (A) methods for Chebyshev polynomials.

n	iterations (W)	relative error (W)	iterations (A)	relative error(A)
10	14	$1.2161e{-}16$	8	$1.2161e{-}16$
20	12	$1.2482e{-}16$	7	$1.2482e{-}16$
30	21	$1.2300e{-}16$	12	$1.2132e{-}16$
50	198	$1.3839e{-}16$		no convergence
100		no convergence	30	$1.5338e{-}16$
150		no convergence	37	$1.3554e{-}16$

6. Conclusions

In this paper we presented certain modifications of classical rootfinding algorithms. These modified methods may be used for example for computing the singular values of matrices or roots of orthogonal polynomials.
Numerical tests performed in Matlab show that these algorithms have good numerical properties, if the zeros are well separated and the initial guess is sufficiently close to the exact solution. Therefore it can be recommended as a method of choice, especially if combined with some techniques of iterative refinement. Moreover, in all algorithms, the classical versions and the modified ones, some parts are computed independently. Hence parallel implementation may be a good choice here.

References (Literatura)

[1] O. Aberth, Iteration methods for finding all zeros of a polynomial simultaneously, Math. Comput. 27 (122) (1973), 339-344.
[2] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA, USA, 1996.
[3] P. A. Clement, A class of triple-diagonal matrices for test purposes, SIAM Review 1 (1) (1959), 50-52.
[4] J. Demmel and W. Kahan, Accurate singular values of bidiagonal matrices, SIAM J. Sci. Stat. Comput. 11 (1990), 873-912.
[5] K. Dočev, Modified Newton method for the simultaneous approximate calculation of all roots of a given algebraic equation, Mat. Spis. Bulgar. Akad. Nauk 5 (1962), 136-139, (in Bulgarian).
[6] E. Durand, *Solutions Numériques des Équations Algébraiques, Tome I: Equations du Type $F(x) = 0$: Racines d'un Polynôme*, Masson, Paris, 1960.
[7] G. H. Golub and W. Kahan, Calculating the singular values and pseudo-inverse of a matrix, SIAM J. Num. Anal. Ser. B, 2 (2) (1965), 205-224.
[8] G. H. Golub and Ch. F. Van Loan, *Matrix Computations*, Third Edition, The Johns Hopkins University Press, Baltimore and London, 1996.
[9] G. H. Golub and J. H. Welsch, Calculation of Gauss quadrature rules, Math. Comp., 23 (1969), 221-230.
[10] D. C. Handscomb, Computation of the latent roots of a Hessenberg matrix by Bairstow's method, Computer J. 5 (1962), 139-141.
[11] I. O. Kerner, Ein Gesamtschrittverfahren zur Berechnung der Nullstellen von Polynomen, Numer. Math. 8 (1966), 290-294.
[12] T. Y. Li, N. H. Rhee, Z. Zeng, An efficient and accurate parallel algorithm for the singular value problem of bidiagonal matrices, Numer. Math. 69 (1995), 283-301.
[13] W. S. Luk, Finding roots of real polynomial simultaneously by means of Bairstow's method, BIT 35 (2) (1996), 302-308.
[14] M. Petković, *Iterative Methods for Simultaneous Inclusion of Polynomial Zeros*, Springer-Verlag, Berlin, New York, 1989.
[15] G. W. Stewart, *Introduction to Matrix Computations*, Academic Press, New York, 1973.
[16] K. Weierstrass, Neuer Beweis des Fundamentalsatzes der Algebra, in *Mathematische Werker, Tome III*, Mayer und Mueller, Berlin, 1903, 251-269.
[17] H. Wilf, *Mathematics for Physical Sciences*, Wiley, New York, 1962.