Ann. Acad. Rom. Sci.
Ser. Math. Appl.
ISSN 2066-6594 Vol. 13, No. 1-2/2021

AN ITERATIVE METHOD FOR DIAGONALIZATION OF THE FROBENIUS COMPANION MATRIX*

Gyurhan H. Nedzhibov†

DOI https://doi.org/10.56082/annalsarscimath.2021.1-2.45

Abstract

In this work, we develop a new and efficient iterative method for diagonalization of the Frobenius companion matrix. The method can be used for approximating all of the eigenvalues and corresponding eigenvectors. It can also be used for simultaneous inclusion of all simple zeros of the corresponding characteristic polynomial. Local convergence analysis of the method is included. We prove that it is locally quadratically convergent. Some numerical examples demonstrating effectiveness of the proposed iterative method are also included.

MSC: 65H04, 65H05, 65F15

keywords: Frobenius companion matrix, Eigen-decomposiotion, Iterative diagonalization, Polynomial zeros, Rayleigh quotient.

1 Introduction

This section gives a very short overview of the theory from the linear algebra concerning the matrix spectral decomposition and diagonalization.

*Accepted for publication on March 9-th, 2020

†g.nedzhibov@shu.bg Faculty of Mathematics and Informatics, Shumen University, Shumen 9700, Bulgaria; Paper written with financial support of Shumen University under Grant RD 08-93/01.02.2019.

1.1 Eigenvalues, eigenvectors and eigendecomposition

We begin by reviewing some basic terminology and definitions (see [1, 2]).

Definition 1 *Let $A \in \mathbf{C}^{n\times n}$. A complex number λ is an eigenvalue of the matrix A if there exists a non-zero vector $v \in \mathbf{C}^n$ such that $Av = \lambda v$. The vector v is a (right) eigenvector associated to the eigenvalue λ and (λ, v) is called an eigenpair of A.*

A nonzero vector $w \in \mathbf{C}^n$ such that $w^*A = \lambda w^*$ is called a *left eigenvector* associated to λ (recall that $w^* = (\overline{w})^T$ is the *conjugate transpose of* w). The set $\sigma(A) = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ of all eigenvalues of the matrix A is reffered to as *spectrum of* A.

Eigenvalues and eigenvectors are a standard tool in the mathematical sciences and in scientific computing. They have many applications, particularly in physics and engineering. Control theory, vibration analysis, electric circuits, advanced dynamics and quantum mechanics are just a few of the application areas. Eigenvalue decompositions play also an important role in the analysis of many numerical methods. Many of the applications involve the use of eigenvalues and eigenvectors in the process of reduction square matrices into matrices that have simpler form. By reduction we mean a transformation that preserves the eigenvalues of a matrix.

Definition 2 *We say that two matrices A and B are similar if there exists a nonsingular matrix X such that $B = XAX^{-1}$.*

The mapping $B \to A$ is called a *similarity transformation.* It is equivalence transformation and preserves the eigenvalues of matrix.

The simplest form in which a matrix can be reduced is undoubtedly the diagonal form.

Definition 3 *The eigenvalue decomposition (spectral decomposition) of the matrix $A \in \mathbf{C}^{n\times n}$ is its factorization in the form*

$$A = S\Lambda S^{-1},$$

where $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$ has all elements of the spectrum on its diagonal and S is a non-singular matrix whose columns are eigenvectors associated with $\lambda_1, \lambda_2, \ldots, \lambda_n$, respectively. Likewise, every row of S^{-1} contains coordinates of a single (left) eigenvector.

But this reduction not always possible, and there are matrices that have no eigenvalue decomposition. A matrix that can be reduced to the diagonal form is called *diagonalizable.* It is well known that a matrix is diagonalizable if it has all distinct eigenvalues, the converse is not necessarily true.

1.2 Polynomial rootfinding using companion matrices

Computing roots of polynomials via eigen solving problem is a classical approach recently revived (see [3, 4, 5, 6] and the bibliography therein).

Definition 4 *The characteristic polynomial of $A \in \mathbf{C}^{n\times n}$, denoted $P_A(z)$ for $z \in \mathbf{C}$, is the degree n polynomial defined by*

$$P_A(z) = \det(A - zI).$$

The matrix A is called *companion matrix* of the polynomial $P_A(z)$. It follows that the roots of the characteristic polynomial of a matrix are exactly the eigenvalues of the matrix, since the matrix $A - \lambda I$ is singular precisely when λ is an eigenvalue of A. We have seen that eigenvalues may be found by solving polynomial equations. The converse is also true.

Let be given a monic polynomial $p(z)$ of degree n

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

with $a_i \in \mathbf{C}$ $(i = 0, \ldots, n-1)$. The companion matrix of $p(z)$ is defined as

$$F_p = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ddots & \ldots \\ 0 & 0 & 0 & \ldots & 1 \\ -a_0 & -a_1 & -a_2 & \ldots & -a_{n-1} \end{pmatrix} \tag{2}$$

which is called *Frobenius companion matrix*. Then the eigenvalues of F_p can be computed by applying certain matrix methods. This is the approach followed by MATLAB command `roots` which uses the QR-algorithm on the balanced Frobenius companion matrix to get its eigenvalues.

In addition if we assume that the polynomial (1) has **n simple roots** $\lambda_1, \lambda_2, \ldots, \lambda_n$, then the corresponding Frobenius companion matrix (2) is diagonalizable. In this case, it is immediate to verify that if λ_s is an eigenvalue of F_p, then the eigenvector associated with λ_s has the form

$$\mathbf{u}_s = (1, \lambda_s, \lambda_s^2, \ldots, \lambda_s^{n-1})^T.$$

Then the eigenvector matrix V actually equals

$$V = V(\lambda) = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \lambda_1 & \lambda_2 & \ldots & \lambda_n \\ \ldots & \ldots & \ddots & \ldots \\ \lambda_1^{n-1} & \lambda_2^{n-1} & \ldots & \lambda_n^{n-1} \end{pmatrix}, \tag{3}$$

which is the well-known *Vandermonde matrix*. Then we have the following eigenvalue decomposition of (2)

$$F_p = V\Lambda V^{-1}. \tag{4}$$

Now the Fobenius companion matrix (2) has complete *biorthogonal systems* of right eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ (the columns of V) and left eigenvectors $\mathbf{w}_1^*, \mathbf{w}_2^*, \ldots, \mathbf{w}_n^*$ (the rows of V^{-1}), i.e.

$$\mathbf{w}_i^*\mathbf{v}_j = \delta_{ij} \quad \text{and} \quad \mathbf{w}_i^* F_p \mathbf{v}_j = \lambda_i \delta_{ij}, \tag{5}$$

for each $i, j = 1, \ldots, n$.

2 Diagonalization of Frobenius companion matrix

The algorithms for the eigenvalue problem are usually distinguished between *direct (full space) methods* and *iterative (subspace) methods*. Direct methods are intended to compute the complete set of eigenvalues and, if necessary, the eigenvectors. Note that direct methods are also of iterative nature, since as we mentioned finding eigenvalues is equivalent to finding zeros of polynomials, for which noniterative methods can not exist. Since direct methods usually transform the original matrices to diagonal or triangular form by applying transformation matrices, they have a complexity of $\mathcal{O}(n^3)$ and therefore have a limited range of applications. Iterative methods typically provide approximations only to a subset of the eigenvalues and, if necessary, the corresponding eigenvectors. These methods work only with matrix vector products on the original matrix, which is inexpensive if it is sparse and hence they can theoretically be applied to large sparse matrices of unlimited size.

In this paper we explore a new approach for diagonalization of a special subclass of the Frobenius companion matrix, namely when all of its eigenvalues are distinct. This new iterative scheme may be classified as direct method, rather than an iterative method. Frobenius companion matrix is important in theory, numerical computations and in applications. It is used to find bounds on eigenvalues of matrices and also is used in algorithms for finding roots of polynomials. Companion matrices are widely used in control theory and signal processing, for example, in the observable canonical form as well as the controllable canonical form (see for example [7, 8] and the references therein).

2.1 Two-sided Rayleigh quotient

Before to state the new iterative scheme we need to introduce the *two-sided Rayleigh quotient*, which will play an important role in the algorithm.

Definition 5 *The two-sided Rayleigh quotient of a square matrix $A \in \mathbf{C}^{n\times n}$ and nonzero vectors $x, y \in \mathbf{C}^n$ is*

$$\rho(x,y) = \rho(A,x,y) = \frac{y^*Ax}{y^*x},$$

provided $y^*x \neq 0$.

The two-sided Rayleigh quotient is introduced by Ostrowski in [9] and have been used in a number of papers (see for example [10, 11, 12, 13]). It has the following basic and known properties:

- Homogenity: $\rho(\alpha x, \beta y, \gamma A) = \gamma\rho(x,y,A)$ for all $\alpha, \beta, \gamma \in \mathbf{C}$.
- Translation invariance: $\rho(x,y,A-\alpha I) = \rho(x,y,A) - \alpha$for all $\alpha \in \mathbf{C}$.
- Stationarity: $\rho(x,y,A)$ is stationary iff x and y are right and left eigenvectors of A with eigenvalue ρ and $y^*x \neq 0$.

If v and w are right and left eigenvectors of F_p associated with same eigenvalue, then from the definition of two-sided Rayleigh quotient and the biorthogonality condition (5) it follows that

$$\rho(v,w) = \rho(F_p, v, w) = \frac{w^*F_pu}{v^*u} = w^*F_pv, \tag{6}$$

where F_p is the Frobenius companion matrix (2) with n distinct eigenvalues.

2.2 Description of the new iterative algorithm

Let F_p be the Frobenius companion matrix (2) with n distinct eigenvalues

$$\lambda_1, \lambda_2, \ldots, \lambda_n$$

and let denote

$$z^{(k)} = (z_1^{(k)}, z_2^{(k)}, \ldots, z_n^{(k)})$$

the approximations of the corresponding eigenvalues. We will denote by $\mathbf{v}_i^{(k)}$ and $\mathbf{w}_i^{(k)}$ the right and left eigenvectors, respectively, associated to eigenvalue λ_i for $i = 1, 2, \ldots, n$. By right upper index in brackets $^{(k)}$ is

Algorithm 1: Diagonalization of the Frobenius companion matrix

Input: F_p, initial vector $\mathbf{z}^{(0)}$(where $z_i^{(0)} \neq z_j^{(0)}$ for $i \neq j$), tolerance $\epsilon \ll 1$.
Output: Approximate eigenvalues and eigenvector matrices:
$\lambda = (\lambda_1, \dots, \lambda_n)$, $V(\lambda)$, $W(\lambda)$.
1: Set k=0.
2: **While** not converged **do**
3: Compute the Vandermonde matrix $V_k = V(\mathbf{z}^{(k)})$.
4: Compute the inverse Vandermonde matrix $W_k^* = V(\mathbf{z}^{(k)})^{-1}$.
5: **For** i=1:n **do**
6: Compute next eigenvalue estimate
$z_i^{(k+1)} = \rho_k = (\mathbf{w}_i^{(k)})^* F_p \mathbf{v}_i^{(k)}$,
where $\mathbf{v}_i^{(k)}$ is the i-*th* column vector of V_k,
and $(\mathbf{w}_i^{(k)})^*$ is the i-*th* row vector of W_k^*
7: **End for**
8: **Set** k=k+1.
8: **If** $\|\mathbf{z}^{(k+1)} - \mathbf{z}^{(k)}\| < \epsilon$ **then**
9: Set $\lambda = \mathbf{z}^{(k+1)}$, $V = V_k$, $W = W_k^*$.
10: **break**
11: **End If**
12: **End While**

denoted the iteration index. We study an iterative process of the following form (see Algorithm 1).

The basic idea of our approach is combining the following two operations:

1. We construct approximate eigenvector matrices from the set of estimated eigenvalues, by using the special structure of the eigenvectors of Frobenius companion matrix (see (3)).

2. The Rayleigh quotient produces an approximate eigenvalue when approximated left and right eigenvectors are given.

Note that on step 6, we use the Rayleigh quotient of the form (6) because on each iteration step we get a complete biorthogonal systems of approximate right and left eigenvectors. Which means that the conditions (5) also hold for the approximate eigenvectors.

3 Convergence analysis

In this section we will prove that the asymptotic rate of convergence of the presented algorithm is quadratic.

For the sake of brevity and without loosing generality for the remainder of this text we fix the value of i and use the following notations:

- $\lambda = \lambda_i$ - the $i-$th eigenvalue of F_p;
- $z_k = z_i^{(k)}$ - the approximate value of λ_i after k iterations;
- $\mathbf{v}_k = \mathbf{v}_i^{(k)}$- approximate right eigenvector of λ_i after k iterations;
- $\mathbf{w}_k = \mathbf{w}_i^{(k)}$- approximate left eigenvector of λ_i after k iterations;
- $\rho_k = \rho(\mathbf{v}_k, \mathbf{w}_k) = \mathbf{w}_k^* F_p \mathbf{v}_k$ - the Rayleigh quotient of $\mathbf{v}_i^{(k)}$ and $\mathbf{w}_i^{(k)}$;
- the vector norm $\|.\|$ is the Euclidean norm $\|a\| = \sqrt{a^* a}$ and the norm of matrices is the subordinated spectral norm.

First, we will state and prove certain auxiliary results.

Lemma 1 *Let $\mathbf{x}$ and $\mathbf{y}$ be right and left eigenvectors associated with eigenvalue λ of the Frobenius companion matrix F_p, which has n distinct eigenvalues. If the corresponding approximate eigenvectors $\mathbf{v}_k$ and $\mathbf{w}_k$ $(k = 1, 2, \ldots)$ are of the form*

$$\mathbf{v}_k = \tau \mathbf{x} + \mathbf{d}_k \quad \textit{and} \quad \mathbf{w}_k = \theta \mathbf{y} + \mathbf{s}_k, \tag{7}$$

where $\tau, \theta \in \mathbf{C}$ and $\mathbf{y}^* \mathbf{d}_k = \mathbf{s}_k^* \mathbf{x} = 0$. Then

$$(i)\ \rho_k - \lambda = \mathbf{w}_k^*(F_p - \lambda I)\mathbf{v}_k = \mathbf{s}_k^*(F_p - \lambda I)\mathbf{d}_k, \tag{8}$$

$$(ii)\ \rho_k - \lambda = \mu(\rho_{k-1} - \lambda), \tag{9}$$

where $\mu = 1 + (\rho_{k-1} - \lambda)^{-1}\mathbf{u}_k$ and $\mathbf{u}_k = \mathbf{w}_k^*(F_p - \rho_{k-1} I)\mathbf{v}_k$.

Proof. Firstly, we recall that the presentations (7) are unique by using spectral projections. Namely the spectral projection of $\mathbf{v}_k$ onto column eigenspace of λ is in the direction of $\mathbf{x}$. Moreover, the column eigenspace of λ has unique invariant complement which is orthogonal to $\mathbf{y}$. Therefore, there is an unique presentation of the form $\mathbf{v}_k = \tau \mathbf{x} + \mathbf{d}_k$, where $\mathbf{d}_k \perp \mathbf{y}$ and $\tau = \mathbf{y}^* \mathbf{v}_k$. Similarly, it follows that there is an unique presentation of the form $\mathbf{w}_k = \theta \mathbf{y} + \mathbf{s}_k$, where $\mathbf{s}_k \perp \mathbf{x}$ and $\theta = \mathbf{x}^* \mathbf{w}_k$.

The assertion (i) follows from (7), definition of ρ_k and the biorthogonality of $\mathbf{v}_k$ and $\mathbf{w}_k$

$$\begin{aligned}\rho_k-\lambda &= \mathbf{w}_k^*F_p\mathbf{v}_k-\lambda=\mathbf{w}_k^*(F_p-\lambda I)\mathbf{v}_k=(\bar{\theta}\mathbf{y}^*+\mathbf{s}_k^*)(F_p-\lambda I)(\tau\mathbf{x}+\mathbf{d}_k)\\ &= \bar{\theta}\tau\mathbf{y}^*(F_p-\lambda I)\mathbf{x}+\bar{\theta}\tau\mathbf{y}^*(F_p-\lambda I)\mathbf{d}_k+\\ &\quad +\tau\mathbf{s}_k^*(F_p-\lambda I)\mathbf{x}+\mathbf{s}_k^*(F_p-\lambda I)\mathbf{d}_k\\ &= \mathbf{s}_k^*(F_p-\lambda I)\mathbf{d}_k.\end{aligned}$$

Part (ii) follows from (i)

$$\begin{aligned}\rho_k-\lambda &= \mathbf{w}_k^*(F_p-\lambda I)\mathbf{v}_k=\mathbf{w}_k^*(F_p-\rho_{k-1}I)\mathbf{v}_k+(\rho_{k-1}-\lambda)\\ &= (\rho_{k-1}-\lambda)(1+(\rho_{k-1}-\lambda)^{-1}\mathbf{w}_k^*(F_p-\rho_{k-1}I)\mathbf{v}_k)\\ &= (\rho_{k-1}-\lambda)(1+(\rho_{k-1}-\lambda)^{-1}\mathbf{u}_k).\end{aligned}$$

The lemma is proved.

Note that the second expression in Lemma 1 can be used to derive the local convergence neighborhoods for the suggested algorithm. In the next lemma we will prove that $(\rho_{k-1}-\lambda)^{-1}\mathbf{u}_k$ in (9) is bounded.

Lemma 2 *Let the assumptions of Lemma 1 be satisfied. Then*

(i) the vector $\mathbf{t}_k=(\rho_{k-1}-\lambda)^{-1}(F_p-\rho_{k-1}I)\mathbf{v}_k$ *is bounded and*

$$\mathbf{t}_k\to(0,0,\dots,0,P_F'(\lambda))^T \quad \textit{as} \quad \rho_{k-1}\to\lambda,$$

where $P_F(z)$ *is the characteristic polynomial of* F_p.

(ii) the vector $\mathbf{t}_k+\tau\mathbf{x}$ *is orthogonal to* $\mathbf{y}$.

Proof. (i) Using $\mathbf{v}_k=(1,\rho_{k-1},\dots,\rho_{k-1}^{n-1})^T$, it is easy to verify that

$$(F_p-\rho_{k-1}I)\mathbf{v}_k=(0,0,\dots,0,-P_F(\rho_{k-1}))^T,$$

hence

$$\mathbf{t}_k=\left(0,0,\dots,0,-\frac{P_F(\rho_{k-1})}{\rho_{k-1}-\lambda}\right)^T.$$

Applying the L'Hospital rule we get

$$\frac{P_F(\rho_{k-1})}{\rho_{k-1}-\lambda}\to P_F'(\lambda)$$

as $\rho_{k-1}\to\lambda$. Which proves the first assertion.

(*ii*) Substitution of $\mathbf{v}_k = \tau\mathbf{x} + \mathbf{d}_k$ in the expression of $\mathbf{t}_k$, gives

$$\mathbf{t}_k = \frac{(F_p - \rho_{k-1}I)\mathbf{v}_k}{\rho_{k-1} - \lambda} = -\tau\mathbf{x} + \frac{(F_p - \rho_{k-1}I)\mathbf{d}_k}{\rho_{k-1} - \lambda},$$

which is equivalent to

$$\mathbf{t}_k + \tau\mathbf{x} = \frac{(F_p - \rho_{k-1}I)\mathbf{d}_k}{\rho_{k-1} - \lambda}. \tag{10}$$

Using that $\mathbf{d}_k \perp \mathbf{y}$, it is easy to verify that

$$\mathbf{y}^* \frac{(F_p - \rho_{k-1}I)\mathbf{d}_k}{\rho_{k-1} - \lambda} = 0,$$

which yields $(\mathbf{t}_k + \tau\mathbf{x}) \perp \mathbf{y}$. The lemma is proved.

Corollary 1 *Under the assumptions of Lemma 1, it follows that the vector* $\mathbf{p}_k + \theta\mathbf{y}$ *is orthogonal to* $\mathbf{x}$*, where* $\mathbf{p}_k = (\rho_{k-1} - \lambda)^{-*}(F_p - \rho_{k-1}I)^*\mathbf{w}_k$.

Proof. The proof is similar to the proof of assertion (ii) in Lemma 2.

Lemma 3 *Let the assumptions of Lemma 1 hold. Then*

$$(i)\ \mathbf{d}_k = (\rho_{k-1} - \lambda)\tilde{\mathbf{d}}_k, \quad \textit{where} \quad \tilde{\mathbf{d}}_k = (F_p - \rho_{k-1}I)^{-1}(\mathbf{t}_k + \tau\mathbf{x}), \tag{11}$$

$$(ii)\ \mathbf{s}_k^* = (\rho_{k-1} - \lambda)^*\tilde{\mathbf{s}}_k^*, \quad \textit{where} \quad \tilde{\mathbf{s}}_k = (F_p - \rho_{k-1}I)^{-*}(\mathbf{p}_k + \theta\mathbf{y}). \tag{12}$$

Proof. (*i*) From the expression (10), it follows that

$$\mathbf{d}_k = (\rho_{k-1} - \lambda)(F_p - \rho_{k-1}I)^{-1}(\mathbf{t}_k + \tau\mathbf{x}),$$

which implies (11). Similarly, we get the expression (12) from Corollary 1.

Note that as $\rho_{k-1} \to \lambda$, the operator $(F_p - \lambda I)^{-1}$ becomes unbounded. However, we are only interested in applying $(F_p - \lambda I)^{-1}$ to vectors in *span*$\{\mathbf{y}^\perp\}$ and in applying $(F_p - \lambda I)^{-*}$ to vectors in *span*$\{\mathbf{x}^\perp\}$. Therefore $(F_p - \lambda I)^{-1} : \mathbf{y}^\perp \to \mathbf{y}^\perp$ is bounded on $\mathbf{y}^\perp$ and $(F_p - \lambda I)^{-*} : \mathbf{x}^\perp \to \mathbf{x}^\perp$ is bounded on $\mathbf{x}^\perp$. The proof is completed.

Now we can prove the main theorem in this section.

Theorem 1 *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be right and left eigenvectors associated with eigenvalue* λ *of the Frobenius companion matrix* F_p*, which has* n *distinct eigenvalues. Then* $\lim_{k\to\infty} \mathbf{v}_k = \mathbf{x}$ *and* $\lim_{k\to\infty} \mathbf{w}_k = \mathbf{y}$ *in and only if* $\mathbf{z}_{k+1} = \rho_k = \rho(\mathbf{v}_k, \mathbf{w}_k)$ *approaches* λ *and the asymptotic convergence rate is quadratic.*

Proof. From the expression (11) it follows that as $\rho_k \to \lambda$

$$\begin{aligned}\|\mathbf{d}_k\| &= \|(\rho_{k-1}-\lambda)(F_p-\rho_{k-1}I)^{-1}(\mathbf{t}_k+\tau\mathbf{x})\| \\ &= |\rho_{k-1}-\lambda|\|((F_p-\lambda I)|_{\mathbf{y}^\perp})^{-1}\|\|\mathbf{t}_k+\tau\mathbf{x})\|+O((\rho_{k-1}-\lambda)^2)\end{aligned} \tag{13}$$

and similarly

$$\begin{aligned}\|\mathbf{s}_k\| &= \|(\rho_{k-1}-\lambda)^*(F_p-\rho_{k-1}I)^{-*}(\mathbf{p}_k+\theta\mathbf{y})\| \\ &= |\rho_{k-1}-\lambda|\,\|((F_p-\lambda I)|_{\mathbf{x}^\perp})^{-*}\|\,\|\mathbf{p}_k+\theta\mathbf{y}\|+O((\rho_{k-1}-\lambda)^2),\end{aligned} \tag{14}$$

where $\mathbf{d}_k, \mathbf{s}_k, \tilde{\mathbf{d}}_k$ and $\tilde{\mathbf{s}}_k$ are bounded. Hence, $\rho_k \to \lambda$ iff $\mathbf{v}_k \to \mathbf{x}$ and $\mathbf{w}_k \to \mathbf{y}$.

To prove the asymptotically quadratic rate of convergence, we use the expression (8) and the statement of Lemma 3

$$\rho_k-\lambda = \mathbf{s}_k^*(F_p-\lambda I)\mathbf{d}_k = (\rho_{k-1}-\lambda)^2\tilde{\mathbf{s}}_k^*(F_p-\lambda I)\tilde{\mathbf{d}}_k$$

and hence

$$|\rho_k-\lambda| = |\rho_{k-1}-\lambda|^2|\tilde{\mathbf{s}}_k^*(F_p-\lambda I)\tilde{\mathbf{d}}_k|+O((\rho_{k-1}-\lambda)^4).$$

Then from (13) and the last expression it follows that, as $k\to\infty$

$$\|\mathbf{x}-\mathbf{v}_k\| \le |\rho_{k-1}-\lambda|^2\|\tilde{\mathbf{d}}_k\|\tilde{\mathbf{s}}_k\|[k((F_p-\lambda I)|_{\mathbf{y}^\perp})\|\mathbf{t}_k+\tau\mathbf{x}\|]+O((\rho_{k-1}-\lambda)^3)$$

for k sufficiently large, and similarly, we can deduce

$$\|\mathbf{y}-\mathbf{w}_k\| \le |\rho_{k-1}-\lambda|^2\|\tilde{\mathbf{d}}_k\|\tilde{\mathbf{s}}_k\|[k((F_p-\lambda I)|_{\mathbf{x}^\perp})\|\mathbf{p}_k+\theta\mathbf{y}\|]+O((\rho_{k-1}-\lambda)^3)$$

which proves the asymptotically quadratic convergence (recall that in the last two expression $k(A|_{z^\perp})$ is the condition number of matrix A restricted to $z^\perp$). Theorem is proved.

4 Numerical Examples

In order to demonstrate the performance of the introduced iterative method (Algorithm 1, in Section 2.2) we performed a series of numerical experiments. We have tested it for the approximation of real and complex eigenvalues and corresponding eigenvectors of many companion matrix examples. In this section we present only the results for three examples concerning the root-finding problem. We illustrate Algorithm 1 with polynomials having only simple zeros, which were also considered from other authors.

The calculations were done using MATLAB. We have taken $\epsilon = 10^{-15}$ as the accuracy and the condition

$$\|z_i^{(k)} - \lambda\| \leq \epsilon$$

as termination criterion.

Example 1 *Consider the polynomial*

$$p(z) = z^3 - 8z^2 - 23z + 30\,,$$

with root vector $\alpha = (-3, 1, 10)$ *and the initial guess* $z^{(0)} = (-4, 2, 9)$ *which is taken from Dochev[14](see also [15]).*

The stopping criteria is reached after six iterations, see Table 1.

Table 1: Numerical results for Example 1.

$iter(k)$	$z_1^{(k)}$	$z_2^{(k)}$	$z_3^{(k)}$
0	-4	2	9
6	-3.000000000000000	1.000000000000000	10.0000000000000

Example 2 *We take the polynomial*

$$p(z) = z^5 - 15.5z^4 + 77.5z^3 - 155z^2 + 124z - 32$$

with root vector $\alpha = (0.5, 1, 2, 4, 8)$*, which was studied in Niell ([16], Ex.7.3). We use the same initial approximation* $z^{(0)} = (0.45, 0.9, 1.8, 3.6, 7.2)$.

Table 2: Numerical results for Example 2.

$iter(k)$	$z_1^{(k)}$	$z_2^{(k)}$	$z_3^{(k)}$	$z_4^{(k)}$	$z_5^{(k)}$
0	0.45	0.9	1.8	3.6	7.2
6	0.5000000000000002	0.999999999999998	1.999999999999994	3.999999999999996	8.000000000000002

The stopping criteria is reached after six iterations, see Table 2.

Example 3 *Consider the polynomial*

$$f(z) = z^9 + 3z^8 - 3z^7 - 9z^6 + 3z^5 + 9z^4 + 99z^3 + 297z^2 - 100z - 300$$

with the zero vector $\alpha = (2i, 2+i, -3, -2i, -1, 1, -2+i, 2-i, -2-i)$.

We use Abert's initial approximation vector $z^{(0)}$ (see [17]) given by

$$z_s^{(0)} = -\frac{a_1}{n} + r_0 \exp i\theta_s\,,\ \ \theta_s = \frac{\pi}{n}\left(2s - \frac{3}{2}\right)\,,\ \ s = 1, \ldots, n\,,$$

where $n = 9$ and $r_0 = 10$ (see also [18]).

The stopping criteria is reached after eleven iterations, see Table 3.

Table 3: Numerical results for Example 3.

iter(k)	$z_1^{(k)}$	$z_2^{(k)}$	$z_3^{(k)}$
0	-1.263 +1.736i	-4.683 +7.660i	-11.11+10i
11	$2.963 \times 10^{-18} + 2i$	2+i	$-3 + 1.009 \times 10^{-18} i$

$z_4^{(k)}$	$z_5^{(k)}$	$z_6^{(k)}$
-17.53 +7.660i	-20.95+1.736i	-19.77 -5i
$5.340 \times 10^{-18} - 2i$	$-1 + 2.222 \times 10^{-19} i$	$1 + 2.568 \times 10^{-18} i$

$z_7^{(k)}$	$z_8^{(k)}$	$z_9^{(k)}$
-14.53-9.396i	-7.690 -9.396i	-2.450-4.999i
$-2+i$	$2-i$	$-2-i$

We compare Algorithm 1 and the well-known Weierstrass' iterative method (see [15]), which also has second order convergence. All the results show that if we use the same initial vector and stopping criteria, the introduced method get the root-vector after same number of iterations as the Weierstrass' method.

Acknowledgement. This paper is supported by Shumen University under Grant RD 08-93/01.02.2019.

References

[1] A. Björck, G. Dahlquist, *Numerical Methods in Scientific Computing Volume II*, Society for Industrial and Applied Mathematics, Philadelphia, PA 2008.

[2] Y. Saad, *Numerical methods for large eigenvalue problems*, Classics in Applied Mathematics, vol. 66, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2011.

[3] J.L. Aurentz, T. Mach, L. Robol, R. Vandebril and D.S. Watkins, Fast and Backward Stable Computation of Roots of Polynomials, Part II: Backward Error Analysis; Companion Matrix and Companion Pencil, *SIAM J. Matrix Anal. Appl.*, 39(3):1245-1269, 2018.

[4] V.Y. Pan and Ai-Long Zheng, "TR-2013013: New Structured Matrix Methods for Real and Complex Polynomial Root-Finding", *Computer Science Technical Reports*, Paper 388, 2013.

[5] D.A. Bini, P. Boito, Y. Eidelman, L. Gemignani and I. Gohberg, A fast implicit QR eigenvalue algorithm for companion matrices, *Linear Algebra and its Applications*, 432:2006-2031, 2010.

[6] D. S. Watkins, *The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2007.

[7] A.J. Laub, *Matrix Analysis for Scientists & Engineers*, Society for Industrial and Applied Mathematics, Philadelphia, 2005.

[8] C. Kenney, A.J. Laub, Controllability and stability radii for companion form systems, *Math. Control Signals Systems*, 1:239-256, 1988.

[9] A. Ostrowski, On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors, I-VI, *Arch. Ration. Mech. Anal.* I 1:233-241, 1958; II 2:423-428, 1958; III 3:325-340, 1959; IV 3:341-347, 1959; V 3:472-481, 1959; VI 4:153-165, 1959.

[10] B.N. Parlett, The Rayleigh quotient iteration and some generalizations for nonnormal matrices, *Math. Comput.* 28: 679-693, 1974.

[11] J. Rommes, G.L.G. Sleijpen, Convergence of the Dominant Pole Algorithm and Rayleigh Quotient Iteration, *SIAM J. Matrix Anal. Appl.*, 30(1):346-363, 2008.

[12] H. Schwetlick, K. Schreiber, Nonlinear Rayleigh functionals, *Linear Algebra and its Applications*, 436(10): 3991-4016, 2012.

[13] A. AmiraslaniEmail, P. Lancaster, Rayleigh quotient algorithms for nonsymmetric matrix pencils, *Numer. Algorithms*, 51(1): 5-22, 2009.

[14] K. Dochev, Modified Newton methodfor the simultaneous computation of all roots of a given algebraic equation, *Phys. Math. J. Bulg. Acad. Sci.* 5:136-139, 1962.

[15] P.D. Proinov, M.D. Petkova, A new semilocal convergence theorem for the Weierstrass method for finding zeros of a polynomial simultaneously, *Journal of complexity*, 30(3):366-380, 2014.

[16] A.M. Niel, The simultaneous approximation of polynomial roots, *Computers and Mathematics with applications*, 41:1-14, 2001.

[17] O. Abert, Iteration methods for finding all zeros of a polynomial simultaneously, *Math. Comput.* 27:339-334, 1973.

[18] M. Petković, S. Ilić, S. Tričković, A family of simultaneous zero-finding methods, *Comput. Math. Appl.* 34:49-59, 1997.