Università di Pisa
Dipartimento di Matematica

Polynomial Eigenproblems: a Root-Finding Approach

Dottorato di Ricerca in Matematica

Tesi per il conseguimento del titolo

Candidato:
Dott. Vanni Noferini

Relatori:
Prof. Dario Andrea Bini
Prof. Luca Gemignani

27 Settembre 2012

Contents

Introduction

A matrix polynomial [44], also known as a polynomial matrix [38] or sometimes as a λ-matrix [70], is a polynomial whose coefficients are matrices; or, equivalently, a matrix whose elements are polynomials.

Consider the $n \times n$ square matrix polynomial of degree k

$$P(x) = \sum_{j=0}^{k} P_j x^j$$

and let $p(x) := \det P(x)$ be a scalar polynomial of degree $N \leq nk$.

If $P(x)$ is regular, that is if $p(x)$ is not identically zero, the polynomial eigenvalue problem (PEP) associated with $P(x)$ is equivalent to the computation of the roots of the polynomial $p(x)$; such roots are called the eigenvalues of the regular matrix polynomial $P(x)$. Sometimes, one is also interested in computing the corresponding (left and right) eigenvectors.

Recently, much literature has been addressed to the polynomial eigenvalue problem. This line of research is currently very active: the theoretical properties of PEPs are studied, and fast and numerically stable methods are sought for their numerical solution. The most commonly encountered case is the one of degree 2 polynomials (see, e.g., [39, 40, 49, 52] and the survey [105]), but there exist applications where higher degree polynomials appear (see, e.g., [63, 85, 87]). More generally, PEPs are special cases belonging to the wider class of nonlinear eigenvalue problems; a survey of nonlinear eigenvalue problem can be found, e.g., in [46, 85], while in [7] a collection of selected nonlinear eigenproblems is presented in order to serve as a benchmark to test new algorithms. Amongst nonlinear eigenvalue problems, rational eigenvalue problems [85] can be immediately conducted to polynomial form, multiplying them by their least common denominator; truly nonlinear eigenvalue problems may be approximated with PEPs, truncating some matrix power series, or with rational eigenproblems, using rational approximants such as Padé approximants.

To approximate numerically the solutions of PEPs, several algorithms have been introduced based on the technique of linearization where the polynomial problem is replaced by a linear pencil with larger size and the customary methods for the generalised eigenvalue problem, like for instance the QZ algorithm [88], are applied. For more details, see for instance [60, 77, 78, 105] and the references therein. Some algorithms that have appeared in the literature are able to avoid the linearization step; we may cite for instance [64, 65, 69, 98]. Also, doubling algorithms such as cyclic reduction [11, 17] or SDA [18, 50] can be adapted to solve certain kinds of PEPs.

This thesis is addressed to the design and analysis of algorithms for the polynomial eigenvalue problem based on a root-finding approach. A root-finder will be applied to the characteristic equation $p(x) = 0$. In particular, we will discuss algorithms that use the Ehrlich-Aberth iteration [1, 14, 27].

The Ehrlich-Aberth iteration (EAI) is a method that simultaneously approximates all the roots of a (scalar) polynomial. It appeared for the first time in [14]. Subsequently, it has been discovered again independently in [1] and [27]. Its theoretical properties in terms of local convergence order are analysed in the books [81, 96]. An efficient implementation of the EAI for scalar polynomials, combined with various techniques based on theoretical results such as the Gerschgorin and Rouché theorems, is described in [8, 9]. Applications of the EAI to tridiagonal eigenvalue problems were presented in [10, 97].

In order to adapt the EAI to the numerical solution of a PEP, we propose a method based on the Jacobi formula [45]; two implementations of the EAI are discussed, of which one uses a linearization and the other works directly on the matrix polynomial. The algorithm that we propose has quadratic computational complexity with respect to the degree k of the matrix polynomial. This leads to computational advantage when the ratio k^2/n, where n is the dimension of the matrix coefficients, is large. Cases of this kind can be encountered, for instance, in the truncation of matrix power series [111]. If k^2/n is small, the EAI can be implemented in such a way that its asymptotic complexity is cubic (or slightly supercubic) in nk, but QZ-based methods appear to be faster in this case. Nevertheless, experiments suggest that the EAI can improve the approximations of the QZ in terms of forward error, so that even when it is not as fast as other algorithms it is still suitable as a refinement method.

The EAI does not compute the eigenvectors. If they are needed, the EAI can be combined with other methods such as the SVD or the inverse iteration. In the experiments we performed, eigenvectors were computed in this way, and they were approximated with higher accuracy with respect to the QZ.

Another root-finding approach to PEPs, similar to the EAI, is to apply in sequence the Newton method to each single eigenvalue, using an implicit deflation of the previously computed roots of the determinant [37, 70] in order to avoid to approximate twice the same eigenvalue. Our numerical experience suggests that in terms of efficiency the EAI is superior with respect to the sequential Newton method with deflation.

Specific attention concerns structured problems where the matrix coefficients P_j have some additional feature which is reflected on structural properties of the roots. For instance, in the case of T-palindromic polynomials [71, 100], where $P_j = P_{k-j}^T \in \mathbb{C}^{n\times n}$, the roots are encountered in pairs $\{x, 1/x\}$. In this case the goal is to design algorithms which take advantage of this additional information about the eigenvalues and deliver approximations to the eigenvalues which respect these symmetries independently of the rounding errors. Within this setting, we study polynomials endowed with specific properties like, for instance, palindromic, T-palindromic, Hamiltonian, symplectic, even/odd, etc., whose eigenvalues have special symmetries in the complex plane [30, 71, 75, 77, 83, 86, 87, 100, 107]. In general, we may consider the case of structures where the roots can be grouped in pairs as $\{x, f(x)\}$, where $f(x)$ is any analytic function such that $f(x) = f^{-1}(x)$ [12, 41].

We propose a unifying treatment of structured polynomials belonging to this class and show how the EAI can be adapted to deal with them in a very

effective way. Several structured variants of the EAI are available to this goal: they are described in this thesis and in [12, 42, 92]. All of such variants enable us either to compute only a subset of eigenvalues or to solve another PEP linked to the original one and with fewer eigenvalues. The sought spectrum is then recovered by means of the symmetries satisfied by the eigenvalues. By exploiting the structure of the problem, this approach leads to a saving on the number of floating point operations and provides algorithms which yield numerical approximations fulfilling the symmetry properties. Our research on the structured EAI can of course be applied also to scalar polynomials: in the next future, we plan to exploit our results and design new features for the software `MPSolve` [9].

When studying the theoretical properties of the change of variable, useful to design one of the structured EAI methods, we had the chance to discover some theorems on the behaviour of the complete eigenstructure of a matrix polynomial under a rational change of variable [93]. Such results are discussed in this thesis.

Some, but not all, of the different structured versions of the EAI algorithm have a drawback: accuracy is lost for eigenvalues that are close to a finite number of critical values, called exceptional eigenvalues. On the other hand, it turns out that at least for some specific structures the versions that suffer from this problem are also the most efficient ones: thus, it is desirable to circumvent the loss of accuracy. This can be done by the design of a structured refinement Newton algorithm [41]. Besides its application to structured PEPs, this algorithm can have further application to the computation of the roots of any scalar polynomial whose roots appear in pairs [48, 73].

In this thesis, we also present the results of several numerical experiments performed in order to test the effectiveness of our approach in terms of speed and of accuracy. We have compared the Ehrlich-Aberth iteration with the Matlab[1] functions `polyeig` and `quadeig` [52]. In the structured case, we have also considered, when available, other structured methods, say, the URV algorithm by Schröder [100]. Moreover, the different versions of our algorithm are compared one with another.

All the numerical experiments discussed in this thesis have been performed on the same machine, equipped with a CPU Intel Xeon 2.80GHz and a Linux Debian 6.02 OS.

The structure of the thesis is the following: Chapter describes the notation that is used throughout the following chapters. In Chapter 1, we recall basic theoretical properties of polynomials, with specific emphasis on matrix polynomials. In Chapter 2, the EAI for a generic PEP is described and analysed. Chapter 3 is devoted to the development of some theory on the change of variable for a matrix polynomials: amongst the theoretical results presented there, there are some that will be used for the construction of one of the three kinds of structured EAI (SEAI) algorithms that we propose. Such SEAI are the subject of Chapter 4, while in Chapter 5 a structured Newton method is introduced for the refinement of those close-to-exceptional eigenvalues that may have suffered from a loss of accuracy. In Chapter 6 conclusions are drawn and a look is given towards future research.

Original research contributions from the author of the thesis, as an author

[1] Matlab is a registered trademark of The MathWorks, Inc.

or a coauthor, are found in the papers [12, 13, 41, 42, 92, 93].

List of symbols and notations

We list in the following some notation we will use throughout the thesis.

- Z: a generic ring (with unity).
- 0_Z: the identity element of the ring Z with respect to addition.
- 1_Z: the identity element of the ring Z with respect to multiplication.
- $\mathbb{F}$, $\mathbb{K}$: generic fields. $\mathbb{K}$ is specifically used for algebraically closed fields, while the closure of a field denoted by the symbol $\mathbb{F}$ depends on the context. If $\alpha, \beta \in \mathbb{F}$ and β^{-1} is the multiplicative inverse of $\beta \neq 0_{\mathbb{F}}$ we will sometimes use the notation $\frac{\alpha}{\beta} := \alpha\beta^{-1} = \beta^{-1}\alpha$.
- $\overline{\mathbb{F}}$: the algebraic closure of the field $\mathbb{F}$.
- $\mathbb{C}$: the field of complex numbers.
- x^*: the complex conjugate of the complex number $x \in \mathbb{C}$.
- $\mathbb{R}$: the field of real numbers.
- $\mathbb{Q}$: the field of rational numbers.
- $\mathbb{N}$: the set of natural numbers and $\mathbb{N}_0 := \mathbb{N} - \{0\}$.
- We formally define $\infty := 0_{\mathbb{F}}^{-1}$ and we write $\mathbb{F}^* := \mathbb{F} \cup \{\infty\}$.
- $Z[[x]]$: the ring of formal power series in the variable x with coefficients in the ring Z.
- $Z[x]$: the ring of univariate polynomials in the variable x with coefficients in the ring Z.
- $\mathbb{F}[x]$: the ring of univariate polynomials in the variable x with coefficients in the field $\mathbb{F}$. Given $r_1, r_2 \in \mathbb{F}[x]$ we say that r_1 divides r_2 and write $r_1 | r_2$ if there exists an $r_3 \in \mathbb{F}[x]$ such that $r_2 = r_1 \cdot r_3$.
- $\mathbb{F}(x)$: the field of rational fractions with coefficients in the field $\mathbb{F}$.

- Given $m, n \in \mathbb{N}_0$, $Z^{m\times n}$: the set of the $m \times n$ matrices with entries in the ring Z. In particular, three commonly encountered cases will be: $\mathbb{F}^{m\times n}$, the set of the $m \times n$ matrices with entries in the field $\mathbb{F}$; $(\mathbb{F}[x])^{m\times n}$, the set of the $m \times n$ matrices with entries in the polynomial ring $\mathbb{F}[x]$; and $(\mathbb{F}(x))^{m\times n}$, the set of the $m \times n$ matrices with entries in the field $\mathbb{F}(x)$.

- Given $P(x) \in (\mathbb{F}[x])^{m\times n}$ and $x_0 \in \mathbb{F}$, $P(x_0)$: the matrix belonging to $\mathbb{F}^{m\times n}$ obtained evaluating each entry in $P(x)$ at $x = x_0$.

- $\mathbb{F}^{m\times n}[x]$: the set of matrix polynomials with coefficients in $\mathbb{F}^{m\times n}$. Such objects can equivalently be seen as matrices whose entries are polynomials with coefficients in $\mathbb{F}$, that is $\mathbb{F}^{m\times n}[x] = (\mathbb{F}[x])^{m\times n}$.

- If $A \in Z^{m\times n}$, $A_{ij} \in Z$: the entry in the ith row and jth column of A. Sometimes the alternative notation $A(i, j)$ is used.

- If $A \in Z^{m\times n}$, $A(i_1 : i_2, j_1 : j_2) \in Z^{(1+i_2-i_1)\times(1+j_2-j_1)}$: the submatrix of A with rows from i_1 to i_2 and columns from j_1 to j_2.

- Z^m: the set of vectors with m elements in Z, denoted by Z^m.

- Given some vectors $v_1, \ldots, v_s \in Z^m$, $\operatorname{span}(\{v_1, \ldots, v_s\}) \subseteq Z^m$: the vector space of all the vectors $w = Vc$, where $V = [v_1, \ldots, v_s] \in Z^{m\times s}$ and $c \in Z^s$.

- If $A \in \mathbb{F}^{m\times n}$, $\ker A \subseteq \mathbb{F}^n$: the vector space of all vectors w such that $Aw = 0_{\mathbb{F}^m}$.

- A^T: the transpose of the matrix A.

- A^H: the conjugate transpose of the matrix A.

- $M_n(Z) := Z^{n\times n}$. If Z is commutative, the determinant of $A \in M_n(Z)$ is defined as usually and denoted by $\det A \in Z$.

- δ_{ij}: the Kronecker delta, defined as $\delta_{ij} = 1_{\mathbb{F}}$ if $i = j$ and $\delta_{ij} = 0_{\mathbb{F}}$ if $i \neq j$.

- Given $n := \min(m, p)$ and $D_1, \ldots, D_n \in Z$, the matrix $\operatorname{diag}(D_1, \ldots, D_n) := D_i\delta_{ij} \in Z^{m\times p}$. Notice that we use this notation and speak of a diagonal matrix also when $m \neq p$. However, when we do not specify the dimensions, it is agreed that $\operatorname{diag}(D_1, \ldots, D_n) := D_i\delta_{ij} \in Z^{n\times n}$.

- I_n: the identity matrix of dimension n, defined as $\operatorname{diag}(1_{\mathbb{F}}, \ldots, 1_{\mathbb{F}})$.

- 0_n: the zero matrix of dimension n, whose n^2 entries are all equal to $0_{\mathbb{F}}$.

- Given $A \in Z^{n_A\times m_A}$, $B \in Z^{n_B\times m_B}$, $A \oplus B \in Z^{(n_A+n_B)\times(m_A+m_B)}$ is the direct sum of A and B, defined as the block matrix $\left[\begin{smallmatrix} A & 0 \\ 0 & B \end{smallmatrix}\right]$.

- Given $A \in Z^{n_A\times m_A}$, $B \in Z^{n_B\times m_B}$, $A \otimes B \in Z^{(n_An_B)\times(m_Am_B)}$: the tensor product (or Kronecker product) of A and B, defined as the matrix such that $(A \otimes B)_{(i-1)n_B+k,(j-1)m_B+\ell} = A_{i,j}B_{k,\ell}$.

- Some Matlab-like notations used inside pseudocode to describe algorithms: *zeros(m,n)* calls the $m \times n$ zero matrix; *eye(n)* calls the $n \times n$ identity matrix; *ones(m,n)* calls an $m \times n$ matrix with all elements equal to 1; given a two-element vector y, *planerot(y)* returns the 2×2 unitary matrix M such that $My = [\star, 0]^T$ where $\star$ is any complex number.

Chapter 1

Basic facts on polynomials

This chapter is devoted to recalling some definitions that will be important in the subsequent chapters. We also expose some basic results on polynomials in general and on matrix polynomials in particular. Some of the facts that we recall in this chapter are provided without citation, because they are classical results that are found in algebra textbooks. We mention, for instance, [74, 108].

1.1 Scalar polynomials: degree, grade, greatest common divisor

Let Z be a ring, and consider a sequence $(a_i)_i \subset Z$. An element $z = \sum_{i=0}^{\infty} a_i x^i \in Z[[x]]$ is called a *polynomial* if either $a_i = 0_Z \ \ \forall i$, or there exists an index k such that $a_k \neq 0_Z$ and $a_i = 0_Z \ \forall \ i > k$. In the former case we say that z is the zero polynomial and write $z = 0_{Z[x]}$. If on the contrary $z \neq 0_{Z[x]}$ we say that k is the *degree* of z, and sometimes write $k = \deg z$. Moreover, we formally define $\deg 0_{Z[x]} = -\infty$ and by convention we agree that $-\infty + (-\infty) = -\infty$, $-\infty + m = -\infty$ and $-\infty < m \ \forall m \in \mathbb{N}$; the following properties hold.

Proposition 1.1. *If $z_1, z_2 \in Z[x]$, then:*

1. *(product)* $\deg(z_1 z_2) \leq \deg z_1 + \deg z_2$, *and* $\deg z_1 z_2 = \deg z_1 + \deg z_2$ *if Z is a domain (i.e. has neither left nor right divisors of the zero element);*

2. *(sum)* $\deg(z_1 + z_2) \leq \max(\deg z_1, \deg z_2)$, *and if one also assumes that* $\deg z_1 \neq \deg z_2$ *then* $\deg(z_1 + z_2) = \max(\deg z_1, \deg z_2)$.

Proof. Suppose first that both z_1 and z_2 are nonzero. Let $k_1 = \deg z_1$, $k_2 = \deg z_2$, $K = k_1 + k_2$ and $M = \max(k_1, k_2)$. Write $z_1 = \sum_{i=0}^{\infty} a_i x^i$ and $z_2 = \sum_{i=0}^{\infty} b_i x^i$. One has $z_1 + z_2 = \sum_{i=0}^{\infty} s_i x^i$ and $z_1 z_2 = \sum_{i=0}^{\infty} p_i x^i$, with $s_i = a_i + b_i$ and $p_i = \sum_{j=0}^{i} a_j b_{i-j}$. Therefore $i > K$ implies that $p_i = 0_Z$, while if there are no zero divisors then $p_K = a_{k_1} b_{k_2} \neq 0_Z$. Finally, $i > M$ implies that $s_i = 0_Z$. If $\deg z_1 = \deg z_2$, then it may be possible that $s_M = 0_Z$ (if $a_M + b_M = 0_Z$); otherwise, this is impossible, since one and only one between a_M and b_M is nonzero.

If the two considered polynomials are not both nonzero, suppose without any loss of generality that $z_2 = 0_{Z[x]}$. Then $z_1 \cdot 0_{Z[x]} = 0_{Z[x]} \cdot z_1 = 0_{Z[x]}$

and $z_1 + 0_{Z[x]} = 0_{Z[x]} + z_1 = z_1$. Recalling the formalism we adopted for the operations with $-\infty$, this completes the proof. □

On the other hand, if $z \in Z[x]$ and we pick any $g \geq k = \deg z$, we can write $z = \sum_{i=0}^{g} a_i x^i$. We say that g is the *grade* [76] of z, and sometimes write $g = \text{grade}(z)$. We underline that the choice of any $g \geq k$ is arbitrary, and that $g = k$ if and only if $a_g \neq 0_Z$. When not otherwise specified, we also agree by convention that if $z_1, z_2 \in Z[x]$ and if $g_1 = \text{grade}(z_1)$, $g_2 = \text{grade}(z_2)$, then $\text{grade}(z_1 \cdot z_2) = g_1 + g_2$ and $\text{grade}(z_1 + z_2) = \max(g_1, g_2)$.

Remark 1.1. *In some sense, the degree of a polynomial is an intrinsic property while the grade depends on its representation. Informally speaking, the grade depends on how many zero coefficients one wishes to add in front of the polynomial.*

Let now g be the grade of $z = \sum_{i=0}^{g} a_i x^i \in Z[x]$. The *reversal* of z with respect to its grade [43, 76] is

$$\text{Rev}_g z := \sum_{i=0}^{g} a_{g-i} x^i. \tag{1.1}$$

In the following, the subscript g will sometimes be omitted when the reversal is taken with respect to the degree of the polynomial, that is $\text{Rev}_k z =: \text{Rev} z$. Notice moreover that, if $Z = \mathbb{F}$ is a field and if we think of z as a function $z(x) : \mathbb{F} \to \mathbb{F}$, then the reversal with respect to g can be written as

$$\text{Rev}_g z(x) = x^g z(x^{-1}).$$

If we consider univariate polynomials with coefficients not just in a generic ring, but in a field $\mathbb{F}$, then $\mathbb{F}[x]$ is a principal ideal domain (the reverse implication is also true, that is $Z[x]$ is a principal ideal domain if and only if Z is a field). Furthermore, $\mathbb{F}[x]$ is also a Euclidean ring. This well-known property will be crucial in Chapter 3. In particular, the concept of greatest common divisor can be defined in any principal ideal domain. Take any $z \in \mathbb{F}[x]$, with $\deg z = k$. We can still write $z = \sum_{i=0}^{g} a_i x^i$ for any $g \geq k$. We say that z is *monic* if $a_k = 1_{\mathbb{F}}$. Given $z_1, z_2 \in \mathbb{F}[x]$, not both zero, we define their greatest common divisor z_3 as the monic polynomial with largest possible degree such that $z_3 | z_1$ and $z_3 | z_2$. We write $z_3 = \text{GCD}(z_1, z_2)$. Notice that the requirement that $\text{GCD}(z_1, z_2)$ is always monic guarantees the uniqueness of the GCD. If $\text{GCD}(z_1, z_2) = 1_{\mathbb{F}[x]}$ we say that z_1 and z_2 are *coprime*.

1.2 Matrix polynomials (a.k.a. polynomial matrices)

A matrix polynomial can be informally defined as a polynomial whose coefficients are matrices.

More formally, let us denote by $Z^{m \times n}$ the set of $m \times n$ matrices with entries in the ring Z. We denote by $(Z^{m \times n})[x]$ the set of univariate matrix polynomials in the variable x with coefficients in $Z^{m \times n}$. The particular case $n = 1$ is referred to as polynomial vector, and is denoted simply by $Z^m[x]$.

Notice that $Z^{m\times n}$ is a group together with the operation of addition (and, in particular, it has a zero element) but, in general, it is not a ring (unless $m = n$). Nevertheless, many of the concepts introduced in Section 1.1 can be easily adapted to the matrix case, also for $m \neq n$.

In particular, the notions of grade and degree of a scalar polynomial can be extended in a straightforward way to matrix polynomials, by noticing that they can be seen as matrices whose entries are polynomials: the grade (resp., the degree) of $A \in (Z^{m\times p})[x]$ is defined as $\max_{i,j} \operatorname{grade}(A_{ij})$ (resp., as $\max_{i,j} \deg A_{ij}$). Analogously, the reversal of a matrix polynomial is defined just as in (1.1), after replacing $a_i \in Z$ with $B_i \in Z^{m\times n}$.

In this thesis we will be specially concerned with the situation where the underlying ring Z is actually some field $\mathbb{F}$, and in particular to the most commonly met case of $\mathbb{F} = \mathbb{C}$. If $A \in \mathbb{F}^{m\times n}[x]$ is a matrix polynomial, it can obviously also be seen as function $A(x) : \mathbb{F} \to \mathbb{F}^{m\times n}$. Some caution must be used in this sense when $\mathbb{F}$ is a finite field, because it may happen that two different polynomials coincide as functions over $\mathbb{F}$, e.g. 0 and $x^2 + x$ if $\mathbb{F} = F_2$.

Remark 1.2. *A simple but important observation is that a matrix polynomial can also be seen as a polynomial matrix, defined as a matrix whose entries are polynomials. In other words,* $(\mathbb{F}^{m\times n})[x] = (\mathbb{F}[x])^{m\times n}$. *The matrix polynomial point of view is more useful when working on the numerical solution of polynomial eigenvalue problems, while the polynomial matrix point of view turns out to be the better tool when dealing with algebraic properties. In this thesis, we will freely switch between the two.*

A matrix polynomial (or, equivalently, a polynomial matrix) is said to be *square* if $m = n$ or *rectangular* otherwise. The set of square polynomial matrices of dimension n with elements in a ring Z is denoted my $M_n(Z)$. In particular, the ring of square polynomial matrices of dimension n with elements in $\mathbb{F}[x]$ is denoted by $M_n(\mathbb{F}[x])$. It is a ring when equipped with the operations of addition and matrix multiplication.

Since a matrix polynomial is also a polynomial matrix, it can be seen as a matrix with elements in a particular ring. If we introduce the field of fractions $\mathbb{F}(x)$, we can go even further and apply to matrix polynomials all the classical results in matrix theory for matrices with elements in a field. In particular the following are valid for matrices whose entries are rational functions and, *a fortiori*, for matrix polynomials:

- the function rank, $\operatorname{rank} : (\mathbb{F}(x))^{m\times n} \to \mathbb{N}$ and the function determinant, $\det : M_n(\mathbb{F}(x)) \to \mathbb{F}(x)$, are defined in the usual sense;
- the operation of transposition is also defined as usually, and it is denoted by the superscript T, so $A(x) \in (\mathbb{F}(x))^{m\times n} \Leftrightarrow [A(x)]^T \in (\mathbb{F}(x))^{n\times m}$;
- if $A(x) \in M_n(\mathbb{F}(x))$ has full rank, then there exists a unique inverse, denoted by $(A(x))^{-1}$, such that $A(x)(A(x))^{-1} = (A(x))^{-1}A(x) = I_n$, having defined the $n \times n$ identity matrix $I_n := \operatorname{diag}(1_{\mathbb{F}(x)}, \ldots, 1_{\mathbb{F}(x)})$.

The determinant can be more in general defined for any square matrix with elements in a commutative ring. In particular, it is clear that $\det P(x) \in \mathbb{F}[x]$ whenever $P(x) \in M_n(\mathbb{F}[x])$.

A matrix polynomial $A(x) \in M_n(\mathbb{F}[x])$ is said to be *regular* if $\det A(x) \neq 0_{\mathbb{F}[x]}$ and *singular* otherwise. $A(x)$ is regular if and only if its rank is n. Some authors use the term "normal rank" when referring to the rank of a polynomial matrix considered as an element of $(\mathbb{F}(x))^{m\times n}$, reserving the word "rank" to constant matrices, i.e. elements of $\mathbb{F}^{m\times n}$. We underline that in this thesis we do not make this formal distinction, since it is quite natural to define the rank for matrices with entries in $\mathbb{F}(x)$ and to consider, when necessary, constant matrices as matrices with entries belonging to the subfield $\mathbb{F}$.

A *unimodular* matrix polynomial is a square matrix polynomial which is regular and has constant determinant, i.e. $A(x) \in M_n(\mathbb{F}[x])$ s.t. $0_{\mathbb{F}} \neq \det A(x) \in \mathbb{F}$. Unimodular matrix polynomials allow to introduce the concept of equivalence [44]. Let $P_1(x),\ P_2(x) \in (\mathbb{F}[x])^{m\times n}$. If there exist two unimodular matrix polynomials $A(x) \in M_m(\mathbb{F}[x])$ and $B(x) \in M_n(\mathbb{F}[x])$ such that $A(x)P_1(x)B(x) = P_2(x)$, then the two matrix polynomials $P_1(x)$ and $P_2(x)$ are said to be *equivalent*. Furthermore, if there exist two constant nonsingular matrices $A \in M_m(\mathbb{F})$ and $B \in M_n(\mathbb{F})$ such that $AP_1(x)B = P_2(x)$, then we say that $P_1(x)$ and $P_2(x)$ are *strictly equivalent* [72]. Notice that the adjugate of a square matrix polynomial is again a matrix polynomial. However, in general, the inverse of a regular square matrix polynomial, considered as an element of $M_n(\mathbb{F}(x))$, is not a polynomial. Unimodular matrix polynomials are an exception in this sense: they have a polynomial inverse, since their determinant is a nonzero element of $\mathbb{F}$. We can thus conclude that both equivalence and strict equivalence are indeed equivalence relations.

Finally, a polynomial matrix (or, more generally, a matrix with elements in the field of fractions $\mathbb{F}(x)$), can be thought of as a linear mapping between two vector spaces. Namely, if $A(x) \in (\mathbb{F}(x))^{m\times n}$, $A(x)$ is a linear function from $\mathbb{F}(x)^n$ to $\mathbb{F}(x)^m$. Moreover, $\ker A(x)$ is defined as the set of all vectors $v(x) \in (\mathbb{F}(x))^n$ such that $A(x)v(x) = 0$.

1.2.1 Complete eigenstructure of a polynomial matrix

In the present subsection we will assume for the sake of simplicity that the underlying field $\mathbb{F}$ is algebraically closed.

Let $A \in (F[x])^{m\times p}$, and let $\nu =: \min(m, p)$. Suppose that there exist $D_1, \ldots, D_\nu \in \mathbb{F}[x]$ such that $A_{ij} = D_i\delta_{ij}$, where δ_{ij} is the Kronecker's delta. Then we write $A = \mathrm{diag}(D_1,\ \ldots,\ D_\nu)$, and we say that A is *diagonal*. Notice that we use the notation indifferently for both square and rectangular polynomial matrices.

An important theoretical result regards the equivalence of any polynomial matrix to its Smith form [103]. Let us first recall what the Smith form is for a square polynomial matrix.

Theorem 1.1. *Let $P(x) \in M_n(\mathbb{F}[x])$. Then $P(x)$ is equivalent to*

$$S(x) = \mathrm{diag}(d_1(x), \ldots, d_n(x)),$$

where $\forall i \leq m\ d_i(x) \in \mathbb{F}[x]$ is monic and $d_i(x)|d_{i+1}(x)\ \ \forall i \leq n-1$.

Such an $S(x) \in M_n(\mathbb{F}[x])$ is called the *Smith form* [44] of $P(x)$, and the $d_i(x)$ are called its *invariant polynomials* [38, 44]. The Smith form, and thus the invariant polynomials, are uniquely determined by $P(x)$. Notice that a square

polynomial matrix $P(x)$ is singular if and only if at least one of its invariant polynomials is zero.

Let us consider a factorization of the invariant polynomials over $\mathbb{F}[x]$: $d_i(x) = \prod_j [\pi_{j,(i)}(x)]^{k_{j,(i)}}$, where $\pi_{j,(i)}(x)$ are monic prime factors. Using the fact that $\mathbb{F}$ is algebraically closed, the factor $(\pi_{j,(i)}(x))^{k_{j,(i)}}$ can be written as $(x - x_0)^{k_{j,(i)}}$ for some $x_0 \in \mathbb{F}$. Factors of this form are called the *elementary divisors* of $P(x)$ [38, 44] corresponding to the *characteristic value* x_0 [38]. The properties of the invariant polynomials imply that if $i < j$ and $(x - x_0)^a$ is an elementary divisor that is a factor of $d_i(x)$ than there exists an integer $b \geq a$ such that $(x - x_0)^b$ is a factor of $d_j(x)$.

Theorem 1.1, which in its most general form is due to Frobenius [36], is in point of fact valid for any matrix, not necessarily square, with entries in any principal ideal domain [38], [44]. Let us state the Theorem again, this time for the more general case.

Theorem 1.2. *Let $P(x) \in (\mathbb{F}[x])^{m \times p}$. Then there exist two unimodular $A(x) \in M_m(\mathbb{F}[x])$ and $B(x) \in M_p(\mathbb{F}[x])$ such that*

$$S(x) = A(x)P(x)B(x) = \mathrm{diag}(d_1(x), \ldots, d_\nu(x)),$$

where $\forall i \leq \nu := \min(m, p)$ $d_i(x) \in \mathbb{F}[x]$ is monic and $d_i(x) | d_{i+1}(x)$ $\forall i \leq \nu - 1$.

Once again $S(x)$ is called the Smith form of $P(x)$, and it is always uniquely determined for any polynomial matrix $P(x)$, either square or rectangular [44]. Therefore, the definitions of invariant polynomials, characteristic values and elementary divisors given above can be immediately extended also to rectangular polynomial matrices.

Let now $\mathcal{V}$ be a vector subspace of $(\mathbb{F}(x))^m$, with $\dim \mathcal{V} = s$. Let $\{v_i\}_{i=1,\ldots,s}$ be a polynomial basis for $\mathcal{V}$ with the property $\deg v_1 \leq \cdots \leq \deg v_s$. Often we will arrange a polynomial basis in the matrix form $V(x) = [v_1(x), \ldots, v_s(x)] \in (\mathbb{F}[x])^{m \times s}$. Clearly, polynomial bases always exist, because one may start from any basis with elements in the (vectorial) field of fractions, and then build a polynomial basis just by multiplying by the least common denominator. Let $\alpha_i := \deg v_i$ be the degrees of the vectors of such a polynomial basis; the *order* of $V(x)$ is defined [34] as $\sum_{i=1}^{s} \alpha_i$. A polynomial basis is called *minimal* [34] if its order is minimal amongst all the polynomial bases for $\mathcal{V}$, and the α_i are called its *minimal indices* [34]. It is possible to prove [34, 38] that, although there is not a unique minimal basis, the minimal indices are uniquely determined by $\mathcal{V}$.

The *right minimal indices* [24] of a polynomial matrix $P(x) \in (\mathbb{F}[x])^{m \times n}$ are defined as the minimal indices of $\ker P(x)$. Analogously, the *left minimal indices* [24] of $P(x)$ are the minimal indices of $\ker P(x)^T$.

Given the grade g of $P(x)$, we say that ∞ is a characteristic value of $P(x)$ if $0_\mathbb{F}$ is a characteristic value of $Rev_g P(x)$. The elementary divisors corresponding to ∞ are defined [53] as the elementary divisors of $Rev_g P(x)$ corresponding to $0_\mathbb{F}$; if x^ℓ is an elementary divisor of $Rev_g P(x)$ we formally write that $(x - \infty)^\ell$ is an infinite elementary divisor of $P(x)$. Notice that the infinite elementary divisors of a polynomial matrix clearly depend on the arbitrary choice of its grade. Details are given in Proposition 3.2.

We are now in the position to define the *complete eigenstructure* [24] of $P(x)$ as the set of both finite and infinite elementary divisors of $P(x)$ and of its left and right minimal indices.

We conclude this section with a couple of examples.

Example 1.1. *Let* $\mathbb{F} = \overline{\mathbb{Q}}$. *Consider the structured (its determinant is in fact a* (-1)*-palindromic polynomial with grade of palindromicity* 4*: see Chapters 4 and 5 for the definition of* κ*-palindromic polynomials) matrix polynomial of grade* 2

$$P(x) = \begin{bmatrix} x & x^2 \\ -1 & \frac{4ix^2+5x-4i}{5} \end{bmatrix};$$

one has that $\det P(x) = (2x - i)(x - 2i)(2ix/5)$*: therefore,* $P(x)$ *is regular. Moreover,* $A(x) = \left[\begin{smallmatrix} 0 & 1 \\ 1 & x \end{smallmatrix}\right]$ *and* $B(x) = \begin{bmatrix} -1 & \frac{4x^2-5ix-4}{4} \\ 0 & -\frac{5i}{4} \end{bmatrix}$ *are unimodular matrix polynomials such that* $A(x)P(x)B(x) = S(x)$*, where* $S(x) = \begin{bmatrix} 1 & 0 \\ 0 & x^3 - \frac{5i}{2}x^2 - x \end{bmatrix}$ *is the Smith form of* $P(x)$*. Furthermore,* $B(-x)^T \mathrm{Rev} P(x) A(-x) = -S(-x)$*, which implies that* $\left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right] S(-x)$ *is the Smith form of* $\mathrm{Rev} P(x)$*.*

The conclusion is that the complete eigenstructure of this matrix polynomial coincides with the elementary divisors $x, x - 2i, x - i/2, x - \infty$*; we recall that the latter formal expression means that there is an infinite elementary divisor of degree* 1*, or in other words* x *is an elementary divisor for* $\mathrm{Rev} P(x)$*.*

Example 1.2. *Suppose that* $\mathbb{F} = \mathbb{C}$ *and consider the matrix polynomial of grade* 2

$$P(x) = \begin{bmatrix} -2 & x+3 & 0 & 1 & -1 \\ 2x^2 - 2x - 4 & 6 - 2x^2 + 3x & x & 2 - x^2 & x^2 - 2 \\ 0 & x^2 - x + 1 & x^2 - x & x^2 - x + 1 & x^2 - x \\ 2 - 2x^2 + 2x & 2x^2 - 4x - 2 & -x & x^2 - 2x & 1 - x^2 \\ 2 & -1 - x^2 & x - x^2 & 1 - x^2 + x & 1 - x^2 + x \end{bmatrix}.$$

In order to investigate what the complete eigenstructure of $P(x)$ *is, let us first notice that* $\mathrm{rank}(P(x)) = 4$*. Therefore,* $P(x)$ *is singular, with* $\dim \ker P(x) = 1$*. A right minimal basis is* $\{[1, 0, 2, 0, -2]^T\}$*, while a left minimal basis is* $\{[2x - 4, 3, 2x - 1, 3, 2x - 1]^T\}$*. It can be checked that there exist a unimodular polynomial* $A(x)$ *and a unimodular polynomial* $B(x)$ *such that* $S(x) = A(x)P(x)B(x) = \mathrm{diag}(1, 1, x, x^4 - x^3 - x^2 + x, 0)$*. An analogous analysis shows that* $\mathrm{Rev} P(x)$ *is equivalent to* $\mathrm{diag}(1, 1, x, x^4 - x^3 - x^2 + x, 0)$*. We can therefore conclude that the complete eigenstructure of* $P(x)$ *is made of the left minimal index* 1*, the right minimal index* 0*, the finite elementary divisors* $x, x, x+1, (x-1)^2$*, and the infinite elementary divisors* $x - \infty, x - \infty$*.*

1.2.2 Root polynomials

In the present section we assume that the field $\mathbb{F}$ is algebraically closed.

Let $P(x) \in \mathbb{F}^{m \times n}[x]$. If $\{u_1(x), \ldots, u_s(x)\}$ is a minimal basis for $\ker P(x)$, we define $\ker_{x_0} P(x) := \mathrm{span}\,(\{u_1(x_0), \ldots, u_s(x_0)\}) \subseteq \mathbb{F}^n$. Notice that in general $\ker_{x_0} P(x)$ is a subset of $\ker P(x_0)$. It is a proper subset when x_0 is a characteristic value of $P(x)$, as is illustrated by the following example in which $\mathbb{F} = \mathbb{C}$.

Example 1.3. *Let*

$$P(x) = \begin{bmatrix} x & 1 & 0 & 0 \\ 0 & x & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x \end{bmatrix}.$$

Then a right minimal basis for $\ker P(x)$ *is* $\{[1, -x, x^2, 0]^T\}$. *Thus,* $\ker_0 P(x) = \mathrm{span}(\{[1, 0, 0, 0]^T\})$. *Evaluating the polynomial at* 0, *we get*

$$P(0) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

so that $\ker P(0) = \mathrm{span}(\{[1, 0, 0, 0]^T, [0, 0, 0, 1]^T\})$.

Remark 1.3. *Notice that a minimal basis is not uniquely determined. Thus, it may seem that the definition of* $\ker_{x_0} P(x)$ *depends on the choice of the particular minimal basis of* $\ker P(x)$ *that we have started from. This is not the case: in fact, let* $V(x)$ *be any different minimal bases of* $\ker P(x)$, *arranged in matrix form. Since* $U(x)$ *and* $V(x)$ *span the same subspace of* $\mathbb{F}(x)^n$, *we can write* $U(x)A(x) = V(x)$ *for some* $A(x) \in \mathbb{F}(x)^{s\times s}$, *with* $\mathrm{rank}(A(x)) = s$. *It is known [34] that* $U(x)$ *is minimal if and only if the equation* $b(x) = U(x)a(x)$, $b(x) \in \mathbb{F}[x]^n$, *implies* $a(x) \in \mathbb{F}[x]^s$, *and* $\deg b(x) = \max_i(\deg a_i(x) + \nu_i)$. *By applying this lemma to each column of* $V(x)$, *we find that every column of* $A(x)$ *must be a polynomial vector. Therefore,* $A(x)$ *must be a polynomial matrix. It is also known [34] that the greatest common divisor of all the* $s \times s$ *minors of a minimal basis must be* $1_{\mathbb{F}}$. *Since the minors of* $V(x)$ *are equal to the minors of* $U(x)$ *multiplied by* $\det A(x)$, *we see that the minimality of both* $M(x)$ *and* $N(x)$ *implies that* $A(x)$ *must be unimodular.*

This means that $A(x_0)$ *is nonsingular for any* $x_0 \in \mathbb{F}$. *Therefore,* $U(x_0)$ *and* $V(x_0) = U(x_0)A(x_0)$ *span the same subspace of* $\mathbb{F}^n$. *Or, equivalently, the space* $\ker_{x_0} P(x)$ *depends only on* $P(x)$ *and* x_0, *but not on the choice of the minimal basis in its definition.*

Given $w(x) \in (\mathbb{F}[x])^n$ and $x_0 \in \mathbb{F}$, suppose that $w(x) = (x - x_0)^i w_{(i)}(x)$ for some $w_{(i)}(x) \in (\mathbb{F}[x])^n$ if and only if $i \leq \ell$. In this instance we say that x_0 is a zero of order ℓ for $w(x)$.

The following definition [93] slightly modifies a definition given in [44] in order to extend it to the case of singular and/or rectangular polynomial matrices. A polynomial vector $v(x) \in (\mathbb{F}[x])^n$ is called a *root polynomial* of order ℓ corresponding to x_0 for $P(x) \in \mathbb{F}^{m\times n}[x]$ if the following conditions are met:

1. x_0 is a zero of order ℓ for $P(x)v(x)$;
2. $v_0 := v(x_0) \notin \ker_{x_0} P(x)$.

Observe that $v_0 \in \ker_{x_0} P(x) \Leftrightarrow \exists\ w(x) \in \ker P(x) \subseteq (\mathbb{F}(x))^n : w(x_0) = v_0$. In fact, let $U(x) = [u_1(x), \ldots, u_s(x)]$. Suppose that $w(x) = U(x)c(x)$ for some $c(x) \in (\mathbb{F}(x))^s$ and that $w(x_0) = v_0$: then $v_0 = U(x_0)c(x_0) \in \ker_{x_0} P(x)$[1]. Conversely, write $v_0 = U(x_0)c$ for some $c \in \mathbb{F}^s$ and notice that $U(x)c \in \ker P(x)$. Hence, condition 2. implies $v(x) \notin \ker P(x)$.

Root polynomials and Smith forms are related. In [44, Proposition 1.11] it is shown that given three *regular* matrix polynomials $P(x), A(x), B(x) \in M_n(x)$, and if x_0 is neither a root of $\det A(x)$ nor a root of $\det B(x)$, then $v(x)$ is a

[1] To be more precise, we should prove that $c(x_0) \in \mathbb{F}^s$. So suppose this is false. Then there exists an integer $\alpha \geq 1$ s.t. $d(x) := (x-x_0)^\alpha c(x)$ and $0 \neq d(x_0) \in \mathbb{F}^s$, so $0 = (x_0-x_0)^\alpha v(x_0) = U(x_0)d(x_0)$, absurd because $U(x_0)$ has rank s.

root polynomial of order ℓ corresponding to x_0 for $A(x)P(x)B(x)$ if and only if $B(x)v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $P(x)$.

The next theorem [93] shows that the result holds for a generic matrix polynomial.

Theorem 1.3. *Let $P(x) \in \mathbb{F}^{m\times n}[x]$, $A(x) \in M_m(\mathbb{F}[x])$ and $B(x) \in M_n(\mathbb{F}[x])$. Suppose that both $A(x_0)$ and $B(x_0)$, with $x_0 \in \mathbb{F}$, are full rank matrices. Then $v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $A(x)P(x)B(x)$ if and only if $B(x)v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $P(x)$.*

Proof. In [44], root polynomials are defined for regular square polynomial matrices, so that condition 2. reduces to $v(x_0) \neq 0$. Nevertheless, the proof given in [44, Proposition 1.11] for condition 1. does not actually use the regularity of $P(x)$, and it is therefore still valid when $P(x)$ is not a regular square polynomial matrix. To complete the proof: $v(x_0) \in \ker_{x_0} A(x)P(x)B(x) \Leftrightarrow \exists w_1(x) \in \ker A(x)P(x)B(x) : w_1(x_0) = v(x_0) \Leftrightarrow \exists w_2(x) \in \ker P(x) : w_2(x_0) = B(x_0)v(x_0) \Leftrightarrow B(x_0)v(x_0) \in \ker_{x_0} P(x)$. To build $w_2(x)$ from $w_1(x)$, simply put $w_2(x) = B(x)w_1(x)$ and use the fact that $A(x)$ is regular. To build $w_1(x)$ from $w_2(x)$, let $(B(x))^{-1}$ be the inverse matrix (over $\mathbb{F}(x)$) of $B(x)$, which exists because $B(x)$ is regular; then, put $w_1(x) = (B(x))^{-1}w_2(x)$. □

We say that the root polynomials corresponding to x_0 $v_1(x), \ldots, v_s(x)$ are x_0-*independent* if $v_1(x_0), \ldots, v_s(x_0)$ are linearly independent. We now wish to define a maximal set of x_0-independent root polynomials [93]. Let the orders of the x_0-independent root polynomials $v_1(x), \ldots, v_s(x)$ be $\ell_1 \leq \cdots \leq \ell_s$: then they form a maximal set if the following conditions are met.

1. It is not possible to find another root polynomial corresponding to x_0, say $v_{s+1}(x)$, such that $v_1(x), \ldots, v_s(x), v_{s+1}(x)$ are x_0-independent; in other words, there are no $(s+1)$-uples of x_0-independent root polynomials corresponding to x_0.

2. ℓ_s is the maximal possible order for the root polynomials of $P(x)$ corresponding to x_0.

3. For all $j = 1, \ldots, s-1$, there does not exist a root polynomial $\hat{v}_j(x)$ of order $\hat{\ell}_j > \ell_j$ such that $\hat{v}_j(x), v_{j+1}(x), \ldots, v_s(x)$ are x_0-independent.

The following theorem relates the elementary divisors and the root polynomials of a given matrix polynomial. Its proof is implicitly contained in [44] for the square and regular case and appears in [93] for the general case.

Theorem 1.4. *$P(x) \in (\mathbb{F}[x])^{m\times n}$ has a maximal set of x_0-independent root polynomials, of orders $\ell_1, \ldots, \ell_s$, if and only if $(x-x_0)^{\ell_1}, \ldots, (x-x_0)^{\ell_s}$ are the elementary divisors of $P(x)$ associated with x_0.*

Proof. Let $S(x)$ be the Smith form of $P(x)$, and recall that the inverse of a unimodular polynomial matrix is still a unimodular polynomial matrix [44]. Thus, in view of Theorem 1.3 and Theorem 1.2, it suffices to prove the thesis for $S(x)$. If $S(x)$ is the zero matrix, it has neither a root polynomial nor an elementary divisor, so there is nothing to prove. Otherwise, let ν be the maximal value of i such that $(S(x))_{ii} \neq 0_{\mathbb{F}[x]}$ and for $j = 1, \ldots, n$ let $e_j \in (\mathbb{F}[x])^n$ be the

polynomial vector such that $(e_j)_i = \delta_{ij}$. If $\nu < n$, $E := [e_{\nu+1}, \dots, e_n]$ is a minimal basis for $\ker S(x)$ and, being of order 0, also for $\ker_{x_0} S(x)$.

Suppose that $\{v_1(x), \dots, v_s(x)\}$ is a maximal set of x_0-independent root polynomials for $S(x)$. Let $k \leq \nu$ be the smallest index such that $(v_s(x_0))_k \neq 0_{\mathbb{F}}$: there must exist such an index because $v_s(x_0) \notin \ker_{x_0} S(x)$. Let $(S(x))_{kk} = (x - x_0)^\mu \theta(x)$, with $\mathrm{GCD}(x - x_0, \theta(x)) = 1_{\mathbb{F}[x]}$. We have that $(S(x)v_s(x))_k = (x - x_0)^\mu \theta(x)(v_s(x))_k$. By definition, x_0 must be a zero of order greater than or equal to ℓ_s of the latter scalar polynomial. However, notice that its order must be precisely ℓ_s: otherwise, e_k would be a root polynomial of order higher than ℓ_s, which is a contradiction because there cannot exist any root polynomials of higher order than $v_s(x)$. Therefore, $\mu = \ell_j$. Then consider $v_{s-1}(x)$, which by assumption is a root polynomial of maximal order ℓ_{s-1} between all the root polynomials that are x_0-independent of $v_s(x)$. Let k' be the smallest index not equal to k and such that $(v_{s-1}(x_0))_{k'} \neq 0_{\mathbb{F}}$. If such an index does not exist, then $v_{s-1}(x_0)$ is, up to a vector lying in $\ker_{x_0} S(x)$, a multiple of e_k and thus $\ell_{s-1} = \ell_s$: in this case, without any loss of generality, replace $v_{s-1}(x)$ with a suitable linear combination of $v_{s-1}(x)$ and $v_s(x)$. Following an argument similar as above, we can show that $(S(x))_{k'k'} = (x - x_0)^{\ell_{s-1}} \hat{\theta}(x)$, $\mathrm{GCD}(x - x_0, \hat{\theta}(x)) = 1_{\mathbb{F}[x]}$. Next, consider the root polynomial $v_{s-2}(x)$ of maximal order ℓ_{s-2} between all the root polynomials that are x_0-independent of $v_s(x)$ and $v_{s-1}(x)$, and so on until it is proved that the elementary divisors of $S(x)$ associated with x_0 are $(x - x_0)^{\ell_1}, \dots, (x - x_0)^{\ell_s}$. There cannot be other elementary divisors associated with x_0, otherwise $\dim \ker S(x_0) - \dim \ker_{x_0} S(x) \geq s+1$, and therefore it would be possible to find an $(s + 1)$-uple of x_0-independent root polynomials, which contradicts the hypothesis.

Conversely, assume that $(x-x_0)^{\ell_1}, \dots, (x-x_0)^{\ell_s}$ are the elementary divisors of $P(x)$ corresponding to x_0. This means that we must have $(S(x))_{\nu\nu} = (x - x_0)^{\ell_s} \phi(x)$, with $\mathrm{GCD}(x - x_0, \phi(x)) = 1_{\mathbb{F}[x]}$. Clearly, e_ν is a root polynomial of order ℓ_s corresponding to x_0 for $S(x)$, and its order is maximal. Then we go on with similar arguments, and it can be easily checked that $\{e_{\nu-s+1}, \dots, e_\nu\}$ is a maximal set of x_0-independent root polynomials. □

1.3 Polynomial eigenvalue problems

Assume now that the underlying field is $\mathbb{C}$. For $j = 0, \dots, k$ let $P_j \in \mathbb{C}^{n \times n}$, $P_k \neq 0$, be constant matrices and consider the matrix polynomial

$$P(x) = \sum_{j=0}^{k} P_j x^j \in M_n(\mathbb{C}[x]).$$

Suppose moreover that $P(x)$ is regular. The generalised polynomial eigenproblem (PEP) associated with $P(x)$ is to find an eigenvalue x_0 such that there exists a nonzero vector $v_0 \in \mathbb{C}^n$ satisfying

$$P(x_0)v_0 = 0. \tag{1.2}$$

If x_0 is an eigenvalue of $P(x)$, then the set $\{v_j\}_{j=0,\dots,\ell}$ is called a Jordan chain of length $\ell + 1$ associated with x_0 if $v_0 \neq 0$ and the following relations

hold [44]:

$$\sum_{i=0}^{j} \frac{P^{(j-i)}(x_0)}{(j-i)!} v_i = 0, \quad j = 0, \ldots, \ell, \tag{1.3}$$

where $P^{(k)}(x_0)$ denotes the k-th derivative of $P(x)$ evaluated at $x = x_0$. The case $j = 0$ corresponds to the definition of an eigenvector.

Remark 1.4. *The notion of a Jordan chain can be extended to any matrix function $F : \mathbb{C} \to \mathbb{C}^{n\times n}$ whose determinant vanishes at x_0, as long as $F(x)$ is analytic in a neighbourhood of x_0.*

Let $g \geq k$ be the grade of $P(x)$. It is agreed that, by definition, ∞ is an eigenvalue of $P(x)$ associated with the Jordan chain $v_0, \ldots, v_\ell$ if 0 is an eigenvalue of $\mathrm{Rev}_g P(x)$ associated with the same Jordan chain.

It is easy to show that the eigenvalues are in fact the characteristic values of $P(x)$, while the eigenvectors and the Jordan vectors are related to the root polynomials of $P(x)$. The following theorem [44] gives the details.

Theorem 1.5. *Let $P(x) \in M_n(\mathbb{C}[x])$ be a square regular matrix polynomial, and let $\mathbb{C}^* := \mathbb{C} \cup \infty$. Then*

- *$x_0 \in \mathbb{C}^*$ is an eigenvalue of $P(x)$ if and only if it is a characteristic value of $P(x)$.*
- *$\{v_j\}_{j=0,\ldots,\ell}$ is a Jordan chain associated with x_0 if and only if $v(x) = \sum_{j=0}^{\ell} v_j (x - x_0)^j$ is a root polynomial of order $\ell + 1$ for $P(x)$.*

Proof. x_0 can be an eigenvalue of $P(x)$ if and only if $P(x_0)$ is singular. By Theorem 1.2, this happens if and only if x_0 is a characteristic value of $P(x)$.

Now expand $P(x) = \sum_{j=0}^{k} P^{(j)}(x_0)(x - x_0)^j$. $v(x)$ is a root polynomial of order $\ell+1$ if and only if $P(x)v(x)$ has a root of order $\ell+1$ at x_0. By equating to zero the coefficients of $(x-x_0)^j$ in $P(x)v(x)$ for $j = 0, \ldots, \ell$ we obtain equations (1.3). □

If x_0 is an eigenvalue, an interesting question to ask is what its geometric multiplicity is, i.e. what is the number $\nu(x_0)$ of the associated linearly independent eigenvectors. Equivalently, $\nu(x_0)$ is the maximal number of Jordan chains corresponding to x_0 whose eigenvectors are linearly independent. Theorem 1.4 and Theorem 1.5 allow us to deduce that $\nu(x_0)$ is equal to the number of invariant polynomials of $P(x)$ that are divided by $x - x_0$, or equivalently the number of elementary divisors of $P(x)$ that have the form $(x - x_0)^{k_i}$. The maximal length of each of such Jordan chains is given by the values of the exponents k_i (sometimes called the *partial multiplicities* of $P(x)$ at x_0 [44]). If one starts with a maximal set of x_0-independent root polynomials, Theorem 1.5 shows how to build a set of $\nu(x_0)$ Jordan chains of maximal length and such that the eigenvectors are linearly independent. Such a set is sometimes referred to as a *canonical set* of Jordan chains [44].

On the other hand, it is clear that the algebraic multiplicity of x_0 is equal to $\sum_{i=0}^{\nu(x_0)} k_i$. An eigenvalue is said to be *semisimple* if its algebraic multiplicity is equal to its geometric multiplicity. If an eigenvalue is not semisimple, it is said to be *defective*.

Remark 1.5. *In Chapter 4, we will often use the expression* Jordan structure of $P(x)$ at x_0 *to refer to the number and maximal length of the Jordan chains that form a canonical set associated with x_0, or equivalently to the number and the value of all the partial multiplicities at x_0. From the discussion above it is clear that the Jordan structure at x_0 is completely determined by the exponents $k_1, \ldots, k_{\nu(x_0)}$ that appear in the elementary divisors $(x-x_0)^{k_1}, \ldots, (x-x_0)^{k_{\nu(x_0)}}$.*

A simple corollary of Theorem 1.5 is that $x_0 \in \mathbb{C}$ is an eigenvalue if and only if it is a root of the characteristic equation

$$\det P(x) = 0. \tag{1.4}$$

Moreover, by definition, ∞ is an eigenvalue of algebraic multiplicity m if and only if $\deg \det P(x) = gn - m$, where n is the dimension of the square matrix coefficients of $P(x)$ and g is the grade of $P(x)$. In the following chapters we will always assume that $g = k$ unless otherwise stated.

Notice that, in contrast to what happens in the classical eigenvalue problem, eigenvectors corresponding to different eigenvalues are not necessarily all linearly independent. In fact, an $n \times n$ matrix polynomial $P(x)$ of degree k can have as many as nk eigenvectors, while for obvious dimensional reasons no more than n vectors can be linearly independent. Not even the Jordan vectors of a given Jordan chain are necessarily independent. In fact, Jordan vectors are also allowed to be zero. Moreover, in general they are not uniquely determined. The following example illustrates this fact.

Example 1.4. *Let* $P(x) = \begin{bmatrix} (x-1)^4 & x(x-1)^2 \\ 0 & (x-1)^4 \end{bmatrix}$. *Let us pick* $\operatorname{grade}(P(x)) = \deg P(x) = 4$.

The only eigenvalue is 1. *Since* $P(1) = 0$, *any vector in* $\mathbb{C}^2$ *is an eigenvector. Let us first pick* $\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$ *as an eigenvector. Then* $\{\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right], \left[\begin{smallmatrix} \alpha \\ \beta \end{smallmatrix}\right]\}$ *is a Jordan chain of length* 2 $\forall\ \alpha, \beta \in \mathbb{C}$. *If we pick the eigenvector* $\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$, *then for any possible choice of the complex parameters* $\gamma, \delta, \epsilon, \phi, \iota, \lambda, \mu$ *the following is a Jordan chain of length* 6*:* $\{\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} \gamma \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} \delta \\ -1 \end{smallmatrix}\right], \left[\begin{smallmatrix} \epsilon \\ 1-\gamma \end{smallmatrix}\right], \left[\begin{smallmatrix} \phi \\ \iota \end{smallmatrix}\right], \left[\begin{smallmatrix} \lambda \\ \mu \end{smallmatrix}\right]\}$,

Remark 1.6. *The notion of eigenvalues can be extended to a singular or a rectangular $P(x)$, by defining them as numbers x_0 such that*

$$\ker_{x_0} P(x) \neq \ker P(x_0).$$

Equivalently, x_0 is an eigenvalue if

$$\operatorname{rank} P(x_0) < \operatorname{rank} P(x).$$

The rank in the latter formula is to be interpreted as a function from $(\mathbb{C}[x])^{m\times n}$ to $\mathbb{N}$. In the left hand side of the inequality, it coincides of course with the rank of $P(x_0)$ as an element of $\mathbb{C}^{m\times n}$.

As we have defined right eigenvectors and right Jordan chains, it is of course possible to define left eigenvectors and left Jordan chains. For instance, a nonzero row vector $w \in \mathbb{C}^{1\times n}$ is a left eigenvector of $P(x)$ corresponding to the eigenvalue x_0 if $wP(x_0) = 0$. Similarly, left Jordan chains can be defined.

We conclude the present subsection by recalling that also the notion of invariant subspaces can be generalised to matrix polynomials (or, even more in

general, to analytic matrix functions). This leads to the concept of standard pairs and standard triples of a matrix polynomial $P(x) \in M_n(\mathbb{C}[x])$ of degree k [44]. Namely, a standard pair (T, V), $T \in M_{kn}(\mathbb{C})$, $V \in \mathbb{C}^{n\times nk}$, is a couple of matrices such that

1. the $nk \times nk$ matrix $U := \begin{bmatrix} V \\ VT \\ \vdots \\ VT^{k-1} \end{bmatrix}$ is nonsingular;
2. $\sum_{i=0}^{k} P_i V T^i = 0$.

Moreover, if $W = U^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ I_n \end{bmatrix}$, then (T, V, W) is called a standard triple. A list of properties of standard pairs and standard triples may be found in [44], while the study of the special case of standard triples of structured matrix polynomials is thoroughly pursued in [2]. The reader interested in the further generalization to the so-called invariant pairs of matrix functions may refer to [57].

1.4 Linearizations

Let $Q(x) \in (\mathbb{F}[x])^{m\times n}$ be a matrix polynomial of grade g. Let $M(x) = xX + Y$, with $X, Y \in \mathbb{F}^{(m+p)\times(n+p)}$, be a matrix polynomial of degree 1 (sometimes referred to as a *pencil*). We say that $M(x)$ is a linearization of $Q(x)$ if there exists a number $p \in \mathbb{N}$ such that $M(x)$ is equivalent to

$$\begin{bmatrix} Q(x) & 0 \\ 0 & I_p \end{bmatrix}. \tag{1.5}$$

Being equivalent to such an embedding of $Q(x)$ in a larger polynomial matrix, it is obvious that a linearization has the same finite elementary divisors of the linearised polynomial. In fact, if $S_Q(x)$ is the Smith form of $Q(x)$, it is straightforward to show that the Smith form of $\operatorname{diag}(Q(x), I_p)$ is $\operatorname{diag}(I_p, S_Q(x))$. Some bounds on the allowed values of the number p, and thus on the dimension of linearizations, were obtained in [21].

Furthermore, the linearization $M(x)$ is said to be strong whenever its reversal, $\operatorname{Rev}(M(x)) = xY + X$, is a linearization of $\operatorname{Rev}_g(Q(x))$. Strong linearizations preserve also infinite elementary divisors of the linearised polynomial. Notice that the definition we just gave depends on the grade of the matrix polynomial, which is arbitrary and whose choice change the infinite elementary divisors of the polynomial (see Proposition 3.2 for more details). Consequently, if g_1 and g_2, $g_1 > g_2 \geq k = \deg Q(x)$, are two different possible choices for the grade of $Q(x)$, then strong linearizations of $Q(x)$ with respect to g_1 and strong linearizations of $Q(x)$ with respect to g_2 actually have different infinite elementary divisors. Throughout the thesis, whenever we speak of a strong linearization of a matrix polynomial without specifying the grade of the polynomial, it is tacitly agreed that we are referring to a linearization that is strong *with respect to the degree* of the matrix polynomial.

In the following, we will be specifically interested in the case $\mathbb{F} = \mathbb{C}$, and we will focus on square regular matrix polynomials. It was proved in [21] that in

such an instance all strong linearizations must have $p = (g-1)n$. Since strong linearizations provide the most desirable numerical properties, we will use in the following the next, more restrictive, definition.

Let $P(x) \in M_n(\mathbb{C}[x])$ be a regular square matrix polynomial of grade g. A linearization of $P(x)$ is defined as a pencil $L(x) = xX + Y$, with $X, Y \in \mathbb{C}^{gn \times gn}$, such that there exist unimodular polynomial matrices $E(x)$ and $F(x)$ for which

$$E(x)L(x)F(x) = \begin{bmatrix} P(x) & 0 \\ 0 & I_{(g-1)n} \end{bmatrix}. \tag{1.6}$$

Once again, this definition depends on the grade, and unless otherwise stated we will assume that the grade is equal to the degree.

We will now recall some of the most important classes of linearizations, starting with the well-known *companion forms* [44].

If $P(x) = \sum_{i=0}^{g} P_i x^i$, the first companion linearization is:

$$C_f(x) = \begin{bmatrix} I_n & & & \\ & I_n & & \\ & & \ddots & \\ & & & P_g \end{bmatrix} x + \begin{bmatrix} 0_n & -I_n & \dots & 0_n \\ \vdots & \ddots & \ddots & \vdots \\ 0_n & \dots & 0_n & -I_n \\ P_0 & \dots & P_{g-2} & P_{g-1} \end{bmatrix}, \tag{1.7}$$

while the second companion linearization is

$$C_s(x) = \begin{bmatrix} I_n & & & \\ & I_n & & \\ & & \ddots & \\ & & & P_g \end{bmatrix} x + \begin{bmatrix} 0_n & \dots & 0_n & P_0 \\ -I_n & \ddots & \vdots & \vdots \\ \vdots & \ddots & 0_n & P_{g-2} \\ 0_n & \dots & -I_n & P_{g-1} \end{bmatrix}. \tag{1.8}$$

Several generalizations of companion pencils exist. Following the work of [78], in the paper [59] the two (right and left) *ansatz vector linearization spaces* are studied: having introduced the vector $\Lambda := [1, x, \dots, x^{g-1}]^T$, these spaces are defined as follows:

$$\hat{\mathcal{L}}_1 := \{L = xX + Y : \exists v \in \mathbb{C}^g s.t. L \cdot (\Lambda \otimes I_n) = v \otimes P(x)\} \tag{1.9}$$

$$\hat{\mathcal{L}}_2 := \{L = xX + Y : \exists w \in \mathbb{C}^g s.t. (\Lambda^T \otimes I_n) \cdot L = w^T \otimes P(x)\}. \tag{1.10}$$

Companion pencils (1.7) and (1.8) belong to these spaces.

It is shown in [78] that almost every pencil in $\hat{\mathcal{L}}_1$ and $\hat{\mathcal{L}}_2$ is a linearization, while in [59] two binary operations on block matrices, called column shifted sum and row shifted sum, are first introduced and then used to characterise the above defined spaces.

In particular the following result [78, Theorem 4.3] holds.

Proposition 1.2. *Let $L(x) \in \hat{\mathcal{L}}_1$ ($\hat{\mathcal{L}}_2$) and let $P(x)$ be a regular matrix polynomial. The following properties are equivalent:*

- *$L(x)$ is a linearization of $P(x)$;*
- *$L(x)$ is a strong linearization of $P(x)$;*

- $L(x)$ *is regular.*

The linearization space $\mathbb{DL} := \hat{\mathcal{L}}_1 \cap \hat{\mathcal{L}}_2$, characterised as the space of all block-symmetric pencils, is also studied in [59, 78]. It has many interesting features, such as elegant results about eigenvalue condition numbers [60]. Namely, in [60] it was shown that some specific pencils within $\mathbb{DL}$ present particularly nice properties with respect to conditioning.

Another generalization of companion pencils is realised by Fiedler pencils and generalised Fiedler pencils. They have been first introduced for scalar polynomials in [33], and then generalised and further studied in [5, 16, 22, 23]. Fiedler pencils (that include companion pencils as a special case) can also be defined as linearizations of rectangular matrix polynomials [23]; moreover, Fiedler linearizations are always strong [23].

Finally, it is possible to consider linearizations in polynomial bases different than the usual ones. This subject will be analysed in more detail in Chapter 4.

We conclude this brief review on theoretical properties of linearizations by mentioning that, when singular matrix polynomials are linearised, it is possible to link the minimal indices of the linearization to the minimal indices of the linearised polynomials. An extensive study of such properties has been recently made. See [16, 22, 24].

Linearizations are important in numerical analysis, because the customary approach for the numerical solution of polynomial eigenproblems consists in two steps. First the matrix polynomial $P(x)$ is linearised into a matrix pencil $L(x)$, and then the eigenvalues of $L(x)$ are computed by some iterative solver. The most common choice is the matrix QZ algorithm, introduced in [88] and later refined in many ways (see, e.g., [47, 67, 109]). The QZ algorithm applied to a companion linearization of $P(x)$ is implemented in the Matlab function `polyeig`. The use of (1.7) or (1.8), however, may not be the best choice in terms of eigenvalue condition number [60]. There exist other pieces of software – e.g. `quadeig` [52], specifically designed for matrix polynomials of grade 2 – that we will mention in the thesis, and that also exploit a linearization of the matrix polynomial.

In this thesis we will propose a different numerical approach, advantageous in the case of high-degree matrix polynomials, that does not necessarily need a linearization to compute the sought eigenvalues. If the dimension of the matrix polynomial is high with respect to its degree, a linearization step is still useful with our approach in order to achieve a competitive efficiency; yet, we stress that in contrast to the classical approach this preliminary step is optional. This allows to avoid any potential problem related to the use of linearizations.

Chapter 2

The Ehrlich-Aberth method for polynomial eigenvalue problems

The present chapter introduces a generalization of a root-finding method, known as the Ehrlich-Aberth iteration, to polynomial eigenvalue problems.

It is based mainly on the paper [12]; the parts dealing with the implementation of the method via a linearization are taken from [42].

2.1 The Ehrlich-Aberth method for matrix polynomials

Given two positive integers $k,\ n$ and matrices $P_j \in \mathbb{C}^{n\times n}$, $j = 0, \ldots, k$ consider the matrix polynomial

$$P(x) = \sum_{j=0}^{k} P_j x^j \tag{2.1}$$

where $P_k \neq 0$, so that $P(x)$ has degree k, and define the scalar polynomial $p(x) = \det P(x)$ of degree $N \leq nk$. Assume that $P(x)$ is regular, that is $p(x)$ is not identically zero.

As we saw in Chapter 1, the polynomial eigenvalue problem associated with $P(x)$ consists in computing the roots of the polynomial $p(x)$ which are called the eigenvalues of the matrix polynomial $P(x)$. Observe that, if P_k has not full rank, then $N < nk$. In this case, we recall that it is convenient to introduce $nk - N$ eigenvalues at infinity and say that the matrix polynomial $P(x)$ has nk eigenvalues including the $nk - N$ eigenvalues at infinity. In other words, the grade of $p(x)$ is always taken equal to nk.

In the present chapter our interest is addressed to the design and analysis of efficient algorithms for PEPs based on the Ehrlich-Aberth iteration [1, 14, 27].

The Ehrlich-Aberth iteration (EAI) was historically first mentioned in [14] and afterwards independently rediscovered many times. It is one of the many simultaneous iteration techniques available in the literature for the numerical

approximation of polynomial roots [81, 96]. In [8, 9] the EAI has been combined with various techniques like the Rouché theorem, the Newton polygon technique, and the Gerschgorin inclusion theorems in order to arrive at efficient and robust software implementations. The package `Polzeros`, designed in [8], provides a robust and reliable tool for approximating roots of polynomial in floating point arithmetic. The package `MPSolve` designed in [9] provides certified approximations to any desired number of digits of the roots of any polynomial. As an example of application of `MPSolve`, we point out that in [15] it has been used to compute the roots of a polynomial of degree $70,000$ with integer coefficients represented with hundreds digits. The EAI has been used in [10] to solve the generalised tridiagonal eigenvalue problem where the software provides effective accelerations in terms of CPU time. It has been used in [97] for quadratic hyperbolic tridiagonal eigenvalue problems.

Given a vector $y^{(0)} \in \mathbb{C}^N$ of initial guesses for the N roots of the polynomial $p(x)$, the EAI provides the sequence of simultaneous approximations $y^{(i)}$ given by

$$\begin{aligned} y_j^{(i+1)} &= y_j^{(i)} - \frac{\mathcal{N}(y_j^{(i)})}{1 - \mathcal{N}(y_j^{(i)})\mathcal{A}_j(y^{(i)})}, \quad \mathcal{N}(x) = \frac{p(x)}{p'(x)} \\ \mathcal{A}_j(y^{(i)}) &= \sum_{\ell=1,\ell\neq j}^{N} \frac{1}{y_j^{(i)} - y_\ell^{(i)}} \end{aligned} \tag{2.2}$$

where $\mathcal{N}(x)$ is the Newton correction.

Besides the Jacobi-style version of EAI we may formulate the Gauss-Seidel version of EAI, that is

$$\begin{aligned} y_j^{(i+1)} &= y_j^{(i)} - \frac{\mathcal{N}(p(y_j^{(i)}))}{1 - \mathcal{N}(p(y_j^{(i)}))\mathcal{A}_j(y^{(i)}, y^{(i+1)})}, \\ \mathcal{A}_j(y^{(i)}, y^{(i+1)}) &= \sum_{\ell=1}^{j-1} \frac{1}{y_j^{(i)} - y_\ell^{(i+1)}} + \sum_{\ell=j+1}^{N} \frac{1}{y_j^{(i)} - y_\ell^{(i)}}. \end{aligned} \tag{2.3}$$

The above formulae (2.2), (2.3)can be derived [81, 96] by applying Newton's method to the rational function

$$r(x) = \frac{p(x)}{\prod_{\ell=1,\ell\neq j}^{N}(x - \xi_\ell)},$$

where ξ_ℓ are the roots of $p(x)$. Therefore, the EAI provides a way to implement the implicit deflation of the roots.

The method, in the Jacobi version, is known to converge cubically for simple roots and linearly for multiple roots [96]. In the Gauss-Seidel version, convergence is slightly faster. In our implementations of the method, the Gauss-Seidel EAI is the default choice. In practice, good global convergence properties are observed; a theoretical analysis of global convergence, though, is still missing and constitutes an open problem.

With the term *vector iteration* of the Ehrlich-Aberth method we refer to the step which provides the vector $y^{(i+1)}$ given the vector $y^{(i)}$. We use the term *scalar iteration* to indicate the single step performed on the generic scalar component of the vector $y^{(i)}$.

In the case of a scalar polynomial of degree N the cost of a scalar iteration is $O(N)$ arithmetic operations. In this way, the cost of a vector iteration is at most $O(N^2)$ floating point operations. Yet, the cost is substantially reduced when most components have been numerically approximated, so that few scalar iterations must be performed in order to carry out the vector iteration.

The number of scalar iterations needed by the floating point implementation in order to find approximations which are exact roots of a slightly perturbed polynomial is, in practice, $O(N)$ if the starting approximations are computed by means of the Newton polygon technique [8]. This technique is particularly effective when the polynomial has roots with moduli which are very unbalanced.

Crucial aspects for an effective implementation of the EAI for polynomial eigenvalue problems are:

1. the computation of the Newton correction $p(x)/p'(x)$ given the value of x and of the input coefficients P_j, $j = 0, \ldots, k$;
2. a criterion for stopping the iterations;
3. the choice of the initial approximations.

In the following sections, we discuss in further detail each of such issues.

2.2 Newton correction

In the literature, methods based on some factorizations of $P(x)$ were developed to compute the Newton correction for functions that have the same zeros of $p(x)$: e.g., the algorithm in [69], later proved to lack theoretical rigour and corrected in [64], or the method in [65]. Other kinds of Newton-like approaches were presented in [98].

If one wishes to work with $p(x)$ itself, a naive way to compute the Newton correction $p(x)/p'(x)$ would be to evaluate first the coefficients of the polynomial $p(x)$, say, by means of the evaluation-interpolation technique, and then to apply right after the Ehrlich-Aberth method to the scalar equation $p(x) = 0$. This approach would however come across numerical problems due to numerical instability and to overflow and underflow situations encountered in the computation of determinants.

It is therefore wise to conceive a strategy to avoid the explicit calculation of the coefficients of $p(x)$.

We propose an effective way to compute the Newton correction for $p(x)$. It rests upon the well-known Jacobi's formula [45] for the differential of the determinant of any invertible square matrix A:

$$d(\ln(\det A)) = \operatorname{tr}(A^{-1}dA). \tag{2.4}$$

Two main options are available at this point. The first is to use (2.4) with $A = P(x)$. The second is to pass through a linearization of the matrix polynomial. In fact, if $L(x)$ is a linearization of $P(x)$, then $\det P(x) = \text{const.} \cdot \det L(x)$. Thus, equation (2.4) with $A = L(x)$ is another possibility. In the following subsections, we will give more detailed comments for each of these two choices.

We mention that, even though we independently formulated it, we found out later on that the possible use of (2.4) in a numerical method for PEPs had

been already suggested in [37, 70]. However, in these instances it was proposed to use it to apply the Newton method to approximate each single eigenvalue in sequence, mentioning the possibility to use an implicit deflation of previously found roots [82], in order to avoid that the method converges twice to the same eigenvalue. This leads to a formula akin to (2.2) and (2.3), with the difference that the summation in the term $\mathcal{A}_j$ is performed only up to the number of roots that have already been approximated. This is a crucial detail, because such a sequential implementation of the Newton method does not seem to achieve the same efficiency that the EAI accomplishes (see Section 2.6).

2.2.1 Computing the Newton correction via a linearization

One method for computing the Newton correction $p(x)/p'(x)$ is to evaluate it as $\det(L(x))/(\det(L(x)))'$, where $L(x) = xE + F$ with $E, F \in \mathbb{C}^{nk\times nk}$ and $L(x)$ is a linearization of $P(x)$. More specifically, a convenient choice is $L(x) = C_f(x)$ in (1.7), so that E is block-diagonal and F is block-Hessenberg. This block structure can be exploited in order to compute the Newton correction.

The formula of Jacobi [45] in this case reads

$$(\det(L(x)))' = \det(L(x))\mathrm{tr}(L^{-1}(x)L'(x)) = \det(L(x))\mathrm{tr}(L^{-1}(x)E)$$

which reduces the evaluation of $\det(L(x))/(\det(L(x)))'$ to computing the trace of $L^{-1}(x)\cdot E$. In the sequel we describe a method for finding the block entries and, *a fortiori*, the trace of the inverse of $L(x)$ from the LQ factorization of the matrix. Then we slightly modify the computation to take into account the contribution due to the matrix E.

We denote as $\mathcal{G}(\theta,\psi)$ the 2×2 unitary Givens rotation given by

$$\mathcal{G}(\theta,\psi) = \begin{bmatrix} \theta & \psi \\ -\bar{\psi} & \bar{\theta} \end{bmatrix}, \quad |\theta|^2 + |\psi|^2 = 1.$$

Let $L(x) = \tilde{L}\cdot Q$ be the (block) LQ factorization of $L(x)$ obtained by means of Givens rotations so that

$$\begin{array}{l} L(x)\mathcal{G}_1\cdot\mathcal{G}_2\cdots\mathcal{G}_{k-1} = \tilde{L}, \quad Q^H = \mathcal{G}_1\cdot\mathcal{G}_2\cdots\mathcal{G}_{k-1}, \\ \mathcal{G}_j = I_{n(j-1)} \oplus (\mathcal{G}(\theta_j,\psi_j)\otimes I_n) \oplus I_{n(k-1-j)}. \end{array} \tag{2.5}$$

The action of G_j is to rotate the jth and $(j+1)$th block columns of $L(x)$; the parameters θ_j and ψ_j are chosen in such a manner that

$$\begin{bmatrix} 0_n & 0_n \\ \vdots & \vdots \\ 0_n & 0_n \\ c_jI_n & -I_n \\ 0_n & xI_n \\ 0_n & 0_n \\ \vdots & \vdots \\ 0_n & 0_n \\ \star & \star \end{bmatrix} \mathcal{G}_j = \begin{bmatrix} 0_n & 0_n \\ \vdots & \vdots \\ 0_n & 0_n \\ \alpha_jI_n & 0_n \\ \beta_jI_n & c_{j+1}I_n \\ 0_n & 0_n \\ \vdots & \vdots \\ 0_n & 0_n \\ \star & \star \end{bmatrix},$$

where the block elements denoted by the symbol $\star$ are not specified. Hence, proceeding by induction on j it can be easily proved that the lower triangular factor $\tilde{L}$ has the following structure

$$\tilde{L} = \begin{bmatrix} \alpha_1 I_n & & & & \\ \beta_1 I_n & \alpha_2 I_n & & & \\ & \ddots & \ddots & & \\ & & \beta_{k-2} I_n & \alpha_{k-1} I_n & \\ \hat{P}_0 & \hat{P}_1 & \dots & \hat{P}_{k-2} & \hat{P}_{k-1} \end{bmatrix},$$

where $\alpha_j \neq 0$, $1 \leq j \leq k-1$. If $\hat{P}_{k-1}$ and, therefore, $L(x)$ is invertible then the LQ factorization can be used to find a condensed representation of the inverse of $L(x)$. Observe that $L^{-1}(x) = Q^H \cdot \tilde{L}^{-1}$. In order to take into account the occurrence of the matrix E in the Jacobi formula let us introduce the matrix $\tilde{P}_k = \hat{P}_{k-1}^{-1} \cdot P_k$. Then we have the following proposition.

Proposition 2.1. *There exist matrices $\tilde{P}_1, \dots, \tilde{P}_{k-1} \in \mathbb{C}^{n \times n}$ such that*

$$L^{-1}(x)E = \begin{bmatrix} \tilde{P}_1 & \psi_1 \tilde{P}_2 & \dots & \psi_1 \cdots \psi_{k-1} \tilde{P}_k \\ & \bar{\theta}_1 \tilde{P}_2 & & \\ & & \ddots & \\ & & & \bar{\theta}_{k-1} \tilde{P}_k \end{bmatrix},$$

where the blank entries are not specified.

Proof. The proof basically follows by applying the (block) Schur decomposition (2.5) of Q^H to the block lower triangular factor $\tilde{L}^{-1}E$. To show it more formally we can proceed by induction. Let us assume that the the $j-$th block row of $\mathcal{G}_j \cdots \mathcal{G}_{k-1} \tilde{L}^{-1} E$ can be represented as

$$\begin{bmatrix} \star & \dots & \star & \tilde{P}_j & \psi_j \tilde{P}_{j+1} & \dots & \psi_j \cdots \psi_{k-1} \tilde{P}_k \end{bmatrix},$$

where $\tilde{P}_j$ is the diagonal entry and the value of the entries in the strictly lower triangular part – denoted by $\star$– is not essential. Then, by applying $\mathcal{G}_{j-1}$ on the left of the matrix we find that the $(j-1)-th$ block row looks like

$$\begin{bmatrix} \star & \dots & \star & \tilde{P}_{j-1} & \psi_{j-1} \tilde{P}_j & \dots & \psi_{j-1} \cdots \psi_{k-1} \tilde{P}_k \end{bmatrix},$$

while the diagonal entry in position j becomes $\bar{\theta}_{j-1} \tilde{P}_j$. □

This result says that the block diagonal entries of $L^{-1}(x)$ can be determined from the entries in its first (block) row. The computation of this row is equivalent to the solution of the linear system

$$[I_n, 0_n, \dots, 0_n] = [X_1, \dots, X_k] \cdot L(x)$$

or, equivalently,

$$[I_n, 0_n, \dots, 0_n] \cdot Q^H = [X_1, \dots, X_k] \cdot \tilde{L}.$$

In view of the structure of Q^H this reduces to

$$\left[\theta_1, \theta_2 \psi_1, \dots, \theta_{k-1} \prod_{j=1}^{k-2} \psi_j, \prod_{j=1}^{k-1} \psi_j \right] \otimes I_n = [X_1, \dots, X_k] \cdot \tilde{L}.$$

Let $D \in \mathbb{C}^{nk \times nk}$ be a block diagonal matrix defined by

$$D = \mathrm{diag}(1, \psi_1, \ldots, \prod_{j=1}^{k-2} \psi_j, \prod_{j=1}^{k-1} \psi_j) \otimes I_n.$$

Using the matrix D to balance the coefficient matrix yields

$$[\theta_1, \theta_2, \ldots, \theta_{k-1}, 1] \otimes I_n = [X_1, \ldots, X_k]\, D^{-1} \cdot D \cdot \tilde{L} \cdot D^{-1}.$$

Observe that

$$\left[\hat{X}_1, \ldots, \hat{X}_k\right] := [X_1, \ldots, X_k]\, D^{-1} = \left[\tilde{P}_1, \ldots, \tilde{P}_{k-2}, \hat{P}_k^{-1}\right],$$

and, therefore, the solution of

$$[\theta_1, \theta_2, \ldots, \theta_{k-1}, 1] \otimes I_n = \left[\hat{X}_1, \ldots, \hat{X}_k\right] \hat{L}, \quad D\tilde{L}D^{-1} = \hat{L},$$

gives the desired unknown matrices $\tilde{P}_1, \ldots, \tilde{P}_{k-1}$. To achieve some computational savings we rewrite the system as

$$[\theta_1, \theta_2, \ldots, \theta_{k-1}, 1] \otimes \hat{P}_{k-1} = \left[\tilde{X}_1, \ldots, \tilde{X}_k\right] \hat{L}$$

and thus we arrive at the following relation

$$\det(L(x))' / \det(L(x)) = \mathrm{tr}(\hat{P}_{k-1}^{-1}(\tilde{X}_1 + \bar{\theta}_1 \tilde{X}_2 + \ldots + \bar{\theta}_{k-2}\tilde{X}_{k-1} + \bar{\theta}_{k-1}P_k)),$$

which is used to compute the reciprocal of the Newton correction. The function **trace_linearization** below implements our resulting algorithm at the cost of $\mathcal{O}(n^2k + n^3)$ operations.

function trace_linearization
Input: $P_0, \ldots, P_k \in \mathbb{C}^{n \times n}$, $x \in \mathbb{C}$, $(\det(P(x)) \neq 0)$
Output: the value of $\eta = p'(x)/p(x)$

$P_{k-1} = P_{k-1} + xP_k$;
$\alpha = x\ ones(1,k)$; $\beta = zeros(1, k-1)$;
$\chi = -ones(1,k)$;
for $j = 1, \ldots, k$
  $v = [\alpha_j; \chi_j]$; $\mathcal{G}^T = planerot(v)$; $q(j,:) = \mathcal{G}(1,:)$; $c_j = q(j,1)$;
  $\alpha_j = \alpha_j \mathcal{G}_{1,1} + \chi_j \mathcal{G}_{2,1}$; $\tilde{\beta} = \beta_j \mathcal{G}_{1,1} + \alpha_{j+1}\mathcal{G}_{2,1}$;
  $\alpha_{j+1} = \beta_j \mathcal{G}_{1,2} + \alpha_{j+1}\mathcal{G}_{2,2}$; $\beta_j = \tilde{\beta}$;
  $\tilde{P} = \mathcal{G}_{1,1}P_{j-1} + \mathcal{G}_{2,1}P_j$; $P_j = \mathcal{G}_{1,2}P_{j-1} + \mathcal{G}_{2,2}P_j$; $P_{j-1} = \tilde{M}$;
end
for $j = 1, \ldots, k-2$
  $\beta_j = \beta_j q(j,2)$;
end
$s = 1$;
for $j = k-1, \ldots, 1$
  $s = s \cdot q(j,2)$; $P_{j-1} = sP_{j-1}$;
end
$\tilde{X}_{k-1} = (c_{k-1}P_{k-1} - P_{k-2})/\alpha_{k-1}$;

for $j = k-2, \ldots, 1$
$\quad \tilde{X}_j = (c_j P_{k-1} - P_{j-1} - \beta_j \tilde{X}_{j+1})/\alpha_j;$
end
$\tilde{P} = \tilde{X}_1;$
for $j = 1, \ldots, k-2$
$\quad \tilde{P} = \tilde{P} + \bar{c}_j \tilde{X}_{j+1};$
end
$\tilde{P} = \tilde{O} + \bar{c}_{k-1} P_k;\ \tilde{P} = P_{k-1} \backslash \tilde{P};\ \eta = \mathrm{tr}(\tilde{P});$

2.2.2 Computing the Newton correction without linearising

It is also possible to substitute $A = P(x)$ into (2.4). This way, we obtain the following expression for the derivative $p'(x)$

$$p'(x) = \frac{d \det(P(x))}{dx} = \det(P(x)) \cdot \mathrm{tr}(P(x)^{-1} \cdot \frac{dP(x)}{dx}).$$

By evaluating numerically the matrix $P(x)^{-1}P'(x)$, this formula allows us to evaluate directly the Newton correction $p(x)/p'(x)$, which is the centrepiece for the EAI, without explicitly calculating $p(x)$:

$$\frac{p(x)}{p'(x)} = \frac{1}{\mathrm{tr}(P(x)^{-1}P'(x))}. \tag{2.6}$$

The following function performs an evaluation of $P(x)$ and $P'(x)$ by means of Horner's method [56, 81], followed by a numerical matrix inversion (see [56] for various considerations on how to perform, or avoid performing, matrix inversions).

function horner_trace
Input: Coefficients P_j**,** $j = 0, \ldots, k$**; initial approximation** x
Output: t**, trace of** $P^{-1}(x) \cdot P'(x)$
$\quad px = P_k; pxx = zeros(n);$
$\quad$ for $i = k-1{:}\,-1{:}\,0$
$\quad\quad px = xpx + P_i;$
$\quad\quad pxx = xpxx + (i+1)P_{i+1};$
$\quad$ end
$aux = pxx/px;$
$t = trace(aux);$

The function **horner_trace** allows to compute the trace of $P(x)^{-1}P'(x)$ in $\mathcal{O}(kn^2 + n^3)$ operations.

2.2.3 Linearization method versus Horner's method

Both the approaches to the computation of Newton's corrections that we presented in the last subsections, that is LQ factorization of a linearization in block-Hessenberg/block-triangular form and Horner's method applied directly to the coefficients of the matrix polynomial, have a computational complexity of $\mathcal{O}(n^3 + kn^2)$ operations per trace computation.

The overall complexity of the EAI is therefore $\mathcal{O}(tn^3 + tkn^2)$, where t is the number of scalar iterations needed before convergence, or in other words the number of times that the algorithm needs to compute the Newton's correction as a trace. As we will discuss in Section 2.4, when an appropriate choice of the initial approximations is made then t grows linearly with the total number of eigenvalues. Therefore, the complexity of the algorithm is $\mathcal{O}(kn^4 + k^2n^3)$ operations. This is to be compared with the complexities of more traditional matrix methods, which typically are $\mathcal{O}(k^3n^3)$ operations or higher.

Since the cost of the algorithm we propose grows as n^4 but it is only quadratic in k, we then expect that the EAI is computationally advantageous in terms of efficiency when the matrix polynomial has high degree and small coefficients, so that k^2/n is large.

It is worth noticing that in the linearization approach the case of large n can still be treated by means of an Ehrlich-Aberth method in $\mathcal{O}(n^3k^3)$ operations. The basic observation is that the factor n^4 comes from the block structure of the linearization involved in the computation of the trace. A reduction of the cost can therefore be achieved by a different strategy where the linearization is initially converted into scalar triangular-Hessenberg form: say, $N(x) = Rx + H$ where R is scalar triangular and H is scalar Hessenberg. The task can virtually be performed by any extension of the fast structured methods for the Hessenberg reduction proposed in [20, 28]. These methods preserve the rank structure which can therefore be exploited also in the triangular-Hessenberg linearization. Once the matrices R and H have been determined then the computation of $\mathrm{tr}(N(x))$ can be performed by the following algorithm which has a cost of $\mathcal{O}(n^2k^2)$ operations:

1. Perform an RQ decomposition of the Hessenberg matrix $N(x)$, obtaining a unitary matrix Q represented as product of $\mathcal{O}(nk)$ Givens transformations (Schur decomposition) and a triangular matrix U.

2. Compute the last row of $N(x)^{-1}R$ by solving the linear system $w^T N(x) = e_{nk}^T$, where $e_{nk} = [0, \ldots, 0, 1]^T$, and then computing $w^T := w^T R$.

3. Recover the diagonal entries of $N(x)^{-1}R$ from the entries of w and the elements of the Schur decomposition of Q.

This alternative road leads to an algorithm of cost $\mathcal{O}(n^2k^2)$ operations per scalar iteration; thus, if the total number of needed scalar iteration t is $\mathcal{O}(nk)$, the overall complexity is cubic in nk, just as the customary methods for the solution of polynomial eigenvalue problems. A more efficient implementation exploiting the rank structures of the matrices involved is also possible in principle, so that the cost in k is subcubic. This is the subject of a future research project.

This flexibility towards a different method, more suitable in the case of $k^2 \lesssim n$, is the main advantage of using linearizations. On the other hand, linearization techniques, if not properly used, may lead to an undesired increasing of the eigenvalue condition numbers [39, 58, 60]. This means that, if the ratio n/k^2 is not high enough to completely discourage the Horner's method approach, working directly on the matrix polynomials may in some situations be useful in order to avoid any potential discomfort. Numerical experiments show that the Horner version of the EAI is able to achieve in many cases higher accuracy than traditional methods. See Section 2.6 for further details.

2.3 Stopping criteria

At the generic ith vector iteration it is crucial to decide whether the update of the jth component of the vector $y^{(i+1)}$ must be performed or the scalar iteration in that component ought to be halted.

In the following, we discuss and justify the choices of the stopping criterion that we made in both the implementation of the EAI.

2.3.1 Linearization version

It is worth observing that the LQ-based method pursued for the trace computation also provides an estimate on the backward error for the generalised eigenvalue problem. From a result in [55] it follows that if $y_j^{(i)}$ is not an eigenvalue of $C_f(x)$ then

$$\eta(y_j^{(i)}) = \left(\| \, (y_j^{(i)} E + F)^{-1} \, \|_2 \, (1 + |y_j^{(i)}|) \right)^{-1}$$

gives an appropriate measure of the backward error for the approximate eigenvalue $y_j^{(i)}$. Since for $y_j^{(i)} E + F = \tilde{L} \cdot Q$ we have that

$$\| \, (y_j^{(i)} E + F)^{-1} \, \|_2 = \| \, \tilde{L}^{-1} \, \|_2 \geq \| \, \hat{P}_{k-1}^{-1} \, \|_2 \geq (\sqrt{n})^{-1} \, \| \, \hat{P}_{k-1}^{-1} \, \|_\infty,$$

in our implementation we consider the quantity

$$\hat{\eta}(y_j^{(i)}) = \sqrt{n} / \left(\| \, \hat{P}_{k-1}^{-1} \, \|_\infty \, (1 + |y_j^{(i)}|) \right)$$

as an error measure. If $\hat{\eta}(y_j^{(i)})$ is smaller than a fixed tolerance then $y_j^{(i)}$ is taken as an approximate eigenvalue and the corresponding scalar iteration is stopped. Simultaneously, our algorithm also checks whether the ratio between the estimated Ehrlich-Aberth correction and the corresponding approximation, that is $|y_j^{(i+1)} - y_j^{(i)}| / |y_j^{(i)}|$, is smaller than a given tolerance. If the answer is positive, the corresponding scalar iteration is halted.

2.3.2 Horner version

Observe that, if ξ is a root of $p(x)$, that is $\det P(\xi) = 0$, then as the approximation x gets close to ξ the matrix $P(x)$ becomes ill-conditioned. This makes quite natural to stop the iterations if the reciprocal of the condition number $\nu(P(x))$ is less than a prescribed tolerance τ_1. This criterion makes sense if the eigenvalue that we want to approximate is semisimple. In the instance of a defective eigenvalue λ with Jordan chains of length at most m, in view of the results in [95], it is more convenient to stop the iterations if the reciprocal of $\nu(P(x))$ is less than τ_1^m. This latter condition is hard to implement since it is not easy to evaluate numerically the maximal length of the Jordan chains of a matrix polynomial. Using the former stopping criterion may lead to a premature halt of the algorithm in the case of defective eigenvalues.

As an alternative to the previous stopping condition, one can follow the same strategy used in the linearization implementation. Following [104], define

$\alpha(x) = \sum_{\ell=0}^{k} |x^\ell|$. If $y_j^{(i)}$ is not an eigenvalue of $P(x)$ then the quantity

$$\eta(y_j^{(i)}) = \left(\| \left(P(y_j^{(i)}) \right)^{-1} \|_2 \, (1 + \alpha(y_j^{(i)})) \right)^{-1}$$

measures the backward error for the approximation $y_j^{(i)}$, and can be cheaply evaluated during the EAI. The iteration can be halted when $\eta(y_j^{(i)})$ is smaller than a given tolerance.

Numerical experiments showed that in the case of multiple eigenvalues the latter criterion lead to premature stops. For simple eigenvalues, no significant differences emerged. Therefore, our default choice was in favour of the criterion based on the condition number.

Similarly to what has been done in the implementation of the EAI via linearization, once again it is also convenient to add, with the "or" logic operator, the following condition:

$$|\mathcal{N}(y_j^{(i)})/(1 - \mathcal{N}(y_j^{(i)})\mathcal{A}_j(y^{(i)}, y^{(i+1)})| \leq \tau_2 |y_j^{(i)}|, \tag{2.7}$$

where τ_2 is a given tolerance. This condition says that the computed correction is too tiny and would not change the significant digits of the current approximation.

2.4 Initial approximations

In practice, the cost of each vector iteration of the algorithm is strongly dependent on the amount of early convergence occurring for a given problem. In other words, a critical point to assess the efficiency of the novel method is the evaluation of the total number t of calls of either the function **horner_trace** or the function **trace_linearization** (depending on which specific implementation of the EAI one is using), and of its dependence on the total number $N := nk$ of the eigenvalues. When the Ehrlich-Aberth method is used to approximate scalar polynomials roots, experiments show that t depends on the choice of the starting points.

2.4.1 Rouché theorem and starting points

As pointed out in [1, 8, 51], when there is not any *a priori* knowledge about the location of the roots, practically effective choices of initial approximations for the EAI are complex numbers equally displaced along circles. For instance, in [1] it is proposed to choose initial approximations displaced along a circle centred at the origin of sufficiently large radius so that it contains all the roots. In [51] the radius of the circle is suitably chosen. This strategy does not work effectively for polynomials having zeros with very large and with very small moduli. In [8] this drawback is overcome by considering different circles centred at the origin of suitable radii. The computation of these radii relies on the Rouché theorem.

Here we try to extend this technique to the case of matrix polynomials. We recall that, according to the Rouché theorem, if $s(x)$ and $q(x)$ are two polynomials such that

$$|s(x)| > |q(x)|, \quad \text{for} \quad |x| = r,$$

then $s(x)$ and $s(x)+q(x)$ have the same number of roots in the open disk $\{z \in \mathbb{C}: \quad |x| < r\}$. Applying this property with $s(x) = x^m$ and $q(x) = p(x) - s(x)$, for $0 \leq m \leq N$, implies that if $r^m > \sum_{j=0, j\neq m}^{N} |a_j| r^j$ then the polynomial $p(x)$ has m roots in the open disk of centre 0 and radius r. This property is at the basis of the criterion described in [8], based on the Newton polygon construction, for choosing initial approximations equidistributed along different circles centred in 0.

In order to extend this criterion to the case of polynomial eigenvalue problems we need a generalization of the Rouché theorem to matrix polynomials. We report the following result of [89] which we rephrase in a simpler way better suited for our problem.

Theorem 2.1. *Let $S(x)$ and $Q(x)$ be matrix polynomials and let r be a positive real. If $S(x)^H S(x) - Q(x)^H Q(x)$ is positive definite for $|x| = r$, then the polynomials* $\det S(x)$ *and* $\det(S(x) + Q(x))$ *have the same number of roots of modulus less than r.*

The following result is an immediate consequence of the above theorem applied to the polynomial $P(x)$ of (2.1) with $S(x) = x^m P_m$ and $Q(x) = \sum_{i=0,\, i\neq m}^{k} x^i P_i$.

Corollary 2.1. *Assume that*

$$P_m^H P_m r^{2m} - \Big(\sum_{j=0, j\neq m}^{k} P_j^H (x^*)^j\Big)\Big(\sum_{j=0, j\neq m}^{k} P_j x^j\Big) \succ 0, \quad \textit{for} \quad |x| = r, \tag{2.8}$$

where $A \succ B$ means that $A - B$ is positive definite. Then the matrix polynomial $P(x)$ has mk eigenvalues in the open disk of centre 0 *and radius r.*

Observe that if $\det P_m = 0$ then condition (2.8) cannot be verified. In fact, the vector v such that $P_m v = 0$ would be such that

$$v^H \Big(\sum_{j=0, j\neq m}^{k} P_j^H \bar{x}^j\Big)\Big(\sum_{j=0, j\neq m}^{k} P_j x^j\Big) v \leq 0$$

which is absurd.

In particular, if $\det P_k \neq 0$ the above corollary, applied with $m = k$, implies that all the eigenvalues of $P(z)$ are included in the disk of centre 0 and radius r provided that

$$r^{2k} P_k^H P_k - \Big(\sum_{j=0}^{k-1} P_j^H \bar{x}^j\Big)\Big(\sum_{j=0}^{k-1} P_j x^j\Big) \succ 0, \quad \text{for } |x| = r. \tag{2.9}$$

Observe that the latter condition is implied by

$$r^{2k} P_k^H P_k \succ \sum_{j=0}^{k-1} r^j P_j^H P_j + I \sum_{j>i} \rho(|P_j|^H |P_i| + |P_i|^H |P_j|). \tag{2.10}$$

Similarly, applying Corollary 2.1 with $m = 0$ provides a disk where $P(x)$ has no eigenvalues.

As an example of application, consider the 5×5 quadratic matrix polynomial $P(x) = Ax^2 + Bx + A^T$, where B is the tridiagonal matrix defined by the entries $[1, 2, 1]$, and A is the matrix with diagonal entries $100, 1, 1/1000, 1/100000$, superdiagonal entries equal to 1 and with zero entries elsewhere. The eigenvalues of $P(x)$ have approximate moduli `2.0050e+05, 1.4969e+03, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 6.6805e-04, 4.9874e-06`. The criterion based on the above corollary in the form (2.8) yields the bound $\texttt{4.4e} - \texttt{6} < |x| < \texttt{2.24e5}$ which is quite good. Applying condition (2.10) yields the bounds $\texttt{1.96e} - \texttt{6} < |x| < \texttt{5.1e5}$ which is still good.

Similar results have been obtained in [101] in the framework of tropical algebras; an ongoing future research project is to extend those results [13].

Another possibility is to choose the radii of the starting circles by applying the prescriptions of [8] to the scalar polynomial whose coefficients are the norms of the matrix coefficients of $P(x)$. This option is cheaper, leads to similar results, and is the default choice in our implementations.

Other options that we considered and that will be analysed in deeper details in Section 2.6, are:

- to pick points lying on only one circle, for instance the circle $|x| = 1$;
- to pick points lying on a small number of circles, chosen according to some criterion;
- to start with randomly chosen points, or more precisely to start with points whose modulus is generated according to some prescribed probability distribution (e.g. a log-normal distribution).

2.4.2 A posteriori error bounds

In the case of a scalar polynomial $p(x)$ of degree N, given a set of approximations $x_1, \ldots, x_N$ to the roots of $p(x)$ it is possible to prove that [102] the set of disks $D_i = D(x_i, r_i)$ of centre x_i and radius $r_i = n \left| \dfrac{p(x_i)}{(p_N \prod_{j=1,\, j\neq i}^{N} (x_i - x_j))} \right|$ is such that

1. the union of the disks contains all the roots of $p(x)$;
2. each connected component formed by the union of, say, c overlapping disks, contains c roots of $p(x)$.

The set formed by D_i, $i = 1, \ldots, N$ with the above properties is called *set of inclusion disks*.

In the case of a matrix polynomial $P(x)$ where $\det P_k \neq 0$, it is quite cheap to compute a set of inclusion disks. In fact, if $P(x) = \Pi LU$ is the PLU factorization of $P(x)$, then $|p(x)| = |\det P(x)| = \prod_{j=1}^{n} |U_{jj}|$. Moreover, the leading coefficient p_N of $\det P(x)$ coincides with $\det P_k$ which can be computed once for all. Observe that the LAPACK routine `zgesv` which solves a linear system with the matrix $P(x)$, is used to compute the Newton correction $1/\mathrm{trace}(P(x)^{-1}P'(x))$ during the EAI. Such routine, applied with $x = x_i$, provides at a negligible cost also the radius r_i.

The availability of a set of inclusion disks enables us to perform a cluster analysis. In fact, once an isolated disk has been detected, we have isolated

a single eigenvalue of the matrix polynomial $P(x)$. Once we have detected a set of c overlapping disks isolated from the remaining inclusion disks, we have detected a cluster formed by c eigenvalues of $P(x)$.

A different *a posteriori* error bound can be obtained by using a classical result (see [54]). The disk of centre x_i and radius $\hat{r}_i = n|p(x_i)/p'(x_i)|$ contains a root of the polynomial $p(x)$. However, the set of disks obtained in this way does not fulfill properties 1 and 2 of the set of inclusion disks. It is worth pointing out that the computation of $\hat{r}_i$ is inexpensive since the Newton correction $p(x_i)/p'(x_i)$ is computed by the EAI. Moreover, this *a posteriori* error bound still holds if the leading coefficient P_k is singular.

2.5 Multiple eigenvalues

Computational difficulties may be encountered in the case of multiple eigenvalues. In fact, the rate of convergence for multiple eigenvalues is linear, with respect to the cubic behaviour for simple eigenvalues. Moreover, for defective eigenvalues the standard stop condition may lead to a premature halt. For this reason, if it is possible to detect *a priori* multiple eigenvalues, it is advisable to deflate them; if it is not feasible to spot all of them theoretically, even lower bounds on the multiplicity are very helpful. If multiple eigenvalues are not predicted theoretically, one must rely on the cluster analysis to identify them and modify accordingly the stopping criterion.

A common situation that leads to multiple eigenvalues is met when the extremal coefficients are singular matrices. In this case, 0 and/or ∞ have algebraic multiplicity greater than or equal to 1. This situation can be circumvented to a certain extent.

In the instance of m eigenvalues at infinity, one may just start with an approximation vector y of length $nk - m$, acknowledging that the determinant $p(x)$ has in fact degree $nk - m$; if there are m zero eigenvalues it is possible to set to zero m components of the vector $y^{(0)}$ avoiding to update them.

The number of null singular values of P_0 provides a lower bound to the number of null eigenvalues of $P(x)$. Similarly, the number of zero singular values of P_k provides a bound to the number of eigenvalues at infinity. This way, the precomputation of the SVD [47] of P_0 and P_k may increase the performance of the EAI. Equivalently, one may perform any rank-revealing factorization (e.g., QR) instead of the SVD. Sometimes, the structure of the coefficients allows to achieve better bounds (this happens when the presence of a Jordan chain is evident without the need to perform any computation: e.g., if the same rows/columns of many consecutive coefficients, either at the beginning or at the end of the sequence $P_0, \ldots, P_k$, are zero).

In our implementation, the ranks of the two extremal coefficients are evaluated. If $\mathrm{rank}(P_0) < n$ (resp., $\mathrm{rank}(P_k) < n$), it is also checked if P_0 and P_1 (resp., P_k and P_{k-1}) share any common zero row/column. Of course, this kind of argument is easily extended in order to discover if there are consecutive coefficients, at either extremity of the matrix polynomial, sharing the same zero rows/columns. Thus, any manifest presence of zero and infinite eigenvalues is exploited, forcing immediate deflation of all the guaranteed roots. Moreover, if the test detects the presence of eigenvalues at 0 (resp., ∞), in order to avoid a premature stop for other undetected eigenvalues at 0 (resp., ∞), if any, the

stopping criterion is made stricter. The stronger stop condition requires that, for eigenvalues smaller (resp., larger) than a given bound, either the relative correction criterion (2.7) is satisfied with tolerance τ_2 or the relative correction criterion (2.7) is satisfied with tolerance $\tau_2^{1/2}$ and, simultaneously, the reciprocal condition number criterion is satisfied with tolerance τ_1. This heuristic device worked very effectively in our experiments, leading to satisfying results also in problems with multiple eigenvalues at either zero or infinity. See Section 2.6.

If the leading coefficient P_k is singular and if the degree of $p(x) = \det P(x)$ is not available together with the leading coefficients of $p(x)$, then it is not possible to generate a set of inclusion disks and to perform a cluster analysis. However, in this case we may apply an effective technique based on a rational transformation of the variable x. For instance, the variable x is replaced by the Möbius function $x = x(z) = (\alpha z + \beta)/(\gamma z + \delta)$ such that $\alpha\delta - \gamma\beta \neq 0$, and the polynomial $P(x)$ is replaced by the polynomial $Q(z) = (\gamma z + \delta)^k P(x(z))$. This way the infinite eigenvalues of $P(x)$ are mapped to the eigenvalues of $Q(z)$ equal to $-\delta/\gamma$. Moreover, $Q(z)$ has no eigenvalues at infinity provided that α/γ is not eigenvalue of $P(x)$. The substitution of variable can be performed implicitly without actually computing the coefficients of $Q(z)$ except for Q_k. We refer the reader to Chapter 3, Chapter 4, and to [12, 93] for more details on this subject.

2.6 Numerical experiments

A vast literature [9, 81, 96] covers the subject of the application of the EAI to scalar polynomials. To our knowledge, there has not been a previous application of the method to PEPs. Therefore, we have conducted a large number of numerical experiments in order to verify the goodness of the extension of the algorithm to the matrix case.

2.6.1 Efficiency: starting points and number of scalar iterations

In the scalar case, it is theoretically predicted [62] and heuristically verified [9] that, when one chooses starting points equally spaced on a circle containing all the eigenvalues, the path followed by the components of the approximation vector are quite regular. As the following figures show, this is also the case when the EAI is applied to polynomial eigenproblems: the left figure refers to a classical eigenvalue problems of a 50×50 matrix whose eigenvalues all have modulus $1/2$ and the starting points lie on the unit circle; the right figure refers to a random polynomial eigenvalue problem of degree $k = 3$ and dimension $n = 3$, with starting points picked on the circle of radius 4.

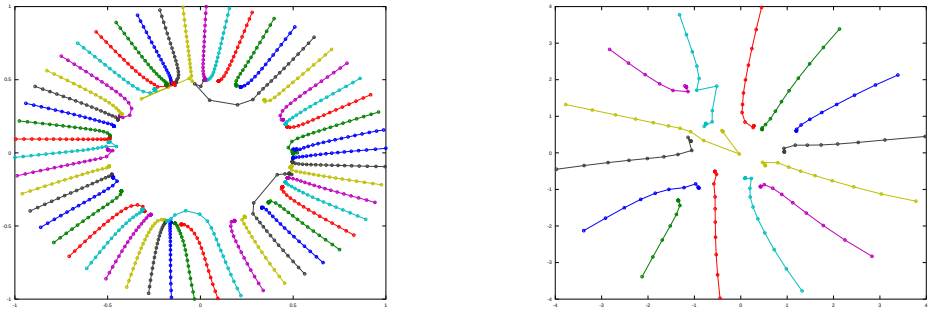

Fig. 1. *History of convergence for a classical eigenproblem (left) and a PEP of degree* 3 *(right)*

The disadvantage of this choice of starting points is the lack of a large enough amount of scalar iterations that are prematurely stopped due to early convergence (for the sake of brevity, in the following we will refer to this phenomenon using the word *deflation*). This drawback leads to a lower efficiency of the algorithm.

In fact, the complexity of the proposed algorithm is of order $tn^3 + tkn^2$, where t is the total number of times that a trace computation is needed before (vectorial) convergence. There is empirical evidence that t heavily depends on the choice of the initial approximation. For the case of scalar polynomials, the use of suitable strategies based on the Rouché theorem and the Newton polygonal (see Section 2.4, [8, 9, 13, 101]) leads to a linear dependence of t with respect to the total number of roots. We have tested several possible strategies in order to achieve the same result in the matrix case: namely, picking all roots on the unit circle, adopting a strategy based on the Newton polygonal as described in Subsection 2.4.1, choosing starting points whose moduli are distributed according to a given probability distribution, and picking the roots on several circles (choosing the number and the radii of such circles according to a "step function" criterion based on the ratio k/n and the product nk; see also [42] and Section 4.6, where this technique is explained in detail for the special case of T-palindromic polynomials).

Figure 2 shows the dependence of t on nk for a suit test of random matrix polynomial with $n = 2$ and variable k, while in Figure 3 the case of small degree ($k = 2$) and variable n is analysed.

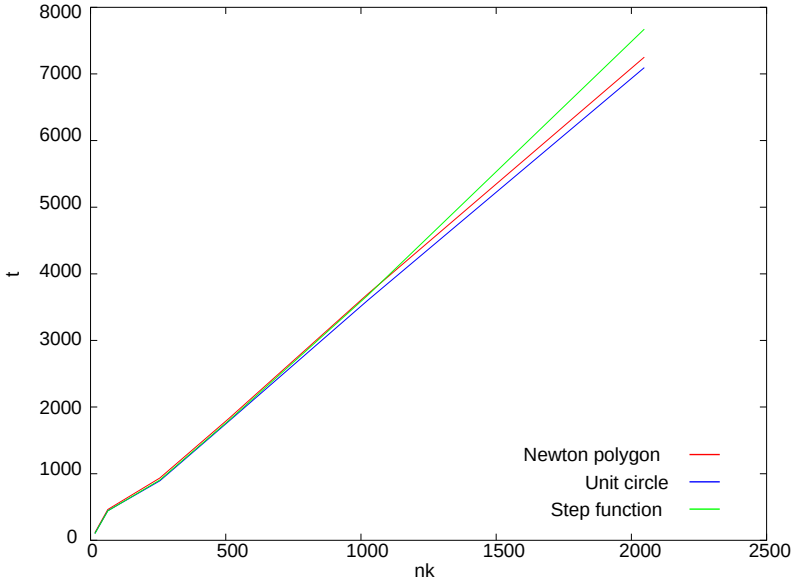

Fig. 2. *Dependence of t on k for $n = 2$*

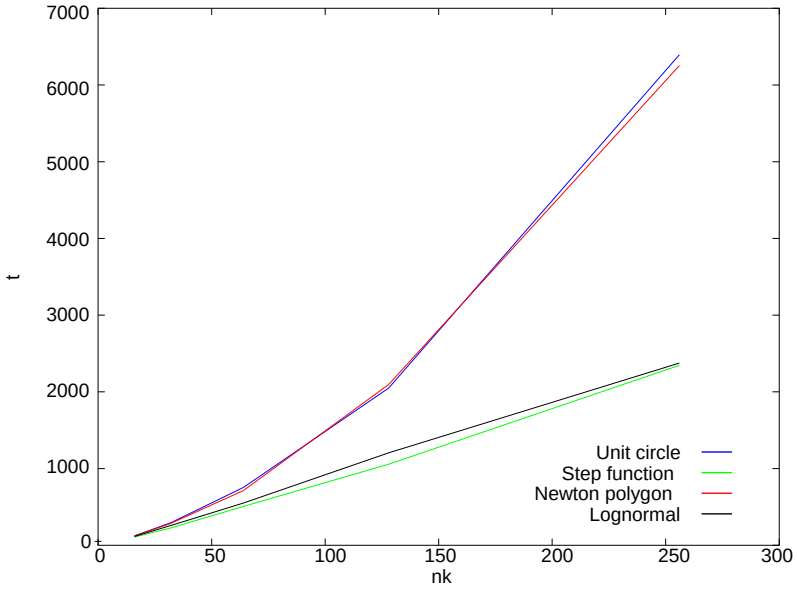

Fig. 3. *Dependence of t on n for $k = 2$*

For the high-degree case, which is the most suitable for being treated with the EAI, the growth appears to be slightly sublinear for all the tested methods (a logarithmic fit gave a growth $t \simeq \text{cost}.k^{0.8}$ in the analysed range of k). For the high-dimension case, the growth for the unit circle method and the newton polygonal method is superlinear (from logarithmic fit, the growth was approximately $t \simeq \text{cost}.n^{1.4}$ for those two choices of starting points); the growth is slightly superlinear for the step function method and the probability distribution method, which we implemented with a lognormal distribution ($t \simeq \text{cost}.n^{1.1}$).

In contrast with the randomly generated tests, the Newton polygon technique is always helpful when there is a remarkable unbalancing of the norms of the matrix coefficients. When n is large and the coefficients are unbalanced, the best results are obtained by combining the Newton polygon technique and an ansatz (e.g. the step function criterion) that picks the starting points on more than one circle whose radii are distributed near the ones computed by the Newton polygon algorithm. Another possibility, leading to results that are similar to those of the Newton polygon, is to exploit results recently obtained with tropical algebra [101]. Ongoing research is being done to investigate the possibility to extend such results [13].

The experimental results are satisfying for high degree polynomials, but not completely for high dimension polynomials. Further work is needed to improve the choice of the starting points in the case of a large ratio n/k^2. We mention, however, that when the polynomial is structured numerical experiments suggest that enough deflations happen, so that a linear growth of t is achieved in this case also in the large n region. See Section 4.6.

We complete this subsection giving evidence for the claim we made that our experimental experience shows that the EAI has superior efficiency with respect to the sequential application of Newton method to single roots of $p(x)$ with implicit deflation of the previously approximated eigenvalues.

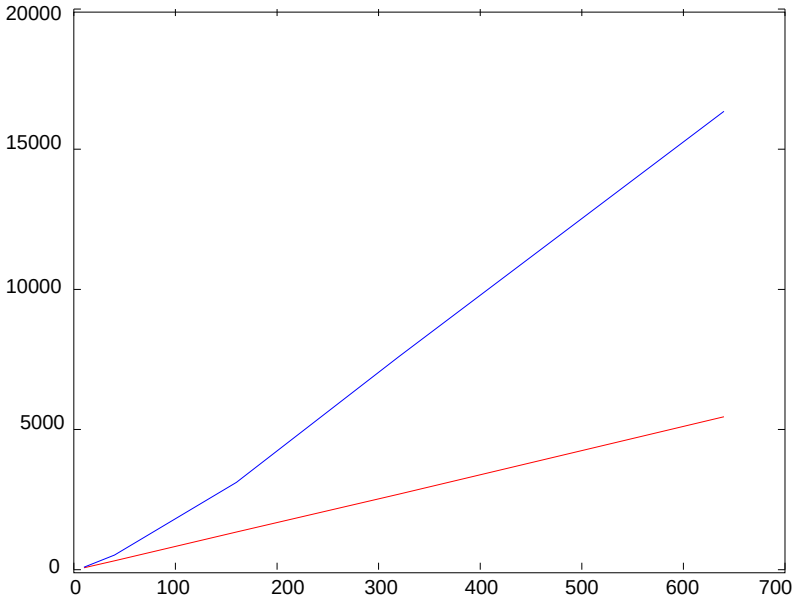

Fig. 4. *Comparison between the EAI (red) and the method in [37, 70] (blue), for* $n = 5$

Figure 4 compares the number of trace computation needed by the EAI (shown in red) and the sequential Newton method with implicit deflations. Random matrix polynomials with $n = 5$ and growing degree were formed. The starting points were chosen with the prescriptions of [37] for the latter method, and for the EAI on the unit circle. Similar results were obtained for the case of large n: we conclude that, in general, the sequential Newton method of [37, 70] is slower than the EAI.

2.6.2 Efficiency: execution time

Experiments we made with our implementation suggest that $t = \mathcal{O}(nk)$ also when the EAI is applied to a matrix polynomial. In particular, this means that the computational complexity of the EAI is $\mathcal{O}(kn^4 + k^2n^3)$, leading to great computational advantages for $k \gg \sqrt{n}$. As noticed in Subsection 2.2.3, if on the contrary $k \lesssim \sqrt{n}$ different implementation of the EAI are possible, with cubic efficiency in kn.

In order to confirm such predictions, we have compared our implementation of the EAI (Horner version) and Matlab's QZ implementation `polyeig` on random matrix polynomials of high degree and small dimension. For very large values of k, we did not actually run `polyeig` due to the very large forecast computation times, but we extrapolated the times from the other experiments; in fact, when doubling the value of k we can expect that the running time of the QZ algorithm grows approximately with a factor 8. Such extrapolated values are marked with a * in the following tables.

Computation times for $n = 2$		
k	EAI	polyeig
50	0.018 s	0.015 s
100	0.044 s	0.064 s
200	0.111 s	0.369 s
400	0.360 s	4.35 s
800	1.29 s	51.9 s
1600	4.76 s	437 s
3200	18.4 s	$\mathcal{O}(50 \text{ min})^*$

Computation times for $n = 5$		
k	EAI	polyeig
20	0.062 s	0.010 s
40	0.121 s	0.057 s
80	0.312 s	0.370 s
160	0.920 s	4.39 s
320	2.92 s	44.0 s
640	10.3 s	398 s
1280	38.1 s	$\mathcal{O}(50 \text{ min})^*$
2560	148 s	$\mathcal{O}(7 \text{ hours})^*$
5120	575 s	$\mathcal{O}(2 \text{ days})^*$

The values in the tables above are in agreement with our prediction that the computation time should asymptotically grow as k^2. Moreover, the experimentation also confirms that, for a given value of n, the ratio of the time needed by EAI with respect to the time needed by the QZ algorithm exhibits an asymptotic growth that is approximately linear in k. This effect is taken to the extreme in the case $n = 5$, $k = 5120$. Had we used Matlab's `polyeig`, it would have taken several days of computation time on our machine to solve such a problem. Our implementation of the EAI gave the approximated eigenvalues in less than 10 minutes.

2.6.3 Accuracy: a comparison with the QZ algorithm

We will present in this section some experiments on the accuracy of the Horner version of the EAI.

In order to test the accuracy of our implementation we used the Matlab toolbox NLEVP[7]. This toolbox has been recently proposed by its authors as an interesting set of benchmark problems that may be used as a standard test for new methods for nonlinear eigenvalue problems. It contains data coming from practical applications as well as model problems known to have peculiar properties.

Amongst the many nonlinear eigenproblems contained in NLEVP, we have selected all the square PEPs with $n < 25k^2$. We discarded the problems with

a larger ratio n/k^2 because they could be better dealt with by a different implementation of the EAI, as was discussed in Subsection 2.2.3. The test suite selected with this criterion consists of 29 problems plus the 2 problems butterfly and wiresaw1 that, being structured, will be treated in Chapter 4.

In all the parameter-dependent problems in the library the default values of the parameters were selected. All methods were directly applied to the original matrices as saved in the library, without preprocessing them with any scaling[1]. Forward errors are evaluated by comparing the approximations with either theoretically known values, when available, or values computed in variable precision arithmetic (VPA) with Matlab's symbolic toolbox.

The graphs below are in logarithmic scale. Whenever the absolute error for a certain eigenvalue λ appeared to be numerically zero, i.e. it was less than λ times the machine epsilon $\epsilon = 2^{-52} \simeq \texttt{2.22e}-\texttt{16}$, we formally set it equal to $\frac{\lambda\epsilon}{2}$. Only absolute errors for the finite eigenvalues are shown in the figures.

For the 3 problems with $k \geq 3$, the eigenvalue forward errors where computed for both the EAI, in its Horner version, and the QZ method (as implemented in `polyeig`). Absolute errors for our implementation of the EAI are marked with a red $*$ symbol, while absolute errors for `polyeig` are marked with a blue $+$ symbol. For this set of experiments, we picked starting points on the unit circle. In our experience the order of magnitude of the forward error is not significantly effected by the choice of the starting points, even though for some problems other choices led to slight improvements (not discussed here).

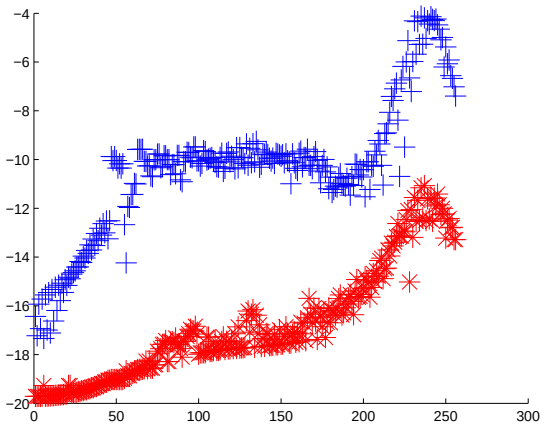

Fig. 5. *Forward absolute errors for the problem orr sommerfeld*

[1] Although we did not alter the coefficients given in input to the various methods, for most quadratic problems `quadeig` has performed a scaling because of its default settings, that prescribe to follow the technique of [29] under certain conditions; see [52].

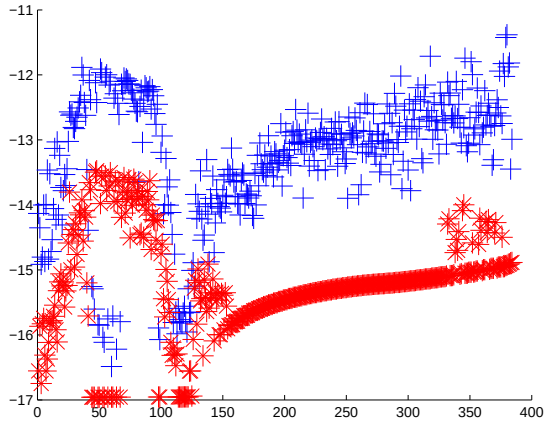

Fig. 6. *Forward absolute errors for the problem plasma drift*

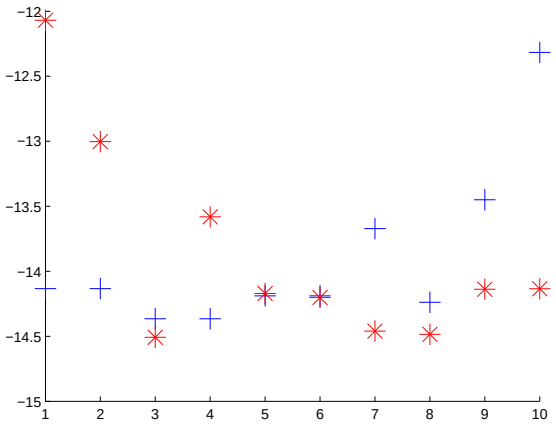

Fig. 7. *Forward absolute errors for the problem relative pose 5pt*

For the 26 problems with $k = 2$, three methods were compared by computing their forward errors with the same method as above: `polyeig` (blue + symbol), EAI (red $*$ symbol) and the software `quadeig` by S. Hammarling, C. Munro and F. Tisseur [52], specifically designed for quadratic PEPs (black x symbol).

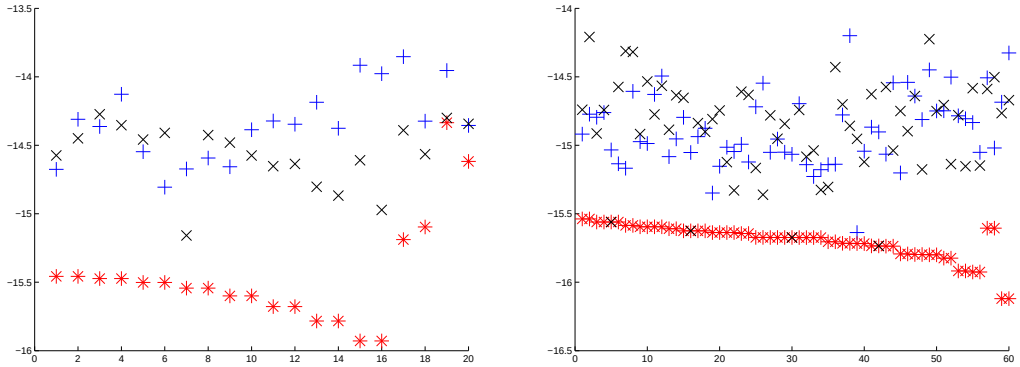

Fig. 8. *Forward absolute errors for the problems acoustic wave 1d (left) and acoustic wave 2d (right)*

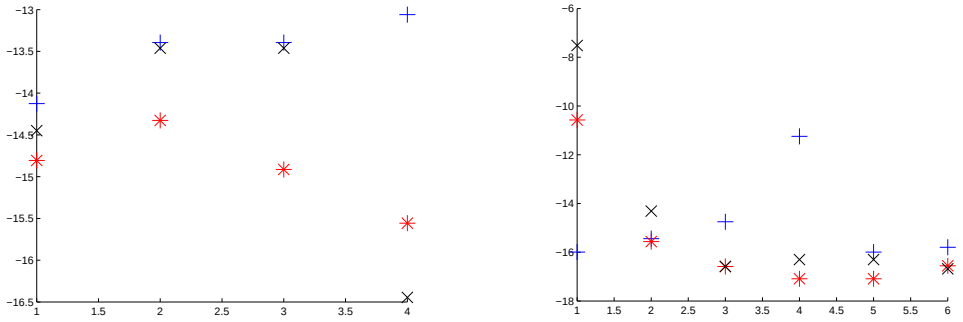

Fig. 9. *Forward absolute errors for the problems bicycle (left) and bilby (right)*

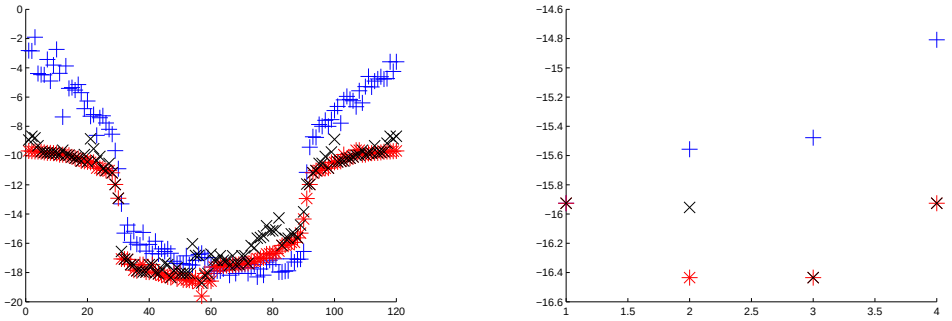

Fig. 10. *Forward absolute errors for the problems cd player (left) and closed loop (right)*

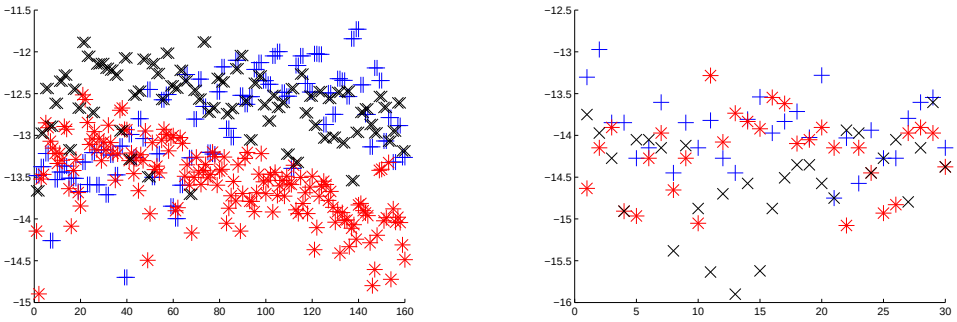

Fig. 11. *Forward absolute errors for the problems dirac (left) and gen hyper 2 (right)*

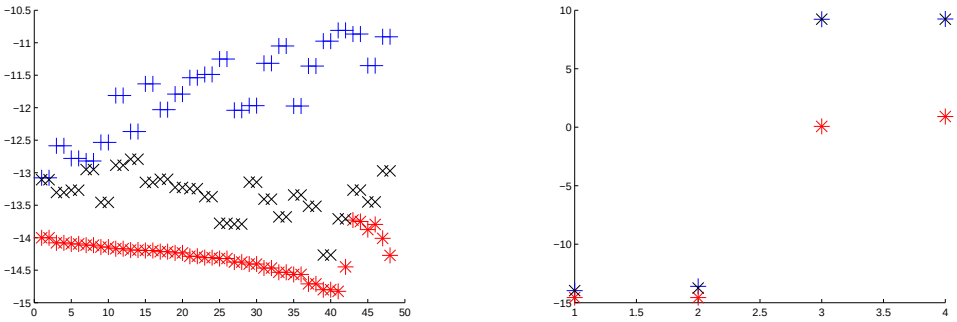

Fig. 12. *Forward absolute errors for the problems hospital (left) and intersection (right)*

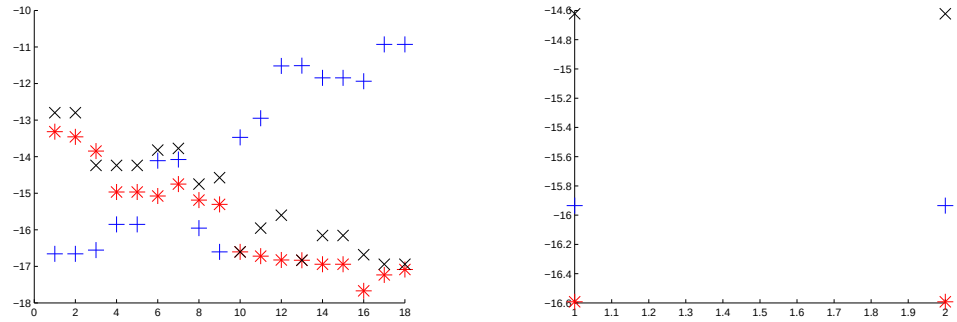

Fig. 13. *Forward absolute errors for the problems metal strip (left) and mobile manipulator (right)*

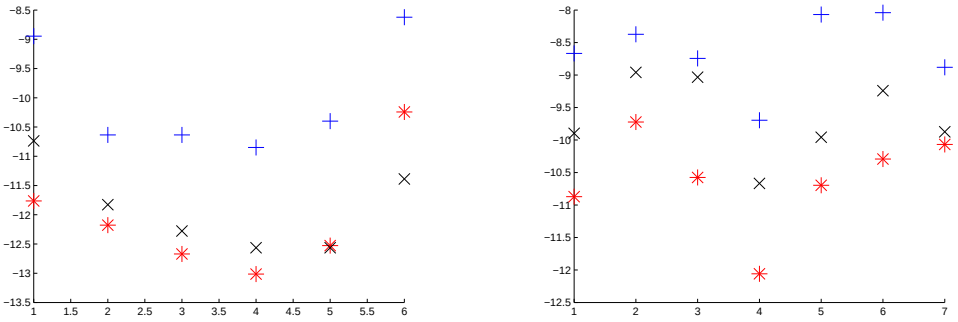

Fig. 14. *Forward absolute errors for the problems omnicam1 (left) and omnicam2 (right)*

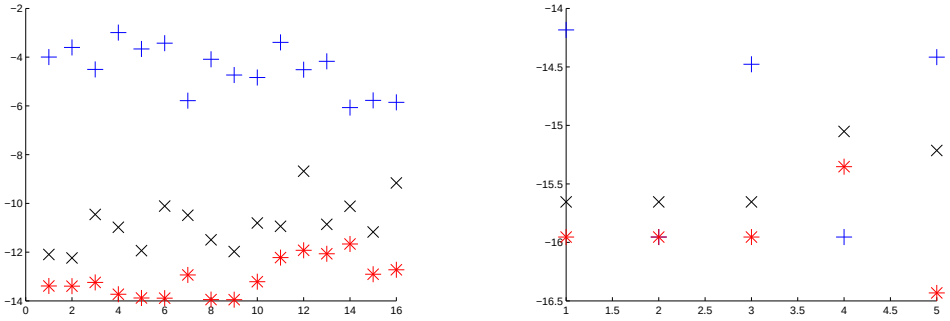

Fig. 15. *Forward absolute errors for the problems power plant (left) and qep1 (right)*

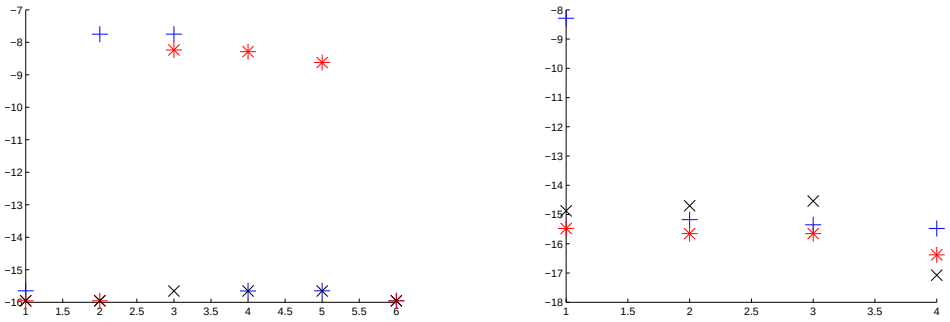

Fig. 16. *Forward absolute errors for the problems qep2 (left) and qep3 (right)*

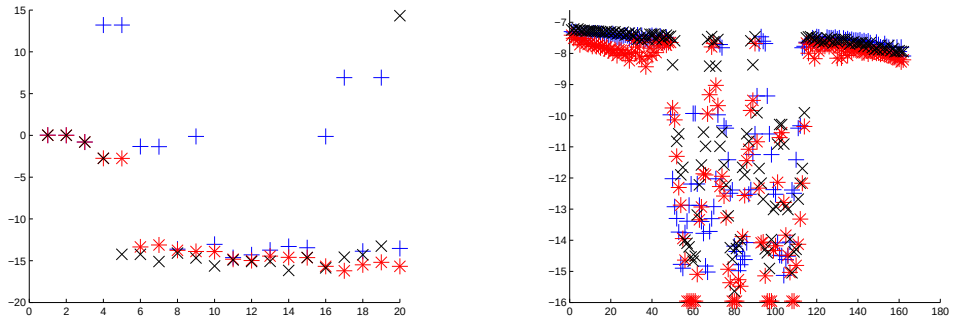

Fig. 17. *Forward absolute errors for the problems relative pose 6pt (left) and sign1 (right)*

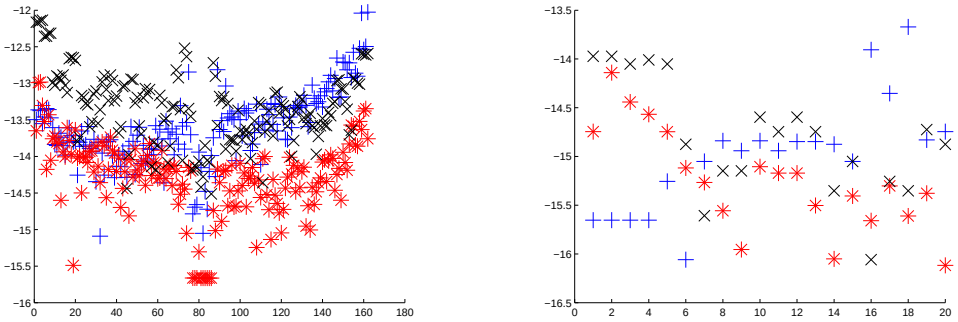

Fig. 18. *Forward absolute errors for the problems sign2 (left) and sleeper (right)*

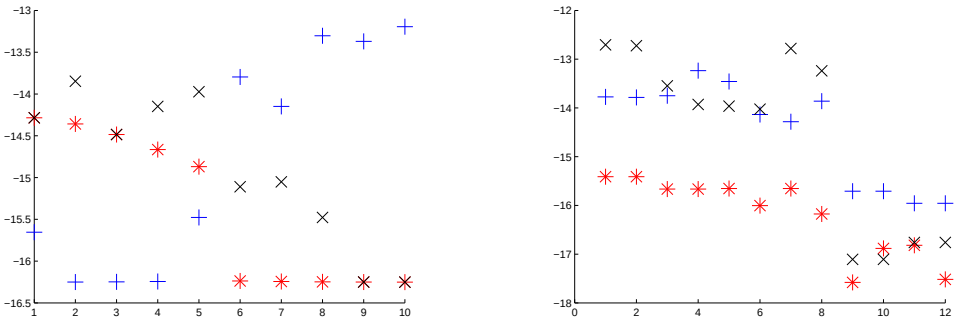

Fig. 19. *Forward absolute errors for the problems spring (left) and spring dashpot (right)*

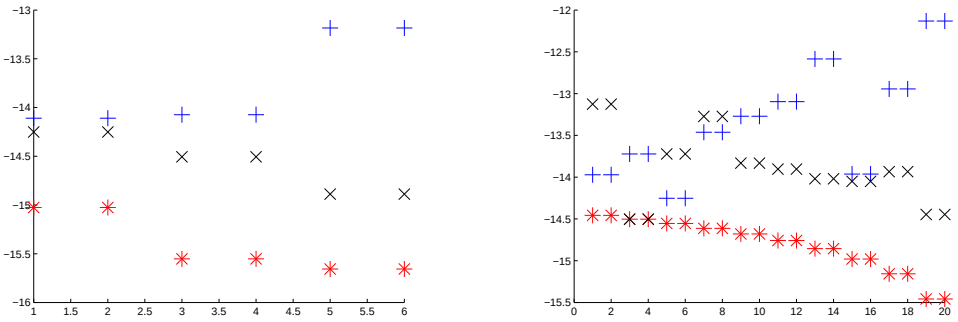

Fig. 20. *Forward absolute errors for the problems wing (left) and wiresaw2 (right)*

As can be seen by the figures above, the approximations of the EAI are competitive, and often more accurate than the approximations of the QZ. In some cases, the improvement is remarkable. We report in the following table the maximal relative error and the average relative error for all the finite (i.e. neither numerically zero nor numerically infinite) eigenvalues of the 29 considered problems, and for both the EAI and the QZ. The average relative error is defined as the geometric mean of all relative errors; numerically zero relative errors have been counted as relative errors equal to $\epsilon/2$. The values reported for the QZ for quadratic problems correspond to the algorithm, picked between `polyeig` and `quadeig`, that achieved the best performance in terms of average relative error for the given problem; as can be deduced by the above pictures and coherently with the results on backward errors presented in [52], such best performance was achieved generally, but not always, by the latter algorithm.

Problem	Rel. errors, EAI		Rel. errors, QZ	
	Max.	Avg.	Max.	Avg.
acoustic wave 1d	1.0e-14	2.1e-16	1.1e-14	1.7e-15
acoustic wave 2d	$\epsilon/2$	$\epsilon/2$	3.2e-15	7.4e-16
bicycle	1.0e-15	4.0e-16	7.6e-15	1.1e-15
bilby	2.4e-14	3.5e-16	5.1e-15	1.8e-15
cd player	5.3e-16	1.2e-16	4.0e-14	3.3e-16
closed loop	$\epsilon/2$	$\epsilon/2$	3.4e-16	1.5e-16
dirac	4.1e-14	5.9e-15	1.9e-13	2.9e-14
gen hyper 2	2.5e-14	9.3e-16	2.4e-15	4.9e-16
hospital	2.7e-15	1.6e-16	2.0e-14	1.4e-15
intersection	4.8 e-9	4.5 e-13	1.0	2.4e-8
metal strip	6.3e-16	1.7e-16	2.3e-15	6.7e-16
mobile manipulator	$\epsilon/2$	$\epsilon/2$	5.1e-16	5.1e-16
omnicam1	9.1e-11	1.2e-12	4.3e-9	6.4e-13
omnicam2	3.9e-10	2.3e-15	4.0e-9	1.2e-13
orr sommerfeld	5.0e-12	9.1e-16	4.8e-5	2.8e-9
plasma drift	3.4e-13	5.1e-16	1.3e-11	4.7e-14
power plant	8.3e-14	1.1e-15	6.1e-11	1.9e-13
qep1	8.9e-16	1.7e-16	1.8e-15	5.1e-16
qep2	5.8e-9	5.3e-11	2.2e-16	1.9e-16
qep3	3.9e-9	1.0e-14	8.0e-10	3.2e-14
relative pose 5pt	2.9e-14	6.8e-15	1.6e-14	6.3e-15
relative pose 6pt	7.5e-14	1.9e-14	1.2e-13	1.4e-14
sign1	3.8e-8	1.1e-10	5.0e-8	3.9e-10
sign2	4.5e-14	2.8e-15	3.1e-13	1.3e-14
sleeper	8.0e-16	2.8e-16	1.8e-15	5.6e-16
spring	$\epsilon/2$	$\epsilon/2$	1.7e-15	3.5e-16
spring dashpot	5.6e-15	3.1e-16	2.8e-13	1.4e-14
wing	$\epsilon/2$	$\epsilon/2$	1.2e-15	8.1e-16
wiresaw2	$\epsilon/2$	$\epsilon/2$	2.4e-15	9.2e-16

As the results above show, the EAI was generally able to improve the accuracy of the approximations with respect to the QZ method. The only problems where the EAI achieved an average performance worse than the QZ are qep2 (loss of accuracy on the multiple eigenvalue 1 with respect to `quadeig`; `polyeig` has problems as well) and gen hyper 2.

It is worth noticing that for some of the problems the detection of (at least some) zero or infinite multiple eigenvalues through a rank-revealing factorization, as described in Section 2.5, helped improving the performance of the algorithm. The test gave satisfactory results also for the problems in the NLEVP library that have defective eigenvalues at either zero or infinity.

The problems in the NLEVP database do not have high degree, so the condition $k^2 \gg n$ is not met. Therefore, in contrast with the high degree case, for those problems using the EAI as a primary algorithm does not bring advantages in term of computation time; on the contrary the Horner method-based implementation used for the experiments discussed in this section is slower than the QZ if $n \gg k^2$. Also, in these cases a linearization-based version of the EAI would be more efficient than the polynomial-based implementation used

for the experiments described in the present section. However, it is worth noticing that the numerical experiments showed that very often the EAI improved the accuracy achieved by `polyeig` and/or `quadeig`; in many cases, the EAI was able to compute correctly all the digits of the eigenvalue up to machine precision. This suggests that, when $k^2 \lesssim n$, it is possible to use the EAI as a refinement algorithm in order to improve the approximations obtained by the QZ. Using such values as starting points offers of course a very good choice, lowering the number of overall scalar iteration needed before convergence and therefore improving the efficiency of the EAI.

2.6.4 Eigenvectors

As we have just seen, the EAI computes the approximated eigenvalues of a PEP. In many applications, only the eigenvalues of a PEP are wanted. In such cases, the implementation of the EAI as described in the previous sections is all that is needed. However, we must consider that other applications exist where the eigenvectors are of interest as well. Let us analyse how they can be found when using the proposed method.

When given a matrix polynomial $P(x)$ as an input the EAI does not compute itself the eigenvectors; however, it gives as an output quite accurate approximations of the eigenvalues. Let us call such approximations y_i, $i = 1, \ldots, nk$. It is possible to give y_i as an input to other algorithms in order to find the eigenvectors in a second moment. We mention, amongst the various possibilities, the computation of the approximated null space via an SVD decomposition [47] or the inverse iteration method, that is, the power method applied to $P(y_i)^{-1}$ [47, 56].

As we have seen, the EAI computes approximations of the eigenvalues in $\mathcal{O}(n^4k + k^2n^3)$ floating point operations, or — via a linearization $L(x)$ and following the strategy described in Subsection 2.2.3— in $\mathcal{O}(n^3k^3)$. Given the evaluation of the matrix polynomial $P(x)$ at y_i (which is already computed by the EAI algorithm as a byproduct), both the inverse iteration method or the SVD decomposition method provide an approximation v_i of the corresponding eigenvector in $\mathcal{O}(n^3)$ floating point operations per eigenvalue; hence, we see that all the eigenvectors can be approximated in $\mathcal{O}(n^4k)$ floating point operations. When $k \gg \sqrt{n}$, such cost is negligible with respect to k^2n^3 and, therefore, the overall complexity of the algorithm is not affected by the computation of the eigenvalues. For what concerns the case $k \ll \sqrt{n}$, the most efficient version of the EAI computes as a byproduct a Hessenberg-triangular linearization $H(x)$ (dimension nk) evaluated at each eigenvalue. In this setting, with $H(y_i)$ as an input for $i = 1, \ldots, nk$, the cost of the computation of all the approximated eigenvectors with the inverse iteration can be reduced to $\mathcal{O}(n^3k^3)$, that is the same order of magnitude as the approximation of the eigenvalues.

Let us now give some experimental results on the quality of eigenvector approximation. The first comparison has been made on random polynomials. Eigenvalues and eigenvectors were computed with three different methods: `polyeig`, the EAI followed by the inverse power method, and the EAI followed by Matlab's function `null` that computes the null space of a numerically singular matrix using the SVD decomposition. Computed eigenvectors were normalised in order to have unit 2-norm. For each eigencouple (y_i, v_i) a residual was computed as $\|P(y_i)v_i\|_2/\|P(y_i)\|_2$. The results for the three methods

were compared. Figure 21 shows, in a logarithmic scale graph, the residuals for `polyeig` (blue + symbol), EAI followed by SVD decomposition (red $*$ symbol) and EAI followed by inverse iteration (green o symbol) for a random PEP with $n = 5, k = 40$.

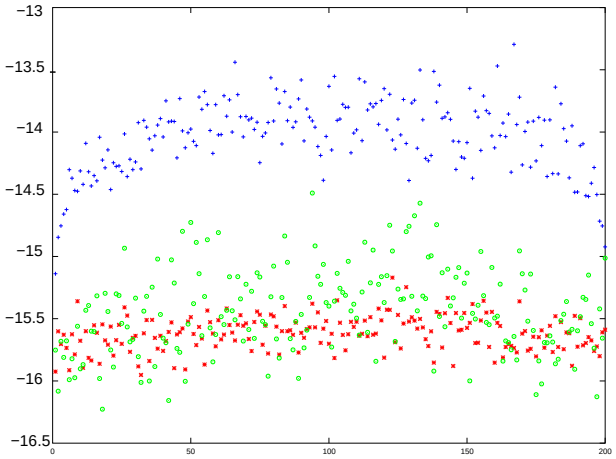

Fig. 21. *Residuals for eigenvectors in a randomly generated problem ($k = 40, n = 5$): polyeig (blue), EAI+SVD (red), and EAI+inverse iteration (green)*

As Figure 21 illustrates, the eigenvector residual on a typical randomly generated PEP is sensibly lower with respect to the one resulting from the QZ algorithm as implemented in `polyeig`, either using the inverse iteration or the SVD. From what concerns the comparison between the SVD and the inverse iteration, the results were comparable in terms of average residual, with a slightly better result for the SVD; in terms of maximum of the residuals the SVD typically worked better, as can be seen by Figure 21.

As a second test, the NLEVP [7] problems gen hyper2, qep1 and qep3 have been solved with the three methods mentioned before and with `quadeig` [52]. For these problems, the solutions are exactly known and provided within the Matlab package NLEVP [7]. If v_i is the approximated eigenvector and ξ_i the corresponding exact eigenvector, a measure of the forward error is the residual 2-norm $\|v_i - \xi_i\|_2$ (after having normalised both v_i and ξ_i in such a way that they have norm 1 and that their first nonzero component is a real positive number). The following table reports the maximal forward error and the geometric mean of forward errors for each problem and each method. Once again, values less then the machine precision ϵ were set equal to $\epsilon/2$.

Problem	EAI+SVD		EAI+inv. it.	
	Max.	Avg.	Max.	Avg.
gen hyper 2	2.5e-13	6.7e-15	2.5e-13	7.0e-15
qep1	3.1e-16	1.3e-16	2.4e-16	1.3e-16
qep3	4.1e-16	1.4e-16	2.3e-16	1.3e-16

Problem	quadeig		polyeig	
	Max.	Avg.	Max.	Avg.
gen hyper 2	2.9e-13	1.6e-14	2.9e-13	1.8e-14
qep1	$\epsilon/2$	$\epsilon/2$	8.6e-15	5.8e-16
qep3	7.2e-15	3.7e-16	1.8e-15	2.2e-16

These and other experiments suggest that eigenvector computation can be attached to the EAI with rather satisfactory outcomes.

2.7 Other simultaneous polynomial root-finders

We review in this section some possible alternatives to the EAI as root-finding methods that may be applied to the characteristic equation $p(x) = 0$.

The modified Ehrlich-Aberth iteration (MEAI) [81, 94] is a slight modification of the EAI that uses (in the Gauss-Seidel style) the formula

$$\begin{aligned} y_j^{(i+1)} &= y_j^{(i)} - \frac{\mathcal{N}(p(y_j^{(i)}))}{1 - \mathcal{N}(p(y_j^{(i)}))\mathcal{A}_j(y^{(i)}, y^{(i+1)})}, \\ \mathcal{A}_j(y^{(i)}, y^{(i+1)}) &= \sum_{\ell=1}^{j-1} \frac{1}{y_j^{(i)} - y_\ell^{(i+1)}} \mathcal{N}(p(y_\ell^{(i+1)})) + \sum_{\ell=j+1}^{N} \frac{1}{y_j^{(i)} - y_\ell^{(i)}} \mathcal{N}(p(y_\ell^{(i)})). \end{aligned} \tag{2.11}$$

Of course an analogous formula can be written for the Jacobi style implementation of the MEAI.

Various authors showed (e.g. [81, 96]) that its order of convergence is 4, which is higher than the usual EAI. In practice, the order of convergence seems not to be as important as the speed in reaching the region of fast convergence. In numerical experiments on efficiency, we did not see a significant improvement on the total number of scalar iterations when using the MEAI with respect to the EAI. However, in some test problems the accuracy was improved. Further work is needed in the future to understand if there exists any class of problems, possibly with specific properties, for which the MEAI achieves better performances.

Another possibility is to design simultaneous root-finding algorithms based on higher order methods in the Householder class. Consider for instance, instead of the Newton method $x \to x - f(x)/f'(x)$, the next one in the Householder family, which is the Halley method: $x \to \dfrac{2f(x)f'(x)}{2(f'(x))^2 - f(x)f''(x)}$. It is possible to build a Halley-like simultaneous root-finder [96]. In order to apply it to PEPs, not only the Newton correction $p(x)/p'(x)$, but also a factor $p(x)/p''(x)$, is needed. However, also the latter can be computed as a trace.

Notice in fact that

$$\frac{d\mathrm{tr}\left(P(x)^{-1}P'(x)\right)}{dx} = \mathrm{tr}\left(P(x)^{-1}P''(x)\right) + \mathrm{tr}\left(P'(x)\frac{dP(x)^{-1}}{dx}\right).$$

This gives

$$\frac{p''(x)}{p(x)} - \left(\frac{p'(x)}{p(x)}\right)^2 = \mathrm{tr}\left(P(x)^{-1}P''(x)\right) - \mathrm{tr}\left(P'(x)P(x)^{-1}P'(x)P(x)^{-1}\right),$$

where we have used the identity

$$0 = \frac{dP(x)P(x)^{-1}}{dx} = P(x)\frac{dP(x)^{-1}}{dx} + P'(x)P(x)^{-1};$$

the above formula easily leads to an expression for the Halley correction in terms of easily computable traces. Also in this case, the investigation on the actual comparison of the Halley-like iteration with respect to the EAI is ongoing.

We finally mention the Durand-Kerner simultaneous iterative method [81, 96], here expressed in Jacobi style (writing its Gauss-Seidel version is immediate):

$$y_j^{(i+1)} = y_j^{(i)} - \frac{p(y_j^{(i)})}{\prod_{\ell \neq j}(y_j^{(i)} - y_\ell^{(i)})}.$$

For the Durand-Kerner method, preliminary experiments showed worse performance with respect to the EAI; this is the reason why we mainly focused on the latter. However, there may be special cases where the Durand-Kerner method achieves good results, and further investigation is needed.

Chapter 3

The behaviour of complete eigenstructures under a rational change of variable

The present chapter deals with the behaviour of the complete eigenstructure of a matrix polynomial when a rational change of variable comes into play.

It is based on the paper [93], that was motivated by the will to generalise the partial results derived in [42], where we considered the particular case of a square and regular polynomial matrix with entries in $\mathbb{C}[x]$ and without infinite elementary divisors, and the Dickson change of variable $x(y) = \frac{y^2+1}{y}$. Moreover, [93] was also inspired by the possibility to extend the results from D. S. Mackey and N. Mackey [80], who described the special case of rational transformations of degree 1, also known as Möbius transformations.

The results exposed in this chapter are rather algebraic. Therefore, the polynomial matrix point of view is preferred to the matrix polynomial point of view.

3.1 Introduction to the change of variable problem

The aim of this chapter is to investigate the link between the complete eigenstructures of two polynomial matrices $P(x)$ and $Q(y)$ related one to another by a rational transformation $x(y)$ of the variable. We will give results that hold for matrix polynomials whose coefficients lay in $\mathbb{F}^{m\times p}$, where $\mathbb{F}$ is a generic field (not necessarily $\mathbb{C}$). For the sake of simplicity, the results are more easily expressed if $\mathbb{F}$ is algebraically closed; we stress however that this is not a necessary assumption. In point of fact, we will assume in the following that the underlying field is algebraically closed, and then we will explain in Section 3.6 how it is possible to extend our arguments to the case of a generic field.

The problem of characterising the behaviour of the complete eigenstructure of a polynomial matrix when a change of variable occurs has an application in the development of a structured version of the EAI to polynomial eigenvalue

problems whose spectrum is known to have a symmetry such that the eigenvalues appear in pairs $\{\lambda, f(\lambda)\}$ where $f(f(\lambda)) \equiv \lambda$.

In order to better explain the question we are interested in, let us consider the following example where $\mathbb{F} = \mathbb{C}$.

Example 3.1. *Suppose that we have to deal with the polynomial matrix*

$$P(x) = \begin{bmatrix} x^2 - 20x & 0 & 0 \\ x - 20 & x^2 - 20x & 0 \\ 0 & 0 & x \\ 0 & 0 & x^2 \\ 0 & 0 & 0 \end{bmatrix};$$

if we choose grade$(P(x))$= 2, then the complete eigenstructure of $P(x)$ is the following:

- *the elementary divisors of $P(x)$ are $(x-20)$, x, $(x-20)$, x^2;*
- *there are no right minimal indices;*
- *the left minimal indices of $P(x)$ are $0,\ 1$.*

The rational change of variable $x(y) = \frac{16y^2-25}{y^2-y}$ induces a mapping Φ_2 such that $\Phi_2(P(x)) = (y^2-y)^2 P(\frac{16y^2-25}{y^2-y}) =: Q(y)$, with grade$(Q(y)) = 4$ and

$$Q(y) = \begin{bmatrix} (25-16y^2)(2y-5)^2 & 0 & 0 \\ (y-y^2)(2y-5)^2 & (25-16y^2)(2y-5)^2 & 0 \\ 0 & 0 & (y^2-y)(16y^2-25) \\ 0 & 0 & (16y^2-25)^2 \\ 0 & 0 & 0 \end{bmatrix}.$$

By studying the complete eigenstructure of $Q(y)$ we find out that

- *the elementary divisors of $Q(y)$ are $(y-\frac{5}{2})^2$, $(y-\frac{5}{4})$, $(y+\frac{5}{4})$, $(y-\frac{5}{2})^2$, $(y-\frac{5}{4})^2$, $(y+\frac{5}{4})^2$;*
- *there are no right minimal indices;*
- *the left minimal indices of $Q(y)$ are $0,\ 2$.*

Notice that $x(\frac{5}{2}) = 20$, $x(\pm\frac{5}{4}) = 0$, and that $y = \frac{5}{2}$ is a root of multiplicity 2 of the equation $x(y) = 20$ while $y = \pm\frac{5}{4}$ are roots of multiplicity 1 of the equation $x(y) = 0$. We can therefore conjecture that if $(x-x_0)^\ell$ is an elementary divisor of $P(x)$ and y_0 is a root of multiplicity m of the equation $x(y) = x_0$ then $(y-y_0)^{m\cdot\ell}$ is an elementary divisor of $Q(y)$. Moreover, looking to what happened to the minimal indices, we see that apparently they have been multiplied by a factor 2; notice that 2 is the degree of the considered rational transformation (that is the maximum of the degrees of the numerator and the denominator).

The main result of the present chapter is the proof that the conjectures above, which will be stated more precisely in Section 3.3, are true for every rational transformation of the variable $x(y)$ and every polynomial matrix $P(x)$. Moreover, analogous properties hold for infinite elementary divisors and right minimal indices.

3.2 Rational transformations of polynomial matrices

Assume that the field $\mathbb{F}$ is algebraically closed.

Let $n(y), d(y) \in \mathbb{F}[y]$ be two nonzero, coprime polynomials. Let us define $N := \deg n(y)$, $D := \deg d(y)$, and $G := \max(N, D)$. We will always suppose $G \geq 1$, that is $n(y)$ and $d(y)$ are not both elements of $\mathbb{F}$. We denote the coefficients of $n(y)$ and $d(y)$ as $n_i \in \mathbb{F}$, $i = 0, \ldots, N$ and $d_j \in \mathbb{F}$, $j = 0, \ldots, D$.

We consider the generic rational function from $\mathbb{F}^*$ to $\mathbb{F}^*$ defined as

$$x(y) = \frac{n(y)}{d(y)}. \tag{3.1}$$

The function (3.1) induces a mapping $\Phi_{g,n(y),d(y)} : (\mathbb{F}[x])^{m\times p} \to (\mathbb{F}[y])^{m\times p}$ defined as

$$\Phi_{g,n(y),d(y)}(P(x)) = Q(y) := [d(y)]^g P(x(y)) \tag{3.2}$$

Here g is the grade of $P(x) \in (\mathbb{F}[x])^{m\times p}$, so for any choice of g a different mapping is defined.

Remark 3.1. *The reversal of a polynomial matrix is a well-known example of one such mapping, corresponding in our notation to* $\Phi_{g,1,y}$.

We will usually omit the functional dependence of Φ on $n(y)$ and $d(y)$ unless the context allows any possible ambiguity; also, if the grade g is chosen to be equal to the degree $k = \deg P(x)$, we will sometimes omit the subscript g, that is $\Phi(P(x)) := \Phi_{k,n(y),d(y)}(P(x))$.

Since a polynomial matrix is also a matrix polynomial, we can write $P(x) = \sum_{i=0}^{g} P_i x^i$ for some $P_i \in \mathbb{F}^{m\times p}$, $i = 0, \ldots, g$. Notice that following the same point of view we can also write $Q(y) = \sum_{i=0}^{g} P_i [n(y)]^i [d(y)]^{g-i}$. We have the following upper bound on the degree of $Q(y)$.

Lemma 3.1. $\deg Q(y) = \deg \Phi_g(P(x)) \leq q := gD + \max_{i:P_i\neq 0}(iN - iD)$. *If* $N \neq D$ *the strict equality* $\deg Q(y) = q$ *always holds. Moreover,* $q \leq gG$.

Proof. Writing $Q(y)$ as above, we can see it as a sum of the $k+1$ polynomial matrices $Q_i(y) = P_i[n(y)]^i[d(y)]^{g-i}$, $0 \leq i \leq k$, with either $Q_i(y) = P_i = 0$ or $\deg Q_i(y) = gD + i(N-D)$. Since the degree of the sum of two polynomials cannot exceed the greatest of the degrees of the considered polynomials, $\deg Q(y)$ cannot be greater than q. Notice that if $N = G$ then $gG \geq q = kG + (g-k)D$ and the maximum is realised by $i = k$, while otherwise the maximum is realised by the smallest index j such that $P_j \neq 0$, and $q = (g-j)G + jN$. This means that if $N < G$ and $P_0 = 0$ then $q < gG$, while $q = gG$ if $N < G$ but $P_0 \neq 0$.

Notice finally that, if $i_1 \neq i_2$, then $Q_{i_1}(y)$ and $Q_{i_2}(y)$ have the same degree if and only if $D = N$. Since $\deg Q_{i_1}(y) \neq \deg Q_{i_2}(y) \Rightarrow \deg(Q_{i_1}(y) + Q_{i_2}(y)) = \max(\deg Q_{i_1}(y), \deg Q_{i_2}(y))$, $D \neq N$ is a sufficient condition for $\deg Q(y) = q$. □

Lemma 3.1 shows that $\deg Q(y) \leq q \leq gG$. The next proposition describes the conditions under which the equality $\deg Q(y) = gG$ holds.

Proposition 3.1. *Let* $Q(y) = \Phi(P(x)), P(x) \neq 0$. *It always holds* $\deg Q(y) \leq gG$, *and* $\deg Q(y) < gG$ *if and only if one of the following is true:*

1. *$N > D$ and $g > k$;*

2. *$N \leq D$, and there exist a natural number $a \geq 1$ and a polynomial matrix $\hat{P}(x) \in (\mathbb{F}[x])^{m \times p}$ such that $P(x) = (x - \hat{x})^a \hat{P}(x)$, where $\hat{x} := n_G d_G^{-1}$ if $N = D = G$ and $\hat{x} := 0_{\mathbb{F}}$ if $N < D = G$.*

Proof. Lemma 3.1 guarantees $\deg Q(y) \leq gG$. To complete the proof, there are three possible cases to be analysed.

- If $G = N > D$, we know from Lemma 3.1 that $\deg Q(y) = q$, and in this instance $q = gD + kN - kD$. Therefore, $\deg Q(y) = gG \Leftrightarrow g = k$.

- If $N = D = G$ and, we get $q = gG$. Let $Q(y) = \sum_{i=0}^{gG} \Theta_i y^i$: then, $\deg Q(y) < gG \Leftrightarrow \Theta_{gG} = 0_{(\mathbb{F}[x])^{m \times p}}$. On the other hand Θ_{gG} is the coefficient of y^{gG} in $Q(y) = \sum_{i=0}^{g} P_i [n(y)]^i [d(y)]^{g-i}$, so $\Theta_{gG} = d_G^g \sum_{i=0}^{g} P_i n_G^i d_G^{-i} = d_G^g P(n_G d_G^{-1})$. Therefore, Θ_{gG} is zero if and only if every entry of $P(n_G d_G^{-1})$ is equal to $0_{\mathbb{F}[x]}$, or in other words if and only if $P(x) = (x - n_G d_G^{-1})^a \hat{P}(x)$ for some $a \geq 1$ and some suitable polynomial matrix $\hat{P}(x)$.

- If $N < D = G$, recalling the proof of Lemma 3.1 we conclude that $\deg Q(y) < gG$ if and only if $P_0 = 0$, which is equivalent to $P(x) = x^a \hat{P}(x)$ for a suitable value of $a \geq 1$ and some polynomial matrix $\hat{P}(x)$.

□

The grade of $Q(y)$ is of course arbitrary, even though it must be greater than or equal to its degree. Since $\deg Q(y) \leq q \leq gG$, we shall define that the grade of $Q(y)$ is gG. This choice has an influence on the infinite elementary divisors of $Q(y)$, as they are equal to the elementary divisors corresponding to zero of the reversal of $Q(y)$ *taken with respect to its grade*, that is $\mathrm{Rev}_{(gG)} Q(y)$.

If one is interested in picking a different choice for the grade of $Q(y)$, the following proposition explains how the infinite elementary divisors change.

Proposition 3.2. *Let $P(x) \in (\mathbb{F}[x])^{m \times p}$, with $k = \deg P(x)$. Then the finite elementary divisors and the minimal indices of $P(x)$ do not depend on its grade, while the infinite elementary divisors do. Namely, let $\nu = \min(m, p)$; $x^{g-k} d_1(x), \ldots, x^{g-k} d_\nu(x)$ are the invariant polynomials of $\mathrm{Rev}_g P(x)$ if and only if $d_1(x), \ldots, d_\nu(x)$ are the invariant polynomials of $\mathrm{Rev}_k P(x)$, for any choice of $g \geq k$.*

Proof. Neither Theorem 1.2 nor the properties of $\ker P(x)$ and $\ker P^T(x)$ depend on the grade, so minimal indices and finite elementary divisors cannot be affected by different choices. Let $S(x) = A(x)\mathrm{Rev}_k P(x) B(x)$ be the Smith form of $\mathrm{Rev}_k P(x)$. We have $\mathrm{Rev}_g P(x) = x^{g-k} \mathrm{Rev}_k P(x)$, which implies that $x^{g-k} S(x) = A(x)\mathrm{Rev}_g P(x) B(x)$. Clearly $d_i(x) | d_j(x) \Leftrightarrow x^{g-k} d_i(x) | x^{g-k} d_j(x)$, and therefore we conclude that $x^{g-k} S(x)$ is the Smith form of $\mathrm{Rev}_g P(x)$. □

Let $\alpha, \beta, \gamma, \delta \in \mathbb{F}$. If $G = 1$, $\Phi_{g, \alpha y + \beta, \gamma y + \delta}$ is clearly invertible and its inverse, with a little abuse of notation, is $\Phi_{g, \beta - \delta x, \gamma x - \alpha} : Q(y) \to [\gamma x - \alpha]^g Q(\frac{\beta - \delta x}{\gamma x - \alpha})$. The most general case is analysed below.

Proposition 3.3. *Let us denote by $\mathbb{F}[x]_g$ the set of the univariate polynomials in x whose degree is less than or equal to g. Given $g, n(y), d(y)$, the mapping $\Phi_{g,n(y),d(y)} : (\mathbb{F}[x]_g)^{m\times p} \to (\mathbb{F}[y]_{(gG)})^{m\times p}$ is always an injective function, but it is not surjective unless $G = 1$.*

Proof. Notice that Φ_g can be thought as acting componentwise, sending $P(x)_{ij}$ to $Q(y)_{ij} = \Phi_g(P(x)_{ij})$. Thus, it will be sufficient to show that, in the scalar case $\Phi_g : \mathbb{F}[x]_g \to \mathbb{F}[y]_{(gG)}$, Φ_g is surjective if and only if $G = 1$. This is true because any polynomial that does *not* belong to the set $R_y := \{a(y) \in \mathbb{F}[y] : a(y) = \sum_{i=0}^{g} a_i [d(y)]^{g-i} [n(y)]^i\}$ cannot belong to the image of Φ_g, and $R_y = \mathbb{F}[y]_{(gG)}$ if and only if $G = 1$. In fact, if we require that a generic $r \in \mathbb{F}[y]_{(gG)}$ belongs to R_y, we find out that the $g+1$ coefficients a_i must satisfy $gG + 1$ linear constraints.

To prove injectivity: $\Phi_g(P_1(x)) = \Phi_g(P_2(x)) \Rightarrow P_1(x(y)) = P_2(x(y)) \Rightarrow P_1(x) = P_2(x)$. □

Proposition 3.3 tells us that, unless $G = 1$ (the Möbius case), not every $Q(y)$ is such that $Q(y) = \Phi(P(x))$ for some $P(x)$.

A couple of additional definitions will turn out to be useful in the following. Let $x_0 \in \mathbb{F}^*$: we define T_{x_0} as the preimage of x_0 under the rational function $x(y)$.

Moreover let $\alpha, \beta \in \mathbb{F}$ be such that $\frac{\alpha}{\beta} = x_0$ and α and β are not both zero. For instance, we can pick $(\alpha, \beta) = (x_0, 1_\mathbb{F})$ if $x_0 \neq \infty$ and $(\alpha, \beta) = (1_\mathbb{F}, 0_\mathbb{F})$ otherwise. Consider the polynomial equation

$$\alpha d(y) = \beta n(y). \tag{3.3}$$

Let S be the degree of the polynomial $\alpha d(y) - \beta n(y)$. Equation (3.3) cannot have more than S finite roots. If $S < G$ then we formally say $\infty \in T_{x_0}$.

Remark 3.2. *Notice that there are three cases that lead to $S < G$:*

1. *$N = D = G$ and $x_0 = n_G d_G^{-1}$, so that (3.3) becomes $d_G n(y) = n_G d(y)$: in this case, S is the maximum value of i such that $n_i \neq x_0 d_i$;*

2. *$N < D = G$ and $x_0 = 0_\mathbb{F}$, so that (3.3) becomes $n(y) = 0_\mathbb{F}$ and $S = N$;*

3. *$D < N = G$ and $x_0 = \infty$, so that (3.3) is $d(y) = 0_\mathbb{F}$ and $S = D$.*

We now define the multiplicity m_0 of any finite $y_0 \in T_{x_0}$ as the multiplicity of y_0 as a solution of the polynomial equation (3.3). If $\infty \in T_{x_0}$, its multiplicity is defined to be equal to $G - S$. Therefore, the sum of the multiplicities of all the (both finite and infinite) elements of T_{x_0} is always equal to G, while the sum of the multiplicities of all the finite elements of T_{x_0} is S.

The finite elements of T_{x_0} are characterised by the following proposition.

Proposition 3.4. *Let $y_0 \in \mathbb{F}$ and $x_0 \in \mathbb{F}^*$. Then $y_0 \in T_{x_0}$ if and only if y_0 is a solution of (3.3) for $\alpha, \beta : x_0 = \frac{\alpha}{\beta}$. Moreover, the two equations $\alpha_1 d(y_0) = \beta_1 n(y_0)$ and $\alpha_2 d(y_0) = \beta_2 n(y_0)$ are simultaneously satisfied if and only if $\frac{\alpha_1}{\beta_1} = \frac{\alpha_2}{\beta_2}$.*

Proof. The definition of T_{x_0} implies the first part of the proposition. The second part comes from the fact that $x(y)$ is a function. □

Proposition 3.4, albeit rather obvious, has the following important implication:

Corollary 3.1. *$x_0 \neq x_1 \Leftrightarrow T_{x_0} \cap T_{x_1} = \emptyset$. Equivalently, $\alpha_1\beta_2 \neq \alpha_2\beta_1$ if and only if $[\beta_1 n(y) - \alpha_1 d(y)]$ and $[\beta_2 n(y) - \alpha_2 d(y)] \in \mathbb{F}[y]$ are coprime.*

In particular, for any finite $x_0 \in \mathbb{F}$, $\Phi(x - x_0)$ and $d(y)$ are coprime.

In order to clarify the latter definitions, let us consider an example.

Example 3.2. *Let $\mathbb{F} = \mathbb{C}$ and take $n(y) = y^4 + y^3 - y^2 - y + 1$, $d(y) = y^4$. T_1 is the set of the solutions of the equation $n(y) = d(y)$, so in this case $T_1 = \{-1, 1, \infty\}$. Moreover, the multiplicity of -1 and 1 are, respectively, 1 and 2; since $S = 3$ and $G = 4$, the multiplicity of ∞ is by definition $G - S = 1$. Within the same example, $T_\infty = \{0\}$; 0 has multiplicity 4 because it is a root of order 4 of the equation $d(y) = 0$.*

3.3 Main result

We are now ready to state the main result of this chapter.

Theorem 3.1. *Given $m, p \in \mathbb{N}_0$ and $n(y), d(y) \in \mathbb{F}[y]$, let $x_0 \in \mathbb{F}^*$ be a characteristic value of $P(x) \in (\mathbb{F}[x])^{m\times p}$, and let $(x - x_0)^{\ell_1}, \ldots, (x - x_0)^{\ell_j}$ be the corresponding elementary divisors. Let g be the grade of $P(x)$ and define $G = \max(\deg n(y), \deg d(y))$. Let $Q(y) = \Phi_g(P(x)) := [d(y)]^g P(\frac{n(y)}{d(y)}) \in (\mathbb{F}[y])^{m\times p}$ and let gG be the grade of $Q(y)$. Then for any $y_0 \in T_{x_0}$:*

- *y_0 is a characteristic value of $Q(y)$;*
- *$(y - y_0)^{m_0\ell_1}, \ldots, (y - y_0)^{m_0\ell_j}$ are the elementary divisors corresponding to y_0 for $Q(y)$, where m_0 is the multiplicity of y_0.*

Conversely, if $Q(y) = \Phi_g(P(x))$ for some $P(x)$, and if $y_0 \in \mathbb{F}^*$ is a characteristic value of $Q(y)$ with corresponding elementary divisors $(y-y_0)^{\kappa_1}, \ldots, (y-y_0)^{\kappa_j}$:

- *$x_0 = \frac{n(y_0)}{d(y_0)}$ is a characteristic value of $P(x)$;*
- *$m_0 | \kappa_i \ \forall i \leq j$, where m_0 is the multiplicity of y_0 as an element of T_{x_0}, and $(x-x_0)^{m_0^{-1}\kappa_1}, \ldots, (x-x_0)^{m_0^{-1}\kappa_j}$ are the elementary divisors corresponding to x_0 for $P(x)$.*

In addition, the following properties hold:

- *the right minimal indices of $P(x)$ are $\beta_1, \ldots, \beta_s$ if and only if the right minimal indices of $Q(y)$ are $G\beta_1, \ldots, G\beta_s$;*
- *the left minimal indices of $P(x)$ are $\gamma_1, \ldots, \gamma_r$ if and only if the left minimal indices of $Q(y)$ are $G\gamma_1, \ldots, G\gamma_r$.*

For any choice of the mapping Φ_g, Theorem 3.1 gives a thorough description of the complete eigenstructure of $\Phi_g(P(x))$ with respect to the complete eigenstructure of $P(x)$. Notice that if $x(y)$ is a Möbius transformation then $m_0 \equiv 1$

and $G = 1$, so the complete eigenstructure is unchanged but for the shift from one set of characteristic values to another. This is not the case for more general rational transformations, where other changes do happen.

The structure of the proof of Theorem 3.1, that was given in [93] and will be recalled in the next sections, is the following: first we shall prove the first part of the theorem (the statement on elementary divisors). This will be done dividing the statement in three cases:

1. $x_0 \in \mathbb{F}$ and $y_0 \in \mathbb{F}$;
2. $x_0 \in \mathbb{F}$ and $y_0 = \infty$;
3. $x_0 = \infty$.

We shall first prove that the statement is true for case 1. Then, we will show that this implies that it is true for case 2. Finally, the validity of cases 1 and 2 implies case 3. This result can be extended also to fields that are not algebraically closed: see Section 3.6.

Finally, we shall prove the second part of the theorem (the statement on minimal indices) with a constructive proof: we shall build a minimal basis of $Q(y)$ given a minimal basis of $P(x)$, and vice versa.

3.4 Proof of Theorem 3.1: elementary divisors

In this section, we prove the first part of Theorem 3.1. The proof relies on the following lemma:

Lemma 3.2. *Let $P(x) \in (\mathbb{F}[x])^{m\times p}$ and let $R(x) = A(x)P(x)B(x)$ where $A(x) \in M_m(\mathbb{F}[x])$ and $B(x) \in M_p(\mathbb{F}[x])$ are both regular, and suppose that $x_0 \in \mathbb{F}$ is neither a root of $\det A(x) \in \mathbb{F}[x]$ nor a root of $\det B(x) \in \mathbb{F}[x]$. Then x_0 is a characteristic value of $P(x)$ if and only if it is a characteristic value of $R(x)$, and $(x - x_0)^{\ell_1}, \dots (x - x_0)^{\ell_j}$ are the elementary divisors associated with x_0 for $R(x)$ if and only if they are the elementary divisors associated with x_0 for $P(x)$.*

Proof. From Theorem 1.4, it suffices to prove that $P(x)$ has a maximal set of j x_0-independent root polynomials, whose orders are $\ell_1 \leq \cdots \leq \ell_j$, if and only if $R(x)$ has a maximal set of j x_0-independent root polynomials, of order $\ell_1 \leq \cdots \leq \ell_j$. We know from Theorem 1.3 that $v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $R(x) = A(x)P(x)B(x)$ if and only if $B(x)v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $P(x)$. Let us prove that x_0-independence is conserved: to this goal, observe that, as long as $\det B(x_0) \neq 0_{\mathbb{F}}$, the polynomial vectors $v_1(x_0), \dots, v_j(x_0)$ are linearly independent if and only if the polynomial vectors $B(x_0)v_1(x_0), \dots, B(x_0)v_j(x_0)$ are linearly independent.

In order to complete the proof, we ought to show that $\{v_1(x), \dots, v_j(x)\}$ is a maximal set of x_0-independent root polynomials of $R(x)$ if and only if $\{B(x)v_1(x), \dots, B(x)v_j(x)\}$ is a maximal set of x_0-independent root polynomials of $P(x)$. Suppose that $v_1(x), \dots, v_j(x)$ do not form a maximal set; recalling the definitions that we gave in Chapter 1, we see that there are three cases for this to happen. In the first case, there exists a root polynomial $w(x)$ of $R(x)$ corresponding to x_0 of order $\ell > \ell_j$: then $B(x)w(x)$ is a root polynomial of

$P(x)$ of order ℓ. In the second case, there exist x_0-independent root polynomials $\hat{v_k}(x), v_{k+1}(x) \ldots, v_j(x)$, such that the order of $\hat{v_k}(x)$ is greater than the order of $v_k(x)$, for some $1 \leq k \leq j$. But this means that the same argument holds for $B(x)\hat{v_k}(x)$ with respect to $B(x)v_k(x)$. In the third case, there exists a root polynomial $v(x)$ such that $v_1(x), \ldots, v_j(x), v(x)$ are x_0-independent. This implies that $B(x)v_1(x), \ldots, B(x)v_j(x), B(x)v(x)$ are x_0-independent. In each of the three cases, we can conclude that $B(x)v_1(x), \ldots, B(x)v_j(x)$ do not form a maximal set either. The proof of the reverse implication is analogous and is omitted.
□

3.4.1 Case 1

Recall first that, if x_0 is finite, then for any $y_0 \in T_{x_0}$ there must hold $d(y_0) \neq 0_{\mathbb{F}}$ because of Corollary 3.1.

Let now $P(x) = A(x)T(x)B(x)$ where $A(x)$ and $B(x)$ are unimodular polynomial matrices, $T(x) =: \mathrm{diag}(\delta_1(x), \ldots, \delta_\nu(x))$ is the Smith form of $P(x)$. Here $\nu := \min(m, p)$ and $\delta_i(x)$ are the invariant polynomials of $P(x)$. We recall that the Smith form exists and is a diagonal matrix even when $P(x)$ is not square (we adopt the terminology and the notations introduced in Subsection 1.2.1). Let $E(y) := \Phi(A(x))$, $F(y) := \Phi(B(x))$, $S(y) := \Phi(T(x))$, and define $\hat{Q}(y) := E(y)S(y)F(y)$. Clearly, $\hat{Q}(y)$ and $Q(y)$ differ only for a multiplicative factor of the form $[d(y)]^\lambda$, $\lambda \in \mathbb{N}$. Moreover, observe that both $\det E(y)$ and $\det F(y)$ are nonzero whenever $d(y) \neq 0_{\mathbb{F}}$. Therefore, Lemma 3.2 implies that $\hat{Q}(y)$ and $S(y)$ have the same elementary divisors of $Q(y)$ for any characteristic value y_0 such that $d(y_0) \neq 0_{\mathbb{F}}$. This condition is necessarily true because of Corollary 3.1, and so it will be sufficient to prove the proposition for $S(y)$.

Unfortunately, $S(y)$ may not be the Smith form of $\hat{Q}(y)$, because neither $E(y)$ nor $F(y)$ are necessarily unimodular and also because $\Phi(\delta_i(x))$ may not be monic. Nevertheless, it has the form $\mathrm{diag}([d(y)]^{k_1}\hat{\delta}_1(y), \ldots, [d(y)]^{k_\nu}\hat{\delta}_\nu(y))$, where $k_1 \geq k_2 \geq \cdots \geq k_\nu$ and $\hat{\delta}_i(y) := \Phi(\delta_i(x))$. From Corollary 3.1, $\hat{\delta}_i(y)$ and $d(y)$ cannot share common roots. To reduce $S(y)$ into a Smith form, we proceed by steps working on 2×2 principal submatrices.

In each step, we consider the submatrix $\begin{bmatrix} [d(y)]^\gamma\hat{\delta}_i(y) & 0 \\ 0 & [d(y)]^\phi\hat{\delta}_j(y) \end{bmatrix}$, where $\gamma := k_i$ and $\phi := k_j$, with $i < j$. If $\gamma = \phi$, then do nothing; if $\gamma > \phi$, premultiply the submatrix by $\begin{bmatrix} 1_{\mathbb{F}} & 1_{\mathbb{F}} \\ -b(y)q(y) & 1_{\mathbb{F}} - b(y)q(y) \end{bmatrix}$ and postmultiply it by $\begin{bmatrix} a(y) & -q(y) \\ b(y) & [d(y)]^{\gamma-\phi} \end{bmatrix}$, where $q(y) = \hat{\delta}_j(y)/\hat{\delta}_i(y)$ while $a(y)$ and $b(y)$ are such that $a(y)[d(y)]^\gamma\hat{\delta}_i(y) + b(y)[d(y)]^\phi\hat{\delta}_j(y) = [d(y)]^\phi\hat{\delta}_i(y)$; the existence of two such polynomials is guaranteed by Bezout's lemma, since $[d(y)]^\phi\hat{\delta}_i(y)$ is the greatest common divisor of $[d(y)]^\gamma\hat{\delta}_i(y)$ and $[d(y)]^\phi\hat{\delta}_j(y)$. It is easy to check that both matrices are unimodular, and that the result of the matrix multiplications is $\begin{bmatrix} [d(y)]^\phi\hat{\delta}_i(y) & 0 \\ 0 & [d(y)]^\gamma\hat{\delta}_j(y) \end{bmatrix}$. Hence, by subsequent applications of this algorithm it is possible to conclude what the Smith form of $S(y)$ is. Namely, after having defined a unimodular diagonal matrix $\Delta \in \mathbb{F}^{\nu\times\nu}$ chosen in such a way that the invariant polynomials of $\tilde{S}(y)$ are monic, we see that the Smith form of $S(y)$ is either $\tilde{S}(y) := \hat{S}(y) \cdot \Delta$ or $\tilde{S}(y) := \Delta \cdot \hat{S}(y)$ (whichever of the two matrix product makes sense, depending

on whether $m \leq p$ or not), where

$$\hat{S}(y) = \mathrm{diag}([d(y)]^{k_\nu}\hat{\delta}_1(y), \ldots, [d(y)]^{k_1}\hat{\delta}_\nu(y)).$$

Thus, the ith invariant polynomial of $P(x)$ has a root of multiplicity ℓ_i at x_0 if and only if the ith invariant polynomial of $\hat{Q}(y)$ has a root of multiplicity $m_0\ell_i$ at $y_0 \in T_{x_0}$.

3.4.2 Case 2

By definition, the infinite elementary divisors for a given polynomial matrix are the elementary divisors corresponding to zero of the reversal of such polynomial matrix. Therefore, in order to prove Theorem 3.1 for the case of $y_0 = \infty$, we have to analyse the polynomial matrix

$$Z(y) := \mathrm{Rev}_{(gG)}Q(y) = y^{gG}[d(y^{-1})]^g P(x(y^{-1})),$$

and find out what its relation to $P(x)$ is, with particular emphasis to its elementary divisors corresponding to $y_0 = 0_{\mathbb{F}}$. Recall Remark 3.2 about the situations where $S < G$. From that analysis we see that there are two distinct situations for which $\infty \in T_{x_0}$ for a finite $x_0 \in \mathbb{F}$:

2.1 $N = D = G$ and $x_0 = n_G d_G^{-1}$;

2.2 $N < D = G$ and $x_0 = 0_{\mathbb{F}}$.

We will consider such subcases separately.

Subcase 2.1

We get $x(y^{-1}) = \dfrac{\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)}$ and $y^G d(y^{-1}) = \mathrm{Rev}d(y)$; therefore we can write $Z(y) = [\mathrm{Rev}d(y)]^g P(\dfrac{\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)})$. This means that we can prove analogous results for $Z(y)$ just by considering this time the new rational transformation $y \to x = \dfrac{\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)}$. Notice also that $0_{\mathbb{F}}$ is a root of multiplicity $G - S$ for the equation $\mathrm{Rev}n(y) = x_0\mathrm{Rev}d(y)$; moreover, since we took the reversal with respect to the degree (or also because of Corollary 3.1), $0_{\mathbb{F}}$ cannot be a root of $\mathrm{Rev}d(y)$.Therefore, following the proof given above, one can state that $P(x)$ has $(x - x_0)^{\ell_1}, \ldots, (x - x_0)^{\ell_j}$ as elementary divisors corresponding to x_0 if and only if $Z(y)$ has the j elementary divisors $y^{(G-S)\ell_1}, \ldots, y^{(G-S)\ell_j}$ corresponding to $0_{\mathbb{F}}$. The thesis follows immediately.

Subcase 2.2

This time, we can write $x(y^{-1}) = \dfrac{y^{G-N}\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)}$; therefore, applying the definition of $Z(y)$, we get $Z(y) = [\mathrm{Rev}d(y)]^g P(\dfrac{y^{G-N}\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)})$. It is therefore sufficient to consider the transformation $y \to x = y^{G-N}\dfrac{\mathrm{Rev}n(y)}{\mathrm{Rev}d(y)}$.

In fact, notice that $0_{\mathbb{F}}$ is a solution of multiplicity $G - N$ for the equation $y^{G-N}\text{Rev}n(y) = 0$ ($0_{\mathbb{F}}$ is neither a root of Rev$n(y)$ nor a root of Rev$d(y)$, because Rev$n(0_{\mathbb{F}}) = n_N \neq 0_{\mathbb{F}}$ and Rev$d(0_{\mathbb{F}}) = d_D \neq 0_{\mathbb{F}}$). Thus, using the correspondence of Smith forms as before and invoking Lemma 3.2, we can state that $P(x)$ has the j elementary divisors $x_1^{\ell}, \ldots, x_j^{\ell}$ corresponding to $0_{\mathbb{F}}$ if and only if $Z(y)$ has the j elementary divisors $y^{(G-N)\ell_1}, \ldots, y^{(G-N)\ell_j}$ corresponding to $0_{\mathbb{F}}$, and the thesis follows.

3.4.3 Case 3

By definition, the infinite elementary divisors of $P(x)$ are the elementary divisors corresponding to the characteristic value $0_{\mathbb{F}}$ for $R(x) = \text{Rev}_g P(x)$. But let $\Psi_{g,n(y),d(y)} = \Phi_{g,d(y),n(y)}$ and $U(y) = \Psi_g(R(x))$, that is to say $U(y) = [n(y)]^g R(\frac{d(y)}{n(y)})$. A simple calculation gives

$$U(y) = [n(y)]^g [\frac{d(y)}{n(y)}]^g P([\frac{d(y)}{n(y)}]^{-1}) = [d(y)]^g P(\frac{n(y)}{d(y)}) = \Phi_g(P(y)) = Q(y).$$

One can therefore follow the proof as in the previous subsections, but starting from $R(x)$ and using a different transformation (notice that the equation that defines T_∞ for the old transformation is $d(y) = 0_{\mathbb{F}}$, and it is equal to the equation that defines $T_{0_{\mathbb{F}}}$ for the new transformation).

3.4.4 A corollary on Jordan chains

By Theorem 3.1, we know that the elementary divisors of $Q(y)$ are linked to the elementary divisors of $P(x)$. On the other hand, Theorem 1.4 tells us that this fact is linked with the existence of an appropriate maximal set of root polynomials.

We will now build y_0-independent root polynomials of $Q(y)$ starting from the x_0-independent root polynomials of $P(x)$. This approach will allow us to say something more on Jordan chains.

Let $v(x)$ be a root polynomial of order ℓ for $P(x)$, corresponding to the characteristic value x_0. We have

$$P(x)v(x) = (x - x_0)^{\ell}\theta(x), \tag{3.4}$$

with $\text{GCD}(x - x_0, \theta(x)) = 1_{\mathbb{F}[x]}$. Let k_v be the degree of $v(x)$, and notice that, if x_0 is finite, then for any $y_0 \in T_{x_0}$ there must hold $d(y_0) \neq 0_{\mathbb{F}}$ (Corollary 3.1). If follows that (3.4) is equivalent to

$$[d(y)]^{g+k_v} P(x(y))v(x(y)) = [d(y)]^{g+k_v}(x(y) - x_0)^{\ell}\theta(x(y))$$

or equivalently

$$\Phi_g(P(x))\Phi(v(x)) = \Phi((x - x_0)^{\ell})\Phi_{g+k_v-\ell}(\theta(x)).$$

Define $w(y) := \Phi(v(x))$. The last equation then reads

$$Q(y)w(y) = (y - y_0)^{m_0 \cdot \ell}\rho(y),$$

where

$$\rho(y) = \kappa \cdot \left[\prod_{y_i \in T_{x_0}, y_i \neq y_0} (y - y_i)^{\ell \cdot m_i} \right] \Phi_{g+k_v-\ell}(\theta(x)).$$

Here $\kappa \in \mathbb{F}$ is a suitable constant, m_i is the multiplicity of y_i as a root of the equation (3.3), and we adopt the formal convention $y - \infty := 1_{\mathbb{F}}$ in order to deal with the possibility of infinite y_i.

In Section 3.5 we will prove that $[v_1(x), \ldots, v_s(x)]$ is a minimal basis of $\ker P(x)$ if and only if $[\Phi(v_1(x)), \ldots, \Phi(v_s(x))]$ is a minimal basis of $\ker Q(y)$: this implies that $v(x_0) \in \ker_{x_0} P(x) \Leftrightarrow w(y_0) \in \ker_{y_0} Q(y)$. Corollary 3.1 allows to conclude that $w(y)$ is a root polynomial of order $m_0\ell$ corresponding to y_0.

We can repeat the same procedure with a whole maximal set of x_0-independent root polynomials: we will now show that the resulting root polynomials of $Q(y)$ are y_0-independent. In fact, denote by $V(x)$ the matrix $[v_1(x), \ldots, v_j(x)]$ formed by the x_0-independent root polynomials of $P(x)$, and let us call $W(y) = [\Phi(v_1(x)), \ldots, \Phi(v_j(x))]$. Since $d(y_0) \neq 0_{\mathbb{F}}$, we have that $\text{rank}W(y_0) = \text{rank}V(x_0)$. Hence the x_0-independence of the root polynomials of $P(x)$ implies, and is implied by, the y_0 independence of the root polynomials of $Q(y)$ built as above.

Finally, by Theorem 1.4 we see that the root polynomials that we have built must be a maximal set, because they correspond to the exponents of the elementary divisors of $Q(y)$, which are uniquely determined.

Observe that this explicit construction of the root polynomials allows to formulate the following Corollary of Theorem 3.1:

Corollary 3.2. *If $Q(y) = \Phi_g(P(x))$ and $y_0 \in T_{x_0}$ with multiplicity m_0, then $v(x)$ is a root polynomial of order ℓ corresponding to x_0 for $P(x)$ if and only if $w(y)$ is a root polynomial of order $m_0\ell$ corresponding to y_0 for $Q(y)$, with*

$$v(x) = \sum_{i=0}^{\ell-1} (x - x_0)^i v_i \Leftrightarrow w(y) = \sum_{i=0}^{\ell-1} [d(y)]^{\ell-1-i} [n(y) - x_0 d(y)]^i v_i, \qquad (3.5)$$

where $v_i \in \mathbb{F}^m$ are suitable constant vectors.

Equation (3.5) contains, via Theorem 1.5, all the information on Jordan chains.

3.5 Proof of Theorem 3.1: minimal indices

We shall only prove the theorem for right minimal indices. The proof for left minimal indices follows from the proof for right minimal indices and from the fact that Φ and the operation of transposition commute, that is $\Phi_g(P^T(x)) = (\Phi_g(P(x)))^T \ \forall \ P(x) \in (\mathbb{F}[x])^{m \times p}$.

3.5.1 Necessity

Let $\dim \ker P(x) = s$, and $V(x) = [v_1(x), \ldots, v_s(x)]$ be a minimal basis for $\ker P(x)$, with minimal indices $\beta_i := \deg v_i \ \forall i = 1, \ldots, s$ and order $B := \sum_{i=1}^{s} \beta_i$. For each value of i let us define $w_i(y) := \Phi_{\beta_i}(v_i(x))$; we also define $W(y) := [w_1(y), \ldots, w_s(y)]$. Clearly $\deg w_i(y) = G\beta_i$. Suppose in fact

$\deg w_i(y) \neq G\beta_i$; applying Proposition 3.1 (in the case $g = k = \beta_i$), this would imply that there exists some $x_0 \in \mathbb{F}$ and some polynomial vector $u(x) \in (\mathbb{F}[x])^p$ such that $v_i(x) = (x - x_0)u(x)$. Hence, $[v_1(x), \ldots, (x - x_0)^{-1}v_i(x), \ldots, v_s(x)]$ would be a polynomial basis of order $B-1$ for $\ker P(x)$, leading to a contradiction. In order to prove that $W(y)$ is a minimal basis for $\ker Q(y)$ we must show that it is a basis and that it is minimal.

Clearly $w_i(y)$ lies in $\ker Q(y)$ for all i. In fact, $P(x)v_i(x) = 0$ implies that $Q(y)w_i(y) = 0$. Moreover, since $\mathrm{rank}(Q(y)) = \mathrm{rank}P(x(y)) = \mathrm{rank}(P(x))$, $\dim \ker Q(y) = s$. Therefore, it is sufficient to show that $W(y)$, considered as an element of $(\mathbb{F}(x))^{p\times s}$, has rank s.

Notice that $W(y) = V(x(y)) \cdot \mathrm{diag}([d(y)]^{\beta_1}, \ldots, [d(y)]^{\beta_s})$. A well-known property of the rank is that, if $A_1 = A_2A_3$ and A_3 is square and regular, then $\mathrm{rank}(A_1) = \mathrm{rank}(A_2)$. Therefore $\mathrm{rank}(W(y)) = \mathrm{rank}(V(x(y))$, because the diagonal matrix above is regular. Let $\hat{V}(x)$ be some regular $s \times s$ submatrix of $V(x)$, which exists because $\mathrm{rank}(V(x)) = s$. By hypothesis, $\det(\hat{V}(x)) \neq 0_{\mathbb{F}[x]}$, which implies $\det(\hat{V}(x(y))) \neq 0_{\mathbb{F}(y)}$. Hence $s = \mathrm{rank}(V(x(y))) = \mathrm{rank}(W(y))$. Then $W(y)$ is a basis.

In order to prove that it is minimal, let us introduce the following lemma whose proof can be found in [34].

Lemma 3.3. *Let $\mathcal{V}$ be a vector subspace of $\mathbb{F}(x)^p$, with $\dim \mathcal{V} = s$. Let $H = [h_1, \ldots, h_s]$ be a polynomial basis of order A for $\mathcal{V}$ and define ξ_i, $i = 1, \ldots, \binom{p}{s}$ to be the $s \times s$ minors (i.e. determinants of $s \times s$ submatrices) of H. Then the following statements are equivalent:*

- *H is a minimal basis for $\mathcal{V}$*
- *The following conditions are both true: (a) $\mathrm{GCD}(\xi_1, \ldots, \xi_r) = 1_{\mathbb{F}[x]}$ and (b) $\max_i \deg \xi_i = A$.*

So let $\xi_i(y)$ be the $s \times s$ minors of $W(y)$. We shall prove that (a) their GCD is $1_{\mathbb{F}[y]}$ and (b) their maximal degree is $GB = G\sum_{i=1}^{s} \beta_i$. By Lemma 3.3, these two conditions imply that $W(y)$ is minimal. Recall that $w_i(y) = \Phi_{\beta_i}(v_i(x))$, that is to say $w_i(y) = [d(y)]^{\beta_i}v_i(x(y))$. Any $s \times s$ submatrix of $W(y)$ is therefore obtained from the corresponding $s \times s$ submatrix of $V(x)$ by applying the substitution $x = x(y)$ and then multiplying the ith column by $[d(y)]^{\beta_i}$ for $i = 1, \ldots, s$. Let us call $\zeta_i(x)$ the $s \times s$ minors of $V(x)$. From the properties of determinants we obtain the relation $\xi_i(y) = \left(\prod_{i=1}^{s}[d(y)]^{\beta_i}\right)\zeta_i(x(y)) = [d(y)^B]\zeta_i(x(y)) = \Phi_B(\zeta_i(x))$.

Now for each i let $\gamma_i := \deg \zeta_i(x)$ and $\delta_i := \deg \xi_i(y) \leq \max_{j\leq\gamma_i}(Nj - Dj) + DB$ where the maximum is taken over those values of j such that the jth coefficient of $\xi_i(y)$ is nonzero (Lemma 3.1). There are two cases. If $N \leq D = G$, $\delta_i \leq GB$, and applying Proposition 3.1 (with $g = B$), the inequality holds if and only if if $(x - \hat{x})|\zeta_i(x)$, where $\hat{x} = 0_{\mathbb{F}}$ if $N < D$ and $\hat{x} = n_G d_G^{-1}$ if $N = D$; notice that there must be at least one value of i for which $\delta_i = GB$, otherwise $(x - \hat{x})$ would be a common factor of all the $\zeta_i(x)$, which is not possible because of Lemma 3.3 (for the same reason at least one $\zeta_i(x)$ must be nonzero). Finally, if $D < N = G$, $\delta_i = \gamma_i G + (B - \gamma_i)D$. Since $V(x)$ is minimal we have $\max_i(\gamma_i) = B$, which implies that also in this case $\max_i(\delta_i) = GB$. This proves condition (b).

Notice moreover that $\xi_i(y) = \Phi_B(\zeta_i(x)) = [d(y)]^{B-\gamma_i}\Phi_{\gamma_i}(\zeta_i(x))$, where the first and the second factor are coprime (because of Corollary 3.1). Let us prove the following lemma.

Lemma 3.4. *Let $p, q, r \in \mathbb{F}[x]$ with r monic. Then,* $\mathrm{GCD}_{\mathbb{F}[x]}(p, q) = r$ *if and only if* $\mathrm{GCD}_{\mathbb{F}[y]}(\Phi_{\deg p}(p), \Phi_{\deg q}(q)) = \kappa \cdot \Phi_{\deg r}(r)$*, where* $\kappa \in \mathbb{F}$ *is such that* $\kappa \cdot \Phi_{\deg r}(r)$ *is monic.*

Proof. Let α, β be two suitable elements of $\mathbb{F}$ and let us write the prime factor decompositions $p = \alpha \cdot \prod (x - p_i)^{\pi_i}$, $q = \beta \cdot \prod (x - q_i)^{\theta_i}$, $r = \prod (x - r_i)^{\rho_i}$. Of course we have that $(x - r_i)^{\rho_i} | r$ if and only if $(x - p_i)^{\pi_i} | p$, $(x - q_i)^{\theta_i} | q$ and $\rho_i = \min(\pi_i, \theta_i)$. Applying (3.1), we have that $\Phi_{\deg p}(p) = \alpha \cdot \prod (n(y) - p_i d(y))^{\pi_i}$, $\Phi_{\deg q}(q) = \beta \cdot \prod (n(y) - q_i d(y))^{\theta_i}$ and $\Phi_{\deg r}(r) = \prod (n(y) - r_i d(y))^{\rho_i}$. The thesis follows by invoking Corollary 3.1. □

Lemma 3.4 implies condition (a). This follows from the equation

$$\mathrm{GCD}_i(\xi_i(y)) = \mathrm{GCD}_i([d(y)]^{B-\gamma_i}) \cdot \mathrm{GCD}_i(\Phi_{\gamma_i}(\zeta_i(x))) = 1_{\mathbb{F}[y]} \cdot 1_{\mathbb{F}[y]}.$$

The first $1_{\mathbb{F}[y]}$ comes from the fact that $\max_i(\gamma_i) = B$, while the second $1_{\mathbb{F}[y]}$ comes by applying the previous lemma to

$$\mathrm{GCD}(\xi_1(y), \ldots, \xi_s(y)) = \mathrm{GCD}(\mathrm{GCD}(\ldots \mathrm{GCD}(\xi_2(y), \xi_1(y)) \ldots))$$

and from the identity $\Phi_0(1_{\mathbb{F}[x]}) = 1_{\mathbb{F}[y]}$.

3.5.2 Sufficiency

To complete the proof, suppose now that $Q(y) = \Phi_g(P(x))$ for some $P(x) \in \mathbb{F}[x]$ and that $\hat{W}(y)$ is a minimal basis for $\ker Q(y)$, with minimal indices $\epsilon_1 \leq \cdots \leq \epsilon_s$. The other implication that we proved in the previous subsection implies that $G | \epsilon_i \ \forall \ i$, so define $\beta_i = \frac{\epsilon_i}{G}$. Suppose that there exists a minimal basis $\hat{V}(x) = (\hat{v}_1(x), \ldots, \hat{v}_s(x))$ for $\ker P(x)$; suppose moreover that an index $i_0 \in \{1, \ldots, s\}$ exists such that $\deg \hat{v}_{i_0} \neq \beta_{i_0}$. Applying the reverse implication, this would imply that there is a minimal basis $\tilde{W}(y) = (\tilde{w}_1(y), \ldots, \tilde{w}_s(y))$ for $\ker Q(y)$ whose i_0th right minimal index is not equal to ϵ_{i_0}. This is absurd because every minimal basis has the same minimal indices.

3.6 Extension to more relaxed hypotheses

For the sake of convenience in exposition, we have so far assumed that $\mathbb{F}$ is algebraically closed. This assumption is heavily used in the construction of Smith forms. Nevertheless, it is possible to state analogous results for fields that are not algebraically closed: to see it, let $\mathbb{K}$ be the algebraic closure of $\mathbb{F}$. Then $(\mathbb{F}[x])^{m \times p} \subseteq (\mathbb{K}[x])^{m \times p}$, so we can use Theorem 3.1 to identify the Smith forms of $P(x)$ and $Q(y) = \Phi_g(P(x))$ over the polynomial rings $\mathbb{K}[x]$ and $\mathbb{K}[y]$. We can then join back elementary divisors in $\mathbb{K}[x]$ and $\mathbb{K}[y]$ to form elementary divisors in $\mathbb{F}[x]$ and $\mathbb{F}[y]$. Of course, in this case an elementary divisor is no more necessarily associated with a characteristic value in $\mathbb{F}$. For instance, if $\mathbb{F} = \mathbb{Q}$, then the elementary divisor $x^2 + 2$ is not associated with any rational characteristic value, but if we consider the field of complex algebraic numbers $\mathbb{K} = \overline{\mathbb{Q}}$ then we can split it as $(x - \sqrt{2}i)(x + \sqrt{2}i)$ and associate it to the characteristic values $\pm\sqrt{2}i$. We have used again the fact that $\mathbb{F}$ is algebraically closed somewhere else (e.g., Lemma 3.4), but once again it is straightforward to extend those results to a generic field $\mathbb{F}$ via an immersion into its algebraic closure $\mathbb{K}$.

Chapter 4

The Ehrlich-Aberth method for structured polynomial eigenproblems

In the present chapter, that relies on the papers [12, 42, 92], we analyse various possible strategies for the adaptation of the EAI to the solution of structured PEPs. There are three main strategies that we used to exactly extract the structure in the spectrum: one is the very simple solution to force the approximation of the EAI to preserve the structure; the second is to explicitly build a new matrix polynomials with less distinct eigenvalues than the original one, and to obtain the sought eigenvalues from such new polynomial; finally, the third method is to make an implicit change of variable.

The second and the third approach deserve more attention, because they are more sophisticated and because from experiments they seem to achieve, in some cases, a better efficiency. We will review the application of both methods to several kinds of structured PEPs.

4.1 Introduction to structured PEPs

The EAI is particularly suited to deal with matrix polynomials endowed with specific structures of the matrix coefficients. We are interested in matrix structures which induce particular symmetries on the location of the eigenvalues. Polynomials of this kind are encountered in the applications and include, for instance, palindromic, T-palindromic, even/odd, skew-Hamiltonian and Hamiltonian/skew-Hamiltonian polynomials.

Customary PEP-solving algorithms, such as the application of the QZ algorithm to any suitable linearization of the polynomial, are not able to fully catch these symmetries of the spectrum. In the literature, there are specific matrix methods that achieve this goal. The EAI enables to exploit the additional information both in the computation of the Newton correction and in the choice and in the management of the (initial) approximation of the roots in view of the structure-induced symmetries. We will often refer to the resulting structured variants of the Ehrlich-Aberth method as the structured Ehrlich-Aberth

iteration (SEAI).

Assume that the structured PEP is such that the eigenvalues appear in pairs $\{x, f(x)\}$, with $f(f(x)) = x \ \forall x$. A naive adaptation of the EAI to this property would be to apply (2.2) or (2.3) updating only the first half of the components of the vector y and simultaneously imposing $y^{(i)} = f(y^{(i-nk/2)})$, $i = nk/2 + 1, \ldots, nk$. From numerical experiments we see that this approach is working well sometimes, while in other instances it does not seem to be very efficient in terms of number of scalar iterations needed for numerical convergence. This motivates the design of more sophisticated structured variants of the EAI, that we are going to describe in the following. A more detailed comparison of the efficiencies of the "naive" approach and the "sophisticated" methods is reported in Section 4.6.

In the following sections, we will analyse various classes of structured matrix polynomials and describe the design of different adaptations of the EAI that aim to solve them. Before doing that, let us recall some basic definitions of special matrices.

An $n \times n$ square matrix $A \in \mathbb{C}^{n\times n}$ is said to be *symmetric* if $A^T = A$ and *skew-symmetric* if $A^T = -A$. Let $n = 2m$. The matrix A is said to be *Hamiltonian* if it is such that $A^T J = -JA$ where J is the matrix $\left[\begin{smallmatrix} 0 & I_m \\ -I_m & 0 \end{smallmatrix}\right]$; A is said to be *skew-Hamiltonian* if it is such that $A^T J = JA$; A is said to be *symplectic* if it is such that $A^T JA = J$.

Remark 4.1. *In general, it is possible to give two different definitions [31, 79] of complex Hamiltonian, skew-Hamiltonian and symplectic matrices, one with respect to transposition and the other one with respect to conjugate transposition. In this thesis, we are mainly interested in the former case, and therefore we always refer to the definitions given above.*

Every skew-Hamiltonian matrix can be obtained as the square of a Hamiltonian matrix, and conversely the square of a Hamiltonian matrix is always skew-Hamiltonian[32]. Symplectic matrices are exponentials of Hamiltonian matrices.

4.2 Even-dimensional skew-symmetric and skew-Hamiltonian PEPs

A skew-symmetric polynomial is a polynomial $P(x)$ whose coefficients P_j, for $j = 0, \ldots, k$, are skew-symmetric constant matrices. If the coefficients have even size $n = 2m$, we say that $P(x)$ is an even-dimensional skew-symmetric polynomial. A skew-Hamiltonian polynomial is defined as a polynomial whose coefficients P_j are all skew-Hamiltonian matrices.

Remark 4.2. *Classical eigenvalue problems for skew-Hamiltonian matrices [107] are a special case of skew-Hamiltonian PEPs.*

These two classes of polynomials are closely related, because multiplication by $J := \left(\begin{smallmatrix} 0 & I_m \\ -I_m & 0 \end{smallmatrix}\right)$ maps one class onto the other. A common feature is that the spectrum of any polynomial in these two classes contains only eigenvalues of even multiplicity. In fact, the determinant of a matrix polynomial $P(x)$ belonging to these two classes can be written as

$$p(x) = \det P(x) = q(x) \cdot q(x),$$

for a suitable polynomial $q(x)$. For the special case of a real skew-symmetric matrix pencil a proof was given in [72] where a special Kronecker form was derived. The more general case comes from classical results on determinants [90]. Let us give here a simple proof of the statement for an even-dimensional skew-symmetric complex matrix polynomial using modern terminology.

Proposition 4.1. *Let $P(x) = -P(x)^T$ be a $2m \times 2m$ skew-symmetric matrix polynomial. Then $p(x) = \det P(x) = q(x) \cdot q(x)$ for some scalar polynomial $q(x)$.*

Proof. We shall prove the proposition by induction on m. For $m = 1$ the statement is obvious.

Suppose now that any $(2m-2) \times (2m-2)$ skew-symmetric polynomial has the desired property. Let Π be a $2m \times 2m$ permutation matrix and let $Q(x) := \Pi P(x) \Pi^T$. Suppose that Π is such that

$$Q_0(x) := Q(1:2, 1:2) =: \begin{bmatrix} 0 & r(x) \\ -r(x) & 0 \end{bmatrix}$$

is nonsingular, where $r(x)$ is a suitable nonzero scalar polynomial. Notice that such an assumption can be safely made because if that was false for any Π then $P(x) = 0$ so $p(x) = 0$ and there would be nothing to prove.

Now let $Q(x) = \begin{bmatrix} Q_0(x) & A(x) \\ -A(x)^T & Q_1(x) \end{bmatrix}$, where the polynomial matrices $A(x)$ and $Q_1(x)$ have, respectively, dimensions $2 \times (2m-2)$ and $(2m-2) \times (2m-2)$; also, let $\rho(x) := r(x)^{m-1}$. Define the rational function $S(x) := Q_1(x) + \frac{1}{r(x)} A(x)^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} A(x)$. Clearly, $r(x)S(x)$ is a $(2m-2)\times(2m-2)$ skew-symmetric matrix polynomial; therefore, by the inductive hypothesis, $\det S(x) = \frac{\theta(x)^2}{\rho(x)^2}$ where $\theta(x)$ is a suitable scalar polynomial.

Moreover, $S(x)$ is the Schur complement of $Q_0(x)$. Thus, $p(x) = \frac{r(x)^2 \theta(x)^2}{\rho(x)^2}$, so $p(x)$ is the square of some scalar rational function $q(x) = \frac{\theta(x)}{r(x)^m}$. Since $p(x)$ is a polynomial, $q(x)$ must be a polynomial as well. □

The property that we have just proved is particularly useful, and it can be fully exploited by the Ehrlich-Aberth method. In fact, instead of applying the EAI to the polynomial $p(x)$ of degree $2mk$, one can apply the EAI to the polynomial $q(x)$ of degree mk even though $q(x)$ is not explicitly known.

More precisely, since $p'(x)/p(x) = 2q'(x)/q(x)$, one can compute the Newton correction $q(x)/q'(x)$ by means of

$$q(x)/q'(x) = 2p(x)/p'(x) = 2/\mathrm{tr}(P(x)^{-1}P'(x)).$$

This way, the length of the vector of the approximations y in (2.2) or in (2.3) is reduced from $2mk$ to mk, moreover, the skew-Hamiltonian or the skew-symmetric structure of the coefficients can be exploited in the computation of $P(x)^{-1}P'(x)$.

4.3 Palindromic PEPs

The polynomial $P(x)$ is called *purely palindromic* if $\mathrm{Rev}_g P(x) = P(x)$. The polynomial $P(x)$ is called *T-palindromic* if $\mathrm{Rev}_g P(x) = P(x)^T$. For instance, the scalar polynomial $x^4+2x^3+3x^2+2x+1$ is palindromic (being scalar, there is

no distinction between purely palindromicity and T-palindromicity). The grade g, with respect to which the palindromic property holds, is called the grade of palindromicity [76]; when we introduce palindromic polynomials without further specifying their grade of palindromicity, in this section and later on in the thesis we assume that $g = k$, i.e. the grade of palindromicity is equal to the degree.

It is well-known that the palindromic structure induces certain symmetries of eigenvalues and eigenvectors: in particular, if x_0 is an eigenvalue, v is a right eigenvector and w^T is a left eigenvector then:

$$\begin{cases} \text{if P is purely palindromic,} & P(\frac{1}{x_0})v = 0, \quad w^T P(\frac{1}{x_0}) = 0; \\ \text{if P is } T\text{-palindromic,} & P(\frac{1}{x_0})w = 0, \quad v^T P(\frac{1}{x_0}) = 0. \end{cases}$$

Focusing more specifically on eigenvalues, both these structures induce a symmetry $\{x, 1/x\}$ in the spectrum. If nk is odd, all eigenvalues appear in couples but a single exceptional one, which is guaranteed to be equal to -1.

Remark 4.3. *The same structure appears in the standard eigenvalue problem for a symplectic matrix [30], which can therefore be in some sense included in the palindromic PEPs class. More in general, matrices in the automorphism group associated with some bilinear form have the $\{x, x^{-1}\}$ spectral symmetry [79].*

There is a vast literature on this kind of structure; see, e.g., [71, 75, 77, 100] and the references given therein. Also, many numerical methods specifically suited to the exact preservation of the symmetry of the spectrum have been conceived. The construction of T-palindromic linearizations of palindromic eigenproblems is the subject of [25, 77], whereas numerical methods based on matrix iterations have been devised in [68, 75, 84, 100] for computing the eigenvalues of these linearizations by maintaining the palindromic structure throughout the computation.

Remark 4.4. *In principle, many other different kinds of palindromicity can be defined.*

A purely antipalindromic [2] (respectively, anti-T-palindromic) polynomial is such that $\mathrm{Rev}_{\mathrm{g}}P(x) = -P(x)$ *(respectively,* $\mathrm{Rev}_{\mathrm{g}}P(x) = -P(x)^T$*). They also have a symmetry $\{x, 1/x\}$ in the spectrum, and the methods we are going to describe in this section are readily adapted to them. For instance, the scalar polynomial $x^4 + 2x^3 - 2x - 1$ is antipalindromic ($g = k = 4$). See Subsection 4.3.2.*

In the literature, palindromicity with respect of complex conjugation or with respect to conjugation-transposition has been studied. In both these cases the spectrum has a $\{x, 1/x^*\}$ symmetry. However, the EAI is best suited to be applied to structure of the kind $\{x, f(x)\}$ where $f(x)$ is an analytic function (the reason is that otherwise the number of special eigenvalues, i.e. eigenvalues that are fixed points of f, is not a priori *determined; see Section 4.5 for further details and Chapter 6 for comments on possible ways to overcome this limitation in the near future): therefore we will focus on pure palindromic polynomials and T-palindromic polynomials as a case study.*

Notice that one can easily go further in generalising palindromicity. For instance, let $0 \neq \kappa \in \mathbb{C}$: then purely κ-palindromic (respectively, κ-T-palindromic) matrix polynomials are defined as matrix polynomials such that $\mathrm{Rev}_{\mathrm{g}}P(x) =$

$\kappa^{-g/2}P(\kappa x)$ (respectively, $\mathrm{Rev}_g P(x) = \kappa^{-g/2}P(\kappa x)^T$) where g is the grade of palindromicity the polynomial (equal to its degree when not otherwise specified). For instance, the scalar polynomial $x^4 + 2x^3 + 3x^2 - 2x + 1$ is (-1)-palindromic ($g = k = 4$) while the scalar polynomial $2\sqrt{2}x^3 + (2 - 3\sqrt{2})x^2 + (\sqrt{2} - 3)x + 1$ is $(1/2)$-palindromic ($g = k = 3$).

The coefficients of a purely κ-palindromic (respectively, κ-T-palindromic) polynomial are characterised by the property $P_j = \kappa^{g/2-j}P_{g-j}$ (respectively, $P_j = \kappa^{g/2-j}P_{g-j}^T$), where g is the grade of palindromicity, and their spectra are made of couples $\{x, \kappa/x\}$. Anti-κ-palindromic and anti-κ-T-palindromic polynomials can also be analogously defined.

Scalar κ-palindromic polynomials are particularly important because the set of the determinants of a structured PEP with a given even grade g and whose eigenvalues come all in pairs $\{x, f(x)\}$, with $f(x)$ analytic and $f(f(x)) \equiv x$, is isomorphic to the set of all the κ-palindromic polynomials of the same grade. This is proved in Theorem 5.3 in Section 5.

4.3.1 First method: a new polynomial represented in the Dickson basis

The description of this method consists in three parts. First, we will see a convenient representation of matrix polynomials in a polynomial basis different than the usual monomial basis; we will also describe what the equivalent of the companion linearization is in this setting and give some natural extension to Laurent polynomials of the theory of Chapter 1. Then, we will apply this theoretical apparatus to purely palindromic polynomials. Finally, we will deal with T-palindromic polynomials.

Representing and linearising polynomials in the Dickson basis

In [3] A. Amiraslani, R. M. Corless and P. Lancaster considered linearizations of a matrix polynomial expressed in some polynomial bases different than the usual monomial one. Equation (7) in [3] resembles closely the defining equation of $\hat{\mathcal{L}}_2$ (see Section 1.4. The authors themselves stress this analogy, that suggests an extension of the results of [59] to the case of different polynomial bases. Let $\{\phi_i\}_{i=0,\dots,k}$ be a basis for the polynomials of degree less than or equal to k. In [3] degree-graded bases that satisfy a three-terms recurrence relation (for instance, orthogonal polynomials always do so) are considered:

$$\lambda\phi_j(x) = \alpha_j\phi_{j+1}(x) + \beta_j\phi_j(x) + \gamma_j\phi_{j-1}(x). \tag{4.1}$$

The α_j are obviously linked to the leading-term coefficients of the ϕ_j. Specifically, calling c_j such coefficients, one has that $c_j = \alpha_j c_{j+1}$.

We wish to consider the expansion of the polynomial $P(x) = \sum_{j=0}^k P_j x^j$ in this basis:

$$P(x) = \sum_{j=0}^{k} A_j\phi_j(x). \tag{4.2}$$

We introduce the vector

$$\Phi := (\phi_0(x), \phi_1(x), \dots, \phi_{k-1}(x))^T.$$

By generalising the linearizations studied in [3], for each choice of Φ two new ansatz vector linearization spaces can be defined:

$$\mathcal{L}_1 := \{L = xX + Y : \exists v \in \mathbb{C}^k s.t. L \cdot (\Phi \otimes I_n) = c_{k-1} v \otimes P(x)\}; \quad (4.3)$$
$$\mathcal{L}_2 := \{L = xX + Y : \exists w \in \mathbb{C}^k s.t. (\Phi^T \otimes I_n) \cdot L = c_{k-1} w^T \otimes P(x)\}. \quad (4.4)$$

It is worth noticing that it is not strictly necessary for the new basis to be degree-graded, nor it is to satisfy a three-term recurrence relation. In fact, it is sufficient that $\{\phi_i\}_{i=0,\dots,k-1}$ are linearly independent and have degree less than or equal to $k-1$, so that there exists an invertible basis change matrix B such that $\Phi = B\Lambda$, where Λ is defined as in Section 1.4. The basis is degree-graded if and only if B is lower triangular.

In the light of the above definitions it is immediately seen that the main results of [59, 78] remain valid in the case of a more general polynomial basis. In particular the following result holds.

Proposition 4.2. *Let $L(x) \in \mathcal{L}_1$ ($\mathcal{L}_2$), and let $P(x)$ be a regular matrix polynomial. Then the following properties are equivalent:*

- *$L(x)$ is a linearization of $P(x)$;*
- *$L(x)$ is a strong linearization of $P(x)$;*
- *$L(x)$ is regular.*

Proof. It is a corollary of Proposition 1.2, that is [78, Theorem 4.3]. In fact, any $L(x) \in \mathcal{L}_1$ (resp., $\mathcal{L}_2$) can be written as $L(x) = c_{k-1}\hat{L}(x) \cdot (B^{-1} \otimes I_n)$ (resp., $L(x) = c_{k-1}(B^{-T} \otimes I_n)\hat{L}(x)$) for some $\hat{L}(x) \in \hat{\mathcal{L}}_1$ (resp., $\hat{\mathcal{L}}_2$). Therefore, $L(x)$ has each of the three properties above if and only if $\hat{L}(x)$ has the corresponding property. □

This proposition guarantees that almost every (more precisely, all but a closed nowhere dense set of measure zero) pencil in $\mathcal{L}_1$ ($\mathcal{L}_2$) is a strong linearization for P. For a proof, see [78, Theorem 4.7]. The eigenvectors of L are related to those of P. More precisely, $(x, \Phi \otimes v)$ is an eigenpair for L if and only if (x, v) is an eigenpair for P. Moreover, if L is a linearization then every eigenvector of L is of the form $\Phi \otimes x$ for some eigenvector x of P. A similar recovery property holds for the left ansatz vector linearizations. These properties can be simply proved as in Theorems 3.8 and 3.14 of [78], that demonstrate them for the special case $\Phi = \Lambda$.

For the numerical treatment of palindromic generalised eigenproblems a crucial role is played by the so-called Dickson basis [26] $\{\phi_i\}_{i\geq 0}$ defined by

$$\begin{cases} \phi_0(z) = 2 \\ \phi_1(z) = z \\ \forall j \geq 1, \quad z\phi_j(z) = \phi_{j+1}(z) + \phi_{j-1}(z). \end{cases} \quad (4.5)$$

If we consider the mapping $z := x + x^{-1}$ (which we will refer to as the Dickson transformation or the Dickson change of variable) then $x^j + x^{-j} = \phi_j(z)$ for $j = 0, 1, \dots$. For $x = e^{i\alpha}$, we obtain that $\phi_j(z) = 2\cos(j\alpha)$. From [3] by choosing e_k as the ansatz vector we find a suitable strong linearization of $P(x)$

represented as in (4.2):

$$\begin{pmatrix} I_n & & & & \\ & I_n & & & \\ & & \ddots & & \\ & & & I_n & \\ & & & & A_k \end{pmatrix} \lambda + \begin{pmatrix} 0 & -2I_n & & & & & \\ -I_n & 0 & -I_n & & & & \\ & -I_n & 0 & -I_n & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -I_n & 0 & -I_n & \\ A_0 & A_1 & \dots & A_{k-3} & A_{k-2}-A_k & A_{k-1} \end{pmatrix}. \tag{4.6}$$

Let us now see how a root-finding eigensolver can be designed for palindromic matrix polynomials via a transformation to a new polynomial. In the case of T-palindromic polynomials, the new built polynomial is skew-Hamiltonian.

Remark 4.5. *In the present subsection, we will only treat the case of even degree palindromic matrix polynomials. Notice in fact that an odd degree palindromic may always be transformed to an even degree palindromic, either by squaring the variable ($x = \xi^2$) or by multiplication by $(x+1)I_n$. Potentially, both actions may introduce problems: squaring the variable adds an additional symmetry $\{\xi, -\xi\}$ to the spectrum while multiplying by $x+1$ increases by n the multiplicity of -1 as an eigenvalue.*

However, the first issue may be solved, after passing to Laurent form, by the use of the change of variable $\zeta = (\xi+\xi^{-1})^2$. See also Remark 4.8 and Subsection 4.3.2.

Regarding the latter issue, since one knows that he is adding n times -1 there is no need to compute it: n of the $(n+1)k$ starting points of the Ehrlich-Aberth iteration shall be set equal to -2, and there they remain with no further corrections.

The following corollary describes the spectral modifications induced by the Dickson change of variable that will provide the basic link between palindromic matrix polynomials and matrix polynomials expressed in the Dickson basis. The corollary relies on the tools developed in Chapter 3. Since we are now in the setting where the underlying field is $\mathbb{C}$, we will state it in terms of Jordan structures (recall Remark 1.5).

Corollary 4.1. *Let $z(x) = x + x^{-1} = \frac{x^2+1}{x}$ and let $M(z)$ be a polynomial in z of grade g. Let $N(x) := \Phi_{g,x^2+1,x}(M(z))$ where Φ is the transformation defined in (3.2), Chapter 3, so that $N(x)$ is a polynomial in x. Let first $z_0 = x_0 + x_0^{-1}$, $x_0 \neq \pm 1$, be a finite eigenvalue of $M(z)$. Then the Jordan structure of $M(z)$ at z_0 is equal to the Jordan structure of $N(x)$ at either x_0 or $1/x_0$. If on the contrary $x_0 = \pm 1$, then there is a Jordan chain of maximal length ℓ at $M(\pm 2)$ if and only if there is a Jordan chain of maximal length 2ℓ at $N(\pm 1)$.*

Proof. The corollary describes a special case of Theorem 3.1, proved in Chapter 3. □

For technical reasons, it is now convenient to extend the notion of complete eigenstructure and of Jordan chains to Laurent polynomials.

The definitions of an eigenvalue and of a Jordan chain (see Section 1.3) can be easily extended to any matrix function $F(x) : \mathbb{C} \to \mathbb{C}^{n\times n}$. We say that x_0 is an eigenvalue of $F(x)$ if $F(x)$ is analytic in a neighbourhood of x_0 and if its determinant vanishes at $x_0 \in \mathbb{C}$, and we define Jordan chains just as in (1.3). In particular, the case of Laurent polynomials is important for our analysis. If

the principal part of a Laurent polynomial $L(x)$ is a polynomial of degree $k \geq 1$ in $1/x$, then $\hat{L}(x) = x^k L(x)$ is a polynomial. Clearly, $L(x) : \mathbb{C} \to \mathbb{C}^{n\times n}$ is a meromorphic matrix function, having a pole of order k in 0 and being analytic anywhere else.

Suppose that $0 \neq x_0 \in \mathbb{C}$ is an eigenvalue for $\hat{L}(x)$ — and, thus, for $L(x)$. The following lemma relates the Jordan chains of the Laurent polynomial $\hat{L}(x)$ and those of $L(x)$; recalling Theorem 1.5, it also immediately allows to extend the definition the complete eigenstructure of a Laurent matrix polynomial corresponding to any characteristic value $x_0 \neq 0$.

Lemma 4.1. *Let $L(x)$ be a (Laurent) polynomial and $\hat{L}(x) = x^k L(x)$ for some natural number k. Then the set $\{v_j\}$ is a Jordan chain of length $\ell + 1$ for $\hat{L}(x)$ associated with the eigenvalue $x_0 \neq 0$ if and only if $\{v_j\}$ is a Jordan chain of length $\ell + 1$ for $L(x)$ associated with the same eigenvalue.*

Proof. The proof is a straightforward application of the product differentiation rule and the definition of Jordan chains (1.3). □

The next lemma extends Theorem 1.3 to Laurent polynomials.

Lemma 4.2. *Let $P_1(x)$, $P_2(x)$ be (Laurent) polynomials and $A(x)$, $B(x)$ be two matrix functions with $P_2(x) = A(x)P_1(x)B(x)$. Suppose that an open neighbourhood Ω of $x_0 \neq 0$ exists such that all the considered functions are analytic in Ω, and also suppose that both $A(x_0)$ and $B(x_0)$ are invertible. Then x_0 is an eigenvalue for P_1 if and only if it is an eigenvalue for P_2, and $\{v_i\}$ is a Jordan chain of length $\ell+1$ for P_2 at x_0 if and only if $\{w_i\}$ is a Jordan chain of length $\ell + 1$ for P_1 at x_0, where $w_i = \sum_{j=0}^{i} \frac{B^{(j)}(\lambda_0)}{j!} v_{i-j}$.*

Proof. If $P_1(x)$ and $P_2(x)$ are classical polynomials then the thesis follows as in the proof of [44, Proposition 1.11], after having represented $A(x)$ and $B(x)$ by their Taylor series expansions. To deal with the Laurent case, let α and β be the minimal integers such that $Q_1(x) := x^\alpha P_1(x)$ and $Q_2(x) := x^\beta P_2(x)$ are classical polynomials. Just follow the previous proof for $Q_2(x) = x^{\beta-\alpha} A(x)Q_1(x)B(x)$ and apply Lemma 4.1. □

We are now going to apply our machinery to purely palindromic and T-palindromic polynomials. The former case is much easier and will be described first.

Purely palindromic polynomials

Let $\hat{P}(x) = \sum_{j=0}^{2k} P_j x^j$, $P_{2k-j} = P_j$, be a purely palindromic polynomial of even degree. It will be more convenient in the following to work with the Laurent polynomial $P(x) = x^{-k}\hat{P}(x) = P_0 + \sum_{j=1}^{k} P_j(x^j + x^{-j})$.

The most obvious way to deal with this kind of palindromicity is via introduction of the change of variable $z = x + x^{-1}$, in order to halve the degree of the polynomial. More explicitly, one can define $Q(z) := P(x(z))$, with $x(z) = (z + \sqrt{z^2 - 4})/2$ or $x(z) = (z - \sqrt{z^2 - 4})/2$, i.e., $x(z)$ is one of the two branches of the inverse function of $z(x) = x + 1/x$. Clearly, the purely palindromic structure of $P(x)$ guarantees that $Q(z)$ is itself a polynomial in the new variable z. Or in other words, using the language of Chapter 3, the purely palindromic property can be expressed as

$$\hat{P}(x) = \Phi_{k,x^2+1,x}(Q(z))$$

for some polynomial $Q(z)$.

The next proposition is a simple application of Corollary 4.1, and it relates eigenvectors and Jordan chains of the two polynomials:

Proposition 4.3. *When $x_0 \neq \pm 1$, the Jordan structure of $Q(z)$ at the eigenvalue $z_0 = x_0 + x_0^{-1}$ is equal to the Jordan structure of $P(x)$ at either x_0 or x_0^{-1}. If $x_0 = \pm 1$, $Q(z)$ has a Jordan chain of maximal length ℓ at $z_0 = \pm 2$ if and only if $P(x)$ has a Jordan chain of maximal length 2ℓ at $x_0 = \pm 1$.*

In particular, the eigenvectors of $Q(z)$ at z_0 are exactly the same of the eigenvectors of $P(x)$ at x_0 (or equivalently at x_0^{-1}, since they are the same).

Albeit very attractive, from a numerical point of view this trick is not very suitable as soon as one considers a high degree polynomial. In fact, the matrix coefficients of $Q(z)$ need to be computed as linear combinations of the ones of $P(x)$. Since the powers of a binomial are involved, the coefficients of these linear combinations would exponentially grow with the polynomial degree. To circumvent this difficulty, we shall make use of the Dickson polynomials (4.5). The polynomial $Q(z)$ is readily expressed in terms of the $\phi_j(z)$s since in the Dickson basis the coefficients are just the old ones and therefore no computation at all is needed, namely,

$$Q(z) = \frac{P_0}{2}\phi_0 + \sum_{j=1}^{k} P_j \phi_j(z). \tag{4.7}$$

The EAI can be applied to either $Q(z)$ or to the associated linearization (4.6): in both cases, the number of roots is halved with respect to the original problem.

Even if one opts to use some matrix method instead of the EAI, the associated linearization (4.6) has several computational advantages with respect to other customary linearizations of $P(x)$. Its size is nk versus $2nk$, the spectral symmetries are preserved and, moreover, the linearization displays a semiseparable structure. More precisely, it is of the form $D_0 + D_1 y$ where D_1 is identity plus low rank while D_0 is Hermitian plus low rank. This kind of structure is preserved under the QZ algorithm and it may be exploited for the design of an efficient and numerically robust root-finder applied to the algebraic equation $\det Q(z) = 0$.

T-palindromic polynomials

Consider now a T-palindromic polynomial of even degree $2k$, $\hat{P}(x)$. Let moreover $P(x) = x^{-k}\hat{P}(x)$ be the corresponding Laurent T-palindromic polynomial, and let us write

$$P(x) = \sum_{j=-k}^{k} P_j x^j. \tag{4.8}$$

The T-palindromic property is then $P_{-j} = P_j^T \ \forall j = 0, \ldots, k$.

Since the symmetry $x \leftrightarrow x^{-1}$ is still present in the spectrum, we expect that the Dickson basis may still play a role. However, unlike the purely palindromic case, it is not possible to directly express a T-palindromic polynomial as a

polynomial in the variable z. In fact, splitting $P(x)$ as the sum of its symmetric part and its skew-symmetric part we obtain that

$$P(x) = P_0 + \sum_{j=1}^{k} \left[\frac{P_j + P_j^T}{2}(x^j + x^{-j}) + \frac{P_j - P_j^T}{2}(x^j - x^{-j}) \right]. \tag{4.9}$$

If we introduce the new variables $z := x + x^{-1}$ and $w := x - x^{-1}$, then $P(x)$ can be expressed as a bivariate polynomial in w and z which is always linear in w, that is,

$$Q(z, w) = P(x(z, w)) =: B(z) + wC(z).$$

The property follows from (4.9) by substituting

$$x^j + x^{-j} = \phi_j(z),\ x^j - x^{-j} = w \left(\frac{1 + (-1)^{j+1}}{2} + \sum_{\ell=1}^{\lceil j/2 \rceil} \phi_{j-2\ell+1}(z) \right), \quad j \geq 1.$$

Notice moreover that $B(z)$ is a symmetric polynomial (that is to say, every matrix coefficient is symmetric), $C(z)$ is skew-symmetric, and the operation of transposition corresponds to changing the sign of w, that is,

$$Q^T(z, w) = P^T(x(z, w)) = B(z) - wC(z).$$

In principle one may think of treating $Q(z, w)$ with available techniques for the bivariate eigenvalue problem (see e.g. [61] and references therein), but actually z and w are not independent. They are related by the trigonometric dispersion relation $w^2 = z^2 - 4$. This suggests that it is possible to obtain a univariate polynomial by doubling the dimensions of the matrix coefficients. Let us define

$$M(z) := \begin{bmatrix} B(z) & w^2 C(z) \\ C(z) & B(z) \end{bmatrix}.$$

Then $M(z)$ is a polynomial in z of degree $k + 1$ at most. Moreover, it has the following property: if x_0 and x_0^{-1} are two distinct (i.e. $x_0 \neq \pm 1$) finite semisimple eigenvalues of $P(x)$ with multiplicity m, then $z_0 = x_0 + x_0^{-1}$ is a semisimple eigenvalue for $M(z)$ with multiplicity $2m$. To see this, notice first that

$$M(z) = \mathrm{diag}(\sqrt{w} I_n, 1/\sqrt{w} I_n) \begin{bmatrix} B(z) & wC(z) \\ wC(z) & B(z) \end{bmatrix} \mathrm{diag}(1/\sqrt{w} I_n, \sqrt{w} I_n)$$

and

$$\begin{bmatrix} B(z) & wC(z) \\ wC(z) & B(z) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} I_n & -I_n \\ I_n & I_n \end{bmatrix} \begin{bmatrix} Q(z, w) & 0 \\ 0 & Q^T(z, w) \end{bmatrix} \begin{bmatrix} I_n & I_n \\ -I_n & I_n \end{bmatrix}.$$

Hence, we find that

$$M(z) = E(w) \begin{bmatrix} Q(z, w) & 0 \\ 0 & Q^T(z, w) \end{bmatrix} E^{-1}(w),\ E(w) := \begin{bmatrix} \sqrt{\frac{w}{2}} & -\sqrt{\frac{w}{2}} \\ \sqrt{\frac{1}{2w}} & \sqrt{\frac{1}{2w}} \end{bmatrix} \otimes I_n.$$

Since, as long as the symplectic matrix function $E(w)$ is defined (that is to say $w \neq 0, \infty$ or $x \neq 0, \pm 1, \infty$), $\det(E(w)) = 1$ then

$$\det(M(z)) = [\det(Q(z, w))]^2, \quad \forall\ (z, w) \in \mathbb{C} \times \mathbb{C}. \tag{4.10}$$

Therefore, x_0 has algebraic multiplicity m for $P(x)$ if and only if z_0 has algebraic multiplicity $2m$ for $M(z)$. This gives the factorization

$$\det(M(z)) = p(z) \cdot p(z), \tag{4.11}$$

for a suitable polynomial $p(z)$ having the zero z_0 of multiplicity m. Concerning eigenvectors, if x_0 is semisimple, then let v_j (resp. b_j), $j = 1, \ldots, m$ be the eigenvectors for $P(x)$ (resp. $P^T(x)$) corresponding to x_0: it can be easily checked that $\{[w_0 v_j^T, v_j^T]^T, [-w_0 b_j^T, b_j^T]^T\}$, where $w_0 = x_0 - x_0^{-1}$, are two linearly independent eigenvectors for $M(z)$ corresponding to z_0. Thus, geometric multiplicity is also $2m$. Indeed, something more can be said in the more general case of Jordan chains. In Proposition 4.4 and further on, by the *union* of two Jordan structures (say, one made by a canonical set of n_1 Jordan chains of length $\ell_1, \ldots, \ell_{n_1}$ and the other made by a canonical set of n_2 Jordan chains of length $\mu_1, \ldots \mu_{n_2}$) we mean a Jordan structure made by a canonical set of $n_1 + n_2$ Jordan chains of length $\ell_1, \ldots, \ell_{n_1}, \mu_1, \ldots, \mu_{n_2}$.

Proposition 4.4. *Let $z_0 = x_0 + x_0^{-1}$ be an eigenvalue of $M(z)$ so that x_0 and x_0^{-1} are eigenvalues for $P(x)$. If $x_0 \neq 0, \pm 1, \infty$ then the Jordan structure of $M(z)$ at z_0 is equal to the union of the Jordan structures of $P(x)$ at x_0 and at x_0^{-1}.*

Proof. Since $P(x)$ is T-palindromic, it is clear that the Jordan structure of

$$R(x) := \begin{bmatrix} P(x) & 0 \\ 0 & P^T(x) \end{bmatrix}$$

at either x_0 or x_0^{-1} is the union of the Jordan structures of $P(x)$ at x_0 and at x_0^{-1}. Define

$$N(x) := \Phi_{k+1, x^2+1, x}(M(z)) = x^{k+1} M(z(x)).$$

The analysis of $M(z)$ given above leads to

$$M(z(x)) = E(w(x)) R(x) E^{-1}(w(x)).$$

The matrix function $E(w)$, defined in the previous page, is analytic everywhere in the w complex plane but on a branch semiline passing through the origin. Since by hypothesis $w_0 \neq 0$, the branch cut can be always chosen in such a way that $E(w)$ is analytic in a neighbourhood of $w_0 = x_0 - x_0^{-1}$, and thus $E(w(x))$ is analytic in a neighbourhood of x_0. Then we can apply Lemma 4.2 to conclude that the Jordan structures of $N(x)$ and $R(x)$ are the same. Application of Corollary 4.1 completes the proof. □

Of course, Proposition 4.4 can equivalently be interpreted (via Theorem 1.5 and Theorem 1.4) in terms of root polynomials or in terms of elementary divisors. Namely, it tells us that the elementary divisors corresponding to $z_0 \neq \pm 2, \infty$ for $M(z)$ are $(z - z_0)^{\ell_1}, (z - z_0)^{\ell_1}, \ldots, (z - z_0)^{\ell_j}, (z - z_0)^{\ell_j}$ if and only if the elementary divisors corresponding to $x_0 \neq \pm 1, 0, \infty$ for $P(x)$ are $(x - x_0)^{\ell_1}, \ldots, (x - x_0)^{\ell_j}$. Analogous considerations can be made for root polynomials.

Remark 4.6. *Another remarkable property of $M(z)$ is that its coefficients are all skew-Hamiltonian, that is to say they can be written as JK where $J = \left[\begin{smallmatrix} 0 & I \\ -I & 0 \end{smallmatrix}\right]$*

and K is some skew-symmetric matrix. This consideration immediately explains the reason why $M(z)$ has all eigenvalues of even multiplicity.

This link between T-palindromic and skew-Hamiltonian polynomials is interesting because it may shed more light on the relation between several polynomial structures. It is known that one can easily transform a palindromic polynomial to an even polynomial by a Cayley transformation, and then to a Hermitian polynomial via a multiplication by i (if one started from a real polynomial) or to a symmetric polynomial by squaring the matrix coefficients. On the other hand, Hamiltonian polynomials can lead to skew-Hamiltonian polynomials by squaring each coefficients, and multiplication by J sends a skew-Hamiltonian polynomial to a skew-symmetric polynomial. The Dickson change of variable, followed by doubling the dimension, is able to map T-palindromic polynomials of even degree to a subset of skew-Hamiltonian polynomials. Unlike some of the other mentioned maps, this is not a bijection between two classes of structured polynomials, because what is obtained is actually a subset of skew-Hamiltonian polynomials. In fact, since the north-west and south-east coefficients of $M(z)$ are the coefficients of $B(z)$ they must be symmetric and there is a relation between the north-east and south-west coefficients of $M(z)$. However, a deeper investigation on this subject is needed in the future.

Equations (4.10) and (4.11) enable the computation of the eigenvalues of $P(x)$ to be reduced to solving algebraic equations. From Proposition 4.4 it follows that possible discrepancies in the Jordan structures can be expected for $z_0 = \pm 2$ and $z_0 = \infty$ corresponding to $x_0 = \pm 1$ and $x_0 = 0, \infty$, respectively.

When $x_0 = \pm 1$ not only the proof we gave is not valid (because, since $w_0 = 0$ is a branch point, there is no neighbourhood of analyticity of the matrix function E), but in fact the proposition itself does not hold. As a counterexample, let $a \neq \pm \frac{i}{\sqrt{2}}$ and consider the polynomial

$$P(x) = \begin{bmatrix} x - 2 + x^{-1} & ax - ax^{-1} \\ -ax + ax^{-1} & x + x^{-1} \end{bmatrix}.$$

We have that $\{[1,0]^T, [0,a]^T\}$ is a Jordan chain for $P(x)$ at $x = 1$. The corresponding $M(z)$ is

$$M(z) = \begin{bmatrix} z-2 & 0 & 0 & az^2 - 4a \\ 0 & z & 4a - az^2 & 0 \\ 0 & a & z-2 & 0 \\ -a & 0 & 0 & z \end{bmatrix},$$

which has a semisimple eigenvalue at $z = 2$ with the corresponding eigenvectors $[0,0,1,0]^T$ and $[2,0,0,a]^T$.

If the leading coefficient of $P(x)$ is not symmetric (if it is symmetric, the degree of $M(z)$ drops), then $M(z)$ has $2n$ extra infinite eigenvalues, where n is the dimension of the matrix coefficients of $P(x)$. These eigenvalues are defective since their geometric multiplicity is only $n + \dim\ker C_{k-1}$, where C_{k-1} is the leading coefficient of $C(z)$. Notice that these informations on the infinite eigenvalues are known theoretically and, therefore, can be exploited when designing an EAI specifically suited for $M(z)$.

For the numerical approximation of the roots of $p(z)$ we can exploit again the properties of the Dickson basis to compute the matrix coefficients of $M(z) =$

$\sum_{j=0}^{k+1} M_j \phi_j(z)$. The code below computes the matrices $M_j \in \mathbb{C}^{2n\times 2n}$, $0 \le j \le k-1$, given in input the coefficients P_j of $P(x)$, $0 \le j \le k$, defined as in (4.8).

function Dickson_transform
Input: $P_0, \ldots, P_k \in \mathbb{C}^{n\times n}$
Output: $M_0, \ldots, M_{k+1} \in \mathbb{C}^{2n\times 2n}$

$B_0 = P_0/2$; $\hat{C}_0 = 0_n$;
for $j = 1, \ldots, k$
  $B_j = (P_j + P_j^T)/2$; $\hat{C}_j = (P_j - P_j^T)/2$;
end
$S_0 = 0_n$, $S_1 = 0_n$;
for $j = k, \ldots, 1$
  $S_{\mathrm{mod}(j,2)} = S_{\mathrm{mod}(j,2)} + \hat{C}_j$
  $C_{j-1} = S_{\mathrm{mod}(j,2)}$;
end
$C_0 = C_0/2$; $C_k = C_{k+1} = 0_n$; $\tilde{C}_0 = C_2$
$\tilde{C}_1 = C_1 + C_3$; $\tilde{C}_2 = 2C_0 + C_4$;
for $j = 4, \ldots, k$
  $\tilde{C}_{j-1} = C_{j-3} + C_{j+1}$;
end
$\tilde{C}_k = C_{k-2}$; $\tilde{C}_{k+1} = C_{k-1}$;
for $j = 1{:}\, k+2$
  $\tilde{C}_{j-1} = \tilde{C}_{j-1} - 2C_{j-1}$;
  $M_{j-1} = [B_{j-1}, \tilde{C}_{j-1}; C_{j-1}, B_{j-1}]$;
end

Remark 4.7. *The coefficients of $C(z)$ are linear combinations of $P_j - P_j^T$. As can be seen by the above algorithm, the coefficients of such combinations expressed in the Dickson basis remain bounded, the upper bound being $1/2$. An analogous result, with upper bound 1, holds for $w^2 C(z)$. This is in contrast with the exponential growth that would have been seen in the purely palindromic case if one had directly applied the Dickson transformation without the use of the Dickson basis.*

The arithmetic cost is $\mathcal{O}(n^2 k)$ operations. Once the coefficients M_j are determined, two possibilities exist in order to devise a structured adaptation of the EAI. The first is to apply the EAI to $M(z)$ following Section 2.2.2. The second is to follow Section 2.2.1: a linearization of $M(z)$ of the form (4.6) can be constructed. The properties of this linearization are investigated in the following in order to devise an adaptation of the EAI in order to approximate the roots of $p(z)$ defined by (4.11).

Very few things change when using (4.6) instead of (1.7); it is very easy to adapt the proof given in Section 2.2.1 to the present case. Namely, let $\mathcal{G}(\theta, \psi)$ the 2×2 unitary Givens rotation given by

$$\mathcal{G}(\theta, \psi) = \begin{pmatrix} \theta & \psi \\ -\bar{\psi} & \bar{\theta} \end{pmatrix}, \quad |\theta|^2 + |\psi|^2 = 1.$$

Let $L(z) = \tilde{L} \cdot Q$ be the block LQ factorization of $L(z)$, where $L(z)$ is a linearization of $M(z)$ in the form (4.6). Such a factorization is obtained by means

of Givens rotations so that

$$\begin{array}{l} L(z)\mathcal{G}_1 \cdot \mathcal{G}_2 \cdots \mathcal{G}_k = \tilde{L}, \quad Q^H = \mathcal{G}_1 \cdot \mathcal{G}_2 \cdots \mathcal{G}_k, \\ \mathcal{G}_j = I_{2n(j-1)} \oplus (\mathcal{G}(\theta_j, \psi_j) \otimes I_{2n}) \oplus I_{2n(k-j)}. \end{array} \tag{4.12}$$

The lower triangular factor $\tilde{L}$ is in the form

$$\tilde{L} = \begin{pmatrix} \alpha_1 I_{2n} & & & & & \\ \beta_1 I_{2n} & \alpha_2 I_{2n} & & & & \\ \gamma_1 I_{2n} & \beta_2 I_{2n} & \ddots & & & \\ & \ddots & \ddots & \ddots & & \\ & & \gamma_{k-2} I_{2n} & \beta_{k-1} I_{2n} & \alpha_k I_{2n} & \\ \hat{M}_0 & \hat{M}_1 & \dots & \hat{M}_{k-2} & \hat{M}_{k-1} & \hat{M}_k \end{pmatrix},$$

where $\alpha_j \neq 0$, $1 \leq j \leq k$. If $\hat{M}_k$ is invertible, then also $L(z)$ is invertible, and $L^{-1}(z) = Q^H \cdot \tilde{L}^{-1}$. Let now $\tilde{M}_{k+1} = \hat{M}_k^{-1} \cdot M_{k+1}$. Analogously to the unstructured case, we can prove the following.

Proposition 4.5. *There exist matrices $\tilde{M}_1, \dots, \tilde{M}_k \in \mathbb{C}^{2n \times 2n}$ such that*

$$L^{-1}(z)E = \begin{pmatrix} \tilde{M}_1 & \psi_1 \tilde{M}_2 & \dots & \psi_1 \cdots \psi_k \tilde{M}_{k+1} \\ & \bar{\theta}_1 \tilde{M}_2 & & \\ & & \ddots & \\ & & & \bar{\theta}_k \tilde{M}_{k+1} \end{pmatrix},$$

where the blank entries are not specified.

Following the same procedure that we have already described in Section 2.2.1, we arrive to the following formula for the Newton correction:

$$\det(M(z))' / \det(M(z)) = \operatorname{tr}(\hat{M}_k^{-1}(\tilde{X}_1 + \bar{\theta}_1 \tilde{X}_2 + \ldots + \bar{\theta}_{k-1} \tilde{X}_k + \bar{\theta}_k M_{k+1})).$$

Once the eigenvalues of $M(z)$ are approximated by the EAI, the back transformation $x = x(z)$ is used to obtain the sought eigenvalues of $P(x)$. It is important to point out that the application $z \to x = (z \pm \sqrt{z^2 - 4})/2$ is ill conditioned at $z = \pm 2$. Therefore, loss of accuracy is expected near $x = \pm 1$. In this case, a refinement step is advisable. Such a refinement may be implemented by an unstructured version of the EAI, by the naive structured EAI, or by other structured refinement methods (see Chapter 5).

Remark 4.8. *Notice that a similar technique can be applied to* even/odd *matrix polynomials, that is polynomials whose coefficients alternate between symmetric and skew-symmetric matrices. In this case, on can apply the transformation $z = x^2$ and use algebraic manipulations, akin to the ones described for the T-palindromic case, in order to build a new polynomial in z with double dimensions. More details will be given in Section 4.4.*

4.3.2 Second method: the implicit Dickson transformation

In the following, we will show how to avoid the explicit use of the Dickson transformation. This has the advantage that there is no potential loss of accuracy for

very small/large eigenvalues of $P(x)$, unlike the case of the method previously described, where the construction of the new matrix polynomial $M(z)$ introduces unwanted defective eigenvalues at infinity which might create numerical problems (in any instance, such possible losses of accuracy can be amended by a structured refinement method: see Chapter 5). Let $P(x)$ be either a purely palindromic or a T-palindromic polynomial of dimension n and degree k, and let $p(x)$ be its determinant. We will once again rely on the Dickson transformation $z = x + x^{-1}$.

If nk is even, then by means of simple formal manipulations one may show that $q(z) := x(z)^{-nk/2} \cdot p(x(z))$ is a polynomial in z, where once again either $x(z) = (z + \sqrt{z^2 - 4})/2$ or $x(z) = (z - \sqrt{z^2 - 4})/2$.

Moreover, taking derivatives in the latter equation leads to an explicit expression for the Newton correction $q(z)/q'(z)$ given in terms of $p(x)/p'(x)$:

$$\frac{q(z)}{q'(z)} = \frac{1 - 1/x^2}{p'(x)/p(x) - nk/(2x)}, \quad p'(x)/p(x) = \mathrm{tr}(P(x)^{-1}P'(x)). \tag{4.13}$$

This equation enables one to apply the EAI to the polynomial $q(z)$ by working directly on $P(x)$. In fact, $p(x)/p'(x)$ is computed as always be means of the Jacobi formula, and then (4.13) is applied. In other words, the explicit computation of a matrix polynomial $M(z)$ whose determinant is (the square of) $q(z)$ is bypassed with this trick.

Once the roots of $q(z)$, $z_1, \ldots, z_{nk/2}$ have been computed, the eigenvalues of $P(x)$ are given by the pairs $\{x_i, 1/x_i\}$ which are the roots of the quadratic polynomial $x^2 - z_i x + 1$. As the method of the explicit construction of $M(z)$, also this approach has the advantage to work with an approximation vector of half the size and to deliver the solution as pairs $\{x, 1/x\}$.

If nk is odd, then -1 is necessarily an eigenvalue of the palindromic PEP and there is no need to approximate it. To calculate approximations of the remaining $nk-1$ eigenvalues, there are two possible strategies. We have already mentioned them in Remark 4.5, but let us consider them again in view of the implicit method.

As a first possibility, one may consider the new matrix polynomial $R(x) = (x+1)P(x)$ which has even degree. The eigenvalues of $R(x)$ are those of $P(x)$. The multiplicities are unchanged for every eigenvalue but -1, whose algebraic multiplicity is increased by n. Therefore, the previously described technique can be applied. Only $nk-1$ roots of $\det(R(x))$ are needed, because $n+1$ roots are a priori known to be equal to -1. Thus, one could apply the EAI (2.2) or (2.3) with an approximation vector y of $n(k+1)$ components of which $n+1$ are set equal to -1 in order to immediately achieve implicit deflation of the roots; or, working in the variable z in order to extract the structure, the SEAI (4.13) can be used setting $(n+1)/2$ starting points equal to -2.

A second possibility is to set $x := w^2$ and to consider the eigenvalues of the polynomial $Q(w) = P(x(w))$. The scalar polynomial $q(w) := \det Q(w)$ has $2nk$ roots, which are the square roots of the solutions of the original equation $p(x) = 0$ that we have to solve. In particular only $2nk - 2$ roots are to be determined, since $q(w) = 0$ has two known solution at $w = \pm i$. It is useful to set $z := (w + 1/w)^2 = x + 1/x + 2$. Defining

$$\widetilde{q}(w) := \frac{q(w)}{w^{nk+1} + w^{nk-1}},$$

it is easy to check that $r(z) := \widetilde{q}(w(z))$ is a polynomial in z. Therefore we may restrict the attention to computing the roots of $r(z)$. Once they have been computed, the evaluation of the function $w(z)$ at these roots provides the roots of $q(w)$. The evaluation of $x(w)$ at these latter roots yields the sought eigenvalues of $P(x)$. In order to compute the roots of $r(z)$ we may apply the EAI to the polynomial $r(z)$. The following equations provides a tool to compute the Newton correction $r(z)/r'(z)$ needed by the EAI.

$$\frac{r(z)}{r'(z)} = \frac{2w(1-1/w^4)}{q'(w)/q(w) - [(nk+1)w^2 + nk - 1]/(w^3+w)},$$

or in terms of the original variable x

$$\frac{r(z)}{r'(z)} = \frac{1-1/x^2}{p'(x)/p(x) - [(nk+1)x + nk - 1]/(2x^2+2x)}.$$

At the moment we have no clear elements to say which of the two possibilities is more convenient. We plan to investigate in this direction.

We conclude this subsection mentioning that also purely antipalindromic and anti-T-palindromic polynomials have an $\{x, 1/x\}$ symmetry. Their determinants are palindromic if n is even and antipalindromic if n is odd [76]. The former case is exactly the same as above. The latter case is also easy, because a scalar antipalindromic polynomial is always equal to $x-1$ times a scalar palindromic polynomial. Moreover, it is possible to prove [76] that 1 is always a root of a scalar antipalindromic polynomial, and -1 is always a root of a scalar even-grade antipalindromic polynomial, so according to the grade there are either one or two exceptional eigenvalues with odd multiplicity. Therefore, it is easy to extend our technique to this class.

4.4 Hamiltonian/skew-Hamiltonian PEPs, even PEPs and odd PEPs

An even (odd) polynomial $P(x) = \sum_{j=0}^{k} P_j x^j$ is such that P_j is symmetric for all even (odd) values of j and is skew-symmetric for all odd (even) j. Similarly, the coefficients of a Hamiltonian/skew-Hamiltonian polynomial are, alternatively, Hamiltonian and skew-Hamiltonian matrices. The classes of even-dimensional even/odd polynomials are easily mapped onto the classes of Hamiltonian/skew Hamiltonian polynomials by a multiplication by J. Amongst the huge literature on these classes of polynomials see, for instance, [83, 86, 87, 100] and the references therein.

The matrix polynomials belonging to these classes have eigenvalues coming in pairs $\{x, -x\}$. In particular, if nk is odd, then either $x = 0$ (if P_0 is skew-symmetric) or $x = \infty$ (if P_k is skew-symmetric) is necessarily an eigenvalue. Notice that nk cannot be odd for Hamiltonian/skew-Hamiltonian polynomials, because they are only defined for even n.

Remark 4.9. *Classical eigenvalue problems for Hamiltonian matrices [107] are a special case of Hamiltonian/skew-Hamiltonian PEPs. More in general, matrices in the Lie algebra associated with some bilinear form have the spectral structure $\{x, -x\}$ [79].*

Let $z := x^2$. Just like the T-palindromic case, also for even/odd polynomials it is possible to follow the ideas exposed in Subsection 4.3.1 and build a new matrix polynomial $M(z)$ whose determinant is equal to $p(x(z)) \cdot p(x(z))$. The following result demonstrates the way it can be done for an even polynomial.

Proposition 4.6. *Let $P(x)$ be an $n \times n$ even matrix polynomial of degree k, and let $z = x^2$. Define $B(z) := \frac{P(x(z))+P^T(x(z))}{2}$ and $C(z) := \frac{1}{x(z)} \frac{P(x(z))-P^T(x(z))}{2}$, so that $P(x) = B(x^2)+xC(x^2)$. Then $M(z) := \left[\begin{smallmatrix} B(z) & zC(z) \\ C(z) & B(z) \end{smallmatrix}\right]$ is a matrix polynomial such that $\det M(z) = [p(x(z))]^2$. If $0 \neq x_0 \in \mathbb{C}$ is an eigenvalue for $P(x)$ associated with a canonical set of Jordan chains of length $\ell_1, \ldots, \ell_k$ then x_0^2 is an eigenvalue for $M(z)$ and its Jordan structure is the union of the Jordan structures of $P(x)$ at x_0 and at $-x_0$.*

Moreover:

1. *concerning eigenvectors associated with any finite nonzero eigenvalue x_0, $P(x_0)v_0 = 0$ and $P(-x_0)w_0 = 0$ if and only if $v_1 = [x_0 v_0^T, v_0^T]^T$ and $w_1 = [-x_0 w_0^T, w_0^T]^T$ are two linearly independent eigenvectors such that $M(x_0^2)v_1 = 0 = M(x_0^2)w_1$;*

2. *$M(z)$ is a $2n \times 2n$ matrix polynomial of degree $\deg M = [(k+1)/2]$, i.e. the integer part of $(k+1)/2$;*

3. *writing $M(z) = \sum_{j=0}^{\deg M} M_j z^j$, the relation $M_j = \left[\begin{smallmatrix} P_{2j} & P_{2j-1} \\ P_{2j+1} & P_{2j} \end{smallmatrix}\right]$ holds for all $0 \leq j \leq \deg M$, where $P_j = 0$ if $j < 0$ or $j > k$;*

4. *if k is odd and n is even (respectively, and n is odd), $M(z)$ has at least n (respectively, $n+1$) eigenvalues at infinity.*

Proof. The proof can be obtained by adapting the arguments used in Subsection 4.3.1 for T-palindromic polynomials to the even case, which is in fact simpler.

More explicitly, let $Q(z) := P(x(z)) = B(z) + \sqrt{z}C(z)$, so that $Q^T(z) = B(z) - \sqrt{z}C(z)$. We get

$$M(z) = E(x) \left[\begin{array}{cc} Q(z) & 0 \\ 0 & Q^T(z) \end{array}\right] E^{-1}(x), \quad E(x) := \left[\begin{array}{cc} \sqrt{\frac{x}{2}} & -\sqrt{\frac{x}{2}} \\ \sqrt{\frac{1}{2x}} & \sqrt{\frac{1}{2x}} \end{array}\right] \otimes I_n.$$

Just like in Proposition 4.4, $E(x)$ is a symplectic unimodular matrix function and it is analytic everywhere but on a branch semiline passing through the origin of the x complex plane. The first part of the Proposition can therefore be proved following the same kind of arguments of Proposition 4.4.

If k is even, $B(z)$ has degree $k/2$ and $C(z)$ has degree $k/2-1$, so $M(z)$ has degree $k/2 = [(k+1)/2]$. If k is odd, both $B(z)$ and $C(z)$ have degree $(k-1)/2$, so $M(z)$ has degree $(k+1)/2 = [(k+1)/2]$.

The statement on the explicit form of M_j is easily checked by direct inspection, and so is the statement on eigenvectors.

Finally, if k is odd $M_{\deg M} = \left[\begin{smallmatrix} 0 & P_k \\ 0 & 0 \end{smallmatrix}\right]$, so its rank is at most $n - \frac{(-1)^{n+1}+1}{2}$ (odd-dimensional skew-symmetric matrices are always rank deficient), which completes the proof. □

Remark 4.10. *Once again, $M(z)$ is a skew-Hamiltonian polynomial, which is coherent with having all double eigenvalues. Just as in the T-palindromic case the eigenvalues ± 2 were exceptional, also in this case Proposition 4.6 does not hold for the special eigenvalue 0. As a simple example consider the scalar even polynomial $p(x) = x^2$ which has a Jordan chain of length 2 at 0; one gets $M(z) = I_2 z$, and $M(z)$ has a semisimple double eigenvalue at $0 = 0^2$.*

Similar results can of course be obtained for odd and Hamiltonian/skew-Hamiltonian matrix polynomials: for all such classes a new matrix polynomial, related to the original polynomial eigenvalue problem, can always be built. We give in the following proposition the result for odd matrix polynomials, omitting the proof which is analogous to the even case. The results for Hamiltonian/skew-Hamiltonian matrix polynomials are easily obtained as corollaries, by noticing that a Hamiltonian (respectively, skew-Hamiltonian) matrix can be obtained multiplying by J a symmetric (respectively, skew-symmetric) matrix of even dimension.

Proposition 4.7. *Let $P(x)$ be an $n \times n$ odd matrix polynomial of degree k, and let $z = x^2$. Define $B(z) := \frac{1}{x(z)} \frac{P(x(z)) + P^T(x(z))}{2}$ and $C(z) := \frac{P(x(z)) - P^T(x(z))}{2}$, so that $P(x) = xB(x^2) + C(x^2)$. Then $M(z) := \begin{bmatrix} zB(z) & C(z) \\ C(z) & B(z) \end{bmatrix}$ is a matrix polynomial such that $\det M(z) = [p(x(z))]^2$. If $0 \neq x_0 \in \mathbb{C}$ is an eigenvalue for $P(x)$ associated with a canonical set of Jordan chains of length $\ell_1, \ldots, \ell_k$ then x_0^2 is an eigenvalue for $M(z)$ and its Jordan structure is the union of the Jordan structures of $P(x)$ at x_0 and at $-x_0$.*

Moreover:

1. *concerning eigenvectors associated with any finite nonzero eigenvalue x_0, $P(x_0)v_0 = 0$ and $P(-x_0)w_0 = 0$ if and only if $v_1 = [v_0^T, x_0 v_0^T]^T$ and $w_1 = [w_0^T, -x_0 w_0^T]^T$ are two linearly independent eigenvectors such that $M(x_0^2)v_1 = 0 = M(x_0^2)w_1$;*

2. *$M(z)$ is a $2n \times 2n$ matrix polynomial of degree $\deg M = [(k+1)/2]$, i.e. the integer part of $(k+1)/2$;*

3. *writing $M(z) = \sum_{j=0}^{\deg M} M_j z^j$, the relation $M_j = \begin{bmatrix} P_{2j-1} & P_{2j} \\ P_{2j} & P_{2j+1} \end{bmatrix}$ holds for all $0 \leq j \leq \deg M$, where $P_j = 0$ if $j < 0$ or $j > k$;*

4. *if k is odd $M(z)$ has at least n eigenvalues at infinity; if k is even and n is odd, $M(z)$ has at least 1 eigenvalue at infinity.*

Applying the EAI to $M(z)$ allows us to extract the spectral structure. An alternative approach, that avoids possible issues about loss of accuracy for very large eigenvalues (this time $M(z)$ has extra infinite eigenvalues only if k is odd), is once again the implicit use of the squaring transformation. Namely, if there are no uncoupled eigenvalues (e.g. an even polynomial with nk even), then defining $z := x^2$ one finds that $q(z) := p(x(z))$ is a polynomial for $x(z) = \sqrt{z}$ or $x(z) = -\sqrt{z}$. Thus, $p'(x)/(2xp(x)) = q'(z)/q(z)$, so that the Newton correction for the polynomial $q(z)$ is readily available

$$q(z)/q'(z) = 2xp(x)/p'(x) = 2x/\mathrm{tr}(P(x)^{-1}P'(x))$$

and the Ehrlich-Aberth algorithm can be implicitly applied to the polynomial $q(z)$ in order to compute its roots $z_1, \ldots, z_{nk/2}$. This way, the roots of $p(x)$ are readily available in pairs as $(\sqrt{z_i}, -\sqrt{z_i})$.

In other situations, one eigenvalue is necessarily either 0 (if $P(x)$ is odd and nk is odd) or ∞ (if $P(x)$ is even and nk is odd); thus, there is no need to approximate it. It may also happen that there is one uncoupled eigenvalue at 0 and one at ∞ (e.g. if $P(x)$ is odd, n is odd and k is even). In the case of an extra eigenvalue at 0, to approximate the other eigenvalues one can notice that $q(z) := p(\sqrt{z})/\sqrt{z}$ is a polynomial and that $q'(z)/q(z) = (1/2x^2)(xp'(x)/p(x) - 1)$. This yields the Newton correction for $q(z)$ as

$$q(z)/q'(z) = 2x/(p'(x)/p(x) - 1/x) = 2x/(\mathrm{tr}(P(x)^{-1}P'(x)) - 1/x), \quad z = x^2,$$

which enables one to apply the EAI to $q(z)$ by using an approximation vector of length $(nk-1)/2$. As in the palindromic case, there is also the alternative option to consider the polynomial $xP(x)$ which is even (odd) if $P(x)$ is odd (even). The new polynomial $xP(x)$ has n additional eigenvalues at 0 that are known and can therefore be immediately deflated.

For the function $x = \sqrt{z}$, the relative condition number is finite near the exceptional point $z = 0$, even though the absolute condition number diverges. In our experience, small eigenvalues were accurately approximated (see also Section 4.6). Should difficulties arise in future experiments, we underline that one may apply a refinement to the exceptional eigenvalues by means of the unstructured EAI or of structured refinement methods: see [41] and Chapter 5.

4.5 A unified treatment of a whole class of structured PEPs

More in general, let $\mathbb{C}^* := \mathbb{C} \cup \{\infty\}$ and let $f : \mathbb{C}^* \to \mathbb{C}^*$ be any self-inverse function, that is $f(f(x)) = x \ \ \forall x \in \mathbb{C}^*$. An example is the subclass of rational functions $f(x) = \frac{ax+b}{cx-a}$, which are self-inverse whenever $a^2 + bc \neq 0$. If we additionally require f to be analytic, having such a form is not only a sufficient condition, but it is also necessary (unless $f(x) = x$) for f to be self-inverse. This follows from the fact that nondegenerate Möbius functions (i.e., rational functions of degree 1 such that the denominator and the numerator are nonzero coprimes) are the only automorphisms of $\mathbb{C}^*$.

Suppose that, because of some structure in the coefficients of $P(x)$, all eigenvalues come in pairs $\{\lambda, f(\lambda)\}$. Eigenvalues such that $\lambda = f(\lambda)$ are called *exceptional*, and are allowed to appear with any multiplicity.

We refer to a matrix polynomial $P(x)$ with the above properties as to an *f-structured* matrix polynomial.

Given a function $f(x)$ and a class of f-structured matrix polynomials, we assume that the exceptional eigenvalues of odd multiplicity, if any, are known. In this way it is possible to collect the unknown eigenvalues into pairs. In practice, this property holds for the most frequently encountered structures associated with an analytic self-inverse function $f(x)$.

On the contrary, for non analytic $f(x)$ the property is often unsatisfied. Consider, for instance, the set of matrix polynomial with real cofficients, which is a subset of f-structured matrix polynomials with $f(x) = x^*$. The set of

exceptional eigenvalues is the whole real line and, in general, there is no way to predict *a priori* which real eigenvalues, if any, appear with odd multiplicity.

If f is analytic, the implicit change of variable method that we have described for the special cases $f(x) = -x$ and $f(x) = 1/x$ can be generalised thanks to the following proposition.

Proposition 4.8. *Let $P(x)$ be an f-structured matrix polynomial with $f(x) = \frac{ax+b}{cx-a}$, $a^2+bc \neq 0$. Denote the eigenvalues of $P(x)$ by x_i, $1 \leq i \leq ng$ and assume that there are no eigenvalues with odd multiplicity.*

Suppose that $a \neq 0$, and define $z(x) := \frac{ax^2+bx}{cx-a} = xf(x)$. If $x(z)$ denotes any of the two branches of the inverse function of $z(x)$, then $q(z) := \frac{p(x(z))}{(cx(z)-a)^{nk/2}}$ is a polynomial.

If $a = 0$, let $z(x) := \frac{cx^2+b}{cx} = x + f(x)$ and let $x(z)$ be any branch of the inverse funcion of $z(x)$. Then $q(z) := \frac{p(x(z))}{x(z)^{nk/2}}$ is a polynomial

Proof. Assume first that $a \neq 0$.

Since there are no eigenvalues with odd multiplicity, ng must be even. Assume first that the eigenvalues x_i are such that $x_i \neq a/c$ for any i. Observe that the polynomial $p(x)$ can be written as $p(x) = \prod_{i=1}^{ng/2}(x - x_i)(x - f(x_i))$. From the definition of $f(x)$ we have $x - f(x_i) = (f(x) - x_i)(cx - a)/(a - cx_i)$. This way, each quadratic factor takes the form $(xf(x) + x_i^2 - x_i(x + f(x))(cx - a)/(a - cx_i)$. Now, we exploit the property that $cxf(x) - a(x + f(x)) = b$ and we conclude that $p(x)$ is the product of $ng/2$ factors of the form $(xf(x) + x_i^2 + \frac{b}{a}x_i - \frac{c}{a}x_i xf(x))(cx - a)/(a - cx_i) = (z - x_i f(x_i))(cx - a)/a$, where we used the fact $z = xf(x)$.

In the instance $x_i = a/c$, notice that $\infty = f(a/c)$ is also a root. In other words, the grade of $p(x)$ is greater than its degree. Therefore, it holds that $p(x) = (cx - a)^\mu \prod_{j \in \mathcal{J}}(x - x_j)(x - f(x_j))$, where $\mathcal{J}$ is the set of indices j such that $x_j \neq a/c, \neq f(x_j)$, and μ is the multiplicity of $x_i = a/c$. The proof is completed noticing that the cardinality of $\mathcal{J}$ is $ng/2 - \mu$.

If on the contrary $a = 0$, let $z(x) := \frac{cx^2+b}{cx} = x + f(x)$. Assume first that $x_i \neq 0, \infty$, and write $p(x) = \prod_{i=1}^{ng/2}(x - x_i)(x - f(x_i)) = \prod_{i=1}^{ng/2}(xf(x) + x_i^2 - x_i(x + f(x))(cx)/(-cx_i)$. Since $xf(x) = b/c$ and $z = x + f(x)$, $p(x)$ is the product of $ng/2$ factors of the form $x(b + cx_i^2 - x_i z)(-cx_i)^{-1}$, which concludes the proof.

If there are some zero and infinite eigenvalues, then $p(x) = x^\mu \prod_{j \in \mathcal{J}}(x - x_j)(x - f(x_j))$, where $\mathcal{J}$ is the set of indices j such that $x_j \neq 0, \infty$ while μ is the multiplicity of 0 as a root; the proof is completed because $|\mathcal{J}| = ng/2 - \mu$. □

Therefore, the EAI can be applied to $q(z)$ to find its roots z_i; then, the eigenvalues of $P(x)$ can be found solving the equations $z(x) = z_i$, $i = 1, \ldots, nk$, for the variable x. If there are eigenvalues with odd multiplicity (e.g. if nk is odd) then there is only a slight complication: there must be some exceptional eigenvalues that can be treated with techniques akin to those described for the special cases considered in the previous sections of this chapter.

Notice that the fixed points of $f(x)$ may lead to computational problems, since they are double roots in the equation $z(x) = \zeta$. Refinements of some kind are advisable there. See also [41] and Chapter 5.

The method of the explicit construction of a new skew-Hamiltonian matrix polynomial whose eigenvalues are related to the sought ones may also be

extended to the general case. This is the subject of a future research project.

4.6 Numerical experiments

As the regular EAI, also the SEAI underwent several tests in order to check its efficiency and its accuracy.

4.6.1 Efficiency of the SEAI

In Section 2.6, when numerical experiments on the unstructured EAI were presented, the importance of testing the number of scalar iterations versus the number of sought eigenvalues was made clear.

Our structured variants of the EAI take advantage by halving the number of eigenvalues; nevertheless, it is still important to verify whether the number of scalar iterations grows linearly with nk, where n is the size of the matrix polynomial and k is its degree.

Let us first introduce the class $\mathrm{H}_{n,k}$ of T-palindromic (Laurent) polynomials. The polynomials are constructed according to the following rules:

$$\begin{array}{l} \mathrm{H}_{n,k} = \sum_{j=-k}^{k} A_j x^j, \quad A_j \in \mathbb{R}^{n\times n}, \\ A_0 = 0_n;\ A_j = I_n + \boldsymbol{e}_n \boldsymbol{e}_1^T,\ A_{-j} = A_j^T,\ 1 \le j \le k. \end{array}$$

From

$$h(x) = \sum_{j=1}^{k} x^j + \sum_{j=1}^{k} x^{-j} = \frac{x^k - 1}{x - 1} \frac{x^{k+1} + 1}{x^k},$$

we find that most of the eigenvalues lie on the unit circle and for k even $x = -1$ is a double root of $h(x)$.

Figure 22 describes the convergence history for the SEAI applied to $H_{5,20}$ with starting values equally spaced on the circle centred in the origin with radius 4. The curves represented are generated by plotting the sequences $\{z_j^{(i)}\}$, $1 \le i \le maxit$, for $j = 1, \ldots, nk$, where $maxit$ is a fixed number of iterations.

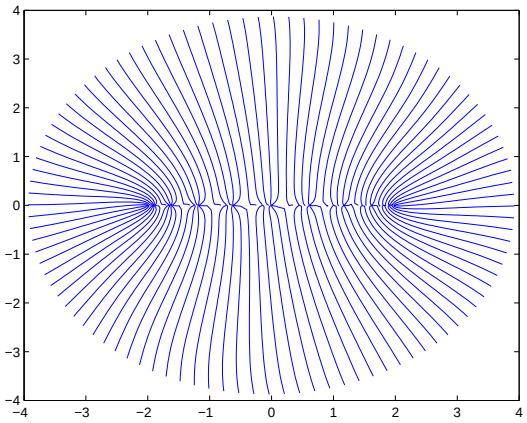

Fig. 22. *History of the convergence for the H problem with* $n = 5$ *and* $k = 20$

As happened for the unstructured EAI (UEAI), with starting points evenly distributed on the unit circle the convergence is quite regular. Once again the drawback is that with this choice of starting approximations we have observed that the number of vector iterations is typically of order of nk but there are not enough early deflations, that is, iterations that are prematurely stopped due to early convergence. This leads to a superlinear growth of t with respect to nk. In order to increase the cost savings due to premature deflation in our program we have employed a slightly refined strategy, which is similar to the step function method discussed in Section 2.6, but takes also in account the Dickson transformation of the eigenvalues. Namely, since the method does not approximate directly the eigenvalues λ_i but their Dickson transform $\alpha_i = \lambda_i + \lambda_i^{-1}$, we have chosen starting points on the Dickson transform of the circles $|z| = \rho$, that is points lying on ellipses $\frac{\mathrm{Re}(z)^2}{(\rho+1/\rho)^2} + \frac{\mathrm{Im}(z)^2}{(\rho-1/\rho)^2} = 1$. More precisely, this is the algorithm we used to pick the starting points:

Input: Number $N = nk/2$ of eigenvalues to approximate and parameters $a \in \mathbb{N}$ and $b \in \mathbb{N}$
Output: Starting points z_k, $k = 1, \ldots, N$

$\theta = 2\pi/N$;
ϕ=randn;
for $j = 1, \ldots, N$
  jj=mod(j, a);
  $\rho = 1 - jj/b$;
  $\alpha = \rho + 1/\rho$;
  $\beta = 1/\rho - \rho$;
  $z_j = \alpha \cos(j * \theta + \phi) + \beta \sin(j * \theta + \phi)$
end

The integer a determines the number of ellipses whereas b is used to tune the lengths α and β, defined as above, of their semiaxes. We expect that a good choice for the parameters a and b depends on the ratio k/n: when $k \gg n$ we expect many eigenvalues to lie on or near to the unit circle, while when $n \gg k$ we expect a situation more similar to the eigenvalues of a random matrix, with no particular orientation towards unimodularity. We therefore expect that a small ratio a/b works well in the former case while on the contrary in the latter case $a \simeq b$ should be a better choice. Moreover, we expect that as nk grows it is helpful to increase the total number a of ellipses as well.

We show here some of the results on random T-palindromic polynomials. Figure 23 refers to an experiment on small-dimensional, high-degree polynomials: the value of n has been set to 5 while k was variable. The average number of t over a set of 1000 random polynomials for each value of $N = nk$ is shown on the graph. The parameters satisfy $a \in \{2, 3\}$ and $b \in \{8, 64\}$ and they are determined by $a = 1 + 2^c$ and $b = 8^{c+1}$, where the integer c is defined as the integer part of $c = \log_{320} N$. The graph shows a linear growth of t with respect to $N = nk$.

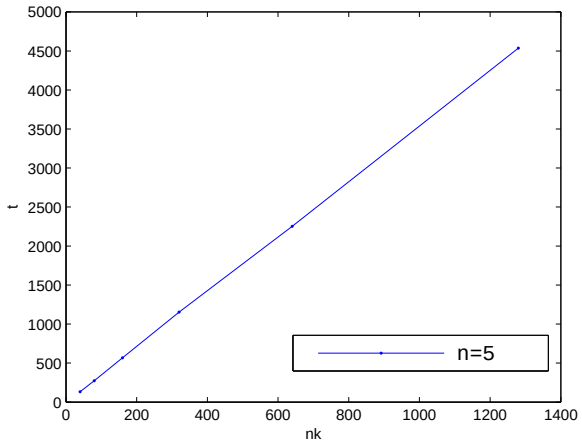

Fig. 23. *Dependence of* t *on* nk *for* $n = 5$

Figure 24 refers to an experiment where on the contrary the case of small k is explored. We have considered here $k = 2$ and let n vary and we show the results for t plotted against nk for several choices of a and b. The choice labelled as "step function" is for $a = \{6, 11\}$ and $b = \{6, 12\}$ generated by $a = 1 + 5\ 2^c$ and $b = 6\ 2^c$. Once again the experiments suggest that when the starting points are conveniently chosen $t \leq \alpha N$ for some constant α and any N in the specified range, and, moreover, the bound still holds for different reasonable choices of the parameters a and b. The experimentation with random polynomials gives $\alpha \simeq 8$ as an estimate for the constant.

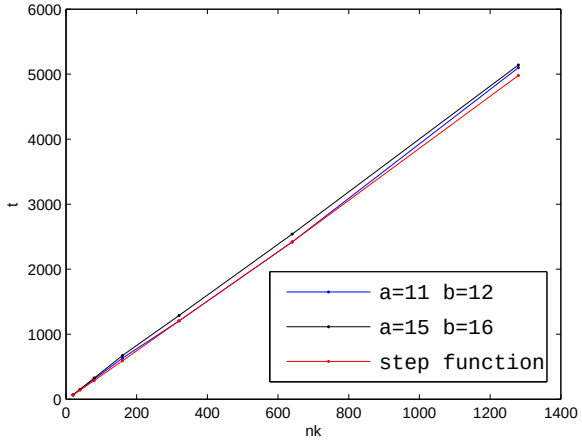

Fig. 24. *Dependence of* t *on* nk *for* $k = 2$

We have reached for the structured case the same conclusions that we drew for the unstructured EAI: the algorithm can greatly benefit from a smart strategy for the selection of the starting points by increasing the number of early deflations. The experiments show that as long as the starting points are suitably

chosen the value of t is proportional to $N = nk$. Notice that for the structured palindromic case linearity is achieved also in the region $n \gg k^2$.

Let us complete this subsection by justifying the claim we made in Section 4.1 by comparing the number of scalar iterations needed before convergence by the naive approach and the number of scalar iterations needed by the change of variable approach. The next figures show the average number of scalar iterations t for $n = 2$ and various values of k for three methods: the naive approach (blue), the implicit change of variable method (red) and the explicit change of variable method (black). Starting points were chosen according to a step function criterion and the tests were performed on sets of random T-palindromic polynomials.

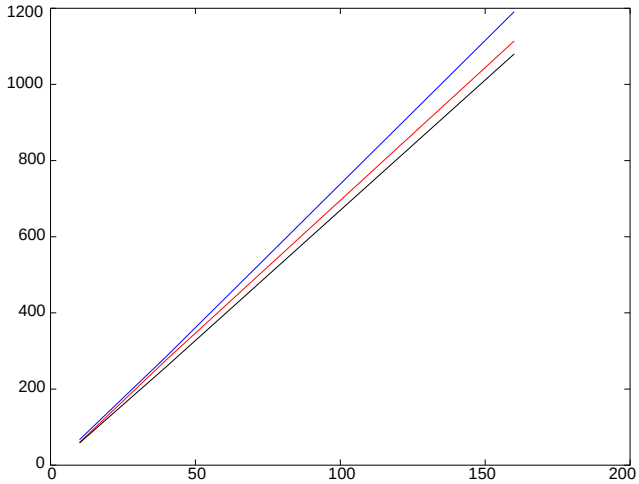

Fig. 25. *Average number of scalar iterations for T-palindromic polynomials*

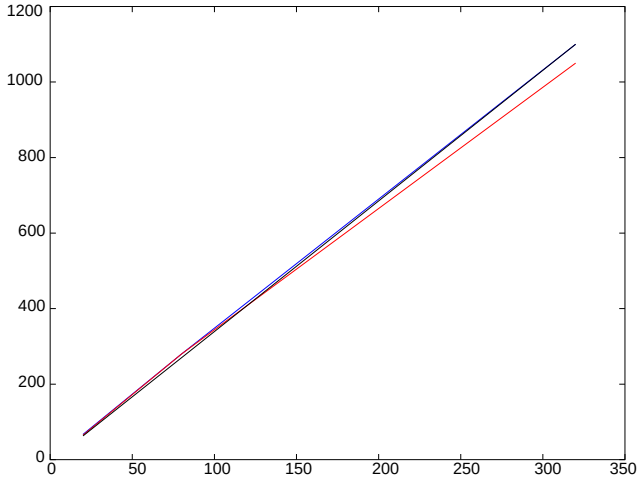

Fig. 26. *Average number of scalar iterations for even polynomials*

Notice that the naive approach turns out to be slower than the implicit substitution approach. The effect is more evident for palindromic than for even

polynomials; moreover, for the palindromic case the explicit substitution method appears to be the fastest, while for even polynomials it does not look faster than the naive approach. The reason for these different behaviours is not clear and deserves further investigation. However, since the (either explicit or implicit) substitution approach is always at least as efficient as the naive approach, and in some cases, such as palindromic, it is definitely faster, our advice is to prefer the former unless either theoretic or heuristic reasons not to do so are identified.

The drawback of the substitution approach is the loss of accuracy for exceptional eigenvalues, discussed in more details in the next subsection. We anticipate however that such a shortcoming can be overcome by means of a structured refinement method that we will present in Chapter 5.

4.6.2 Accuracy of the SEAI

The other important aspect of our solver based on polynomial root-finding concerns the accuracy of computed approximations. In our experience, the unstructured EAI competes very well in accuracy with the customary QZ-algorithm. For the SEAI, the accuracy of the computed non-exceptional roots for the random polynomials is always at least comparable with the accuracy of the approximation obtained with the QZ method as implemented in `polyeig`.

Let us give in the following some evidence of this claim by some examples of the SEAI applied to different structures. The results of other numerical experiments confirm the robustness of the novel method.

Let us consider first the T-palindromic structure. We will present experiments performed by means of the method of the explicit construction of a new matrix polynomial $M(z)$, $z = x + x^{-1}$ (Subsection 4.3.1). Figure 27 illustrates the computed eigenvalues for the problem $H_{5,40}$. Figure 28 also reports the plot of the absolute error vector $|y_{EA} - \tilde{y}|$ and $|y_{QZ} - \tilde{y}|$, where $\tilde{y}$ is the vector formed by the eigenvalues computed in high precision arithmetic by Mathematica[1] while y_{EA} and y_{QZ} are, respectively, the vectors formed from the eigenvalues returned by the SEAI and suitably sorted by the internal function *polyeig*.

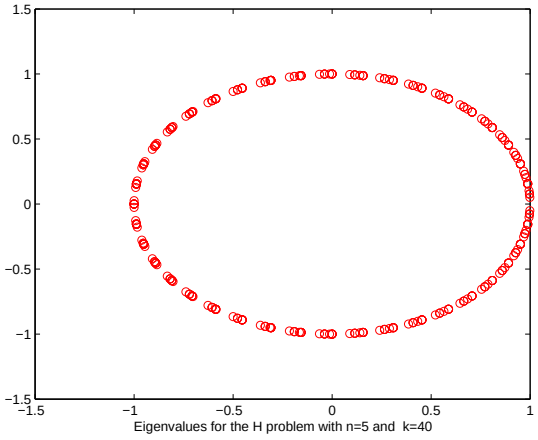

Fig. 27. *Approximations of the eigenvalues of* $H(5, 40)$

[1] Mathematica is a registered trademark of Wolfram Research, Inc.

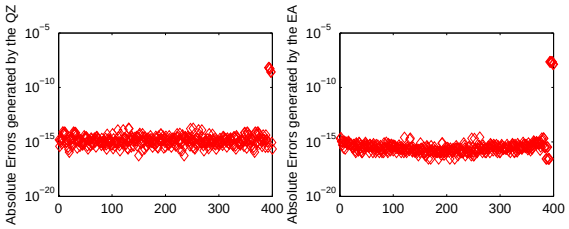

Fig. 28. *Absolute errors for the eigenvalues of* $H(5, 40)$*: polyeig (left) and SEAI (right)*

The numerical results put in evidence the following important aspects:

1. Poor approximations for the exact eigenvalue -1 are in accordance with the theoretical predictions: in fact the reverse transformation from $z = x + x^{-1}$ to $x = \frac{1}{2}(z \pm \sqrt{z^2 - 4})$ is known to be ill-conditioned near $z = \pm 2$ (or $x = \pm 1$). Since in this example -1 is a defective eigenvalue, the approximations returned by *polyeig* have comparable absolute errors of order 10^{-8} which are in accordance with the unstructured backward error estimates given in [55]. On the contrary in problems where -1 is semisimple, the QZ algorithm achieves approximations with lower forward error.

2. The accuracy of the remaining approximations is unaffected from the occurrence of near-to-critical eigenvalues and is in accordance with the results returned by *polyeig*. For most non-exceptional eigenvalues, the accuracy of approximations computed by our method is slightly better.

3. This kind of behaviour is confirmed by many other experiments. Our method performs similarly to the QZ for non-exceptional eigenvalues and for defective exceptional eigenvalues, but generally worse than QZ and the structure-preserving methods [100] for exceptional eigenvalues.

Let us now consider the even/odd structure. We will present for this case experiments performed on the implicit change of variable. We recall that the theoretical analysis given in the previous sections suggest the substitution $z = x^2$ for even or odd matrix polynomials.

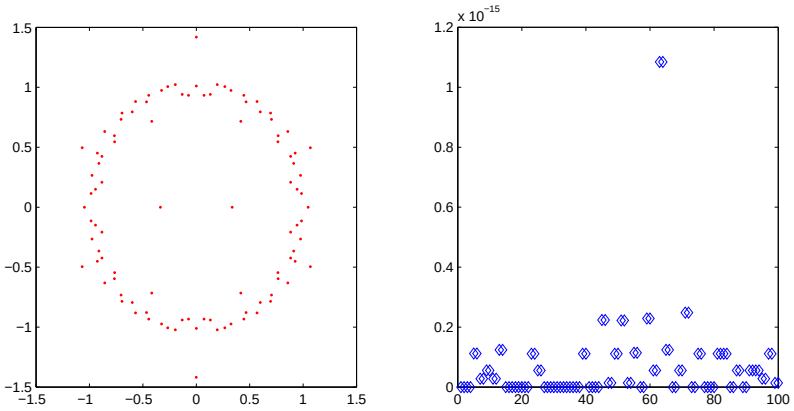

Fig. 29. *Approximated eigenvalues and absolute errors for a random odd matrix polynomial*

Figure 29 shows the Ehrlich-Aberth approximations of the eigenvalues of a random odd real matrix polynomial $P(x)$ of degree 50 and size 2. It also shows the absolute error in terms of distance between the approximated solutions and the exact solutions, computed solving the equation $\det P(x) = 0$ with `MPSolve`, with the assumption that the randomly generated coefficients are exact. The worst absolute error turns out to be the one associated with the largest purely imaginary eigenvalue: for this eigenvalue, the corresponding approximation calculated by `polyeig` shows an absolute error about 4 times larger than our algorithm.

To further test the accuracy of the method, the following example [68] has been used: let $A = \begin{bmatrix} 0 & 1-\phi & 0 \\ \phi-1 & 0 & i \\ 0 & -i & 0 \end{bmatrix}$ and $M = \left[\begin{smallmatrix} 2A & 0 \\ 0 & A \end{smallmatrix}\right]$. The exact eigenvalues of $xI_6 + M$ are $\{0, 0, \pm w, \pm 2w\}$ with $w := \sqrt{2\phi - \phi^2}$. As $\phi \to 0$ the unstructured condition number grows [68], leading to decreasing accuracy for customary matrix algorithms. There exist structured matrix methods that perform much better [68], so it would be satisfying that the structured variant of Ehrlich-Aberth outperforms `polyeig` as well. Experiments have been performed for several values of $k \in \mathbb{N}$ and $\phi = 2^{-k}$.

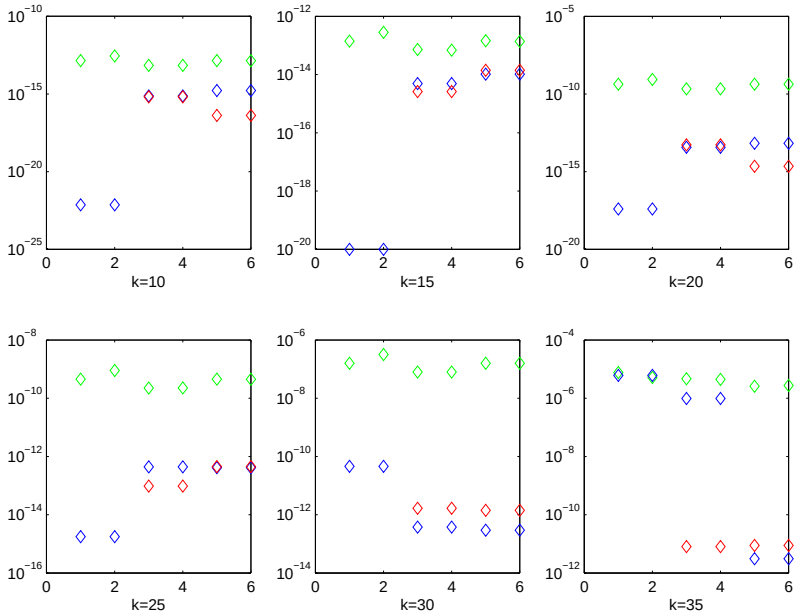

Fig. 30. *Absolute errors in computed eigenvalues of the skew-symmetric matrix M*

Figure 30 shows the absolute errors of the eigenvalues approximated with three different methods: `polyeig` (green), SEAI (blue), and SEAI with preliminary deflation of the exactly known eigenvalue 0 (red). The absolute error of the deflated eigenvalue is not shown, being zero.

Notice that the experiment indicates that there is no loss of accuracy for near-zero eigenvalues. Even when 0 is not deflated the performance in accuracy is considerably better than the unstructured matrix method. To explain the effect, notice that, in contrast with the palindromic structure, for even/odd polynomials the back transformation is $x = \pm\sqrt{z}$, for which only the absolute condition number, but not the relative condition number, diverges near 0. The SEAI seems to be very effective for the $\{\lambda, -\lambda\}$ eigenstructure, and this time exceptional eigenvalues do not suffer from a lower accuracy.

In the NLEVP library [7] there are two structured problems with even/odd structure. We further verify the reliability of our method by testing it on these two problems. The next figures compare four algorithms applied to the NLEVP problems butterfly and wiresaw1, which are even [7]. Forward errors are computed by comparing the approximation with values computed in high precision arithmetic. The outcome of such computation is shown in logarithmic scale for four algorithms: `polyeig`'s QZ (blue + symbol), the UEAI (red $*$ symbol), the structured matrix method URV applied to an even linearization [100] (black x symbol), and the SEAI relying on the change of variable $z = x^2$ method (green o symbol).

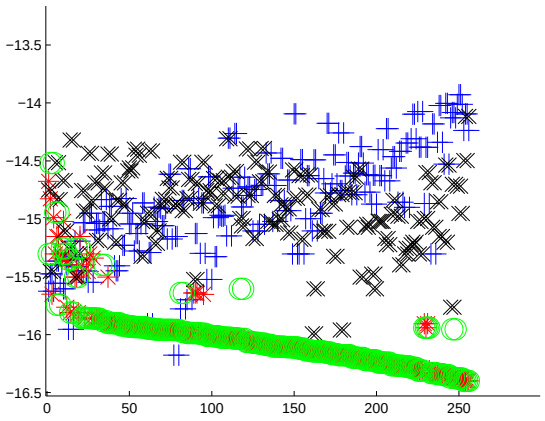

Fig. 31. *Forward absolute errors for the problem butterfly*

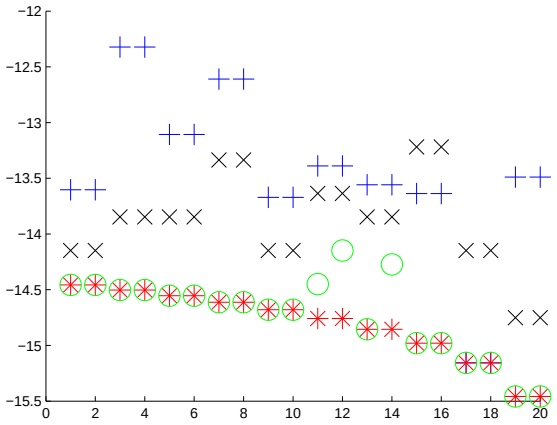

Fig. 32. *Forward absolute errors for the problem wiresaw1*

It is clear from the figures above that for the problem butterfly the EAI was more accurate than the two matrix methods, but structured methods did not improve much the accuracy of each unstructured counterpart. A possible explanation is that for these matrix polynomials the unstructured condition numbers for the eigenvalues are not much different from the structured condition numbers; under such circumstances, structured methods do not improve much the accuracy, even though they improve the efficiency.

As a final test, we have used the SEAI on a more generic structure such that the eigenvalues are paired. A matrix polynomial $W(x)$ with $n = 2$ and $k = 10$ was built in such a way that its eigenvalues appear in couples of the form $\{\lambda, \frac{\lambda+1}{\lambda-1}\}$. In order to devise a problem not too easy to solve numerically, the determinant of the polynomial was designed to be Wilkinson-like[2]: $\det(W(x)) = const. \cdot \theta(x) \cdot \theta(\frac{x+1}{x-1})$, $\theta(x) = x \cdot \prod_{j=2}^{10}(x-j)$. The next figure shows the absolute forward errors of the computed approximations with respect to the known exact

[2]That is, $\theta(x)$ mimics the notorious Wilkinson polynomial [110].

eigenvalues for three methods: QZ (`polyeig`, blue + symbol), UEAI (red $*$ symbol) and SEAI relying on the change of variable $z = \frac{x^2+x}{x-1}$ (green o symbol). Numerically zero errors were formally set equal to $\epsilon/2$.

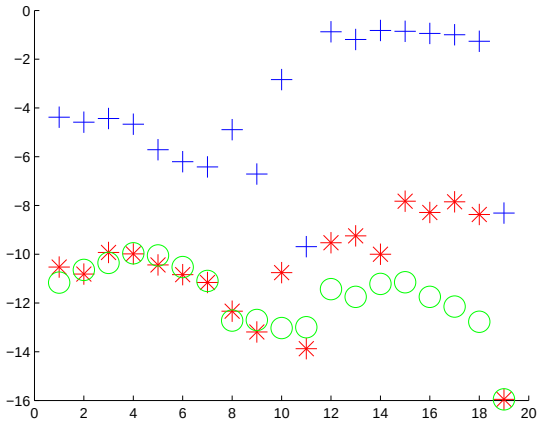

Fig. 33. *Forward absolute errors for the structured problem* $\det W(x) = 0$

The following table reports the relative errors of the three methods considered above for all the eigenvalues but 0 (all the three algorithms detected the zero eigenvalue with an absolute error smaller than the machine epsilon). It also reports the relative error for the Matlab's function `eig` (without scaling) applied to a suitable linearization, chosen according to the prescriptions of [60]. Notice that here all the nonzero eigenvalues have modulus ≥ 1, so the suggested (near-to-optimal) linearization in the space $\mathbb{DL}$ according to [60] would in principle be the pencil in $\mathbb{DL}$ corresponding to the ansatz vector e_1 (see [78] for further details). Unluckily, since one eigenvalue is zero, that pencil is not a linearization at all [78]. Following the suggestions on conditioning of [60] and the theory on linearizations of [78], we have therefore taken the slightly perturbed vector $e_1 + 2^{-23}e_{10}$ as an ansatz vector. The factor 2^{-23} has been heuristically chosen picking the integer $\alpha \leq 52$ that minimises the average relative error when applying `eig` to the linearizations in $\mathbb{DL}$ associated with the ansatz vectors $e_1 + 2^{-\alpha}e_{10}$.

Eigenvalue	R. e., polyeig	R. e., eig	R. e., UEAI	R. e., SEAI
-1	4.9e-9	$\epsilon/2$	$\epsilon/2$	$\epsilon/2$
11/9	4.5e-2	4.8e-8	3.5e-9	1.4e-13
5/4	8.1e-2	2.2e-7	1.1e-8	5.7e-13
9/7	8.9e-2	4.2e-7	4.0e-9	1.4e-12
4/3	1.1e-3	4.1e-7	1.1e-8	4.6e-12
7/5	9.9e-2	2.2e-7	7.1e-11	5.1e-12
3/2	8.9e-2	6.3 e-8	3.8e-10	2.5e-12
5/3	3.9e-2	8.9e-9	1.8e-10	1.1e-12
2	1.0e-10	4.9e-10	8.8e-12	5.1e-14
2	7.3e-4	2.3e-12	6.7e-15	4.7e-14
3	4.3e-6	6.3e-12	2.2e-14	6.3e-14
3	6.5e-8	5.8e-12	1.5e-13	6.7e-14
4	9.6e-8	1.5e-10	1.7e-12	2.0e-12
5	1.3e-7	8.1e-10	2.9e-12	5.9e-12
6	3.2e-7	2.2e-9	6.1e-12	1.5e-11
7	3.1e-6	3.3e-9	1.5 e-11	1.6e-11
8	4.6e-6	2.7e-9	1.5e-11	5.5e-12
9	2.9e-6	1.1e-9	1.7e-12	2.5e-12
10	4.2e-6	1.7e-10	3.0e-12	6.9e-13
Average	3.8e-5	9.6e-10	5.5e-12	4.1e-13

We may conclude that on this structured problem the EAI outperforms the QZ method for what concerns accuracy. Apparently `polyeig` struggles quite a bit here, which is coherent with the results of [60]. It is therefore more appropriate to compare with the strategy of [60], that works better, but still worse than the EAI. Although there are some approximations that do not benefit from the use of the structured version of the Ehrlich-Aberth algorithm, the SEAI has an overall advantage in accuracy over the UEAI, besides the obvious efficiency advantage. The computation time for the SEAI was about one third of the computation time for the UEAI.

4.6.3 Conclusions

Let us now summarise the results of the experimentation on the SEAI.

- When the determinant of the matrix polynomial has the property that its roots appear in pairs (in Chapter 5, we will introduce the adjective *twined* to indicate such class of scalar polynomials), a structured EAI can be applied in order to halve the number of computed approximations, with an immediate computational advantage.

- Three different strategies are available to this aim: the first one is to force half of the eigenvalue to respect the known structure, but it is not always as efficient as the other ones. The other two strategies are more sophisticated: one may either build a new matrix polynomial whose properties can be investigated using the tools of Chapter 3, or use an implicit change of variable.

- All the methods above can be implemented to design efficient algorithms.

- There are examples for which the accuracy of the approximation benefits from the use of the SEAI with respect to the UEAI; the SEAI leads to forward errors that are generally lower than unstructured customary matrix methods, and at least comparable with structure-preserving matrix methods;
- For some structures, some of the novel methods suffer from a loss of accuracy near exceptional eigenvalues; this is not due to less accurate approximations from the EAI, but to the ill-conditioning of back transformation to the original variable.
- The SEAI would therefore obtain great benefit from a structured refinement of exceptional or near-to-exceptional eigenvalues, in order to overcome its lower performances when dealing with such roots of the determinant; the development of such a refining method is the subject of the next chapter.

Chapter 5

A structured Newton algorithm for the refinement of roots of twined scalar polynomials

As we have seen in Chapter 4, in some cases the SEAI leads to some loss of accuracy for special eigenvalues. This is mainly due to ill-conditioned back transformations after a change of variable. A refinement method that still preserves the spectral structure is then needed.

In the present chapter, a Newton method for the structured refinement of such inaccurate approximations is presented. The chapter is based on the results of [41].

5.1 Introduction

The aim of this section is to briefly review the attributes of structured scalar polynomials that have the property of having roots that appear in couples. We will use the term *twined polynomials* (more formally defined below) to indicate such special subclass of scalar polynomials.

Let $a(z) = \sum_{j=0}^{n} a_j z^j$, be a polynomial of grade $n = 2m$. Assume that $a_j = a_{n-j}$ for all $j \leq n$, so that $a(z)$ is a scalar palindromic polynomial. In the following we will use the acronym *egp* to refer to the class of even-grade palindromic polynomials.

Suppose that $b, c, d \in \mathbb{C}$, $d^2 + bc \neq 0$, and let $f(x) = \frac{dx+b}{cx-d}$, so that $f(f(x)) \equiv x$. We agree that $f(\infty) := d/c$ and $f(d/c) := \infty$. The palindromic property can be generalised to $a(x) = 0 \Leftrightarrow a(f(x)) = 0$. An even-grade $f(x)$-twined (*f(x)-egt*) polynomial is defined [41] as a polynomial of even grade satisfying such a symmetry of its roots. Besides palindromic polynomials, other well-known examples of *egt* polynomials are even polynomials ($f(x) = -x$, j odd $\Rightarrow a_j = 0$) and odd polynomials ($f(x) = -x$, j even $\Rightarrow a_j = 0$)

We have already met even-graded twined polynomials in Chapter 4, as determinants of a class of structured matrix polynomials. Actually, a broader class of

structured PEPs was then discussed: for instance, an anti-T-palindromic matrix polynomial of grade 3 and dimension 3 has as determinant an odd-graded antipalindromic scalar polynomial $p(x)$, which is neither twined nor even-graded. Nevertheless, it can always be written as $p(x) = (x-1)q(x)$, where $q(x)$ is an even-grade palindromic (and, thus, twined) polynomial. Moreover, the factorization can be exploited by the SEAI.

Such a situation is general. The determinants of the structured PEPs studied in Chapter 4 can be reconducted, when using the SEAI, to the even-grade case by the use of some tricks. In the most general case of our interest, they are in fact the product of an even-grade $f(x)$-twined polynomial and of a factor that takes into account exceptional eigenvalues of odd multiplicity. Usually, this latter factor can be easily dealt with, because it is known *a priori* from theoretical considerations. We can therefore focus on the even grade case without any loss of generality.

Moreover, from the analyses we made in Chapter 4 we know that there exists a rational function of degree 2, $z = z(x)$, such that the change of variable from x to z can be exploited in order to design a structured, effective version of the EAI. Such a change of variable can be performed either implicitly or explicitly, by building a new matrix polynomial whose eigenvalues α_j are such that $\alpha_j = z(\lambda_j) = z(f(\lambda_j))$, where the λ_j are the eigenvalues of the original PEP. Whatever the choice, the λ_j are then obtained in a second phase, by solving a quadratic equation whose coefficients depend on the corresponding computed approximation of α_j. More details have been described in Chapter 4.

Our extensive numerical experience reported in [12, 42, 92] and in Chapter 4 says that the computed approximations of α_j are generally very accurate, but there can be some problems in the second step of the algorithm. A potential pitfall of such a SEAI is that the reconstruction of $\{\lambda_j, f(\lambda_j)\}$ from α_j becomes ill-conditioned as α_j approaches $z(\phi)$, where ϕ is a fixed point of f, i. e. $f(\phi) = \phi$. For instance, in the case of T-palindromic PEPs, for which $f(x) = 1/x$, the two fixed points of f correspond to the two exceptional eigenvalues ± 1; if one uses the Dickson change of variable $z = \frac{x^2+1}{x}$, then $z(\pm 1) = \pm 2$ and the back transformation becomes numerically ill-conditioned when α_j is closed to ± 2 so that the original eigenvalues are near-to-exceptional. Moreover, for the explicit change of variable implementation of the SEAI numerical difficulties could be encountered for approximating eigenvalues corresponding to very large values of α_j, due to the presence of extra defective infinite eigenvalues of the new matrix polynomial.

For T-palindromic polynomials, this loss of accuracy has been discussed in detail in [42] and in Chapter 4. This motivates the search of numerical methods for the iterative refinement of the roots of the *egp* polynomial $a(z)$. Similarly, the application of root-finders to other structured eigenvalue problems may lead to the problem of the refinement of certain roots of $f(x)$-twined polynomials in correspondence to the fixed points of $f(x)$, as discussed in [12] and Chapter 4.

In the present chapter we develop algorithms for the simultaneous approximation of two symmetric roots of an *f(x)-egt* polynomial. Besides the above mentioned application to structured PEPs, there exist other situations where *f(x)-egt* polynomials appear: see [48, 73] and the references given therein. For the sake of concreteness, we will focus in particular on the example of the *egp* polynomials; but we will mention also how such algorithms can be easily adapted

to the more general case. The algorithms can be viewed as a specialization of the Newton method for $f(x)$-twined polynomials.

More specifically, we introduce some symmetric division processes for palindromic polynomials by showing that for a given $a(z)$ of grade $n = 2m$ and a given $\xi \in \Omega$, Ω open subset in $\mathbb{C}$, under some mild assumptions there exist a unique palindromic polynomial $q(z)$ of grade $n-2$ and a unique number $r \in \mathbb{C}$ such that $a(z) = q(z)(z-\xi)(z-\xi^{-1}) + rs(z)$, where $s(z)$ is a fixed palindromic polynomial. The Newton method is applied to the solution of the nonlinear equation $r(\xi) = 0$, $r\colon \Omega \to \mathbb{C}$, yielding our strategy for factoring out palindromic quadratic factors of $a(z)$. Analogous results hold for the more general case of *egt* polynomials.

Suitable choices of the polynomial $s(z)$ characterising the division process lead to iterative methods in the style of the EAI. The computational work per iteration basically reduces to evaluating the Newton correction $a(\xi)/a'(\xi)$, $\xi \in \Omega$. The cost of the evaluation depends on the representation of $a(z)$. If the polynomial is implicitly given as the determinant of a matrix polynomial then the Jacobi formula can be used for computing the Newton correction. If on the contrary one has to deal directly with a scalar *f(x)-egt* polynomial, the evaluation of the Newton correction is cheaper.

5.2 Symmetric Division of Palindromic Polynomials

Specialised division algorithms for polynomials with symmetries have been described in several papers: see, e.g., [99] and the references given therein. In the present section we introduce some novel symmetric division processes for palindromic polynomials aimed to exploit the symmetric distribution of the coefficients and, *a fortiori*, of the spectrum of the polynomial.

The first division scheme relies upon the following theorem [41].

Theorem 5.1. *Let $a(z) \in \mathbb{C}[z]$ be a palindromic polynomial of even grade n, i.e., $a(z) = \sum_{j=0}^{n} a_j z^j$, $n = 2m$, $a_j = a_{n-j}$ for $0 \le j \le n$. For any $\xi \in \mathbb{C}$ satisfying $\xi^{n-1} + \xi \neq 0$ there exist uniquely determined a palindromic polynomial $q(z)$ of grade $n-2$ and a scalar $r \in \mathbb{C}$ such that*

$$a(z) = q(z)(z-\xi)(z-\xi^{-1}) + r(z^{n-1} + z). \tag{5.1}$$

Proof. The proof of Theorem 5.1 is constructive. Moreover, it can be translated into a computational procedure for determining the symmetric quotient and the remainder.

Let us set $q(z) = \sum_{j=0}^{n-2} q_j z^j$, $q_j = q_{n-2-j}$, $0 \le j \le n-2$, and $-\alpha = \xi + \xi^{-1}$, with $\xi \neq 0$. Relation (5.1) can be translated into the circulant-like linear system

$$\left[\begin{array}{c} a_1 \\ a_2 \\ \vdots \\ a_2 \\ a_1 \\ \hline 2a_0 \end{array}\right] = \left[\begin{array}{ccccc|c} \alpha & 1 & & & & 1 \\ 1 & \alpha & 1 & & & 0 \\ & \ddots & \ddots & \ddots & & \vdots \\ & & 1 & \alpha & 1 & 0 \\ & & & 1 & \alpha & 1 \\ \hline 1 & 0 & \dots & 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} q_0 \\ q_1 \\ \vdots \\ q_1 \\ q_0 \\ \hline r \end{array}\right]. \tag{5.2}$$

The tridiagonal Töplitz matrix T, which is the block element $(1,1)$ of the block matrix in (5.2), can be diagonalised as $T = U \cdot D \cdot U$, where $U = \sqrt{\frac{2}{n}}(\sin\frac{ij\pi}{n})$ and $D = \operatorname{diag}(\alpha + 2\cos\frac{\pi}{n}, \ldots, \alpha + 2\cos\frac{(n-1)\pi}{n})$. Using the Schur form of T, we can reduce the coefficient matrix of (5.2) into arrowhead form. In addition, from the structure of U and the palindromic property of $\boldsymbol{q} = [q_0, q_1, \ldots, q_1, q_0]^T$, $\boldsymbol{a} = [a_1, a_2, \ldots, a_2, a_1]^T$ and $\boldsymbol{e} = \boldsymbol{e}_1 + \boldsymbol{e}_{n-1}$ we find that the system can be further compressed, obtaining the equivalent form of size $m+1$

$$\left[\begin{array}{c|c} \operatorname{diag}(\alpha + 2\cos\frac{(2[1:m]-1)\pi}{n}) & P \cdot U\boldsymbol{e} \\ \hline \boldsymbol{e}^T U \cdot P^T & 0 \end{array}\right] \left[\begin{array}{c} \boldsymbol{w} \\ \hline r \end{array}\right] = \left[\begin{array}{c} P \cdot U\boldsymbol{a} \\ \hline 2a_0 \end{array}\right], \tag{5.3}$$

where $P^T\boldsymbol{w} = U\boldsymbol{q}$ and $P \in \mathbb{R}^{m\times(n-1)}$ is the restriction matrix defined by $P = (\delta_{2i-1,j})$, where $\delta_{i,j}$ is the Kronecker delta. Thus we deduce that there exist unique $q(z)$ and r satisfying (5.1) if and only if the coefficient matrix of the system is nonsingular. If T is invertible then for the determinant d of the coefficient matrix in (5.3) we find

$$d = \pm s \cdot t_m(\alpha/2), \quad s = -\boldsymbol{e}^T \cdot T^{-1}\boldsymbol{e},$$

where $t_k(x)$ denotes the Chebyshev polynomial of the first kind of degree k. By using the representation of the semiseparable matrix T^{-1} [19] we find that

$$d = \pm 2\frac{u_{m-1}(\alpha/2)t_{m-1}(\alpha/2)}{u_{m-1}(\alpha/2)t_m(\alpha/2)}t_m(\alpha/2) = \pm 2t_{m-1}(\alpha/2),$$

where $u_k(x)$ denotes the Chebyshev polynomial of the second kind of degree k. By continuity, this gives the determinant for any value of α. Hence, we may finally conclude that the condition for the invertibility is $t_{m-1}(\alpha/2) \neq 0$ or, equivalently, $\xi^{n-2} + 1 \neq 0$. □

Remark 5.1. *The linear system (5.2) can be solved in linear time, using the QR factorization method. The reduced linear system (5.3) can also be solved in linear time, but the computation of $U\boldsymbol{a}$ and $\boldsymbol{q} = UP^T\boldsymbol{w}$ costs a bit more, requiring a sine transform of order n. The advantage of the reduction is that the structure of $q(z)$ can be computed exactly, by exploiting the symmetries in the matrix U. It is an interesting problem to devise a linear complexity algorithm that is able to compute $q(z)$ while preserving the palindromic property of the coefficients.*

The previous result enables the derivation of another symmetric division scheme for *egp* polynomials. Observe that $0 + z\cdot a(z) + 0z^{n+2}$ is an *egp* polynomial of grade $n+2$. From Theorem 5.1 we obtain that there exist a unique scalar r and a unique polynomial $q(z) = q_0 + z \cdot \hat{q}(z) + q_0 z^n$ of grade n such that

$$z \cdot a(z) = q(z)(z - \xi)(z - \xi^{-1}) + r(z^{n+1} + z).$$

Since $q_0 = q(0) = 0$, it follows that

$$a(z) = \hat{q}(z)(z - \xi)(z - \xi^{-1}) + r(z^n + 1). \tag{5.4}$$

We can express this result with the following corollary.

Corollary 5.1. *Let $a(z) \in \mathbb{C}[z]$ be a palindromic polynomial of even grade n, i.e., $a(z) = \sum_{j=0}^{n} a_j z^j$, $a_n \neq 0$, $n = 2m$, $a_j = a_{n-j}$ for $0 \leq j \leq n$. For any $\xi \in \mathbb{C}$ satisfying $\xi^{n+1} + \xi \neq 0$ there exist uniquely determined a palindromic polynomial $q(z)$ of grade $n-2$ and a scalar $r \in \mathbb{C}$ such that*

$$a(z) = q(z)(z - \xi)(z - \xi^{-1}) + r(z^n + 1). \tag{5.5}$$

This result can be further generalised along two different directions. The first is to consider remainder polynomials of the form $z^{n-\ell} + z^{\ell}$, $0 \leq \ell \leq m$. The second, even more interesting, generalization of Corollary 5.1 comes from the observation that $z^n + 1$ is a palindromic polynomial of degree n which can be factorised as the product of quadratic factors with pairwise distinct roots. The following extension is easily proved [41].

Theorem 5.2. *Let $s(z) = \prod_{j=1}^{m}(z^2 - \alpha_j z + 1)$, $\alpha_j \neq \alpha_k$ for $j \neq k$, be a fixed polynomial of degree $n = 2m$. For any palindromic polynomial $a(z)$ of grade n and any $\xi \in \mathbb{C}$ satisfying $\xi^{-1} + \xi \neq \alpha_j$, $1 \leq j \leq m$, there exist uniquely determined a palindromic polynomial $q(z)$ of grade $n-2$ and a scalar $r \in \mathbb{C}$ such that*

$$a(z) = q(z)(z - \xi)(z - \xi^{-1}) + rs(z). \tag{5.6}$$

Proof. Define $t := z + z^{-1}$. Let us introduce the functions $c_j(t) = z^j + z^{-j}$, $j \geq 0$. Such functions are monic Chebyshev-like polynomials (that is, they differ from Chebyshev polynomials only for a linear change of variable and an overall constant factor) of degree j; thus, they are linearly independent. Moreover, they satisfy the prosthaphaeresis identities

$$c_j(t)c_k(t) = c_{j+k}(t) + c_{|j-k|}(t). \tag{5.7}$$

The relation (5.6) can be rewritten as

$$\begin{aligned} \sum_{j=0}^{m-1} a_j c_{m-j}(t) + \frac{a_m}{2} c_0(t) &= (c_1(t) - \frac{\alpha}{2} c_0(t)) \left(\sum_{j=0}^{m-2} q_j c_{m-1-j}(t) + \right. \\ &+ \left. \frac{q_{m-1}}{2} c_0(t)) + r \prod_{j=1}^{m} (c_1(t) - \frac{\alpha_j}{2} c_0(t) \right), \end{aligned}$$

where $\alpha = \xi + \xi^{-1}$.

With the use of (5.7) and equating the coefficients of $c_j(t)$, $j = 0, \ldots, m$, this is equivalent to a linear system of $m+1$ equations in $m+1$ unknowns. Therefore the coefficients q_j, $j = 0, \ldots, m-1$, and the scalar r are uniquely determined whenever the matrix of such a linear system is nonsingular. An argument relying on interpolation techniques at the nodes $\alpha, \alpha_1, \ldots, \alpha_m$ can be used in order to show that this happens whenever $\alpha \neq \alpha_j \ \ \forall\ 1 \leq j \leq m$. □

Observe that in principle some α_j could also be allowed to be infinite, adopting the convention $z^2 + \infty z + 1 := 0z^2 + z + 0 = z(0c_0(t) + c_1(t)/2)$. This extends Theorem 5.2 to a generic $s(z)$ of grade $2m$.

All the considered symmetric division processes can be carried out by numerically robust algorithms. For instance, the function **symdiv**, that follows, implements the division algorithm described in the proof of Theorem 5.1 by

solving the reduced system (5.3). Given in input the coefficients of $a(z)$ and $\alpha = -(\xi + \xi^{-1})$ it returns as output the coefficients of $q(z)$ and r satisfying (5.1).

function symdiv
Input: $a_0, \ldots, a_{m-1}, a_m, a_{m-1}, \ldots, a_0$, α;
Output: $q_0, \ldots, q_{m-2}, q_{m-1}, q_{m-2}, \ldots, q_0$, r;

$n = 2m$; $n1 = n - 1$; $\boldsymbol{q} = \text{zeros}(n1, 1)$
$\boldsymbol{e} = \text{zeros}(n1, 1)$; $e(1) = e(n1) = 1$; $\boldsymbol{d} = \text{zeros}(n1, 1)$;
$\boldsymbol{a1} = [a_0, \ldots, a_{m-1}]^T$; $F = \text{zeros}(m + 1)$; $U = \text{zeros}(n1)$;
for $i = 1 : n1$
  for $j = 1 : n1$
    $U(i, j) = \sin \dfrac{ij\pi}{n}$;
  end;
  $\boldsymbol{d}(i) = m(\alpha + 2\cos \dfrac{i\pi}{n})$;
end;
$\boldsymbol{e} = U^T \boldsymbol{e}$;
for $i = 1 : m$
  $i1 = i + 1$; $F(1, i1) = \boldsymbol{e}(2i - 1)$;
  $F(i1, 1) = F(1, i1)$; $F(i1, i1) = \boldsymbol{d}(2i - 1)$;
end;
$\boldsymbol{a12} = [\boldsymbol{a1}(2 : m); a_m; \boldsymbol{a1}(m: -1{:}2)]$; $\boldsymbol{a12} = U^T \boldsymbol{a12}$;
$[r; \boldsymbol{q1}] = F \backslash [2a_0; \boldsymbol{a12}(1{:}2{:}2m - 1)]$;
for $i = 1 : m$
  $\boldsymbol{q}(2i - 1) = \boldsymbol{q1}(i + 1)$;
end;
$\boldsymbol{q} = U\boldsymbol{q}$

The procedure turns out to be numerically robust even when $\alpha/2$ approaches the roots of $t_{m-1}(z)$. Figure 34 reports the plot of the computed backward error

$$err = \frac{\| \boldsymbol{a} - \text{conv}(\boldsymbol{q}, [1; \alpha; 1]) - r(\boldsymbol{e}_2 + \boldsymbol{e}_{n-1}) \|_\infty}{\max\{1, \| \boldsymbol{q} \|_\infty\}},$$

for $\boldsymbol{a} = [1{:}\ m; 0; m{:}\ -1{:}\ 1]$ and $\alpha = 2\cos \dfrac{\pi}{n-2} + \sqrt{\epsilon}$, where ϵ denotes the machine precision.

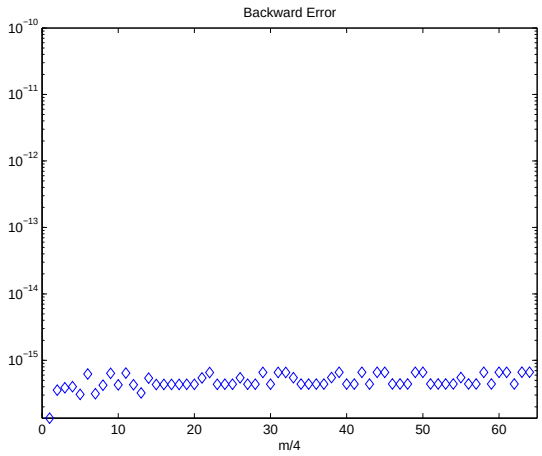

Fig. 34. *Plot of the error err(m)*

Concerning the measure of palindromicity $pm = \frac{\|q - \text{flipud}(q)\|_\infty}{\|q\|_\infty}$ we have always found $pm = 0$ in all the experiments performed. The Matlab built-in function `flipud` flips the elements of a column vector, generating another vector of the same size and with the same elements, but in the reverse order.

5.3 Symmetric Division of Twined Polynomials

The present section is devoted to the extension of Theorem 5.2 to the more general case of *f(x)-egt* polynomials [41].

The extension from palindromic to generic twined polynomials is done in two steps. Consider first the following intermediate generalization of a palindromic polynomial. We recall (Remark 4.4) that, given $\kappa \neq 0$, any $a(z) = \sum_{j=0}^{2m} a_j z^j$ satisfying $a_j = \kappa^{m-j} a_{2m-j}$ is called an even-grade κ-palindromic ($eg\kappa p$) polynomial; its roots come in pairs $\{\lambda, \frac{\kappa}{\lambda}\}$. See also Remark 4.4 for more details.

An analogous division process can be designed for such polynomials. The following corollary holds.

Corollary 5.2. *Let $s(z) = \prod_{j=1}^{m}(z^2 - \alpha_j z + \kappa)$, $\alpha_j \neq \alpha_k$ for $j \neq k$, be a fixed polynomial of grade $n = 2m$. For any κ-palindromic polynomial $a(z)$ of grade n and any $\xi \in \mathbb{C}$ satisfying $\kappa\xi^{-1} + \xi \neq \alpha_j$, $1 \leq j \leq m$, there exist uniquely determined a κ-palindromic polynomial $q(z)$ of grade $n-2$ and a scalar $r \in \mathbb{C}$ such that*

$$a(z) = q(z)(z - \xi)(z - \kappa\xi^{-1}) + rs(z). \tag{5.8}$$

Proof. The corollary can be proved just as in Theorem 5.2, after defining $t := z + \kappa z^{-1}$, $c_j(t) := z^j + \kappa^j z^{-j}$, and observing that they are linearly independent polynomials satisfying $c_j(t)c_k(t) = c_{j+k}(t) + \kappa^{\min(j,k)} c_{|j-k|}(t)$. □

Given a self-inverse analytic function $f(x) = \frac{dx+b}{cx-d}$, $d^2 + bc \neq 0$, consider now the class of the *f(x)-egt* polynomials $a(z)$, satisfying the property $a(\lambda) = 0 \Leftrightarrow a(\frac{d\lambda+b}{c\lambda-d}) = 0$.

Theorem 5.3. *Let $s(z) = \prod_{j=1}^{m}(z-\xi_j)(z-f(\xi_j))$, $\xi_j \neq \xi_k$ for $j \neq k$, be a fixed f(x)-egt polynomial of grade $n = 2m$. For any f(x)-egt polynomial $a(z)$ of grade n and any $\xi \in \mathbb{C}$ satisfying $\xi \neq \xi_j$, $1 \leq j \leq m$, there exist uniquely determined an f(x)-egt polynomial $q(z)$ of grade $n-2$ and a scalar $r \in \mathbb{C}$ such that*

$$a(z) = q(z)(z-\xi)(z-f(\xi)) + rs(z). \tag{5.9}$$

Proof. Let $\kappa = d^2 + bc \neq 0$. We will provide a group isomorphism between *f(x)-egt* polynomials of grade $2m$ and *egκp* polynomials of the same grade; the thesis will then follow from Corollary 5.2. Suppose that there exists a function $g(x) = \frac{\alpha x+\beta}{\gamma x+\delta}$, $\alpha\delta \neq \beta\gamma$, such that $g(f(x))g(x) \equiv \kappa$. Let $a(y) = \prod_{j=1}^{m}(y-\psi_i)(y-f(\psi_i))$ be an *f(x)-egt*; all *f(x)-egt*, up to a constant factor, can be factorised in this way adopting the formal convention $(y+\infty) := (0 \cdot y + 1)$ in order to deal with the case of infinite roots. The application

$$\Phi : a(y) \to \Phi_{\delta z-\beta,\alpha-\gamma z}(a(y)) := p(z) = [\alpha-\gamma z]^{\mathrm{grade}(a(y))}a(g^{-1}(z))$$

is a bijection between polynomials (Proposition 3.3) that preserves the grade. To see it, apply Lemma 3.1 and Proposition 3.1 with $N = D = G = 1$ and $g = \mathrm{grade}(a(y))$ and recall (Section 3.2) that by definition $\mathrm{grade}(p(z)) = gG = \mathrm{grade}(a(y))$ because $G = 1$. Moreover, the application above clearly preserves the group structure since $\Phi(a_1 + a_2) = \Phi(a_1) + \Phi(a_2)$ as long as a_1, a_2 and $a_1 + a_2$ have the same grade. Furthermore, if once again we formally agree that $(z+\infty) := (0 \cdot z + 1)$, then

$$p(z) = \mathrm{const.} \cdot \prod_{j=1}^{m}(\alpha-\gamma z)^2(g^{-1}(z)-\psi_i)(g^{-1}(z)-f(\psi_i)) =$$

$$= \mathrm{const.} \cdot \prod_{j=1}^{m}(z-g(\psi_i))(z-g(f(\psi_i))).$$

Therefore $p(z)$ is *egκp* if and only if $a(y)$ is *f(x)-egt*.

The existence of such a $g(x)$ is provided by solving the linear system

$$\begin{bmatrix} d & c & -\kappa & 0 \\ b & -d & 0 & -\kappa \\ -1 & 0 & d & c \\ 0 & -1 & b & -d \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} = 0,$$

which has nontrivial solutions satisfying the additional condition $\alpha\delta \neq \beta\gamma$ for any values of b, c, d. For instance, $g(x) = cx - d$ is a solution whenever $c \neq 0$, $g(x) = \frac{dx+b}{x}$ is a solution whenever $b \neq 0$, and $g(x) = d\frac{x-1}{x+1}$ is a solution if $b = c = 0$. □

Theorem 5.3 allows to devise division refinement algorithms for any kind of *egt* polynomials, providing an antidote to the loss of accuracy near the fixed points of $f(x)$ in structured root-finders for $f(x)$-twined polynomials [12]. In the following section, we focus on the special case of *egp* polynomials, for which we extensively tested the structured refinement derived above. More generic applications of Theorem 5.3 will be tested in future experiments.

The proof that we gave for Theorem 5.3 also shows the fact, not difficult to see yet interesting, that *f(x)-egt* polynomials of a given grade are an additive

group, i.e. the sum of two polynomials (of the same grade) whose roots have the $\{x, f(x)\}$ symmetry share the same property with the two addends. Moreover, if $f(x) = \frac{dx+b}{cx-d}$, and $\kappa := d^2 + bc \neq 0$,then the *f(x)-egt* polynomials of grade g are isomorphic (as an additive group) to *egκp* polynomials of the same grade.

As an additional corollary, notice that there is no loss of generality in assuming $\kappa = 1$ in the previous proof: it suffices to start from $f(x) = \frac{dx/\sqrt{\kappa}+b/\sqrt{\kappa}}{cx/\sqrt{\kappa}-d/\sqrt{\kappa}}$. Thus, all the *f(x)-egt* polynomials of grade $2m$ are isomorphic to *egp* polynomials of the same grade.

5.4 An Iterative Refinement Algorithm

The symmetric division processes (5.1), (5.4), (5.6) and (5.5) are at the basis of iterative methods for the refinement of a quadratic factor of *egp* polynomials. For the applications that we have in mind, we are mainly interested in the case where most of the quadratic factors are known at high accuracy while some roots are to be refined.

More precisely, given $a(z) = \sum_{j=0}^{n} a_j z^j = \prod_{j=1}^{m}(z - \xi_j)(z - \xi_j^{-1})$, with $\xi_j + \xi_j^{-1} \neq \xi_k + \xi_k^{-1}$ for $j \neq k$ and $\xi_j + \xi_j^{-1} \neq 0$ for $1 \leq j \leq m$, suppose that an approximation $p(z) = \prod_{j=1}^{k}(z - \hat{\xi}_j)(z - \hat{\xi}_j^{-1})$, $k < m$, of a factor of $a(z)$ is known and then set $s(z) = p(z)(z^{n-2k} + 1)$. Let Ω be the open subset of $\mathbb{C}$ defined by $\Omega = \{z \in \mathbb{C}\colon s(z) \neq 0\}$. From Theorem 5.2 there is defined a function $r = r(\xi)\colon \Omega \to \mathbb{C}$ such that

$$a(z) = q(z)(z - \xi)(z - \xi^{-1}) + rs(z),$$

which gives

$$r(\xi) = \frac{a(\xi)}{p(\xi)(\xi^{n-2k} + 1)}.$$

Let us now apply the Newton method to pursue the solution of the nonlinear equation $r(\xi) = 0$. This yields the iteration

$$\xi_{\ell+1} = \xi_\ell - \frac{a(\xi_\ell)/a'(\xi_\ell)}{1 - \frac{a(\xi_\ell)}{a'(\xi_\ell)}\left(\sum_{j=1}^{k} \frac{2\xi_\ell + \alpha_j}{\xi_\ell^2 + \alpha_j\xi_\ell + 1} + \sum_{j=1}^{m-k} \frac{2\xi_\ell + \theta_j}{\xi_\ell^2 + \theta_j\xi_\ell + 1}\right)}, \quad \ell \geq 0, \tag{5.10}$$

where $\alpha_j = -(\hat{\xi}_j + \hat{\xi}_j^{-1})$ for $j = 1, \ldots, k$, and $\theta_j = -2\cos\frac{(2j-1)\pi}{j}$, for $j = 1, \ldots, m-k$.

The arithmetic cost per step is determined by the evaluation of the Newton correction $\mathcal{N}(\xi) = a(\xi)/a'(\xi)$, which can be performed in several different ways depending on the representation of the polynomial $a(z)$. If $a(z)$ is not the by-product of a PEP but it is a genuinely scalar problem, and if it is given by its coefficients a_j with respect to the standard power basis then the Horner method can be applied with linear complexity. Differently, if $a(z)$ is implicitly specified as the determinant of a linear matrix polynomial, as it happens in the EAI algorithm via linearization, say $a(z) = \det(A + zB)$, $A, B \in \mathbb{C}^{nk\times nk}$, then the Jacobi formula $N(\xi) = (\mathrm{tr}((A + \xi B)^{-1} \cdot B))^{-1}$ provides the correction at the cost of computing the trace of the matrix $(A + \xi B)^{-1}B$. The same argument

applies if, as in the EAI algorithm without linearization, $a(z) = \det P(z)$ for a generic (even-grade or even-dimensional) matrix polynomial.

Several numerical tests have been performed in order to investigate the robustness and the accuracy of the proposed iteration (5.10) applied for the refinement of the solutions of $a(\xi) = 0$. In particular, we are interested:

- in the behaviour of the iterative method around $\alpha = -(\xi + \xi^{-1}) = \pm 2$, which are critical values for two of the structured polynomial eigenvalue-finding algorithms devised in Chapter 4;
- in the use of the method for the refinement of both small and large roots, which in principle may be computed with worse accuracy when the method of the explicit construction of a new skew-Hamiltonian matrix polynomial is used (see Chapter 4).

The first example is taken from [68]. The matrix pencil $A - zA^T$, where

$$A = \begin{bmatrix} 1 & \rho - 1 & 0 & 0 \\ 1 - \rho & 1 & \sqrt{-1} & 0 \\ 0 & -\sqrt{-1} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad 0 < \rho \leq 1,$$

has generalised eigenvalues given by

$$\lambda_1 = \lambda_2 = 1, \quad \lambda_3(\rho) = \frac{1 + \sqrt{2\rho - \rho^2}}{1 - \sqrt{2\rho - \rho^2}}, \quad \lambda_4(\rho) = \frac{1 - \sqrt{2\rho - \rho^2}}{1 + \sqrt{2\rho - \rho^2}}.$$

As ρ approaches zero the structured eigenvalue condition numbers tend to become smaller than the unstructured ones and, therefore, it can be expected that the process (5.10) is able to refine the approximations returned by the Matlab function `eig`.

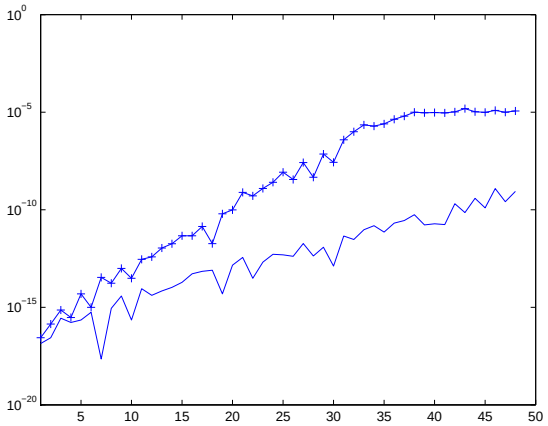

Fig. 35. *Plots of the errors* $\boldsymbol{e_r}(k)$ *(solid line) and* $\boldsymbol{e_m}(k)$ *(dotted line) computed by using the Gaussian elimination method*

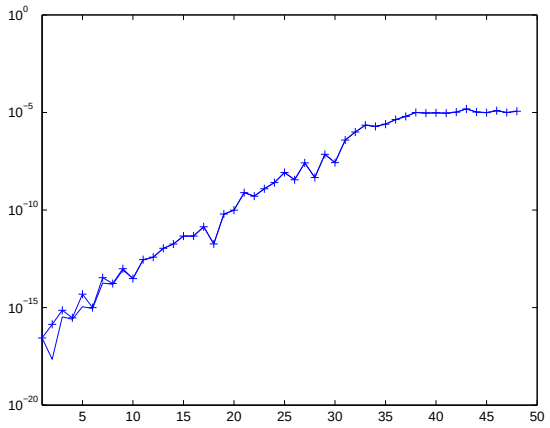

Fig. 36. *Plots of the errors* $\boldsymbol{e_r}(k)$ *(solid line) and* $\boldsymbol{e_m}(k)$ *(dotted line) computed by using the QR factorization method*

Figure 35 shows the plots of the errors $\boldsymbol{e_m}(k) = |\lambda_4(2^{-k}) - \hat{\lambda}_4(2^{-k})|$ and $\boldsymbol{e_r}(k) = |\lambda_4(2^{-k}) - \tilde{\lambda}_4(2^{-k})|$, $1 \le k \le 48$, where $\hat{\lambda}(\rho)$ and $\tilde{\lambda}(\rho)$ are the approximations returned, respectively, by `eig` and the refinement method (5.10) applied with initial guess $\xi_0 = \hat{\lambda}(\rho)$ and $p(z) = (z-1)^2$. Here the Newton correction $\mathcal{N}(\xi) = a(\xi)/a'(\xi)$ is evaluated by means of Gaussian elimination applied to the pencil $A - zA^T$. The plot of $\boldsymbol{e_r}(k)$ is comparable with the error plot generated using the palindromic QR method in [68].

A different scenario is observed if Gaussian elimination is replaced by the QR factorization process. Figure 36 is the same as Figure 35 with the use of the QR method for evaluating the Newton correction.

The difference is due to the greater accuracy of the Gaussian elimination method as compared with the QR factorization method for evaluating the Newton correction $N(\xi) = a(\xi)/a'(\xi)$. Figure 37 reports the plots of the absolute errors computed at the nodes $\xi_k = 1 + \sqrt{\epsilon}\, e^{\mathrm{i}2\pi(k-1)/48}$, for $1 \le k \le 48$.

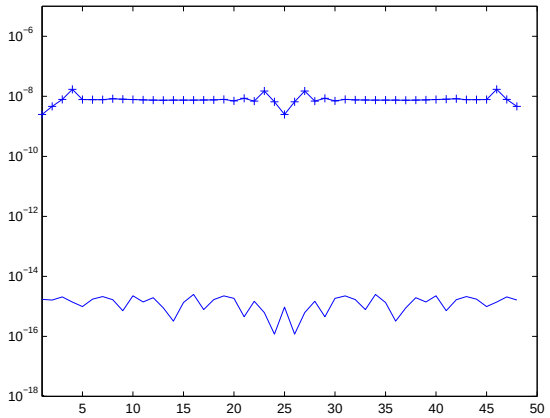

Fig. 37. *Plots of the absolute errors* $\boldsymbol{e_{GE}}(k)$ *(solid line) and* $\boldsymbol{e_{QR}}(k)$ *(dotted line) generated in the computation of the Newton corrections* $N(\xi_k) = a(\xi_k)/a'(\xi_k)$, $1 \le k \le 48$

The second numerical test is the generalised eigenvalue problem for the matrix pencil $A - zA^T$ where

$$A = \begin{bmatrix} -2/\alpha - 2\alpha & 2/\alpha + 2\alpha & 2/\alpha - 2\alpha - 4\beta & 2/\alpha - 2\alpha + 4\beta \\ 2/\alpha + 2\alpha & -2/\alpha - 2\alpha & -2/\alpha + 2\alpha - 4\beta & -2/\alpha + 2\alpha + 4\beta \\ -2/\alpha + 2\alpha - 4\beta & 2/\alpha - 2\alpha - 4\beta & 2/\alpha + 2\alpha & 2/\alpha + 2\alpha \\ -2/\alpha + 2\alpha + 4\beta & 2/\alpha - 2\alpha + 4\beta & 2/\alpha + 2\alpha & 2/\alpha + 2\alpha \end{bmatrix},$$

with $\alpha = \sqrt{-1} + r$, $\beta = 1 + r$, $r \in \mathbb{R}$. The exact eigenvalues are

$$\lambda_1(r) = (1+r)^2,\ \lambda_2(r) = 1/\lambda_1(r),\ \lambda_3(r) = (\sqrt{-1}+r)^2,\ \lambda_4(r) = 1/\lambda_3(r).$$

In our numerical experiments we have simulated a small error on the last digit of $\gamma(r) = \lambda_1(r) + \lambda_2(r)$ by setting $\tilde{\gamma}(r) = \gamma(r) + 10 \cdot \epsilon$. We have computed an initial approximation of $\lambda_2(r)$ by solving the quadratic equation $z^2 - \tilde{\gamma}(r)z + 1 = 0$ and then we have used (5.10) with $p(z) = 1$ to refine the value of the root. Figure 38 illustrates the plot of the initial error $\boldsymbol{e_i}(r)$ and of the final error $\boldsymbol{e_r}(r)$ for $r = 4^{-k}$, $k = 1, \ldots, 48$.

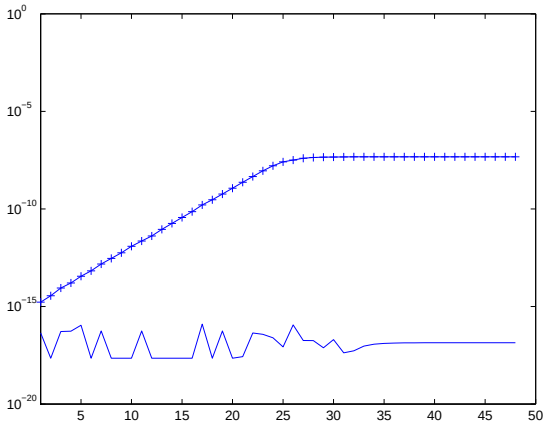

Fig. 38. *Plots of the errors* $\boldsymbol{e_i}(k)$ *(dotted line) and* $\boldsymbol{e_r}(k)$ *(solid line)*

The last example regards the approximation of small and large roots. We first consider the palindromic polynomial $a(z)$ of degree $2k+2$ whose coefficient vector $\boldsymbol{a}$ is generated by the rule $\boldsymbol{a} = [\sqrt{\epsilon}, 1\colon k, 0, k\colon -1\colon 1, \sqrt{\epsilon}]$, $1 \le k \le 48$. The polynomial has one root ξ of order 10^{-8}. The SEAI computes approximations α_j, $1 \le j \le k+1$, such that $a(z) \simeq \sqrt{\epsilon} \prod_{j=1}^{k+1} (z^2 - \alpha_j z + 1)$ and $\alpha_{k+1} \simeq \xi + \xi^{-1}$. Figure 39 shows the absolute error at the end of the iterative process (5.10) applied with $s(z) = (z^2+1) \prod_{j=1}^{k} (z^2 - \alpha_j z + 1)$ for the refinement of ξ. The

starting point is $\xi_0 = 0$ so that the initial absolute error before the application of our algorithm is of order 10^{-8} for any value of k.

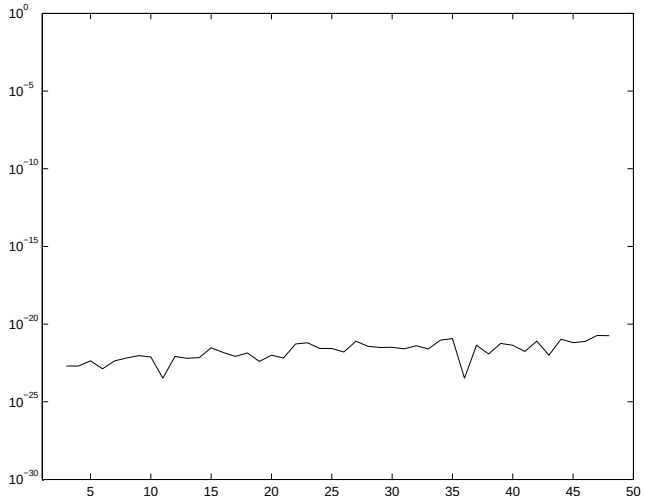

Fig. 39. *Plots of the absolute errors for the case of one small (large) root*

In a different set of experiments the polynomial $a(z)$ has been modified according to the rule $\boldsymbol{a} = [\sqrt{\epsilon}, \epsilon, 1\colon k, 0, k\colon -1\colon 1, \epsilon, \sqrt{\epsilon}]$, $1 \leq k \leq 48$; it has now two distinct roots ξ_1 and ξ_2 of order 10^{-4}. The approximations returned by the algorithm in Chapter 4 are $\alpha_1, \ldots, \alpha_{k+2}$ with $\alpha_{k+1} \simeq \xi_1 + \xi_1^{-1}$ and $\alpha_{k+2} \simeq \xi_2 + \xi_2^{-1}$. Figure 40 shows the error at the end of the iterative process (5.10) applied with $s(z) = (z^2+1)\prod_{j=1}^{k+1}(z^2 - \alpha_j z + 1)$ for the refinement of ξ_2. The starting point is $\xi_0 = 0$ so that the refinement starts with a poor approximation affected by an absolute error of order 10^{-4} for any value of k.

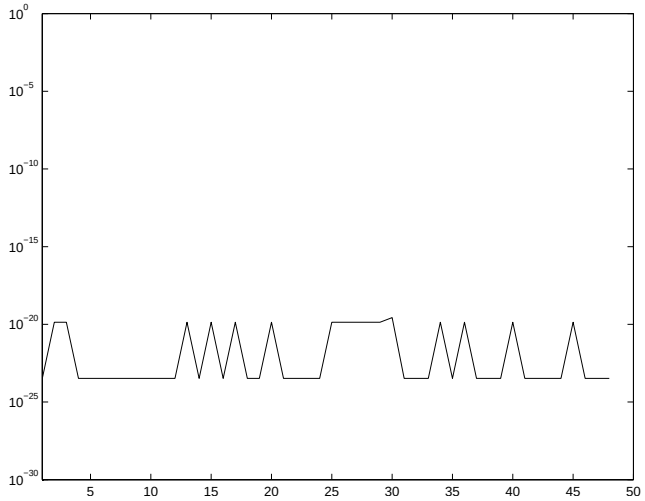

Fig. 40. *Plots of the absolute errors for the case of two small (large) roots*

In both Figures 39 and 40, numerically zero errors were set equal to 10^{-24} in order to show them on the graph.

For comparison, the sequence generated by the classical Newton method applied to the polynomial $a(z)$ of degree 100 with initial guess $\xi_0 = 0$ exhibits a divergent behaviour. This is paradigmatic of the problems which can be encountered by using the Newton method in the occurrence of clustered or even very large (infinite) eigenvalues. Moreover, a classical refinement approach is not able to extract the spectral symmetries.

5.5 Remarks

The proposed modification of the Newton method is able to provide a structured iterative refinement for the roots of a twined polynomial of even grade. When applied to a PEP, the method seems to work better when Gaussian elimination is used instead of QR factorization for evaluating the Newton correction.

In the palindromic case, the refinement method relies upon certain symmetric division processes for palindromic polynomials and it generates a quotient and a remainder with the same property of the coefficients. The performed experiments show that when the approximations of the roots are known then the information can be incorporated into the division process, thus leading to an iterative method in the style of the Ehrlich-Aberth root-finding algorithm. The refinement differs from the original EAI because of the presence of some fixed "artificial" roots, as can be seen by comparing (5.10) and (2.2), (2.3). For *egp* polynomials, such artificial roots are conveniently chosen as proportional to Chebyshev nodes. This can be seen once again as an occurrence of the Dickson (or, almost equivalently, Chebyshev) basis in the treatment of palindromic polynomials.

Analogous algorithms can be used to treat κ-palindromic polynomials and, thanks to the isomorphism described in Theorem 5.3, also for generic *f(x)-egt* polynomials.

An important consequence of the results presented in the present chapter is that the known inconvenience caused by the back-transformation of the variable in some implementations of the SEAI can be effectively amended by these structured division algorithms, which therefore provide the sought structured refinement for near-to-exceptional eigenvalues.

We can conclude that the approach resulting from the combination of the SEAI and the structured division refinement algorithm is very effective and can reach at least the same level of accuracy that the structure-preserving methods based on matrix iterations achieve. Moreover, we stress that the structured division refinement algorithm is independent of the SEAI and can be used to refine approximations computed with any method. Even more, in principle it may be used to impose the spectral structure to a set of approximations obtained by an unstructured algorithm: suppose for instance to use first the UEAI, or the unstructured QZ, and to sort the obtained approximations. There will be pairs $\{\lambda, \mu\}$ with $\mu \simeq f(\lambda)$; one can thus start with either of the two, say λ, and apply the structured division refinement algorithm starting from $\{\lambda, f(\lambda)\}$, until an exactly paired eigencouple $\{\lambda_r, f(\lambda_r)\}$ is obtained.

The method achieves good results also in cases where a classical Newton refinement approach fails.

Chapter 6

Conclusions and future research

In this chapter we wish to give some final remarks on the analysis made in this work.

6.1 The EAI for PEPs: an overview

The Ehrlich-Aberth iteration is a root-finding algorithm that can be used as a tool for a root-finding approach to polynomial eigenvalue problems. The resulting algorithm outperforms customary methods in terms of computation time if the degree of the matrix polynomial is high, while it is slower (although of comparable asymptotic complexity) for the large dimension case. To our knowledge, no proof of global convergence is available. Yet, in practice the method appears reliable and effective. A comparison with backward stable methods such as the QZ algorithm show that the EAI compute eigenvalues with forward errors at least comparable with the ones obtained by robust implementation of the QZ. Often, the approximations given by the EAI are more accurate. This means that, also in the situations where it is not faster than the matrix iteration methods, the EAI could be used as a refinement algorithm to improve the output of customary algorithms. It is worth to add that the SEAI, in any of its variants described in this thesis, can also be used in order to impose the right structure to the output of an unstructured method.

The main weakness of the method at its current stage of development is that, at least in principle, it may have problems to deal with multiple eigenvalues. Convergence is slower for multiple eigenvalues, and if they are defective it is difficult to design an appropriate stopping criterion. If any *a priori* knowledge exist on multiple eigenvalues, it should be used to deflate such roots. For the NLEVP library test problems, the presence of multiple zero or infinite eigenvalues could be easily forecast, at least with a lower bound on their number, by the analysis of the extremal coefficients. The deflation of such eigenvalues helped improving the efficiency of the algorithm: for those problems, very satisfactory results were achieved in terms of accuracy despite the presence, in some of them, of defective eigenvalues.

For the case of a structured matrix polynomial, we have proposed three

different approaches in order to adapt the EAI and to extract the spectral structure. A common positive feature of such methods is that, in contrast with many structured variants of the QZ or other iterative methods that work on a linearised pencils, the complexity is never increased with respect to the unstructured algorithm; on the contrary, the number of the sought roots is halved.

One of the three SEAI methods, the so-called naive approach, is very simply implemented, and it is the only one that at least in principle could be also used for non-analytic self-inverse functions $f(x)$ (e.g., a real matrix polynomial with $f(x) = x^*$.) Since for non-analytic $f(x)$ there is usually no way to predict a priori the number of exceptional eigenvalues (e.g. the number of real eigenvalues for a generic real matrix polynomial), such an application is however possible only with the support of a multiprecision cluster analysis algorithm, which —at least at the moment— is only available for the scalar case. Nevertheless, it may lead in the next future to the design of an improvement of the software `MPSolve` [9] in the case of structured scalar polynomials (e.g. real polynomials: in contrast to what happens now within the software, it could be possible to guarantee that complex eigenvalues will be paired and real eigenvalues will be computed with exactly zero imaginary part). The drawback of the naive approach is that for some class of matrix polynomials (e.g. palindromic) it appears to be slower in terms of needed scalar iterations.

The other two approaches, i.e. the mapping of the structured PEP to a related skew-Hamiltonian eigenvalue problem and the implicit change of variable method, are also implemented with the same asymptotic complexity of the unstructured EAI. Also for these strategies, an implementation within `MPSolve` is possible. The weakness of these two more sophisticated approaches is the possible loss of accuracy for exceptional eigenvalues in the cases where the back transformation is ill-conditioned. The existence of structured refinement algorithms, such as the Newton refinement method that we have described in this thesis, is an effective remedy to this problem. As a future research project, we also plan to extend the refinement method to non-analytic self-inverse functions, also in the matrix case.

6.2 Future research

We list here some topics that are currently included amongst the future research projects that either we are currently working on or we plan to work on in the nearest future.

- The exploitation of our result on the structured case for the development of new features in `MPSolve`. This has been discussed in the previous section.
- The generalization of the Newton refinement method to the non-analytic case (e.g. H-palindromic matrix polynomials). Notice that the ideas developed in Chapter 5 are in some sense similar to those in Bairstow method [6], an algorithm that approximates quadratic factors for a real polynomial ($f(x) = x^*$). Further research is needed to investigate what happens more in general when the self-inverse function $f(x)$ is not analytic.
- The development of a better strategy to choose starting points. Our current strategy is only partially satisfactory, since when n/k^2 is high the

total number of scalar iteration t is slightly superlinear in nk. Some new results on eigenvalue localization might come using tools from tropical algebra, by generalising the work of [101], or by other techniques. This will be the subject of the paper [13], in preparation at the moment.

- The implementation of the version of the EAI for high values of n/k^2, with the strategy described in Section 2.2.3.

- The study of the possibility to use different root-finding algorithms. We have mainly focused on the Ehrlich-Aberth method because in the scalar case it proved itself the most efficient and stable algorithm to simultaneously approximate all the roots. Nevertheless, other choices do exist, e.g. the Durand-Kerner, the modified EAI and the Halley-like iterations, all described in Section 2.7. A thorough comparison of such alternatives with the EAI in the various special cases is ongoing.

- The direct application of the EAI to genuinely nonlinear eigenvalue problems. Of course, a nonlinear eigenvalue problem can in principle be approximated with a polynomial eigenvalue problem, or with a rational eigenvalue problem that can be turned into a PEP after multiplication by least common denominator. Due to possible numerical issues, some caution may be needed for the choice of what polynomial or rational approximant should be used; see for instance discussions in [35, 66] for the case of an exponential function. Such approximations, however, may have drawbacks, and the direct approach may give better results. Current research is focused on dealing with the difficulties coming from the possible infinite number of eigenvalues when the problem is genuinely nonlinear.

Research on polynomial eigenvalue problems can currently be considered one of the hottest topics in linear algebra. As an example of a recent achievement in the field, we wish to conclude this chapter mentioning a beautiful result that very recently appeared in [106]. A result hidden in the proof of [44, Theorem 1.7] was there extended from monic to regular matrix polynomials, showing that the latter are always triangularisable over $\mathbb{C}$ via unimodular equivalence, i.e. they can be transformed to triangular matrix polynomials of the same degree. Also, Schur-like theorems are proved. The authors of [106] mention that their current research is focused on developing algorithms that implement the triangularization for the quadratic case [91]. If it were possible to triangularise matrix polynomials, or at least to bring them to Hessenberg form, with a stable and cheap numerical algorithm also when the degree is high, then it would be easy to transform a PEP to n independent scalar polynomial equations of high degree, for which the EAI can be very efficiently implemented.

Acknowledgements

Since before enrolling in the Ph.D. program in Pisa I had graduated and done research in theoretical physics, I have come in touch quite recently with the linear algebra scientific community. After getting to meet some of the persons of such community, I could not be happier with the choice I have made when I got interested in this field. Many people welcomed me in the community during meetings and conferences, and encouraged me, and many people could find the time to discuss topics with me and sometimes to give me suggestions. My gratitude goes to all of them.

I wish to thank, in particular, SIAM, the organisers of the Gene Golub International Summer School in Fasano, the four bright lecturers that in many instances inspired me with beautiful ideas and suggestions, all the other students present there, and of course the people at the tennis club. The two weeks I could spend there were priceless under all the possible points of view.

I am also in debt with some anonymous referees, who provided useful suggestions and intelligent remarks after carefully reading manuscripts submitted for publication. I wish to thank the two referees of this thesis as well for their comments and suggestions.

Not every Ph.D. student in the academic world is happy with his/her advisor. This means I have been very lucky, since I picked two wonderful ones. The scientific competence of both Dario and Luca is indubitable and made our research exciting, but the support and friendship that I received during this period spent in Pisa is beyond any academic duty. I really thank them for this.

My M.Sc. advisor Filippo was also very supportive, and we kept exchanging nice chats. He was also one of the two coauthors in my very first peer-reviewed paper (on some algebraic curves appearing in a model in theoretical physics: it has nothing to do with numerical solution of PEPs!), whose publication was a good thing for my morale.

Finally, any professional task would be much harder without my friends, my family, Martina, and anyone else who has a special role in my private life.

Bibliography

[1] O. Aberth. Iteration methods for finding all zeros of a polynomial simultaneously. *Math. Comput*, 27:339–344, 1973.

[2] M. Al-Ammari and F. Tisseur. Standard triples of structured matrix polynomials. To appear in *Linear Algebra Appl.*

[3] A. Amiraslani, R. M. Corless and P. Lancaster. Linearization of matrix polynomials expressed in polynomial bases. *IMA J. Numer. Anal.*, 29(1):141–157, 2009.

[4] B. D. O. Anderson. Second-order convergent algorithms for the steady-state Riccati equation. *Internat. J. Control*, 28(2):295–306, 1978.

[5] E. N. Antoniou and S. Vologiannidis. A new family of companion forms of polynomial matrices. *Electron. J. Linear Algebra*, 11:78–87, 2004.

[6] L. Bairstow. *Applied aerodynamics*. Longmans, Green & Co., London, 1920.

[7] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder and F. Tisseur. NLEVP: a collection of nonlinear eigenvalue problems. MIMS EPrint 2010.98, 2010. Avaliable online, http://www.mims.manchester.ac.uk/research/numerical-analysis/nlevp.html

[8] D. A. Bini. Numerical computation of polynomial zeros by means of Aberths method. *Numer. Algorithms*, 13:179200, 1996.

[9] D. A. Bini and G. Fiorentino. Design, analysis, and implementation of a multiprecision polynomial rootfinder. *Numer. Algorithms*, 23(2-3):127–173, 2000.

[10] D. A. Bini, L. Gemignani and F. Tisseur. The Ehrlich-Aberth method for the nonsymmetric tridiagonal eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 27(1):153–175, 2005.

[11] D. A. Bini and B. Meini. The cyclic reduction algorithm: from Poisson equation to stochastic processes and beyond. In memoriam of Gene H. Golub. *Numer. Algorithms*, 51(1):23–60, 2009.

[12] D. A. Bini and V. Noferini. Solving polynomial eigenvalue problems by means of the Ehrlich-Aberth method. Submitted.

[13] D. A. Bini, V. Noferini and M. Sharify. Locating the eigenvalues of matrix polynomials. Submitted.

[14] W. Borsch-Supan. A posteriori error bounds for the zeros of polynomials. *Numer. Math.*, 5:380–398, 1963.

[15] R. P. Boyer and W. M. Y. Goh. Partition polynomials: asymptotics and zeros. Tapas in experimental mathematics, *Contemp. Math.*, 457:99-111,, Amer. Math. Soc., Providence, RI, 2008.

[16] M. I. Bueno, F. De Terán and F. M. Dopico. Recovery of eigenvectors and minimal bases of matrix polynomials from generalized Fiedler linearizations. *SIAM J. Matrix Anal. Appl.* 32:463–483, 2011.

[17] B. L. Buzbee, G. H. Golub and C. W. Nielson. On direct methods for solving Poisson's equations. *SIAM J. Numer. Anal.* 7:627–656, 1970.

[18] E. K. W. Chu, T. Li and W. W. Lin. A structure-preserving doubling algorithm for quadratic eigenvalue problems arising from time-delay systems. *J. Comp. Appl. Math.*, 223:1799–1745, 2010.

[19] C. M. Da Fonseca and J. Petronilho. Explicit inverses of some tridiagonal matrices. *Linear Algebra Appl.*, 325(1-3):7–21, 2001.

[20] S. Delvaux and M. Van Barel. A Hessenberg reduction algorithm for rank structured matrices. *SIAM J. Matrix Anal. Appl.*, 29:895–926, 2007.

[21] F. De Terán and F. M. Dopico. Sharp lower bounds for the dimension of linearizations of matrix polynomials. *Electron. J. Linear Algebra*, 17:518–531, 2008.

[22] F. De Terán, F. M. Dopico and D. S. Mackey. Fiedler companion linearizations and the recovery of minimal indices. *SIAM J. Matrix Anal. Appl.* 31:2181–2204, 2010.

[23] F. De Terán, F. M. Dopico and D. S. Mackey. Fiedler companion linearizations for rectangular matrix polynomials. Submitted. Avaliable online at http://gauss.uc3m.es/web/personal_web/fdopico/papers/laa2011-fiedler-rect.pdf

[24] F. De Terán, F. M. Dopico and D. S. Mackey. Linearizations of singular matrix polynomials and the recovery of minimal indices. *Electron. J. Linear Algebra*, 18:371–402, 2009.

[25] F. De Terán, F. M. Dopico and D. S. Mackey. Palindromic companion forms for matrix polynomials of odd degree. *J. Comput. Appl. Math.*, 236:1464–1480, 2011.

[26] L. E. Dickson. The analytic representation of substitutions on a power of a prime number of letters with a discussion of the linear group. *Ann. of Math.*, 11:65–120, 11:161–183, 1897.

[27] L. W. Ehrlich, A modified Newton method for polynomials. *Comm. ACM*, 10(2):107–108, 1967.

[28] Y. Eidelman, I. Gohberg and L. Gemignani. On the fast reduction of a quasiseparable matrix to Hessenberg and tridiagonal forms. *Linear Algebra Appl.*, 420:86–101, 2007.

[29] H.-Y. Fan, W.-W. Lin and P. Van Dooren. Normwise scaling of second order polynomial matrices. *SIAM J. Matrix Anal. Appl.*, 26:252–256, 2004.

[30] H. Faßbender. *Symplectic methods for the symplectic eigenproblem*. Kluwer, New York, 2002.

[31] H. Faßbender and K. D. Ikramov. Several observations on symplectic, Hamiltonian, and skew-Hamiltonian matrices. *Linear Algebra Appl.*, 422(1):29–38, 2007.

[32] H. Faßbender, D. S. Mackey, N. Mackey and H. Xu. Hamiltonian square roots of Skew-Hamiltonian matrices. *Linear Algebra Appl.*, 287:125–159, 1999.

[33] M. Fiedler. A note on companion matrices. *Linear Algebra Appl.*, 372: 325–331, 2003.

[34] G. D. Forney Jr. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM J. Control*, 13(3):493–520, 1975.

[35] G. E. Forsythe, M. A. Malcolm and C. B. Moler. *Computer methods for mathematical computations*. Prentice Hall, Englewood Cliffs, 1977.

[36] G. Frobenius. Theorie der linearen Formen mit ganzen Coefficienten. *J. Reine Angew. Math. (Crelle)*, 86:146–208, 1878.

[37] W. Gander. Zeros of determinants of λ-matrices. In V. Olshevsky and E. Tyrtyshnikov, editors, *Matrix methods: theory, algorithms and applications, Dedicated to the memory of G. Golub*. World Scientific Publ., Singapore, 2010.

[38] F. R. Gantmacher. *The theory of matrices*. AMS Chelsea, Providence, 1998.

[39] S. D. Garvey, N. Higham, D. S. Mackey and F. Tisseur. Scaling, sensitivity and stability in the numerical solution of quadratic eigenvalue problems. *Internat. J. Numer. Methods Eng.*, 73(3):344–360, 2008.

[40] S. D. Garvey, C. Munro and F. Tisseur. Deflating quadratic matrix polynomials with structure preserving transformations. *Linear Algebra Appl.*, 435:464–479, 2011.

[41] L. Gemignani and V. Noferini. Modifications of Newton's method for even-grade palindromic polynomials and other twined polynomials. *Numer. Algorithms*, 61(2):1–15, 2012.

[42] L. Gemignani and V. Noferini. The Ehrlich-Aberth method for palindromic matrix polynomials represented in the Dickson basis. *Linear Algebra Appl.* 2011, doi: 10.1016/j.laa.2011.10.035

[43] I. Gohberg, M. A. Kaashoek and P. Lancaster. General theory of regular matrix polynomials and band Toeplitz operators. *Integral Equations Operator Theory*, 11(6):776–882, 1988.

[44] I. Gohberg, P. Lancaster and L. Rodman. *Matrix polynomials*. Academic Press, New York, 1982.

[45] M. A. Golberg The derivative of a determinant. *The American Math. Montly*, 79:1124–1126, 1972.

[46] G. H. Golub and H. A. van der Vorst. Eigenvalue computation in the 20th century. *J. Comput. Appl. Math.*, 123:35–65, 2000.

[47] G. H. Golub and C. F. Van Loan. *Matrix computations*. The John Hopkins University Press, 1996.

[48] T. N. T. Goodman, C. A. Micchelli, G. Rodriguez, and S. Seatzu. Spectral factorization of Laurent polynomials. *Adv. Comput. Math.*, 7(4):429–454, 1997.

[49] C.-H. Guo, N. Higham and F. Tisseur. Detecting and solving hyperbolic quadratic eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 30(4):1593–1613, 2009.

[50] C. H. Guo and W. W. Lin. Solving a structured quadratic eigenvalue problem by a structure-preserving doubling algorithm. *SIAM Matrix Anal. Appl.* 31(5):2784–2801, 2010.

[51] H. Guggenheimer. Initial approximations in Durand-Kerner's root finding method. *BIT* 26(4):537–539, 1986.

[52] S. Hammarling, C. Munro and F. Tisseur. An algorithm for the complete solution of quadratic eigenvalue problems. MIMS EPrint 2011.86. Avaliable online, http://eprints.ma.man.ac.uk/1730/01/covered/MIMS_ep2011_86_MATLAB_M-file.pdf

[53] G. E. Hayton, A. C. Pugh and P. Fretwell. Infinite elementary divisors of a matrix polynomial and implications. *Internat. J. Control*, 47:53–64, 1988.

[54] P. Henrici. *Applied and computational complex analysis*. Vol. 1. Wiley-Interscience, John Wiley & Sons, New York-London-Sydney, 1974

[55] D. J. Higham and N. J. Higham. Structured backward error and condition of generalized eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 20(2):493–512 (electronic), 1999.

[56] N. J. Higham. *Accuracy and stability of numerical algorithms*. SIAM, Philadelphia, 2002.

[57] N. J. Higham. *Functions of matrices: theory and computation*. SIAM, Philadelphia, 2008.

[58] N. J. Higham, R.-C. Li and F. Tisseur. Backward error of polynomial eigenproblems solved by linearization. *SIAM J. Matrix Anal. Appl.*, 29(4):1218–1241, 2007.

[59] N. J. Higham, D. S. Mackey, N. Mackey and F. Tisseur. Symmetric linearizations for matrix polynomials. *SIAM J. Matrix Anal. Appl.*, 29(1):143–159, 2006.

[60] N. J. Higham, D. S. Mackey and F. Tisseur. The conditioning of linearizations of matrix polynomials. *SIAM J. Matrix Anal. Appl.*, 28(4):1005–1028, 2006.

[61] M. Hochstenbach, T. Košir and B. Plestenjak. A Jacobi-Davidson type method for a right definite two-parameter eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 26:477–497, 2005.

[62] J. Hubbard, D. Schleicher and S. Sutherland. How to find all roots of complex polynomials by Newton's method. *Invent. Math.*, 146:1–33, 2001.

[63] T. M. Hwang, W. W. Lin, W. C. Wang and W. Wang. Numerical simulation of three-dimensional pyramid quantum dot. *J. Comput. Phys.*, 196:208–232, 2004.

[64] N. K. Jain and K. Singhal. On Kublanovskaya's approach to the solution of the generalized latent value problem for functional λ-matrices. *SIAM J. Numer. Anal.*, 20(5) :1062–1070, 1983.

[65] N. K. Jain, K. Singhal and K. Huseyin. On roots of functional lambda matrices. *Comput. Meth. Appl. Mech. Engrg.*, 40:277–292, 1983.

[66] E. Jarlebring. The spectrum of delay-differential equations: numerical methods, stability and perturbation. Ph.D. thesis, Inst. Comp. Math, TU Braunschweig, 2008

[67] D. Kressner. *Numerical methods for general and structured eigenvalue problems*. Lecture notes in Computational Science and Engineering, 46. Springer, Berlin, 2000.

[68] D. Kressner, C. Schroeder and D. S. Watkins. Implicit QR algorithms for palindromic and even eigenvalue problems. *Numer. Algorithms*, 51(2):209–238, 2009.

[69] V. Kublanovskaya. On an approach to the solution of the generalized latent value problem for λ-matrices. *SIAM J. Numer. Anal.*, 7 :532–537, 1970.

[70] P. Lancaster. *Lambda-matrices and vibrating systems*. Pergamon Press, Oxford, 1966.

[71] P. Lancaster, U. Prells and L. Rodman. Canonical structures for palindromic matrix polynomials. *Oper. Matrices*, 1:469–489, 2007.

[72] P. Lancaster and L. Rodman. Canonical forms for symmetric/skew-symmetric real matrix pairs under strict equivalence and congruence. *Linear Algebra Appl.*, 406:1–76, 2005.

[73] M. Lang and B. C. Frenzel. Polynomial root finding. *IEEE Signal Process. Letters*, 1(10), 1994.

[74] S. Lang. *Algebra*. Springer-Verlag, New York, 2002.

[75] D. S. Mackey, N. Mackey, C. Mehl and V. Mehrmann. Numerical methods for palindromic eigenvalue problems: Computing the anti-triangular Schur form. *Numerical Linear Algebra with Appl.*, 16:63–86, 2009.

[76] D. S. Mackey, N. Mackey, C. Mehl and V. Mehrmann. Smith forms of palindromic matrix polynomials. *Electron. J. Linear Algebra*, 22:53–91, 2011.

[77] D. S. Mackey, N. Mackey, C. Mehl and V. Mehrmann. Structured polynomial eigenvalue problems: good vibrations from good linearizations. *SIAM J. Matrix Anal. Appl.*, 28(4):1029–1051 (electronic), 2006.

[78] D. S. Mackey, N. Mackey, C. Mehl and V. Mehrmann. Vector spaces of linearizations for matrix polynomials. *SIAM J. Matrix Anal. Appl.*, 28(4):971–1004 (electronic), 2006.

[79] D. S. Mackey, N. Mackey and F. Tisseur. Structured factorizations in scalar product spaces. *SIAM J. Matrix Anal. Appl.*, 27(3):821–850 (electronic), 2006.

[80] N. Mackey. Moebius transformations of matrix polynomials. *Talk given on Friday, July 22nd 2011*, 7th International Congress on Industrial and Applied Mathematics, Vancouver.

[81] J. M. MacNamee. *Numerical methods for roots of polynomials.* Studies in Computational Mathematics, 14. Elsevier, Amsterdam, 2007.

[82] H. Maehly. Zur Iterativen Auflösung Algebraisher Gleichungen. *Z. Angew. Math. Physik* 5:260–263, 1954.

[83] C. Mehl. Condensed forms for skew-Hamiltonian/Hamiltonian pencils. *SIAM J. Matrix Anal. Appl.*, 21:454–476, 1999.

[84] C. Mehl. Jacobi-like algorithms for the indefinite generalized Hermitian eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 25:964–985, 2004.

[85] V. Mehrmann and H. Voss. Nonlinear eigenvalue problems: a challenge for modern eigenvalue methods. *GAMM Mitt. Ges. Angew. Math. Mech.*, 27:121–151, 2005.

[86] V. Mehrmann and D. Watkins. Structure-preserving methods for computing eigenpairs of large sparse skew-Hamiltonian/Hamiltonian pencils. *SIAM J. Sci. Comp.*, 22:1905–1925, 2001.

[87] V. Mehrmann and D. Watkins. Polynomial eigenvalue problems with Hamiltonian structure. *Electr. Trans. Num. Anal.*, 13:106–113, 2002.

[88] C. B. Moler and G. W. Stewart. An algorithm for generalized matrix eigenvalue problems. *SIAM J. Numer. Anal.* 10(2):241–256 (1973).

[89] Y. Monden and S. Arimoto. Generalized Rouche's theorem and its applications to multivariate autoregressions. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-28:733–738, 1980.

[90] T. Muir. *A treatise on the theory of determinants.* Macmillan & Co., London, 1882.

[91] Y. Nakatsukasa, L. Taslaman, F. Tisseur and I. Zaballa. A Schur approach to triangularize quadratic matrix polynomials. In preparation.

[92] V. Noferini. The application of the Ehrlich-Aberth method to structured polynomial eigenvalue problems. *Proc. Appl. Math. Mech.*, 11:919–922, 2011.

[93] V. Noferini. The behaviour of the complete eigenstructure of a polynomial matrix under a generic rational transformation. *Electron. J. Linear Algebra*, 23:607–624, 2012.

[94] A.-W. M. Nourein. An iterative formula for the simultaneous determination of the zeros of a polynomial. *J. Comput. Appl. Math.*, 1(4): 251–254, 1975.

[95] N. Papathanasiou and P. Psarrakos. On condition numbers of polynomial eigenvalue problems. *Appl. Math. Comput.*, 216(4):1194–1205, 2010.

[96] M. Petkovic. *Point estimation of root finding methods*. Lecture Notes in Mathematics, 1933. Springer-Verlag, Berlin Heidelberg, 2008.

[97] B. Plestenjak. Numerical method for the tridiagonal hyperbolic quadratic eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 28(4):1157–1172, 2006

[98] A. Ruhe. Algorithms for the nonlinear eigenvalue problem. *SIAM J. Numer. Anal.*, 10:674–689, 1973.

[99] P. Saux Picart and C. Brunie. Symmetric subresultants and applications. *J. Symbolic Comput.*, 42(9):884–919, 2007.

[100] C. Schröder. Palindromic and even eigenvalue problems - Analysis and numerical methods. Ph.D. thesis, Technischen Universitat Berlin, April 2008.

[101] M. Sharify. Scaling algorithms and tropical methods in numerical matrix analysis: application to the optimal assignment problem and to the accurate computation of eigenvalues. Ph.D. Thesis, Ecóle Polytechnique ParisTech, Spécialité: Mathématiques Appliquées.

[102] B. T. Smith. Error bounds for zeros of a polynomial based upon Gerschgorin's theorems. *JACM*, 17:661–674, 1970.

[103] H. J. S. Smith. On systems of linear indeterminate equations and congruences. *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.*, 151:293–326, 1861.

[104] F. Tisseur. Backward error and condition of polynomial eigenvalue problems. *Linear Algebra Appl.*, 309(1-3):339–361, 2000.

[105] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Rev.*, 43(2):235–286, 2001.

[106] F. Tisseur and I. Zaballa. Triangularizing quadratic matrix polynomials. Preprint, avaliable online at http://eprints.ma.man.ac.uk/1786/01/covered/MIMS_ep2012_30.pdf.

[107] C. Van Loan. A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix. *Linear Algebra Appl.*, 61:233–251, 1984.

[108] S. Warner. *Modern algebra*. Dover, Mineola, 1990.

[109] D. S. Watkins. *The matrix eigenvalue problem — GR and Krylov subspace methods*. SIAM, Philadelphia, 2007.

[110] J. H. Wilkinson. The perfidious polynomial. In G. H. Golub, editor, *Studies in numerical analysis, vol. 24 of Studies in mathematics*. Mathematical association of America, Washington, 1984.

[111] G. T. Wilson, The factorization of matricial spectral densities. *SIAM J. Appl. Math.*, 23:420–426, 1972.