Discrete-Time Symmetric Polynomial Equations with Complex Coefficients*

Didier Henrion[1,2,3]
Jan Ježek[3]
Michael Šebek[3,4]

January 29, 2002

Abstract

Discrete-time symmetric polynomial equations with complex coefficients are studied in the scalar and matrix case. New theoretical results are derived and several algorithms are proposed and evaluated. Polynomial reduction algorithms are first described to study theoretical properties of the equations. Sylvester matrix algorithms are then developed to solve numerically the equations. The algorithms are implemented in the Polynomial Toolbox for Matlab.

Key Words

Polynomial Matrices, Complex Coefficients, Discrete-Time Systems.

[1]Corresponding author. E-mail `henrion@laas.fr`. FAX +33 5 61 33 69 69.

[2]Laboratoire d'Analyse et d'Architecture des Systèmes, Centre National de la Recherche Scientifique, 7 avenue du Colonel Roche, 31 077 Toulouse, cedex 4, France.

[3]Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 182 08 Prague 8, Czech Republic

[4]Center for Applied Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Prague 6, Czech Republic

*This work was supported by the Barrande Project No. 03080XJ/2001-031-1 and by the Ministry of Education of the Czech Republic under contract No. LN00B096.

1 Introduction

Polynomial equations are at the core of the study of dynamical systems, especially when pursuing the polynomial approach to control system analysis and design [14, 15]. Throughout the last two decades, several algorithms were developed to handle scalar or matrix polynomial equations stemming from practical control problems, see [5] for a recent overview. A MATLAB[1] toolbox including several of these algorithms was recently developed [19].

In several problems of signal processing or control design with quadratic criteria, one non-standard operation comes into play in the ring of polynomials, in addition to standard operations such as addition and multiplication. This operation is the conjugation. It is usually associated to the concept of symmetry and gives rise to symmetric polynomial equations. For continuous-time systems, the conjugate of a polynomial is also a polynomial. For discrete-time systems, in order to accomodate the conjugacy and the symmetry, a slightly modified algebra must be introduced: the ring of two-sided polynomials, with both positive and negative powers of the indeterminate. One objective of this paper is precisely to give new insights into this algebra of two-sided discrete-time polynomials.

It must be underlined that polynomial or polynomial matrix coefficients involved in the above mentioned equations are generally supposed to be real, whereas complex numbers may naturally arise when studying real-world problems. Indeed, it is common in communication applications of signal processing to work with complex-valued signals [17, 18]. Such a representation carries information about both amplitude and phase. For example, several mobile radio communication filtering algorithms hinge upon complex Diophantine equations [16]. Polynomials with complex coefficients also occur in the Kharitonov's theorem, a fundamental tool for studying robust stability of linear systems [1, §6.9]. Whirling shafts, vibrational systems and filters are additional examples of systems whose models involve complex coefficients [2]. In the sequel, we describe some problems in mechanics and signal processing that get simplified when complex coefficients come into play. Obviously, no practical extension to complex coefficients of the currently available polynomial equation solvers can be achieved without proper theoretical justification. Up to the authors' knowledge, nothing is available in the linear algebra literature that covers such an extension.

This paper is an attempt to partly fill this gap. It aims at providing a thorough study of a special kind of two-sided discrete-time polynomial equations for which the extension to complex coefficients is by no way straightforward, namely the bilateral symmetric matrix polynomial equation. The problematic feature is mainly the presence of unknown terms at both sides of the equation, some of which are transpose conjugated. As will be shown, this requires a special machinery and new theoretical notions coming from the algebra of complex polynomial matrices.

Previous works on real instances of the symmetric matrix polynomial equation can be traced back to [9], where theoretical properties are investigated in the scalar case. This study is deepened and extended to the matrix case in [10, 11], where some applications

[1]MATLAB is a trademark of the MathWorks, Inc.

to spectral factorization and computation of the covariance matrix of an ARMA process are proposed. Signal processing applications are also to be found in the recent publication [20]. The resolution algorithms described in these references strongly rely upon elementary polynomial operations that have bad reputation as far as numerical properties are concerned. For this reason, the symmetric matrix polynomial equation was recently revisited in discrete-time [4] and continuous-time [6] with the objective of designing numerically reliable algorithms based on Sylvester matrices and interpolation. In the present paper, we extend several of these results to the case of complex coefficients. Polynomial reduction algorithms [9, 10, 11] will be shown to provide constructive proofs of several new theoretical results, whereas Sylvester matrix algorithms [4, 6] will prove to be efficient and numerically reliable alternatives.

The outline of the paper is as follows. Notations and concepts used throughout the paper are collected in Section 2. Systems with complex coefficients naturally arising in mechanics and signal processing are described in Section 3, hence illustrating the point in studying equations with complex coefficients. The symmetric polynomial equation is then first studied in the scalar case (Section 4), then in the matrix case (Section 5). In both sections, two types of algorithms are systematically developed, respectively based on polynomial reductions and Sylvester matrices. Polynomial reduction algorithms are aimed at studying theoretical properties of the equation, namely the existence and the uniqueness of solutions. Sylvester matrix algorithms are aimed at overcoming numerical difficulties that may be faced with polynomial reduction algorithms. They are based on efficient and reliable numerical routines. Both kind of algorithms are illustrated on simple numerical examples. In Section 6, numerical considerations are pointed out. Finally, Section 7 collects some concluding remarks.

2 Notations and Preliminaries

In the paper, we use the following standard notations.

- $\mathbb{C}$ is the field of complex numbers.
- j is the basic imaginary unit in $\mathbb{C}$.
- $\mathrm{Re}A$ is the real part of a complex matrix A.
- $\mathrm{Im}A$ is the imaginary part of a complex matrix A.
- $\overline{A}$ is the complex conjugate of a complex matrix A, i.e. $\overline{A} = \mathrm{Re}A - \mathrm{j}\mathrm{Im}A$.
- A^T is the non-conjugate transpose of a complex matrix A.
- $\mathbf{I}_n$ is the identity matrix of dimension n.
- $\mathbf{0}$ is a matrix with zero elements, whose dimension can be guessed from the context.

Some definitions are also in order.

- A matrix
$$A(\mathsf{z}) = \sum_{i=-m}^{n} A_i \mathsf{z}^i,$$
considered as a function from $\mathbb{C}$ to $\mathbb{C}^{p\times q}$, is referred to as a *two-sided polynomial matrix.*

- A two-sided polynomial matrix $A(\mathsf{z})$ is one-sided, or simply *polynomial*, if it does not feature negative powers of z, i.e. $A_{-m} = A_{-m+1} = \cdots = A_{-1} = \mathbf{0}$.

- The *degree* of a polynomial matrix $A(\mathsf{z})$ is the highest power of z occurring in $A(\mathsf{z})$. It is denoted by δA. For a two-sided polynomial matrix, δA means the same.

- A square polynomial matrix $A(\mathsf{z})$ is *Schur*, or stable in the discrete-time sense, if it has no zero within the closed unit disc, i.e. $\det A(\mathsf{z}) \neq 0$ for all $\mathsf{z} \in \mathbb{C}$ such that $|\mathsf{z}| \leq 1$.

- The complex conjugate of a two-sided polynomial matrix $A(\mathsf{z})$ is
$$\overline{A(\mathsf{z})} = \sum_{i=-m}^{n} \overline{A_i}\,\overline{\mathsf{z}}^i.$$

- The latter matrix is not to be confused with the transpose conjugate of a two-sided polynomial matrix
$$A^\star(\mathsf{z}) = \overline{A^T(\overline{\mathsf{z}}^{-1})} = \sum_{i=-m}^{n} \overline{A_i}^T \mathsf{z}^{-i}.$$

- A square two-sided polynomial matrix $A(\mathsf{z})$ is *symmetric* if it satisfies $A(\mathsf{z}) = A^\star(\mathsf{z})$. Its coefficients satisfy $A_{-i} = \overline{A_i}^T$.

In relation to the above definitions, some clarifying remarks are in order.

- For z on the unit circle, i.e. $\overline{\mathsf{z}} = \mathsf{z}^{-1}$, it holds $A^\star(\mathsf{z}) = \overline{A^T(\mathsf{z})}$.

- One can check that a two-sided polynomial matrix $A(\mathsf{z})$ with *real* coefficients satisfies $A(\overline{\mathsf{z}}) = \overline{A(\mathsf{z})}$. As a consequence, we have $A^\star(\mathsf{z}) = A^T(\mathsf{z}^{-1})$, thus matching the definition found in [10] for real polynomial matrices. However, we underline the fact that this formula *does not hold* for two-sided polynomial matrices with *complex* coefficients.

Finally, we recall a simple and systematic way for converting the *complex* equation
$$\overline{A_1} X + A_2 \overline{X} = B \tag{1}$$
where A_1, A_2, B are given complex matrices of compatible dimensions and X is a complex matrix to be found, into an equation over the field of *real* numbers. Just write equation (1) as
$$(\mathrm{Re}A_1 - \mathrm{j}\mathrm{Im}A_1)(\mathrm{Re}X + \mathrm{j}\mathrm{Im}X) + (\mathrm{Re}A_2 + \mathrm{j}\mathrm{Im}A_2)(\mathrm{Re}X - \mathrm{j}\mathrm{Im}X) = \mathrm{Re}B + \mathrm{j}\mathrm{Im}B.$$

By separating real and complex parts, the above equation reads

$$\begin{aligned}
\mathrm{Re}A_1\mathrm{Re}X + \mathrm{Im}A_1\mathrm{Im}X + \mathrm{Re}A_2\mathrm{Re}X + \mathrm{Im}A_2\mathrm{Im}X &= \mathrm{Re}B \\
-\mathrm{Im}A_1\mathrm{Re}X + \mathrm{Re}A_1\mathrm{Im}X + \mathrm{Im}A_2\mathrm{Re}X - \mathrm{Re}A_2\mathrm{Im}X &= \mathrm{Im}B.
\end{aligned}$$

Now defining

$$\mathbf{A} = \begin{bmatrix} \mathrm{Re}A_1 & \mathrm{Im}A_1 \\ -\mathrm{Im}A_1 & \mathrm{Re}A_1 \end{bmatrix} + \begin{bmatrix} \mathrm{Re}A_2 & \mathrm{Im}A_2 \\ \mathrm{Im}A_2 & -\mathrm{Re}A_2 \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \mathrm{Re}X \\ \mathrm{Im}X \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathrm{Re}B \\ \mathrm{Im}B \end{bmatrix},$$

we obtain the equivalent real matrix equation

$$\mathbf{AX} = \mathbf{B}. \tag{2}$$

Fact 1 *Complex matrix equation (1) and real matrix equation (2) are equivalent.*

3 Dynamical Systems with Complex Coefficients

Usually, dynamical systems are described by differential equations or transfer functions with real coefficients. Sometimes the complex coefficients may prove useful. Two real variables x, y may be replaced by $z = x + \mathrm{j}y$, some operations getting simplified.

3.1 Complex Coefficients in Mechanics

Let us first consider the Coriolis force in mechanics. In the 2-dimensional plane x, y, the law of motion can be written

$$m\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} = f$$

with complex z, f. Suppose the plane x', y' rotates with constant angular velocity ω. Let us derive the law of motion in this plane. The substitution is

$$z = z'\mathrm{e}^{\mathrm{j}\omega t}, \qquad f = f'\mathrm{e}^{\mathrm{j}\omega t}$$

Differentiating

$$\begin{aligned}
\frac{\mathrm{d}z}{\mathrm{d}t} &= \left(\frac{\mathrm{d}z'}{\mathrm{d}t} + \mathrm{j}\omega z'\right)\mathrm{e}^{\mathrm{j}\omega t} \\
\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} &= \left(\frac{\mathrm{d}^2 z'}{\mathrm{d}t^2} + 2\mathrm{j}\omega\frac{\mathrm{d}z'}{\mathrm{d}t} - \omega^2 z'\right)\mathrm{e}^{\mathrm{j}\omega t}
\end{aligned}$$

yields the required law

$$m\frac{\mathrm{d}^2 z'}{\mathrm{d}t^2} = f' - m2\mathrm{j}\omega\frac{\mathrm{d}z'}{\mathrm{d}t} + m\omega^2 z'$$

In the right-hand side, the second term is the Coriolis force, the third one is the centrifugal force.

As an example of a dynamical system, consider a pendulum under the Coriolis force (Foucault pendulum). In presenting the (linearized) equations of 2-dimensional motion and its solution, begin with the usual pendulum (without the Coriolis force):

$$m\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} = f - m\Omega^2 z.$$

Here $\Omega = \sqrt{g/l}$ is the angular frequency of pendulum oscillations, determined by length l and gravity g. The external force is f. Laplace transform yields the solution

$$Z(p) = \frac{pz_0 + v_0 + \frac{F(p)}{m}}{p^2 + \Omega^2}$$

responding to initial z_0, v_0 and to external $F(p)$. The response to initial conditions is

$$z(t) = z_0 \cos\Omega t + \frac{v_0}{\Omega}\sin\Omega t.$$

In the simplest case with $v_0 = 0$, the pendulum oscillations run (forth and back) permanently in the same direction, see Figure 1.

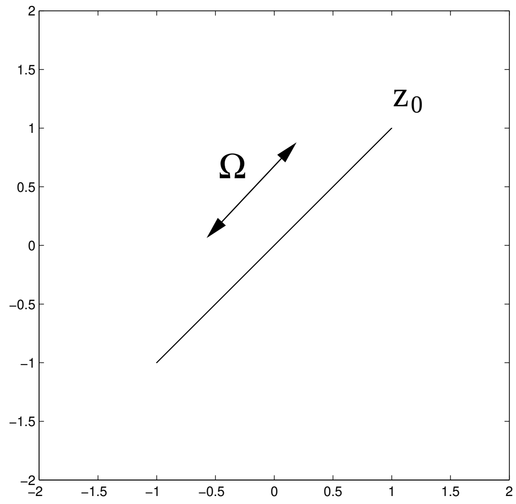

Figure 1: Pendulum oscillations without Coriolis force.

Now introduce the Coriolis force. Denote $\omega = \omega_E \sin\theta$ where ω_E is the rotation of planet Earth and θ is the geographical latitude. The motion equation is

$$m\frac{\mathrm{d}^2 z}{\mathrm{d}t^2} = f - m\Omega^2 z - 2m\mathrm{j}\omega\frac{dz}{dt}$$

with assumption $\Omega >> \omega$. The Laplace transform yields

$$Z(p) = \frac{(p + 2\mathrm{j}\omega)z_0 + v_0 + \frac{F(p)}{m}}{p^2 + 2\mathrm{j}\omega p + \Omega^2}.$$

Assumption $\Omega >> \omega$ makes it possible to replace the denominator by

$$(p + \mathrm{j}\omega)^2 + \Omega^2 = (p + \mathrm{j}\omega + \mathrm{j}\Omega)(p + \mathrm{j}\omega - \mathrm{j}\Omega).$$

With a simplification also in the numerator, the response to initial conditions is

$$\begin{aligned}
Z(p) &= \frac{p+\mathrm{j}\omega}{(p+\mathrm{j}\omega)^2+\Omega^2}z_0 + \frac{1}{(p+\mathrm{j}\omega)^2+\Omega^2}v_0 \\
z(t) &= z_0\mathrm{e}^{-\mathrm{j}\omega t}\cos\Omega t + \frac{v_0}{\Omega}\mathrm{e}^{-\mathrm{j}\omega t}\sin\Omega t
\end{aligned}$$

When $v_0 = 0$ the direction of pendulum oscillations (angular frequency Ω) slowly rotates with angular frequency ω, see Figure 2. This is a well-known result obtained by Foucault.

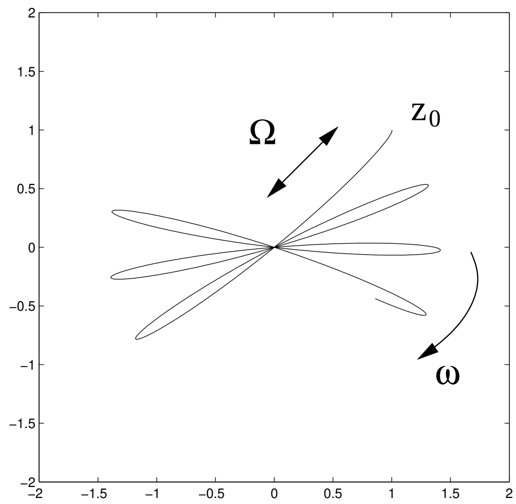

Figure 2: Pendulum oscillations with Coriolis force.

Even in the simpler case of the usual pendulum with equations

$$\frac{\mathrm{d}x}{\mathrm{d}t} = v, \qquad \frac{\mathrm{d}v}{\mathrm{d}t} = -\Omega^2 x$$

and with solution

$$\begin{aligned}
x &= x_0\cos\Omega t + \frac{v_0}{\Omega}\sin\Omega t \\
v &= -\Omega x_0 \sin\Omega t + v_0\cos\Omega t,
\end{aligned}$$

the complex variable can be used advantageously. First, by substitution

$$x = \frac{x'}{\Omega}, \qquad v = v', \qquad t = \frac{t'}{\Omega}$$

the equations get normalized

$$\frac{\mathrm{d}x'}{\mathrm{d}t'} = v', \qquad \frac{\mathrm{d}v'}{\mathrm{d}t'} = -x'.$$

Second, by introducing the complex variable $z' = x' + \mathrm{j}v'$, $z'_0 = x'_0 + \mathrm{j}v'_0$, we obtain equation

$$\frac{\mathrm{d}z'}{\mathrm{d}t'} = -\mathrm{j}z'$$

with solution

$$z' = z'_0 e^{-jt'}.$$

By back substitutions, the original solution is easily recovered.

Note how the device of complex variables made all the derivations simple and transparent. Similar cases can occur in satellite and cosmic vehicles control, and also in electrodynamics: motion of electrons or rigid bodies in magnetic fields.

3.2 Complex Coefficients in Signal Processing

Transferring of an amplitude-modulated signal is a next example of a dynamical system with complex coefficients. The scheme is standard and consists of a modulator, a channel and a demodulator.

The carrier frequency Ω is capable of transferring two input signals $u_1(t), u_2(t)$. The first one is multiplied by $\cos \Omega t$, the second one by $-\sin \Omega t$ and the results are added. The signal $v(t) = u_1(t)\cos \Omega t - u_2(t)\sin \Omega t$ is the input to the channel. The output of the channel has the form $x(t) = y_1(t)\cos \Omega t - y_2(t)\sin \Omega t$. The demodulator recovers $y_1(t), y_2(t)$ by respective multiplication by $2\cos \Omega t, -2\sin \Omega t$ and by low frequency filtering:

$$2[y_1(t)\cos \Omega t - y_2(t)\sin \Omega t]\cos \Omega t = y_1(t)(1+\cos 2\Omega t) - y_2(t)\sin 2\Omega t$$

$$-2[y_1(t)\cos \Omega t - y_2(t)\sin \Omega t]\sin \Omega t = -y_1(t)\sin 2\Omega t + y_2(t)(1-\cos 2\Omega t).$$

The filter absorbs the frequency 2Ω.

Now, given Ω and F, what are the transfer functions from $u_1(t), u_2(t)$ to $y_1(t), y_2(t)$? By Fourier transform analysis, we finally arrive at

$$Y_1(\omega) = H_1(\omega)U_1(\omega) - H_2(\omega)U_2(\omega)$$

$$Y_2(\omega) = H_2(\omega)U_1(\omega) + H_1(\omega)U_2(\omega)$$

where

$$H_1(\omega) = \frac{F(\Omega+\omega) + \overline{F(\Omega-\omega)}}{2}$$

$$H_2(\omega) = \frac{F(\Omega+\omega) - \overline{F(\Omega-\omega)}}{2j}$$

Here the 2x2 transfer function system has a special form. Instead of 4 transfer functions, only 2 are present: the "synphase" $H_1(\omega)$, given by the Ω-symmetric part of $F(\omega)$, and the "quadrature" $H_2(\omega)$, given by the Ω-antisymmetric part. These terms have the following meaning: H_1 describes the transfer where carrier frequencies in the input and in the output have the same phase, whereas H_2 describes the transfer where the phases differ by a right angle.

Due to this speciality, the complex coefficients can nicely come into play here. Two real equations can be equivalently written as one complex equation:

$$Y_1(\omega) + \mathrm{j}Y_2(\omega) = [H_1(\omega) + \mathrm{j}H_2(\omega)][U_1(\omega) + \mathrm{j}U_2(\omega)]$$
$$Y(\omega) = H(\omega)U(\omega)$$

where $H(\omega) = F(\Omega + \omega)$. This is a great simplification.

As an example, consider the delay T:

$$F(\omega) = \mathrm{e}^{-\mathrm{j}\omega T}, \qquad H(\omega) = \mathrm{e}^{-\mathrm{j}(\Omega+\omega)T}$$
$$H_1(\omega) = \cos \Omega T \mathrm{e}^{-\mathrm{j}\omega T}, \qquad H_2(\omega) = -\sin \Omega T \mathrm{e}^{-\mathrm{j}\omega T}$$

One simplification more is possible. Suppose the transfer function $F(\omega)$ can be expressed as $F(\omega) = G(\omega) + \overline{G(-\omega)}$ where the spectrum $G(\omega)$ is concentrated mainly on $0 < \omega < +\infty$. Then $G(\omega)$ can be used instead of $F(\omega)$. The result is $H(\omega) = G(\Omega + \omega)$.

As an illustrating example, consider the RLC circuit whose resonant frequency is just Ω:

$$F(\omega) = \frac{b + \mathrm{j}\omega}{(b + \mathrm{j}\omega)^2 + \Omega^2}, \qquad G(\omega) = \frac{1}{2}\frac{1}{b + \mathrm{j}(\omega - \Omega)}$$
$$H(\omega) = H_1(\omega) = \frac{1}{2}\frac{1}{b + \mathrm{j}\omega}, \qquad H_2(\omega) = 0$$

Here $G(\omega)$ is Ω-symmetric ($F(\omega)$ is approximately Ω-symmetric) and only the synphase component is present, so $H(\omega)$ has real coefficients.

With a different resonant frequency Ω', it holds

$$H(\omega) = \frac{1}{2}\frac{1}{b + \mathrm{j}(\omega + \Delta)}, \qquad \nabla = \Omega - \Omega'$$
$$H_1(\omega) = \frac{1}{2}\frac{b + \mathrm{j}\omega}{(b + \mathrm{j}\omega)^2 + \Delta^2}, \qquad H_2(\omega) = \frac{1}{2}\frac{-\Delta}{(b + \mathrm{j}\omega)^2 + \Delta^2}$$

Here both the synphase and the quadrature component are present, $H(\omega)$ having complex coefficients.

All the examples presented above illustrate the relevance of complex coefficients in the study of real-world systems. Note that the examples were described in continuous-time, but that discrete-time systems with complex coefficients naturally arise after discretization. When approaching signal processing or system control through polynomial techniques [14, 15], complex coefficients in mathematical representations naturally entails solving complex matrix polynomial equations.

More specifically, when solving spectral factorization problems with the iterative linearization scheme described in [10], at each step of the algorithm a two-sided matrix equation

$$A^\star(z)X(z) + X^\star(z)A(z) = B(z)$$

with complex coefficients must be solved. When studying stability of dynamical systems described by higher order differential equations, the above two-sided equation is also sometimes referred to as the polynomial Lyapunov equation, see Section 4 in [21].

4 Scalar Case

First, we study the discrete-time symmetric polynomial equation

$$a^\star(\mathsf{z})x(\mathsf{z}) + x^\star(\mathsf{z})a(\mathsf{z}) = b(\mathsf{z}) \tag{3}$$

when the following assumptions are made.

Assumption 1

- *$a(\mathsf{z})$ is a given Schur complex polynomial,*
- *$b(\mathsf{z}) = b^\star(\mathsf{z})$ is a given two-sided symmetric complex polynomial, and*
- *$x(\mathsf{z})$ is a complex polynomial to be found.*

We also assume that

Assumption 2

- $\mathrm{Re}\, a(0) \neq 0$

Note that Assumption 2 is made without loss of generality since if $\mathrm{Re}\, a(0) = 0$ then necessarily $\mathrm{Im}\, a(0) \neq 0$ as $a(\mathsf{z})$ is assumed to be Schur. Then we can work on the transformed equation

$$\tilde{a}^\star(\mathsf{z})\tilde{x}(\mathsf{z}) + \tilde{x}^\star(\mathsf{z})\tilde{a}(\mathsf{z}) = b(\mathsf{z})$$

where $\tilde{a}(z) = \mathrm{j}a(\mathsf{z})$, $\tilde{x}(z) = \mathrm{j}x(\mathsf{z})$ and $\mathrm{Re}\,\tilde{a}(0) \neq 0$.

Typically, polynomial equations have general solutions and particular solutions. Usually the general solution is parametrized by some parameter, and a particular solution is obtained by fixing this parameter. This particular solution may or may not be unique. The situation is somewhat similar to differential equations, where under proper requirements (e.g. boundary conditions), a particular solution can be selected.

In Section 4.1, we will show that, provided one additional requirement, equation (3) admits a unique particular solution $x(\mathsf{z})$. Our proof is constructive and consists in a first technique for solving equation (3), referred to as the complex reduction algorithm. In Section 4.2, we propose a second numerical method for solving equation (3), this time relying upon Sylvester matrices.

4.1 Complex Reduction Algorithm

Theorem 1 *Under Assumptions 1 and 2, a polynomial solution $x(\mathtt{z})$ to equation (3) such that $\delta x \le \max(\delta a, \delta b)$ always exists. Moreover, under the additional requirement that* $\operatorname{Im} x(0) = 0$*, the solution is unique.*

Proof of Theorem 1 (Algorithm ScalRed) The principle lying behind the proof of existence of a polynomial solution to equation (3) under the assumptions that $a(\mathtt{z})$ is stable can be found in [9, 10, 11, 12]. This proof is constructive and consists in the complex polynomial reduction algorithm, an extension of the Euclidean division algorithm for polynomials with complex coefficients. Let

$$\begin{aligned} a(\mathtt{z}) &= a_0 + a_1\mathtt{z} + \cdots + a_{\delta a}\mathtt{z}^{\delta a} \\ b(\mathtt{z}) &= \overline{b}_{\delta b}\mathtt{z}^{-\delta b} + \cdots + \overline{b}_1\mathtt{z}^{-1} + b_0 + b_1\mathtt{z} + \cdots + b_{\delta b}\mathtt{z}^{\delta b} = b^\star(\mathtt{z}) \end{aligned}$$

with b_0 real.

- If $\delta b > \delta a$ then the substitution

  $$x(\mathtt{z}) = \hat{x}(\mathtt{z}) + \frac{b_{\delta b}}{\overline{a}_0}\mathtt{z}^{\delta b}$$

  into equation (3) leads to the equation

  $$a^\star(\mathtt{z})\hat{x}(\mathtt{z}) + \hat{x}^\star(\mathtt{z})a(\mathtt{z}) = \hat{b}(\mathtt{z})$$

  where

  $$\hat{b}(\mathtt{z}) = b(\mathtt{z}) - \frac{b_{\delta b}}{\overline{a}_0}\mathtt{z}^{\delta b}a^\star(\mathtt{z}) - \frac{\overline{b}_{\delta b}}{a_0}\mathtt{z}^{-\delta b}a(\mathtt{z}).$$

  Since $a(\mathtt{z})$ is Schur, the above substitution can always be performed. One can check that if $x(\mathtt{z})$ is polynomial, then $\hat{x}(\mathtt{z})$ is also polynomial. Moreover $\delta\hat{b} < \delta b$. Thus equation (3) is replaced by another equation of the same kind but with lower degree in $b(\mathtt{z})$.

- If $\delta b \le \delta a$ then the substitution

  $$x(\mathtt{z}) = \hat{x}(\mathtt{z}) - \frac{a_{\delta a}}{\overline{a}_0}\mathtt{z}^{\delta a}\hat{x}^\star(\mathtt{z})$$

  into equation (3) leads to the equation

  $$\hat{a}^\star(\mathtt{z})\hat{x}(\mathtt{z}) + \hat{x}^\star(\mathtt{z})\hat{a}(\mathtt{z}) = b(\mathtt{z})$$

  where

  $$\hat{a}(\mathtt{z}) = a(\mathtt{z}) - \frac{a_{\delta a}}{\overline{a}_0}\mathtt{z}^{\delta a}a^\star(\mathtt{z}).$$

  One can check that if $x(\mathtt{z})$ is polynomial, then $\hat{x}(\mathtt{z})$ is also polynomial. Moreover $\delta\hat{a} < \delta a$. Thus equation (3) is replaced by another equation of the same kind but with lower degree in $a(\mathtt{z})$.

By repeating the two steps we eventually come to the case $\delta a = 0$, $\delta b = 0$ that can be solved directly for a constant solution. Upon performing all the substitutions backwards, we recover the original polynomial solution $x(\mathtt{z})$ to equation (3).

By the complex reduction algorithm, we have shown that at least one particular polynomial solution to equation (3) exists. Now we prove its uniqueness provided that $\mathrm{Im}\, x(0) = 0$. Given a particular solution $x_p(\mathtt{z})$ to equation (3), one can easily check that any other solution to equation (3) reads

$$x(\mathtt{z}) = x_p(\mathtt{z}) + x_g(\mathtt{z}) \tag{4}$$

where $x_g(\mathtt{z})$ is the general polynomial solution of the homogeneous equation

$$a^\star(\mathtt{z})x(\mathtt{z}) + x^\star(\mathtt{z})a(\mathtt{z}) = 0. \tag{5}$$

In order to find $x_g(\mathtt{z})$, consider the equation

$$a^\star(\mathtt{z})x(\mathtt{z}) + y(\mathtt{z})a(\mathtt{z}) = 0 \tag{6}$$

in the ring of two-sided polynomials. As $a(\mathtt{z})$ and $a^\star(\mathtt{z})$ are coprime, the general solution to equation (6) reads

$$\begin{array}{rcl} x(\mathtt{z}) & = & a(\mathtt{z})q(\mathtt{z}) \\ y(\mathtt{z}) & = & -a^\star(\mathtt{z})q(\mathtt{z}) \end{array}$$

where $q(\mathtt{z})$ is an arbitrary two-sided polynomial. Recalling equation (5), this polynomial must be antisymmetric, i.e. $q^\star(\mathtt{z}) = -q(\mathtt{z})$. Now, $a(\mathtt{z})$ being a Schur polynomial, for $x_g(\mathtt{z}) = a(\mathtt{z})q(\mathtt{z})$ to be a polynomial, $q(\mathtt{z})$ must be a polynomial. The only possible choice is an imaginary constant. Since we enforce $\mathrm{Im}\, x(0) = 0$, we have

$$q(\mathtt{z}) = -\mathrm{j}\frac{\mathrm{Im}\, x_p(0)}{\mathrm{Re}\, a(0)} \tag{7}$$

in equation (4). Because of Assumption 2, this is always possible.

Finally, we would like to show that $\delta x \leq \max(\delta a, \delta b)$. First suppose that $\delta b \leq \delta a$. Since $\delta(x^\star a) \leq \delta a$, we also have $\delta(a^\star x) \leq \delta a$. Because $a(0) \neq 0$, $\delta(x^\star a) = \delta(a^\star x) = \delta x \leq \delta a$. Conversely, suppose that $\delta b > \delta a$. Since $\delta(x^\star a) \leq \delta b$, we also have $\delta(a^\star x) \leq \delta b$ and $\delta x \leq \delta b$.

$\square$

4.2 Sylvester Matrix Algorithm

An alternative numerical method is now proposed for solving equation (3) without resorting to polynomial operations. It is based on Sylvester matrices and relies upon resolution of a linear system of equations.

The Sylvester matrix formulation of equation (3) merely consists in equating the coefficients of equal powers of the indeterminate $\mathtt{z}$. Let

$$x(\mathtt{z}) = x_0 + x_1\mathtt{z} + \cdots + x_{\delta x}\mathtt{z}^{\delta x}$$

and suppose for notational ease that $d = \delta a = \delta b = \delta x$. If it is not the case then some leading coefficients of $a(\mathsf{z})$, $b(\mathsf{z})$ or $x(\mathsf{z})$ may be zero. By inspection, equation (3) is equivalent to the linear system of equations

$$\underbrace{\begin{bmatrix} \bar{a}_0 & \bar{a}_1 & \cdots & \bar{a}_d \\ & \bar{a}_0 & & \vdots \\ & & \ddots & \bar{a}_1 \\ \mathbf{0} & & & \bar{a}_0 \end{bmatrix}}_{A_1} \underbrace{\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_d \end{bmatrix}}_{X} + \underbrace{\begin{bmatrix} a_0 & a_1 & \cdots & a_d \\ a_1 & & a_d & \\ \vdots & \cdot^{\cdot^{\cdot}} & & \\ a_d & & & \mathbf{0} \end{bmatrix}}_{A_2} \underbrace{\begin{bmatrix} \bar{x}_0 \\ \bar{x}_1 \\ \vdots \\ \bar{x}_d \end{bmatrix}}_{\bar{X}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_d \end{bmatrix}}_{B}$$

where A_1 and A_2 are complex Toeplitz matrices. Using Fact 1, the above system can be written over the field of real numbers as

$$\underbrace{\left(\begin{bmatrix} \mathrm{Re}A_1 & \mathrm{Im}A_1 \\ -\mathrm{Im}A_1 & \mathrm{Re}A_1 \end{bmatrix} + \begin{bmatrix} \mathrm{Re}A_2 & \mathrm{Im}A_2 \\ \mathrm{Im}A_2 & -\mathrm{Re}A_2 \end{bmatrix} \right)}_{\tilde{\mathbf{A}}} \underbrace{\begin{bmatrix} \mathrm{Re}X \\ \mathrm{Im}X \end{bmatrix}}_{\tilde{\mathbf{X}}} = \underbrace{\begin{bmatrix} \mathrm{Re}B \\ \mathrm{Im}B \end{bmatrix}}_{\mathbf{B}} \tag{8}$$

Under the assumptions of Theorem 1, the imaginary part of $x(0)$ is zero. Thus the above equation can further be simplified to

$$\mathbf{AX} = \mathbf{B} \tag{9}$$

where square matrix $\mathbf{A}$ is obtained upon removing the column of matrix $\tilde{\mathbf{A}}$ corresponding to the imaginary part of $x(0)$ and the row of matrix $\tilde{\mathbf{A}}$ corresponding to the zero imaginary part of $b(0)$, and vector $\mathbf{X}$ is obtained upon removing the corresponding row of vector $\tilde{\mathbf{X}}$. The dimension of matrix $\mathbf{A}$ is $2d + 1$.

Theorem 2 (Algorithm ScalSyl) *Under the requirements of Theorem 1, every solution $x(\mathsf{z})$ to symmetric polynomial equation (3) corresponds to a solution of linear system of equations (8). In particular, the unique solution such that $\mathrm{Im}\, x(0) = 0$ can be found by solving the non-singular linear system of equations (9).*

4.3 Illustration

Algorithms ScalRed and ScalSyl are now illustrated on a simple example. Let

$$\begin{aligned} a(\mathsf{z}) &= 4 + (1 - \mathrm{j})\mathsf{z} \\ b(\mathsf{z}) &= (9 - 11\mathrm{j})\mathsf{z}^{-1} + 6 + (9 + 11\mathrm{j})\mathsf{z} \end{aligned}$$

and suppose that we are to solve equation (3) for a solution $x(\mathsf{z})$ of degree $d = 1$.

4.3.1 Polynomial Reduction Algorithm

Algorithm ScalRed consists of the following steps.

- $\delta b \leq \delta a$, so let
$$\begin{aligned} x(\mathsf{z}) &= \hat{x}(\mathsf{z}) - \tfrac{1-\mathrm{j}}{4}\mathsf{z}\hat{x}^\star(\mathsf{z}) \\ \hat{a}(\mathsf{z}) &= a(\mathsf{z}) - \tfrac{1-\mathrm{j}}{4}\mathsf{z}(4+(1+\mathrm{j})\mathsf{z}^{-1}) = \tfrac{7}{2} \\ \hat{b}(\mathsf{z}) &= b(\mathsf{z}). \end{aligned}$$
Polynomial $\hat{x}(\mathsf{z})$ is a solution to the symmetric equation $\hat{a}^\star(\mathsf{z})\hat{x}(\mathsf{z})+\hat{x}^\star(\mathsf{z})\hat{a}(\mathsf{z}) = \hat{b}(\mathsf{z})$ where now $\delta\hat{a} = 0$ and $\delta\hat{b} = 1$.

- $\delta\hat{b} > \delta\hat{a}$, so let
$$\begin{aligned} \hat{x}(\mathsf{z}) &= \tilde{x}(\mathsf{z}) + \tfrac{2(9+11\mathrm{j})}{7}\mathsf{z} \\ \tilde{a}(\mathsf{z}) &= \hat{a}(\mathsf{z}) \\ \tilde{b}(\mathsf{z}) &= \hat{b}(\mathsf{z}) - \tfrac{2(9+11\mathrm{j})}{7}\mathsf{z}\hat{a}^\star(\mathsf{z}) - \tfrac{2(9-11\mathrm{j})}{7}\mathsf{z}\hat{a}(\mathsf{z}) = 6. \end{aligned}$$
Polynomial $\tilde{x}(\mathsf{z})$ is a solution to the symmetric equation $\tilde{a}^\star(\mathsf{z})\tilde{x}(\mathsf{z})+\tilde{x}^\star(\mathsf{z})\tilde{a}(\mathsf{z}) = \tilde{b}(\mathsf{z})$ where now $\delta\tilde{a} = 0$ and $\delta\tilde{b} = 0$.

The latter equation only involves real numbers and can be solved directly. We have
$$\tilde{x}(\mathsf{z}) = \frac{6}{7}.$$
Upon performing the substitutions backwards, we get
$$\hat{x}(\mathsf{z}) = \frac{6}{7} + \frac{18+22\mathrm{j}}{7}\mathsf{z}$$
and eventually come up with a particular solution to equation (3), denoted
$$x_p(\mathsf{z}) = 1 + \frac{10}{7}\mathrm{j} + \frac{33+47\mathrm{j}}{14}\mathsf{z}.$$

According to equation (7), the choice $q(\mathsf{z}) = -\mathrm{j}5/14$ in (4) yields the unique solution to equation (3) such that $\mathrm{Im}\, x(0) = 0$, namely
$$x(\mathsf{z}) = 1 + (2+3\mathrm{j})\mathsf{z}.$$

4.3.2 Sylvester Matrix Algorithm

Following the development of Algorithm ScalSyl in Section 4.2, we obtain the linear system of complex equations
$$\begin{bmatrix} 4 & 1+\mathrm{j} \\ 0 & 4 \end{bmatrix}\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} + \begin{bmatrix} 4 & 1-\mathrm{j} \\ 1-\mathrm{j} & 0 \end{bmatrix}\begin{bmatrix} \overline{x}_0 \\ \overline{x}_1 \end{bmatrix} = \begin{bmatrix} 6 \\ 9+11\mathrm{j} \end{bmatrix}.$$
Using Fact 1, we obtain
$$\begin{bmatrix} 8 & 2 & 0 & -2 \\ 1 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 4 \end{bmatrix}\begin{bmatrix} \mathrm{Re}\, x_0 \\ \mathrm{Re}\, x_1 \\ \mathrm{Im}\, x_0 \\ \mathrm{Im}\, x_1 \end{bmatrix} = \begin{bmatrix} 6 \\ 9 \\ 0 \\ 11 \end{bmatrix}.$$

Upon suppression of the third column corresponding to $\mathrm{Im}\, x_0$ and the third row corresponding to $\mathrm{Im}\, b_0$, we get

$$\begin{bmatrix} 8 & 2 & -2 \\ 1 & 4 & 0 \\ -1 & 0 & 4 \end{bmatrix} \begin{bmatrix} \mathrm{Re}\, x_0 \\ \mathrm{Re}\, x_1 \\ \mathrm{Im}\, x_1 \end{bmatrix} = \begin{bmatrix} 6 \\ 9 \\ 11 \end{bmatrix}.$$

This linear system of equations can readily be solved and yields the unique solution to equation (3) with real absolute coefficient, namely

$$x(\mathsf{z}) = x_0 + x_1\mathsf{z} = 1 + (2 + 3\mathrm{j})\mathsf{z}.$$

5 Matrix Case

In this section, we study the discrete-time symmetric polynomial equation

$$A^{\star}(\mathsf{z})X(\mathsf{z}) + X^{\star}(\mathsf{z})A(\mathsf{z}) = B(\mathsf{z}) \tag{10}$$

when the following assumptions are made.

Assumption 3

- $A(\mathsf{z})$ *is a given square complex Schur polynomial matrix of dimension* n*,*
- $B(\mathsf{z}) = B^{\star}(\mathsf{z})$ *is a given two-sided symmetric complex polynomial matrix, and*
- $X(\mathsf{z})$ *is a complex polynomial matrix to be found.*

We shall also need the following additional assumption.

Assumption 4

- $A(0)$ *has non-zero leading minors,*

Assumption 4 is made without loss of generality since non-singular matrix $A(0)$ can always be transformed to a matrix with non-zero leading minors via suitable permutations.

Note that matrix equation (10) was already studied in [10, 11, 6] over the field of real polynomials.

Symmetric matrix equation (10) is far more complicated than its scalar counterpart (3), but our study follows along the same lines as in Section 4. First, we establish a condition of existence of a unique solution to equation (10), obtained through a matrix version of the complex reduction algorithm. Second, we propose a Sylvester matrix formulation from which follows an alternative method for solving equation (10).

5.1 Complex Reduction Algorithm

The development of the complex reduction algorithm for solving matrix polynomial equation (10) first requires studying another type of scalar polynomial equation. This is the topic of Section 5.1.1. In Section 5.1.2, a special instance of equation (10) is studied where matrix $A(\mathsf{z})$ is triangular. In Section 5.1.3, a technical lemma on the LU decomposition of a polynomial matrix is proposed. It will be the last component required in Section 5.1.4 to state our main result of existence of a solution to equation (10).

5.1.1 Non-Symmetric Polynomial Equation

First, consider the scalar non-symmetric equation

$$a^\star(\mathsf{z})x(\mathsf{z}) + y^\star(\mathsf{z})b(\mathsf{z}) = c(\mathsf{z}) + d^\star(\mathsf{z}) \tag{11}$$

when the following assumptions are made.

Assumption 5

- *$a(\mathsf{z})$, $b(\mathsf{z})$ are given Schur complex polynomials,*
- *$c(\mathsf{z})$, $d(\mathsf{z})$ are given complex polynomials, and*
- *$x(\mathsf{z})$, $y(\mathsf{z})$ are complex polynomials to be found.*

We also require that

Assumption 6

- $\mathrm{Re}\, a(0) \neq 0$ *and* $\mathrm{Re}\, b(0) \neq 0$.

Note that Assumption 6 is made without loss of generality, see Assumption 2 for more details.

Some aspects of non-symmetric equation (11) were already studied in [9, 10, 11].

Theorem 3 *Under Assumptions 5 and 6, a polynomial solution $x(\mathsf{z})$, $y(\mathsf{z})$ to equation (11) such that $\delta x \leq \max(\delta a, \delta c)$ and $\delta y \leq \max(\delta b, \delta d)$ always exists. Moreover, under the additional requirement that either $\mathrm{Im}\, x(0) = 0$ or $\mathrm{Im}\, y(0) = 0$, the solution is unique.*

Proof (Algorithm SCALRED2**)** The proof of existence of at least one polynomial solution to equation (11) under the given assumptions closely follows the first part of the proof of Theorem 1. The approach is constructive and consists in a complex polynomial reduction algorithm for solving equation (11). Let

$$\begin{aligned} a(\mathsf{z}) &= a_0 + a_1\mathsf{z} + \cdots + a_{\delta a}\mathsf{z}^{\delta a} \\ b(\mathsf{z}) &= b_0 + b_1\mathsf{z} + \cdots + b_{\delta b}\mathsf{z}^{\delta b} \\ c(\mathsf{z}) &= c_0 + c_1\mathsf{z} + \cdots + c_{\delta c}\mathsf{z}^{\delta c} \\ d(\mathsf{z}) &= d_0 + d_1\mathsf{z} + \cdots + d_{\delta d}\mathsf{z}^{\delta d} \end{aligned}$$

- If $\delta c > \delta b$ then the substitution

  $$x(\mathsf{z}) = \hat{x}(\mathsf{z}) + \frac{c_{\delta c}}{\bar{a}_0}\mathsf{z}^{\delta c}$$

  into equation (11) leads to the equation

  $$a^\star(\mathsf{z})\hat{x}(\mathsf{z}) + y^\star(\mathsf{z})b(\mathsf{z}) = \hat{c}(\mathsf{z}) + d^\star(\mathsf{z})$$

  where

  $$\hat{c}(\mathsf{z}) = c(\mathsf{z}) - \frac{c_{\delta c}}{\bar{a}_0}\mathsf{z}^{\delta c}a^\star(\mathsf{z}).$$

  Since $a(\mathsf{z})$ is Schur, $a_0 \neq 0$ and the above substitution can always be performed. One can check that if $x(\mathsf{z})$ is polynomial then $\hat{x}(\mathsf{z})$ is also polynomial. Moreover $\delta\hat{c} < \delta c$. Thus equation (11) is replaced by another equation of the same kind but with lower degree in $c(\mathsf{z})$.

- If $\delta d > \delta a$ then the substitution

  $$y(\mathsf{z}) = \hat{y}(\mathsf{z}) + \frac{d_{\delta d}}{\bar{b}_0}\mathsf{z}^{\delta d}$$

  into equation (11) leads to the equation

  $$a^\star(\mathsf{z})x(\mathsf{z}) + \hat{y}^\star(\mathsf{z})b(\mathsf{z}) = c(\mathsf{z}) + \hat{d}^\star(\mathsf{z})$$

  where

  $$\hat{d}^\star(\mathsf{z}) = d^\star(\mathsf{z}) - \frac{d_{\delta d}}{\bar{b}_0}\mathsf{z}^{\delta d}b^\star(\mathsf{z}).$$

  Since $b(\mathsf{z})$ is Schur, $b_0 \neq 0$ and the above substitution can always be performed. One can check that if $y(\mathsf{z})$ is polynomial then $\hat{y}(\mathsf{z})$ is also polynomial. Moreover $\delta\hat{d} < \delta d$. Thus equation (11) is replaced by another equation of the same kind but with lower degree in $d(\mathsf{z})$.

- If $\delta d \leq \delta a \leq \delta b$ then the substitution

  $$x(\mathsf{z}) = \hat{x}(\mathsf{z}) - \frac{b_{\delta b}}{\bar{a}_0}\mathsf{z}^{\delta b}y^\star(\mathsf{z})$$

  into equation (11) leads to the equation

  $$a^\star(\mathsf{z})\hat{x}(\mathsf{z}) + y^\star(\mathsf{z})\hat{b}(\mathsf{z}) = c(\mathsf{z}) + d^\star(\mathsf{z})$$

where

$$\hat{b}(\mathsf{z}) = b(\mathsf{z}) - \frac{b_{\delta b}}{\bar{a}_0}\mathsf{z}^{\delta b} a^\star(\mathsf{z}).$$

One can check that if $x(\mathsf{z})$ is polynomial then $\hat{x}(\mathsf{z})$ is also polynomial. Moreover $\delta\hat{b} < \delta b$. Thus equation (11) is replaced by another equation of the same kind but with lower degree in $b(\mathsf{z})$.

- If $\delta c \leq \delta b \leq \delta a$ then the substitution

$$y(\mathsf{z}) = \hat{y}(\mathsf{z}) - \frac{a_{\delta a}}{\bar{b}_0}\mathsf{z}^{\delta a} x^\star(\mathsf{z})$$

into equation (11) leads to the equation

$$\hat{a}^\star(\mathsf{z})x(\mathsf{z}) + \hat{y}^\star(\mathsf{z})b(\mathsf{z}) = c(\mathsf{z}) + d^\star(\mathsf{z})$$

where

$$\hat{a}(\mathsf{z}) = a(\mathsf{z}) - \frac{a_{\delta a}}{\bar{b}_0}\mathsf{z}^{\delta a} b^\star(\mathsf{z}).$$

One can check that if $y(\mathsf{z})$ is polynomial then $\hat{y}(\mathsf{z})$ is also polynomial. Moreover $\delta\hat{a} < \delta a$. Thus equation (11) is replaced by another equation of the same kind but with lower degree in $a(\mathsf{z})$.

By repeating the above four steps we eventually come to the case $\delta a = 0$, $\delta b = 0$, $\delta c = 0$, $\delta d = 0$ that can be solved directly for a constant solution. Upon performing all the substitutions backwards, we recover the original polynomial solution $x(\mathsf{z})$, $y(\mathsf{z})$ to equation (11).

The existence of a unique polynomial solution such that $\mathrm{Im}\,x(0) = 0$ or $\mathrm{Im}\,y(0) = 0$ follows as in the proof of Theorem 1 from the parametrization of all the two-sided solutions to (11):

$$\begin{aligned} x(\mathsf{z}) &= x_p(\mathsf{z}) + b(\mathsf{z})q(\mathsf{z}) \\ y(\mathsf{z}) &= y_p(\mathsf{z}) - a(\mathsf{z})q^\star(\mathsf{z}) \end{aligned}$$

where $x_p(\mathsf{z})$, $y_p(\mathsf{z})$ is a particular solution to equation (11) and $q(\mathsf{z})$ is an arbitrary two-sided symmetric complex polynomial. As $a(\mathsf{z})$ and $b(\mathsf{z})$ are Schur polynomials, the only possibility for $x(\mathsf{z})$, $y(\mathsf{z})$ to be polynomial is that $q(\mathsf{z})$ is an imaginary constant. For example, if the imaginary part of $x(0)$ is to be zeroed, then select

$$q(\mathsf{z}) = -\mathrm{j}\frac{\mathrm{Im}\; x_p(0)}{\mathrm{Re}\; b(0)}.$$

This is always possible, recall Assumption 6.

The proof of the degree property on $x(\mathsf{z})$ and $y(\mathsf{z})$ mimics the proof of Theorem 1 and is not repeated here. $\square$

5.1.2 Matrix $A(\mathsf{z})$ is Triangular

Lemma 1 *Under Assumptions 3 and 4, suppose that $A(\mathsf{z})$ is upper-triangular. Then there exists a unique solution $X(\mathsf{z})$ to (10) such that $X(0)$ is upper-triangular with real diagonal entries.*

Proof Write equation (10) entrywise

$$\begin{bmatrix} a_{11}^\star & & \mathbf{0} \\ \vdots & \ddots & \\ a_{1n}^\star & \cdots & a_{nn}^\star \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix} + \begin{bmatrix} x_{11}^\star & \cdots & x_{n1}^\star \\ \vdots & \ddots & \vdots \\ x_{1n}^\star & \cdots & x_{nn}^\star \end{bmatrix} \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ & \ddots & \vdots \\ \mathbf{0} & & a_{nn} \end{bmatrix} = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{1n}^\star & \cdots & b_{nn} \end{bmatrix}.$$

- Entry $(1,1)$ reads
$$a_{11}^\star(\mathsf{z})x_{11}(\mathsf{z}) + x_{11}^\star(\mathsf{z})a_{11}(\mathsf{z}) = b_{11}(\mathsf{z}). \tag{12}$$
This is a symmetric polynomial equation. Since $a_{11}(\mathsf{z})$ is Schur, from Theorem 1 there exists a unique polynomial solution to equation (12) such that $\operatorname{Im} x_{11}(0) = 0$.

- Entry $(1,2)$ reads
$$a_{11}^\star(\mathsf{z})x_{12}(\mathsf{z}) + x_{21}^\star(\mathsf{z})a_{22}(\mathsf{z}) = b_{12}(\mathsf{z}) - x_{11}^\star(\mathsf{z})a_{12}(\mathsf{z}). \tag{13}$$
This is a non-symmetric polynomial equation. Since $a_{11}(\mathsf{z})$ and $a_{22}(\mathsf{z})$ are Schur and $x_{11}(\mathsf{z})$ is known from the previous step, from Theorem 3 there exists a unique polynomial solution $x_{12}(\mathsf{z})$, $x_{21}(\mathsf{z})$ to (13) such that $x_{21}(0) = 0$.

- Entry $(2,2)$ reads
$$a_{22}^\star(\mathsf{z})x_{22}(\mathsf{z}) + x_{22}^\star(\mathsf{z})a_{22}(\mathsf{z}) = b_{22}(\mathsf{z}) - a_{12}^\star(\mathsf{z})x_{12}(\mathsf{z}) - x_{12}^\star(\mathsf{z})a_{12}(\mathsf{z}). \tag{14}$$
This is a symmetric polynomial equation. Since $a_{22}(\mathsf{z})$ is Schur and $x_{12}(\mathsf{z})$ is known from the previous step, from Theorem 1 there exists a unique polynomial solution $x_{22}(\mathsf{z})$ to (14) such that $\operatorname{Im} x_{22}(0) = 0$.

Proceeding as above for each upper triangular entry in matrix polynomial equation (10), one successively solves two types of scalar polynomial equations:

- Symmetric equations
$$a_{pp}^\star(\mathsf{z})x_{pp}(\mathsf{z}) + x_{pp}^\star(\mathsf{z})a_{pp}(\mathsf{z}) = b_{pp}(\mathsf{z}) - \sum_{k=1}^{p-1} a_{kp}^\star(\mathsf{z})x_{kp}(\mathsf{z}) - \sum_{k=1}^{p-1} x_{kp}^\star(\mathsf{z})a_{kp}(\mathsf{z})$$
for $p = 1, \ldots, n$. Each equation has a unique polynomial solution $x_{pp}(\mathsf{z})$ such that $\operatorname{Im} x_{pp}(0) = 0$.

- Non-symmetric equations

$$a^{\star}_{pp}(\mathsf{z})x_{pq}(\mathsf{z}) + x^{\star}_{qp}(\mathsf{z})a_{qq}(\mathsf{z}) = b_{pq}(\mathsf{z}) - \sum_{k=1}^{p-1} a^{\star}_{kp}(\mathsf{z})x_{kq}(\mathsf{z}) - \sum_{k=1}^{q-1} x^{\star}_{kp}(\mathsf{z})a_{kq}(\mathsf{z})$$

  for $p = 1, \ldots, n$ and $q = p+1, \ldots, n$. Each equation has a unique polynomial solution $x_{pq}(\mathsf{z})$, $x_{qp}(\mathsf{z})$ such that $x_{qp}(0) = 0$.

As a result, the matrix $X(\mathsf{z})$ whose entries are built according to the above procedure is unique and $X(0)$ is upper-triangular with real diagonal entries. □

5.1.3 LU Decomposition for Polynomial Matrices

Lemma 2 *Given a square polynomial matrix $A(\mathsf{z})$ such that $A(0)$ has non-zero leading minors, there always exists a decomposition*

$$A(\mathsf{z}) = L(\mathsf{z})U(\mathsf{z})$$

where $L(\mathsf{z})$ is a unimodular matrix such that $L(0)$ is lower-triangular with unit diagonal entries and $U(\mathsf{z})$ is an upper-triangular polynomial matrix whose diagonal entries have non-zero real parts.

Proof (Algorithm LURED**)** Existence of a unimodular matrix $V(\mathsf{z})$ such that non-singular matrix $A(\mathsf{z})$ is transformed into upper-triangular form $H(\mathsf{z}) = V^{-1}(\mathsf{z})A(\mathsf{z})$ directly follows from the existence of the row Hermite form of $A(\mathsf{z})$, see for instance [13, Theorem 6.3-2]. Existence of matrices $L(\mathsf{z})$ and $U(\mathsf{z})$ satisfying the required properties stems for the complex LU-decomposition of absolute coefficient matrix $A(0) = L_0U_0$ obtained through Gaussian elimination, see [3, Section 4.2]. In matrix notation, we have

$$\begin{aligned} A(\mathsf{z}) &= V(\mathsf{z})H(\mathsf{z}) \\ &= V(\mathsf{z})V^{-1}(0)A(0)H^{-1}(0)H(\mathsf{z}) \\ &= \underbrace{V(\mathsf{z})V^{-1}(0)L_0}_{L(\mathsf{z})}\underbrace{U_0H^{-1}(0)H(\mathsf{z})}_{U(\mathsf{z})}. \end{aligned}$$

□

5.1.4 Main Result

With the help of the lemmas developed in the previous sections, we can now state our main result.

Theorem 4 *Under Assumptions 3 and 4, a polynomial solution $X(\mathsf{z})$ to equation (10) such that $\delta X \leq \max(\delta A, \delta B)$ always exists. Moreover, under the additional requirement that $X(0)$ is upper-triangular with real diagonal entries, the solution is unique.*

Proof (Algorithm MatRed) The proof of Theorem 4 is constructive and provides us with a first algorithm for solving equation (10). First, factorize $A(\mathsf{z}) = L(\mathsf{z})U(\mathsf{z})$ as in Lemma 2, using the LU-decomposition of $A(0)$. Second, build the equation

$$\underbrace{U^\star(\mathsf{z})}_{\hat{A}^\star(\mathsf{z})}\underbrace{L^\star(\mathsf{z})X(\mathsf{z})}_{\hat{X}(\mathsf{z})}+\underbrace{X^\star(\mathsf{z})L(\mathsf{z})}_{\hat{X}^\star(\mathsf{z})}\underbrace{U(\mathsf{z})}_{\hat{A}(\mathsf{z})} = B(\mathsf{z})$$

where $\hat{A}(\mathsf{z}) = U(\mathsf{z})$ is upper-triangular with diagonal entries with non-zero real parts. From Lemma 1, there exists a unique solution $\hat{X}(\mathsf{z})$ such that $\hat{X}(0)$ is upper-triangular with real diagonal entries. As in the proof of Lemma 1, this solution can be computed from successive resolutions of scalar symmetric equations (3) and non-symmetric equations (11), through complex reduction Algorithms ScalRed and ScalRed2. Since $L^\star(0)$ is upper-triangular with unit diagonal entries and $\hat{X}(0) = L^\star(0)X(0)$, then $X(0)$ is upper-triangular with real diagonal entries. $\square$

5.2 Sylvester Matrix Algorithm

As in the scalar case, the Sylvester matrix approach for solving equation (10) relies upon the resolution of a linear system of equations over the field of reals. Suppose for notational ease that $d = \delta A = \delta B = \delta X$. If it is not the case then some leading coefficients of $A(\mathsf{z})$, $B(\mathsf{z})$ or $X(\mathsf{z})$ may be zero.

Let $\otimes$ stand for the Kronecker product, $\mathrm{vec}A$ for the column vector obtained by stacking the columns of matrix A and P for the orthogonal permutation matrix such that for any arbitrary matrices of compatible dimensions it holds $\mathrm{vec}(AXB) = (B^T \otimes A)\mathrm{vec}X$, $\mathrm{vec}(X^T) = P\mathrm{vec}X$ and $(A \otimes B)P = P(B \otimes A)$.

Upon application of the Kronecker product [6], equation (10) reads

$$[\mathbf{I}_n \otimes A^\star(\mathsf{z})]\mathrm{vec}X(\mathsf{z}) + [A^T(\mathsf{z}) \otimes \mathbf{I}_n]\mathrm{vec}X^\star(\mathsf{z}) = \mathrm{vec}B(\mathsf{z})$$

or, equivalently,

$$[\mathbf{I}_n \otimes A^\star(\mathsf{z})]\mathrm{vec}X(\mathsf{z}) + P[\mathbf{I}_n \otimes A^T(\mathsf{z})]\mathrm{vec}X^{*T}(\mathsf{z}) = \mathrm{vec}B(\mathsf{z}). \tag{15}$$

Now define

$$Q(\mathsf{z}) = \mathbf{I}_n \otimes A^T(\mathsf{z}) = Q_0 + Q_1\mathsf{z} + \cdots Q_d(\mathsf{z}),$$

so that

$$\mathbf{I}_n \otimes A^\star(\mathsf{z}) = \overline{Q}_0 + \overline{Q}_1\mathsf{z}^{-1} + \cdots + \overline{Q}_d\mathsf{z}^{-d}.$$

Moreover, let

$$V(\mathsf{z}) = V_0 + V_1\mathsf{z} + \cdots + V_d\mathsf{z}^d = \mathrm{vec}X(\mathsf{z})$$

and

$$W(\mathsf{z}) = P\overline{W}_d\mathsf{z}^{-d} + \cdots + P\overline{W}_1\mathsf{z}^{-1} + W_0 + W_1\mathsf{z} + \cdots + W_d\mathsf{z}^d = \mathrm{vec}B(\mathsf{z}).$$

With these notations, equating the powers of $\mathbf{z}$ in (15) yields

$$\begin{bmatrix} \overline{Q}_d & & & \mathbf{0} \\ \vdots & \overline{Q}_d & & \\ \overline{Q}_1 & \vdots & \ddots & \\ \overline{Q}_0 & \overline{Q}_1 & & \overline{Q}_d \\ & \overline{Q}_0 & & \vdots \\ & & \ddots & \overline{Q}_1 \\ \mathbf{0} & & & \overline{Q}_0 \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_d \end{bmatrix} + \begin{bmatrix} \mathbf{0} & & & PQ_0 \\ & & \iddots & PQ_1 \\ & PQ_0 & & \vdots \\ PQ_0 & PQ_1 & & PQ_d \\ PQ_1 & \vdots & \iddots & \\ \vdots & PQ_d & & \\ PQ_d & & & \mathbf{0} \end{bmatrix} \begin{bmatrix} \overline{V}_0 \\ \overline{V}_1 \\ \vdots \\ \overline{V}_d \end{bmatrix} = \begin{bmatrix} P\overline{W}_d \\ \vdots \\ P\overline{W}_1 \\ W_0 \\ W_1 \\ \vdots \\ W_d \end{bmatrix}. \tag{16}$$

This linear system features $2d+1$ row blocks of n^2 equations for $(d+1)n^2$ unknowns. Now we prove that slightly less than half of these equations are actually redundant.

On the one hand, consider for instance the p-th row block in (16). It reads

$$\sum_{q=0}^{d} \overline{Q}_{-p+q+d+1} V_q + PQ_{p+q-d-1} \overline{V}_q = P\overline{W}_{-p+d+1}$$

where it is understood that $1 \leq p \leq d$ and $Q_q = \overline{Q}_q = \mathbf{0}$ if $q < 0$ or $q > d$. Upon multiplication by P and complex conjugation, the above equation becomes

$$\sum_{q=0}^{d} \overline{Q}_{p+q-d-1} V_q + PQ_{-p+q+d+1} \overline{V}_q = W_{-p+d+1}$$

which is nothing but the $(p+d+1)$-th row block in (16). As a result, the first d row blocks and the last d row blocks describe the same equalities.

On the other hand, one can check that in the central $(d+1)$-th row block

$$\begin{bmatrix} \overline{Q}_0 & \overline{Q}_1 & \cdots & \overline{Q}_d \end{bmatrix} \underbrace{\begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_d \end{bmatrix}}_{X_M} + P \begin{bmatrix} Q_0 & Q_1 & \cdots & Q_d \end{bmatrix} \underbrace{\begin{bmatrix} \overline{V}_0 \\ \overline{V}_1 \\ \vdots \\ \overline{V}_d \end{bmatrix}}_{\overline{X}_M} = W_0 \tag{17}$$

there are $n(n-1)/2$ redundant equalities since $B_0 = \overline{B}_0^T$, hence $W_0 = P\overline{W}_0$. They can be replaced by $n(n-1)/2$ equalities forcing the lower-triangular components of $X(0)$ to zero. In matrix notation, (17) is then replaced by the non-redundant n^2 equalities

$$\left[\begin{array}{cccc} \multicolumn{4}{c}{T} \\ \hline \Pi\overline{Q}_0 & \Pi\overline{Q}_1 & \cdots & \Pi\overline{Q}_d \end{array}\right] X_M + \left[\begin{array}{cccc} \multicolumn{4}{c}{\mathbf{0}} \\ \hline \Pi PQ_0 & \Pi PQ_1 & \cdots & \Pi PQ_d \end{array}\right] \overline{X}_M = \begin{bmatrix} \mathbf{0} \\ \Pi W_0 \end{bmatrix}$$

where matrix Π extracts the $n(n+1)/2$ non-redundant equalities in (17) and T selects the $n(n-1)/2$ lower-triangular components of $X(0)$ in X_M. Based on the above two points, linear system (16) is equivalent to

$$\underbrace{\left[\begin{array}{ccccc} \multicolumn{5}{c}{T} \\ \hline \Pi\overline{Q}_0 & \Pi\overline{Q}_1 & \Pi\overline{Q}_2 & \cdots & \Pi\overline{Q}_d \\ \mathbf{0} & \overline{Q}_0 & \overline{Q}_1 & \cdots & \overline{Q}_{d-1} \\ & & \overline{Q}_0 & & \vdots \\ & & & \ddots & \overline{Q}_1 \\ \mathbf{0} & & & & \overline{Q}_0 \end{array}\right]}_{\overline{A}_{1_M}} X_M + \underbrace{\left[\begin{array}{ccccc} \multicolumn{5}{c}{\mathbf{0}} \\ \hline \Pi PQ_0 & \Pi PQ_1 & \cdots & \Pi PQ_{d-1} & \Pi PQ_d \\ PQ_1 & PQ_2 & \cdots & PQ_d & \mathbf{0} \\ \vdots & & \iddots & & \\ PQ_{d-1} & PQ_d & & & \\ PQ_d & & & & \mathbf{0} \end{array}\right]}_{A_{2_M}} \overline{X}_M = \underbrace{\left[\begin{array}{c} \mathbf{0} \\ \hline \Pi W_0 \\ W_1 \\ \vdots \\ W_d \end{array}\right]}_{B_M}$$

which is a linear system of $(d+1)n^2$ complex equations with $(d+1)n^2$ complex unknowns. Note that one can also obtain this reduced system of equations through the equivalent approach pursued in [4] for the case of real polynomial matrices.

Using Fact 1, the above complex system can be written over the field of real numbers as

$$\underbrace{\left(\begin{bmatrix} \mathrm{Re}A_{1_M} & \mathrm{Im}A_{1_M} \\ -\mathrm{Im}A_{1_M} & \mathrm{Re}A_{1_M} \end{bmatrix} + \begin{bmatrix} \mathrm{Re}A_{2_M} & \mathrm{Im}A_{2_M} \\ \mathrm{Im}A_{2_M} & -\mathrm{Re}A_{2_M} \end{bmatrix}\right)}_{\tilde{\mathbf{A}}_{\mathbf{M}}} \underbrace{\begin{bmatrix} \mathrm{Re}X_M \\ \mathrm{Im}X_M \end{bmatrix}}_{\tilde{\mathbf{X}}_{\mathbf{M}}} = \underbrace{\begin{bmatrix} \mathrm{Re}B_M \\ \mathrm{Im}B_M \end{bmatrix}}_{\mathbf{B}_{\mathbf{M}}} \tag{18}$$

Under Assumptions 3, imaginary parts of the diagonal entries of $X(0)$ are zero. Thus the above equation can further be simplified to

$$\mathbf{A}_{\mathbf{M}}\mathbf{X}_{\mathbf{M}} = \mathbf{B}_{\mathbf{M}} \tag{19}$$

where square matrix $\mathbf{A}_{\mathbf{M}}$ is obtained upon removing the n columns of matrix $\tilde{\mathbf{A}}_{\mathbf{M}}$ corresponding to the imaginary part of the diagonal entries of $X(0)$ and the n rows of matrix $\tilde{\mathbf{A}}_{\mathbf{M}}$ corresponding to the zero imaginary parts of the diagonal entries of $B(0)$, and vector $\mathbf{X}_{\mathbf{M}}$ is obtained upon removing the corresponding n rows of vector $\tilde{\mathbf{X}}_{\mathbf{M}}$. The dimension of matrix $\mathbf{A}_{\mathbf{M}}$ is $2(d+1)n^2 - n$.

Theorem 5 (Algorithm MATSYL) *Under the requirements of Theorem 4, Every solution $X(\mathsf{z})$ to symmetric polynomial equation corresponds to a solution to linear system of equations (18). In particular, the unique solution such that $X(0)$ is upper-triangular with real diagonal entries can be found by solving the non-singular system of equations (19).*

5.3 Illustration

Algorithms MATRED and MATSYL are now illustrated by way of an example. For simplicity, we assume that matrix $A(\mathsf{z})$ is already in the triangular form described in Lemma 1. We choose

$$A(\mathsf{z}) = \begin{bmatrix} (1-4\mathrm{j}) + 3\mathrm{j}\mathsf{z} & 4+\mathsf{z} \\ 0 & 5+(1-2\mathrm{j})\mathsf{z} \end{bmatrix}$$

and

$$B(\mathsf{z}) = \begin{bmatrix} -3\mathrm{j}\mathsf{z}^{-1} + 2 + 3\mathrm{j}\mathsf{z} & 6\mathsf{z}^{-1} - (4+\mathrm{j}) + (2+4\mathrm{j})\mathsf{z} \\ (2-4\mathrm{j})\mathsf{z}^{-1} - (4-\mathrm{j}) + 6\mathsf{z} & (7+8\mathrm{j})\mathsf{z}^{-1} + 32 + (7-8\mathrm{j})\mathsf{z} \end{bmatrix}.$$

5.4 Polynomial Reduction Algorithm

In matrix equation (10), entry (1,1) reads

$$(-3\mathrm{j}\mathtt{z}^{-1} + (1+4\mathrm{j}))x_{11}(\mathtt{z}) + x_{11}^{\star}(\mathtt{z})((1-4\mathrm{j}) + 3\mathrm{j}\mathtt{z}) = -3\mathrm{j}\mathtt{z}^{-1} + 2 + 3\mathrm{j}\mathtt{z}.$$

Using Algorithm ScalRed, we get

$$x_{11}(\mathtt{z}) = 1.$$

Entry (1,2) reads

$$\begin{array}{c}(-3\mathrm{j}\mathtt{z}^{-1} + (1+4\mathrm{j}))x_{12}(\mathtt{z}) + x_{11}^{\star}(\mathtt{z})(4+\mathtt{z}) + x_{21}^{\star}(\mathtt{z})(5+(1-2\mathrm{j})\mathtt{z}) = \\ 6\mathtt{z}^{-1} - (4+\mathrm{j}) + (2+4\mathrm{j})\mathtt{z}\end{array}$$

or equivalently

$$(-3\mathrm{j}\mathtt{z}^{-1} + (1+4\mathrm{j}))x_{12}(\mathtt{z}) + x_{21}^{\star}(\mathtt{z})(5+(1-2\mathrm{j})\mathtt{z}) = 6\mathtt{z}^{-1} - (8+\mathrm{j}) + (1+4\mathrm{j})\mathtt{z}.$$

Using Algorithm ScalRed2, we get

$$\begin{aligned} x_{12}(\mathtt{z}) &= 2\mathrm{j} + \mathtt{z} \\ x_{21}(\mathtt{z}) &= 0. \end{aligned}$$

Finally, entry (2,2) reads

$$\begin{array}{c}(\mathtt{z}^{-1}+4)x_{12}(\mathtt{z}) + ((1+2\mathrm{j})\mathtt{z}^{-1}+5)x_{22}(\mathtt{z}) + x_{12}^{\star}(\mathtt{z})(4+\mathtt{z}) + x_{22}^{\star}(\mathtt{z})(5+(1-2\mathrm{j})\mathtt{z}) = \\ (7+8\mathrm{j})\mathtt{z}^{-1} + 32 + (7-8\mathrm{j})\mathtt{z}\end{array}$$

or equivalently

$$((1+2\mathrm{j})\mathtt{z}^{-1}+5)x_{22}(\mathtt{z}) + x_{22}^{\star}(\mathtt{z})(5+(1-2\mathrm{j})\mathtt{z})) = (3+6\mathrm{j})\mathtt{z}^{-1} + 30 + (3-6\mathrm{j})\mathtt{z}.$$

Using Algorithm ScalRed, we get

$$x_{22}(\mathtt{z}) = 3,$$

hence the required solution to equation (10) reads

$$X(\mathtt{z}) = \begin{bmatrix} 1 & 2\mathrm{j}+\mathtt{z} \\ 0 & 3 \end{bmatrix}.$$

5.5 Sylvester Matrix Algorithm

Following the development of Algorithm MatSyl in Section 5.2, we obtain the linear system of complex equations

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1-4\mathrm{j} & 0 & 0 & 0 & 3\mathrm{j} & 0 & 0 & 0 \\ 0 & 0 & 1-4\mathrm{j} & 0 & 0 & 0 & 3\mathrm{j} & 0 \\ 0 & 0 & 4 & 5 & 0 & 0 & 1 & 1-2\mathrm{j} \\ 0 & 0 & 0 & 0 & 1-4\mathrm{j} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1-4\mathrm{j} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 5 \end{bmatrix}}_{\overline{A_1}_M} X_M +$$

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1-4\mathrm{j} & 0 & 0 & 0 & 3\mathrm{j} & 0 & 0 & 0 \\ 4 & 5 & 0 & 0 & 1 & 1-2\mathrm{j} & 0 & 0 \\ 0 & 0 & 4 & 5 & 0 & 0 & 1 & 1-2\mathrm{j} \\ 3\mathrm{j} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3\mathrm{j} & 0 & 0 & 0 & 0 & 0 \\ 1 & 1-2\mathrm{j} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1-2\mathrm{j} & 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{2M}} \overline{X}_M = \underbrace{\begin{bmatrix} 0 \\ 2 \\ -4-\mathrm{j} \\ 32 \\ 3\mathrm{j} \\ 6 \\ 2+4\mathrm{j} \\ 7-8\mathrm{j} \end{bmatrix}}_{B_M}.$$

Using Fact 1 and upon suppression of the 9th and 12th columns corresponding to the imaginary part of the diagonal entries of $X(0)$ and the 10th and 12th rows corresponding to the zero imaginary parts of the diagonal entries of $B(0)$, we get the equivalent real system of dimension 14:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 0 & 0 & 0 \\ 4 & 5 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & -4 & 0 & -2 & 3 & 0 \\ 0 & 0 & 8 & 10 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & -4 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 5 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -2 & 0 & 0 & 0 & -4 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 4 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & -2 & -3 & 0 & -5 & 1 & -1 & -1 & 0 & 0 \\ 3 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 5 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 & 0 & 4 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 4 & 5 \end{bmatrix} \mathbf{X_M} = \begin{bmatrix} 0 \\ 2 \\ -4 \\ 32 \\ 0 \\ 6 \\ 2 \\ 7 \\ 0 \\ -1 \\ 3 \\ 0 \\ 4 \\ -8 \end{bmatrix}.$$

Solving this linear system of equations yields the unique solution $X(\mathtt{z})$ to (10) such that $X(0)$ is upper-triangular with real diagonal entries, namely

$$X(\mathtt{z}) = \begin{bmatrix} 1 & 2\mathrm{j}+\mathtt{z} \\ 0 & 3 \end{bmatrix}.$$

6 Numerical Considerations

We have proposed two kinds of techniques for dealing with the scalar symmetric polynomial equation (3) and its matrix counterpart (10).

- Algorithms based on complex polynomial reduction, a sophisticated version of the Euclidean division for polynomials: Algorithm ScalRed for the scalar case and Algorithm MatRed for the matrix case, the latter relying upon intermediate Algorithms ScalRed2 and LURed.
- Algorithms based on Sylvester matrices and linear systems of equations: Algorithm ScalSyl for the scalar case and Algorithm MatSyl for the matrix case.

The first category of algorithms is clearly of theoretical interest, since at the core of the proofs of main Theorems 1 and 4. However, it is now well recognized that these algorithms are unfortunately prone to numerical instability since based on elementary polynomial operations. Moreover, in order to apply the polynomial reduction algorithms, restrictive additional assumptions are required.

The second category of algorithms only relies upon the resolution of linear systems of equations, for which powerful and numerically reliable tools such as the singular value decomposition are now widely available [3]. Therefore Algorithms ScalSyl and MatSyl can be considered as numerically reliable alternatives to Algorithms ScalRed and MatRed, respectively. Note that, in contrast to polynomial reduction algorithms, no additional assumptions are required to apply Sylvester matrix algorithms.

Matlab implementations of algorithms ScalRed, MatRed, ScalSyl and MatSyl are included to the latest version of the Polynomial Toolbox [19].

7 Concluding Remarks

After a generalization of classical polynomial matrix notions to the complex case, we proposed conditions of existence of a unique solution to scalar and matrix symmetric polynomial equations. Our proofs were constructive and resulted in a first family of resolution algorithms, based on polynomial reductions, a sophisticated version of the Euclidean division algorithm for polynomials. In order to overcome potential numerical instability, we also developed alternative Sylvester matrix algorithms only relying upon well-known and reliable tools from numerical linear algebra. It must be underlined that all these algorithms are available in the latest version of the Polynomial Toolbox for Matlab, see [19].

Possible directions for further research are now mentioned. The extension to the complex case must be generalized to other types of algorithms on polynomial matrices. In particular, interpolation techniques as presented in [6, 7] would undoubtedly benefit from such an extension. Indeed, it is well-known that the conditioning of the Vandermonde matrix is perfect when interpolation points are chosen as the complex roots of unity [8, §21.1]. It is also necessary to distinguish between matrix polynomial algorithms that really require a special treatment to handle complex coefficients. Some of them, such as the basic linear matrix polynomial equation solver implemented in [19], straightforwardly generalize to the complex case because they rely on standard numerical linear algebra tools (e.g. QR decomposition or singular value decomposition [3]) already dealing with complex numbers.

References

[1] B. R. Barmish, New Tools for Robustness of Linear Systems, *Macmillan*, New York, 1994.

[2] N. K. Bose and Y. Q. Shi, A Simple General Proof of Kharitonov's Generalized Stability Criterion, *IEEE Transactions on Circuits and Systems*, Vol. 34, pp. 1233–1237, 1987.

[3] G. H. Golub and C. F. Van Loan, Matrix Computations, Third Edition, *Johns Hopkins University Press*, Baltimore, Maryland, 1996.

[4] D. Henrion and M. Šebek, An Efficient Numerical Method for the Discrete-Time Symmetric Matrix Polynomial Equation, *IEE Proceedings, Control Theory and Applications*, Vol. 145, No. 5, pp. 443–448, 1998.

[5] D. Henrion, Reliable Algorithms for Polynomial Matrices, *Ph. D. Thesis*, Control Theory Department, Institute of Information Theory and Automation, Prague, Czech Republic, 1998.

[6] D. Henrion and M. Šebek, Symmetric Matrix Polynomial Equation: Interpolation Results, *Automatica*, Vol. 34, No. 7, pp. 811–824, 1998.

[7] D. Henrion and M. Šebek, Reliable Numerical Methods for Polynomial Matrix Triangularization, *IEEE Transactions on Automatic Control*, Vol. 44, No. 3, pp. 497–508, 1999.

[8] N. J. Higham, Accuracy and Stability of Numerical Algorithms, *SIAM*, Philadelphia, Pennsylvania, 1996.

[9] J. Ježek, Conjugated and Symmetric Polynomial Equations, Part II: Discrete-Time Systems, *Kybernetika*, Vol. 19, No. 3, pp. 196–211, 1983.

[10] J. Ježek and V. Kučera, Efficient Algorithm for Matrix Spectral Factorization, *Automatica*, Vol. 21, No. 6, pp. 663–669, 1985.

[11] J. Ježek, Symmetric Matrix Polynomial Equations, *Kybernetika*, Vol. 22, No. 1, pp. 19–30, 1986.

[12] J. Ježek and K. J. Hunt, Coupled Polynomial Equations of LQ Control Synthesis and an Algorithm for Solution, *International Journal of Control*, Vol. 58, No. 5, pp. 1155–1167, 1993.

[13] T. Kailath, Linear Systems, *Prentice Hall*, Englewood Cliffs, New Jersey, 1980.

[14] V. Kučera, Discrete Linear Control: The Polynomial Approach, *John Wiley and Sons*, Chichester, Great Britain, 1979.

[15] V. Kučera, Analysis and Design of Discrete Linear Control Systems, *Prentice Hall*, London, Great Britain, 1991.

[16] L. Lindbom, A Wiener Filtering Approach to the Design of Tracking Algorithms with Applications to Mobile Radio Communications, *Ph. D. Thesis*, Signal Processing Group, Department of Technology, Uppsala University, Sweden, 1995.

[17] O. MacChi, Adaptive Processing: the Least Mean Squares Approach and Applications in Transmission, *John Wiley and Sons*, Chichester, Great Britain, 1995.

[18] J. G. Proakis and M. Salehi, Communication Systems Engineering, *Prentice Hall*, Englewood Cliffs, New Jersey, 1994.

[19] M. Šebek, H. Kwakernaak, D. Henrion and S. Pejchová, Recent Progress in Polynomial Methods and Polynomial Toolbox for MATLAB Version 2.0, *Proceedings of the Conference on Decision and Control*, IEEE, pp. 3661–3668, Tampa, Florida, 1998. See also the web page `www.polyx.com`.

[20] T. Söderström, J. Ježek and V. Kučera, An Efficient and Versatile Algorithm for Computing the Covariance Function of an ARMA Process, *IEEE Transactions on Signal Processing*, Vol. 46, No. 6, pp. 1591–1600, 1998.

[21] J. C. Willems and H. L. Trentelman, On Quadratic Differential Forms, *SIAM Journal on Control and Optimization*, Vol. 36, No. 5, pp. 1703–1749, 1998.