Applied Numerical Mathematics 109 (2016) 1–18

Contents lists available at ScienceDirect

Applied Numerical Mathematics

www.elsevier.com/locate/apnum

Adaptive model order reduction for the Jacobian calculation in inverse multi-frequency problem for Maxwell's equations

M. Kordy [a,b,*], E. Cherkaev [a,1], P. Wannamaker [b,2]

[a] Department of Mathematics, University of Utah, 155 S 1400 E JWB 233, Salt Lake City, UT 84112, USA
[b] Energy & Geoscience Institute, University of Utah, 423 Wakara Way #300, Salt Lake City, UT 84108, USA

ARTICLE INFO

Article history:
Received 9 March 2015
Received in revised form 18 February 2016
Accepted 25 February 2016
Available online 11 June 2016

Keywords:
Rational Krylov subspace
Model order reduction
Inverse problem
Frequency-domain Maxwell's equations

ABSTRACT

This work develops a model order reduction method for a numerical solution of an inverse multi-frequency eddy current problem using a rational interpolation of the transfer function in the complex plane. We use the Pade interpolation in the complex frequency plane; this allows us to speed up the calculation of the frequency-dependent Jacobian in the inversion procedure without loosing accuracy. Interpolating frequencies are chosen adaptively to reduce the maximal approximation error. We use the error indicator that is equivalent to a seminorm of the residual. The efficiency of the developed approach is demonstrated by applying it to the inverse magnetotelluric problem, which is a geophysical electromagnetic remote sensing method used in mineral, geothermal, and groundwater exploration. In this application, the transfer function values are needed for shifts in a purely imaginary interval. Thus we consider the interpolating shifts in the same interval as well as in a purely real interval, containing the spectrum of the operator. Numerical tests show an excellent performance of the proposed methods characterized by a significant reduction of computational time without loss of accuracy of the calculated Jacobian.

1. Introduction

Model order reduction (MOR) is a powerful technique for reduction of the dimensionality of a problem. It is especially efficient when the low dimensional subspace is generated using rational Krylov subspaces. This approach has become popular recently and has been used in a variety of contexts [33,1,4,14,28,32,31,2,11,13,15,16]. The efficiency of the method is amplified significantly by an optimal selection of the shifts for generating the rational Krylov subspace. For a particular matrix with uniform spectrum, the problem of selection of the optimal shifts has been investigated in [27,34,36]. However, this selection is not optimal for matrices with non-uniform spectrum. An excellent review of this topic is presented in [17].

A different approach, in which the shifts are added one by one in a greedy fashion, was developed in [7,5]. In this approach, the shifts are adapted to the spectrum of the matrix. The authors of [7] consider an adaptive choice of shifts for the approximation of the transfer function using rational Krylov subspaces, with an application to a time-domain electromagnetic geophysical forward problem. This adaptive choice of shifts was later generalized to non-symmetric matrices

* Corresponding author at: Energy & Geoscience Institute, University of Utah, 423 Wakara Way #300, Salt Lake City, UT 84108, USA. Tel.: +1 801 581 5126.
E-mail addresses: kordy@math.utah.edu (M. Kordy), elena@math.utah.edu (E. Cherkaev), pewanna@egi.utah.edu (P. Wannamaker).
[1] Tel.: +1 801 581 7315.
[2] Tel.: +1 801 581 3547.

http://dx.doi.org/10.1016/j.apnum.2016.02.010
0168-9274/

[8] and to approximation of matrix functions other than the transfer function [18]. Numerical simulations in [7,18] show that the adaptive approach gives better results than the choice of the shifts that do not depend on the spectrum of the operator. Moreover, these numerical results demonstrate that the number of required shifts is not increased with the size of the system, and it is not strongly dependent on the spectrum. Those results encourage us to pursue an adaptive choice of the shifts in the current work.

The work of [7] and the rational Krylov subspace approach was extended to the inverse problem [9,38], where the model order reduction may be viewed as constraint on the set of admissible conductivity models or as a way to calculate a good approximation of the Jacobian with a greatly reduced computational time. For many inverse problems solved using a method (like Gauss–Newton) requiring a Jacobian, significant and sometimes dominant computational cost is related to its calculation. In the current paper, we use the model order reduction through rational Krylov subspaces only to speedup the calculations of the transfer function used in the evaluation of the Jacobian, with no loss of accuracy. The proposed method can be used with any standard inversion approach that requires a calculation of the Jacobian.

We develop the adaptive choice of optimal shifts method with an application to magnetotellurics (MT), which is a frequency domain electromagnetic remote-sensing geophysical method used in mineral, geothermal, and groundwater exploration. Numerically simulating the scattering of EM waves from complex three-dimensional (3D) structure is a computationally demanding problem [6,12]. In particular, the scattering response usually needs to be calculated over a broad frequency range. Typically this may be five orders of magnitude or more with frequency sampling of 5–10 base points per decade. A considerable savings in computational time could result if an accurate method of interpolation of responses across a coarser selection of base points can be achieved.

In this case, for the forward problem, the transfer function $\tilde{h}(s) = (\tilde{A} + s\tilde{B})^{-1}\tilde{b}$ has a complex valued right hand side $\tilde{b}$ dependent on frequency; the model order reduction method that allows to treat such problem is developed in [24]. In the case of the calculation of the Jacobian, the rhs $\tilde{b}$ is not dependent on frequency and is real valued. In our application, $\tilde{B}$ is not the identity matrix. We show that it is related to the case of $\tilde{B} = I$ through a scaling $\tilde{B}^{1/2}$. The scaling is useful to simplify the analysis, but its calculation is not required in the approximation procedure. We also present a proof of the transfer function approximation error formula. The proof is different than the one in [22].

The transfer function values are required in a purely imaginary interval, so we consider the interpolating shifts in the same imaginary interval. We also consider real shifts, following the suggestion of [7]. The shifts are chosen to reduce the maximal error of interpolation and as the true error is unknown, an error indicator is needed. We consider error indicator suggested by [7] and we show that it is equivalent to any seminorm of the residual. The speedup of the algorithm with the model order reduction is higher when the number of frequencies considered in MT survey increases. In our numerical tests, the speedup of calculation of the Jacobian for 30 frequencies, is 4 times.

The paper is organized as follows. In section 2 we present the formulation of the forward magnetotelluric problem and Jacobian required in the inversion procedure. We explain how the approximation of the transfer function may be used to speedup the calculations. Next we show the theory of the approximation of the transfer function using rational Krylov subspaces. In section 3 we present the error indicator function and present two algorithms based on them. We also give details of the numerical implementation and discuss a possibility of using quadruple precision for some of the non-computationally demanding calculations. In section 4 we show results of numerical tests for a 3D magnetotelluric model with non-constant conductivity structure and with a hill and a valley in topography.

2. Theory

2.1. Inverse magnetotelluric problem

The forward and inverse magnetotelluric (MT) problem is described in detail in [25,26]. We consider a domain Ω that includes both the air and the earth's subsurface. The earth's surface is allowed to have a non-flat topography. In order to calculate the MT response due to an arbitrary 3D conductivity structure $\sigma > 0$ we consider the edge finite element discretization of the equation for the secondary electric field E. Though the numerical tests presented are done using lowest order edge hexahedral discretization, all the methods may be applied to other discretizations, such as the tetrahedral mesh, higher order edge elements as well as the finite difference method.

The solution space for the unknown electric field is defined as

$$\mathcal{H}_0(\nabla\times,\Omega) = \left\{F:\Omega\to\mathbb{C}^3 : \int_\Omega \left(|F|^2 + |\nabla\times F|^2\right) < \infty,\ \ n\times F|_{\partial\Omega} = 0\right\} \tag{1}$$

Consider Maxwell's equations in the frequency domain for a low angular frequency ω. We denote the magnetic permeability by μ and the permittivity by ϵ. The term $i\omega\epsilon$, related to the displacement current is neglected. The equation for the secondary field $E \in \mathcal{H}_0(\nabla\times)$ is

$$\int_\Omega \frac{1}{\mu}\nabla\times E\cdot\nabla\times F + i\omega\int_\Omega \sigma E\cdot F = \int_\Omega -i\omega(\sigma-\sigma^p)E^p\cdot F \tag{2}$$

for all test functions $F \in \mathcal{H}_0(\nabla\times)$. The source term in (2) depends on primary electric field E^p, which is a plane wave traveling in the medium of the primary conductivity σ_p of a 1D earth. The conductivity σ is an arbitrary nonnegative function of position in the 3D domain, satisfying the assumption that $\sigma \approx \sigma_p$ close to the domain boundaries.

The electric field over Ω is represented as a linear combination of the edge shape functions S_i with coefficients ξ_i:

$$E = \sum_{i=1}^{N} \xi_i S_i \tag{3}$$

where $i = 1, \ldots, N$ are indices of the edges in Ω that do not lie on the boundary. By substituting this to equation (2) and using S_j as test functions, one obtains a linear system

$$(\tilde{A} + \mathrm{i}\omega\tilde{B})\xi = g \tag{4}$$

$$\tilde{A}_{j,k} = \int_{\Omega} \frac{1}{\mu} \nabla \times S_j \cdot \nabla \times S_k, \quad \tilde{B}_{j,k} = \int_{\Omega} \sigma S_j \cdot S_k, \quad g_j = g_j(\omega, \sigma) = \int_{\Omega} -\mathrm{i}\omega(\sigma - \sigma^p) E^p \cdot S_j \tag{5}$$

The secondary magnetic field H is calculated as

$$H = \frac{-\nabla \times E}{i\omega\mu} \tag{6}$$

The total field E^t, H^t is a sum of secondary and primary fields:

$$E^t = E + E^p, \quad H^t = H + H^p \tag{7}$$

The MT response at a receiver is obtained by finding the impedance Z and the tipper K such that

$$\begin{bmatrix} E_x^t \\ E_y^t \\ H_z^t \end{bmatrix} = \begin{bmatrix} Z_{xx} & Z_{xy} \\ Z_{yz} & Z_{yy} \\ K_{zx} & K_{zy} \end{bmatrix} \begin{bmatrix} H_x^t \\ H_y^t \end{bmatrix} \tag{8}$$

is satisfied no matter what the polarization of the primary (E^p, H^p) plane wave is.

A receiver can be positioned at an arbitrary location $\mathbf{r}$ with respect to element edges via appropriate interpolation. In hexahedral discretization, an element has edges $e_1, \ldots, e_{12}$. The field E at a location $\mathbf{r}$ inside this element is given by

$$E(\mathbf{r}) = \sum_{l=1}^{12} S_{e_l}(\mathbf{r}) \xi_{e_l} = \begin{bmatrix} (v_x^E)^T \xi \\ (v_y^E)^T \xi \\ (v_z^E)^T \xi \end{bmatrix} \tag{9}$$

Here v_x^E, v_y^E, v_z^E contain interpolation vectors with at most 12 non-zero values corresponding to x, y and z components of the edge shape functions $S_{e_1}(\mathbf{r}), \ldots, S_{e_{12}}(\mathbf{r})$.

Similarly, the secondary magnetic field $H(\mathbf{r})$, calculated using (6) at the location $\mathbf{r}$, is given by

$$H(\mathbf{r}) = \sum_{l=1}^{12} \frac{\nabla \times S_{e_l}(\mathbf{r})}{-i\omega\mu} \xi_{e_l} = \frac{1}{i\omega} \begin{bmatrix} (v_x^H)^T \xi \\ (v_y^H)^T \xi \\ (v_z^H)^T \xi \end{bmatrix} \tag{10}$$

This time the only non-zero values of v_x^H, v_y^H, v_z^H are x, y and z components of

$$\left(\frac{\nabla \times S_{e_1}(\mathbf{r})}{-\mu}, \ldots, \frac{\nabla \times S_{e_{12}}(\mathbf{r})}{-\mu} \right)$$

As a result each component of the secondary electric and magnetic fields E, H at a specific receiver location may be represented using

$$v^T \xi = v^T \left[(\tilde{A} + \mathrm{i}\omega\tilde{B})^{-1} g \right] \tag{11}$$

where v is a real valued vector, $g = g(\omega)$ is complex valued and one should write $\frac{g}{\mathrm{i}\omega}$ in place of g in the case of H. The evaluation of (11) is done in such a way that the quantity in square brackets is calculated first and then it is multiplied by v^T. This gives us the values of the secondary electric and magnetic fields at a receiver location, this is sufficient for calculation of the MT response.

In the inversion of the MT data one seeks a conductivity model that fits the measured data. One of the ways to find it is to use the Gauss–Newton algorithm (see for example [26]) to minimize the functional that consists of a data misfit term and a regularization term. In order to use this algorithm one needs the Jacobian J of the response functional. Let the domain be split into N_m inversion cells C_j, which form a partition of the subsurface part of the domain (see Fig. 1). We assume that each cell consists of a number of finite elements and the conductivity σ is constant in each cell. Let σ_j denote

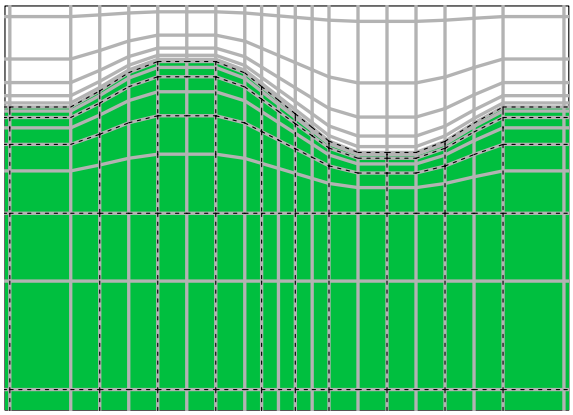

Fig. 1. A central part of a cross-section of an example domain Ω with the earth's subsurface in green and the air in white. Finite element mesh is plotted in gray, while the boundaries of inversion cells are plotted with a dashed black line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the value of the conductivity in an inversion cell C_j. The conductivity in the whole earth's subsurface is given by the vector of conductivities $(\sigma_j)_{j=1}^{N_m}$. Each entry of the Jacobian matrix J consists of values of the derivative of the magnetotelluric response Z, K at some receiver location $\mathbf{r}$ with respect to σ_j, for some $\mathbf{r}$ and some $j = 1, \ldots, N_m$. The response is a function of the secondary electromagnetic field E, H, so its derivative with respect to σ_j may be calculated using the chain rule, if the derivatives

$$\frac{\partial E}{\partial \sigma_j}, \quad \frac{\partial H}{\partial \sigma_j}$$

are found. In order to find the latter, one has to evaluate the derivative of expression (11):

$$\begin{aligned}
&\tfrac{\partial}{\partial\sigma_j}\left(v^T\left((\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}g\right)\right) = v^T\tfrac{\partial\left((\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}g\right)}{\partial\sigma_j}\\
&= v^T\left(\tfrac{\partial(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}}{\partial\sigma_j}g + (\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tfrac{\partial g}{\partial\sigma_j}\right)\\
&= v^T\left(-(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tfrac{\partial(\tilde{A}+\mathrm{i}\omega\tilde{B})}{\partial\sigma_j}(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}g + (\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tfrac{\partial g}{\partial\sigma_j}\right)\\
&= v^T\left(-(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tfrac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}\xi + (\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tfrac{\partial g}{\partial\sigma_j}\right)\\
&= v^T\left((\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\left(-\tfrac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}\xi + \tfrac{\partial g}{\partial\sigma_j}\right)\right)\\
&= \left(v^T(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\right)\left(-\tfrac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}\xi + \tfrac{\partial g}{\partial\sigma_j}\right) = \left[(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}v\right]^T\left(-\tfrac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}\xi + \tfrac{\partial g}{\partial\sigma_j}\right)
\end{aligned} \tag{12}$$

where ξ and g should be replaced by $\frac{\xi}{\mathrm{i}\omega}$ and $\frac{g}{\mathrm{i}\omega}$ in the case of H. In the last calculation, the expression in square brackets is evaluated first and then it is multiplied by $\left(-\frac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}\xi + \frac{\partial g}{\partial\sigma_j}\right)$ for each σ_j. Notice that one has to find ξ before the multiplication is done and that both the matrix $\frac{\partial\mathrm{i}\omega\tilde{B}}{\partial\sigma_j}$ and the vector $\frac{\partial g}{\partial\sigma_j}$ are sparse.

Jacobian entries are smooth functions of frequency, so they may be efficiently interpolated between frequencies. In Fig. 2 we present as an example, an element of the Jacobian matrix related to K_{zx}, plotted in the complex plane as the frequency is changed. One can see that the Jacobian entries may be complicated functions of frequency. In this case the piecewise linear or a high order polynomial interpolation is not accurate enough. The rational interpolation through the model order reduction, which is the topic of this paper, gives a more accurate approximation.

2.2. The model order reduction

Values of the E, H fields (11) and the Jacobian (12) may be approximated in such a way, that the vectors in square brackets in (11), (12) are approximated first, and then they are multiplied by the remaining part of expressions (11) or (12). Thus in both cases we are interested in an approximation of the expression

$$\tilde{h}(s) = (\tilde{A}+s\tilde{B})^{-1}\tilde{b} \tag{13}$$

Here the shift $s = \mathrm{i}\omega$ for some chosen finite number of frequencies ω at which the MT measurements are taken. Those frequencies are usually log-uniformly distributed in an interval $[\omega_{\min}, \omega_{\max}]$. In the case of (11), $\tilde{b} = g(\omega)$, so it is dependent

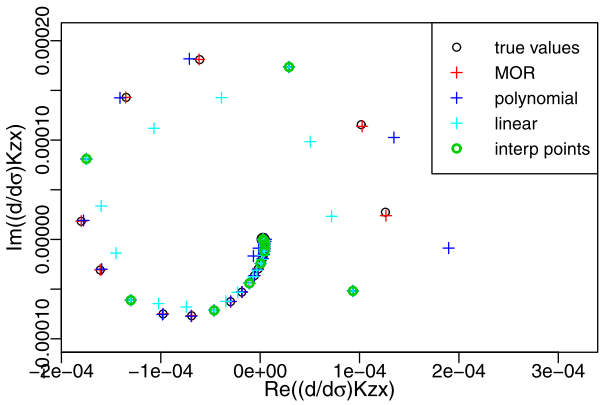

Fig. 2. Derivative of K_{zx} in the complex plane for the model considered in section 4 for 31 frequencies log-uniformly distributed in the interval [1 Hz, 1000 Hz]. True values are shown together with the high order polynomial, piecewise linear and model order reduction interpolation. Every third value (shown by green circles) is used as an interpolation point. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

on s and complex valued. In the case of (12), $\tilde{b} = v$, so it is not dependent on s and real valued. Matrix $\tilde{A}$, defined by (5) is real valued, symmetric, nonnegative definite, with a significant null space. We assume that the conductivity $\sigma > 0$, so matrix $\tilde{B}$, defined in (5) is real valued, symmetric, positive definite. In this paper we consider the interpolation of the Jacobian, when $\tilde{b}$ does not depend on s. For the approximation of the forward response, given by (11), see [24].

We will construct an approximation of (13) using the model order reduction method [9,38,22,7,35,37,19,2,13,15,16].

Let us start with an equation satisfied by $\tilde{h}$:

$$(\tilde{A} + s\tilde{B})\tilde{h} = \tilde{b} \tag{14}$$

Consider $\tilde{V}$, which is an $N \times n$ matrix whose columns span the space

$$\operatorname{colsp}(\tilde{V}) = \operatorname{span}\left\{(\tilde{A} + s_1\tilde{B})^{-1}\tilde{b}, (\tilde{A} + s_2\tilde{B})^{-1}\tilde{b}, \ldots, (\tilde{A} + s_n\tilde{B})^{-1}\tilde{b}\right\} \tag{15}$$

for some chosen complex values $s_1, \ldots, s_n$, which satisfy

$$s_i \neq s_j \text{ if } i \neq j \tag{16}$$

As $\tilde{A}$ is nonnegative definite and $\tilde{B}$ is positive definite, eigenvalues of $\tilde{B}^{-1/2}\tilde{A}\tilde{B}^{-1/2}$ are in $[0, \infty)$. Thus, in order for the equation (13) to have a solution, we assume that

$$s,\ s_j \notin (-\infty, 0] \tag{17}$$

Let $\beta \in \mathbb{C}^n$ be such that the approximation of the solution of (14) in $\operatorname{colsp}(\tilde{V})$ is given by $\tilde{h}_{\tilde{V}} = \tilde{V}\beta$. Making the residual orthogonal to $\operatorname{colsp}(\tilde{V})$, we obtain an equation for β:

$$\tilde{V}^*(\tilde{A} + s\tilde{B})(\tilde{V}\beta) = \tilde{V}^*\tilde{b} \tag{18}$$

where * denotes Hermitian conjugation. If this equation has a unique solution (see Theorem 2.1), then we obtain the approximation $h_{\tilde{V}}(s) = \tilde{V}\beta$ to $h(s)$:

$$\tilde{h}_{\tilde{V}}(s) = \tilde{V}\left(\tilde{V}^*(\tilde{A} + s\tilde{B})\tilde{V}\right)^{-1}\tilde{V}^*\tilde{b} \tag{19}$$

We assume that $n \ll N$, so equation (18) is much easier to solve than equation (14). In the simplest case, $s_j = \mathrm{i}\omega_j$, for $\omega_j \in [\omega_{\min}, \omega_{\max}]$. In this case $\omega_1, \ldots, \omega_n$ may be called interpolating frequencies, and $s_1, \ldots, s_n$ may be called interpolating shifts as the following theorem holds:

Theorem 2.1. *If the matrix $\tilde{V}$ satisfying (15) is full rank, then for $s \in \mathbb{C}$, the solution to (18) exists and is unique. Moreover,*

$$\tilde{h}_{\tilde{V}}(s_j) = \tilde{h}(s_j), \quad j = 1, \ldots, n \tag{20}$$

Proof. First we show that the matrix in equation (18) is not singular. Indeed

$$\tilde{V}^*(\tilde{A} + s\tilde{B})\tilde{V} = \tilde{V}^*\tilde{A}\tilde{V} + s\tilde{V}^*\tilde{B}\tilde{V} = A_1 + sB_1$$

As $\tilde{V}$ is full rank and $\tilde{A}$ is symmetric nonnegative definite, A_1 is hermitian, nonnegative definite. Similarly as $\tilde{B}$ is symmetric positive definite, B_1 is hermitian positive definite. Thus the matrix may be written as:

$$A_1 + sB_1 = B_1^{\frac{1}{2}} \left(B_1^{-\frac{1}{2}} A_1 B_1^{-\frac{1}{2}} + sI \right) B_1^{\frac{1}{2}}$$

with $B_1^{1/2}$ symmetric, invertible. Matrix $B_1^{-1/2} A_1 B_1^{-1/2}$ is hermitian, nonnegative definite, so it has real, nonnegative eigenvalues $a_1, \ldots, a_n$. As a result, the eigenvalues of the matrix $(B_1^{-1/2} A_1 B_1^{-1/2} + sI)$ are $a_1 + s, \ldots, a_n + s$. With the assumption (17), that $s \notin (-\infty, 0]$, none of the eigenvalues $a_i + s$ can be equal to zero, thus matrix $(B_1^{-1/2} A_1 B_1^{-1/2} + sI)$ is invertible and so is $A_1 + sB_1$ as a product of invertible matrices. We have proven that (18) has a unique solution.

Next notice that because of (15), for each j, there is β_j such that $\tilde{V}\beta_j = (\tilde{A} + s_j\tilde{B})^{-1}\tilde{b}$. Thus β_j satisfies (18) for $s = s_j$. This implies

$$\tilde{h}_{\tilde{V}}(s_j) = \tilde{V}\beta_j = (\tilde{A} + s_j\tilde{B})^{-1}\tilde{b} = \tilde{h}(s_j) \qquad \square \tag{21}$$

2.3. A relationship with $(A + sI)^{-1}b$

Let us relate our problem to the situation when $\tilde{B}$ is an identity matrix I. If we define

$$A = \tilde{B}^{-\frac{1}{2}} \tilde{A} \tilde{B}^{-\frac{1}{2}}, \quad b = \tilde{B}^{-\frac{1}{2}} \tilde{b}, \quad h(s) = (A + sI)^{-1} b \tag{22}$$

we can rewrite $\tilde{h}(s)$ as

$$\tilde{h}(s) = (\tilde{A} + s\tilde{B})^{-1}\tilde{b} = \tilde{B}^{-\frac{1}{2}}(A + sI)^{-1}b = \tilde{B}^{-\frac{1}{2}} h(s) \tag{23}$$

Moreover, if we define the matrix V as $V = \tilde{B}^{1/2}\tilde{V}$, then

$$\operatorname{colsp}(V) = \operatorname{span}\left\{ (A + s_1 I)^{-1} b, (A + s_2 I)^{-1} b, \ldots, (A + s_n I)^{-1} b \right\} \tag{24}$$

Notice that a particular form of $\tilde{V}$ does not matter, as long as (15) is satisfied. So to make the presentation easier, we will assume that the columns of V are orthonormal. With this assumption we obtain

$$\tilde{V}^* \tilde{A} \tilde{V} = V^* A V, \quad \tilde{V}^* \tilde{B} \tilde{V} = V^* V = I \tag{25}$$

and

$$\tilde{h}_{\tilde{V}}(s) = \tilde{V} \left(\tilde{V}^* (\tilde{A} + s\tilde{B}) \tilde{V} \right)^{-1} \tilde{V}^* \tilde{b} = \tilde{B}^{-\frac{1}{2}} V \left(V^* (A + sI) V \right)^{-1} V^* b = \tilde{B}^{-\frac{1}{2}} h_V(s) \tag{26}$$

Combining (23) and (26) together allows us to relate the error of the approximation of $\tilde{h}(s)$ to the error of the approximation of $h(s)$:

$$\tilde{h}(s) - \tilde{h}_{\tilde{V}}(s) = \tilde{B}^{-\frac{1}{2}} (h(s) - h_V(s)) \tag{27}$$

Consider a diagonalization of A:

$$A = U \Lambda U^* \tag{28}$$

where Λ is a diagonal matrix with eigenvalues λ_k as entries and the columns of U are eigenvectors u_k. With this notation, each element of vector $h(s)$ is a rational function of the shift s:

$$h(s) = (A + sI)^{-1} b = \sum_{k=1}^{N} \frac{u_k (u_k^* b)}{\lambda_k + s} \tag{29}$$

This rational vector function is approximated by $h_V(s)$:

$$h_V(s) = V \left(V^* A V + sI \right)^{-1} V^* b = \sum_{j=1}^{n} \frac{V \gamma_j (\gamma_j^* V^* b)}{\hat{\lambda}_j + s} \tag{30}$$

where $\hat{\lambda}_j$ and γ_j are eigenvalues and eigenvectors of $V^* A V$ (with the assumption that $V^* V = I$). Notice that h_V is analytic, but not entire. It has many poles. For such a function, the Laurent series may exist outside the Taylor series circle of convergence. As a result, the approximation of such a function by a rational function could be more accurate than a polynomial approximation. Extensive theory of the rational Pade approximation of an analytic function may be found in [3].

Intuitively the approximation (30) will be good if $\hat{\lambda}_j$, $V\gamma_j$, $j = 1, \ldots, n$ approximate λ_k, u_k, $k = 1, \ldots, N$ for k such that $u_k^* b \neq 0$. Yet the quality of this approximation depends on the choice of V, which in turn depends on the choice of the interpolating shifts $(s_j)_{j=1}^n$. We will discuss algorithms that are able to choose $(s_j)_{j=1}^n$ to adapt to the part of the spectrum of A for which $u_k^* b \neq 0$.

2.4. Analysis

In this section we present some mathematical theory related to the rational approximation of the transfer function h_V using rational Krylov subspaces.

Since for any vector z and any s_1, s_2,

$$A(A+s_1I)^{-1}(A+s_2I)^{-1}z + s_1(A+s_1I)^{-1}(A+s_2I)^{-1}z = (A+s_2I)^{-1}z \tag{31}$$

using assumption (16), it can be shown that the space (24) may be written as a Krylov subspace [22]:

$$\operatorname{colsp}(V) = \operatorname{span}\left\{q, Aq, \ldots, A^{n-1}q\right\}, \quad q = \left(\prod_{j=1}^{n}(A+s_jI)^{-1}\right)b \tag{32}$$

Define a space $\mathbf{U}(b)$ as:

$$\mathbf{U}(b) = \operatorname{span}\{u_k : u_k^* b \neq 0\} \tag{33}$$

Assume that the basis $(u_k)_{k=1}^N$ is chosen in such a way that the eigenvalues corresponding to vectors u_k in (33) are distinct. If for a particular eigenvalue λ the dimension of the subspace $\operatorname{span}\{u_k : \lambda_k = \lambda\}$ is more than 1, one can rotate the eigenvectors forming a basis of this subspace, so that only one of them (normalized projection of b onto this subspace) is not orthogonal to b.

With this notation we can formulate and prove the following theorem:

Theorem 2.2. *If (16) is satisfied and the dimension of $\mathbf{U}(b)$ is greater than or equal to n, then V is full rank.*

Proof. Assume that V is not full rank and take a linear combination of columns of V and assume that it is 0. Using the representation (32), there is a polynomial p of the order $\leq n-1$ such that

$$0 = p(A)q$$

Using diagonalization (28) and the definition of q in (32), we obtain:

$$0 = p(A)q = Up(\Lambda)U^*q = Up(\Lambda)\left(\prod_{j=1}^{n}(\Lambda+s_jI)^{-1}\right)U^*b$$

From the latter, as the columns of U are linearly independent, we can conclude that for each $k = 1, \ldots, N$ we have

$$0 = \frac{p(\lambda_k)}{\prod_{j=1}^{n}(\lambda_k+s_j)}u_k^*b$$

so for each $k = 1, \ldots, N$

$$0 = \frac{p(\lambda_k)}{\prod_{j=1}^{n}(\lambda_k+s_j)} \quad \text{or} \quad 0 = u_k^*b$$

Given the assumption, the number of distinct λ_k such that $u_k^*b \neq 0$ is greater than or equal to n, we obtain that the polynomial p has at least n distinct roots. As p is of order not greater than $n-1$, it has to be identically zero:

$$p \equiv 0$$

This proves that V is full rank. □

Notice that the assumption in Theorem 2.2 is true in most practical applications. If the number of distinct eigenvalues λ_k is greater than or equal to n, the assumption not satisfied is equivalent for b to be in one of a finite number of subspaces of $\mathbb{C}^N$. Each of the subspaces has a Lebesgue measure 0, so if b is chosen randomly with the probability distribution absolutely continuous with respect to the Lebesgue measure, the probability of the assumption not being satisfied is 0.

Moreover the following theorem is true.

Theorem 2.3. *If $s_1, \ldots, s_n$ satisfy (16) and the space $\mathbf{U}(b)$ defined in (33) has dimension n, then the approximation $h_V(s)$ to $h(s)$ is exact for any s.*

Remark. Notice that with the assumptions, given in Theorem 2.2, V is full rank and thus the approximation $h_V(s)$ is defined. Adding any additional vector will make the assumptions of Theorem 2.2 fail and thus will make the corresponding V rank deficient. Yet at this point, we do not need to add any vectors as the approximation is already exact. Thus failure to satisfy the assumption of Theorem 2.2 is a lucky failure.

Proof. Given (29), for any s, we have $h(s) \in \mathbf{U}(b)$, so using (24), the definition of V, we have $\mathrm{colsp}(V) \subset \mathbf{U}(b)$. Moreover,

$$\dim(\mathrm{colsp}(V)) = \mathrm{rank}(V) = n = \dim(\mathbf{U}(b))$$

thus

$$\mathrm{colsp}(V) = \mathbf{U}(b)$$

This implies that for any s, $h(s) \in \mathrm{colsp}(V)$, so there exists β_s such that $h(s) = V\beta_s$ given uniqueness of solution to (18) (see Theorem 2.1). Hence

$$h_V(s) = h(s) \qquad \square$$

Next, we will follow [22] to present an interesting interpretation of the approximation $h_V(s)$ to $h(s)$, which allows us to derive the error of the approximation.

Define a space of rational functions:

$$\mathcal{V} = \left\{ \frac{p(\lambda)}{\prod_{j=1}^{n}(\lambda + s_j)} \ : \ p \text{ is a polynomial of the order } \leq n-1 \right\} \tag{34}$$

For any diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ and for $g \in \mathcal{V}$ we will write

$$g(\Lambda) = \mathrm{diag}(g(\lambda_1), \ldots, g(\lambda_N)) \tag{35}$$

One could expand the domain of g to any hermitian matrix A.

Notice, that every vector $x \in \mathrm{colsp}(V)$, using (32), may be written as

$$x = Ug(\Lambda)U^*b, \quad g \in \mathcal{V} \tag{36}$$

One may write an equation for h_V as:

$$(V\alpha)^*(A + sI)h_V(s) = (V\alpha)^*b \tag{37}$$

for all $\alpha \in \mathbb{R}^n$. Since $h_V(s) \in \mathrm{colsp}(V)$, from (36) one can conclude that there is $f \in \mathcal{V}$ such that

$$Uf(\Lambda)U^*b = h_V(s) = V\left(V^*(A + sI)V\right)^{-1}V^*b$$

Similarly each $V\alpha \in \mathrm{colsp}(V)$, so it may be represented as $V\alpha = Ug(\Lambda)U^*b$ for some $g \in \mathcal{V}$. Using those representations, one can rewrite (37) as

$$(Ug(\Lambda)U^*b)^*(A + sI)(Uf(\Lambda)U^*b) = (Ug(\Lambda)U^*b)^*b$$

for all $g \in \mathcal{V}$. The latter may be rewritten as

$$b^*U\overline{g(\Lambda)}(\Lambda + sI)f(\Lambda)U^*b = b^*U\overline{g(\Lambda)}U^*b$$
$$\sum_{k=1}^{N}\overline{g(\lambda_k)}(\lambda_k + s)f(\lambda_k)|u_k^*b|^2 = \sum_{k=1}^{N}\overline{g(\lambda_k)}|u_k^*b|^2$$

The last equation may be presented as

$$\langle(\lambda + s)f - 1, g\rangle_\mu = 0 \quad \forall g \in \mathcal{V} \tag{38}$$

where the measure μ is defined on the spectrum of A (which is a subset of $[0, \infty)$) as a linear combination of delta measures:

$$\mu = \sum_{k=1}^{N}|u_k^*b|^2\delta_{(\lambda=\lambda_k)} \tag{39}$$

Notice that for each value of shift s, there is a different $f \in \mathcal{V}$. We will indicate this dependence by writing $f(\lambda, s)$.

Theorem 2.4. *If the dimension of* $\mathbf{U}(b)$ *is greater than or equal to* n*, then* $f \in \mathcal{V}$ *satisfying* (38) *approximates* $\frac{1}{\lambda+s}$ *with a relative error*

$$\frac{f(\lambda, s) - \frac{1}{\lambda+s}}{\frac{1}{\lambda+s}} = (\lambda + s)f(\lambda, s) - 1 = -\prod_{j=1}^{n}\frac{(s - s_j)(\lambda - \hat{\lambda}_j)}{(\lambda + s_j)(s + \hat{\lambda}_j)} \tag{40}$$

where $\hat{\lambda}_j$, $j = 1, \ldots, n$ *are eigenvalues of* V^*AV*, where* V *is chosen in such a way that* $V^*V = I$.

The proof, different from the one in [22], is given in Appendix A.

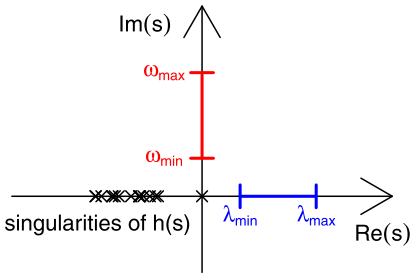

Fig. 3. Complex plane, which is a domain of h, is shown. The interval over which the values of h are needed is $\{i\omega : \omega \in [\omega_{\min}, \omega_{\max}]\}$ (shown in red). The interval of the effective spectrum of A is $\{\lambda_{\min}, \lambda_{\max}\}$ (shown in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. Numerical algorithms

3.1. Part of the norm of the residual as an error indicator

In this section, we present the algorithms for choosing the interpolating shifts s_j. We will consider the values of the shifts in an interval I in the complex plane. I may be a purely imaginary interval $I = \{s = i\omega : \omega \in [\omega_{\min}, \omega_{\max}]\}$, or a purely real interval $I = [\lambda_{\min}, \lambda_{\max}]$. Both intervals are presented in Fig. 3. Considering that we need to evaluate the transfer function in the purely imaginary interval, the best choice of shifts would be such that minimizes the maximum relative error of approximation:

$$\min_{s_1,\ldots,s_n \in I} \max_{\omega \in [\omega_{\min}, \omega_{\max}]} \frac{\|\tilde{h}(i\omega) - \tilde{h}_{\tilde{V}}(i\omega)\|_2}{\|\tilde{h}(i\omega)\|_2} \tag{41}$$

Notice that the choice of $s_1, \ldots, s_n$ determines $\tilde{V}$, which determines the approximation $\tilde{h}_{\tilde{V}}$. Such an approach is not pursued in this paper for two reasons. Firstly because the true relative error of approximation is not known, so we have to use an error indicator $e_{(s_1,\ldots,s_n)}(i\omega)$. Secondly, if (41) was used, the n optimal shifts would most likely not appear among the $n+1$ optimal shifts.

Because of that we consider an approach in which at each iteration, given $(s_1, \ldots, s_n)$, we will add one value s_{n+1} in order to form $(s_1, \ldots, s_n, s_{n+1})$. We use the fact that at each interpolation shift s_j, the interpolation is exact, so the error of approximation is 0, thus the value of the next interpolation shift s_{n+1} is chosen as the maximum of the error indicator function in the interval I:

$$s_{n+1} = \operatorname{argmax}_{s \in I} e_{(s_1,\ldots,s_n)}(s) \tag{42}$$

In such a way, we set the error to 0 at the location that needs that the most, the location at which the error is the largest.

The error indicator that we use has been introduced in [7]. Using Theorem 2.4 one can show that it is equivalent to an arbitrary seminorm of the residual. This statement is expressed in the following theorem.

Theorem 3.1. *Let $\|.\|$ denote an **arbitrary seminorm**. The seminorm of the residual as a function of s may be expressed as*

$$\|(A + sI)h_V(s) - b\|(s) = \left| \prod_{j=1}^{n} \frac{s - s_j}{s + \hat{\lambda}_j} \right| C$$

where C is a constant dependent on the seminorm.

Proof. Using the relative error (40) one can rewrite the residual as:

$$\begin{aligned} R &= (A + sI)h_V(s) - b = (A + sI)(h_V(s) - h(s)) = U((\Lambda + sI)f(\Lambda, s) - I)U^*b \\ &= -\left[\prod_{j=1}^{n} \tfrac{s-s_j}{s+\hat{\lambda}_j}\right] \left[U\left(\prod_{j=1}^{n}(\Lambda - \hat{\lambda}_j I)\prod_{j=1}^{n}(\Lambda + s_j I)^{-1}\right)U^*b\right] = R_s(s)R_V \end{aligned} \tag{43}$$

The quantity above is split into two parts. The first part, R_s, is a scalar, the second, R_V, is a vector. Notice that R_s is dependent on s and R_v is not.

$$\|R\| = \|R_s(s)R_V\| = |R_s(s)|\,\|R_v\| = \left| \prod_{j=1}^{n} \frac{s - s_j}{s + \hat{\lambda}_j} \right| C \tag{44}$$

where $C = \|R_v\|$ does not depend on s. □

Remark.

- Theorem 3.1 justifies the use of the following error indicator

$$e(s) = \left| \prod_{j=1}^{n} \frac{s - s_j}{s + \hat{\lambda}_j} \right| \tag{45}$$

  where $s_1, \ldots, s_n$ are shifts chosen so far and $\hat{\lambda}_1, \ldots, \hat{\lambda}_n$ are eigenvalues of V^*AV with the assumption $V^*V = I$.
- The theorem says that one can use any seminorm of the residual as an error indicator and the result would be the same. One could use the absolute value of a single entry of the residual. If the values are not equal to 0 for all shifts, then the new interpolating shift s_{n+1} would be the same as for (45).
- Authors of [7] noticed that once the interpolation shifts s_j are fixed, the Ritz values $\hat{\lambda}_j$ minimize the norm of R_V:

$$\left\| U \left(\prod_{j=1}^{n} (\Lambda - \hat{\lambda}_j I) \prod_{j=1}^{n} (\Lambda + s_j I)^{-1} \right) U^* b \right\|_2 \tag{46}$$

  For the proof of this statement see Theorem in Appendix B.1. This property allows to conclude that if the above error indicator is used, both the new value of shift and the Ritz values are chosen in an optimal way to reduce the second norm of the residual.

Authors of [7] consider real valued shifts in the interval $[\lambda_{\min}, \lambda_{\max}]$ where $\lambda_{\min}, \lambda_{\max} > 0$ are the smallest and the largest eigenvalues of A. The algorithm of [7] is presented below:

Algorithm ARR (Adaptive choice of Real shifts; the norm of the Residual used as the error indicator):

1. Set $n = 2$, choose $s_1 = \lambda_{\min}$, $s_2 = \lambda_{\max}$ and set $\tilde{V} = \tilde{V}_{1:2}$ in such a way that
$$\operatorname{colsp}(\tilde{V}_{1:2}) = \operatorname{span}\{(\tilde{A} + s_1\tilde{B})^{-1}\tilde{b}, (\tilde{A} + s_2\tilde{B})^{-1}\tilde{b}\}$$
and $\tilde{V}^*\tilde{B}\tilde{V} = I$.
2. Find s_{n+1} as the maximizer of
$$\left| \prod_{j=1}^{n} \frac{s - s_j}{s + \hat{\lambda}_j} \right| \tag{47}$$
over $s \in [\lambda_{\min}, \lambda_{\max}]$, where $\hat{\lambda}_1, \ldots, \hat{\lambda}_n$ are eigenvalues of $\tilde{V}^*\tilde{A}\tilde{V}$.
3. Set $\tilde{V} = \tilde{V}_{1:(n+1)}$ in such a way that
$$\operatorname{colsp}(\tilde{V}_{1:(n+1)}) = \operatorname{colsp}(\tilde{V}_{1:n}) \oplus \operatorname{span}\{(\tilde{A} + s_{n+1}\tilde{B})^{-1}\tilde{b}\}$$
and $\tilde{V}^*\tilde{B}\tilde{V} = I$.
4. If exit criteria met (the approximation is good enough), stop
5. Set $n = n + 1$ and jump to 2

The details of the update of $\tilde{V}$ at step 3 are presented in section 3.2. Notice that according to (25), $\tilde{V}^*\tilde{A}\tilde{V} = V^*AV$, so those two matrices have the same eigenvalues.

This approach is valid for A symmetric positive definite or for b orthogonal to the null space of A (in this case $\lambda_{\min}$ is the smallest of nonzero eigenvalues). The case when matrix A has a significant null space and b is not orthogonal to the null space is analyzed in depth in [24]. In the current paper, in such a case we just set $\lambda_{\min}$ to be the smallest positive eigenvalue. The advantage of this approach is that at each iteration, to enlarge $\tilde{V}$, one needs to solve the linear system (14) for a real s. In this case matrix $\tilde{A} + s\tilde{B}$ is real, symmetric positive definite and also $\tilde{b}$ is real, thus solving the linear system with iterative solvers is faster than in the complex case. One matrix vector multiplication is 4 times faster, and the total number of iterations is less as iterative methods for hermitian matrices (like Conjugate Gradient method) may be used.

Next we propose a similar algorithm, but for purely imaginary values of shifts $s = \mathrm{i}\omega$, where the frequency is in an interval of interest $\omega \in [\omega_{\min}, \omega_{\max}]$. This is a natural approach for MT, as one needs the response for a number of frequencies log-uniformly distributed in a purely imaginary interval. Also if $\tilde{b} \in \mathbb{R}$, $\omega \in \mathbb{R}$ then

$$\overline{\tilde{h}(\mathrm{i}\omega)} = \overline{(\tilde{A} + \mathrm{i}\omega\tilde{B})^{-1}\tilde{b}} = (\tilde{A} - \mathrm{i}\omega\tilde{B})^{-1}\tilde{b} = \tilde{h}(-\mathrm{i}\omega) \tag{48}$$

so if one calculates $\tilde{h}$ at $s = \mathrm{i}\omega_j$, then one simultaneously gets the value of $\tilde{h}$ at $s = -\mathrm{i}\omega_j$. Hence with one interpolating frequency ω_j, the dimension of $\operatorname{colsp}(\tilde{V})$ may be increased by 2, using $\tilde{h}(\mathrm{i}\omega_j)$ and $\overline{\tilde{h}(\mathrm{i}\omega_j)}$. In this setting when at each iteration we add two shifts $s = \pm\mathrm{i}\omega_j$, the squared norm of the first part of the residual (43) evaluated at $s = \mathrm{i}\omega$ is

$$\left|\prod_{j=1}^{2n}\frac{s-s_j}{s+\hat{\lambda}_j}\right|^2=\left|\frac{\prod_{j=1}^{n}(\mathrm{i}\omega-\mathrm{i}\omega_j)(\mathrm{i}\omega+\mathrm{i}\omega_j)}{\prod_{j=1}^{2n}(\mathrm{i}\omega+\hat{\lambda}_j)}\right|^2=\frac{\prod_{j=1}^{n}(\omega^2-\omega_j^2)^2}{\prod_{j=1}^{2n}(\omega^2+\hat{\lambda}_j^2)} \tag{49}$$

where $\hat{\lambda}_j\in\mathbb{R}$, $j=1,\dots,2n$ are the eigenvalues of $V^*AV=\tilde{V}^*\tilde{A}\tilde{V}$. The resulting algorithm is presented below:

Algorithm AIR (Adaptive choice of Imaginary shifts, the norm of the Residual used as the error indicator):

1. Set $n=2$, choose $\omega_1=\omega_{\min}$, $\omega_2=\omega_{\max}$ and set $\tilde{V}=\tilde{V}_{1:4}$ in such a way that
$$\operatorname{colsp}(\tilde{V}_{1:4})=\operatorname{span}\left\{(\tilde{A}+\mathrm{i}\omega\tilde{B})^{-1}\tilde{b}:\omega\in\{\omega_1,-\omega_1,\omega_2,-\omega_2\}\right\}$$
and $\tilde{V}^T\tilde{B}\tilde{V}=I$.
2. Find ω_{n+1} as the maximizer of
$$\frac{\prod_{j=1}^{n}(\omega^2-\omega_j^2)^2}{\prod_{j=1}^{2n}(\omega^2+\hat{\lambda}_j^2)} \tag{50}$$
over $\omega\in[\omega_{\min},\omega_{\max}]$, where $\hat{\lambda}_1,\dots,\hat{\lambda}_{2n}$ are eigenvalues of $\tilde{V}^*\tilde{A}\tilde{V}$.
3. Set $\tilde{V}=\tilde{V}_{1:2(n+1)}$ in such a way that
$$\operatorname{colsp}(\tilde{V}_{1:2(n+1)})=\operatorname{colsp}(\tilde{V}_{1:2n})\oplus\operatorname{span}\{(\tilde{A}+\mathrm{i}\omega_{n+1}\tilde{B})^{-1}\tilde{b},(\tilde{A}-\mathrm{i}\omega_{n+1}\tilde{B})^{-1}\tilde{b}\}$$
and $\tilde{V}^T\tilde{B}\tilde{V}=I$.
4. If exit criteria met (approximation is good enough), stop
5. Set $n=n+1$ and jump to 2

In this algorithm the update 3 may be done in such a way that $\tilde{V}$ is a real matrix. This saves some computational time (see section 3.2).

What seems to be an advantage over ARR algorithm is that one does not need to estimate $\lambda_{\min},\lambda_{\max}$. Moreover, the approximation is tuned for the interval of interest $[\omega_{\min},\omega_{\max}]$.

3.2. The update of matrix $\tilde{V}$

Let v_i denote the columns of $\tilde{V}$. We update V in such a way that the columns of $\tilde{V}$ are orthonormal. If we need to add a vector u to the column space $\tilde{V}$, we orthogonalize it according to the algorithm:

$$\begin{aligned}&v_{n+1}=u\\&\text{for }(j=1:n)\{\\&\quad v_{n+1}=v_{n+1}-(v_{n+1},v_j)v_j\\&\}\\&v_{n+1}=\tfrac{v_{n+1}}{\|v_{n+1}\|_2}\end{aligned} \tag{51}$$

In the case of ARR algorithm, the vector to be added is

$$u=\tilde{h}(s)=(\tilde{A}+s\tilde{B})^{-1}\tilde{b}$$

for some real s. As a result $u\in\mathbb{R}^N$ and the matrix V is real.

In the case of AIR algorithm when $\tilde{b}\in\mathbb{R}$, for some ω, instead of adding $\tilde{h}(\mathrm{i}\omega)$ and its complex conjugate $\overline{\tilde{h}(\mathrm{i}\omega)}=\tilde{h}(-\mathrm{i}\omega)$, we add its real and imaginary parts $\operatorname{Re}(\tilde{h}(\mathrm{i}\omega))$, $\operatorname{Im}(\tilde{h}(\mathrm{i}\omega))$. The latter two vectors span the same two dimensional space as $\tilde{h}(\mathrm{i}\omega)$, $\overline{\tilde{h}(\mathrm{i}\omega)}$, yet they are real valued and as a result, the matrix V is real valued. This allows for computational savings whenever columns of $\tilde{V}$ have to be multiplied by matrices $\tilde{A},\tilde{B}$ or when inner products have to be calculated.

3.3. A note on numerical precision

To use the error indicator (45) one has to calculate the entries of the matrix $\tilde{V}^*\tilde{A}\tilde{V}$ and then calculate its eigenvalues. We tested the numerical accuracy by comparing the values of (45) to relative residual norm and for all considered calculations the accuracy was sufficient. To visualize our test, we present a comparison of the error indicator and the relative residual norm for AIR algorithm at iteration $n=23$ in Fig. 4.

In order to be able to get a good approximation across a range of frequencies, one needs to use a good solver to compute an accurate value of $\tilde{h}(\mathrm{i}\omega)$ for a particular frequency. The solver needs to work well also for low frequencies. In our case,

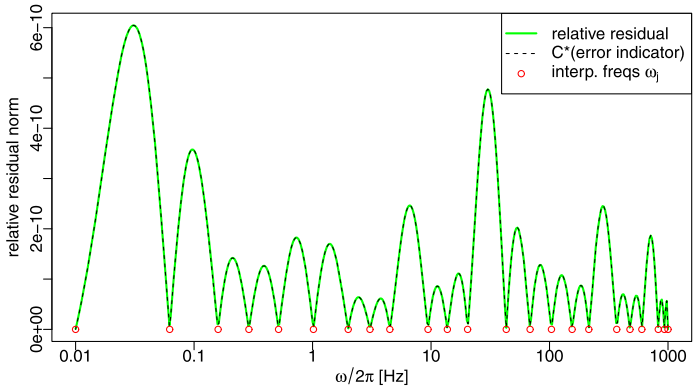

Fig. 4. The relative residual norm calculated at the iteration $n = 23$ of algorithm AIR, compared with the error indicator 45 multiplied by a constant chosen in such a way that the two match. We present the case of $\tilde{b}$ for the Jacobian of H_x evaluated at a receiver position described in section 4.

we use a direct solver for a linear system and we use a carefully implemented divergence correction (see [25]). Even though the direct solver, together with the divergence correction is used, for low frequencies it didn't allow us to obtain a solution of the linear system with the relative residual less than 10^{-10}, which is larger than the machine precision. What is more encouraging, though, is the fact that even though a parallelized solver is used (which introduced some randomness in the obtained solution due to random calculation order), for all frequencies the solution obtained in two different runs of the code differ by a relative error almost as small as the machine precision. The latter would not be true if the divergence correction was not used.

In order to evaluate the approximation $\tilde{h}_{\tilde{V}}(s)$ for a given shift value s, one needs to evaluate:

$$\tilde{h}_{\tilde{V}}(s) = \tilde{V}(\tilde{V}^*\tilde{A}\tilde{V} + s\tilde{V}^*\tilde{B}\tilde{V})^{-1}\tilde{V}^*\tilde{b} \tag{52}$$

In order to make the columns of matrix $\tilde{V}$ $\tilde{B}$-orthogonal (see section 3.2), due to insufficient numerical precision, multiple reorthogonalizations are needed to allow replacing matrix $\tilde{V}^*\tilde{B}\tilde{V}$ in (52) with the identity matrix. Alternatively one could evaluate matrix $\tilde{V}^*\tilde{B}\tilde{V}$ and use it. This allows to obtain a better approximation $\tilde{h}_{\tilde{V}}(s)$ of $\tilde{h}(s)$.

Moreover, this approximation will fail to reduce the approximation error below some level. If a better approximation is needed, for some of the calculations one can use the quadruple precision, which is only 4 times slower than the double precision on modern machines [20] or the double–double precision [21] if quadruple precision on the machine used is slow. In our implementation, higher precision is used to evaluate inner products between columns of $\tilde{V}$ and columns of $\tilde{A}\tilde{V}$ to obtain $\tilde{V}^*(\tilde{A}\tilde{V})$. In this case the inner product is evaluated in such a way that the input vectors are in double precision and only the calculation is done in the higher precision, thus only the CPU calculations are slower, reading from RAM memory is not. An example code in Fortran showing precisely what we mean is presented in Appendix C. A similar approach is used to evaluate $\tilde{V}^*\tilde{B}\tilde{V}$. Once those matrices are evaluated, they are stored in higher precision and added to form matrix $\tilde{V}^*\tilde{A}\tilde{V} + \mathrm{i}\omega\tilde{V}^*\tilde{B}\tilde{V}$, which is a very small matrix with the number of columns and rows not exceeding 100 in our numerical examples. The application of the inverse of this matrix to $\tilde{V}^*\tilde{b}$ is also performed in the higher precision. The resulting vector is converted to double precision and multiplied by the matrix $\tilde{V}$ to obtain $\tilde{h}_{\tilde{V}}(s)$. As we show in the numerical experiments, this allows to reduce the level of stagnation of AIR algorithm by two orders of magnitude. It does not affect the stagnation level of ARR. And as only some of the calculations are done in higher precision, for low rank of $\tilde{V}$, the additional computational time will be insignificant compared with the time of solution of the linear system for a single value of shift.

4. Numerical results

Although the proposed methods are designed primarily for iterative solvers, in order to test the algorithms, we use the forward solver of [25]. We consider a model with a 3d hill and a 3d valley, presented in Fig. 5. Below the surface there is a conductive (1 Ω m) object in 50 Ω m background. The conductive object is placed at $[-700\text{ m}, 700\text{ m}] \times [-328.3\text{ m}, 328.3\text{ m}]$ in XY plane and extends from on average 450 m to 1153 m depth. The air is approximated by 10^7 Ω m and the term $\mathrm{i}\omega\epsilon$ is dropped completely in all of the domain. In the numerical tests we consider magnetotelluric response Z, K at one receiver location at the bottom of the valley, marked by a blue dot in Fig. 5. We are interested in the response $\tilde{h}(\mathrm{i}\omega)$ for a range of frequencies considered in magnetotellurics $\frac{\omega}{2\pi} \in [0.01\text{ Hz}, 1000\text{ Hz}]$. YZ cross-section plotted in Fig. 5, shows the location of the object. The hexahedral mesh consists of 31, 31 and 25 elements in x, y and z directions respectively and it extends to 45 km from the center in x and y directions. In z direction the mesh extends to 32 km above the surface and 47 km deep. The number of columns (and rows) of the system matrix is equal to 67,140.

To visualize the iterations of the algorithms, in Fig. 6 we show the relative error of approximation and the error indicator used as a function of frequency ω for three consecutive iterations of the AIR algorithm. This figure presents how the

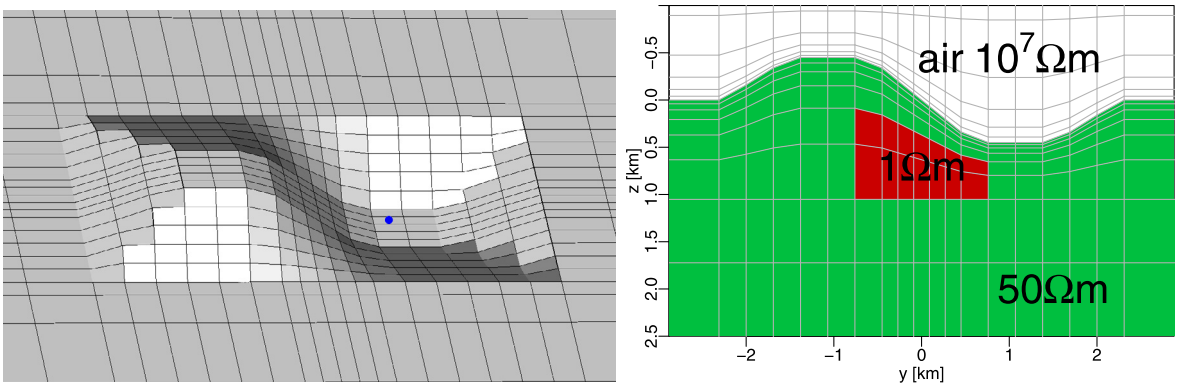

Fig. 5. The central part of the surface mesh, together with the location of the receiver shown by a blue dot (left); The central part of the YZ cross-section at $x = 0$ (right). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

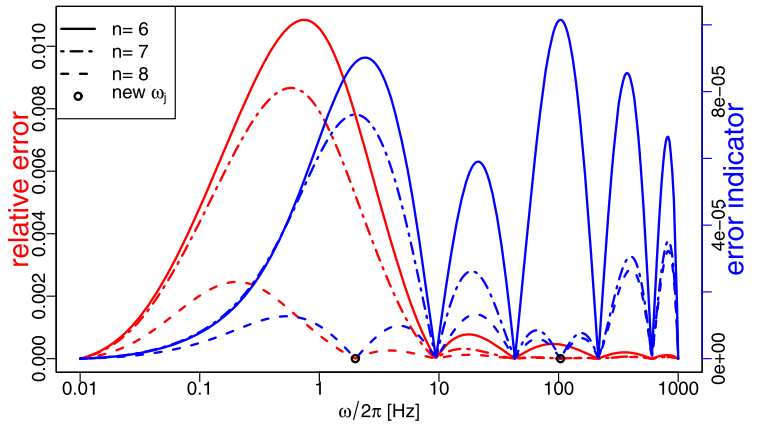

Fig. 6. Three consecutive iterations of AIR algorithm. We present the case of b related to the Jacobian of H_x. The error indicator is shown in blue, the true relative error is shown in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

new interpolating frequency ω_j is chosen at a location of the maximum of the error indicator and also how the error of approximation is reduced in consecutive iterations. It also shows that the error indicator we use tends to focus more on high frequencies.

Recollect that $\tilde{b} = v$ for v defined in (9) and (10) for the electric field E and the magnetic field H respectively. We consider v for E corresponding to x and y directions and H corresponding to x, y and z directions.

To compare the algorithms we calculate the maximum relative error of an approximation

$$\max_{\omega\in[\omega_{\min},\omega_{\max}]} \frac{\|\tilde{h}_{\tilde{V}}(\mathrm{i}\omega) - \tilde{h}(\mathrm{i}\omega)\|_2}{\|\tilde{h}(\mathrm{i}\omega)\|_2} \tag{53}$$

We also consider the maximum relative residual norm:

$$\max_{\omega\in[\omega_{\min},\omega_{\max}]} \frac{\|(\tilde{A} + \mathrm{i}\omega\tilde{B})\tilde{h}_{\tilde{V}}(\mathrm{i}\omega) - \tilde{b}\|_2}{\|\tilde{b}\|_2} \tag{54}$$

First, consider the case of the magnetic field H. In Fig. 7, we compare AIR and ARR algorithms for two cases of the rhs $\tilde{b}$ corresponding to the Jacobian of H_x and H_z. One can see that with the same dimension of the Krylov subspace AIR algorithm approximation is better than the approximation using ARR algorithm. Moreover, ARR algorithm stagnates once it reaches the level of the maximum relative error of 10^{-8}, whereas AIR algorithm allows to reduce the maximum relative error to a level two orders of magnitude less. Further, if some of the calculations of AIR algorithm (see section 3.3) are done in quadruple precision, one could reduce the max relative error another two orders of magnitude. The stagnation of the ARR algorithm might be explained by a need to do an analytic continuation of the values of a function on purely real interval of shifts, to a purely imaginary interval of shifts (see Remark 5 in [10]). This is confirmed by our experience that even if some of the calculations of ARR algorithm are done in quadruple precision the stagnation occurs at the same level.

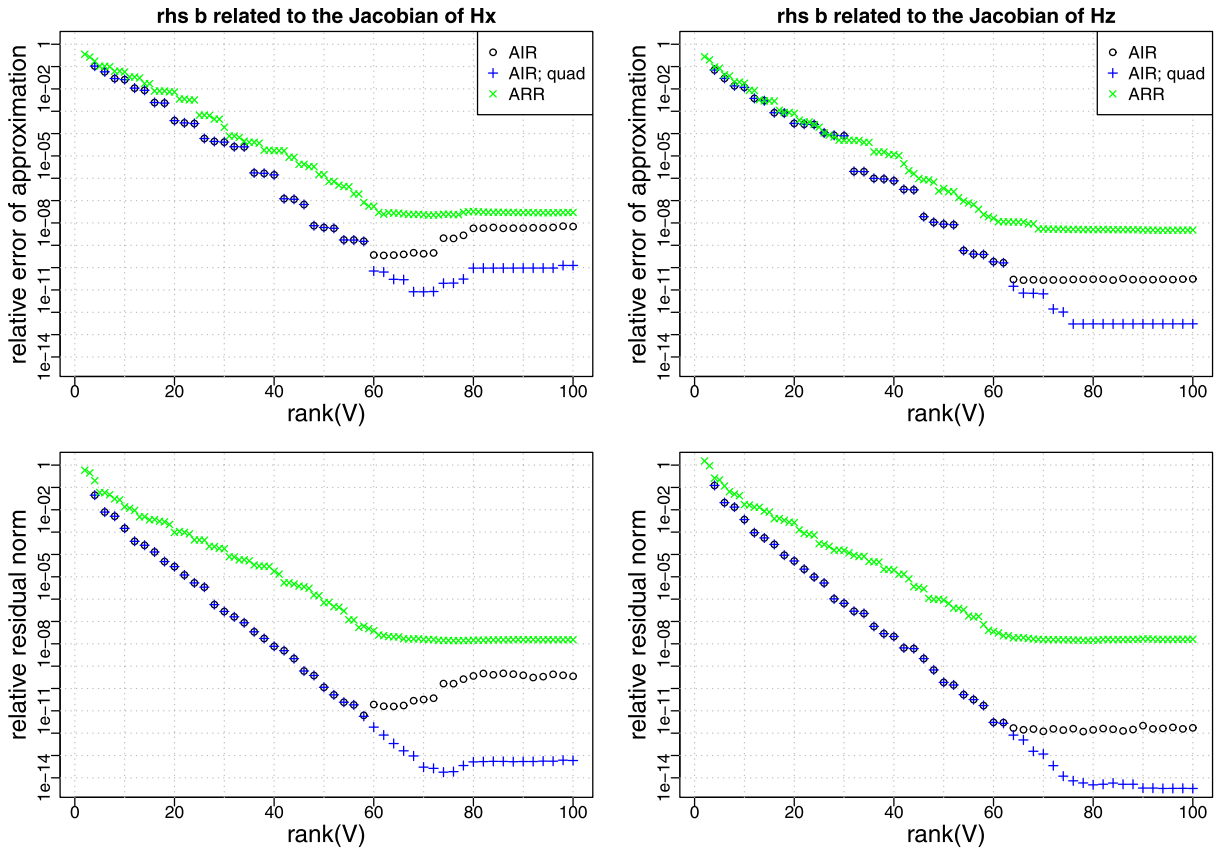

Fig. 7. The relative error of approximation and the relative residual norm as a function of the dimension of the Krylov subspace for AIR and ARR algorithms. Notice each iteration of AIR increases the dimension by two and each iteration of ARR increases the dimension by 1. "AIR; quad" stands for AIR algorithm, for which some of the calculations are done in quadruple precision. Two cases are presented: b related to the Jacobian of H_x and H_z.

On the horizontal axis of the figure, we show the dimension of the Krylov subspace used to approximate the solution. For ARR algorithm it is equal to the iteration number n, yet for AIR algorithm it is two times larger than n. One iteration of each of the algorithms requires a single solution of the linear system, yet in the case of ARR everything is real valued, so the solution time is at least 4 times less. To account for that in Fig. 8 we present a similar plot, but with the horizontal axis representing the computational time. Such a comparison shows that before ARR algorithm stagnates, it is able to reduce the approximation error about 1.5 faster than AIR algorithm. Another advantage of ARR is that the values of the max relative residual norm are close to the values of the max relative error, which allows to use the max relative residual norm as a good guess of the max relative error. For AIR the max relative residual norm is usually less then the max relative error. The tests for b related to the Jacobian of the electric field show similar performance of the considered algorithms.

A significant and sometimes dominant computational cost at each iteration of Gauss–Newton method is due to the calculation of the Jacobian. If the same mesh is used for all receivers, one needs two solves to obtain two vectors ξ (one for each plane wave polarization) that are needed in the forward problem calculation (11) and subsequently as much as 5 times the number of receivers solves to obtain the Jacobian in (12). With 100 MT receivers in the domain, one needs 500 solves to calculate the Jacobian and only 2 solves for the forward response.

To estimate the speedup of the calculation of the Jacobian, we assume that an iterative solver used to solve the linear systems has a property that the speed of reducing the log of the relative residual norm does not depend on the value of the relative residual norm, i.e. the same number of iterations is needed to reduce the relative residual norm from 10^{-1} to 10^{-2} as is needed to reduce the relative residual norm from 10^{-5} to 10^{-6}. It is true for solvers that use multigrid techniques, divergence correction, and if the mesh does not have high aspect ratio elements [29,23]. Suppose further that the approximation $\tilde{h}_{\tilde{V}}(\mathrm{i}\omega)$ is used in place of $\tilde{h}(\mathrm{i}\omega)$ if it has sufficiently small residual and if not, it is used as a starting guess for the iterative solver. Assume that the cost of applying the model reduction techniques is negligible compared to the cost of calculating $\tilde{h}(s) = (\tilde{A} + s\tilde{B})^{-1}\tilde{b}$ for one s using an iterative solver. Our numerical tests show that the speedup of using AIR versus not using the model order reduction is 4 times in the case of approximations of $\tilde{h}(\mathrm{i}\omega)$ at 30 frequencies with the relative residual norm less than 10^{-7}. The speedup gets higher if values of $\tilde{h}(\mathrm{i}\omega)$ for more frequencies are needed. For example for 60 frequencies, the speedup would be 8 times.

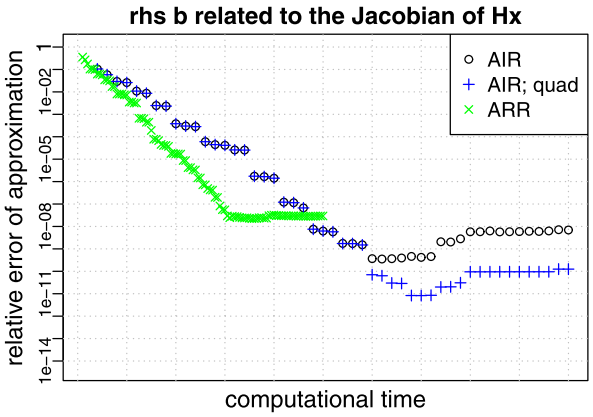

Fig. 8. A plot similar to Fig. 7, but an estimate of the computational time is presented on the horizontal axis. The estimate takes into consideration the fact that the calculations for real numbers are 4 times faster than for complex numbers. Only the max relative error of approximation is plotted for the case of b related to the Jacobian of $H - x$.

5. Conclusions

For many inverse problems solved using a method (like Gauss–Newton) requiring a Jacobian, significant and sometimes dominant computational cost is related to its calculation. We developed a method to significantly speed up the calculations of the Jacobian using model order reduction interpolation based on Pade approximation and rational Krylov subspaces.

Acknowledgements

We acknowledge the support of this work from the U.S. Dept. of Energy under contract DE-EE0002750 to PW. EC acknowledges the partial support of the U.S. National Science Foundation through grant DMS-1413454.

Appendix A. Proof of Theorem 2.4

Proof. Let $\hat{\lambda}_j$, γ_j be the eigenvalues and the eigenvectors of V^*AV. As V^*AV is hermitian, nonnegative definite, the eigenvalues are nonnegative:

$$\hat{\lambda}_j \geq 0, \quad j = 1, \ldots, n \tag{A.1}$$

and γ_j might be chosen to be orthonormal. Using representation (36), define $z_j \in \mathcal{V}$ such that

$$V\gamma_j = Uz_j(\Lambda)U^*b \tag{A.2}$$

Consider an operator of multiplication by λ acting on the linear space $\mathcal{V}$. We show that z_j are μ orthonormal eigenvectors of this operator:

$$\delta(i=j) = \gamma_i^*\gamma_j = \gamma_i^*V^*V\gamma_j = (V\gamma_i)^*(V\gamma_j) = (Uz_i(\Lambda)U^*b)^*Uz_j(\Lambda)U^*b = \langle z_j, z_i\rangle_\mu \tag{A.3}$$

$$\hat{\lambda}_j\langle z_j, z_i\rangle_\mu = \hat{\lambda}_j\gamma_i^*\gamma_j = \gamma_i^*(\hat{\lambda}_j\gamma_j) = \gamma_i^*(V^*AV\gamma_j) = (V\gamma_i)^*A(V\gamma_j) = \langle \lambda z_j, z_i\rangle_\mu \tag{A.4}$$

Using the latter and (38) one can calculate the coefficients of f in eigendirections z_j to obtain:

$$f(\lambda, s) = \sum_{j=1}^{n} \frac{\langle 1, z_j\rangle_\mu}{\hat{\lambda}_j + s} z_j(\lambda) \tag{A.5}$$

Using the definition (34) of $\mathcal{V}$, we introduce p_j as a polynomial of the order at most $n-1$ such that

$$z_j(\lambda) = \frac{p_j(\lambda)}{\prod_{i=1}^{n}(\lambda + s_i)} \tag{A.6}$$

This allows us to rewrite (A.5) as

$$f(\lambda, s) = \sum_{j=1}^{n} \left(\frac{\langle 1, z_j\rangle_\mu}{\hat{\lambda}_j + s} \frac{p_j(\lambda)}{\prod_{i=1}^{n}(\lambda + s_i)} \right) = \frac{p(\lambda, s)}{\prod_{j=1}^{n}(\hat{\lambda}_j + s)(\lambda + s_j)} \tag{A.7}$$

for some p, which is a polynomial of the order at most $n-1$ with respect to λ as well as with respect to s. Define

$$\check{f}(\lambda,s)=\frac{1}{\lambda+s}\left[1-\prod_{j=1}^{n}\frac{(s-s_j)(\lambda-\hat{\lambda}_j)}{(\lambda+s_j)(s+\hat{\lambda}_j)}\right]=\frac{\prod_{j=1}^{n}(\lambda+s_j)(s+\hat{\lambda}_j)-\prod_{j=1}^{n}(s-s_j)(\lambda-\hat{\lambda}_j)}{(\lambda+s)\prod_{j=1}^{n}(\lambda+s_j)(s+\hat{\lambda}_j)}$$

Following [30], we notice that if we plug in $-s$ for λ then the numerator is 0. Thus the polynomial in the numerator is divisible by $(\lambda+s)$. This allows us to write $\check{f}$ as

$$\check{f}(\lambda,s)=\frac{\check{p}(\lambda,s)}{\prod_{j=1}^{n}(\hat{\lambda}_j+s)(\lambda+s_j)}$$

where $\check{p}$ is a polynomial of the order at most $n-1$ of s and of λ. In order to prove Theorem 2.4 it remains to show that $p=\check{p}$.

Define

$$\check{h}_V(s)=U\check{f}(\Lambda,s)U^*b \tag{A.8}$$

Using (20) and (27), and the definition of $\check{f}$, we have

$$\begin{aligned}
h_V(s)&=\check{h}_V(s), && \text{for } s=s_j,\ j=1,\dots,n\\
Uf(\Lambda,s)U^*b&=U\check{f}(\Lambda,s)U^*b, && \text{for } s=s_j,\ j=1,\dots,n\\
f(\Lambda,s)U^*b&=\check{f}(\Lambda,s)U^*b, && \text{for } s=s_j,\ j=1,\dots,n\\
f(\lambda_k,s)u_k^*b&=\check{f}(\lambda_k,s)u_k^*b, && \text{for } s=s_j,\ j=1,\dots,n,\quad k=1,\dots,N
\end{aligned}$$

Let us now fix λ_k. Assuming $u_k^*b\neq 0$, we have

$$\frac{p(\lambda_k,s)}{\prod_{i=1}^{n}(\hat{\lambda}_i+s)(\lambda_k+s_i)}=\frac{\check{p}(\lambda_k,s)}{\prod_{i=1}^{n}(\hat{\lambda}_i+s)(\lambda_k+s_i)},\ \text{for } s=s_j,\ j=1,\dots,n$$

As $\lambda_k,\hat{\lambda}_i\geq 0$, with the assumption (17), we can conclude that

$$\begin{aligned}
p(\lambda_k,s)&=\check{p}(\lambda_k,s), && \text{for } s=s_j,\ j=1,\dots,n\\
p(\lambda_k,s)-\check{p}(\lambda_k,s)&=0, && \text{for } s=s_j,\ j=1,\dots,n
\end{aligned}$$

$p(\lambda_k,s)-\check{p}(\lambda_k,s)$ is a polynomial of the order at most $n-1$. As it has n distinct roots, it has to be equal to 0. We have obtained:

$$u_k^*b\neq 0\quad\Rightarrow\quad\forall s\ \ p(\lambda_k,s)=\check{p}(\lambda_k,s)$$

Now, fix s, then $p(\lambda,s)-\check{p}(\lambda,s)$ is a polynomial of λ of the order at most $n-1$. The assumption that the dimension of $\mathbf{U}(b)$ is greater than or equal to n implies that number of distinct λ_k such that $u_k^*b\neq 0$ is greater than or equal to n, so the polynomial has at least n distinct roots, thus it is equal to 0. We have obtained that

$$p(\lambda,s)=\check{p}(\lambda,s),\quad\forall\lambda,s$$

which implies

$$f(\lambda,s)=\check{f}(\lambda,s),\quad\forall\lambda,s$$

which in turn implies (40). □

Appendix B. Minimizing property of Ritz values

Theorem Appendix B.1. *For any values $s_j\notin(-\infty,0]$, the eigenvalues $\hat{\lambda}_j$ of V^*AV minimize the norm (46).*

Proof. The square of the norm may be rewritten as

$$\begin{aligned}
&\left\|U\left(\prod_{j=1}^{n}(\Lambda-\hat{\lambda}_jI)\prod_{j=1}^{n}(\Lambda+s_jI)^{-1}\right)U^*b\right\|_2^2\\
&=\int_0^\infty\left|\prod_{j=1}^{n}(\lambda-\hat{\lambda}_j)\right|^2\left|\prod_{j=1}^{n}(\lambda-s_j)\right|^{-2}\mu(d\lambda)\\
&=\int_0^\infty\prod_{j=1}^{n}(\lambda-\hat{\lambda}_j)^2d\tau(\lambda)=\|\tilde{\pi}_n\|_\tau^2
\end{aligned}\tag{B.1}$$

where the measure τ is defined as

$$d\tau(\lambda) = \left| \prod_{j=1}^{n} (\lambda - s_j) \right|^{-2} \mu(d\lambda) \tag{B.2}$$

and the polynomial $\tilde{\pi}_n$ is

$$\tilde{\pi}_n(\lambda) = \prod_{j=1}^{n} (\lambda - \hat{\lambda}_j) \tag{B.3}$$

The expression (A.4) for eigenvalues $\hat{\lambda}_j$ and eigenvectors z_j, defined in (A.2) may be written using the measure τ as

$$\langle (\lambda - \hat{\lambda}_j) p_j, p_i \rangle_\tau = 0 \tag{B.4}$$

for all $i, j = 1, \ldots, n$. Polynomial p_j has been defined by (A.6). Consider a polynomial $\pi_n(\lambda)$ of the order n, which is τ orthogonal to all the polynomials of the order at most $n-1$. If the measure τ has support consisting of at least n points (which is true if the number of distinct eigenvalues λ_k for which $u_k^* b \neq 0$, is at least n) then π_n has n distinct zeros. This is a standard result of the theory of orthogonal polynomials, the proof is not given here. According to (B.4), we have

$$\pi_n(\lambda) = C_j (\lambda - \hat{\lambda}_j) p_j \tag{B.5}$$

where C_j is a constant. This shows that $\hat{\lambda}_j$ for $j = 1, \ldots, n$ are roots of π_n. Thus

$$\pi_n(\lambda) = C \prod_{j=1}^{n} (\lambda - \hat{\lambda}_j) = C \tilde{\pi}_n(\lambda) \tag{B.6}$$

for some constant C.

If one varies the values $\hat{\lambda}$, the quantity (B.1) yields $\|p(\lambda)\|_\tau^2$ for some polynomial p of the order n and the coefficient of λ^n equal to 1. Therefore p can be represented as

$$p(\lambda) = \pi_n(\lambda) + r(\lambda) \tag{B.7}$$

where the order of r is at most $n-1$. Because of (B.6), π_n is τ orthogonal to r and thus

$$\|p\|_\tau^2 = \|\pi_n\|_\tau^2 + \|r\|_\tau^2$$

so

$$\|p\|_\tau^2 \geq \|\pi_n\|_\tau^2 \qquad \square$$

Appendix C. Example code for high precision inner product

We present an example code for high precision inner product. Calculations are done in quadruple precision, yet input vectors are in double precision.

```
subroutine HighPrecisionInnerProduct(N,x,y,res)
  implicit none
  integer ,intent(in)  :: N
  real(8) ,intent(in)  :: x(N),y(N)  ! input in double precision
  real(16),intent(out) :: res        ! output in quadruple precision
  ! local variables:
  integer :: i
  res = 0
  do i = 1,N
     ! calculations in quadruple prec.
     res = res + real(y(i),16)*real(x(i),16)
  enddo
endsubroutine HighPrecisionInnerProduct
```

References

[1] A.C. Antoulas, D.C. Sorensen, S. Gugercin, A survey of model reduction methods for large-scale systems, in: Contemp. Math., vol. 280, 2001, pp. 193–220.
[2] Z. Bai, Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems, Appl. Numer. Math. 43 (2002) 9–44.
[3] G.A. Baker Jr., P. Graves-Morris, Padé Approximants, Encycl. Math. Appl., 1996.
[4] P. Benner, V. Mehrmann, D.C. Sorensen, Dimension Reduction of Large-Scale Systems, vol. 45, Springer, 2005.
[5] A. Bodendiek, M. Bollhöfer, Adaptive-order rational Arnoldi-type methods in computational electromagnetism, BIT Numer. Math. (2013) 1–24.

[6] R.-U. Boerner, Numerical modelling in geo-electromagnetics: advances and challenges, Surv. Geophys. 31 (2010) 225–245.
[7] V. Druskin, C. Lieberman, M. Zaslavsky, On adaptive choice of shifts in rational Krylov subspace reduction of evolutionary problems, SIAM J. Sci. Comput. 32 (5) (2010) 2485–2496.
[8] V. Druskin, V. Simoncini, Adaptive rational Krylov subspaces for large-scale dynamical systems, Syst. Control Lett. 60 (8) (2011) 546–560.
[9] V. Druskin, V. Simoncini, M. Zaslavsky, Solution of the time-domain inverse resistivity problem in the model reduction framework part I: one-dimensional problem with SISO data, SIAM J. Sci. Comput. 35 (3) (2013) A1621–A1640.
[10] V. Druskin, M. Zaslavsky, On convergence of Krylov subspace approximations of time-invariant self-adjoint dynamical systems, Linear Algebra Appl. 436 (10) (2012) 3883–3903.
[11] H.C. Elman, K. Meerbergen, A. Spence, M. Wu, Lyapunov inverse iteration for identifying Hopf bifurcations in models of incompressible flow, SIAM J. Sci. Comput. 34 (3) (2012) A1584–A1606.
[12] M.E. Everett, Theoretical developments in electromagnetic induction geophysics with selected applications in the near-surface, Surv. Geophys. 33 (2012) 29–63.
[13] R.W. Freund, Model reduction methods based on Krylov subspaces, Acta Numer. 12 (2003) 267–319.
[14] K. Gallivan, G. Grimme, P. Van Dooren, A rational Lanczos algorithm for model reduction, Numer. Algorithms 12 (1) (1996) 33–63.
[15] E.J. Grimme, Krylov projection methods for model reduction, Ph.D. thesis, University of Illinois, Urbana-Champaigne, IL, 1997.
[16] S. Gugercin, A.C. Antoulas, C. Beattie, H_2 model reduction for large-scale linear dynamical systems, SIAM J. Matrix Anal. Appl. 30 (2) (2008) 609–638.
[17] S. Güttel, Rational Krylov approximation of matrix functions: numerical methods and optimal pole selection, GAMM-Mitt. 36 (1) (2013) 8–31.
[18] S. Güttel, L. Knizhnerman, A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions, BIT Numer. Math. 53 (3) (2013) 595–616.
[19] J.S. Han, E.B. Rudnyi, J.G. Korvink, Efficient optimization of transient dynamic problems in MEMS devices using model order reduction, J. Micromech. Microeng. 15 (2005) 822.
[20] H. Hasegawa, Utilizing the quadruple-precision floating-point arithmetic operation for the Krylov subspace methods, in: The 8th SIAM Conference on Applied Linear Algebra, 2003.
[21] HidaY.1 , X.S. Li, D.H. Bailey, Library for double–double and quad-double arithmetic, NERSC Division, Lawrence Berkeley National Laboratory, 2007.
[22] L. Knizhnerman, V. Druskin, M. Zaslavsky, On optimal convergence rate of the rational Krylov subspace reduction for electromagnetic problems in unbounded domains, SIAM J. Numer. Anal. 47 (2) (2009) 953–971.
[23] T.V. Kolev, P.S. Vassilevski, Some experience with a H1-based auxiliary space AMG for H(curl) problems, Lawrence Livermore Nat. Lab., Livermore, CA, 2006, Rep. UCRL-TR-221841.
[24] M. Kordy, E. Cherkaev, P. Wannamaker, Null space correction and adaptive model order reduction in multi-frequency Maxwell's problem, submitted, 2016.
[25] M. Kordy, P. Wannamaker, V. Maris, E. Cherkaev, Three-dimensional magnetotelluric inversion including topography using deformed hexahedral edge finite elements and direct solvers parallelized on SMP computers, Part I: forward problem and parameter jacobians, Geophys. J. Int. 204 (1) (2016) 74–93.
[26] M. Kordy, P. Wannamaker, V. Maris, E. Cherkaev, G.J. Hill, Three-dimensional magnetotelluric inversion including topography using deformed hexahedral edge finite elements and direct solvers parallelized on SMP computers, part II: direct data-space inverse solution, Geophys. J. Int. 204 (1) (2016) 94–110.
[27] E. Levin, E.B. Saff, Potential theoretic tools in polynomial and rational approximation, in: Harmonic Analysis and Rational Approximation, Springer, 2006, pp. 71–94.
[28] I. Moret, Rational Lanczos approximations to the matrix square root and related functions, Numer. Linear Algebra Appl. 16 (6) (2009) 431–445.
[29] W.A. Mulder, A multigrid solver for 3D electromagnetic diffusion, Geophys. Prospect. 54 (5) (2006) 633–649.
[30] I.V. Oseledets, Lower bounds for separable approximations of the Hilbert kernel, Sb. Math. 198 (3) (2007) 425.
[31] M. Popolizio, V. Simoncini, Acceleration techniques for approximating the matrix exponential operator, SIAM J. Matrix Anal. Appl. 30 (2) (2008) 657–683.
[32] S. Ragni, Rational Krylov methods in exponential integrators for European option pricing, Numer. Linear Algebra Appl. 21 (4) (2014) 494–512.
[33] B. Ralph-Uwe, O.G. Ernst, K. Spitzer, Fast 3-D simulation of transient electromagnetic fields by model reduction in the frequency domain using Krylov subspace projection, Geophys. J. Int. 173 (3) (2008) 766–780.
[34] T. Ransford, Potential Theory in the Complex Plane, vol. 28, Cambridge University Press, 1995.
[35] M. Rewinski, J. White, A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 22 (2) (2003) 155–170.
[36] E.B. Saff, V. Totik, Logarithmic Potentials With External Fields, vol. 316, Springer, 1997.
[37] X.-D.T. Sheldon, H. Lei, Advanced Model Order Reduction Techniques in VLSI Design, Cambridge University Press, 2007.
[38] M. Zaslavsky, V. Druskin, A. Abubakar, T. Habashy, V. Simoncini, Large-scale Gauss–Newton inversion of transient CSEM data using the model order reduction framework, Geophysics 78 (4) (2013) E161–E171.