THE INFINITE BI-LANCZOS METHOD FOR NONLINEAR EIGENVALUE PROBLEMS *

SARAH W. GAAF†, ELIAS JARLEBRING‡, AND MICHIEL E. HOCHSTENBACH*

Abstract. We propose a two-sided Lanczos method for the nonlinear eigenvalue problem. This two-sided approach provides approximations to both the right and left eigenvectors of the eigenvalues of interest. The method implicitly works with matrices and vectors with infinite size, but because particular (starting) vectors are used, all computations can be carried out efficiently with finite matrices and vectors. We specifically introduce a new way to represent infinite vectors that span the subspace corresponding to the conjugate transpose operation for approximating the left eigenvectors. Furthermore, we show that also in this infinite dimensional interpretation the short recurrences inherent to the Lanczos procedure offer an efficient algorithm regarding both the computational cost and the storage.

Key words. Nonlinear eigenvalue problem, infinite bi-Lanczos, infinite two-sided Lanczos.

AMS subject classifications. 65F15, 65H17, 65F50

1. Introduction.

Let $M : \mathbb{C} \to \mathbb{C}^{n\times n}$ be a matrix depending on a parameter with elements which are analytic in $\rho\bar{\mathbb{D}}$, where $\bar{\mathbb{D}}$ is the closed unit disk and $\rho > 0$ a constant. We present a new method for the nonlinear eigenvalue problem: find $(\lambda, x, y) \in \rho\mathbb{D} \times \mathbb{C}^n \times \mathbb{C}^n$, where $x \neq 0$, $y \neq 0$, such that

$$M(\lambda)\, x = 0 \tag{1.1a}$$
$$M(\lambda)^* y = 0 \tag{1.1b}$$

where $\mathbb{D}$ is the open unit disk. We are interested in both the left and the right eigenvectors of the problem. The simultaneous approximation of both left and right eigenvectors is useful, e.g., in the estimation of the eigenvalue condition number and the vectors can be used as initial values for locally convergent two-sided iterative methods, e.g., those described in [10]. The NEP (1.1) has received considerable attention in the numerical linear algebra community, and there are several competitive numerical methods. There are for instance, so-called single vector methods such as Newton type methods [10, 11], which often can be improved with subspace acceleration, see [14], and Jacobi–Davidson methods [4]. These have been extended in a block sense [7]. There are methods specialized for symmetric problems that have an (easily computed) Rayleigh functional [12]. There is also a recent class of methods which can be interpreted as either dynamically extending an approximation or carrying out an infinite-dimensional algorithm, see for instance [6, 2, 5] and references therein. For recent developments see the summary papers [9, 8] and the benchmark collection [3].

We propose a new method that is based on the two-sided Lanczos method for non-Hermitian problems. An intuitive derivation of the main idea of this paper is the following. Suppose (λ, x) is a solution to (1.1a). By adding trivial identities we have

*Version January 20, 2016.

†Department of Mathematics and Computer Science, TU Eindhoven, PO Box 513, 5600 MB Eindhoven, The Netherlands, s.w.gaaf@tue.nl, www.win.tue.nl/∼hochsten/. These authors were supported by an NWO Vidi research grant.

‡Department of Mathematics, Royal Institute of Technology (KTH), Stockholm, SeRC Swedish e-Science Research Center, eliasj@kth.se.

an equality between vectors of infinite length (cf. [6])

$$\begin{bmatrix} -M(0) & & & \\ & I & & \\ & & I & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} \frac{\lambda^0}{0!}x \\ \frac{\lambda^1}{1!}x \\ \frac{\lambda^2}{2!}x \\ \vdots \end{bmatrix} = \lambda \begin{bmatrix} \frac{1}{1}M'(0) & \frac{1}{2}M''(0) & \cdots \\ \frac{1}{1}I & & \\ & \frac{1}{2}I & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \frac{\lambda^0}{0!}x \\ \frac{\lambda^1}{1!}x \\ \frac{\lambda^2}{2!}x \\ \vdots \end{bmatrix}. \tag{1.2}$$

Here, I is the $n \times n$ identity matrix. Throughout the paper we assume that 0 is not an eigenvalue, so that $M(0)^{-1}$ exists. (This does not represent a loss of generality, as we can apply a shift in case 0 is an eigenvalue.) Let $\mathbf{N} \in \mathbb{C}^{n\times\infty}$ be defined by

$$\begin{aligned} \mathbf{N} &:= \begin{bmatrix} N_1 & N_2 & N_3 & \cdots \end{bmatrix} \\ &:= \begin{bmatrix} -M(0)^{-1}M'(0) & -\frac{1}{2}M(0)^{-1}M''(0) & -\frac{1}{3}M(0)^{-1}M^{(3)}(0) & \cdots \end{bmatrix} \end{aligned}$$

and define a vector of infinite length $\mathbf{v} := [v_j]_{j=1}^{\infty} = [\frac{\lambda^{(j-1)}}{(j-1)!}x]_{j=1}^{\infty}$, where $v_j \in \mathbb{C}^n$ for $j = 1, 2, \ldots$. Relation (1.2) can now be more compactly expressed as

$$\mathbf{v} = \lambda\left(\mathbf{e}_1 \otimes \mathbf{N} + \mathbf{S} \otimes I\right)\mathbf{v}, \qquad \text{where} \quad \mathbf{S} := \begin{bmatrix} 0 & 0 & \cdots \\ \frac{1}{1} & & \\ & \frac{1}{2} & \\ & & \ddots \end{bmatrix}, \tag{1.3}$$

and $\mathbf{e}_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots \end{bmatrix}^T$ is the first basis vector. Equations (1.2) and (1.3) may be viewed as a companion linearization for the nonlinear eigenvalue problem. Note that a solution λ to (1.3) corresponds to a reciprocal eigenvalue of the infinite-dimensional matrix

$$\mathbf{A} := \mathbf{e}_1 \otimes \mathbf{N} + \mathbf{S} \otimes I. \tag{1.4}$$

The derivation of our new two-sided Lanczos procedure is based on applying the Lanczos method (for non-Hermitian problems) to the infinite-dimensional matrix $\mathbf{A}$. The method builds two bi-orthogonal subspaces using short recurrences. One subspace serves the approximation of right eigenvectors, the other the approximation of the left eigenvectors. In Section 2 we use that, analogous to companion linearizations for polynomial eigenvalue problems, relation (1.3) is equivalent to (1.1a), which has been used in [6] to derive one version of the infinite Arnoldi method. For the approximation of solutions to (1.1b) we derive a new and more involved relationship for the left eigenvectors, also presented in Section 2. This leads to a new way to represent infinite vectors that span the subspace corresponding to the conjugate transpose operation for approximating the left eigenvectors. With two particular types of (starting) vectors, we can carry out an algorithm for the infinite-dimensional operator $\mathbf{A}$ using only finite arithmetic. This is covered in Section 3, where we also treat the computation of scalar products and matrix-vector products for the infinite dimensional case. In Section 4 we present an example to illustrate the performance of the new method, and we conclude with a short discussion.

Throughout this paper we use bold symbols to indicate matrices or vectors of infinite dimensions, i.e., an infinite matrix is denoted by $\mathbf{A} \in \mathbb{R}^{\infty\times\infty}$, and an infinite-dimensional vector is denoted by $\mathbf{x} \in \mathbb{R}^{\infty}$. Unless otherwise stated, the n-length blocks of a vector of infinite length are denoted with subscript, e.g., $\mathbf{w} = [w_1^T, w_2^T, \ldots]^T$ where $w_j \in \mathbb{C}^n$ for $j \geq 1$.

2. Infinite dimensional reformulation. In our formalization of the operator $\mathbf{A}$ we first need to define its domain. This is necessary to prove equivalence between (λ, x, y) which is a solution to (1.1) and the eigentriplet $(\mu, \mathbf{v}, \mathbf{w})$ of $\mathbf{A}$, where $\mu = \lambda^{-1}$. Let $\|\cdot\|$ denote the 2-norm. It will turn out to be natural to define the operators on a weighted, mixed 1-norm and 2-norm space defined by

$$\ell_1(\rho) := \Big\{ \mathbf{w} = [w_j]_{j=1}^{\infty} \in \mathbb{R}^{\infty} : \sum_{j=1}^{\infty} \tfrac{\rho^j}{j!} \|w_j\| < \infty \Big\}. \tag{2.1}$$

Note that some vectors in $\ell_1(\rho)$ correspond to sequences of vectors which are unbounded, i.e., $\|w_j\| \to \infty$ as $j \to \infty$, but do not grow arbitrarily fast, since $\mathbf{w} \in \ell_1(\rho)$ implies that

$$\tfrac{\rho^j}{j!} \|w_j\| \to 0 \quad \text{as } j \to \infty. \tag{2.2}$$

In the proofs of the propositions below we need to allow the vectors to have this growth, to accommodate the fact that derivatives of analytic functions are not necessarily bounded. We will let ρ be the convergence radius of the power series expansion of the analytic function M, and set $\mathcal{D}(\mathbf{A}) = \mathcal{D}(\mathbf{A}^*) = \ell_1(\rho)$ as the domain of the operator. The following two propositions do not only show the equivalence between the nonlinear eigenvalue problem and the operator $\mathbf{A}$, but also reveal the structure of the left and right eigenvectors of $\mathbf{A}$. The first result is an adaption of [6, Thm. 1] for our discrete operator and only assuming a finite convergence radius. Its proof is omitted for brevity.

PROPOSITION 2.1 (Right eigenvectors of $\mathbf{A}$). *Suppose M is analytic in $\lambda \in \rho\bar{\mathbb{D}}$ and let $\mathbf{A}$ be defined by* (1.4).

(i) If $(\mu, \mathbf{v}) \in \mathbb{C} \times \mathcal{D}(\mathbf{A}) \backslash \{0\}$ is an eigenpair of $\mathbf{A}$ and $\lambda = \mu^{-1} \in \rho\mathbb{D}$, then there exists a vector $x \in \mathbb{C}^n$ such that

$$\mathbf{v} = \Big[\tfrac{\lambda^{j-1}}{(j-1)!} x \Big]_{j=1}^{\infty}. \tag{2.3}$$

(ii) The pair $(\lambda, x) \in \rho\mathbb{D}\backslash\{0\} \times \mathbb{C}^n \backslash \{0\}$ is a solution to (1.1a) *if and only if the pair $\big(\lambda^{-1}, \mathbf{v}\big) \in (\mathbb{C}\backslash\rho^{-1}\bar{\mathbb{D}}) \times \mathcal{D}(\mathbf{A})$ is an eigenpair of $\mathbf{A}$, where $\mathbf{v}$ is given by* (2.3).

We now study the equivalence between a left eigenpair of the nonlinear eigenvalue problem and a left eigenpair of $\mathbf{A}$. Also, the structure of the left eigenvectors of $\mathbf{A}$ will be concretized.

PROPOSITION 2.2 (Left eigenvectors of $\mathbf{A}$). *Suppose M is analytic in $\lambda \in \rho\bar{\mathbb{D}}$ and let $\mathbf{A}^*$ be defined by* (1.4).

(i) If $(\mu, \mathbf{w}) \in \mathbb{C} \times \mathcal{D}(\mathbf{A}^*) \backslash \{0\}$ is an eigenpair of $\mathbf{A}^*$ and $\lambda = \mu^{-1} \in \rho\mathbb{D}$, then there exists a vector $z \in \mathbb{C}^n$ such that

$$\mathbf{w} = \sum_{j=1}^{\infty} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \lambda^j z. \tag{2.4}$$

(ii) The pair $(\lambda, y) \in \rho\mathbb{D}\backslash\{0\} \times \mathbb{C}^n \backslash \{0\}$ is a solution to (1.1b) *if and only if the pair $\big(\lambda^{-1}, \mathbf{w}\big) \in (\mathbb{C}\backslash\rho^{-1}\bar{\mathbb{D}}) \times \mathcal{D}(\mathbf{A}^*)$ is an eigenpair of $\mathbf{A}^*$, where $\mathbf{w}$ is given by* (2.4) *with $z = M(0)^* y$.*

Proof. Suppose $\lambda \mathbf{A}^* \mathbf{w} = \mathbf{w}$, where $\mathbf{w} \in \ell_1(\rho)$. We use induction to show that

$$w_1 = \sum_{j=1}^{k} \tfrac{\lambda^j}{(j-1)!} N_j^* \, w_1 + \tfrac{\lambda^k}{k!} \, w_{k+1} \tag{2.5}$$

for any k. Relation (2.5) is easy to see for $k = 1$. Suppose (2.5) is satisfied for $k-1$, i.e.,

$$w_1 = \sum_{j=1}^{k-1} \tfrac{\lambda^j}{(j-1)!} N_j^* \, w_1 + \tfrac{\lambda^{k-1}}{(k-1)!} \, w_k. \tag{2.6}$$

Block row k of $\lambda \mathbf{A}^* \mathbf{w} = \mathbf{w}$ reduces to

$$\lambda \, N_k^* \, w_1 + \tfrac{\lambda}{k} \, w_{k+1} = w_k. \tag{2.7}$$

The induction is completed by inserting (2.7) in relation (2.6), which yields (2.5). Due to the fact that $\mathbf{w} \in \ell_1(\rho)$, (2.2) holds, and since $|\lambda| < \rho$, $\|\frac{\lambda^k}{k!} w_{k+1}\| < \frac{\rho^k}{k!} \|w_{k+1}\| \to 0$ as $k \to \infty$. This implies that (2.5) holds also in the limit $k \to \infty$ and

$$w_1 = \sum_{j=1}^{\infty} \tfrac{\lambda^j}{(j-1)!} N_j^* \, w_1 = (\mathbf{e}_1^T \otimes I)\Big(\sum_{j=1}^{\infty} \lambda^j (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* w_1 \Big). \tag{2.8}$$

In the last equality in (2.8) we used that

$$\mathbf{S}^j \mathbf{e}_k = \tfrac{(k-1)!}{(j+k-1)!} \, \mathbf{e}_{k+j} \tag{2.9}$$

and therefore $(\mathbf{e}_k^T \otimes I)(\mathbf{S}^T \otimes I)^{j-1} = \mathbf{e}_k^T (\mathbf{S}^T)^{j-1} \otimes I = \frac{(k-1)!}{(j+k-2)!} \mathbf{e}_{k+j-1}^* \otimes I$ for any k, as well as $(\mathbf{e}_j^T \otimes I)\mathbf{N}^* = N_j^*$. By setting $z = w_1$ we have with (2.8) proven the first block row of (2.4). The proof of the other rows follows from induction, since assuming that $w_k = (\mathbf{e}_k^T \otimes I)\mathbf{w}$, where $\mathbf{w}$ is the right-hand side of (2.4), and using (2.7) we find that $w_{k+1} = (\mathbf{e}_{k+1}^* \otimes I)\mathbf{w}$.

To show (ii), first assume that $\mathbf{w} \in \ell_1(\rho)$ satisfies $\lambda \mathbf{A} \mathbf{w} = \mathbf{w}$. This is the same assumption as in (i) and therefore (2.8) is satisfied. By setting $y = M(0)^{-*} z = M(0)^{-*} w_1$, we have that $M(0)^* y = \sum_{j=1}^{\infty} M^{(j)}(0)^* y$, i.e., (1.1b) is satisfied.

To show the backward implication in (ii), we now assume that (λ, y) is a solution to (1.1b). Let $z = M(0)^* y$ and define a vector $\mathbf{w}$ as (2.4). Then

$$\begin{aligned}
\lambda \mathbf{A}^* \mathbf{w} &= \lambda \sum_{j=1}^{\infty} (\mathbf{e}_1^T \otimes \mathbf{N}^* + \mathbf{S}^T \otimes I)(\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \lambda^j z \\
&= \lambda \mathbf{N}^* \sum_{j=1}^{\infty} \tfrac{1}{(j-1)!} (\mathbf{e}_j^T \otimes I) \mathbf{N}^* \lambda^j z + \lambda \sum_{j=1}^{\infty} (\mathbf{S}^T \otimes I)^j \mathbf{N}^* \lambda^j z \\
&= \lambda \mathbf{N}^* \sum_{j=1}^{\infty} \tfrac{-1}{j!} M^{(j)}(0)^* y + \sum_{j=2}^{\infty} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \lambda^j z \\
&= \mathbf{N}^* \lambda z + \sum_{j=2}^{\infty} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \lambda^j z = \sum_{j=1}^{\infty} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \lambda^j z = \mathbf{w}.
\end{aligned}$$

To show $\mathbf{w} \in \ell_1(\rho)$ we now study the weighted ℓ_1-norm,

$$\sum_{k=1}^{\infty} \tfrac{\rho^k}{k!}\|w_k\| \le \sum_{k=1}^{\infty} \tfrac{\rho^k}{k!} \sum_{j=1}^{\infty} \tfrac{|\lambda|^j (k-1)!}{(j+k-2)!}\|M^{(k+j-1)}(0)^*\|\|\widehat{y}\| \tag{2.10}$$
$$\le \sum_{k=1}^{\infty} \tfrac{\rho^k}{k!} \sum_{j=1}^{\infty} M_\rho \tfrac{|\lambda|^j (k-1)!(k+j-1)!}{(j+k-2)!\rho^{j+k-1}}\|\widehat{y}\| = \tfrac{M_\rho\|\widehat{y}\|}{r} \sum_{j=1}^{\infty}\sum_{k=1}^{\infty} \tfrac{|\lambda|^j}{\rho^j}\tfrac{j+k-1}{k!},$$

where, since M is analytic, there exists a constant M_ρ such that $\|M^{(j)}(0)\| \le M_\rho \frac{j!}{\rho^j}$. Now note that Taylor expansion of e^x gives the explicit expression $\sum_{k=1}^{\infty} \frac{j+k-1}{k!} = (j-1)(e-1)+e$. By combining this with (2.10) and $|\lambda| < \rho$ we find that the right-hand side of (2.10) is finite and therefore $\mathbf{w} \in \ell_1(\rho)$. □

3. Derivation of the infinite bi-Lanczos method. The algorithm proposed in this paper is based on the Lanczos method for non-Hermitian eigenvalue problems specified in [1, Section 7.8.1]. We first introduce the standard method and then adapt the algorithm in such a way that it can be used for the infinite-dimensional problem. Therefore we need to define various operations for vectors and matrices of infinite dimension and also the Krylov subspaces created by the algorithms are described.

3.1. The bi-Lanczos method for standard eigenvalue problems. We base our theory on the Lanczos method for non-Hermitian eigenvalue problems, also called the bi-Lanczos method. The method uses an oblique projection building two bi-orthogonal subspaces for the simultaneous approximation of left and right eigenvectors. The short recurrences that are typical for this method lead to far less storage requirements with respect to orthogonal projection methods for the same problem. After k iterations we obtain the relations:

$$\begin{aligned} AQ_k &= Q_k T_k + \beta_{k+1} q_{k+1} e_k^T, \\ A^*\widetilde{Q}_k &= \widetilde{Q}_k T_k^* + \bar{\gamma}_{k+1}\widetilde{q}_{k+1} e_k^T, \\ \widetilde{Q}_k^* Q_k &= I_k, \end{aligned}$$

where for $i = 1, \ldots, k$ the columns of Q_k are equal to the vectors q_i, and $\widetilde{q}_i$ are the columns of $\widetilde{Q}_k$, e_k is the kth unit vector, and the tridiagonal matrix T_k is defined as

$$T_k = \begin{bmatrix} \alpha_1 & \gamma_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \gamma_k \\ & & \beta_k & \alpha_k \end{bmatrix}.$$

Furthermore, the relations $\widetilde{q}_{k+1}^* Q_k = 0$ and $\widetilde{Q}_k^* q_{k+1} = 0$ hold. After k iterations, one can compute the eigentriplets $(\theta_i^{(k)}, z_i^{(k)}, \widetilde{z}_i^{(k)})$, $i = 1, 2, \ldots, k$, of T_k. The Ritz values $\theta_i^{(k)}$ are the approximate eigenvalues of A, and the corresponding right and left Ritz vectors are $x_i^{(k)} = Q_k z_i^{(k)}$ and $y_i^{(k)} = \widetilde{Q}_k \widetilde{z}_i^{(k)}$, respectively.

3.2. Krylov subspace and infinite-dimensional vector representations. For our infinite-dimensional problem where we work with the matrix $\mathbf{A}$ and vectors of infinite length, we need to build infinite dimensional Krylov spaces, $\mathcal{K}_k(\mathbf{A}, \mathbf{x})$ and $\mathcal{K}_k(\mathbf{A}^*, \widetilde{\mathbf{y}})$, for some starting vectors $\mathbf{x}$ and $\widetilde{\mathbf{y}}$ of infinite length. To adapt the algorithm

to the infinite-dimensional problem we have to address the issue of storing vectors with infinite length. By choosing the starting vectors carefully we will be able to store only a finite number of vectors of length n. The Krylov subspaces will contain vectors that are consistent with eigenvector approximations.

PROPOSITION 3.1. *Suppose* $\mathbf{x} = \mathbf{e}_1 \otimes x_1$ *and* $\widetilde{\mathbf{y}} = \mathbf{N}^* \widetilde{y}_1$, *where* $x_1, \widetilde{y}_1 \in \mathbb{C}^n$.

(a) For any $k \in \mathbb{N}$, $\mathbf{A}^k \mathbf{x} = \sum_{j=1}^{k+1} (\mathbf{e}_j \otimes z_{k-j+1})$, *where* $z_0 = \frac{1}{k!} x_1$ *and for* $i \in \{1, \ldots, k\}$ z_i *is given by the recursion* $z_i = \sum_{\ell=1}^{i} \frac{(k-i+\ell)!}{(\ell-1)!(k-i)!} N_\ell z_{i-\ell}$.

(b) For any $k \in \mathbb{N}$, $(\mathbf{A}^*)^k \widetilde{\mathbf{y}} = \sum_{j=1}^{k+1} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \widetilde{z}_{k-j+1}$, *where* $\widetilde{z}_0 = \widetilde{y}_1$ *and for* $i \in \{1, \ldots, k\}$ $\widetilde{z}_i$ *is given by the recurrence relation* $\widetilde{z}_i = \sum_{\ell=1}^{i} \frac{1}{(\ell-1)!} N_\ell^* z_{i-\ell}$.

Proof. For brevity the details of the proofs (using induction) are left out. ∎

As we have seen in Propositions 2.1 and 2.2, the right and left eigenvectors of interest have the form (2.3) and (2.4), respectively. Proposition 3.1 has shown that by choosing starting vectors $\mathbf{x} = \mathbf{e}_1 \otimes x_1$ and $\widetilde{\mathbf{y}} = \mathbf{N}^* \widetilde{y}_1$ the vectors that span the Krylov subspaces are of the form

$$\mathbf{a} = \sum_{j=1}^{k_a} (\mathbf{e}_j \otimes a_j), \tag{3.1a}$$

$$\widetilde{\mathbf{a}} = \sum_{j=1}^{k_{\widetilde{a}}+1} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \widetilde{a}_j, \tag{3.1b}$$

respectively. Also linear combinations of vectors from the same Krylov subspaces, and therefore also the approximate eigenvectors, will be of this form. We will distinguish the two types of vectors (3.1a) and (3.1b) by a tilde. These vectors can be seen as a truncated version of the vectors in (2.3) and (2.4). Vectors of the form (3.1a) have a finite number of nonzeros and therefore storing only the nonzero entries gives a finite representation of the vector of infinite length, i.e., by storing the vectors a_j, for $j = 1, \ldots, k$. The vectors of infinite length of type (3.1b) can also be stored with a finite number of vectors in $\mathbb{C}^n$, namely by storing the vectors $\widetilde{a}_j$, for $j = 1, \ldots, k$.

3.3. Scalar products and matrix vector products. The previously introduced types of infinite-dimensional vectors, (3.1a) and (3.1b), will be used in the algorithm for the infinite-dimensional problem. Various operations involving these types of vectors of infinite length, such as scalar products and matrix vector products, have to be adapted to the infinite-dimensional case. First we introduce two different scalar products.

PROPOSITION 3.2. *Suppose* $\mathbf{a}, \mathbf{b} \in \mathbb{C}^\infty$ *are two vectors of type* (3.1a) *given by* $\mathbf{a} = \sum_{j=1}^{k_a} (\mathbf{e}_j \otimes a_j)$ *and* $\mathbf{b} = \sum_{j=1}^{k_b} (\mathbf{e}_j \otimes b_j)$. *Then,* $\mathbf{a}^* \mathbf{b} = \sum_{j=1}^{\min(k_a, k_b)} a_j^* b_j$.

Proof. This follows straightforwardly from the definition of the vectors. ∎

Another scalar product used in the bi-Lanczos algorithm is a product of vectors of type (3.1a) and (3.1b). It can be computed efficiently in infinite dimensions as explained in the next proposition.

PROPOSITION 3.3. *Suppose* $\widetilde{\mathbf{a}}, \mathbf{b} \in \mathbb{C}^\infty$ *are of type* (3.1b) *and* (3.1a)*, respectively, given by* $\widetilde{\mathbf{a}} = \sum_{j=1}^{k_{\widetilde{a}}} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \widetilde{a}_j$ *and* $\mathbf{b} = \sum_{\ell=1}^{k_b} (\mathbf{e}_\ell \otimes b_\ell)$*. Then,*

$$\widetilde{\mathbf{a}}^* \mathbf{b} = \sum_{j=1}^{k_{\widetilde{a}}} \sum_{\ell=1}^{k_b} \tfrac{(\ell-1)!}{(j+\ell-2)!} \widetilde{a}_j^T N_{j+\ell-1} b_\ell. \tag{3.2}$$

Proof. The proof follows easily from the definition of the infinite vectors and rearrangement of sums. It is omitted for brevity. ∎

To translate the finite dimensional matrix vector multiplication to the infinite-dimensional case two variants of matrix vector products have to be investigated, one with the matrix $\mathbf{A}$ and a vector of type (3.1a), and one with the matrix $\mathbf{A}^*$ and a vector of type (3.1b).

PROPOSITION 3.4 (Action of $\mathbf{A}$). *Suppose* $\mathbf{a} \in \mathbb{C}^\infty$ *is of type* (3.1a) *given by* $\mathbf{a} = \sum_{j=1}^{k_a} (\mathbf{e}_j \otimes a_j)$*. Then,*

$$\mathbf{A}\mathbf{a} = \sum_{j=1}^{k_a+1} (\mathbf{e}_j \otimes b_j), \tag{3.3}$$

where $b_j = \frac{1}{j-1} a_{j-1}$ *for* $j = 2, \ldots, k_a + 1$*, and* $b_1 = \sum_{j=1}^{k_a} N_j a_j$*.*

Proof. This can be proven with induction. The computation is analogous to the one needed in the proof of Proposition 3.1(a). ∎

PROPOSITION 3.5 (Action of $\mathbf{A}^*$). *Suppose* $\widetilde{\mathbf{a}} \in \mathbb{C}^\infty$ *is of type* (3.1b) *given by* $\widetilde{\mathbf{a}} = \sum_{j=1}^{k_{\widetilde{a}}} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \widetilde{a}_j$*. Then,*

$$\mathbf{A}^* \widetilde{\mathbf{a}} = \sum_{j=1}^{k_{\widetilde{a}}+1} (\mathbf{S}^T \otimes I)^{j-1} \mathbf{N}^* \widetilde{b}_j, \tag{3.4}$$

where $\widetilde{b}_j = \widetilde{a}_{j-1}$ *for* $j = 2, \ldots k_{\widetilde{a}} + 1$*, and* $\widetilde{b}_1 = \sum_{j=1}^{k_{\widetilde{a}}} \frac{1}{(j-1)!} N_j^* \widetilde{a}_j$*.*

Proof. Analogous to the computation in the proof of Proposition 3.1(b). ∎

3.4. The infinite bi-Lanczos method. As we have seen, every vector of infinite length can be represented by a finite number of vectors of length n. In the algorithm these vectors of infinite length are denoted by matrices whose columns correspond to the length-n vectors representing the infinite-dimensional vector. The index of the matrices in the new algorithm indicate the number of columns of the matrix, i.e., $R_k \in \mathbb{C}^{n \times k}$, and we denote the ℓth column of R_k by $R_{k,\ell}$, i.e., $R_k = [R_{k,1}, \ldots, R_{k,k}]$.

Because of the short recurrences, k steps of the algorithm require the storage of only six vectors of length $\leq kn$, three for each subspace. Furthermore, the computation of the approximate eigenvectors for the solution of (1.1) entails the storage of (only) k vectors of length n for each subspace. To clarify this, recall from Section 3.1 that from the eigentriplet $(\theta_1^{(k)}, z_1^{(k)}, \widetilde{z}_1^{(k)})$ of T_k we can deduce an approximate eigentriplet $(\theta_1^{(k)}, Q_k z_1^{(k)}, \widetilde{Q}_k \widetilde{z}_1^{(k)})$ for $\mathbf{A}$. Note that the columns of Q_k represent vectors

of type (3.1a) and thus a linear combination of the columns is itself a representation of a vector of this type. The same reasoning holds for the columns of $\widetilde{Q}_k$ with vectors of type (3.1b). Suppose s_r stands for the first n-length block of $Q_k z_1^{(k)}$ and the first n-length block of $\widetilde{Q}_k \widetilde{z}_1^{(k)}$ is represented by s_ℓ. Proposition 2.1 states that s_r is an approximation to x and that $((\theta_1^{(k)})^{-1}, s_r)$ is an approximate solution to (1.1a). Similarly, by Proposition 2.2 we know that s_ℓ is an approximation to $\lambda z = \lambda M(0)^* y$. Hence $((\theta_1^{(k)})^{-1}, \theta_1^{(k)} M(0)^{-*} s_\ell)$ is an approximate solution to (1.1b).

Algorithm: Infinite bi-Lanczos

Input: Vectors $q_1, \widetilde{q}_1 \in \mathbb{C}^n$, with $\widetilde{q}_1^* M'(0) q_1 = 1$, $P_0 = \widetilde{P}_0 = [\ ]$, $P_1 = [q_1]$, $\widetilde{P}_1 = [\widetilde{q}_1]$, $\gamma_1 = \beta_1 = 0$.
Output: Approximate eigentriplets $((\theta_i^{(k)})^{-1}, x_i^{(k)}, y_i^{(k)})$ to nonlinear eigenvalue problem (1.1).

---

for $k = 1, 2, \ldots,$ until convergence

1: Compute $R_{k+1} := [b_1, \ldots, b_{k+1}] \in \mathbb{C}^{n \times (k+1)}$ with (3.3) where, $k_a = k$, $a_\ell = P_{k,\ell}$ for $\ell = 1, \ldots, k$.
2: Compute $\widetilde{R}_{k+1} := [\widetilde{b}_1, \ldots, \widetilde{b}_{k+1}] \in \mathbb{C}^{n \times (k+1)}$ with (3.4) where, $k_{\widetilde{a}} = k$, $\widetilde{a}_\ell = \widetilde{P}_{k,\ell}$ for $\ell = 1, \ldots, k$.
3: $R_{k+1} = R_{k+1} - \gamma_k [P_{k-1}, 0, 0]$
4: $\widetilde{R}_{k+1} = \widetilde{R}_{k+1} - \bar{\beta}_k [\widetilde{P}_{k-1}, 0, 0]$
5: Compute $\alpha_k = \widetilde{\mathbf{a}}^T \mathbf{b}$ with (3.2) where $\widetilde{a}_\ell = \widetilde{P}_{k,\ell}$, $\ell = 1, \ldots, k$, and $b_\ell = R_{k+1,\ell}$ for $\ell = 1, \ldots, k+1$ and $k_{\widetilde{a}} = k$ and $k_b = k+1$.
6: $R_{k+1} = R_{k+1} - \alpha_k [P_k, 0]$
7: $\widetilde{R}_{k+1} = \widetilde{R}_{k+1} - \bar{\alpha}_k [\widetilde{P}_k, 0]$
8: Compute $\omega_k = \widetilde{\mathbf{a}}^T \mathbf{b}$ with (3.2) where $\widetilde{a}_\ell = \widetilde{R}_{k+1,\ell}$, $b_\ell = R_{k+1,\ell}$ for $\ell = 1, \ldots, k+1$, where $k_{\widetilde{a}} = k_b = k+1$.
9: $\beta_{k+1} = |\omega_k|^{1/2}$
10: $\gamma_{k+1} = \bar{\omega}_k / \beta_{k+1}$
11: $P_{k+1} = R_{k+1} / \beta_{k+1}$
12: $\widetilde{P}_{k+1} = \widetilde{R}_{k+1} / \bar{\gamma}_{k+1}$
13: Compute eigentriplets $(\theta_i^{(k)}, z_i^{(k)}, \widetilde{z}_i^{(k)})$ of T_k.
14: Test for convergence.

end

Compute approximate eigenvectors $x_i^{(k)}$ and $y_i^{(k)}$ (as described in Section 3.4).

The dominating component in terms of computational complexity of the algorithm corresponds to the scalar products of type (3.2) in steps 5 and 8 of the algorithm. This particular scalar product involves a double sum and leads to a total complexity $\mathcal{O}(nk^3)$, but for some applications, e.g., the one considered in Section 4, it can be computed by using matrix-matrix products (for relatively small matrices) reducing the cost significantly. Furthermore, per iteration two linear systems of equations must be solved, in steps 1 and 2, which is done using a pre-computed LU-factorization. In the same two steps, $\sum_{j=1}^{k_a} M^{(j)}(0)\, a_j$, and $\sum_{j=1}^{k_{\widetilde{a}}} (M^{(j)}(0))^*\, \widetilde{a}_j$ are computed.

4. Numerical experiments. Our approach is intended for large and sparse problems. We illustrate the properties and competitiveness of the algorithm by computing solutions to an artificial large-scale NEP stemming from a second order delay-differential equation,

$$M(\lambda) = -\lambda^2 I + A_0 + e^{-\lambda} A_1, \tag{4.1}$$

where A_0 and A_1 are randomly generated sparse matrices with normally distributed random entries. For the experiments we choose the matrices to be of dimension $n = 1000$. The total number of iterations is equal to $k = 50$.

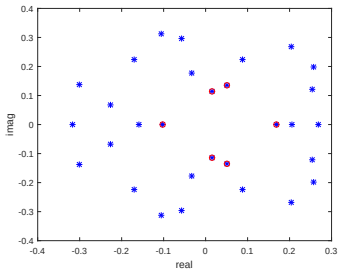

Figure 4.1: Eigenvalue approximations of the infinite bi-Lanczos method applied to problem (4.1). Circles correspond to approximations that have converged after $k_0 = 30$.

i	$\lvert\theta_i^{(k_0)}\rvert^{-1}$	$\kappa((\theta_i^{(k_0)})^{-1})$
1	$1.029 \cdot 10^{-1}$	$1.267 \cdot 10^{3}$
2	$1.157 \cdot 10^{-1}$	$2.510 \cdot 10^{3}$
3	$1.157 \cdot 10^{-1}$	$2.510 \cdot 10^{3}$
4	$1.440 \cdot 10^{-1}$	$1.697 \cdot 10^{3}$
5	$1.440 \cdot 10^{-1}$	$1.697 \cdot 10^{3}$
6	$1.593 \cdot 10^{-1}$	$1.846 \cdot 10^{3}$

Table 4.1: The condition numbers for the converged eigenvalues. The values are computed using the approximate eigentriplets after $k_0 = 30$ iterations of infinite bi-Lanczos.

Figure 4.1 shows the approximated eigenvalues, and distinguishes the ones converged after $k_0 = 30$ iterations by a circle around them, which are obviously the ones closest to zero. The two-sided approach has the advantage that during the process a condition number estimate is available, enabling the user to define a satisfying convergence criterion. The condition numbers shown in Table 4.1 correspond to the converged eigenvalues and can be computed as (cf. [13])

$$\kappa(\lambda, M) := \frac{\alpha \|x\|_2 \|y\|_2}{|\lambda|\,|y^* M'(\lambda) x|} = \frac{\left(|\lambda|^2 \|I\|_2 + \|A_0\|_2 + |e^{-\lambda}| \|A_1\|_2\right) \|x\|_2 \|y\|_2}{|\lambda|\,|y^*(-2\lambda I - e^{-\lambda} A_1) x|}.$$

We also compare the infinite bi-Lanczos method to the infinite Arnoldi method (IAR) as presented in [6]. Figure 4.2 shows for both methods the error in the eigenvalues against the iterations, and Figure 4.3 contains the error of both methods against the computation time in seconds. For the infinite bi-Lanczos method the Ritz values converge in fewer iterations.

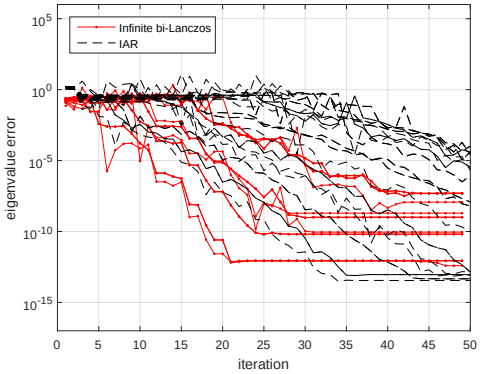

Figure 4.2: Convergence diagram, eigenvalue error against the iterations.

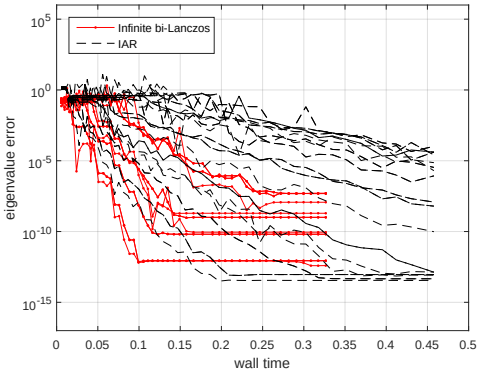

Figure 4.3: Convergence diagram, eigenvalue error against the computation time (s).

This increasing convergence behavior can be explained by the two subspaces that are build in the infinite bi-Lanczos. In fact, with respect to one multiplication with $\mathbf{A}$

per iteration of IAR, infinite bi-Lanczos contains per iteration a multiplication with both $\mathbf{A}$ and $\mathbf{A}^*$. Because of the short recurrences the computation time of infinite bi-Lanczos can be kept decently low (and may even outperform IAR), as shown in Figure 4.3. One has to take into account that for growing dimensions infinite bi-Lanczos might become unstable, which here explains the stagnation of the approximation error in Figure 4.2 and Figure 4.3. Although the observations above have been seen for several problems, the figures are very much problem dependent; the sparsity pattern and the nonlinearity of the problem will be decisive for the performance of different methods.

5. Discussion and conclusions. We have proposed a new two-sided Lanczos method for the nonlinear eigenvalue problem. The method works implicitly with matrices and vectors with infinite size. The new way of representing left type of infinite vectors is crucial to frame the two-sided method. We intend to make the code adaptive, as the condition numbers which become available as the iterations proceed may be used to define a satisfying convergence criterion. We have seen that infinite bi-Lanczos can have faster convergence per iteration than the infinite Arnoldi method (IAR), which could be expected because in general two-sided methods have faster convergence (per iteration), and moreover, since infinite bi-Lanczos uses a low-term recurrence it has a lower orthogonalization cost per iteration than IAR.

REFERENCES

[1] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. A. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, Philadelphia, 2000.

[2] R. Van Beeumen, K. Meerbergen, and W. Michiels. A rational Krylov method based on Hermite interpolation for nonlinear eigenvalue problems. *SIAM J. Sci. Comput.*, 35(1):A327–A350, 2013.

[3] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur. NLEVP: A collection of nonlinear eigenvalue problems. *ACM Trans. Math. Softw.*, 39(2):1–28, 2013.

[4] T. Betcke and H. Voss. A Jacobi-Davidson type projection method for nonlinear eigenvalue problems. *Future Generation Computer Systems*, 20(3):363–372, 2004.

[5] S. Güttel, R. Van Beeumen, K. Meerbergen, and W. Michiels. NLEIGS: a class of fully rational Krylov methods for nonlinear eigenvalue problems. *SIAM J. Sci. Comput.*, 36(6):A2842–A2864, 2014.

[6] E. Jarlebring, W. Michiels, and K. Meerbergen. A linear eigenvalue algorithm for the nonlinear eigenvalue problem. *Numer. Math.*, 122(1):169–195, 2012.

[7] D. Kressner. A block Newton method for nonlinear eigenvalue problems. *Numer. Math.*, 114(2):355–372, 2009.

[8] V. Mehrmann and H. Voss. Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods. *GAMM-Mitt.*, 27:121–152, 2004.

[9] A. Ruhe. Algorithms for the nonlinear eigenvalue problem. *SIAM J. Numer. Anal.*, 10:674–689, 1973.

[10] K. Schreiber. *Nonlinear Eigenvalue Problems: Newton-type Methods and Nonlinear Rayleigh Functionals*. PhD thesis, TU Berlin, 2008.

[11] D. B. Szyld and F. Xue. Local convergence analysis of several inexact Newton-type algorithms for general nonlinear eigenvalue problems. *Numer. Math.*, 123:333–362, 2012.

[12] D. B. Szyld and F. Xue. Preconditioned eigensolvers for large-scale nonlinear Hermitian eigenproblems with variational characterizations. I. Extreme eigenvalues. *Mathematics of Computation*, 2015. Accepted for publication.

[13] F. Tisseur. Backward error and condition of polynomial eigenvalue problems. *Linear Algebra Appl.*, 309:339–361, 2000.

[14] H. Voss. An Arnoldi method for nonlinear eigenvalue problems. *BIT*, 44:387 – 401, 2004.