Journal of Computational and Applied Mathematics 226 (2009) 10–21

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A two-directional Arnoldi process and its application to parametric model order reduction

Yung-Ta Li [a], Zhaojun Bai [a,b,*], Yangfeng Su [c]

[a] Department of Mathematics, University of California, Davis, CA 95616, USA

[b] Department of Computer Science, University of California, Davis, CA 95616, USA

[c] School of Mathematical Sciences, Fudan University, Shanghai 200433, China

ARTICLE INFO

Article history:
Received 2 October 2007
Received in revised form 18 April 2008

Keywords:
Krylov subspace
Arnoldi process
Parameterized systems
Model order reduction
Projective techniques

ABSTRACT

We consider a two-directional Krylov subspace $\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$, where besides the dimensionality k of the subspace increases, the matrix $\mathbf{A}_{[j]}$ and vector $\mathbf{b}_{[j]}$ which induce the subspace may also augment. Specifically, we consider the case where the matrix $\mathbf{A}_{[j]}$ and the vector $\mathbf{b}_{[j]}$ are augmented by block triangular bordering. We present a two-directional Arnoldi process to efficiently generate a sequence of orthonormal bases $\mathbf{Q}_k^{[j]}$ of the Krylov subspaces. The concept of a two-directional Krylov subspace and an Arnoldi process is triggered by the need of a multiparameter moment-matching based model order reduction technique for parameterized linear dynamical systems. Numerical examples illustrate computational efficiency and flexibility of the proposed two-directional Arnoldi process.

1. Introduction

A Krylov subspace $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ based on a matrix $\mathbf{A}$ and a vector $\mathbf{b}$ is a subspace spanned by a sequence of column vectors:

$$\mathcal{K}_k(\mathbf{A}, \mathbf{b}) = \text{span}\left\{\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \ldots, \mathbf{A}^{k-1}\mathbf{b}\right\}. \tag{1}$$

The Arnoldi process [1] is an efficient numerical procedure to generate an orthonormal basis of the subspace. The Krylov subspace and Arnoldi process play an important role in modern computational techniques for large-scale matrix computation problems, such as solving linear systems of equations [13,15,23,24], computing a few selected eigenpairs [3,25,21] and reduced-order modeling of dynamical systems [18,2,16]. A Krylov subspace-based method is often the method of choice due to its simplicity and efficiency.

In this paper, we consider a two-directional Krylov subspace:

$$\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]}) = \text{span}\left\{\mathbf{b}_{[j]},\ \mathbf{A}_{[j]}\mathbf{b}_{[j]},\ \mathbf{A}_{[j]}^2\mathbf{b}_{[j]},\ \ldots,\ \mathbf{A}_{[j]}^{k-1}\mathbf{b}_{[j]}\right\}, \tag{2}$$

where besides the dimensionality k of the subspace increases, the matrix $\mathbf{A}_{[j]}$ and vector $\mathbf{b}_{[j]}$ that induce the subspace are augmented by block triangular bordering. We present a two-directional Arnoldi process to efficiently generate a sequence of orthonormal bases $\mathbf{Q}_k^{[j]}$ of the Krylov subspaces. The concept of a two-directional Krylov subspace and an Arnoldi process is triggered by the need of a multiparameter moment-matching based model order reduction technique for parameterized linear dynamical systems. Numerical examples illustrate computational efficiency and flexibility of the proposed two-directional Arnoldi process.

* Corresponding author.
E-mail addresses: ytli@math.ucdavis.edu (Y.-T. Li), bai@cs.ucdavis.edu (Z. Bai), yfsu@fudan.edu.cn (Y. Su).

0377-0427/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2008.05.059

The rest of this paper is organized as follows. In Section 2, we first review the Arnoldi decomposition and process associated with the Krylov subspace $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$, and then introduce a two-directional Krylov subspace and Arnoldi decomposition. In Section 3, we derive a two-directional Arnoldi process for orthonormal bases of the Krylov subspaces and the corresponding Arnoldi decompositions. In Section 4, we discuss an origin of the concept of the two-directional Krylov subspace and its application. Numerical examples and concluding remarks are in Sections 5 and 6, respectively.

Throughout the paper, we follow the notational convention used in matrix computation literature. Specifically, boldface letters denote vectors (lower cases) and matrices (upper cases), $\mathbf{I}$ is the identity matrix, $\mathbf{e}_i$ is the ith column of the identity matrix $\mathbf{I}$, $\mathbf{0}$ denotes zero vectors or matrices. The dimensions of these vectors and matrices are conformed with dimensions used in the context. $\cdot^{\mathrm{T}}$ denotes the transpose. $\mathrm{span}\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\}$ and $\mathrm{span}\{\mathbf{Q}_k\}$ denote the subspace spanned by the sequence $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k$ and the columns of the matrix $\mathbf{Q}_k$, respectively. $\|\cdot\|_1$ and $\|\cdot\|$ designate the 1-norm and 2-norm, respectively. $\mathbf{x}(i:j)$ denotes the ith to jth entries of the vector $\mathbf{x}$. $\mathbf{A}(i:j, k:\ell)$ consists of elements in the ith through jth rows and kth through ℓth columns of the matrix $\mathbf{A}$. $\mathbf{A}(:, \ell)$ denotes the ℓth column of the matrix $\mathbf{A}$.

2. Arnoldi decompositions

The Arnoldi process [1] is an algorithm for computing an orthonormal basis $\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\}$ of the kth Krylov subspace $\mathcal{K}_k(\mathbf{A}, \mathbf{q}_1)$, namely,

$$\mathrm{span}\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\} = \mathcal{K}_k(\mathbf{A}, \mathbf{b}). \tag{3}$$

Let

$$\mathbf{Q}_k = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_k \end{bmatrix},$$

then the Arnoldi process can be summarized by the following governing equation:

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_k + h_{k+1,k}\mathbf{q}_{k+1}\mathbf{e}_k^{\mathrm{T}}, \tag{4}$$

where $\mathbf{H}_k$ is a $k \times k$ unreduced upper Hessenberg matrix. If we denote

$$\mathbf{Q}_{k+1} = \begin{bmatrix} \mathbf{Q}_k & \mathbf{q}_{k+1} \end{bmatrix} \quad \text{and} \quad \widehat{\mathbf{H}}_k = \begin{bmatrix} \mathbf{H}_k \\ h_{k+1,k}\mathbf{e}_k^{\mathrm{T}} \end{bmatrix},$$

then the Eq. (4) can be recast as the following compact form:

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\widehat{\mathbf{H}}_k. \tag{5}$$

The decomposition (4) or (5) is referred to as an *order-k Arnoldi decomposition* induced by $\mathbf{A}$ and $\mathbf{b}$. The following is a pseudo-code of the Arnoldi process.

Algorithm 1. Arnoldi process

$[\mathbf{Q}_{k+1}, \widehat{\mathbf{H}}_k] = \textsc{Arnoldi}(\mathbf{A}, \mathbf{b}, k)$
(1) $\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|$
(2) for $j = 1, 2, \ldots, k$
(3) $\quad \mathbf{v}_j = \mathbf{A}\mathbf{q}_j$
(4) $\quad$ for $i = 1, \ldots, j$
(5) $\quad\quad h_{ij} = \mathbf{q}_i^{\mathrm{T}}\mathbf{v}_j$
(6) $\quad\quad \mathbf{v}_j := \mathbf{v}_j - h_{ij}\,\mathbf{q}_i$
(7) $\quad$ end for i
(8) $\quad h_{j+1,j} = \|\mathbf{v}_j\|$
(9) $\quad$ If $h_{j+1,j} = 0$, break
(10) $\quad \mathbf{q}_{j+1} = \mathbf{v}_j/h_{j+1,j}$
(11) end for j.

Note that when $h_{j+1,j} = 0$ for some j (line 8), the Arnoldi process breaks down. This is a pleasant but unlikely possibility since it happens if and only if $\mathcal{K}_j(\mathbf{A}, \mathbf{b})$ is an invariant subspace of $\mathbf{A}$. In this case, $\{\mathbf{q}_1, \ldots, \mathbf{q}_j\}$ form an orthonormal basis of $\mathcal{K}_j(\mathbf{A}, \mathbf{b})$.

Algorithm 1 is known as an implementation of the Arnoldi process in the modified Gram–Schmidt (MGS) orthogonalization form. In the presence of finite precision arithmetic, the MGS-based implementation is numerically more accurate than the mathematically equivalent classical Gram–Schmidt (CGS)-based implementation; for examples see [4,22]. There is also an implementation based on the Householder transformation [26]. It is numerically more accurate than the MGS-based implementation but doubles the number of floating points operations.

We regard the Krylov subspace (3) and the corresponding Arnoldi decomposition (4) as a *one-directional* subspace and decomposition, where the matrix $\mathbf{A}$ and vector $\mathbf{b}$ are fixed; only the dimensionality k of the subspace increases. We now

consider a situation where the underlying matrix and vector and the dimensionality of the subspace are varied. Specifically, we consider a (j, k)th *Krylov subspace* defined as

$$\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]}) = \operatorname{span}\left\{\mathbf{b}_{[j]},\ \mathbf{A}_{[j]}\mathbf{b}_{[j]},\ \mathbf{A}_{[j]}^2\mathbf{b}_{[j]},\ \ldots,\ \mathbf{A}_{[j]}^{k-1}\mathbf{b}_{[j]}\right\}, \tag{6}$$

where $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$ are given by the following block triangular recursion:

$$\mathbf{A}_{[j]} = \begin{bmatrix} \mathbf{A}_{[j-1]} & \mathbf{0} \\ \mathbf{A}_{[j,:]} & \mathbf{A}_j \end{bmatrix}, \qquad \mathbf{b}_{[j]} = \begin{bmatrix} \mathbf{b}_{[j-1]} \\ \mathbf{b}_j \end{bmatrix} \tag{7}$$

with the initials $\mathbf{A}_{[1]} = \mathbf{A}_1$ and $\mathbf{b}_{[1]} = \mathbf{b}_1$. We assume that the diagonal submatrix $\mathbf{A}_j$ is a square matrix of the order n_j and $\mathbf{b}_j$ is a column vector of n_j elements. Consequently, the off-diagonal submatrix $\mathbf{A}_{[j,:]}$ is an $n_j \times n_{[j-1]}$ rectangular matrix, where $n_{[j-1]} = n_1 + n_2 + \cdots + n_{j-1}$. In Section 4, we will see that a sequence of matrices and vectors of the triangular block recursion (7) arises from a model order reduction technique for parameterized linear dynamical systems.

When the index j is fixed, by (4), an order-k Arnoldi decomposition based on $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$ is given by

$$\mathbf{A}_{[j]}\mathbf{Q}_k^{[j]} = \mathbf{Q}_k^{[j]}\mathbf{H}_k^{[j]} + h_{k+1,k}^{[j]}\mathbf{q}_{k+1}^{[j]}\mathbf{e}_k^{\mathrm{T}} \tag{8}$$

$$= \mathbf{Q}_{k+1}^{[j]}\widehat{\mathbf{H}}_k^{[j]}, \tag{9}$$

where $\mathbf{Q}_k^{[j]}$ is an orthonormal basis of the (j, k)th Krylov subspace $\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$, and

$$\mathbf{Q}_{k+1}^{[j]} = \begin{bmatrix} \mathbf{Q}_k^{[j]} & \mathbf{q}_{k+1}^{[j]} \end{bmatrix} \quad \text{and} \quad \widehat{\mathbf{H}}_k^{[j]} = \begin{bmatrix} \mathbf{H}_k^{[j]} \\ h_{k+1,k}^{[j]}\mathbf{e}_k^{\mathrm{T}} \end{bmatrix}.$$

The decomposition (9) can also vary with the index j, say from an order-k Arnoldi decomposition induced by $\mathbf{A}_{[j-1]}$ and $\mathbf{b}_{[j-1]}$ to an order-k Arnoldi decomposition induced by $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$. Therefore, we call the decomposition (8) or (9) an *order-(j, k) Arnoldi decomposition*.

Let $\mathbf{Q}_k^{[j-1]}$ and $\mathbf{Q}_k^{[j]}$ be orthonormal bases of the $(j-1, k)$th and (j, k)th Krylov subspaces $\mathcal{K}_k(\mathbf{A}_{[j-1]}, \mathbf{b}_{[j-1]})$ and $\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$, respectively. The following theorem characterizes the relationship between the orthonormal bases $\mathbf{Q}_k^{[j-1]}$ and $\mathbf{Q}_k^{[j]}$.

Theorem 1. *Let $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$ be recursively defined in (7). Then the orthonormal bases $\mathbf{Q}_k^{[j-1]}$ and $\mathbf{Q}_k^{[j]}$ satisfy the relation*

$$\mathbf{Q}_k^{[j]} = \begin{bmatrix} \mathbf{Q}_k^{[j-1]}\mathbf{R} \\ \mathbf{L} \end{bmatrix}, \tag{10}$$

where $\mathbf{R}$ is a $k \times k$ nonsingular upper triangular matrix and $\mathbf{L}$ is an $n_j \times k$ matrix.

Proof. By the recursion (7), for any integer $\ell \geq 1$, the Krylov vectors $(\mathbf{A}_{[j]})^\ell\mathbf{b}_{[j]}$ and $(\mathbf{A}_{[j-1]})^\ell\mathbf{b}_{[j-1]}$ satisfy

$$(\mathbf{A}_{[j]})^\ell\mathbf{b}_{[j]} = \begin{bmatrix} (\mathbf{A}_{[j-1]})^\ell\mathbf{b}_{[j-1]} \\ \mathbf{w}_\ell \end{bmatrix}, \tag{11}$$

where $\mathbf{w}_\ell$ is a vector of n_j elements.

By the Arnoldi decomposition (9), we know that the Krylov vector $(\mathbf{A}_{[j]})^\ell\mathbf{b}_{[j]}$ can be expressed as a linear combination of the orthonormal basis $\mathbf{Q}_\ell^{[j]}$. Namely, we have the QR decomposition of the Krylov matrix

$$\begin{bmatrix} \mathbf{b}_{[j]} & \mathbf{A}_{[j]}\mathbf{b}_{[j]} & \cdots & (\mathbf{A}_{[j]})^{k-1}\mathbf{b}_{[j]} \end{bmatrix} = \mathbf{Q}_k^{[j]}\mathbf{R}_1, \tag{12}$$

where $\mathbf{R}_1 \in \mathbb{R}^{k\times k}$ is nonsingular upper triangular.

Combining the expressions (11) and (12), we have

$$\mathbf{Q}_k^{[j]}\mathbf{R}_1 = \begin{bmatrix} \mathbf{Q}_k^{[j-1]}\mathbf{R}_2 \\ \mathbf{W} \end{bmatrix}$$

where $\mathbf{W}$ is an $n_j \times k$ matrix, and $\mathbf{R}_1$ and $\mathbf{R}_2$ are $k \times k$ nonsingular upper triangular matrices. The theorem is proved by taking $\mathbf{R} = \mathbf{R}_2\mathbf{R}_1^{-1}$ and $\mathbf{L} = \mathbf{W}\mathbf{R}_1^{-1}$. □

In the next section, we derive a computational procedure to compute the matrices $\mathbf{R}$ and $\mathbf{L}$. In Section 4, we will show that the relationship (10) can be used to characterize orthonormal bases of a required projection subspace for parametric model order reduction. In fact, it is the matrix $\mathbf{L}$ that is exactly needed for the application.

3. Two-directional Arnoldi process

In this section we derive a computational procedure, referred to as a *two-directional Arnoldi process*, to compute the Arnoldi decomposition (9). Specifically, assume we have an order-$(j-1,k)$ Arnoldi decomposition induced by $\mathbf{A}_{[j-1]}$ and $\mathbf{b}_{[j-1]}$:

$$\mathbf{A}_{[j-1]}\mathbf{Q}_k^{[j-1]} = \mathbf{Q}_{k+1}^{[j-1]}\widehat{\mathbf{H}}_k^{[j-1]}, \tag{13}$$

where $\mathbf{q}_1^{[j-1]} = \mathbf{b}_{[j-1]}/\|\mathbf{b}_{[j-1]}\|$. We want to compute an order-(j,k) Arnoldi decomposition induced by $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$:

$$\mathbf{A}_{[j]}\mathbf{Q}_k^{[j]} = \mathbf{Q}_{k+1}^{[j]}\widehat{\mathbf{H}}_k^{[j]}, \tag{14}$$

where $\mathbf{q}_1^{[j]} = \mathbf{b}_{[j]}/\|\mathbf{b}_{[j]}\|$, and the orthonormal basis matrix $\mathbf{Q}_{k+1}^{[j]}$ is of the form

$$\mathbf{Q}_{k+1}^{[j]} = \begin{bmatrix} \mathbf{Q}_{k+1}^{[j-1]}\mathbf{R} \\ \mathbf{L} \end{bmatrix}, \tag{15}$$

where $\mathbf{R}$ is $(k+1)\times(k+1)$ upper triangular and $\mathbf{L}$ is $n_j\times(k+1)$.

Let us begin with a proper normalization of the initial vector $\mathbf{b}_{[j]}$ to get the first column $\mathbf{q}_1^{[j]}$ of $\mathbf{Q}_{k+1}^{[j]}$. Let $\gamma_j = \|\mathbf{b}_{[j]}\|$, then $\mathbf{q}_1^{[j]}$ is given by

$$\mathbf{q}_1^{[j]} = \frac{1}{\gamma_j}\mathbf{b}_{[j]} = \frac{1}{\gamma_j}\begin{bmatrix} \mathbf{b}_{[j-1]} \\ \mathbf{b}_j \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1^{[j-1]}\gamma_{j-1}/\gamma_j \\ \mathbf{b}_j/\gamma_j \end{bmatrix}, \tag{16}$$

where for the last equality, we use the facts $\mathbf{b}_{[j-1]} = \gamma_{j-1}\mathbf{q}_1^{[j-1]}$ and $\gamma_{j-1} = \|\mathbf{b}_{[j-1]}\|$ in the order-$(j-1,k)$ decomposition (13). Hence, let

$$\mathbf{r}_1 = \frac{\gamma_{j-1}}{\gamma_j} \quad \text{and} \quad \mathbf{l}_1 = \frac{\mathbf{b}_j}{\gamma_j}, \tag{17}$$

then the vector $\mathbf{q}_1^{[j]}$ has the desired form (15):

$$\mathbf{q}_1^{[j]} = \begin{bmatrix} \mathbf{q}_1^{[j-1]}\mathbf{r}_1 \\ \mathbf{l}_1 \end{bmatrix}, \tag{18}$$

where $\mathbf{r}_1 = \mathbf{R}(1:1,1)$ is the first entry of the first column of the upper triangular matrix $\mathbf{R}$ and $\mathbf{l}_1 = \mathbf{L}(:,1)$ is the first column of the matrix $\mathbf{L}$. Note that if let

$$\tau_j = \frac{\gamma_j}{\gamma_{j-1}},$$

then by (16) it is easy to see

$$\tau_j = \left\| \begin{bmatrix} (\mathbf{q}_1^{[j-1]})^{\mathrm{T}} & \left(\dfrac{\mathbf{b}_j}{\gamma_{j-1}}\right)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \right\| = \left[1 + \left(\frac{\|\mathbf{b}_j\|}{\gamma_{j-1}}\right)^2\right]^{1/2}.$$

Therefore, the scaling factor γ_j can be computed recursively:

$$\gamma_j = \gamma_{j-1}\,\tau_j.$$

In general, at the ith step, we have computed orthonormal basis vectors $\mathbf{q}_\ell^{[j]}$ of the forms

$$\mathbf{q}_\ell^{[j]} = \begin{bmatrix} \mathbf{Q}_\ell^{[j-1]}\mathbf{r}_\ell \\ \mathbf{l}_\ell \end{bmatrix}, \tag{19}$$

where $\mathbf{r}_\ell = \mathbf{R}(1:\ell;\ell)$ and $\mathbf{l}_\ell = \mathbf{L}(:,\ell)$ for $\ell = 1,2,\ldots,i$. Following the ith column of the order-(j,k) decomposition (14):

$$\mathbf{A}_{[j]}\mathbf{q}_i^{[j]} = \mathbf{q}_1^{[j]}h_{1i} + \mathbf{q}_2^{[j]}h_{2i} + \cdots + \mathbf{q}_i^{[j]}h_{ii} + \mathbf{q}_{i+1}^{[j]}h_{i+1,i}, \tag{20}$$

our task is to compute the coefficients $h_{\ell i}$ such that the vector $\mathbf{q}_{i+1}^{[j]}$ is orthonormal to $\mathbf{q}_\ell^{[j]}$ and is of the form

$$\mathbf{q}_{i+1}^{[j]} = \begin{bmatrix} \mathbf{Q}_{i+1}^{[j-1]}\mathbf{r}_{i+1} \\ \mathbf{l}_{i+1} \end{bmatrix} \tag{21}$$

for some vectors $\mathbf{r}_{i+1}$ and $\mathbf{l}_{i+1}$.

First, by the order-$(j-1,k)$ decomposition (13), the matrix-vector product $\mathbf{A}_{[j]}\mathbf{q}_i^{[j]}$ in the left-hand side of Eq. (20) has the form

$$\begin{aligned}\mathbf{A}_{[j]}\mathbf{q}_i^{[j]} &= \begin{bmatrix}\mathbf{A}_{[j-1]}\mathbf{Q}_i^{[j-1]}\mathbf{r}_i\\ \mathbf{A}_{[j,:]}\mathbf{Q}_i^{[j-1]}\mathbf{r}_i+\mathbf{A}_j\mathbf{l}_i\end{bmatrix} = \begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\widehat{\mathbf{H}}_i^{[j-1]}\mathbf{r}_i\\ \mathbf{A}_{[j,:]}\mathbf{Q}_i^{[j-1]}\mathbf{r}_i+\mathbf{A}_j\mathbf{l}_i\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{A}_{[j,:]}\mathbf{Q}_i^{[j-1]}\mathbf{r}_i+\mathbf{A}_j\mathbf{l}_i\end{bmatrix} \equiv \begin{bmatrix}\mathbf{v}_t\\ \mathbf{v}_b\end{bmatrix} = \mathbf{v},\end{aligned}$$

where

$$\mathbf{x}_t = \widehat{\mathbf{H}}_i^{[j-1]}\mathbf{r}_i.$$

Premultiplying Eq. (20) by $(\mathbf{q}_1^{[j]})^{\mathrm{T}}$ and using the orthogonality condition of the vectors $\{\mathbf{q}_\ell^{[j]}\}$ for $\ell \le i$, we have

$$h_{1i} = (\mathbf{q}_1^{[j]})^{\mathrm{T}}\mathbf{v} = \begin{bmatrix}\mathbf{q}_1^{[j-1]}\mathbf{r}_1\\ \mathbf{l}_1\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\mathbf{v}_t\\ \mathbf{v}_b\end{bmatrix} = \mathbf{r}_1(\mathbf{q}_1^{[j-1]})^{\mathrm{T}}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t + \mathbf{l}_1^{\mathrm{T}}\mathbf{v}_b = \mathbf{r}_1\mathbf{e}_1^{\mathrm{T}}\mathbf{x}_t + \mathbf{l}_1^{\mathrm{T}}\mathbf{v}_b,$$

where for the last equality, it uses the assumption of $\mathbf{Q}_{i+1}^{[j-1]}$ being an orthogonal matrix.

Subtracting $\mathbf{q}_1^{[j]}\,h_{1i} = \mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\mathbf{v}$ from Eq. (20) we get

$$\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \mathbf{q}_2^{[j]}\,h_{2i}+\cdots+\mathbf{q}_i^{[j]}\,h_{ii}+\mathbf{q}_{i+1}^{[j]}\,h_{i+1,i}. \tag{22}$$

By writing the left-hand side of (22) in the partitioned form

$$\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \begin{bmatrix}\mathbf{v}_t\\ \mathbf{v}_b\end{bmatrix} - \begin{bmatrix}\mathbf{q}_1^{[j-1]}\mathbf{r}_1\\ \mathbf{l}_1\end{bmatrix}h_{1i} = \begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{v}_b\end{bmatrix} - \begin{bmatrix}\mathbf{q}_1^{[j-1]}\mathbf{r}_1h_{1i}\\ \mathbf{l}_1h_{1i}\end{bmatrix} \tag{23}$$

and updating the first entry of $\mathbf{x}_t$ and the vector $\mathbf{v}_b$ by

$$\mathbf{x}_t(1) := \mathbf{x}_t(1) - \mathbf{r}_1h_{1i} \quad\text{and}\quad \mathbf{v}_b := \mathbf{v}_b - \mathbf{l}_1h_{1i},$$

the left-hand side of (22) is recast as

$$\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{v}_b\end{bmatrix}. \tag{24}$$

To continue the process, premultiply (24) by $(\mathbf{q}_2^{[j]})^{\mathrm{T}}$ to get

$$\begin{aligned}h_{2i} &= (\mathbf{q}_2^{[j]})^{\mathrm{T}}\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \begin{bmatrix}\mathbf{Q}_2^{[j-1]}\mathbf{r}_2\\ \mathbf{l}_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{v}_b\end{bmatrix}\\ &= \mathbf{r}_2^{\mathrm{T}}(\mathbf{Q}_2^{[j-1]})^{\mathrm{T}}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t + \mathbf{l}_2^{\mathrm{T}}\mathbf{v}_b = \mathbf{r}_2^{\mathrm{T}}\begin{bmatrix}\mathbf{I}_2 & \mathbf{0}\end{bmatrix}\mathbf{x}_t + \mathbf{l}_2^{\mathrm{T}}\mathbf{v}_b = \mathbf{r}_2^{\mathrm{T}}\mathbf{x}_t(1:2) + \mathbf{l}_2^{\mathrm{T}}\mathbf{v}_b.\end{aligned}$$

Subtracting $\mathbf{q}_2^{[j]}\,h_{2i}$ from (22), we get

$$\left(\mathbf{I}-\mathbf{q}_2^{[j]}(\mathbf{q}_2^{[j]})^{\mathrm{T}}\right)\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \mathbf{q}_3^{[j]}\,h_{3i}+\cdots+\mathbf{q}_i^{[j]}\,h_{ii}+\mathbf{q}_{i+1}^{[j]}\,h_{i+1,i}. \tag{25}$$

Again, after updating the first two entries of $\mathbf{x}_t$ and $\mathbf{v}_b$ by

$$\mathbf{x}_t(1:2) := \mathbf{x}_t(1:2) - \mathbf{r}_2h_{2i} \quad\text{and}\quad \mathbf{v}_b := \mathbf{v}_b - \mathbf{l}_2\,h_{2i},$$

the left-hand side of (25) can be recast as

$$\left(\mathbf{I}-\mathbf{q}_2^{[j]}(\mathbf{q}_2^{[j]})^{\mathrm{T}}\right)\left(\mathbf{I}-\mathbf{q}_1^{[j]}(\mathbf{q}_1^{[j]})^{\mathrm{T}}\right)\mathbf{v} = \begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{v}_b\end{bmatrix}.$$

This process can be continued to compute the coefficients $h_{3i}, h_{4i}, \ldots, h_{ii}$ until all that is left is the term $\mathbf{q}_{i+1}^{[j]}\,h_{i+1,i}$, and we have

$$\begin{bmatrix}\mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t\\ \mathbf{v}_b\end{bmatrix} = \mathbf{q}_{i+1}^{[j]}\,h_{i+1,i}. \tag{26}$$

It is immediately seen that if we let

$$\mathbf{r}_{i+1} = \frac{1}{h_{i+1,i}}\mathbf{x}_t, \qquad \mathbf{l}_{i+1} = \frac{1}{h_{i+1,i}}\mathbf{v}_b,$$

then the vector $\mathbf{q}_{i+1}^{[j]}$ has the desired form (21), where

$$h_{i+1,i} = \left\| \begin{bmatrix} \mathbf{Q}_{i+1}^{[j-1]}\mathbf{x}_t \\ \mathbf{v}_b \end{bmatrix} \right\| = \left\| \begin{bmatrix} \mathbf{x}_t \\ \mathbf{v}_b \end{bmatrix} \right\|.$$

The following pseudo-code is a complete list of the aforementioned scheme. On input, it is assumed that we have the order-$(j-1, k)$ Arnoldi decomposition (13). On output, $\mathbf{Q}_{k+1}^{[j]}$ and $\widehat{\mathbf{H}}_k^{[j]}$ satisfy the order-(j, k) Arnoldi decomposition (14). Note that for clarity of exposition, the superscript $[j-1]$ of $\mathbf{Q}_{k+1}^{[j-1]}$ and $\widehat{\mathbf{H}}_k^{[j-1]}$ is omitted.

Algorithm 2. Two-directional Arnoldi Process (TAP)

$\left[\mathbf{Q}_{k+1}^{[j]}, \widehat{\mathbf{H}}_k^{[j]}, \gamma_j\right] = \text{TAP}\left(\mathbf{A}_j, \mathbf{A}_{[j,:]}, \mathbf{b}_j, k, \mathbf{Q}_{k+1}, \widehat{\mathbf{H}}_k, \gamma_{j-1}\right)$

(1) $\tau_j = (1 + (\|\mathbf{b}_j\|/\gamma_{j-1})^2)^{1/2}$
(2) $\gamma_j = \gamma_{j-1}\,\tau_j$
(3) $\mathbf{r}_1 = 1/\tau_j$; $\mathbf{q}_1^{[j]}(1 : n_{[j-1]}) = \mathbf{Q}_1\mathbf{r}_1$
(4) $\mathbf{l}_1 = (\mathbf{b}_j/\gamma_j)$; $\mathbf{q}_1^{[j]}(n_{[j-1]} + 1 : n_{[j]}) = \mathbf{l}_1$
(5) $\widehat{\mathbf{H}}_0^{[j]} = [\,]$
(6) for $i = 1, 2, \ldots, k$
(7) $\quad \mathbf{x}_t = \widehat{\mathbf{H}}_k(1 : i+1, 1 : i)\,\mathbf{r}_i$
(8) $\quad \mathbf{v}_b := \mathbf{A}_{[j,:]}\mathbf{q}_i^{[j]}(1 : n_{[j-1]}) + \mathbf{A}_j\mathbf{l}_i$
(9) $\quad$ for $\ell = 1, 2, \ldots, i$
(10) $\quad\quad h_{\ell,i} = \mathbf{r}_\ell^{\mathrm{T}}\,\mathbf{x}_t(1 : \ell) + \mathbf{l}_\ell^{\mathrm{T}}\,\mathbf{v}_b$
(11) $\quad\quad \mathbf{x}_t(1 : \ell) := \mathbf{x}_t(1 : \ell) - h_{\ell,i}\,\mathbf{r}_\ell$
(12) $\quad\quad \mathbf{v}_b := \mathbf{v}_b - h_{\ell,i}\,\mathbf{l}_\ell$
(13) $\quad$ end for ℓ
(14) $\quad h_{i+1,i} = \left(\|\mathbf{x}_t\|^2 + \|\mathbf{v}_b\|^2\right)^{1/2}$
(15) $\quad$ If $h_{i+1,i} = 0$, break
(16) $\quad \mathbf{r}_{i+1} = \mathbf{x}_t/h_{i+1,i}$; $\mathbf{q}_{i+1}^{[j]}(1 : n_{[j-1]}) = \mathbf{Q}_{i+1}\mathbf{r}_{i+1}$
(17) $\quad \mathbf{l}_{i+1} = \mathbf{v}_b/h_{i+1,i}$; $\mathbf{q}_{i+1}^{[j]}(n_{[j-1]} + 1 : n_{[j]}) = \mathbf{l}_{i+1}$
(18) $\quad \widehat{\mathbf{H}}_i^{[j]} = \begin{bmatrix} \widehat{\mathbf{H}}_{i-1}^{[j]} & \mathbf{h}_i \\ \mathbf{0} & h_{i+1,i} \end{bmatrix}$
(19) end for i.

We note that the initial orthonormal basis matrix $\mathbf{Q}_k^{[1]}$ and the Hessenberg matrix $\widehat{\mathbf{H}}_k^{[1]}$ for the order-$(1, k)$ Arnoldi decomposition are computed by the standard Arnoldi process (Algorithm 1). The scalar γ_1 is set as $\|\mathbf{b}_{[1]}\|$. Similar to the standard Arnoldi process, the algorithm breaks down when $h_{i+1,i} = 0$ for some i (line 15). This occurs if and only if the subspace $\mathcal{K}_i(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$ is an invariant subspace of $\mathbf{A}_{[j]}$.

To end this section, we note that so far we have only considered the case where as the index j is varying, the dimensionality k of the Krylov subspaces is the same. In practice, we may want to use different dimensionality k_j. There are three possible cases: (1) $k_j < k_{j-1}$. This can be easily done by inputting the desired dimensionality k_j in Algorithm 2. On output, $\mathbf{Q}_{k_j+1}^{[j]}$ is an orthonormal basis matrix of order $n_{[j]} \times (k_j + 1)$. (2) $k_j > k_{j-1}$ and the columns of $\mathbf{Q}_{k_{j-1}}^{[j-1]}$ span an invariant subspace of $\mathbf{A}_{[j-1]}$. In this case, the desired orthonormal basis matrix $\mathbf{Q}_{k_j+1}^{[j]}$ is still of the form (15) with a $k_{j-1} \times (k_j + 1)$ upper trapezoidal matrix $\mathbf{R}$ and an $n_j \times (k_j + 1)$ matrix $\mathbf{L}$. Algorithm 2 can be modified slightly to accommodate it. The detail is omitted due to the limitation of the length of this paper. (3) $k_j > k_{j-1}$ and the columns of $\mathbf{Q}_{k_{j-1}}^{[j-1]}$ do not span an invariant subspace of $\mathbf{A}_{[j-1]}$. In this case, one cannot directly generate the desired orthonormal basis matrix $\mathbf{Q}_{k_j+1}^{[j]}$, since the required information of the matrix-vector multiplications with respect to the matrix $\mathbf{A}_{[j-1]}$ is not available.

4. Application

In this section we show how the concept of the two-directional Krylov subspace $\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$ arises from a multiparameter moment-matching based model order reduction technique of parameterized linear dynamical systems. We consider the following parameterized linear dynamical systems:

$$\begin{cases} (\mathbf{C}_0 + \lambda\mathbf{C}_1)\dot{\mathbf{x}} + (\mathbf{G}_0 + \lambda\mathbf{G}_1)\mathbf{x} = \mathbf{b}u \\ y = \mathbf{l}^{\mathrm{T}}\mathbf{x}, \end{cases} \tag{27}$$

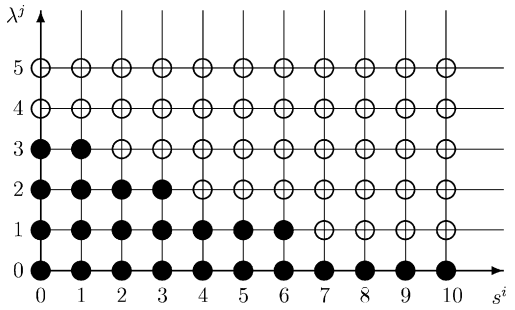

Fig. 1. $(p_0, p_1, p_2, p_3) = (10, 6, 3, 1)$ and $q = 3$.

where $\mathbf{C}_0, \mathbf{C}_1, \mathbf{G}_0$ and $\mathbf{G}_1$ are constant matrices of the order n. The parameter vector λ is a scalar and is referred to as a geometric parameter. $\mathbf{x} \in \mathbb{R}^n$ is the state vector. $u, y \in \mathbb{R}$ are the input and output functions, respectively. $\mathbf{b}, \mathbf{l} \in \mathbb{R}^n$ are input and output distribution vectors. For simplicity of presentation, we only consider the simple single geometric parameter λ. The method discussed in this section can be generalized to treat multiple parameters $\lambda = (\lambda_1, \lambda_2, \ldots)$. See [7,17] for details.

The transfer function of the system (27) is defined as the Laplace transform of the impulse response of the system:

$$h(s, \lambda) = \mathbf{l}^{\mathrm{T}}(\mathbf{G}_0 + \lambda\mathbf{G}_1 + s(\mathbf{C}_0 + \lambda\mathbf{C}_1))^{-1}\mathbf{b},$$

where $s = 2\pi f\mathtt{i}, f$ is referred to as the *frequency* and $\mathtt{i} = \sqrt{-1}$. Assuming that $\mathbf{G}_0$ is nonsingular, a series expansion of $h(s, \lambda)$ around $(s, \lambda) = (0, 0)$ is formally given by

$$h(s, \lambda) = \sum_{j=0}^{\infty}\sum_{i=0}^{\infty} m_{ij}s^i\lambda^j,$$

where $m_{ij} = \mathbf{l}^{\mathrm{T}}\mathbf{r}_i^j$ are referred to as *multiparameter moments*, and $\mathbf{r}_i^j$ are moment generating vectors satisfying the two-directional recurrence:

$$\mathbf{r}_i^j = -\mathbf{G}_0^{-1}\left(\mathbf{C}_0\mathbf{r}_{i-1}^j + \mathbf{G}_1\,\mathbf{r}_i^{j-1} + \mathbf{C}_1\,\mathbf{r}_{i-1}^{j-1}\right) \tag{28}$$

with the initial vector $\mathbf{r}_0^0 = \mathbf{G}_0^{-1}\mathbf{b}$. Note that $\mathbf{r}_i^j = \mathbf{0}$ if $i < 0$ or $j < 0$.

Let q be a prescribed approximation order of the geometric parameter λ of the transfer function, and p_j be a prescribed approximation order of frequency parameter s with respect to the geometric term λ^j, then it is necessary to match the moments m_{ij} for $j = 0, 1, 2, \ldots, q$ and $i = 0, 1, 2, \ldots, p_j$. The corresponding reduced-order system is given by

$$\begin{cases}(\widehat{\mathbf{C}}_0 + \lambda\widehat{\mathbf{C}}_1)\dot{\mathbf{z}} + (\widehat{\mathbf{G}}_0 + \lambda\widehat{\mathbf{G}}_1)\mathbf{z} = \widehat{\mathbf{b}}u\\ \widehat{y} = \widehat{\mathbf{l}}^{\mathrm{T}}\mathbf{z},\end{cases} \tag{29}$$

where $(\widehat{\mathbf{C}}_0, \widehat{\mathbf{C}}_1, \widehat{\mathbf{G}}_0, \widehat{\mathbf{G}}_1) \equiv \mathbf{V}^{\mathrm{T}}(\mathbf{C}_0, \mathbf{C}_1, \mathbf{G}_0, \mathbf{G}_1)\mathbf{V}$ and $(\widehat{\mathbf{b}}, \widehat{\mathbf{l}}) \equiv \mathbf{V}^{\mathrm{T}}(\mathbf{b}, \mathbf{l})$. $\mathbf{V}$ is an orthonormal basis of the following projection subspace:

$$\mathcal{V} = \mathrm{span}\{\mathbf{r}_0^j, \mathbf{r}_1^j, \ldots, \mathbf{r}_{p_j}^j : j = 0, 1, \ldots, q\}. \tag{30}$$

We note that, in practice, the frequency approximation orders p_j are chosen to satisfy $p_0 \geq p_1 \geq \cdots \geq p_q$. This is due to the dominant effect of the lower-order geometric terms λ^j in the approximation of the transfer function. Fig. 1 illustrates the choice of $(p_0, p_1, p_2, p_3) = (10, 6, 3, 1)$ and $q = 3$. The total of 24 matched-moments are the filled circles.

Hence the gist of the approximation of the transfer function, or the computation of a reduced-order system is to generate an orthonormal basis $\mathbf{V}$ of the projection subspace $\mathcal{V}$ defined as (30). Let us rewrite the vectors of the subspace $\mathcal{V}$ in an array as follows:

$$\mathcal{V} = \mathrm{span}\left\{\begin{matrix}\mathbf{r}_0^0, & \mathbf{r}_1^0, & \mathbf{r}_2^0, & \ldots, & \ldots & \ldots & \mathbf{r}_{p_0}^0,\\ \mathbf{r}_0^1, & \mathbf{r}_1^1, & \mathbf{r}_2^1, & \ldots, & \ldots & \mathbf{r}_{p_1}^1, & \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \\ \mathbf{r}_0^q, & \mathbf{r}_1^q, & \mathbf{r}_2^q, & \ldots, & \mathbf{r}_{p_q}^q & & \end{matrix}\right\}. \tag{31}$$

Let $\mathbf{r}_{[i]}^{[j]}$ be the vector by stacking the vectors of the first j rows in the ith column of the array (31), i.e.,

$$\mathbf{r}_{[i]}^{[j]} = \left[\begin{array}{c}\mathbf{r}_{i-1}^0\\ \mathbf{r}_{i-1}^1\\ \vdots\\ \hline \mathbf{r}_{i-1}^{j-1}\end{array}\right] \equiv \left[\begin{array}{c}\mathbf{r}_{[i]}^{[j-1]}\\ \hline \mathbf{r}_{i-1}^{j-1}\end{array}\right]$$

with the initial $\mathbf{r}_{[i]}^{[1]} = \mathbf{r}_{i-1}^{0}$. Then by the recurrence (28), we see that, when $j = 1$, the vectors $\mathbf{r}_{[i]}^{[1]}$ satisfy the linear recurrence

$$\mathbf{r}_{[i]}^{[1]} = \mathbf{A}_{[1]}\mathbf{r}_{[i-1]}^{[1]},$$

where

$$\mathbf{A}_{[1]} = -\mathbf{G}_0^{-1}\mathbf{C}_0 \quad \text{and} \quad \mathbf{r}_{[1]}^{[1]} = \mathbf{r}_0^0 = \mathbf{b}_{[1]}.$$

Hence the first row vectors of the array (31) span a $(1, k_1)$th Krylov subspace:

$$\operatorname{span}\left\{\mathbf{r}_{[1]}^{[1]}, \mathbf{r}_{[2]}^{[1]}, \ldots, \mathbf{r}_{[k_1]}^{[1]}\right\} = \operatorname{span}\left\{\mathbf{r}_0^0, \mathbf{r}_1^0, \ldots, \mathbf{r}_{p_0}^0\right\} = \mathcal{K}_{k_1}(\mathbf{A}_{[1]}, \mathbf{b}_{[1]}). \tag{32}$$

In general, it can be shown that the first j row vectors $\mathbf{r}_{[i]}^{[j]}$ satisfy the linear recurrence:

$$\mathbf{r}_{[i]}^{[j]} = \mathbf{A}_{[j]}\mathbf{r}_{[i-1]}^{[j]},$$

where the matrix $\mathbf{A}_{[j]}$ and the initial vector $\mathbf{r}_{[1]}^{[j]}$ are the bordered matrix and vector of $\mathbf{A}_{[j-1]}$ and $\mathbf{b}_{[j-1]}$:

$$\mathbf{A}_{[j]} = \begin{bmatrix} \mathbf{A}_{[j-1]} & \mathbf{0} \\ \mathbf{A}_{[j,:]} & \mathbf{A}_j \end{bmatrix}, \qquad \mathbf{r}_{[1]}^{[j]} = \mathbf{b}_{[j]} = \begin{bmatrix} \mathbf{b}_{[j-1]} \\ \mathbf{b}_j \end{bmatrix}, \tag{33}$$

and

$$\begin{aligned} &\mathbf{A}_j = -\mathbf{G}_0^{-1}\mathbf{C}_0, \\ &\mathbf{A}_{[j,:]} = -\mathbf{G}_0^{-1}\left(\begin{bmatrix}\mathbf{0} & \mathbf{G}_1\end{bmatrix}\mathbf{A}_{[j-1]} + \begin{bmatrix}\mathbf{0} & \mathbf{C}_1\end{bmatrix}\right), \\ &\mathbf{b}_j = \mathbf{r}_0^{j-1} = -\mathbf{G}_0^{-1}\begin{bmatrix}\mathbf{0} & \mathbf{G}_1\end{bmatrix}\mathbf{b}_{[j-1]}. \end{aligned}$$

Hence, the vectors $\mathbf{r}_{[i]}^{[j]}$ are Krylov vectors of the (j, k_j)th Krylov subspace:

$$\operatorname{span}\left\{\mathbf{r}_{[1]}^{[j]}, \mathbf{r}_{[2]}^{[j]}, \ldots, \mathbf{r}_{[k_j]}^{[j]}\right\} = \mathcal{K}_{k_j}(\mathbf{A}_{[j]}, \mathbf{b}_{[j]}). \tag{34}$$

By the expression (32), an orthonormal basis $\mathbf{V}_1$ of the subspace spanned by the vectors in the first row of $\mathcal{V}$ can be generated by the standard AP (Algorithm 1) with $\mathbf{A}_{[1]}$ and $\mathbf{b}_{[1]}$. By (34), we can recursively apply the TAP (Algorithm 2) with $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$ to compute an orthonormal basis $\mathbf{Q}_{k_j}^{[j]}$ of the (j, k_j)th Krylov subspace $\mathcal{K}_{k_j}(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$:

$$\mathbf{Q}_{k_j}^{[j]} = \begin{bmatrix} \mathbf{Q}_{k_j}^{[j-1]}\mathbf{R}_{[j]} \\ \mathbf{L}_{[j]} \end{bmatrix}, \tag{35}$$

where $\mathbf{L}_{[j]}$ is $n \times k_j$. Furthermore, $\mathbf{L}_{[j]}$ is a basis matrix of the subspace spanned by the vectors in the jth row of $\mathcal{V}$,

$$\operatorname{span}\{\mathbf{L}_{[j]}\} = \operatorname{span}\{\mathbf{r}_0^{j-1}, \mathbf{r}_1^{j-1}, \ldots, \mathbf{r}_{p_{j-1}}^{j-1}\}.$$

As a result, an orthonormal basis $\mathbf{V}$ of $\mathcal{V}$ is given by

$$\mathbf{V} = \operatorname{orth}\left(\begin{bmatrix}\mathbf{V}_1 & \mathbf{L}_{[2]} & \cdots & \mathbf{L}_{[q+1]}\end{bmatrix}\right), \tag{36}$$

where $\operatorname{orth}(\mathbf{X})$ denotes an orthonormal basis for the range of $\mathbf{X}$.

There is an alternative way to compute the desired orthonormal basis of $\mathcal{V}$. If the approximation order q for the geometric parameter λ is fixed, we can apply the AP (Algorithm 1) with $\mathbf{A}_{[q+1]}$ and $\mathbf{b}_{[q+1]}$ to obtain an orthonormal basis $\widetilde{\mathbf{Q}}_{k_1}^{[q+1]}$ of Krylov subspace $\mathcal{K}_{k_1}(\mathbf{A}_{[q+1]}, \mathbf{b}_{[q+1]})$, where $k_1 = p_0 + 1$. Let $\widetilde{\mathbf{Q}}_{k_1}^{[q+1]}$ be partitioned into $q + 1$ blocks:

$$\widetilde{\mathbf{Q}}_{k_1}^{[q+1]} = \begin{bmatrix} \widetilde{\mathbf{Q}}_1 \\ \widetilde{\mathbf{Q}}_2 \\ \vdots \\ \widetilde{\mathbf{Q}}_{q+1} \end{bmatrix}, \tag{37}$$

then an orthonormal basis $\widetilde{\mathbf{V}}$ of $\mathcal{V}$ is given by

$$\widetilde{\mathbf{V}} = \operatorname{orth}\left(\begin{bmatrix}\widetilde{\mathbf{Q}}_1 & \widetilde{\mathbf{Q}}_2(:, 1:k_2) & \cdots & \widetilde{\mathbf{Q}}_{q+1}(:, 1:k_{q+1})\end{bmatrix}\right). \tag{38}$$

To end this section, we note that the difference in terms of the number of floating point operations (flops) of the TAP-based and AP-based methods are on computing $\mathbf{L}_{[j]}$ in (36) and $\widetilde{\mathbf{Q}}_j$ in (38). By a straightforward calculation, we can derive that the number of floating point operations (flops) of the TAP-based method is $f_t = (2p_0^2 + \sum_{j=1}^{q}(2+j)p_j^2)n$, plus the number

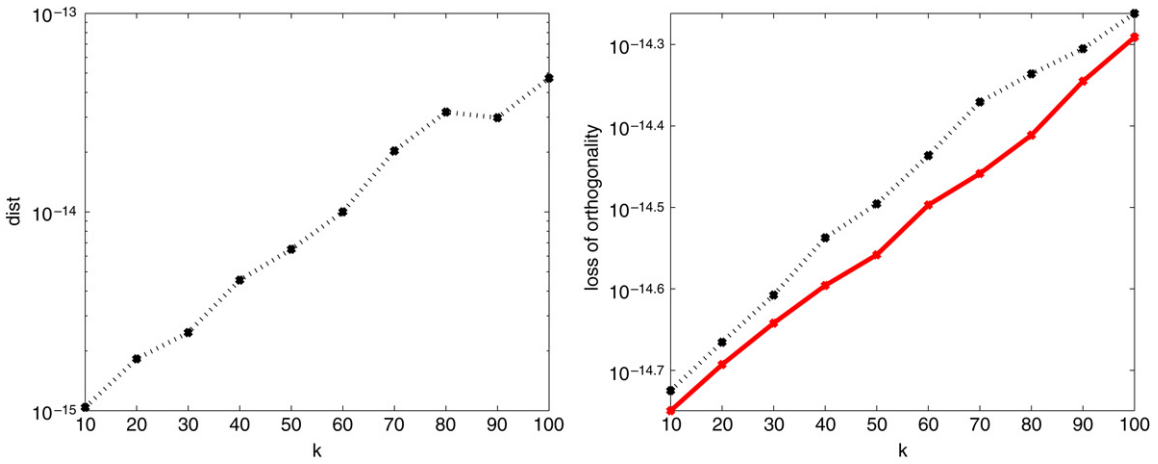

Fig. 2. The principal angle between subspaces spanned by $\mathbf{Q}_k^{[5]}$ and $\tilde{\mathbf{Q}}_k^{[5]}$ (left). The loss of orthogonality of $\mathbf{Q}_k^{[5]}$ (solid line) and $\tilde{\mathbf{Q}}_k^{[5]}$ (dotted line) (right).

of flops for the required matrix-vector multiplications, which depends on the sparsity of the underlying matrices. On the other hand, the number of flops of the AP-based method needs $f_a = 2p_0^2(q+1)n$ plus the same number of the matrix-vector multiplications. If one wants to match the same number of moments with respect to all geometric terms λ^j, namely, $p_0 = p_1 = \cdots = p_q$, then the TAP-based method needs a factor of $(1 + q/4)$ more flops than the AP-based method, $f_t = (1 + q/4)f_a$. If we consider the practical application where it is sufficient to choose p_j as a monotonically decreasing sequence as we discussed in Section 4, then the TAP-based method is much more efficient than the AP-based method. For example, if $p_j = (1/2)^j p_0$, then the number of flops of the TAP-based method is $f_t \leq \frac{28}{9}p_0^2 n$. Therefore, the flop cost of the TAP-based method is about a factor of q less than the cost of the AP-based method.

5. Numerical examples

In this section, we use three numerical examples to illustrate numerical properties and application of the proposed two-directional Arnoldi process (TAP) in Section 4. All numerical results are conducted under the MATLAB environment and run on a PC with 1.6 GHz Intel CoreDuo T2050 processor.

Example 1. In this example we show that the TAP (Algorithm 2) and AP (Algorithm 1) generally have similar numerical behaviors in terms of the accuracy. Consider the matrices $\mathbf{A}_{[j]}$ and vectors $\mathbf{b}_{[j]}$ recursively generated by block triangular bordering (7), where $\mathbf{A}_{[j]}$ and $\mathbf{b}_{[j]}$ are random matrices with entries chosen from a normal distribution with mean zero and variance one. Let $\mathbf{Q}_k^{[j]}$ and $\widetilde{\mathbf{Q}}_k^{[j]}$ be orthonormal bases of the (j, k)th Krylov subspace $\mathcal{K}_k(\mathbf{A}_{[j]}, \mathbf{b}_{[j]})$ generated by the TAP and AP, respectively.

The left plot of Fig. 2 shows the distance between the subspaces $\text{span}\{\mathbf{Q}_k^{[j]}\}$ and $\text{span}\{\widetilde{\mathbf{Q}}_k^{[j]}\}$ are under the order of 10^{-13} for $j = 5$ and k up to 100. The order of the submatrices $\mathbf{A}_i$ is $n_i = 200$. Recall that the distance between two subspaces $\mathcal{X}$ and $\mathcal{Y}$ of equal dimension is defined by $\text{dist}(\mathcal{X}, \mathcal{Y}) = \sin(\theta_k)$, where θ_k is the principal angle of $\mathcal{X}$ and $\mathcal{Y}$, which is the arc-cosine of the largest singular value of the matrix $\mathbf{X}^{\mathrm{T}}\mathbf{Y}$ and $\text{span}(\mathbf{X}) = \mathcal{X}$ and $\text{span}(\mathbf{Y}) = \mathcal{Y}$ [5,11]. The loss of orthogonality, measured by $\|\mathbf{I} - (\mathbf{Q}_k^{[j]})^{\mathrm{T}}\mathbf{Q}_k^{[j]}\|_1$, of computed orthonormal basis $\mathbf{Q}_k^{[j]}$ is shown in the right plot of Fig. 2 as the solid line. The dotted line is for the loss of orthogonality of $\widetilde{\mathbf{Q}}_k^{[5]}$. We observe that the two algorithms have almost the same behavior in terms of the loss of orthogonality.

Example 2. It is well known the AP (Algorithm 1) may suffer severe loss of orthogonality due to the ill-conditioning of the underlying matrix [8,12]. This example shows that the TAP behaves similarly. Let us consider the FS1836 matrix $\mathbf{A}_{[1]}$ from the Harwell–Boeing collection [6]. The order of the matrix $\mathbf{A}_{[1]}$ is $n_1 = 207$. Let

$$\mathbf{A}_{[2]} = \begin{bmatrix} \mathbf{A}_{[1]} & \mathbf{0} \\ \mathbf{A}_{[2,:]} & \mathbf{A}_2 \end{bmatrix},$$

where $\mathbf{A}_2 = \mathbf{A}_{[1]}$ and the elements of $\mathbf{A}_{[2,:]}$ are chosen from a normal distribution with mean zero and variance one. The elements of initial vectors $\mathbf{b}_{[1]}$ and $\mathbf{b}_{[2]}$ are set to be ones. Matrices $\mathbf{A}_{[1]}$ and $\mathbf{A}_{[2]}$ are very ill-conditioned, with the condition numbers $\kappa(\mathbf{A}_{[1]}) = 1.5 \times 10^{11}$ and $\kappa(\mathbf{A}_{[2]}) = 1.2155 \times 10^{13}$. The loss of orthogonality of the computed orthonormal basis $\widetilde{\mathbf{Q}}_k^{[2]}$ by the AP is shown by the dash–dotted line in Fig. 3. On the other hand, by using the TAP (Algorithm 2), we first compute $\mathbf{Q}_k^{[1]}$ and then $\mathbf{Q}_k^{[2]}$. We observe a similar behavior in terms of the loss of orthogonality of $\mathbf{Q}_k^{[2]}$, shown by the solid line in Fig. 3.

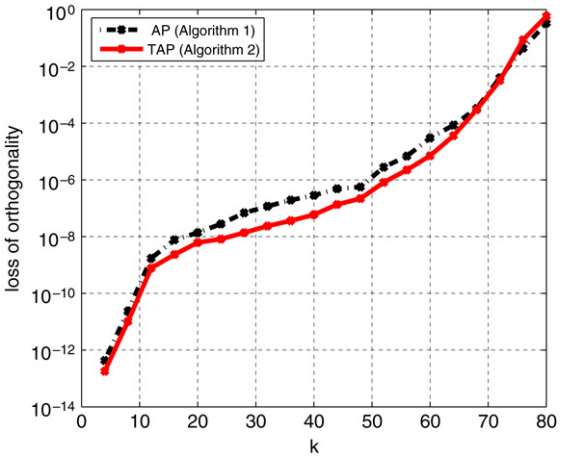

Fig. 3. The behaviors of loss of orthogonality.

The loss of orthogonality of the AP is studied in [8,12]. It is characterized in terms of the condition number of the underlying matrix. It is a subject of our future study to analyze the loss of orthogonality and present a proper reorthogonalization scheme of the proposed TAP.

Example 3. In this example, we show the application of the TAP for generating an orthonormal basis of the projection subspace for parametric model order reduction as discussed in Section 4. The matrices $\mathbf{C}_0$, $\mathbf{C}_1$, $\mathbf{G}_0$, $\mathbf{G}_1$ in the system (27) are originated from a modified nodal analysis formulation of a linear RLC subcircuit that models the circuit's interconnect and package [14,20], and are of the following forms:

$$\mathbf{C}_0 + \lambda\mathbf{C}_1 = \begin{bmatrix} (1+\lambda)\mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \end{bmatrix}, \qquad \mathbf{G}_0 + \lambda\mathbf{G}_1 = \begin{bmatrix} (1+\lambda)\mathbf{G} & \mathbf{E} \\ -\boldsymbol{E}^{\mathrm{T}} & \mathbf{0} \end{bmatrix},$$

where $\mathbf{C}$, $\mathbf{L}$ and $\mathbf{G}$ are capacitance, inductance and resistance matrices, respectively. $\mathbf{E}$ is the incident matrix associated with the connectivity of the circuit. λ represents the fabrication variation, and is subject to $\pm 10\%$ variation. The order of the full system is $n = 3298$, where the order of the submatrices $\mathbf{C}$ and $\mathbf{G}$ is 2210 and the order of $\mathbf{L}$ is 1088.

As we discussed in Section 4, the kernel of computing a reduced-order model (29) is on computing an orthonormal basis $\mathbf{V}$ of the projection subspace $\mathcal{V}$ defined in (30). To have the relative error under $O(10^{-3})$ for the approximate transfer function $\widehat{h}(s, \lambda)$ on the frequency range $[0,\ 4]$ GHz, it turns out that a TAP reduced-order model with $q = 2$, $p_j = (1/2)^j p_0$ and $p_0 = 200$ is sufficient. The orthonormal basis $\mathbf{V}$ is given by

$$\mathbf{V} = \mathrm{orth}\left(\begin{bmatrix} \mathbf{V}_1 & \mathbf{L}_{[2]} & \mathbf{L}_{[3]} \end{bmatrix}\right),$$

where the matrices $\mathbf{V}_1$, $\mathbf{L}_{[2]}$ and $\mathbf{L}_{[3]}$ are defined in (36). The total number of basis vectors is $k = k_1 + k_2 + k_3 = (p_0 + 1) + (p_1 + 1) + (p_2 + 1) = 353$. Fig. 4 shows the magnitudes of the original and approximate transfer functions $h(s, \lambda)$ and $\widehat{h}(s, \lambda)$ for $\lambda = 0.02$, and the relative errors on the frequency range $[3,\ 4]$ GHz. The two curves are visually indistinguishable. We do not show the results on the frequency range $[0,\ 3]$ GHz since the relative errors are substantially smaller than 10^{-4}. It took a total 55.91 s, including 34.99 s for computing $\mathbf{V}_1$, and 11.64 and 5.53 s for $\mathbf{L}_{[2]}$ and $\mathbf{L}_{[3]}$, respectively.

We also use the AP (Algorithm 1) to compute an orthonormal basis $\widetilde{\mathbf{V}}$ of the same projection subspace $\mathcal{V}$:

$$\widetilde{\mathbf{V}} = \mathrm{orth}\left(\begin{bmatrix} \widetilde{\mathbf{Q}}_1 & \widetilde{\mathbf{Q}}_2(:, 1:k_2) & \widetilde{\mathbf{Q}}_3(:, 1:k_3) \end{bmatrix}\right), \tag{39}$$

where the $n \times k_1$ matrices $\widetilde{\mathbf{Q}}_j$ are from the partition of an orthonormal basis $\widetilde{\mathbf{Q}}_{k_1}^{[3]}$ of the Krylov subspace $\mathcal{K}_{k_1}(\mathbf{A}_{[3]}, \mathbf{b}_{[3]})$ as defined in (37). It took a total 174.74 s to compute the basis $\widetilde{\mathbf{V}}$, including 171.12 s for computing $\widetilde{\mathbf{Q}}_{k_1}^{[3]}$.

The results of the approximate transfer function $\widetilde{h}(s, \lambda)$ computed by the AP method are also shown in Fig. 4. They reveal that the TAP model is more than an order of the magnitude accurate than the AP model. Meanwhile, the CPU elapsed time of the TAP-based method is only about one third of the AP-based method.

The evidence of the loss of the accuracy of the AP method is shown in the loss of accuracy in the computed first basis vectors $\widetilde{\mathbf{Q}}_1$. If we use the Householder transformation-based AP to compute the first basis vectors, denoted as $\mathbf{V}_{\mathrm{HO}}$, then we observe that $\mathrm{dist}(\widetilde{\mathbf{Q}}_1, \mathbf{V}_{\mathrm{HO}}) = 9.95 \times 10^{-3}$. The Householder transformation-based AP is considered as the most accurate and also most expensive implementation [26]. In contrast, for the first block vectors $\mathbf{V}_1$ computed in the TAP, we have $\mathrm{dist}(\mathbf{V}_1, \mathbf{V}_{\mathrm{HO}}) = 1.32 \times 10^{-9}$.

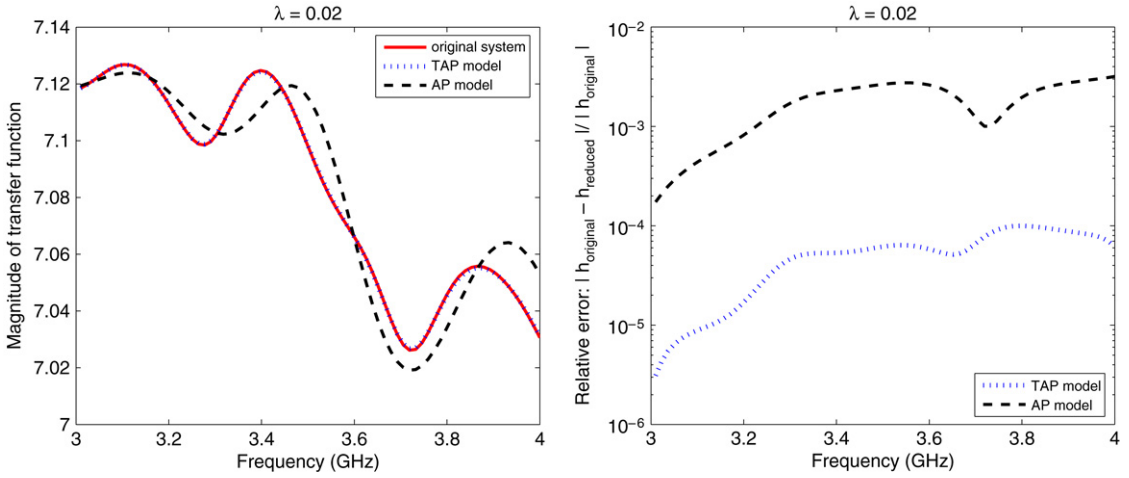

Fig. 4. The magnitude of the transfer function $h(s, \lambda)$ and approximations on the frequency range [3, 4] GHz (left). The relative errors of the approximations (right), where $\lambda = 0.02$.

Finally, we note that an additional advantage of the TAP-based method is the flexibility. The approximation order j of the geometric parameter λ and the associated approximation order p_j of the frequency parameter can adaptively be selected based on the required number of moments to be matched for achieving the desired accuracy, with only marginal increase in the computational cost. Due to the scope of this paper, a further discussion of this issue is to be presented elsewhere.

6. Concluding remarks

The concept of a two-directional Krylov subspace and Arnoldi decomposition is motivated from the problem of multiparameter moment-matching for model order reduction of parameterized dynamical systems. Numerical examples illustrate the advantages of the TAP-based method in terms of CPU efficiency and accuracy. Our further study includes the stability analysis of the two-directional Arnoldi process and reorthogonalization when necessary. The applications of the two-directional Krylov subspace and the Arnoldi decomposition to other areas, such as the two-directional dynamical systems described by the Fornasini–Marchesini model [9,10,19].

Acknowledgments

We thank the referees for valuable comments and suggestions to improve the presentation of the paper.

The first and second authors were supported in part by NSF grant DMS-0611548. The third author was supported in part by Shanghai Dawn Project 200601 and China NSF.

References

[1] W.E. Arnoldi, The principle of minimized iteration in the solution of the matrix eigenvalue problem, Quart. Appl. Math. 9 (1951) 17–29.
[2] Z. Bai, Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems, Appl. Numer. Math. 43 (2002) 9–44.
[3] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, H. van der Vorst (Eds.), Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, SIAM, Philadelphia, 2000.
[4] Å. Björck, Numerics of Gram–Schmidt orthogonalization, Linear Algebra Appl. 197–198 (1994) 297–316.
[5] Å. Björck, G.H. Golub, Numerical methods for computing angles between linear subspaces, Math. Comp. 27 (1973) 579–594.
[6] R.F. Boisvert, R. Pozo, K. Remington, R. Barrett, J.J. Dongarra, The matrix market: A web resource for test matrix collections, in: R.F. Boisvert (Ed.), Quality of Numerical Software, Assessment and Enhancement, Chapman & Hall, London, 1997, URL http://citeseer.ist.psu.edu/boisvert97matrix.html.
[7] L. Daniel, O.C. Siong, L.S. Chay, K.H. Lee, J. White, A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models, IEEE Trans. Comput-Aided Des. Integrated Circuit. Syst. 23 (5) (2004) 678–693.
[8] J. Drkošová, A. Greenbaum, M. Rozložník, Z. Strakoš, Numerical stability of GMRES, BIT 35 (1995) 309–330.
[9] E. Fornasini, G. Marchesini, Doubly-indexed dynamical systems: State-space models and structural properties, Math. Syst. Theory 12 (1978) 59–72.
[10] E. Fornasini, G. Marchesini, Stability analysis of 2-D systems, IEEE Trans. Circuit Syst. CAS-27 12 (1980) 1210–1217.
[11] G.H. Golub, C.F.V. Loan, Matrix Computations, 3rd ed., The John Hopkins University Press, Baltimore, 1996.
[12] A. Greenbaum, M. Rozložník, Z. Strakoš, Numerical behaviour of the modified Gram-Schmidt GMRES implementation, BIT 37 (3) (1997) 706–719.
[13] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Natl. Bur. Stand. 49 (6) (1952) 409–436.
[14] S.-Y. Kim, N. Gopal, L.T. Pillage, Time-domain macromodels for VLSI interconnect analysis, IEEE Trans. Comput.-Aided Des. 13 (10) (1994) 1257–1270.
[15] C. Lanczos, Solution of systems of linear equations by minimized iterations, J. Res. Natl. Bur. Stand. 49 (1) (1952) 33–53.
[16] R.-C. Li, Z. Bai, Structure-preserving model reduction using a Krylov subspace projection formulation, Commun. Math. Sci. 3 (2) (2005) 179–199.
[17] Y.-T. Li, Z. Bai, Y. Su, X. Zeng, Parameterized model order reduction via a two-directional Arnoldi process, in: Proc. of IEEE/ACM Int. Conf. Computer-Aided Design, 2007.
[18] A. Odabasioglu, M. Celik, L.T. Pileggi, PRIMA: Passive reduced-order interconnect macromodeling algorithm, IEEE Trans. Comput.-Aided Des. Integrated Circuit. Syst. 17 (8) (1998) 645–654.
[19] T. Ooba, On stability robustness of 2-D systems described by the Fornasini–Marchesini model, Multidimens. Syst. Signal Process. 12 (1) (2001) 81–88.

[20] L. Pileggi, Coping with RC(L) interconnect design headaches, in: Proc. of IEEE/ACM Int. Conf. Computer-Aided Design, 1995.
[21] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Manchester University Press, Oxford Road, Manchester, UK, 1992.
[22] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., SIAM, Philadelphia, 2003.
[23] Y. Saad, M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.
[24] P. Sonneveld, CGS, a fast Lanczos-type solver for nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 10 (1) (1989) 36–52.
[25] G.W. Stewart, Matrix Algorithms, vol. 2, SIAM, Philadelphia, 2002.
[26] H.F. Walker, Implementation of the GMRES method using householder transformations, SIAM J. Sci. Stat. Comput. 9 (1) (1988) 152–163.