Model Reduction and the Solution of Sylvester Equations (Extended Abstract)

K. A. Gallivan
Florida State University
Tallahassee, FL, USA

A. Vandendorpe and P. Van Dooren
Catholic University of Louvain
Louvain la Neuve, Belgium

December 30, 2005

1 Introduction

Model reduction techniques are often required in computationally tractable algorithms for the solution of simulation, optimization, or control problems for large scale dynamical systems. For linear systems, essentially all reduced models can be produced using projection onto subspaces determined by the approximation constraints of the problem. For example, rational interpolation, e.g., Rational Krylov methods, and its generalization to tangential interpolation require projection onto so-called generalized Krylov subspaces whose bases solve a particular family of Sylvester equations. In this paper, we derive a numerically reliable way to compute an orthogonal basis of these generalized Krylov subspaces. The residual error of the large linear systems of equations that are solved in order to produce the bases are controlled so as to yield a small backward error in the associated Sylvester equations and in the model reduction problem. The efficiency and effectiveness of the algorithm is demonstrated for single and multipoint tangential interpolation examples.

2 Subspaces and Model Reduction

Generalized Krylov subspaces appear naturally in the framework of model reduction via tangential interpolation [4, 6], which is a natural extension of the Multipoint Padé technique [3, 5] to MIMO systems. In the context of dynamical systems, these techniques provide a reduced order approximation of a given $p \times m$ rational matrix

$$T(s) := C(sE - A)^{-1}B$$

of large state dimension N by another $p \times m$ rational matrix

$$\hat{T}(s) := \hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B}$$

of smaller state dimension $n \ll N$. The system quadruples $\{E, A, B, C\}$ and $\{\hat{E}, \hat{A}, \hat{B}, \hat{C}\}$ are the so-called *generalized state-space system matrices* of the *transfer matrices* $T(s)$ and $\hat{T}(s)$, respectively. There exist several ways of constructing reduced order transfer functions, but in this paper we focus on projection methods using interpolation techniques. The *projected system matrices* $\{\hat{E}, \hat{A}, \hat{B}, \hat{C}\}$ are obtained as follows

$$\{\hat{E}, \hat{A}, \hat{B}, \hat{C}\} = \{Z^*EV, Z^*AV, Z^*B, CV\},$$

using $N \times n$ projection matrices $V,\ Z$ chosen to be orthonormal ($Z^*Z = V^*V = I_n$). Notice that for simplicity, we have assumed all matrices to be complex, since interpolation conditions will be formulated at points in the complex plane. By combining complex conjugate interpolation conditions for real systems, one can always guarantee that the projection matrices V and Z are also real.

In order to formalize the interpolation conditions, we first introduce some notation. Let us expand $(sE - A)^{-1}B$ and the $m \times 1$ polynomial vector $x(s)$ around $\alpha \in \mathbb{C}$

$$x(s) := \sum_{i=0}^{n-1} x_i (s-\alpha)^i, \quad (sE-A)^{-1}B = \sum_{i=0}^{\infty} A_\alpha^i B_\alpha (s-\alpha)^i \tag{1}$$

using the matrices

$$A_\alpha := (A - \alpha E)^{-1}E, \quad B_\alpha := (A-\alpha E)^{-1}B. \tag{2}$$

If we use the matrix

$$X := \left[\begin{array}{cccc} x_0 & x_1 & \dots & x_{n-1} \end{array} \right] \tag{3}$$

to denote the coefficients of $x(s)$, then we have the following definition.

Definition 2.1 *The generalized Krylov matrix of order k of the triple (A_α, B_α, X) is the $N \times k$ matrix defined as :*

$$\begin{aligned} & K_n(A_\alpha, B_\alpha, X) \\ & := \left[\begin{array}{cccc} B_\alpha, A_\alpha B_\alpha & \dots & A_\alpha^{n-1} B_\alpha \end{array} \right] \left[\begin{array}{ccc} x_0 & \dots & x_{n-1} \\ & \ddots & \vdots \\ & & x_0 \end{array} \right]. \end{aligned} \tag{4}$$

The corresponding generalized Krylov subspace of order k is the image of this matrix :

$$\mathcal{K}_n(A_\alpha, B_\alpha, X) := Im\left\{K_n(A_\alpha, B_\alpha, X)\right\}.$$

With the above notation, the main result of [4] can be stated as follows:

Theorem 2.1 *Let (E, A, B, C) be a generalized state space realization of $T(s)$. Assume that the matrix $V \in \mathbb{C}^{N\times k}$ is such that $\mathcal{K} \subseteq Im(V)$. If $\{\hat{E}, \hat{A}, \hat{B}, \hat{C}\} = \{Z^*EV, Z^*AV, Z^*B, CV\}$, with $Z^*Z = V^*V = I_n$, then the transfer function $\hat{T}(s) := \hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B}$ satisfies the following tangential interpolation condition :*

$$(T(s) - \hat{T}(s))x(s) = O(s-\alpha)^n,$$

i.e. the k first derivatives of $T(s)x(s)$ and $\hat{T}(s)x(s)$ at $s = \alpha$ are equal (provided α is neither a pole of $T(s)$ nor a pole of $\hat{T}(s)$).

A very interesting property of this theorem is that only the image of the generalized Krylov matrix K_n plays a role, and not the matrix K_n itself.

One can also impose similar conditions of the transposed system, which essentially amounts to imposing tangential conditions of the type

$$x(s)^*(T(s) - \hat{T}(s)) = O(s-\alpha)^n,$$

which are obtained by choosing Z such that

$$\mathcal{K}_n(A_\alpha^*, C_\alpha^*, X) \subseteq Im(Z),\ A_\alpha := E(A-\alpha E)^{-1},\ C_\alpha := C(A-\alpha E)^{-1}.$$

If we choose to interpolate at infinity, rather than at a finite point α, the expansion matrices change :

$$x(s) = \sum_{i=0}^{n-1} x_i s^{-i},\ (sE-A)^{-1}B = \sum_{i=0}^{\infty} A_\infty^i B_\infty s^{-i-1}$$

with

$$A_\infty := E^{-1}A,\ B_\infty := E^{-1}B,$$

and the above theorem still holds with the interpolation conditions

$$(T(s) - \hat{T}(s))x(s) = O(s^{-n-1}).$$

Left and right tangential interpolation conditions, and finite and infinite ones can also be combined, and they all amount to imposing conditions of the same type as in Theorem 2.1. We refer to [4] for more details.

In order to develop numerically reliable model reduction algorithms, it is therefore important to find an efficient way of computing an orthonormal basis of such generalized Krylov subspaces.

3 Relation to Sylvester equations

For ease of notation, we write in the sequel K for $K_n(A_\alpha, B_\alpha, X)$ and $\mathcal{K}$ for $\mathcal{K}_n(A_\alpha, B_\alpha, X)$. We first derive a simple way of constructing an orthonormal basis for the space $\mathcal{K}$. Let us denote the columns of the matrix K by k_j, $1 \le j \le n$. From (4), the following relations are then obtained for $j = 2, \ldots, n$:

$$\begin{aligned} k_j &= \sum_{i=1}^{j} A_\alpha^{i-1} B_\alpha x_{j-i} \\ &= B_\alpha x_{j-1} + A_\alpha \sum_{i=1}^{j-1} A_\alpha^{i-1} B_\alpha x_{j-1-i} \\ &= B_\alpha x_{j-1} + A_\alpha k_{j-1}. \end{aligned} \tag{5}$$

The following result, implicitly shown in [6] and [4], rewrites these equations in a more compact form using the Jordan block matrix $J_\alpha \in \mathbb{C}^{n\times n}$:

$$J_\alpha = \begin{bmatrix} \alpha & 1 & 0 & \ldots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \ldots & \ldots & 0 & \alpha \end{bmatrix}. \tag{6}$$

Lemma 3.1 *Let* $A_\alpha := (A - \alpha E)^{-1}E,\ B_\alpha := (A - \alpha E)^{-1}B$. *Then the Krylov matrix* $K := K_n(A_\alpha, B_\alpha, X)$ *satisfies the following Sylvester equation :*

$$AK - EKJ_\alpha - BX = 0. \tag{7}$$

Moreover, the image of any $N \times n$ *matrix* $\bar{K}$ *spanning* $\mathcal{K}$ *satisfies a transformed equation*

$$A\bar{K} - E\bar{K}\bar{J}_\alpha - \bar{B}\bar{X} = 0, \tag{8}$$

where

$$\bar{J}_\alpha = TJ_\alpha T^{-1}, \quad \bar{X} = XT^{-1}.$$

Proof. Equation (5) yields the relation $K = A_\alpha K J_0 + B_\alpha X$ from which one obtains (7) after multiplication by $(A - \alpha E)$. If $Im(\bar{K}) = Im(K)$, there exists a nonsingular matrix T such that $\bar{K}T = K$. By inserting this into (7), we obtain (8). □

The equivalent result for interpolation at infinity is derived in a similar fashion and is also shown in [6] and [4].

Lemma 3.2 *Let* $A_\infty := E^{-1}A,\ B_\infty := E^{-1}B$. *Then the Krylov matrix* $K := K_n(A_\infty, B_\infty, X)$ *satisfies the following Sylvester equation :*

$$EK - AKJ_0 - BX = 0. \tag{9}$$

Moreover, the image of any $N \times n$ *matrix* $\bar{K}$ *spanning* $\mathcal{K}$ *satisfies a transformed equation*

$$E\bar{K} - A\bar{K}\bar{J}_0 - B\bar{X} = 0, \tag{10}$$

where

$$\bar{J}_0 = TJ_0T^{-1}, \quad \bar{X} = XT^{-1}, \quad \bar{K} = KT^{-1}.$$

If one wants to solve the multi-point tangential interpolation problem the we have the following relation with Sylvester equations

Theorem 3.1 *Let* α_i *be set of interpolation points and let* X_i *be* $N \times n_i$ *matrices defining the right tangential directions at these points. Then the matrix* V *whose block columns are the generalized Krylov matrices* $K_{n_i}(A_{\alpha_i}, B_{\alpha_i}, X_i)$ *satisfies the following Sylvester equation :*

$$EVF - AVG - BX = 0, \tag{11}$$

where

$$sF - G = \text{diag}\ \{sI - J_{\alpha_i, n_i}\}$$

is a regular pencil with Jordan blocks J_{α_i,n_i} *of dimension* n_i *at* α_i. *Moreover, the image of any* $N \times n$ *matrix* $\bar{V}$ *spanning* $Im(V)$ *satisfies a transformed equation*

$$E\bar{V}\bar{F} - A\bar{V}\bar{G} - B\bar{X} = 0, \tag{12}$$

where

$$(s\bar{F} - \bar{G}) = T(sF - G)S, \quad \bar{X} = XS, \quad \bar{V} = VT^{-1}.$$

This result easily follows from the two earlier lemmas and was also proven in [6], [4]. A dual result holds for the left tangential interpolation problem, but we omit it here. Notice also that the interpolation points can be chosen at infinity. Notice also that we can choose $S = I$ if we do not insist on having $F = \bar{F} = I_n$.

The projection matrices V and Z are typically chosen to be orthonormal, even though the interpolation conditions are satisfied for any basis of $Im(V)$ and $Im(Z)$. It can be shown that the choice of orthonormal bases yields less sensitivity to errors when defining the reduced order model in terms of backward errors on the original model. This analysis will be included in the full paper.

4 Perturbed Sylvester Equations

We would like to assess how accurately the Sylvester equations are being solved and how well the computed solution represents the space in which we are interested. For this, we first focus on the interpolation at a single finite point α since the general problem is very similar in nature. In order to simplify the notation, we also drop the index α in the matrix J_α. It follows directly from the Sylvester equation (7) that an orthonormal basis for $\mathcal{K}$ can be obtained by essentially computing the columns of K and then orthonormalizing them, which is done by the following algorithm.

Algorithm 4.1 *1. Compute the first vector k_1 by solving*

$$(A - \alpha E)k_1 = Bx_0 \tag{13}$$

2. For $i = 2$ to n, compute recursively the vectors k_i by solving

$$(A - \alpha E)k_i = Bx_{i-1} + Ek_{i-1} \tag{14}$$

3. Orthonormalize the columns of K by a QR factorization

The most important part of the computational complexity resides in solving the n linear systems (13,14). For large scale systems, this can be done using iterative techniques, for instance by using a Krylov method. The problem with this algorithm is that residual numerical errors in each solution of a linear equation will lead to numerical errors that propagate to the next solutions of linear equations. This may lead to a poor approximation of the exact subspace as we now point out.

Let us assume that the residual errors in the above linear systems are given by f_i, i.e.

$$(A - \alpha E)k_1 - Bx_0 = f_1, \tag{15}$$

$$(A - \alpha E)k_i - Bx_{i-1} - Ek_{i-1} = f_i, \tag{16}$$

and that we impose the norm of these residuals to be bounded by δ.

We need to orthonormalize the computed matrix K, which is equivalent to computing a QR factorization of K. This is known to be delicate from a numerical point of view if the original matrix K is poorly conditioned [1]. This is typically the case for Krylov matrices since their subsequent columns tend to converge to the dominant eigenvector of A_α and hence get closer to each other. The above construction yields a Sylvester equation with a small matrix residual :

$$AK - EKJ - BX = F, \quad \|F\| \approx \delta. \tag{17}$$

The subsequent QR factorization yields

$$K = QR, \quad \bar{K} := Q = KR^{-1}, \quad T := R \tag{18}$$

and in the coordinate system of the orthonormalized matrix $\bar{K}$ we therefore obtain a Sylvester equation with a considerably larger residual $\bar{F} := FR^{-1}$:

$$A\bar{K} - E\bar{K}\bar{J} - B\bar{X} = \bar{F}, \quad \|\bar{F}\|_2 \approx \delta\|R^{-1}\|_2, \tag{19}$$

since the transformation matrix R is often badly conditioned.

We can interpret the error $\bar{F}$ as a backward error on the model. The easiest way to do this is to construct a backward error $\Delta_A := \bar{F}Q^*$ which yields

$$(A - \Delta_A)\bar{K} - E\bar{K}\bar{J} - B\bar{X} = 0, \quad \|\Delta_A\|_2 = \|\bar{F}\|_2, \tag{20}$$

since $Q^*\bar{K} = I_n$. This says that we performed an exact interpolation of a perturbed model, in which the perturbation Δ_A is of the norm of the residual error $\bar{F}$. In the next section, we show how to compute the successive columns of the orthonormalized matrix $\bar{K}$ directly, while yet ensuring a small residual $\bar{F}$.

5 Merging into Gram-Schmidt

The idea of the new algorithm we propose in this section is to merge the Gram-Schmidt orthogonalization of K with the recursive calculation of the columns of K. At each step of the algorithm, say for instance step i, the i-th column of Q (denoted by q_i) is computed. To this end, we (implicitly) construct matrices $K^{(i)}$, $X^{(i)}$ and $J^{(i)}$ such that

$$AK^{(i)} - K^{(i)}J^{(i)} + BX^{(i)} = 0,$$

and an upper triangular matrix $R^{(i)}$ such that

$$K^{(i)}R^{(i)} = K, \quad J^{(i)} = R^{(i)}JR^{(i)^{-1}}, \quad X^{(i)} = XR^{(i)^{-1}} \quad \textit{and}$$

$$K^{(i)} = \left[\begin{array}{ccccccc} q_1 & \ldots & q_i & k_{i+1} & \ldots & k_n \end{array} \right],$$

i.e., at each step of the algorithm, the i first columns of $K^{(i)}$ are equal to the i first columns of the unitary matrix Q and the $k-i$ last columns of $K^{(i)}$ are equal to the corresponding columns of K. Denote the j-th column of $K^{(i)}$ by $k_j^{(i)}$ and the j-th column of $X^{(i)}$ by $x_j^{(i)}$. Let $j_{s,t}^{(i)} := J^{(i)}(s,t)$.

Our objective to derive an algorithm that constructs an orthonormal basis Q of the generalized Krylov matrix K such that the residual error of the corresponding Sylvester equation is bounded by a prescribed tolerance δ:

$$\|AQ - Q(RJR^{-1}) + B(XR^{-1})\| < \delta. \tag{21}$$

We next present the algorithm in detail.

Initialization

Define $J^{(0)} := J$, $X^{(0)} := X$ and $K^{(0)} := K$. Now, compute the first column $k_1^{(0)}$ of $K^{(0)}$ that satisfies

$$AK^{(0)} - K^{(0)}J^{(0)} + BX^{(0)} = 0.$$

From the preceding definitions, the first column of $K^{(0)}$ satisfies the equation

$$(\alpha I - A)k_1^{(0)} = Bx_0. \tag{22}$$

First step

Define

$$R^{(1)} := \begin{bmatrix} \|v\| & \\ & I_{n-1} \end{bmatrix}, \quad J^{(1)} := R^{(1)}J^{(0)}R^{(1)^{-1}}, \quad X^{(1)} := X^{(0)}R^{(1)^{-1}}. \tag{23}$$

Because $J^{(0)}$ and $R^{(1)}$ are upper triangular, so is $J^{(1)}$. Moreover, the diagonal elements of $J^{(0)}$ remain unchanged into $J^{(1)}$. Note that $J^{(1)}$ is equal to J except for element $(1,2)$: $J^{(1)}(1,2) = -\|k\|$ instead of -1. The matrix $K^{(1)}$ is defined to be the solution of

$$AK^{(1)} - K^{(1)}J^{(1)} + BX^{(1)} = 0.$$

From Lemma 3.1, this implies that $K^{(1)}R^{(1)} = K^{(0)}$. Denote the first column of $K^{(1)}$ by $k_1^{(1)}$. From the preceding definitions,

$$k_1^{(1)} = \frac{v_1^{(0)}}{\|v_1^{(0)}\|} = q_1. \tag{24}$$

So far, we have only computed the linear system (22), the modified matrices $J^{(1)}$ and X_1 that satisfy (23) and the new vector $k_1^{(1)}$ from (24). Moreover, note that the $k-1$ columns of $K^{(1)}$ and K are equal, i.e., for $2 \leq i \leq k$, $k_i^{(1)} = k_i$.

Step i+1

Assume that the i first columns of $K^{(i)}$ are equal to the i first columns of Q, that the $k-i$ last columns of $K^{(i)}$ are equal to the corresponding columns of K and that

$$AK^{(i)} - K^{(i)}J^{(i)} + BX^{(i)} = 0,$$

where $J^{(i)}$ is upper triangular. At this step, the i first columns of $K^{(i)}$ (and only these i first columns) have already been computed. These are the i first columns of Q (denoted by $q_1, \ldots, q_i$). From the preceding equations,

$$(A - \alpha I)k_{i+1} + \sum_{s=1}^{i} q_s^{(i)} j_{s,i+1}^{(i)} + Bx_{i+1}^{(i)} = 0. \tag{25}$$

For two vectors $x, y \in \mathbb{C}^N$, the usual scalar product is denoted by

$$< x, y >:= x^* y.$$

The next orthogonal vector q_{i+1} is equal to

$$q_{i+1} = \frac{k_{i+1} - \sum_{j=1}^{i} < q_j, k_{i+1} > q_j}{\|k_{i+1} - \sum_{j=1}^{i} < q_j, k_{i+1} > q_j\|}. \tag{26}$$

In order to find q_{i+1}, we thus need to compute k_{i+1} by solving the linear system (25) and the i scalar products $< q_j, k_{i+1} >$, with $j = 1$ to i.

Define the matrix $R^{(i+1)}$ as follows:

$$r^{(i+1)}(s,t) := \delta_{s,t} \qquad if\ t \neq i+1, \tag{27}$$

$$r^{(i+1)}(i+1,i+1) := \|k_{i+1} - \sum_{j=1}^{i} < q_j, k_{i+1} > q_j\|, \tag{28}$$

$$r^{(i+1)}(j,i+1) := < q_j, k_{i+1} > \qquad for\ 1 \leq j \leq i, \tag{29}$$

$$r^{(i+1)}(j,i+1) := 0 \qquad for\ i+2 \leq j \leq r. \tag{30}$$

At each step of the algorithm, the structure of $R^{(i)}$ is the following:

$$R^{(i)} := \begin{bmatrix} 1 & & & < q_1, k_i > & \\ & \ddots & & \vdots & \\ & & 1 & < q_{i-1}, k_i > & \\ & & & r^{(i)}(i,i) & \\ & & & & I_{n-j} \end{bmatrix}. \tag{31}$$

Note that the inverse of this matrix is easy to compute exactly:

$$R^{(i)^{-1}} := \begin{bmatrix} 1 & & & -\frac{<q_1,k_i>}{r^{(i)}(i,i)} & \\ & \ddots & & \vdots & \\ & & 1 & -\frac{<q_{i-1},k_i>}{r^{(i)}(i,i)} & \\ & & & \frac{1}{r^{(i)}(i,i)} & \\ & & & & I_{n-j} \end{bmatrix}. \tag{32}$$

As usual, the updated matrices are

$$J^{(i+1)} := R^{(i+1)} J^{(i)} R^{(i+1)^{-1}}, \quad X^{(i+1)} := X^{(i)} R^{(i+1)^{-1}}. \tag{33}$$

Moreover, the matrix $K^{(i+1)}$ that satisfies

$$AK^{(i+1)} - K^{(i+1)} J^{(i+1)} + BX^{(i+1)} = 0,$$

(where $J^{(i+1)}$ is upper triangular) is such that its first $i+1$ columns are equal to the first $i+1$ columns of Q and its $k-i-1$ last columns are equal to the corresponding columns of K.

Consider the equations when we do not perform exact solves at each step but where we allow for a residual error term e_{i+1} in the equation

$$(A - \alpha I)k_{i+1} + \sum_{s=1}^{i} q_s^{(i)} j_{s,i+1}^{(i)} + Bx_{i+1}^{(i)} = e_{i+1}, \quad \|e_{i+1}\|_2 < \delta. \tag{34}$$

In practice, (25) is computed using iterative solvers such as GMRES until the desired tolerance is reached. As the vectors k_{i+1} are iteratively computed within a given tolerance, the residual error of the (7) is small as well:

$$AK - KJ + BX = E, \qquad \|E\| < \delta.$$

The problem is that the orthogonalization step (26) can lead to a large residual error for q, especially when the vectors k_i become parallel. This can be overcome by iteratively solving the linear system (25) for the orthogonalized vector q_i until the tolerance is reached.

6 Proposed Algorithm

The following algorithm, written in pseudo-code Matlab, implements the proposed strategy for controlling the influence of approximate solutions to the linear systems.

Algorithm 6.1 *In order to compute the orthonormal basis Q of K within a given residual error δ, proceed as follows :*

1. *Initialization: Construct the matrices X and J as defined in* (6).
2. *For i=1:k,*
   - (a) $error = \infty$; $R = eye(n)$; $Rinverse = eye(n)$; $\delta = 10^{-6}$; $V = zeros(n, k)$;
   - (b) $vecright = -BX(:, i) + \sum_{j=1}^{i-1} V(:, j)J(j, i)$;
   - (c) *While* $error > \delta$,
     - i. $V(:, i) = V(:, i) + gmres((A - \lambda I_n), vecright, [], \delta)$;
     - ii. *Updating of the matrices J and X:*
       - A. $vproj = zeros(n, 1)$;
       - B. *For* $j = 1 : (i - 1)$, $R(j, i) = (V(:, j)')V(:, i)$; $vproj = vproj + R(j, i)V(:, j)$; $R(i, i) = norm(V(:, i) - vproj)$;
       - C. *For* $j = 1 : (i - 1)$, $Rinv(j, i) = -R(j, i)/R(i, i)$; $Rinv(i, i) = 1/R(i, i)$;
       - D. $J = R.J.Rinv$; $X = XRinv$;
     - iii. *Orthogonalization:*
       - A. $V(:, i) = V(:, i)/norm(V(:, i))$;
       - B. *for* $j = 1 : i - 1$, $sp = (V(:, j)')V(:, i)$; $V(:, i) = V(:, i) - sp.V(:, j)$; $V(:, i) = V(:, i)/norm(V(:, i))$;
     - iv. *residual error verification:*
       $res = AV(:, i) - VJ(:, i) + BX(:, i)$; $error = norm(res)$;
     - v. *if* $error > \delta$,
       $vecright = -gmres((A - \lambda I_n), res, [], \delta)$;

Remark 6.1 *In order to compute q_{i+1}, it is numerically preferable to perform an orthogonalization at each step:*

1. $\bar{q}^{(0)}(i + 1) := k_{i+1}$

2. *For $s = 1$ to i, compute*

$$\bar{q}^{(s)}(i+1) := \frac{\bar{q}^{(s)}(i+1) - < q_s, k_{i+1} > q_i}{\|\bar{q}^{(s)}(i+1) - < q_s, k_{i+1} > q_i\|}.$$

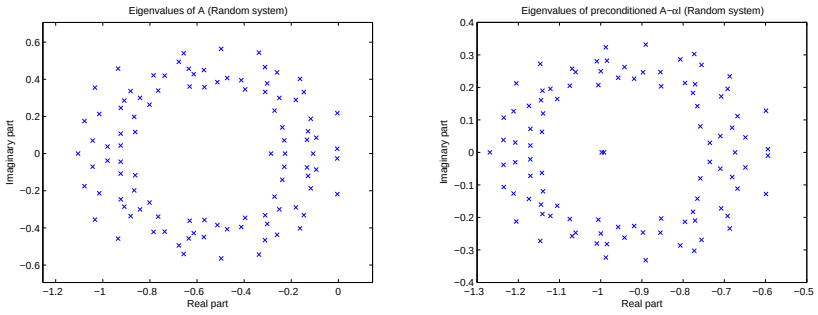
3. *After the last step i, we obtain $q_{i+1} = \bar{q}^{(i)}(i+1)$. This corresponds to step 2(c)iii of Algorithm 6.1.*

7 Numerical examples

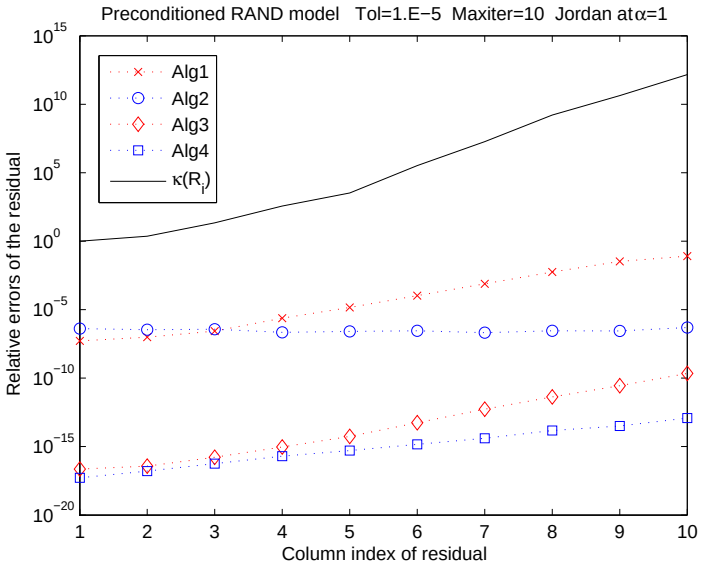
In this section we show numerical experiments to illustrate the accuracy we can obtain with the proposed new algorithm (Alg2) and compare its results with the so-called naive algorithm (Alg1). In both algorithms the solves were performed with the GMRES code of Matlab with a prescribed tolerance of tol=10^{-5}. We also compared these results with what one could call "exact" methods where we replaced the GMRES calls by sparse solves (Matlab's backslash for sparse matrices). These two other algorithms were called respectively Alg3 and Alg4 in the experiments. In all the solves we used a simple diagonal preconditioner scaling the row norms to 1.

7.1 Random Matrix

This is the benchmark RAND from the Benchmark Collection reported in [2]. The matrices A, B, C are sparse randomly generated. The matrix E is the identity. The state dimension is $n = 100$, the input dimension is $m = 2$ and the output dimension is $p = 2$. We generated the $n \times m$ matrix X randomly using the Matlab function `randn`. We ran two experiments : one with the interpolation point $\alpha = 1$ and J_α a Jordan block of dimension 10, and one with multipoint interpolation using a 10×10 upper triangular matrix J with eigenvalues in the right half plane. The figure below describes the spectrum of the original matrix A and of the preconditioned shifted matrix $A - \alpha I_n$ needed for the solves.



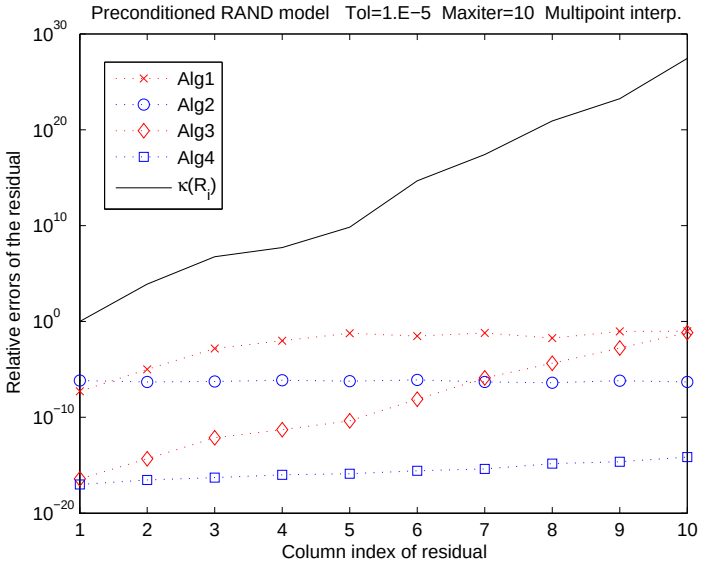
An orthonormal basis of the generalized Krylov subspace $\mathcal{K}$ is computed using the four algorithms described above. Since GMRES performs reasonably well for this random matrix example, we set the maximum number of inner iterations in GMRES to 10. The accompanying table gives the total number of GMRES steps performed by algorithms 1 and 2.



Alg1	24	23	23	23	24	25	25	25	25	25
Alg2	12	12	13	15	17	19	20	22	23	24

We observe here that the total amount of work performed by the two methods is comparable, but that Algorithm 2 indeed yields the required error level in the transformed coordinate system, whereas the naive algorithm looses all accuracy by column 7 of the residual. When replacing the iterative solves by exact solves (Algorithms 3 and 4) the starting accuracy is, as expected, much better, but as one progresses further in the Krylov subspace, accuracy decreases. The solid line equals $\kappa(R(1:i, 1:i))$ which is a progressive way to measure how poorly conditioned the original Krylov matrix was.

In the second test we look at multipoint interpolation.

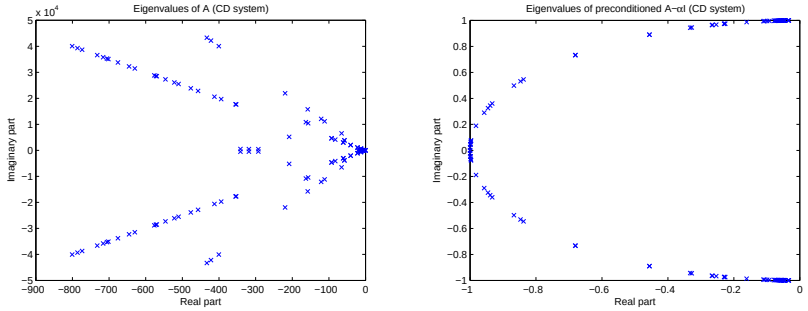


Alg1	28	30	29	36	36	31	28	30	30	30
Alg2	15	25	27	34	38	39	41	46	54	58

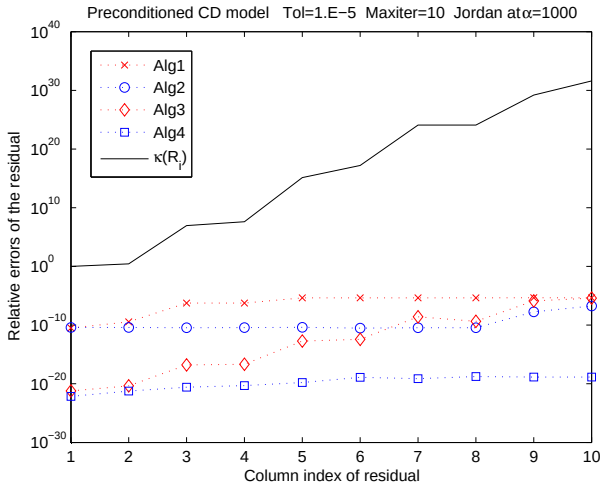
Here we see that even the exact solves used in the naive algorithm cannot produce the same accuracy as Algorithm 2 with approximate solves. The loss of precision in both naive algorithms is very drastic but Algorithm 2 has to compensate this with more work as can be seen from the number of GMRES steps needed.

7.2 CD Player

This system describes the dynamics between the lens actuator and the radial arm position of a portable compact disc player [2]. The matrices A, B, C are sparse and E is the identity. The state dimension is $n = 120$, the input dimension is $m = 2$ and the output dimension is $p = 2$. We again generated the $n \times m$ matrix X randomly using the Matlab function `randn` and ran two experiments : one with the interpolation point $\alpha = 1000$ and J_α a Jordan block of dimension 10, and one with multipoint interpolation using a 10×10 upper triangular matrix J with eigenvalues in the right half plane. The figure below describes the spectrum of the original matrix A and of the preconditioned shifted matrix $A - \alpha I_n$ needed for the solves.



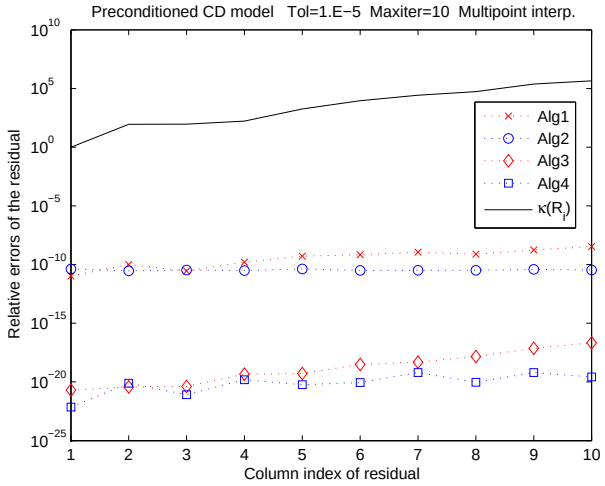
An orthonormal basis of the generalized Krylov subspace $\mathcal{K}$ is again computed using the four algorithms. The original spectrum of A is quite dispersed but the diagonal preconditioning puts all eigenvalues on a half circle which is a considerable improvement. Good results were obtained when setting the maximum number of inner iterations in GMRES equal to 10. A first experiment was done with $\alpha = 1000$ and a Jordan block J_α of dimension 10 (the size of the eigenvalues of A is also in the thousands). The accompanying table gives the total number of GMRES steps performed by algorithms 1 and 2.



Alg1	28	24	29	28	26	29	29	26	28	29
Alg2	17	20	43	45	73	76	99	94	100	100

Both naive algorithms again have significant difficulty getting any reasonable accuracy, and Algorithm 2 again pays the price of the precision it manages to obtain.

The last experiment involves a multipoint approximation : J is chosen random upper triangular with eigenvalues in the right half plane. This problem turns out to be better conditioned as the growth of $\kappa(R(1:i,1:i))$ indicates.



Alg1	34	38	34	39	35	38	35	38	36	37
Alg2	21	22	23	23	23	25	27	27	33	34

8 Concluding Remarks

It is clear that Algorithm 6.1 performs better than Algorithm 4.1. Nevertheless, problems can appear some of which will be discussed in the full paper.

One drawback of Algorithm 6.1 is that at each step, we first compute the vector k_i, i.e. the i-th column of the Krylov matrix K, before the iterative step. It turns out that the norm of the

successive columns of the Krylov matrix K can grow very fast. This can lead to poor convergence of GMRES. Moreover, important numerical errors can appear in the orthogonalization step. In order to remedy this problem, it should be possible to normalize the matrices J and X iteratively. It is also possible to address this by considering applying the suggested modification to more advanced Rational Krylov algorithms that can be placed in this framework, e.g., Dual Rational Arnoldi and Rational Lanczos.

The choice of preconditioning use with GMRES clearly has an effect on the efficiency possible given a desired tolerance. More detailed work is required on this.

The ideas presented here can also be applied in a nontangential sense. A comparison with nontangential techniques, e.g., using an SVD in order to find the most important directions iteratively for X, is needed.

Finally, we briefly noted that we can interpret the effect of the error in terms of a backward error on the original system. The form here was a simple one-sided statement. This can be done in a true two-sided form and can be coupled with constraints on the form of the error to provide a more complete framework in which to conduct a rigorous error analysis of the Rational Krylov family of methods with approximate solutions to the linear systems and Sylvester equations.

Acknowledgements

A research fellowship from the Belgian National Fund for Scientific Research is gratefully acknowledged by A. Vandendorpe. This paper presents research supported by the Belgian Programme on Inter-university Poles of Attraction, initiated by the Belgian State, Prime Ministers Office for Science, Technology and Culture. This work was also supported by the National Science Foundation under Grant No. ACI 0324944.

References

[1] A. Bjorck. **Numerical Methods for Least Squares Problems**, SIAM, 1996.

[2] Y. Chahlaoui and P. Van Dooren. Benchmark examples for model reduction of linear time-invariant dynamical systems, in *Dimension reduction of large-scale systems*, Eds. P. Benner et al., LNCSE Vol. 45, PP. 379–392, Springer, 2005.

[3] C. de Villemagne and R. E. Skelton. Model reductions using a projection formulation. *Internat. J. Control*, 46(6):2141–2169, 1987.

[4] K. Gallivan, A. Vandendorpe, and P. Van Dooren. Model reduction of MIMO systems via tangential interpolation. *SIAM Journal on Matrix Analysis and Applications*, 26(2): 213–229, 2004.

[5] E. Grimme. *Krylov projection methods for model reduction.* PhD thesis, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 1997.

[6] A. Vandendorpe. *Model Reduction of Linear Systems, an Interpolation Point of View*, PhD thesis, Université catholique de Louvain, Louvain-la- Neuve, Belgium, 2004 (copies available at *http* : *//www.bib.ucl.ac.be*).