CHAPTER 15

LARGE-SCALE MATRIX COMPUTATIONS IN CONTROL: KRYLOV SUBSPACE METHODS

15.1 INTRODUCTION

Numerically effective computational methods for various control problems discussed in preceding chapters are viable only for dense computations. Unfortunately, these methods are not suitable for solutions of many large practical problems such as those arising in the design of large sparse structures, power systems, etc. There are two main reasons for this. First, they destroy the sparsity, inherited in most large practical problems and second, they are $O(n^3)$ methods and thus, are computationally prohibitive for large problems. The sparsity is lost by the use of canonical forms such as, triangular Hessenberg and real-Schur, which are obtained by using Gaussian eliminations, Householder and Givens transformations, and those techniques are well-known to destroy the sparsity.

On the other hand, there have been some fine recent developments in the area of large-scale matrix computations. A class of classical methods known as the **Krylov subspace methods** (Lanczos 1950; Arnoldi 1951) have been found to be suitable for sparse matrix computations. The reason is that these methods can be implemented using matrix-vector multiplications only; therefore, the sparsity in the original problem can be preserved. The examples are the **Generalized Minimal Residual** (GMRES) and the **Quasi-Minimal Residual** (QMR) methods for linear systems problem; the **Arnoldi, Lanczos**, and the **Jacobi–Davidson** methods, and several variants of them such as the **restarted** and **block Arnoldi methods and band Lanczos method** for eigenvalue problems.

It is only natural to develop algorithms for large-scale control problems using these effective large-scale techniques of matrix computations. Some work to this effect has been done in the last few years.

In this chapter, we will briefly review some of these methods. In Section 15.2, we give a brief description of the basic Arnoldi and Lanczos methods to facilitate the understanding of how these methods are applied to solve large-scale control problems. We stress that *the descriptions of our Krylov subspace methods are basic.* For practically implementable versions of these methods and associated software, we refer the readers to the books by Bai *et al.* (2000), and Saad (1992a, 1996). *In particular, the homepage, ETHOME of the book by Bai et al.* (2000) *contains valuable information of available software.* Our only goal of this chapter is to show the readers how these modern iterative numerical methods can be gainfully employed to solve some of the large and sparse matrix problems arising in control.

15.2 THE ARNOLDI AND BLOCK ARNOLDI METHODS

In this section, we summarize the essentials of the scalar Arnoldi and block Arnoldi methods.

15.2.1 The Scalar Arnoldi Method

Given an $n \times n$ matrix A, a vector v, and an integer $m \leq n$, the scalar Arnoldi method computes simultaneously a set of orthonormal vectors $\{v_1, \ldots, v_{m+1}\}$, an $(m+1) \times m$ matrix $\tilde{H}_m$ such that

$$AV_m = V_{m+1}\tilde{H}_m, \tag{15.2.1}$$

where $V_m = (v_1, \ldots, v_m)$ and $V_{m+1} = (V_m, v_{m+1})$. *The vectors* $\{v_1, \ldots, v_m\}$ *form an orthonormal basis of the Krylov subspace* $K_m(A, v_1) = \text{span}\{v_1, Av_1, \ldots, A^{m-1}v_1\}$. Furthermore, it is easy to establish that

$$V_m^{\mathrm{T}} A V_m^{\mathrm{T}} = H_m, \tag{15.2.2}$$

where H_m is an $m \times m$ upper Hessenberg matrix obtained from $\tilde{H}_m$ by deleting its last row. **The algorithm breaks down at step** j**, i.e.,** $v_{j+1} = 0$**, if and only if the degree of the minimal polynomial of** v_1 **is exactly** j, that is, it is a combination of j eigenvectors.

15.2.2 The Block Arnoldi Method

The block Arnoldi method is a generalization of the scalar Arnoldi method. Starting with a block vector V_1 of order $n \times p$ and norm unity, the block Arnoldi method constructs a set of block vectors $\{V_1, \ldots, V_{m+1}\}$ such that if $U_m = (V_1, \ldots, V_m)$, then $U_m^{\mathrm{T}} U_m = I_{mp \times mp}$, and $U_m^{\mathrm{T}} A U_m$ is an upper block Hessenberg matrix $H_m = (H_{ij})$. Furthermore, $AU_m - U_m H_m = V_{m+1} H_{m+1,m} E_m^{\mathrm{T}}$, where E_m is the last p columns of the $mp \times mp$ identity matrix. **The block Arnoldi algorithm is particularly suitable for handling multivariable control problems.**

Algorithm 15.2.1. *The Block Arnoldi Algorithm (Modified Gram–Schmidt Version).*

```
Let $V$ be an $n \times p$ matrix.
Step 0. Compute the $n \times p$ orthogonal matrix $V_1$ by finding the QR factorization of $V$: $V = V_1 R$ (Note that $R$ is here $p \times p$). (Use column pivoting if $V$ does not have full rank).
Step 1. For $k = 1, 2, \dots, m$ do
    Compute $\hat{V} = AV_k$.
    For $j = 1, 2, \dots, k$ do
    $H_{j,k} = V_j^{\mathrm{T}} \hat{V}$
    $\hat{V} = \hat{V} - V_j H_{j,k}$
    End
Compute $H_{k+1,k}$ by finding the QR factorization of $\hat{V}$: $\hat{V} = V_{k+1} H_{k+1,k}$
End
```

The block Arnoldi algorithm clearly breaks down if $H_{k+1,k}$ becomes zero for some k. ***Such a breakdown has positive consequences in some applications.*** *(See Section 4.1.1.)*

Remarks

- Define the block $mp \times mp$ upper Hessenberg matrix $H_m = (H_{ij})$,

  $$U_m = (V_1, V_2, \dots, V_m)$$

  and

  $$U_{m+1} = (U_m, V_{m+1}).$$

  Then relations analogous to (15.2.1) and (15.2.2) hold:

  $$AU_m = U_{m+1} \tilde{H}_m,$$

  where

  $$\tilde{H}_m = \begin{pmatrix} H_m \\ 0 \dots 0 H_{m+1,m} \end{pmatrix}_{(m+1)p \times mp},$$

  and

  $$U_m^{\mathrm{T}} A U_m = H_m.$$

15.2.3 The Lanczos and Block Lanczos Methods

For a nonsymmetric matrix A, the Lanczos algorithm constructs, starting with two vectors v_1, and w_1, a **pair** of **biorthogonal bases** $\{v_1, \dots, v_m\}$ and $\{w_1, \dots, w_m\}$ for the two Krylov subspaces: $K_m(A, v_1) = \text{span}\{v_1, Av_1, \dots, A^{m-1}v_1\}$ and $K_m(A^{\mathrm{T}}, w_1) = \text{span}\{w_1, A^{\mathrm{T}}w_1, \dots, (A^{\mathrm{T}})^{m-1}w_1\}$.

Algorithm 15.2.2. *The Nonsymmetric Lanczos Algorithm*

Step 0. *Scale the vectors v and w to get the vectors v_1 and w_1 such that $w_1^{\mathrm{T}} v_1 = 1$. Set $\beta_1 \equiv 0$, $\delta_1 \equiv 0$, $w_0 = v_0 \equiv 0$.*

Step 1. *For $j = 1, 2, \ldots, m$ do*

```
        $\alpha_j = w_j^{\mathrm{T}} A v_j$
        $\hat{v}_{j+1} = A v_j - \alpha_j v_j - \beta_j v_{j-1}$
        $\hat{w}_{j+1} = A^{\mathrm{T}} w_j - \alpha_j w_j - \delta_j w_{j-1}$
        $\delta_{j+1} = \sqrt{|\hat{w}_{j+1}^{\mathrm{T}} \hat{v}_{j+1}|}$
        $\beta_{j+1} = \hat{w}_{j+1}^{\mathrm{T}} \hat{v}_{j+1} / \delta_{j+1}$
        $w_{j+1} = \hat{w}_{j+1} / \beta_{j+1}$
        $v_{j+1} = \hat{v}_{j+1} / \delta_{j+1}$
    End.
```

If the algorithm does not break down before completion of m steps, then, defining $V_m = (v_1, \ldots, v_m)$ and $W_m = (w_1, \ldots, w_m)$, we obtain (i) $W_m^{\mathrm{T}} A V_m = T_m$, (ii) $A V_m = V_m T_m + \delta_{m+1} v_{m+1} e_m^{\mathrm{T}}$, and $A^{\mathrm{T}} W_m = W_m T_m^{\mathrm{T}} + \beta_{m+1} w_{m+1} e_m^{\mathrm{T}}$, where T_m is tridiagonal $(\alpha_1, \ldots, \alpha_m; \beta_2, \ldots, \beta_m; \delta_2, \ldots, \delta_m)$.

Breakdown of the Lanczos method: If neither v_j nor w_j is zero, but $w_j^{\mathrm{T}} v_j = 0$, then we have a breakdown (see Wilkinson (1965, p. 389)). In that case, the **look-ahead Lanczos** idea has to be applied (see Bai *et al.* 2000), Parlett *et al.* (1985), and Freund *et al.* (1993).

The Block Lanczos Method

Starting with $n \times p$ block vectors P_1 and Q_1 such that $P_1^{\mathrm{T}} Q_1 = I$, the block Lanczos method generates right and left Lanczos block vectors $\{Q_j\}$ and $\{P_i\}$ of dimension $n \times p$, and a block tridiagonal matrix $T_{\mathrm{B}} =$ Tridiagonal $(T_1, \ldots, T_m; L_2, \ldots, L_m;\ M_2, \ldots, M_m)$ such that defining

$$P_{[m]} = (P_1, P_2, \ldots, P_m) \quad \text{and} \quad Q_{[m]} = (Q_1, Q_2, \ldots, Q_m),$$

we have (i) $Q_{[m]}^{\mathrm{T}} A Q_{[m]} = T_B$, (ii) $A Q_{[m]} = Q_{[m]} T_B + Q_{m+1} M_{m+1} E_m^{\mathrm{T}}$, and (iii) $A^{\mathrm{T}} P_{[m]} = P_{[m]} T_B^{\mathrm{T}} + P_{m+1} L_{m+1}^{\mathrm{T}} E_m^{\mathrm{T}}$, where E_m is an $mp \times m$ matrix of which bottom square is an identity matrix and zeros elsewhere.

For details of the algorithm, see Bai *et al.* (2000) and Golub and Van Loan (1996). *The block Lanczos method breaks down if $P_{j+1}^{\mathrm{T}} Q_{j+1}$ is singular.* In such a situation, an *adaptively blocked Lanczos method* (Bai *et al.* 1999) can be used to deal with the situation of breakdown. ABLE adaptively changes the block size and maintains the full or semi biorthogonality of the block Lanczos vectors.

15.3 SCOPES OF USING THE KRYLOV SUBSPACE METHODS IN CONTROL

Since the Krylov subspace methods (such as, the Arnoldi and Lanczos methods) are the projection methods onto K_m, it is only natural to use these methods as the projection techniques to solve large-scale control problems, as has been done in numerical linear algebra for matrix problems.

A **template** is then as follows: First, the original large control problem is projected onto an m-dimensional Krylov subspace by constructing a basis of the subspace. The projected smaller problem is then solved using a standard well-established technique. Finally, an approximate solution of the original problem is obtained form the solution of the projected problem. The solution of the projected problem is constructed such that either a **Galerkin property** is satisfied, that is, the residual is orthogonal to the associated Krylov subspace, or the norm of the residual error is minimized (**GMRES type**). These projected methods usually give cheaply computed residual error norms, which, in turn can be used as a stopping criteria in case the methods need to be restarted. For a description of the GMRES method, see Saad and Schultz (1986).

15.4 ARNOLDI METHODS FOR LYAPUNOV, SYLVESTER, AND ALGEBRAIC RICCATI EQUATIONS

Numerical methods for solving the **Lyapunov equations** $AX + XA^{\mathrm{T}} + BB^{\mathrm{T}} = 0$ **(Continuous-time)**, and $AXA^{\mathrm{T}} - X + BB^{\mathrm{T}} = 0$ **(Discrete-time)** have been discussed in Chapter 8. The standard **Schur-method (Section 8.5.2, Section 8.5.4)**, and **Algorithms 8.6.1** and **8.6.2**, based on the Schur decomposition of A, is not suitable for sparse problems. In the following subsections, we show the use of scalar Arnoldi to solve the *single-input continuous-time* and that of the block Arnoldi to solve the *multi-input discrete-time problem.*

The matrix A is assumed to be *stable* in each case; that is, in the continuous-time case, A is assumed to have all eigenvalue negative real parts and in the discrete case, A is assumed to have all its eigenvalues within the unit circle.

Algorithm 15.4.1. *An Arnoldi Method for the Single-Input Stable Lyapunov Equation*

Step 1. *Run m steps of the Arnoldi algorithm with* $v_1 = b/\|b\|_2 = b/\beta$. *Obtain* V_m *and* H_m.

Step 2. *Solve the projected* $m \times m$ *Lyapunov matrix equation:* $H_m G_m + G_m H_m^{\mathrm{T}} + \beta^2 e_1 e_1^{\mathrm{T}} = 0$, *using the Schur-method (see* **Section 8.5.2**).

Step 3. *Compute* X_m, *an approximation to* X: $X_m = V_m G_m V_m^{\mathrm{T}}$.

Galerkin condition, residual error and re-start: (i) It is shown (Saad 1990; Jaimoukha and Kasenally 1994) that the residual $\text{Res}(X_m) = AX_m + X_m A^{\mathrm{T}} + bb^{\mathrm{T}}$ satisfies: $V_m^{\mathrm{T}}\,\text{Res}(X_m)V_m = 0$ and (ii) the residual error-norm for the projected solution: $\|\text{Res}(G_m)\|_{\mathrm{F}} = \sqrt{2}\|h_{m+1,m}e_m^{\mathrm{T}}G_m\|_{\mathrm{F}}$. Using this cheaply computed residual error-norm as a stopping criterion, Algorithm 15.4.1 can be restarted at every fixed number (say m_1) of iterations, wherever needed.

Algorithm 15.4.2. *A Block Arnoldi Algorithm for Stable Discrete-Time Lyapunov Equation*

Step 1. *Find the QR factorization of B to compute V_1 of order $n \times p$:*

$$B = V_1 R.$$

Step 2. *Run m steps of the block Arnoldi algorithm to obtain H_m, U_m, and $H_{m+1,m}$ with V_1 as obtained in Step 1.*

Step 3. *Obtain an $mp \times mp$ matrix G_m by solving the projected discrete Lyapunov equation using the Schur-method* **(Section 8.5.4):**

$$H_m G_m H_m^{\mathrm{T}} + \begin{pmatrix} R \\ 0 \end{pmatrix} (R^{\mathrm{T}}\ 0) = G_m.$$

Step 4. *Compute the approximate solution $X_m = U_m G_m U_m^{\mathrm{T}}$.*

Galerkin condition, residual error norm, and Restart

1. The residual $\text{Res}(X_m) = AX_m A^{\mathrm{T}} - X_m + BB^{\mathrm{T}}$ satisfies the **Galerkin property**: $U_m^{\mathrm{T}}\text{Res}(X_m)U_m = 0$.

   Furthermore, the **residual error norm** for the solution of the projected problem is given by (Jaimoukha and Kasenally 1994):

$$\|\text{Res}(G_m)\|_{\mathrm{F}} = \left\| H_{m+1,m}E_m^{\mathrm{T}}G_m \left(\sqrt{2}H_m^{\mathrm{T}}\quad E_m H_{m+1,m}^{\mathrm{T}}\right) \right\|_{\mathrm{F}}.$$

2. If H_m is also discrete-stable, then the error bound $\|X - X_m\|_2$ converges to zero as m increases (Boley 1994).
3. *As in the continuous-time case, the cheaply computed residual can be used to restart the process if necessary.*

Arnoldi Methods for Sylvester Equation

Let A, B, and C be the matrices of order n. (**Note that the matrix B here is not the usual control matrix.**) We have seen in **Section 8.2.1** that the Sylvester equation: $AX - XB = C$ can be written as the linear systems of equations: $(I \otimes A - B^{\mathrm{T}} \otimes I)x = c$, where $\otimes$ denotes the Kronecker product, and x and c are vectors with

n^2 components. Solving the Sylvester equation this way will require a formidable amount of storage and time for large and sparse problems. Hu and Reichel (1992) have proposed a method to solve this system requiring a considerable reduced amount of storage space. Their idea is to replace the Krylov subspace $K_m(I \otimes A - B^{\mathrm{T}} \otimes I, r_0)$ with a subspace of the form $K_m(B^{\mathrm{T}}, g) \otimes K_m(A, f)$ for certain vectors f and g. The vectors f and g are chosen so that the initial residual vector $r_0 = b - Ax_0$, where x_0 is the initial approximate solution, lies in the Krylov subspace $K_m(B^{\mathrm{T}}, g) \otimes K_m(A, f)$.

Algorithm 15.4.3. *A Restarted Arnoldi Algorithm for the Sylvester Equation $AX - XB = C$ (Galerkin type)*

Step 1. *Choose x_0 and compute $r_0 \equiv c - (I \otimes A - B^{\mathrm{T}} \otimes I)x_0$.*

Step 2. *If $\|r_0\|_2 \leq \epsilon$, then compute the approximate solution matrix X_0 of the equation $AX - XB = C$ from the entries of x_0.*

Step 3. *Choose f and g using the following scheme:*

Let R_0 be defined by: $e_j^{\mathrm{T}} R_0 e_k = e_{j+n(k-1)}^{\mathrm{T}} r_0$, $1 \leq j, k \leq n$. Then, if $\|R_0\|_1 \geq \|R_0\|_\infty$, determine $g = R_0^{\mathrm{T}} f / \|f\|^2$, taking f as a column of R_0 of the largest norm. Else, determine $f = R_0 g / \|g\|^2$, taking g as a row of R_0 of the largest norm.

Using the Arnoldi algorithm, compute the orthonormal bases of $K_{m+1}(A, f)$ and $K_{m+1}(B^{\mathrm{T}}, g)$; that is, obtain $H_A, H_B, \tilde{H}_A, \tilde{H}_B, V_m, V_{m+1}, W_m, W_{m+1}$.

Step 4. *Compute $\tilde{r}_0 = (W_m \otimes V_m)^{\mathrm{T}} r_0$.*

Step 5. *Determine Q_A and R_A from H_A, and Q_B and R_B from H_B by Schur factorizations. That is, find Q_A, R_A; Q_B, R_B such that $H_A = Q_A R_A Q_A^*$ and $H_B = Q_B R_B Q_B^*$. Compute $r_0' = (Q_B \otimes Q_A)^* \tilde{r}_0$.*

Step 6. *Solve the triangular or the quasi-triangular system: $(I \otimes R_A - R_B \otimes I)y_0' = r_0'$ and compute $y_0 = (Q_B \otimes Q_A)y_0'$.*

Step 7. *Compute the correction vector: $z_0 = (W_m \otimes V_m)y_0$ and update the solution: $x_0 \equiv x_0 + z_0$.*

Step 8. *Compute the updated residual vector: $r_0 \equiv r_0 - (W_m \otimes V_{m+1}\tilde{H}_A)y_0 + (W_{m+1}\tilde{H}_B \otimes V_m)y_0$ and go to Step 2.*

A **breakdown of the algorithm** occurs **when the matrix of the linear system in Step 6 becomes singular**. In this case, one can either reduce m or restart the algorithm with different f and g, for example, random vectors. The same action should be taken when $\dim K_m(A, f) < m$ or $\dim K_m(B^{\mathrm{T}}, g) < m$.

Block Arnoldi Methods for Sylvester Equation

While the Hu–Reichel algorithm is a projection algorithm on the linear algebraic system associated with the Sylvester equation, projection algorithms on the actual

Sylvester equation have recently been developed (El Guennouni *et al.* 2003; Robbé and Sadkane 2002). Furthermore, the Hu–Reichel algorithm has been extended to the block form by Simoncini (1996). Let $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{p\times p}$.

Algorithm 15.4.4. *Block Arnoldi Methods for the Sylvester Equation $AX + XB = C$.*

Step 1. *Choose X_0 and compute the residual matrix $R_0 = C-(AX_0-X_0B)$. Assume that* $\text{rank}(R_0) = q$.

Step 2. *Obtain an upper triangular matrix Λ_1 by computing the full rank QR factorization of R_0: $R_0 = V_1\Lambda_1$ and run m steps of the block Arnoldi algorithm with V_1 to obtain U_m, H_m, and $\tilde{H}_m$* (**Algorithm 15.2.1**).

Step 3. *(**Galerkin-type**): Compute the approximate solution: $X_m^G = X_0 + U_mZ_m$, obtaining Z_m by solving the Sylvester equation using the Hessenberg–Schur method* (**Algorithm 8.5.1**)*:*

$$H_mZ_m - Z_mB = \Lambda, \text{ where } \Lambda = \begin{pmatrix}\Lambda_1\\ 0\end{pmatrix} \in \mathbb{R}^{mq\times p}.$$

*(**GMRES-type**): Compute the approximate solution: $X_m^{\mathrm{GM}} = X_0 + U_mZ_m$, obtaining Z_m by solving the minimization problem:*

$$\min_{Z\in R^{mq\times p}} \|\bar{\Lambda} - \bar{S}_m(Z)\|_{\mathrm{F}}, \text{ where } \bar{\Lambda} = \begin{pmatrix}\Lambda_1\\ 0\end{pmatrix} \in \mathbb{R}^{(m+1)q\times p}$$

and

$$\bar{S}_m(Z) = \tilde{H}_mZ - \begin{pmatrix}Z\\ 0\end{pmatrix} B,$$

Residuals and restart: It can be shown (Robbé and Sadkane (2002)) that the residuals $R_m^{\mathrm{GM}} = S(X_m^{\mathrm{GM}}) - C$ and $R_m^{\mathrm{G}} = S(X_m^{\mathrm{G}}) - C$ satisfy, respectively:

$$\|R_m^{\mathrm{GM}}\|_{\mathrm{F}} = \|\bar{\Lambda} - \bar{S}_m(Z_m)\|_{\mathrm{F}} \quad \text{and} \quad \|R_m^{\mathrm{G}}\|_{\mathrm{F}} = \|H_{m+1,m}Z_m^{\mathrm{L}}\|_{\mathrm{F}},$$

where Z_m^{L} is the last $q \times p$ block of Z_m. Using these easily computed residuals, the method should be periodically restarted with $X_0 = X_m^{\mathrm{G}}$ or $X_0 = X_m^{\mathrm{GM}}$, where $X_m^{\mathrm{G}}|X_m^{\mathrm{GM}}$ is the last computed approximate solution with Galerkin/GMRES method.

Convergence analysis (Robbé and Sadkane 2002)

1. The GMRES algorithm converges if the field of values of A and B are disjoint. If the Galerkin algorithm converges, then the GMRES algorithm also converges. However, if **GMRES** stagnates (i.e., $\|R_m^{\mathrm{GM}})\|_{\mathrm{F}} = \|R_0\|_{\mathrm{F}}$), then the Galerkin algorithm fails.

   Note: It is assumed that R_0 and the parameter m are the same in both these algorithms.

2. (*Breakdown*). If the block Arnoldi algorithm breaks down at iteration m; that is, if $H_{m+1,m} = 0$, then the approximate solutions computed by GMRES and the Galerkin algorithm are exact; that is, $X_m^{\mathrm{G}} = X_m^{\mathrm{GM}} = X$.

Arnoldi Method for Sylvester-Observer Equation (Single-Output Case)

The Sylvester-observer equation $A^{\mathrm{T}}X - XH = C^{\mathrm{T}}G$ arises in the construction of Luenberger observer (see Chapter 12; Datta 1994). For a full-rank solution X, it is necessary that (A, C) is observable and (H^{T}, G) is controllable. If H is an upper Hessenberg matrix (as in the case of the scalar Arnoldi), then in the single-output case g can be chosen to be $e_n = (0, 0, \dots, 0, 1)$ and the Sylvester-observer equation in this case reduces to $A^{\mathrm{T}}X - XH = (0, 0, \dots, 0, c^{\mathrm{T}})$. An Arnoldi method was developed by Datta and Saad (1991) to solve this equation by observing the striking resemblance of this equation with the Arnoldi equation: $AV_m - V_mH_m = (0, 0, \dots, 0, h_{m+1,m}v_{m+1})$. Naturally, the Arnoldi vector v_1 should be chosen so that the last vector v_{m+1} becomes the vector c, given a priori. This is done by observing that, apart from a multiplicative scalar, the polynomial $p_m(x)$ such that $v_{m+1} = p_m(A)v_1$, is the characteristic polynomial of H_m (see Saad 1992a). The matrix H_m is constructed to have a pre-assigned spectrum $\{\mu_1, \dots, \mu_m\}$ for which an eigenvalue assignment algorithm (e.g., Datta 1987) is invoked at the end of $(m - 1)$ steps of the Arnoldi algorithm with the chosen vector v_1.

Algorithm 15.4.5. *An Arnoldi Algorithm for Single-output Sylvester-Observer Equation*

Step 1. *Solve the linear system:* $q(A^{\mathrm{T}})x = c^{\mathrm{T}}$, *and compute* $v_1 = x/\|x\|$; *where* $q(t) = (t - \mu_1)(t - \mu_2)\cdots(t - \mu_m)$.

Step 2. *Run* $m - 1$ *steps of the Arnoldi method on* A^{T} *with* v_1 *as the initial vector to generate* V_m *and the first* $m - 1$ *columns of* H_m. *Let* $\tilde{H}_{m-1}$ *denote the matrix of the first* $m - 1$ *columns of* H_m.

Step 3. *Find a column vector* y *such that* $\Omega([\tilde{H}_{m-1}, y]) = \Omega(H_m) = \{\mu_1, \dots, \mu_m\}$, *where* $\Omega(K)$ *denotes the spectrum of the matrix* K.

Step 4. *Compute* $\alpha = (c')^{\mathrm{T}}c'/\|c\|^2$, *where* c' *is the last column of* $A^{\mathrm{T}}V_m - V_mH_m$.

Step 5. *Set* $X_m = (1/\alpha)V_m$.

Solving the equation $\boldsymbol{q(A^{\mathrm{T}})x = c^{\mathrm{T}}}$ *using the partial fraction approach:* A partial fraction approach suggested in Datta and Saad (1991) to solve the above polynomial system of equations consists in decomposing the system into m linearly independent systems: $(A^{\mathrm{T}} - \mu_iI)x_i = c^{\mathrm{T}}$, $i = 1, \dots, m$ and then obtaining the solution x as the linear combination: $x = \sum_{i=1}^{n} \frac{1}{q'(\mu_i)}x_i$, where $q'(\mu_j) = \Pi_{i=1,\dots,m,i\neq j}$. Each of these systems can be solved by applying k steps of the Arnoldi method, constructing an orthonomial basis V_k of the span $\{c, A, c, \dots, A^{k-1}c\}$ and then solving k independent small $m \times m$ Hessenberg linear systems. The bulk of the work is in constructing V_k, and this is done only once. A detailed stability (numerical) property of the approach studied in Calvetti *et al.* (1995), Calvetti and Reichel (1997), and Calvetti *et al.* (2001) shows that *the performance of the scheme can be improved by choosing μ_is as the equidistant points on a circle or on the zeros of a certain Chebyshev polynomial.*

Remarks

- Observe that the solution obtained by this algorithm has the nice additional property of being **orthonormal**.
- A full Arnoldi-type of method ($m = n$) for the construction of an orthogonal solution to the multi-output Sylvester-observer equation has been developed by Datta and Hetti (1997). Also, there now exists a singular value decomposition (SVD)-based algorithm (Datta and Sarkissian 2000) for solving the multi-output Sylvester-observer equation, which might be suitable for large-scale computing.

Arnoldi Method for Continuous-Time Algebraic Riccati Equation (CARE)

In **Chapter 13**, we have described numerical solutions of the algebraic Riccati equations. *The Schur method, the generalized Schur method, or similar methods based on matrix decompositions are not practical for large problems.* An idea to solve the CARE using the block Arnoldi method developed by Jaimoukha and Kasenally (1994) is as follows. For simplicity, we write the CARE as: $XA + A^{\mathrm{T}}X - XBB^{\mathrm{T}}X + LL^{\mathrm{T}} = 0$ (i.e., $R = I$ and $Q = LL^{\mathrm{T}}$). **Assume that the associated Hamiltonian matrix does not have a purely imaginary eigenvalue**.

Algorithm 15.4.6. *An Arnoldi Algorithm for CARE (Galerkin-type)*

Step 1. *Compute U_m, H_m, $H_{m+1,m}$ by running m steps of the block Arnoldi method starting with V_1 given by: $L = V_1 R$ (QR* **factorization of** *L). Define B_m by $U_m^{\mathrm{T}} B = B_m$ and L_m by $U_m L_m = L$.*

Step 2. *Solve the projected equation for G_m:*

$$G_m H_m + H_m^{\mathrm{T}} G_m - G_m B_m B_m^{\mathrm{T}} G_m + L_m L_m^{\mathrm{T}} = 0$$

Step 3. *Compute approximation X_m of X: $X_m = U_m G_m U_m^{\mathrm{T}}$*

Galerkin condition and restart

1. **The residual norm satisfies Res(X_m) the Galerkin property: V_m^{T} Res(X_m)$V_m = 0$.**

2. Algorithm 15.4.6 can be **restarted** by using the cheaply computed residual error norm: $\|\mathrm{Res}(G_m)\|_{\mathrm{F}} = \sqrt{2}\|H_{m+1,m}E_m^{\mathrm{T}}G_m\|_{\mathrm{F}}$, as a stopping criterion.

15.5 ARNOLDI METHOD FOR PARTIAL EIGENVALUE ASSIGNMENT

Let the spectrum of an $n \times n$ matrix A be denoted by $\Omega(A) = \{\lambda_1, \ldots, \lambda_p, \lambda_{p+1}, \ldots, \lambda_n\}$. Recall from Chapter 11 that the Partial Eigenvalue Assignment (PEVA) is defined as follows: Given an $n \times n$ large and sparse matrix A, with partial spectrum $\{\lambda_1, \ldots, \lambda_p\}$, an $n \times m$ control matrix B, and a set of self-conjugate scalars $\{\mu_1, \ldots, \mu_p\}$, the problem is the one of finding a feedback matrix K such that $\Omega(A - BK) = \{\mu_1, \ldots, \mu_p; \lambda_{p+1}, \ldots, \lambda_n)$. The problem naturally arises in feedback stabilization of large systems such as large space structures, power plants.

We have described a Sylvester equation approach due to Datta and Sarkissian (2002) in Chapter 11. Here we describe a projection method due to Saad (1988), which can be implemented using the Arnoldi method. It is based on computing an orthonormal basis for the left invariant subspace associated with the p eigenvalues that are to be reassigned.

Algorithm 15.5.1. *A Projection Algorithm for Partial Pole-Placement*

Step 1. *Compute the partial Schur decomposition:* $A^{\mathrm{T}}Q = QR$ *associated with the eigenvalues* $\lambda_1, \lambda_2, \ldots, \lambda_p$.

Step 2. *Compute* $S_0 = Q^{\mathrm{T}}B$ *and solve the projected* $p \times p$ *eigenvalue assignment problem. That is, find a matrix* G *such that* $\Omega(R^{\mathrm{T}} - S_0G^{\mathrm{T}}) = \{\mu_1, \mu_2, \ldots, \mu_p\}$, *using a* ***standard multi-input EVA method*** **(Algorithm 11.3.1)**.

Step 3. *Form the feedback matrix:* $K = (QG)^{\mathrm{T}}$.

15.6 LANCZOS AND ARNOLDI METHODS FOR MODEL REDUCTION

In **Chapter 14**, we have described several techniques for model reduction. These include model reduction via **balancing** and **the Schur method**. Since these methods require reduction of the state-matrix A to real-Schur form, they are not suitable for large and sparse computations. Here we describe some Krylov-subspace ideas. These Krylov methods are designed to construct a reduced-order model (ROM) such that the first few Markov parameters (see **Chapter 9**) of this model match with those of the original model.

Several Krylov subspace methods for model reduction have been developed in recent years. These include the *Padé via Lanczos (PVL)* approach, the *interpolation approach*, based on the rational Krylov method of Ruhe, *implicitly restarted Lanczos method*, and *Arnoldi and implicitly restarted dual Arnoldi methods. The PVL technique has been proven to be effective in circuit simulation and the multipoint rational interpolation approach has been successful in moment matching of the transfer function at selected frequencies.* The machinery needed to describe these techniques has not been developed here and, therefore, we have to skip the descriptions of these techniques. For state-of-the-art survey on this topic, see Antoulas (2003) and Van Dooren (2000), and Datta (2003).

We will describe here only a basic Lanczos and an Arnoldi method for model reduction in the single-input, single-output (SISO) case and just mention the existence of the block Lanczos and band Lanczos methods in the multi-input, multi-output (MIMO) case.

15.6.1 Lanczos Methods for Model Reduction

Algorithm 15.6.1. *A Lanczos Algorithm for SISO Model Reduction*

Step 0. *Scale the vectors b and c to obtain the vectors v_1 and w_1 such that $w_1^{\mathrm{T}} v_1 = 1$.*

Step 1. *Run k steps of the Lanczos algorithm* (**Algorithm 15.2.2**) *to generate the matrices W_k and V_k and then compute $A_k = W_k^{\mathrm{T}} A V_k$, $b_k = W_k^{\mathrm{T}} b$, $c_k = c V_k$.*

Step 2. *Form the reduced-order model (A_k, b_k, c_k).*

It can be shown that the reduced-order model defined by (A_k, b_k, c_k) preserves the first $2k$ Markov parameters of the original system. (See Gragg 1974; Gragg and Lindquist 1983). That is, $cA^{i-1}b = c_k A_k^{i-1} b_k, i = 1, 2, \ldots, 2k$.

Numerical Disadvantages and Possible Cures

There are several numerical difficulties with the above algorithm: first, there can be serious "breakdowns" in the Lanczos process due to the ill-conditioning of the submatrices in the system's Hankel matrix; second, the steady-state error can be large; third, the stability of the ROM is not guaranteed even though the original model is stable. An **implicit restated Lanczos scheme** due to Grimme *et al.* (1996), to stabilize the ROM is as follows: Suppose that the matrix A_k is not stable and assume that there are q unstable modes: $\mu_1, \ldots, \mu_q$. Then the idea is to restart Algorithm 15.6.1 with the new starting vectors $\bar{v}_1 = \bar{p}_\vartheta (A - \mu_q I) \cdots (A - \mu_1 I) v_1$, and $\bar{w}_1 = \bar{p}_w (A^{\mathrm{T}} - \mu_q I) \cdots (A^{\mathrm{T}} - \mu_1 I) w_1$, where $\bar{p}_\vartheta$ and $\bar{p}_w$ are certain scalars. The scheme is implemented implicitly using a technique similar to the one proposed in Sorensen (1992). There also exist relations between the modified

Markov parameters of the original system and the above restarted Lanczos model (see Grimme *et al.* 1996).

15.6.2 Block Lanczos and Band Lanczos Methods for MIMO Model Reduction

In the MIMO case, when $m = r$, the **block Lanczos** method can be used. Specifically, the following result (see Boley 1994) can be proved.

Theorem 15.6.1. *Let j steps of the block Lanczos method be applied to the MIMO system (A, B, C), starting with block vectors generated from the QR decompositions of B and C, obtaining the matrices $P_{[j]}$ and $Q_{[j]}$. Define $\hat{A} = Q_{[j]}^{\mathrm{T}} A P_{[j]}$, $\hat{B} = Q_{[j]}^{\mathrm{T}} B$, $\hat{C} = C P_{[j]}$. Then the ROM $(\hat{A}, \hat{B}, \hat{C})$ has the following properties: $\hat{C}\hat{A}^i\hat{B} = CA^iB$ for $i = 0, 1, \ldots, 2(j-1)$.*

The *band Lanczos method* is an extension of the standard nonsymmetric Lanczos method for single vectors to blocks of starting vectors of different sizes. This method is thus ideal for the MIMO case when $m \neq r$. For space limitations, the detailed description of the algorithm cannot be given here. For description of the algorithm, we refer the readers to the paper by Aliga *et al.* (2000). For application of the band Lanczos algorithm to the MIMO model reduction, see Freund (1999) and the paper by Freund in Bai *et al.* (2000, pp. 205–216). See also, Bai *et al.* (1997), Bai and Freund (1999), and Freund (1997).

15.6.3 An Arnoldi Method for SISO Model Reduction

The idea is to use the Arnoldi method simultaneously on (A, b) and $(A^{\mathrm{T}}, c^{\mathrm{T}})$ and then combine the results to obtain ROMs. The ROMs have been shown to satisfy the Galerkin conditions (Jaimoukha and Kasenally 1997).

Algorithm 15.6.2. *An Arnoldi Algorithm for SISO Model Reduction*

Step 1. *Perform m steps of the Arnoldi method with (A, b) to obtain H_m, $\tilde{H}_m$, V_m, $\tilde{V}_m$ and l_m, with $v_1 = b/\|b\|_2$. ($\tilde{V}_m = v_{m+1}$, $\tilde{H}_m = h_{m+1,m}e_m^{\mathrm{T}}$ and $l_m = \|b\|_2 e_1$).*

Step 2. *Perform m steps of the Arnoldi method with $(A^{\mathrm{T}}, c^{\mathrm{T}})$ to produce G_m, $\tilde{G}_m$, W_m, $\tilde{W}_m$ and k_m, with $w_1 = c^{\mathrm{T}}/\|c\|_2$, $((\tilde{G}_m)^{\mathrm{T}} = g_{m,m+1}e_m^{\mathrm{T}}$, $\tilde{W}_m = w_{m+1}$ and $k_m = \|c\|_2 e_1^{\mathrm{T}}$).*

Step 3. *Form $T_m = W_m^{\mathrm{T}} V_m$, $\hat{H}_m = T_m^{-1} W_m^{\mathrm{T}} A V_m = H_m + T_m^{-1} W_m^{\mathrm{T}} \tilde{V}_m \tilde{H}_m$ and $\hat{G}_m = W_m^{\mathrm{T}} A V_m T_m^{-1} = G_m + \tilde{G}_m \tilde{W}_m^{\mathrm{T}} V_m T_m^{-1}$*

Step 4. *Form the ROM $(\hat{H}_m, l_m, k_m T_m)$ or $(\hat{G}_m, T_m l_m, k_m)$.*

Galerkin conditions and residual errors: Let $h_m(s) = (sI - \hat{H}_m)^{-1} l_m$ and $g_m(s) = k_m(sI - \hat{G}_m)^{-1}$. Then the Galerkin conditions $W_m^{\mathrm{T}}((sI - A)V_m h_m(s) - b) = 0$, and $(g_m(s)W_m^{\mathrm{T}}(sI - A) - c)V_m = 0$, $\forall s$ are satisfied.

Remarks

- *Jaimoukha and Kasenally (1997) have described a restarted Arnoldi framework which may be employed to make the ROMs stable and to remove redundant modes in the models.* For space limitation, we skip the description of this implicit method here.
- Antoulas *et al.* (2001) have recently proposed a restarted Arnoldi method, closely related to the one described above, based on the concept of the **Cross Grammian**. For space limitation, we skip the description here and refer the readers to the above paper.

15.7 CHAPTER NOTES AND FURTHER READING

In this chapter, we have provided a very brief review of some of the existing Krylov subspace methods for a few large problems arising in design and analysis of control problems. These include Arnoldi methods for Lyapunov and Sylvester equations by Saad (1990), Hu and Reichel (1992), Jaimoukha and Kasenally (1994); Arnoldi method for the single-output Sylvester-observer equation by Datta and Saad (1991); a projection algorithm (which can be implemented using Arnoldi method) for PEVA problem by Saad (1988); and Lanczos and Arnoldi methods for model reduction by Boley (1994), Grimme *et al.* (1996), and Jaimoukha and Kasenally (1995, 1997). See Boley and Golub (1984, 1991) for Krylov subspace methods for determining controllability.

The Hu–Reichel algorithm was extended by Simoncini (1996) to block form. There have also been some recent developments on the Krylov subspace methods for Sylvester equation. El Guennouni *et al.* (2001) have developed block Arnoldi and nonsymmetric block Lanczos algorithms for Sylvester equation. Robbé and Sadkane (2002) have proposed new block Arnoldi and block GMRES methods for Sylvester equation and analyzed their convergence properties in details.

In the context of model reduction, it is noted that there are other important methods, such as the PVL, the interpolation methods, etc., which have not been included here. For details of these methods, the readers are referred to the associated papers cited in the reference section of this Chapter. In particular, for Lanczos methods of model reduction see, Feldman and Freund (1995a, 1995b, 1995c), Jaimoukha and Kasenally (1997), Grimme *et al.* (1996), Papakos and Jaimoukha (2001), Papakos (2001), Papakos and Jaimoukha (2002), Gallivan *et al.* (1996), etc. The paper by Papakos and Jaimoukha (2002) contains a procedure for model reduction combining nonsymmetric Lanczos algorithm and Linear Fractional Transformations (LFT). The delightful recent surveys by Freund (1999), the recent research monograph by Antoulas (2003), Ph.D. thesis by Grimme (1994), and short course lecture notes by Van Dooren (1995, 2000) and Feldman and Freund (1995b) are good sources of knowledge for model reduction. The paper by Freund (1999) includes 123 references on large-scale matrix computations using Krylov methods and their applications to model reduction. The earlier general surveys on Krylov subspace methods in control include the papers by Boley (1994), Datta (1997), Boley and Datta (1996), Van Dooren (1992), Bultheel and Van Barel (1986), and Fortuna *et al.* (1992). Some other papers of interest on Krylov subspace methods for model reduction include the papers by Villemagne and Skelton (1987), and Su and Craig, Jr. (1991).

For recent algorithms on partial eigenvalue and eigenstructure assignments which are not Krylov subspace methods, but suitable for large-scale computations, see Sarkissian (2001) and Datta and Sarkissian (2002). See also Calvetti *et al.* (2001).

Research Problems

1. Develop a block Arnoldi type algorithm to solve the multi-output Sylvester-observer equation $AX - XB + GC$, analogous to single-output algorithm (**Algorithm 15.4.5**).
2. Develop a block Arnoldi algorithm for the discrete-time Algebraic Riccati equation (DARE): $A^{\mathrm{T}}XA - X + Q - A^{\mathrm{T}}XB(R + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA = 0$, analogous to Algorithm 15.4.6 in the continuous-time case.
3. Develop a block Arnoldi algorithm for the generalized Sylvester equation: $AXB - X = C$.
4. Develop a block Arnoldi algorithm for MIMO model reduction that preserves stability of the original system.

References

Aliaga J.I., Boley D.L., Freund R.W., and Hernández V. "A Lanczos-type method for multiple starting vectors," *Math. Comp.*, Vol. 69, pp. 1577–1601, 2000.

Antoulas A.C., Sorensen D.C., and Gugercin S. "A survey of model reduction methods for large-scale systems," in *Contemporary Math.* (Olshevsky V., ed.), American Mathematical Society, Providence, Vol. 280, pp. 193–219, 2001.

Antoulas A.C. *Lectures on Approximations of Large-Scale Dynamical Systems,* SIAM, Philadelphia, 2003 (To appear).

Arnoldi W.E. "The principle of minimized iterations in the solution of the matrix eigenvalue problem," *Quart. Appl. Math.,* Vol. 9, pp. 17–29, 1951.

Bai Z., Feldmann P., and Freund R.W. "Stable and passive reduced-order models based on partial Padé approximation via the Lanczos process," *Numerical Analysis Manuscript*, No. 97-3-10, Bell Laboratories, Murray Hill, NJ, 1997.

Bai Z., Day D., and Ye Q. "ABLE: an adaptive block Lanczos method for nonhermitian eigenvalue problem," *SIAM J. Matrix Anal. Appl.*, pp. 1060–1082, 1999.

Bai Z. and Freund R. "A band symmetric Lanczos process based on coupled recurrences with applications in model-order reduction," *Numerical Analysis Manuscript*, Bell Laboratories, Murray Hill, NJ, 1999.

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

Boley D.L. "Krylov space methods on state-space control models," *Proc. Circuit Syst. Signal*, Vol. 13, pp. 733–758, 1994.

Boley D. and Datta B.N. "Numerical methods for linear control systems," in *Systems and Control in the Twenty-First Century* (Byrnes C., Datta B., Gilliam D., and Martin C., eds.), pp. 51–74, Birkhauser, Boston, 1996.

Boley D. and Golub G. "The Lanczos–Arnoldi algorithm and controllability," *Syst. Control Lett.*, Vol. 4, pp. 317–327, 1984.

Boley D. and Golub G. "The nonsymmetric Lanczos algorithm and controllability," *Syst. Control Lett.*, Vol. 16, pp. 97–105, 1991.

Brezinski C., Redivo Zaglia M., and Sadok H. "Avoiding breakdown and near-breakdown in Lanczos type algorithms," *Numer. Algorithms*, Vol. 1, pp. 26–284, 1991.

Bultheel A. and Van Barel M. "Padé techniques for model reduction in linear systems theory: a survey," *J. Comput. Appl. Math.*, Vol. 14, pp. 401–438, 1986.

Calvetti D. and Reichel L. "Numerical aspects of solution methods for large Sylvester-like observer-equations," *Proc. IEEE Conf. Decision Control*, pp. 4389–4397, 1997.

Calvetti D., Gallopoulas E., and Reichel L. "Incomplete partial fractions for parallel evaluations of rational matrix functions," *J. Comp. Appl. Math.*, Vol. 59, pp. 349–380, 1995.

Calvetti D., Lewis B., and Reichel L. "On the solution of large Sylvester-observer equation," *Num. Lin. Alg. Appl.*, Vol. 8, pp. 435–451, 2001.

Calvetti D., Lewis B., and Reichel L. "Partial eigenvalue assignment for large linear control systems," in *Contemporary Mathematics* (Olshevsky V. ed.), American Mathematical Society, Providence, RI, Vol. 28, pp. 24–254, 2001.

Datta B.N. "An algorithm to assign eigenvalues in a Hessenberg matrix: single-input case," *IEEE Trans. Autom. Control*, AC-32, pp. 414–417, 1987.

Datta B.N. "Linear and numerical linear algebra in control theory: some research problems," *Lin. Alg. Appl.*, Vol. 197/198, pp. 755–790, 1994.

Datta B.N. *Numerical Linear Algebra and Applications*, Brooks/Cole Publishing Company, Pacific Grove, CA, 1995.

Datta B.N. "Krylov-subspace methods for control: an overview," *Proc. IEEE Conf. Decision Control*, 1997.

Datta B.N. "Krylov-subspace methods for large-scale matrix problems in control," *Future Generation of Computer Systems*, Vol. 19, pp. 125–126, 2003.

Datta B.N. and Hetti C. "Generalized Arnoldi methods for the Sylvester-observer matrix equation and the multi-input eigenvalue assignment problems," *Proc. IEEE Conf. Decision Control*, pp. 4379–4383, 1997.

Datta B.N. and Saad Y. "Arnoldi methods for large Sylvester-like observer matrix equations, and an associated algorithm for partial spectrum assignment," *Lin. Alg. Appl.*, Vol. 154–156, pp. 225–244, 1991.

Datta B.N. and Sarkissian D. "Block algorithms for state estimation, and functional observers," *Proc. IEEE Joint Conf. on Control Appl.*, pp. 19–23, 2000.

Datta B.N. and Sarkissian D. Partial eigenvalue assignment: Existence, uniqueness, and numerical solutions, Proc. Mathematical Theory of Networks and Systems, Notre Dame, August, 2002.

El Guennouni A., Jbilou K., and Riquet J. "Block Krylov subspace methods for solving large Sylvester equations," preprint, LMPA, No. 132, Université du Littoral, 2001 (To appear in *Numer. Algorithms*), 2003.

Feldmann P. and Freund R.W. "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Trans. Comput.-Aided Design*, Vol. 14, pp. 639–649, 1995a.

Feldman P. and Freund R.W. *Numerical Simulation of Electronic Circuits: State-of-the-Art Techniques and Challenges*, Course Notes, 1995b (*Available on-line from http:/cm.bell-labs.com/who/Freund*).

Feldman P. and Freund R.W. "Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm," *Proc. 32nd Design Autom. Conf., Assoc. Comp. Mach.*, pp. 474–479, 1995c.

Fortuna L., Nunnari G., and Gallo A. *Model Order Reduction Techniques with Applications in Electric Engineering*, Springer-Verlag, London, UK, 1992.

Freund R.W. "Computing minimal partial realization via a Lanczos-type algorithm for multiple starting vectors," *Proc. IEEE Conf. Decision Control*, pp. 4394–4399, 1997.

Freund R.W. "Reduced-order modeling techniques based on Krylov subspace methods and their uses in circuit simulation," in *Applied and Computational Control, Signals, and Circuits* (Datta B.N. *et al.*, eds.), Vol. 1, pp. 435–498, Birkhauser, Boston, 1999.

Freund R.W., Gutknecht M.H., and Nachtigal N.M. "An implementation of the look-ahead Lanczos algorithm for non-hermitian matrices," *SIAM J. Sci. Comput.*, Vol. 14, pp. 137–158, 1993.

Gallivan K., Grimme E.J., and Van Dooren P. "A rational Lanczos algorithm for model reduction," *Numer. Algorithms*, Vol. 12, pp. 33–63, 1996.

Golub G.H. and Van Loan C.F. *Matrix Computations*, 3rd edn, Johns Hopkins University, Baltimore, MD, 1996.

Gragg W.B. "Matrix interpolations and applications of the continued fraction algorithm," *Rocky Mountain J. Math.*, Vol. 4, pp. 213–225, 1974.

Gragg W.B. and Lindquist A. "On the partial realization problem," *Lin. Alg. Appl.*, Vol. 50, pp. 277–319, 1983.

Grimme E.J., Sorensen D.C., and Van Dooren P. "Model reduction of state space systems via an implicitly restarted Lanczos method," *Numer. Algorithms*, Vol. 1, pp. 1–32, 1996.

Grimme E.J. *Krylov Projection Methods for Model Reduction*, Ph.D. Thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1994.

Hu D.Y. and Reichel L. "Krylov subspace methods for the Sylvester equation," *Lin. Alg. Appl. Appl.*, Vol. 172, pp. 283–313, 1992.

Jaimoukha I.M. "A general minimal residual Krylov subspace method for large-scale model reduction," *IEEE Trans. Autom. Control,* Vol. 42, pp. 1422–1427, 1997.

Jaimoukha I.M. and Kasenally E.M. "Krylov subspace methods for solving large Lyapunov equations," *SIAM J. Numer. Anal.*, Vol. 31, pp. 227–251, 1994.

Jaimoukha I.M. and Kasenally E.M. "Oblique projection methods for large scale model reduction," *SIAM J. Matrix Anal. Appl.*, Vol. 16, pp. 602–627, 1995.

Jaimoukha I.M. and Kasenally E.M. "Implicitly restarted Krylov Subspace methods for stable partial realizations," *SIAM J. Matrix Anal. Appl.*, Vol. 18(3), pp. 633–652, 1997.

Lanczos C. "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *J. Res. Nat. Bur. Standards*, Vol. 45, pp. 255–282, 1950.

Papakos V. and Jaimoukha I.M. "Implicitly restarted Lanczos algorithm for model reduction," *Proc. 40th IEEE Conf. Decision Control*, pp. 3671–3672, 2001.

Papakos V. *An Implicitly Restarted Nonsymmetric Lanczos Algorithm for Model Reduction*, M.Phil. to Ph.D. Transfer Report, Imperial College, London, UK, 2001.

Papakos V. and Jaimoukha I.M. "Model reduction via an LFT-based explicitly restarted Lanczos algorithm (preprint)," *Proc. Math Theory Networks Syst.* (MTNS' 2002), Notre Dame, 2002.

Parlett B.N., Taylor D.R., and Liu Z.A. "A look-ahead Lanczos algorithm for unsymmetric matrices," *Math. Comp.*, Vol. 44, pp. 105–124, 1985.

M. Robbé and Sadkane M. "A convergence analysis of GMRES and FOM methods for Sylvester equations," *Numer. Algorithms,* Vol. 30, pp. 71–84, 2002.

Saad Y. "Projection and deflation methods for partial pole assignment in linear state feedback," *IEEE Trans. Autom. Control,* Vol. 33, pp. 290–297, 1988.

Saad Y. "Numerical solutions of large Lyapunov equations," in *Signal Processing, Scattering, Operator Theory, and Numerical Methods* (Kaashoek M.A., Van Schuppen J.H., and Ran A.C. eds.), pp. 503–511, Birkhauser, 1990.

Saad Y. *Numerical Methods for Large Eigenvalue Problems,* John Wiley, New York, 1992a.

Saad Y. *Iterative Methods for Sparse Linear Systems,* PWS, Boston, MA, 1996.

Saad Y. and Schultz M.H. "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Statist. Comput.*, Vol. 7, pp. 856–869, 1986.

Sarkissian D. *Theory and Computations of Partial Eigenvalue and Eigenstructure Assignment Problems in Matrix Second-order and Distributed Parameter Systems,* Ph.D. Dissertation. Northern Illinois University, DeKalb, Illinois, 2001.

Simoncini V. "On the numerical solution of $AX - XB = C$," *BIT,* Vol. 36, pp. 814–830, 1996.

Sorensen D.C. "Implicit application of polynomial filters in a k-step Arnoldi method," *SIAM J. Matrix Anal. Appl.*, Vol. 13, pp. 357–385, 1992.

Sorensen D.C. and Antoulas A.C. *Projection Methods for Balanced Model Reduction*, Unpublished manuscript, 2001 (available from the webpage: *http://www.ece.rice.edu/ aca*).

Su T.-J. and Craig R.R., Jr. "Model reduction and control of flexible structures using Krylov vectors," *J. Guidance Control Dynam.*, Vol. 14, pp. 260–267, 1991.

Van Dooren P. *The Lanczos Algorithm and Padé Approximations*, Short Course, Benelux Meeting on Systems and Control, 1995.

Van Dooren P. *Gramian-Based Model Reduction of Large-scale Dynamical Systems,* Short Course SIAM Annual Meeting, San Juan, Puerto Rico, July 2000.

Van Dooren P. "Numerical linear algebra techniques for large-scale matrix problems in systems and control," *Proc. IEEE Conf. Decision Control,* 1992.

de Villemagne C. and Skelton R.E. "Model reductions using a projection formulation," *Int. J. Control,* Vol. 46, pp. 2141–2169, 1987.

Wilkinson J.H. *The Algebraic Eigenvalue Problem,* Clarendon Press, Oxford, UK, 1965.