

Efficient Approximate Balanced Truncation of General Large-Scale RLC Systems via Krylov Methods*

Q. Su, V. Balakrishnan and C-K. Koh

School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
Email: {qsu, ragu, chengkok}@ecn.purdue.edu

Abstract

We present an efficient implementation of an approximate balanced truncation model reduction technique for general large-scale RLC systems, described by a state-space model where the C matrix in the time-domain modified nodal analysis (MNA) circuit equation $C\dot{x} = -Gx + Bu$ is not necessarily invertible. The large sizes of the models that we consider make most implementations of the balance-and-truncate method impractical from the points of view of computational load and numerical conditioning. This motivates our use of Krylov subspace methods to directly compute approximate low-rank square roots¹ of the Gramians of the original system. The approximate low-order general balanced and truncated model can then be constructed directly from these square roots. We demonstrate using three practical circuit examples that our new approach effectively gives approximate balanced and reduced order coordinates with little truncation error.

1 Introduction

As VLSI technology advances, accurate modeling of RLC interconnect effects becomes increasingly important. These interconnect models typically involve thousands of tightly coupled RLC components. The analysis and design of large-scale RLC systems can stretch the limits of computing resources.

Model reduction, i.e., finding an approximate model with far fewer variables, is one technique that facilitates the analysis and design of large-scale systems. One well-known approach is “moment-matching” [1, 2, 3].

Another model reduction approach involves truncating the state vector by projecting it on the principal subspace, which can be computed efficiently by Krylov methods [4, 5].

A third technique, one that underlies the approach in this paper, is the balance-and-truncate method (see for example, [6]). This model reduction method is very attractive from a theoretical point of view, as the reduced-order models are guaranteed to be stable, and there is a simple bound for the approximation error [7]. However, its direct use for the model reduction of large-scale systems is impractical because of its need to solve two large-size Lyapunov equations, and a large-size eigen-decomposition. This issue has been addressed using several different approaches. One of the techniques is to solve the original large size Lyapunov equations approximately, for example, the Smith method [8], and the Alternate Direction Iteration (ADI) method [9]. In the Smith method, a closed-form expression for the Gramians is derived as an infinite summation; an approximation can be found simply via a finite summation. The ADI method can be thought of as a generalization of the Smith method. However, for an implementation of the ADI method to be computationally competitive, the original system matrix must be tridiagonalized first [9]; this is computationally demanding and possibly numerically ill-conditioned [10, §9.3.6]. One approach towards addressing the issue of tridiagonalization is presented with the VADI (vector-ADI) method in [11, 12]. Note that in [11, 12], after the authors find the approximate Gramians by the VADI method, they project the original system to the principal Gramian eigenspace to perform model reduction, rather than balanced truncation.

Our approach is based on the idea of balance-and-truncate method, and is close in spirit to the VADI method proposed in [11, 12]: We directly compute the approximate square-roots of the Gramians (rather than

*This work was supported in part by NSF under contract number CCR-9984553, SRC under contract number 99-TJ-689, and Purdue Research Foundation.

¹We use the term “square root” to mean the not necessarily symmetric square root of a matrix: If $M = M^T = NN^T$, we say N is the square root of M .

the Gramians themselves); as is well known, this results in a much better-conditioned numerical implementation. The method proposed in [11, 12] requires J matrix-vector solves of a system with N variables, where J denotes the rank of the approximate square root of the Gramians and N denotes the original state-space dimension. We show in Section 3.3 that as a consequence, for large-size problems, the dominant cost incurred with the VADI method is approximately $J/2$ times the cost with the method that we propose here. Therefore, our argument is that for large-scale models encountered in VLSI systems (with the state variables numbering in the hundreds or even thousands), our method offers considerable computational savings over the VADI method.

Our approach begins with the modified Smith method for computing the approximate square-roots of the system Gramians. Krylov methods are employed to compute these square-roots, leading to enormous computational savings. This is our first contribution. Our second contribution is to extend the balanced truncation schemes based on Krylov methods to handle general RLC networks, where the C matrix in the time-domain modified nodal analysis (MNA) circuit equation $C\dot{x} = -Gx + Bu$ may not be invertible. Our approach is to first derive an equivalent system $\hat{C}\hat{x} = -\hat{G}\hat{x} + \hat{B}u$ where \hat{C} is invertible. However, our implementation of the Krylov methods avoids inverting \hat{C} . Thus, we argue that our implementations are particularly suited for VLSI system model reduction, from the points of view of both computational demand and numerical conditioning.

Notation

Given an RLC circuit comprising only passive linear elements, we can extract the MNA equations as follows [5]:

$$C\dot{x} = -Gx + Bu, \quad y = Lx + Du, \quad (1)$$

where $x(t) \in \mathbb{R}^N$, $u(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$. $C = \begin{bmatrix} Q & 0 \\ 0 & H \end{bmatrix}$, is block diagonal, symmetric and non-negative definite. $G = \begin{bmatrix} R & E \\ -E^T & 0 \end{bmatrix}$, where the matrix R is symmetric and non-negative definite. We will consider only single-input single-output systems in this paper; the extension of the results presented herein to multi-input multi-output systems is straightforward. We use 5-tuple $(C, -G, B, L, D)$ to denote the state-space realization of the system. Since the MNA matrices come from a real circuit, the system is stable. In other words, all the finite generalized eigenvalues of $(-G, C)$ have negative real parts.

The objective in model reduction is to obtain another linear system

$$C_r \dot{x}_r = -G_r x_r + B_r u, \quad y_r = L_r x_r + D_r u,$$

where $x_r(t) \in \mathbb{R}^n$, with $n \ll N$, and with the mapping $u \mapsto y$ well-approximated by $u \mapsto y_r$.

The remainder of this paper is organized as follows: In Section 2, we show how for the system (1), we may obtain an equivalent realization $(\hat{C}, -\hat{G}, \hat{B}, \hat{L}, \hat{D})$ where \hat{C} is nonsingular. In Section 3 we describe an algorithm, implemented using Krylov subspace methods, that approximately balances and truncates the realization $(\hat{C}, -\hat{G}, \hat{B}, \hat{L}, \hat{D})$ to obtain a reduced-order model. In Section 4, we present three representative examples that demonstrate that our technique performs very well, while offering significant savings in computation.

2 An equivalent state-space model

With state-space models of the form (1) that represent RLC circuits, the singularity of the matrix C arises from its entire columns and rows being zero [5]. Thus, without loss of generality², we assume that $C = \begin{bmatrix} \hat{C} & 0 \\ 0 & 0 \end{bmatrix}$, where \hat{C} is an $r \times r$ matrix that is symmetric, block diagonal and nonsingular. Partitioning the matrices from (1), we therefore have

$$\begin{aligned} \hat{C}\dot{x}_1 &= -G_{11}x_1 - G_{12}x_2 + B_1u, \\ 0 &= -G_{21}x_1 - G_{22}x_2 + B_2u, \\ y &= L_1x_1 + L_2x_2 + Du, \end{aligned} \quad (2)$$

where $x_1(t) \in \mathbb{R}^r$. We make the important observation that since (2) models an RLC circuit, the matrix G_{22} is nonsingular. Then, eliminating the variable x_2 , we have

$$\hat{C}\dot{x}_1 = -\hat{G}x_1 + \hat{B}u, \quad y = \hat{L}x_1 + \hat{D}u, \quad (3)$$

where

$$\begin{aligned} \hat{G} &= G_{11} - G_{12}G_{22}^{-1}G_{21}, & \hat{B} &= B_1 - G_{12}G_{22}^{-1}B_2, \\ \hat{L} &= L_1 - L_2G_{22}^{-1}G_{21}, & \hat{D} &= D + L_2G_{22}^{-1}B_2. \end{aligned}$$

Note that the state-space dimension with the model (3) is $r \leq N$. This possible reduction in the number of state-variables required to describe the RLC circuit comes at the cost of the manipulations required to arrive at (3) from (1); the major cost with these manipulations is computing G_{22}^{-1} , the inverse of an $(N-r) \times (N-r)$ matrix. When $r \approx N$, this cost is negligible. When $r \ll N$, the cost is significant, but is easily offset by the savings that accrue from the fact that the number of state variables required to describe the model is substantially reduced without sacrificing any model fidelity. We also

²If C is not already of the form noted, a simple permutation of the components of the state vector will render it so.

note that this step of reducing the state dimension to r can be performed as a precursor to *any* model reduction scheme.

3 Approximate balanced truncation

3.1 Balanced truncation

Balanced truncation is one well-known model reduction scheme [6]. In the literature, it is typically derived for systems with state-space realizations of the form $(I, -\hat{C}^{-1}\hat{G}, \hat{C}^{-1}\hat{B}, \hat{L}, \hat{D})$. However, inverting \hat{C} may not be desirable from the points of view of computation and numerical conditioning. We therefore use the notion of “generalized” controllability and observability Gramians, described in [13].

For a state-space model $(\hat{C}, -\hat{G}, \hat{B}, \hat{L}, \hat{D})$, the generalized controllability Gramian, denoted by W_c , is defined as the unique solution to the linear equation

$$-\hat{G}W_c\hat{C}^T - \hat{C}W_c\hat{G}^T + \hat{B}\hat{B}^T = 0. \quad (4)$$

The generalized observability Gramian, denoted by W_o , is defined as $W_o = \hat{C}^T\tilde{W}_o\hat{C}$, where \tilde{W}_o is the unique solution to the linear equation

$$-\hat{G}^T\tilde{W}_o\hat{C} - \hat{C}^T\tilde{W}_o\hat{G} + \hat{L}^T\hat{L} = 0. \quad (5)$$

With the eigenvalues sorted in decreasing order, the corresponding eigenvectors of W_c yield directions in state-space that are increasingly hard to reach from the input u , and the eigenvectors of W_o yield directions that are increasingly hard to observe from the output y .

Let $W_c = XX^T$, and $W_o = YY^T$, and let $X^TY = U\Sigma V^T$ be a singular value decomposition. Then, it can be shown that with the coordinate transformation $T = XUS^{-1/2} = (\Sigma^{-1/2}V^TY^T)^{-1}$, $x_b \triangleq T^{-1}x$, both the generalized controllability and observability Gramians of the balanced realization are diagonal and equal to each other. In other words,

$$W_{c,b} = T^{-1}W_cT^{-T} = \Sigma = W_{o,b} = T^TW_oT,$$

where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$, with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$. The σ_i s are called the Hankel singular values of the system. Thus in the balanced realization, the state components are as reachable from the input as they are observable at the output, with the corresponding Hankel singular value quantifying their reachability and observability. This motivates the next step, that of “truncating” of the state-vector, i.e., simply “throwing away” state components for which the corresponding diagonal entry σ_i of Σ is small.

Solving the two large-size linear equations (4) and (5), and the large-size SVD, has a high computational cost. We address this issue with an approximate balance-and-truncate technique that requires far

less computation. The idea is to directly compute low-rank square roots of the generalized Gramians; these square-roots can be combined to yield “approximate” balancing transformations that automatically truncate the state space.

3.2 Approximate balanced truncation via Krylov subspace methods

For every real scalar $p < 0$, the linear equation (4) is equivalent to

$$A_p W_c A_p^T - W_c + B_p B_p^T = 0,$$

where $A_p \triangleq (p\hat{C} - \hat{G})^{-1}(p\hat{C} + \hat{G})$, and $B_p \triangleq \sqrt{(-2p)}(p\hat{C} - \hat{G})^{-1}\hat{B}$. Therefore, we have

$$W_c \approx \sum_{j=0}^{k-1} A_p^j B_p B_p^T (A_p^T)^j = X_k X_k^T, \quad (6)$$

where

$$X_k \triangleq \mathcal{X}(A_p, B_p, k) = [B_p \quad A_p B_p \quad \dots \quad A_p^{k-1} B_p],$$

a k th order Krylov matrix, is a square root of the generalized controllability Gramian W_c .

Similar manipulations yield, with $\tilde{A}_p \triangleq (p\hat{C} + \hat{G})(p\hat{C} - \hat{G})^{-1}$ and $L_p \triangleq \sqrt{(-2p)}\hat{L}(p\hat{C} - \hat{G})^{-1}$,

$$\tilde{W}_o \approx \sum_{j=0}^{k-1} (\tilde{A}_p^T)^j L_p^T L_p \tilde{A}_p^j = \tilde{Y}_k \tilde{Y}_k^T, \quad (7)$$

where

$$\tilde{Y}_k \triangleq \mathcal{X}(\tilde{A}_p^T, L_p^T, k) \triangleq [L_p^T \quad \tilde{A}_p^T L_p^T \quad \dots \quad (\tilde{A}_p^T)^{k-1} L_p^T]$$

is another k th order Krylov matrix. Then, $Y_k \triangleq \hat{C}^T \tilde{Y}_k$ is a square root of the generalized observability Gramian W_o . Krylov methods, such as Arnoldi method, can be employed to calculate X_k and \tilde{Y}_k , for computational efficiency and numerical well-conditioning.

The discussion of choosing an optimal shift p for good approximation of X_k and Y_k , the square-roots of the system Gramians, and the implementation details of the algorithm and stopping criterion for the iterations are similar to what we presented in an earlier paper [14].

After the approximate k -th order low-rank square roots X_k and $Y_k = \hat{C}^T \tilde{Y}_k$ of the generalized Gramians are computed, we perform a $k \times k$ SVD

$$X_k^T \hat{C}^T \tilde{Y}_k = \hat{U} \hat{\Sigma} \hat{V}^T.$$

The diagonal entries $\hat{\sigma}_i$ of $\hat{\Sigma}$ approximate the first k Hankel singular values of the system. Suppose that the first n of the k Hankel singular values are significant. Define

$$T_l = [I_n \quad 0_{n \times (r-n)}] \hat{\Sigma}^{-\frac{1}{2}} \hat{V}^T \tilde{Y}_k^T \in \mathbb{R}^{n \times r},$$

$$T_r = X_k \hat{U} \hat{\Sigma}^{-\frac{1}{2}} \begin{bmatrix} I_n \\ 0_{(r-n) \times n} \end{bmatrix} \in \mathbf{R}^{r \times n},$$

then, we have an approximately balanced and truncated reduced-order model:

$$\begin{aligned} C_r &= T_l \hat{C} T_r = I_n, & G_r &= T_l \hat{G} T_r, \\ B_r &= T_l \hat{B}, & L_r &= \hat{L} T_r, & D_r &= \hat{D}. \end{aligned} \quad (8)$$

For this reduced order model, it can be verified that the controllability and observability Gramians are

$$W_{c,r} \approx \mathbf{diag}(\hat{\sigma}_1, \dots, \hat{\sigma}_n) \approx W_{o,r}.$$

3.3 Flop count

The flop count of the major steps involved in the implementations of the balance-and-truncate method is $(30 + \frac{2}{3})N^3 + O(N^2)$. For VADI method, the flop count is $\frac{2k}{3}N^3 + O(N^2)$. Our method, fast approximate balancing and truncation with Arnoldi (FABT-Arnoldi), uses about $\frac{4}{3}N^3 + O(N^2)$ flops. Here, the number of states in the full-order model is N , and we assume that the reduced-order model has n states. We assume that k iterations are performed with VADI method, and FABT-Arnoldi method.

It is clear from the above data that the dominant cost with the Arnoldi method is $\frac{4}{3}N^3$, which arises from the two LU factorizations of $N \times N$ matrices required with our procedure to perform matrix-vector solves. In the VADI method proposed in [11, 12], k matrix-vector solves of the form $A_i x_i = b_i$ are needed, with different $N \times N$ matrices A_i , which results in k LU factorizations. Therefore the dominant cost with the VADI methods is $\frac{2k}{3}N^3$. Consequently, for large N , the computation required by VADI methods is roughly $k/2$ times the computation required with our methods. As mentioned before k is no smaller than n , the dimension of the reduced system; thus, significant computational savings can be expected to accrue with our procedure. It will be evident from the flop counts in Table 1 in Section 4.1 that this is indeed the case.

4 Numerical Results

We present a few numerical examples that are representative of the performance of the model reduction technique presented in this paper.

4.1 Examples with $C = I$

We consider two examples cited in [11, 12]. The first is an on-chip planar square spiral inductor suspended over a copper plane. The original system has 500 states. We applied the approximate balanced truncation method (FABT) proposed in this paper on it. 33

Table 1. Spiral inductor flop counts.

	TBR	MMVA	VADI	FABT-Arnoldi
Flops	10.2e9	0.60e9	0.67e9	0.25e9

Arnoldi iterations were employed for computing the approximate square-roots of the system Gramians. We refer to this method as FABT-Arnoldi. Figure 1 shows the relative inductance errors of the resulting reduced-order models generated by our method (FABT-Arnoldi), and the standard balance-and-truncate method (TBR). Both models are of order 7. Comparing this result to the one shown in [12] section 6, figures 1-2, it is evident that our method brings the same accurate approximation as the other model reduction methods: the standard balance-and-truncate method (TBR), the moment matching via Arnoldi method (MMVA), and the vector ADI method (VADI). Furthermore, the computational cost of our method is much lower. See table 1 for comparison of the flop counts required by various model reduction methods. (The flop counts for TBR, MMVA and VADI in the table are reproduced from [11].)

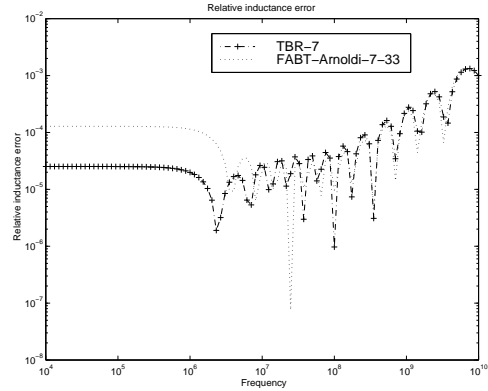


Figure 1. Relative inductance error of the reduced-order models with TBR and FABT-Arnoldi methods.

The second example that we consider has a number of significant oscillatory modes. The example data is from the discretization of a transmission line. The original system has 256 states. Figures 2 and 3 show the frequency responses of the original system and the reduced systems generated by TBR method, and FABT-Arnoldi with 64 Arnoldi iterations. Both the reduced models in figure 2 are of order 10, which is the same as that used in [12]. Both the reduced models in figure 3 are of order 24, which is the number of the significant Hankel singular values of the original systems, approximated by singular values of $X_k^T \hat{Y}_k$. It is evident that our reduced-order model captures the global frequency response behavior as well as the TBR method. Table 2 lists the flop counts

required by various model reduction methods.

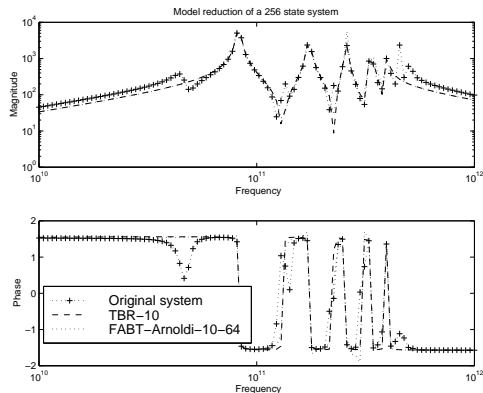


Figure 2. Frequency responses of the original system and reduced systems of order 10.

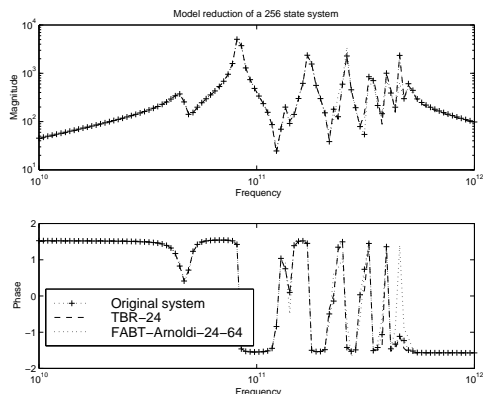


Figure 3. Frequency responses of the original system and reduced systems of order 24.

For purposes of comparison, we also ran the PRIMA model-reduction algorithm [5] on this example to obtain reduced-order models of sizes 10 and 24. Figure 4 shows the frequency response of the original system, as well as those of the two PRIMA-based reduced-order models. It is clear that the frequency responses of the reduced-order models deviate significantly from that of the original system at high frequencies. We point out that PRIMA takes five to six times less computation than that required by FABT-Arnoldi (see Table 2); thus, the computational effort with FABT-Arnoldi is equivalent to that of using PRIMA with five to six expansion points. However, we note that with our method, we have bounds on the frequency response approximation error (inherited from the balance-and-truncate theory); moreover, there are no issues such as the selection of expansion points with our method.

Table 2. Flop counts, reduced models of order 10 and 24.

Order of the reduced models	TBR	FABT-Arnoldi	PRIMA
10	3.18e9	7.95e7	1.41e7
24	3.21e9	8.26e7	1.84e7

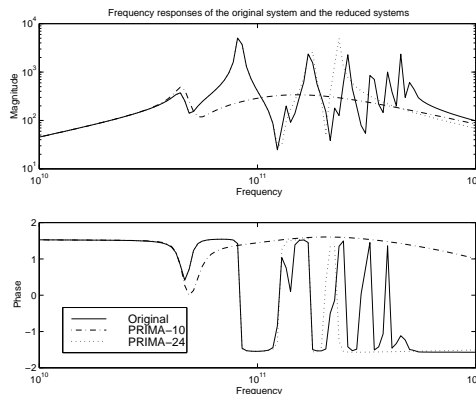


Figure 4. Frequency responses of the original system and reduced systems generated by PRIMA.

4.2 Example with singular C

With the model in (1), the matrix C can be singular, with several columns and rows of zeros. To illustrate the performance of our techniques for such cases, we consider the example from [5, § V.A]. This corresponds to a lossy transmission line that was modeled with 40 lumped RLC sections. After extracting the MNA equations for the RLC circuits, there are $N = 120$ states. The rank of C is 80. Thus, the size of the state-space model $(\hat{C}, -\hat{G}, \hat{B}, \hat{L}, \hat{D})$ with nonsingular \hat{C} (see Section 2) is $r = 80$. We ran 29 Arnoldi iterations to obtain approximations of the square roots of the generalized Gramians, and then performed an approximate balanced truncation. The size of the reduced system $(C_r, -G_r, B_r, L_r, D_r)$ is $n = 12$. Figures 5 and 6 show the frequency response and time domain response (to a ramp input with a rise time of 0.1 ns) of the reduced system and the original system. From these plots, it is evident that the reduced system generated by FABT-Arnoldi performs very well.

5 Conclusion

We have presented an efficient implementation of the approximate balanced truncation model reduction technique for general large-scale RLC systems, using Arnoldi iterations. The distinguishing features of our algorithm are: (i) We provide an efficient way to perform an approximate balanced truncation to the general

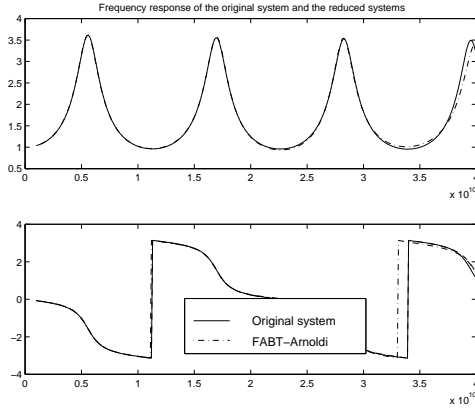


Figure 5. Frequency responses of the original system and reduced system (original size: 120).

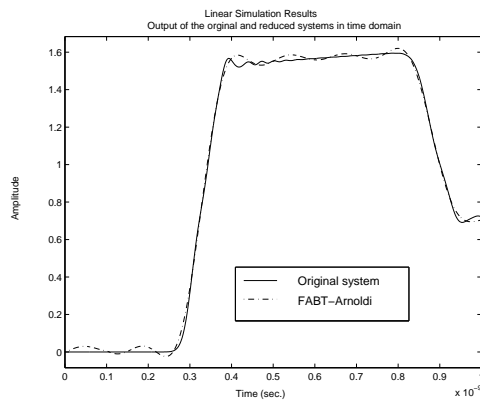


Figure 6. Time domain responses of the original system and reduced system (original size: 120).

state-space model as shown in equations (1), where C may be singular. (ii) We directly compute state coordinate transformations that approximately balance-and-truncate the state vector. (iii) The coordinate transformations are computed directly from Krylov subspace methods and a small-size SVD, without the need for solving any Lyapunov equations. Numerical simulations show that our approach holds much promise in the balance-and-truncate model reduction of large-scale systems.

References

- [1] X. Huang. *Padé Approximation of Linear(ized) Circuit Responses*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1990.
- [2] L. T. Pillage and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Trans. Computer-Aided Design*, 9:352–366, 1990.
- [3] P. Feldmann and R. W. Freund. Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Trans. Computer-Aided Design*, 14:639–649, May 1995.
- [4] E. Grimme and K. Gallivan. Approximate solves in Krylov-based modeling methods. In *Proc. IEEE Conf. on Decision and Control*, pages 3849–3854, December 1997.
- [5] A. Odabasioglu, M. Celik, and L. T. Pileggi. PRIMA: Passive reduced-order interconnect macromodeling algorithm. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 17(8):645–654, 1998.
- [6] M. Green and D. J. N. Limebeer. *Linear Robust Control*. Information and System sciences. Prentice Hall, Englewood Cliffs, NJ, 1995.
- [7] K. Glover. All optimal Hankel-norm approximations of linear multivariable systems and their L_∞ -error bounds. *Int. J. Control*, 39(6):1115–1193, 1984.
- [8] R. A. Smith. Matrix equation $XA + BX = C$. *SIAM Journal on Applied Mathematics*, 16(1):198–201, 1968.
- [9] A. Lu and E. L. Wachspress. Solution of Lyapunov equations by alternating direction implicit iteration. *Computers Math. Appl.*, 21(9):43–58, 1991.
- [10] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins Univ. Press, Baltimore, second edition, 1989.
- [11] J. Li, F. Wang, and J. White. An efficient Lyapunov equation-based approach for generating reduced-order models of interconnect. In *Proc. Design Automation Conference*, pages 1–6, 1999.
- [12] J. Li and J. White. Efficient model reduction of interconnect via approximate system Gramians. In *Proc. Int. Conf. on Computer-Aided Design*, pages 380–383, 1999.
- [13] C. S. Hsu, U. B. Desai, and R. J. Darden. Reduction of large-scale systems via generalized gramians. In *Proc. IEEE Conf. on Decision and Control*, pages 1409–1410, 1983.
- [14] V. Balakrishnan, Q. Su, and C-K. Koh. Efficient balance-and-truncate model reduction for large scale systems. In *Proc. American Control Conf.*, Arlington, VA, June 2001.