Computation of Matrix Padé Approximations of Transfer Functions Via a Lanczos-Type Process

Roland W. Freund

Abstract. We present an approach for the stable and efficient computation of matrix Padé approximations of transfer functions of general multi-input multi-output linear dynamical systems. The proposed method is based on a recently developed Lanczos-type process for multiple starting vectors. We establish a connection between this Lanczos process and matrix Padé approximants of transfer functions.

§1. Introduction

A *transfer function* is a matrix-valued function $H\colon \mathbb{C} \longmapsto (\mathbb{C} \cup \{\infty\})^{p\times m}$ of the form

$$H(z) = L^T (I - zA)^{-1} R, \tag{1}$$

where $A \in \mathbb{C}^{N\times N}$, $R \in \mathbb{C}^{N\times m}$, and $L \in \mathbb{C}^{N\times p}$ are given matrices; see, e.g., [9]. In (1), I denotes the $N \times N$ identity matrix, and $M^T = [\, m_{kj}\,]$ denotes the transpose of the matrix $M = [\, m_{jk}\,]$. Transfer functions (1) arise in the study of the frequency-domain behavior of time-invariant multi-input multi-output linear dynamical systems of the type

$$\begin{aligned} \frac{dx}{dt} &= Ax + Ru(t), \\ y(t) &= L^T x(t). \end{aligned} \tag{2}$$

Here, $u(t)$, $y(t)$, and $x(t)$ are vector-valued functions of length m, p, and N, respectively. The functions $u(t)$ and $y(t)$ represent the input and the output of the system (2), respectively, and $x(t)$ is the unknown function of state variables. Assuming zero initial conditions for (2), the transfer function (1) connects the Laplace-transformed input and output functions, $U(z) = \int_0^\infty u(t)e^{-zt}\,dt$ and $Y(z) = \int_0^\infty y(t)e^{-zt}\,dt$, through the relation $Y(z) = H(z)U(z)$.

Transfer functions (1) are rational matrix functions, since each component of H is a scalar rational function with numerator and denominator degree at most $N-1$ and N, respectively. Often, the number N of state variables is very large. In particular, this is the case for the transfer functions that arise in the simulation of large electronic circuits, where

$N \geq 10,000$ can occur. For such large N, it becomes prohibitive to compute the transfer function; note that the evaluation of $H(z)$ for each z requires the solution of a system of linear equations with m right-hand sides, $(I - zA)X = R$. Instead, for large N, one approximates the transfer function H by a rational function H_n of order $n \ll N$. A common choice is to employ Padé approximation, which has become especially popular in the circuit-simulation community in recent years; see [11,2] and the references given there. An nth *Padé approximant* H_n of the transfer function H is a rational matrix function of type $(n-1, n)$ (in the sense of §2 below) whose Taylor expansion about $z = 0$ matches as many leading Taylor coefficients of H as possible, i.e., $H_n(z) = H(z) + \mathcal{O}(z^{m(n)})$, where the integer $m(n)$ is maximal. Note that, by (1),

$$H(z) = \sum_{i=0}^{\infty} M_i z^i, \tag{3}$$

where the Taylor coefficients M_i are $p \times m$ matrices given by

$$M_i = L^T A^i R, \quad i = 0, 1, \ldots\ . \tag{4}$$

The coefficients (4) are called the *moments* of the transfer function H.

The standard approach to computing Padé approximants H_n is to first generate a sufficient number of moments, followed by the solution of a linear system with a Hankel matrix whose entries are the moments, to obtain the numerator and denominator of H_n. Unfortunately, this approach is numerically unstable due to the typical extreme ill-conditioning of the Hankel matrix, and also because, in finite-precision arithmetic, the moments M_i with increasing i convey only information about the extreme poles of the transfer function H; see, e.g., [2,5]. As a result, moment-based approaches can only be used to compute Padé approximants of very moderate order n, and typically, they are limited to $n \leq 10$.

On the other hand, for the single-input single-output case $m = p = 1$, Padé approximations H_n of scalar transfer functions can be computed stably and efficiently by exploiting the well-known connection [7,8] between Padé approximation and the classical Lanczos process [10] for the successive tridiagonalization of large matrices. Algorithms for the computation of H_n based on this connection have been proposed recently by Feldmann and Freund [2], and independently by Gallivan, Grimme, and Van Dooren [5]. The main point of these algorithms is that they bypass the ill-conditioned moment computations, and instead, they generate the Padé approximant directly from the tridiagonal Lanczos matrix.

However, the classical Lanczos process is limited to the case $m = p = 1$, and the question arises if similar techniques exist for the stable computation of matrix Padé approximants for matrix-valued transfer function for the general case $m, p \geq 1$. In this paper, we give an affirmative answer to this question. We show that a Lanczos-type process for multiple starting vectors, which was recently developed by Aliaga, Boley, Freund, and Hernández [1], can be used to compute matrix Padé approximants of multi-input multi-output transfer functions. This result generalizes the connection between the classical Lanczos process and scalar Padé approximants of single-input single-output transfer functions.

The remainder of the paper is organized as follows. In §2, we briefly review the Lanczos-Padé connection for the single-input single-output case, and we motivate our representation of matrix Padé approximants for the general case. In §3, we describe some basic properties of the Lanczos-type process for multiple starting vectors. Finally, in §4, we present the connection between matrix Padé approximants and the Lanczos process.

§2. The single-input single-output case

For the moment, we now assume that $m = p = 1$. Then, in (1), $R, L \in \mathbb{C}^N$, and H is a scalar rational function. Furthermore, the nth Padé approximant H_n can be represented as

$$H_n(z) = \frac{\varphi_{n-1}(z)}{\psi_n(z)}, \tag{5}$$

where φ_{n-1} and ψ_n are polynomials of degree at most $n-1$ and n, respectively. In the standard approach to computing Padé approximants, one attempts to determine the coefficients of the polynomials φ_{n-1} and ψ_n.

Instead of (5), the Lanczos-based approach to computing H_n uses the following representation:

$$H_n(z) = \left(\eta D_n^T e_1\right)^T (I - zT_n)^{-1} (\rho e_1), \tag{6}$$

where $e_1 = [1 \quad 0 \quad \cdots \quad 0]^T \in \mathbb{R}^n$. All the quantities involved in (6) are generated in the course of the first n steps of the Lanczos algorithm [10]. Starting with $v_1 = R/\rho$ and $w_1 = L/\eta$, where $\rho, \eta > 0$ are arbitrary normalization factors, the first n Lanczos steps produce two sequences of vectors,

$$v_1, v_2, \ldots, v_n \quad \text{and} \quad w_1, w_2, \ldots, w_n, \tag{7}$$

such that

$$\begin{aligned} \operatorname{span}\{ v_1, v_2, \ldots, v_n \} &= \operatorname{span}\{ R, AR, \ldots, A^{n-1}R \}, \\ \operatorname{span}\{ w_1, w_2, \ldots, w_n \} &= \operatorname{span}\{ L, A^T L, \ldots, (A^T)^{n-1} L \}, \end{aligned} \tag{8}$$

and the vectors (7) are biorthogonal:

$$w_i^T v_l = 0 \quad \text{for all} \quad i \neq l, \ i, l = 1, 2, \ldots, n. \tag{9}$$

Setting

$$V_n = [\, v_1 \quad v_2 \quad \cdots v_n \,] \quad \text{and} \quad W_n = [\, w_1 \quad w_2 \quad \cdots w_n \,], \tag{10}$$

the matrices D_n and T_n in (6) are now defined as

$$D_n = W_n^T V_n \quad \text{and} \quad T_n = D_n^{-1} W_n^T A V_n.$$

Note that, by (9), the matrix D_n is diagonal. Moreover, the vectors (7) can be generated by three-term recurrences, and the entries of T_n are just the coefficients of these recursions. Consequently, the matrix T_n is tridiagonal. Finally, we remark that, in the classical Lanczos process [10], so-called breakdowns or near-breakdowns due to division by zero or a near-zero number can occur. However, these problems can be remedied by incorporating *look-ahead* techniques into the Lanczos process; see, e.g., [4]. The basic idea is to relax the biorthogonality (9) slightly whenever a breakdown or near-breakdown would occur in the classical Lanczos algorithm. To keep the exposition as simple as possible, in this paper, we always assume that breakdowns will not occur in the Lanczos process. However, we stress that both the Lanczos-Padé connection (6) and the result in Theorem 1 below also hold true if look-ahead Lanczos steps are performed to avoid breakdowns or near-breakdowns. Only the structure of the matrices D_n and T_n are affected by look-ahead steps. More precisely, in (6), D_n is block diagonal and T_n is block tridiagonal when look-ahead steps occur.

We now return to the general case $m, p \geq 1$. We say that a matrix-valued rational function $H_n\colon \mathbb{C} \longmapsto (\mathbb{C} \cup \{\infty\})^{p \times m}$ is of *type* $(n-1, n)$ if H_n can be represented as

$$H_n(z) = L_n^T \, (I - z A_n)^{-1} \, R_n, \tag{11}$$

with matrices $A_n \in \mathbb{C}^{n \times n}$, $R_n \in \mathbb{C}^{n \times m}$, and $L_n \in \mathbb{C}^{n \times p}$. Note that, for the case $m = p = 1$, the representation (6) of the nth Padé approximant H_n is just of the form (11), and thus H_n is of type $(n-1, n)$, in accordance with (5). We remark that a matrix-valued rational function of type $(n-1, n)$ can always be written in terms of a pair of matrix polynomials, in analogy to (5). However, since matrix polynomials do not commute and also for other reasons, such a representation involves a number of difficulties (see the discussion in [13]) that can be avoided by using (11). Furthermore, the representation (11) is also natural, in that it reflects the intimate connection between matrix Padé approximants of transfer functions and the minimal partial realization problem in systems theory; see, e.g., [9,12].

§3. A Lanczos-type process

A Lanczos-type process connected to matrix Padé approximants of transfer functions H given by (1) needs to be able to handle multiple starting vectors, namely the columns of the matrices R and L in (1). Such an algorithm was recently proposed by Aliaga, Boley, Freund, and Hernández [1]. In this section, we give an outline of the procedure.

Like the classical Lanczos process, the algorithm for multiple starting vectors generates two sequences of vectors (7) that are again constructed to fulfill the biorthogonality relations (9). However, instead of (8), the vectors (7) now span the subspaces generated by the first n linearly independent columns of the *Krylov matrices*

$$\begin{aligned} K^{(r)} &= [\, R \quad AR \quad A^2R \quad \cdots \quad A^{N-1}R \,] \\ \text{and} \quad K^{(l)} &= [\, L \quad A^TL \quad (A^T)^2L \quad \cdots \quad (A^T)^{N-1}L \,], \end{aligned} \tag{12}$$

respectively. In particular, the process of constructing the vectors (7) now also needs to be able to detect linearly dependent columns in (12) and to delete these vectors. We refer to this process as *deflation*. In our Lanczos process, a linearly dependent vector occurs if, and only if, $v_n = 0$ or $w_n = 0$, and deflation simply consists of deleting v_n or w_n, respectively. As a result, the Lanczos vectors (7) now span the spaces generated by the first n columns of the *deflated* Krylov matrices

$$\begin{aligned} K_{\text{defl}}^{(r)} &= [\, R_1 \quad AR_2 \quad A^2R_3 \quad \cdots \quad A^{j_{\max}-1}R_{j_{\max}} \,] \\ \text{and} \quad K_{\text{defl}}^{(l)} &= [\, L_1 \quad A^TL_2 \quad (A^T)^2L_3 \quad \cdots \quad (A^T)^{k_{\max}-1}L_{k_{\max}} \,], \end{aligned} \tag{13}$$

respectively. The matrices (13) are obtained from (12) by deleting all linearly dependent vectors that one encounters when scanning the columns of (12) from left to right. In (13), $R_j \in \mathbb{C}^{N\times m_j}$ and $L_k \in \mathbb{C}^{N\times p_k}$ are submatrices of R and L, respectively. Moreover, R_j is a submatrix of R_{j-1}, and L_j is a submatrix of L_{j-1}, so that $m_1 \geq m_2 \geq \cdots \geq m_{j_{\max}} \geq 1$ and $p_1 \geq p_2 \geq \cdots \geq p_{k_{\max}} \geq 1$.

We now set $N_0 = \min\{\operatorname{rank} K^{(r)}, \operatorname{rank} K^{(l)}\}$. Then, for each integer n with $\max\{m_1, p_1\} \leq n \leq N_0$, we define $j = j(n)$ and $k = k(n)$ as the maximal integers such that

$$m_1 + m_2 + \cdots + m_j \leq n \quad \text{and} \quad p_1 + p_2 + \cdots + p_k \leq n, \tag{14}$$

respectively. The integers $j(n)$ and $k(n)$ will be used in §4 to state our main result. We remark that if no deflation occurs, then $j(n) = \lfloor n/m \rfloor$ and $k(n) = \lfloor n/p \rfloor$.

After these preliminaries, we can now present a sketch of the actual algorithm. First, the starting vectors are biorthogonalized, resulting in relations of the following form:

$$R = [\, v_1 \quad v_2 \quad \cdots \quad v_{m_1} \,]\, \rho \quad \text{and} \quad L = [\, w_1 \quad w_2 \quad \cdots \quad w_{p_1} \,]\, \eta,$$

where $\rho \in \mathbb{C}^{m_1 \times m}$ and $\eta \in \mathbb{C}^{p_1 \times p}$. Moreover, we initialize the indices $m_c = m_1$ and $p_c = p_1$, which are used to keep track of the current block sizes; m_c respectively p_c are reduced by 1 every time deflation occurs.

Then, for $n = 1, 2, \ldots, N_0 - \max\{m_c, p_c\}$, we repeat the following steps:

$$\text{Set} \quad v_{n+m_c} = Av_n - \sum_{i=n-p_c}^{n+m_c-1} v_i \, \frac{w_i^T A v_n}{w_i^T v_i};$$

$$\text{If } v_{n+m_c} = 0, \text{ then delete } v_{n+m_c} \text{ and set } m_c = m_c - 1;$$

$$\text{Set} \quad w_{n+p_c} = A^T w_n - \sum_{i=n-m_c}^{n+p_c-1} w_i \, \frac{v_i^T A^T w_n}{w_i^T v_i};$$

$$\text{If } w_{n+p_c} = 0, \text{ then delete } w_{n+p_c} \text{ and set } p_c = p_c - 1.$$

Using the matrices V_n and W_n introduced in (10), the above recurrences to generate the Lanczos vectors (7) can be summarized in compact matrix form as follows:

$$\begin{aligned} AV_n &= V_n T_n + [\, \underbrace{0 \quad 0 \quad \cdots \quad 0}_{n-m_c} \quad \underbrace{\star \quad \star \quad \cdots \quad \star}_{m_c} \,], \\ A^T W_n &= W_n \tilde{T}_n^T + [\, \underbrace{0 \quad 0 \quad \cdots \quad 0}_{n-p_c} \quad \underbrace{\star \quad \star \quad \cdots \quad \star}_{p_c} \,]. \end{aligned} \tag{15}$$

Here, T_n and $\tilde{T}_n$ are $n \times n$ banded matrices with lower bandwidth $m+1$ and upper bandwidth $p+1$. Moreover, each deflation step reduces the lower or upper bandwidth by 1. Finally, the matrices T_n and $\tilde{T}_n$ are connected by the relation

$$\tilde{T}_n = D_n T_n D_n^{-1}, \quad \text{where } D_n = W_n^T V_n. \tag{16}$$

§4. The connection to matrix Padé approximation

We continue to use the notation introduced in §3. In particular, ρ, η, D_n, and V_n are the quantities generated by the Lanczos-type process sketched in §3. In addition, we denote by $0_{i \times l}$ the $i \times l$ zero matrix. Our result about the connection between the Lanczos-type process and matrix Padé approximants can now be stated as follows.

Theorem 1. *Let $\max\{m_1, p_1\} \leq n \leq N_0$, and let $j = j(n)$ and $k = k(n)$ be the integers defined in (14). Then,*

$$H_n(z) = \begin{bmatrix} D_{p_1}^T \eta \\ 0_{(n-p_1)\times p} \end{bmatrix}^T (I - zT_n)^{-1} \begin{bmatrix} \rho \\ 0_{(n-m_1)\times m} \end{bmatrix} \tag{17}$$

is an nth Padé approximant of the function H defined in (1), and

$$H_n(z) = H(z) + \mathcal{O}(z^{j+k}). \tag{18}$$

Moreover, if $n = m_1 + m_2 + \cdots + m_j$ or $n = p_1 + p_2 + \cdots + p_k$, then H_n is a Padé approximant of minimal order n that matches the first $j+k$ Taylor coefficients of H.

Proof: Using (15), the band structure of T_n and $\tilde{T}_n$, the definition of j and k in (14), and (16), one can verify the following relations:

$$A^{i_1} R = V_n T_n^{i_1} \begin{bmatrix} \rho \\ 0_{(n-m_1)\times m} \end{bmatrix}, \quad i_1 = 0, 1, \ldots, j-1,$$

$$\begin{aligned} L^T A^{i_2} &= [\, \eta^T \quad 0_{p\times(n-p_1)} \,]\, \tilde{T}_n^{i_2} W_n^T, \\ &= \begin{bmatrix} D_{p_1}^T \eta \\ 0_{(n-p_1)\times p} \end{bmatrix}^T T_n^{i_2} D_n^{-1} W_n^T, \quad i_2 = 0, 1, \ldots, k-1. \end{aligned}$$

These relations, together with (4) and $D_n = W_n^T V_n$, imply that

$$M_{i_1+i_2} = L^T A^{i_1+i_2} R = \begin{bmatrix} D_{p_1}^T \eta \\ 0_{(n-p_1)\times p} \end{bmatrix}^T T_n^{i_1+i_2} \begin{bmatrix} \rho \\ 0_{(n-m_1)\times m} \end{bmatrix} \tag{19}$$

for all $i_1 + i_2 = 0, 1, \ldots, j+k-2$. Using (15) and the fact that the Lanczos vectors are biorthogonal, it is possible to show that (19) even holds true for $i_1 + i_2 = j + k - 1$. Thus, by (19), the first $j+k$ coefficients of the Taylor expansions about $z = 0$ of the functions (3) and (17) match, and this concludes the proof of (18). In remains to verify that the number $j+k$ of matched coefficients is maximal so that H_n is indeed an nth Padé approximant. This can be done by exploiting the connection between matrix Padé approximants and the partial minimal realization problem and by showing that the criterion given in [6, Theorem 0.1] is satisfied. ∎

Remark Results of actual computations of matrix Padé approximants based on the representation (17) are presented in [3].

References

1. Aliaga, J. I., D. L. Boley, R. W. Freund, and V. Hernández, A Lanczos-type algorithm for multiple starting vectors, Numerical Analysis Manuscript, AT&T Bell Laboratories, Murray Hill, NJ, 1995.
2. Feldmann P. and R. W. Freund, Efficient linear circuit analysis by Padé approximation via the Lanczos process, in *Proc. EURO-DAC '94 with EURO-VHDL '94*, IEEE Computer Society Press, Los Alamitos, CA, 1994, 170–175.
3. Feldmann P. and R. W. Freund, Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm, Numerical Analysis Manuscript 94-11, AT&T Bell Laboratories, Murray Hill, NJ, 1995.
4. Freund, R. W., M. H. Gutknecht, and N. M. Nachtigal, An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices, SIAM J. Sci. Stat. Comput. **14** (1993), 137–158.
5. Gallivan, K., E. Grimme, and P. Van Dooren, Asymptotic waveform evaluation via a Lanczos method, Appl. Math. Lett. **7** (1994), 75–80.
6. Gohberg, I., M. A. Kaashoek, and L. Lerer, On minimality in the partial realization problem, Systems Control Lett. **9** (1987), 97–104.
7. Gragg, W. B., Matrix interpretations and applications of the continued fraction algorithm, Rocky Mountain J. Math. **4** (1974), 213–225.
8. Gutknecht, M. H., A completed theory of the unsymmetric Lanczos process and related algorithms, Part I, SIAM J. Matrix Anal. Appl. **13** (1992), 594–639.
9. Kailath, T., *Linear Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
10. Lanczos, C., An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Nat. Bur. Standards **45** (1950), 255–282.
11. Pillage, L. T. and R. A. Rohrer, Asymptotic waveform evaluation for timing analysis, IEEE Trans. Computer-Aided Design **9** (1990), 352–366.
12. Tether, A. J., Construction of minimal linear state-variable models from finite input-output data, IEEE Trans. Automat. Control AC-**15** (1970), 427–436.
13. Xu, G.-L. and A. Bultheel, Matrix Padé approximation: definitions and properties, Linear Algebra Appl. **137/138** (1990), 67–136.

Roland W. Freund
AT&T Bell Laboratories
600 Mountain Avenue, Room 2C–420
Murray Hill, New Jersey 07974–0636
freund@research.att.com