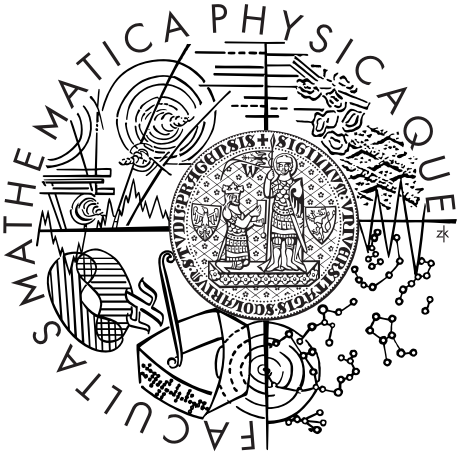
Charles University
Faculty of Mathematics and Physics

HABILITATION THESIS



Stefano Pozza

Decay phenomenon and quadratures in matrix function approximation

Department of Numerical Mathematics

Prague 2024

WWW: http://www.karlin.mff.cuni/~pozza
e-mail: pozza@karlin.mff.cuni.cz

Contents

Preface

Let $f(A)$ be a matrix function, with A a sparse matrix and f a regular enough scalar function. This thesis first analyzes the *decay phenomenon*, i.e., the phenomenon in which the magnitude of $f(A)$'s elements decays exponentially to zero in certain parts of the matrix function. Second, it describes and studies the use of Krylov subspace methods for approximating the bilinear form $w^H f(A)v$, connecting the methods with the Gauss quadrature. The decay phenomenon and results on the Gauss quadrature are at the core of a new approach to compute (symbolically and numerically) the *time-ordered exponential*, a time-dependent generalization of the matrix function concept emerging from the solution of systems of linear non-autonomous ordinary differential equations. This new approach is illustrated in the third part of the thesis.

The main part of the thesis contains a selection of 11 published papers contributing to these topics co-authored by Stefano Pozza (Charles University). They are the results of the collaborations with Pierre-Louis Giscard (University of the Littoral Opal Coast), Davide Palitta (Bologna University), Miroslav Pranić (Banja Luka University), Valeria Simoncini (Bologna University), Zdeněk Strakoš (Charles University), Francesco Tudisco (Edinburgh University), and Niel Van Buggenhout (University Carlos III).

The thesis explains how to extend several properties and techniques from the case in which A is a Hermitian matrix to the non-Hermitian and the time-dependent ones. To exemplify the content, consider the linear ordinary differential equation

$$\frac{\partial}{\partial t}u(t) = Au(t), \quad u(0) = v, \quad t \geq 0,$$

whose solution is given through the matrix exponential as $u(t) = \exp(At)v$. When A is Hermitian, the bilinear form $v^H \exp(At)v$ can be approximated by running the Lanczos algorithm with A, b as inputs. The nth iteration of the algorithm produces the smaller $n \times n$ tridiagonal matrix J_n giving the approximation

$$v^H \exp(At)v \approx e_1^T \exp(J_n t)e_1,$$

with e_1 the first vector of the canonical basis. Since A is Hermitian, $v^H \exp(At)v$ is equivalent to a certain Riemann-Stieltjes integral and $e_1^T \exp(J_n)e_1$ to the related Gauss quadrature, as well-explained in [60]. Moreover, the off-diagonal elements of $\exp(J_n)$ decay to zero, moving away from the diagonal. This decay can be exploited to enhance Krylov subspace methods [17]. Extending these results to non-Hermitian matrices is not trivial since many valuable properties are lost, e.g., $v^H \exp(At)v$ is not a Riemann-Stieltjes integral anymore. The extension becomes even more cumbersome when the

matrix A is time-dependent, i.e., the A's coefficients are functions of t. In this case, the differential equation becomes the non-autonomous equation:

$$\frac{\partial}{\partial t}u(t) = A(t)u(t), \quad u(0) = v, \quad t \geq 0.$$

The solution $u(t)$ is no longer given in terms of the matrix exponential but through the so-called time-ordered exponential. This requires the use of more convoluted expressions for $u(t)$ and a larger computational cost. The thesis also studies the extension of decays and approximation results to the rational Krylov subspace method.

Chapter 1 summarizes our contributions to analyzing the decay of the entries of non-Hermitian matrix functions and of the reduced-order matrix produced by the rational Krylov subspace method. It includes the following three papers:

[C1] S. Pozza, and F. Tudisco, *On the stability of network indices defined by means of matrix functions*, SIAM J. Matrix Anal. Appl. 39 (4) (2018), pp. 1521–1546. DOI: 10.1137/17M1133920

[C2] S. Pozza, and V. Simoncini, *Inexact Arnoldi residual estimates and decay properties for functions of non-Hermitian matrices*, BIT 59 (2019), pp. 969–986. DOI: 10.1007/s10543-019-00763-6

[C3] S. Pozza, and V. Simoncini, *Functions of rational Krylov space matrices and their decay properties*, Numer. Math. 148 (2021), pp. 99–126. DOI: 10.1007/s00211-021-01198-4

The paper [C2] presents upper bounds to describe the decay phenomenon in banded non-Hermitian matrix functions. The bounds are then exploited to obtain residual-type estimates for the Arnoldi's algorithm approximation of a matrix function, which are, in turn, employed to improve the inexact version of the Arnoldi's algorithm. Similar bounds are used in the paper [C1], where the decay phenomenon is exploited to estimate the changes in a matrix function $f(A)$ when A is subjected to a sparse perturbation. The estimate is useful in network analysis, where matrix functions are employed to identify the "most important" nodes, and a sparse perturbation of the matrix corresponds to modifying a few network edges. Paper [C3] derives upper bounds for the entries of functions of the reduced-order matrix produced by the rational Krylov subspace method. In contrast to the analogous reduced-order matrix of Arnoldi's algorithm treated in [C2], the rational Krylov reduced-order matrix is generally not sparse.

Chapter 2 outlines our work on the connections between the Lanczos algorithm and the Gauss quadrature in the non-Hermitian case. It also presents our contribution to the rational Lanczos algorithm and briefly points out the connection with the rational Gauss quadrature. It includes the following four papers:

[C4] S. Pozza, M. S. Pranić, and Z. Strakoš, *Gauss quadrature for quasi-definite linear functionals*, IMA J. Numer. Anal. 37 (3) (2017), pp. 1468–1495. DOI: 10.1093/imanum/drw032

[C5] S. Pozza, M. S. Pranić, and Z. Strakoš, *The Lanczos algorithm and complex Gauss quadrature*, Electron. Trans. Numer. Anal. 50 (2018), pp. 1–19. DOI: 10.1553/etna_vol50s1

[C6] S. Pozza, and M. S. Pranić, *The Gauss quadrature for general linear functionals, Lanczos algorithm, and minimal partial realization*, Numer. Algorithms 88 (2021), pp. 647–678.
DOI: 10.1007/s11075-020-01052-y

[C7] D. Palitta, S. Pozza, and V. Simoncini, *The short-term rational Lanczos method and applications*, SIAM J. Sci. Comput. 44 (4) (2022), pp. A2843–A2870.
DOI: 10.1137/21M1403254

The papers [C4, C5, C6] explain how to extend the main results of the monograph [60] to the non-Hermitian case, specifically, the extension of the Gauss quadrature to linear functional approximation and its connection with the non-Hermitian Lanczos algorithm. As it is well-known, the non-Hermitian Lanczos algorithm is affected by breakdowns. In [C6], we illustrate the relation between breakdowns and the extended Gauss quadrature and how the look-ahead strategy theoretically resolves the breakdown problem from the Gauss quadrature point of view. The paper [C7] proposes a more efficient implementation of the rational Lanczos algorithm, explains the connection with the rational Gauss quadrature, and discusses several applications. Moreover, it presents preliminary results on the loss of orthogonality of the algorithm in finite precision arithmetic.

Chapter 3 compactly presents our work on the analytical expression and numerical computation of the time-ordered exponential. It includes the following four papers:

[C8] P-L. Giscard, and S. Pozza, *Tridiagonalization of systems of coupled linear differential equations with variable coefficients by a Lanczos-like method*, Linear Algebra Appl. 624 (2021), pp. 153–173.
DOI: 10.1016/j.laa.2021.04.011

[C9] P-L. Giscard, and S. Pozza, *A Lanczos-like method for non-autonomous linear ordinary differential equations*, Boll. Unione Mat. Ital. 16 (2023), pp. 81–102.
DOI: 10.1007/s40574-022-00328-6

[C10] S. Pozza, *A new closed-form expression for the solution of ODEs in a ring of distributions and its connection with the matrix algebra*, Linear Multilinear Algebra (2024), online.
DOI: 10.1080/03081087.2024.2303058

[C11] S. Pozza, and N. Van Buggenhout, *A new Legendre polynomial-based approach for non-autonomous linear ODEs*, Electron. Trans. Numer. Anal. 60 (2024), pp. 292–326
DOI: 10.1553/etna_vol60s292

The papers [C8, C9] provide the first exact expression for the elements of the time-ordered exponential in terms of a polynomial number of integro-differential equations. This is done by introducing a new algebra of distributions (the $\star$-algebra [106]) where a system of non-autonomous linear ordinary differential equations is transformed into a system of $\star$-algebra linear equations. The $\star$-algebra linear system is then solved through the so-called $\star$-Lanczos algorithm (a symbolic method). The $\star$-Lanczos algorithm and the related properties are obtained by extending the results of Chapter 2. In [C10],

the $\star$-algebra system is transformed into a (usual) linear system involving infinite-size matrices. These matrices are characterized by an off-diagonal decay phenomenon, which can be exploited for numerical computations. By appropriately truncating the infinite-size matrices, the paper [C11] derives and analyzes a new spectral method for the solution of non-autonomous ordinary differential equations.

Unless otherwise specified, the algorithms are presented as working in exact arithmetic. Their effectiveness in practical computations is shown by numerical experiments included in the papers. The numerical analysis of rounding errors in our original methods, techniques, and variants is an open issue and constitutes a long-term goal for our work. The main difficulties in this analysis are related to Krylov subspace methods with short recurrences.

This work has been possible thanks to the support of Charles University Research programs PRIMUS/21/SCI/009, UNCE/SCI/023, and UNCE/24/SCI/005, and of the Magica project ANR-20-CE29-0007 funded by the French National Research Agency.

I want to thank Iveta Hnětynková, Václav Kučera, and Zdeněk Strakoš for their help, precious suggestions, and useful comments. A special thanks also to Alessandro Benassi and Marguerite G. Farag for their support and their help in the editorial revision.

Chapter 1

Matrix functions and decay phenomenon

1.1 Introduction

In matrix computations, matrices are often characterized as being *dense* or *sparse*. Such categories do not have a formal definition. Still, we can think of a sparse matrix as having enough zero elements to be conveniently exploitable (while a dense matrix is a matrix that is not sparse). From the numerical point of view, the notion of sparsity is useful but limited since it ignores the magnitude of the nonzero elements. By rounding to zero the elements with a small enough magnitude (truncation), it is possible to reduce the computational cost with a negligible loss of accuracy. This is especially true when working with dense matrices where most elements are nearly zero (*numerically sparse* matrices). Knowing where the elements are close to zero can substantially improve numerical techniques for linear systems, eigenvalues, matrix functions, and other matrix problems; see, for instance, [9] and references therein. The large-magnitude elements of a numerically sparse matrix are often localized in some parts of the matrix (for example, the main diagonals). Moving away from these parts, the magnitude of the elements typically tends to decay to zero. This is known as *decay phenomenon*. A classical case is the inverse of banded matrices (first analyzed in [43]).

Example 1.1 (Matrix inverse decay) *Consider the* 60×60 *tridiagonal symmetric positive definite matrix*

$$A = \begin{bmatrix} 3 & 1 & 0 & \cdots & 0 \\ 1 & 3 & 1 & & \vdots \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & 3 \end{bmatrix}.$$

While A^{-1} *is dense, a closer inspection reveals that the magnitude of most of the elements is negligible; see Figure 1.1, left. The significant elements of* A^{-1} *are found around the band of* A *(localization). Moreover, the magnitude of the elements exponentially decays to zero with the distance from the main diagonal (off-diagonal decay).*

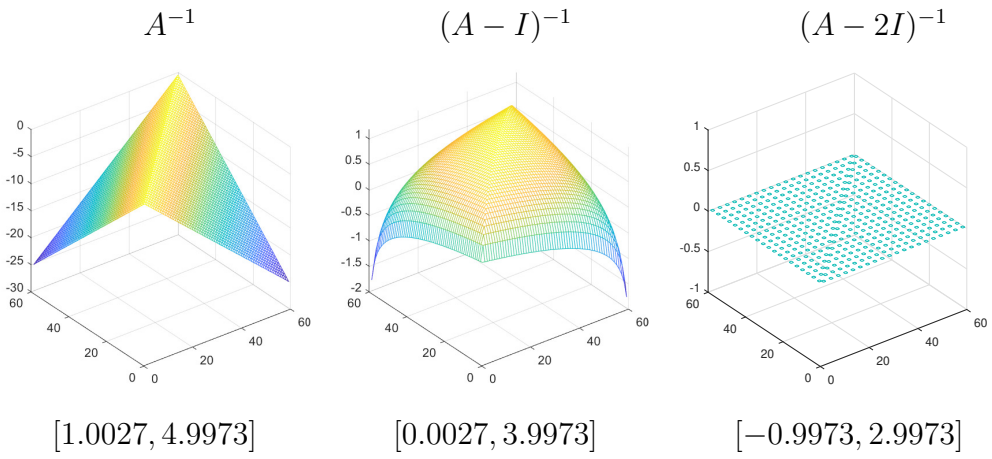


Figure 1.1: Above: Plots of the magnitude of the matrix inverse elements (logarithmic scale). Below: Spectral intervals of, respectively, $A, A-I$, and $A-2I$.

In Figure 1.1, we can also see the influence of the matrix spectrum on the decay phenomenon. Indeed, the inverses of the shifted matrices $A-I, A-2I$ display a different behavior. The elements of $(A-I)^{-1}$ are barely decaying to zero, while in $(A-2I)^{-1}$ the decay phenomenon has disappeared.

The decay phenomenon extends beyond the matrix inverse case to the more general case of matrix functions. Matrix functions are well-established tools for modeling and solving problems (analytically and numerically) emerging from the most diverse applications; see, e.g., [72]. For instance, the applications considered in this thesis span from network analysis [10, 12, 14, 46–48] to optimization problems in Gaussian processes [101] to systems of non-autonomous differential equations in quantum chemistry [19, 25, 56, 57, 73, 78].

Matrix functions are defined in several ways that are equivalent under certain assumptions [72, Section 1]. Below, we recall the definitions used in the thesis papers' reprints. First, let us denote with λ_i the eigenvalues of the matrix A and with s_i the associated *index* (i.e., the size of the largest Jordan block in which λ_i appears). A function f is said to be *defined on the spectrum of the matrix A* if, for every λ_i, the value of the jth derivative $f^{(j)}(\lambda_i)$ exists for $j = 0, 1, \ldots, s_i - 1$. We use the notation $\mathrm{diag}(X_1, \ldots, X_n)$ to denote a (block) diagonal matrix whose diagonal elements (blocks) are $X_1, \ldots, X_n$.

Definition 1.2 (Matrix function via Jordan canonical form) *Let f be a function defined on the spectrum of the matrix $A \in \mathbb{C}^{n\times n}$. Moreover, consider the Jordan decomposition of A*

$$A = W\mathrm{diag}(\Lambda_1, \ldots, \Lambda_\nu)W^{-1},$$

where $\Lambda_1, \ldots, \Lambda_\nu$ are the Jordan blocks and W the matrix of the generalized eigenvectors. The matrix function $f(A)$ *is defined as*

$$f(A) = W\mathrm{diag}(f(\Lambda_1), \ldots, f(\Lambda_\nu))W^{-1},$$

where

$$f(\Lambda) = \begin{bmatrix} f(\lambda) & \frac{f'(\lambda)}{1!} & \frac{f^{(2)}(\lambda)}{2!} & \dots & \frac{f^{(s-1)}(\lambda)}{(s-1)!} \\ 0 & f(\lambda) & \frac{f'(\lambda)}{1!} & \dots & \frac{f^{(s-2)}(\lambda)}{(s-2)!} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \frac{f'(\lambda)}{1!} \\ 0 & \dots & \dots & 0 & f(\lambda) \end{bmatrix}$$

is the function of an s-size Jordan block Λ with eigenvalue λ.

Definition 1.3 (Matrix function via Cauchy integral) *Let $A \in \mathbb{C}^{n\times n}$ and f be an analytic function on an open set $\Omega \subset \mathbb{C}$ and let $\Gamma \subset \Omega$ be a system of Jordan curves encircling each eigenvalue of A exactly once, with mathematical positive orientation. Then the matrix function $f(A)$ is defined as*

$$f(A) = \int_\Gamma f(z)\,(zI - A)^{-1}\,dz.$$

Definitions 1.2 and 1.3 are equivalent when f is analytic; see [72, Theorem 1.12]. Assume that the analytic function f can be expanded into the power series $f(z) = \sum_{n\geq 0} \theta_n\, z^n$ for $|z| \leq r$, with convergence radius $r > 0$. If r is larger than the spectral radius of A, then the expansion $\sum_{n\geq 0} \theta_n\, A^n$ converges to $f(A)$ in the sense of Definitions 1.2 and 1.3; see, e.g., [72, Theorem 4.7]. Note that if $f(z) = a_0 + a_1 z + \dots a_n z^n$ is a polynomial of degree n, the matrix function $f(A)$ is in fact the matrix polynomial

$$f(A) = a_0 I + a_1 A + \dots a_n A^n,$$

with I the identity matrix. Moreover, the usual matrix inverse A^{-1} is a matrix function, as are the matrix rational functions in the form $p(A)(q(A))^{-1}$, with p and q polynomials. Below, we present an introductory example using the matrix exponential $\exp(A)$. Other matrix functions found in the thesis are $\exp(-\sqrt{A}), \log(A), \sqrt{A}$.

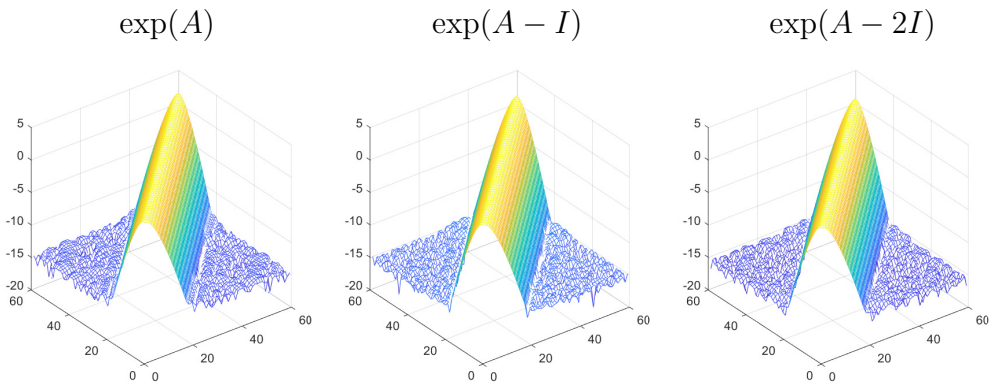


Figure 1.2: Magnitude of the matrix exponential elements (logarithmic scale). The matrix A is the same as in Figure 1.1.

Example 1.4 (Matrix Exponential) *A classic example of a matrix function is the matrix exponential* $\exp(A)$*, which is typically introduced using the expansion*

$$\exp(A) = \sum_{j=0}^{\infty} \frac{A^j}{j!}.$$

Since the exponential is an entire function and the series above converges for every $A \in \mathbb{C}^{n\times n}$*, this definition is equivalent to the Definitions 1.2 and 1.3. In this thesis, the matrix exponential is used in network analysis applications as well as for solving ordinary differential equations.*

Consider the matrix A *from Example 1.1. Figure 1.2 displays the decay phenomenon for* $\exp(A)$*. The decay is similar to the one in* A^{-1} *(Figure 1.1) but faster since the magnitude of the elements decays superexponentially. As explained, e.g., in [8], this is because the exponential is an entire function. Moreover, note that there is no significant difference in the decay rate of the exponential of the shifted matrices* $\exp(A-I), \exp(A-2I)$*.*

In general, given a large enough banded Hermitian matrix A and a function f analytic on an ellipse containing the spectrum of A, $f(A)$ shows an off-diagonal exponential decay; see [15]. Similar conditions apply for the off-diagonal decay of banded non-Hermitian matrix functions; see, e.g., [8, 11] and [C2]. Describing the decay phenomenon a priori, i.e., without computing $f(A)$, is important as it can be used to devise algorithms with a lower numerical and memory cost; e.g., [9, 16] and [C2]. Other applications are found, for instance, in [104]. The a priori description of the decay phenomenon usually employs upper bounds in the following, or equivalent, form:

$$|f(A)_{k,\ell}| \leq K \exp(-\alpha|k-\ell|), \quad k,\ell = 1,\dots,n; \tag{1.1}$$

for some $K, \alpha > 0$; see, among many other, [8, 9, 15, 43, 53] and [C2]. The values of α and K usually depend on i) the upper and lower bandwidths of A, ii) properties of the function f (e.g., singularities), iii) spectral properties of A (e.g., the spectral interval). When f is an entire function, α can increase with the distance $|k-\ell|$, leading to a super-exponential decay rate (Figure 1.2); see also [C2, Corollary 2.4] for an example of a superexponential bound.

Relying on the spectral interval as the only source of spectral information might be insufficient for nonnormal matrices as it might lead to values of K in (1.1) that are too large to be meaningful. For this reason, more advanced bounds (for instance, in [8, 11, C2]) are based on information from the field of values (numerical range), i.e., the convex set in the complex field defined as

$$W(A) := \{v^H A v \,|\, v \in \mathbb{C}^n, ||v|| = 1\}.$$

The decay phenomenon extends to functions of sparse matrices (Section 1.2) and even to some special dense matrices (Section 1.3.1). Section 1.3 discusses some of its uses in Krylov subspace methods.

1.2 Decay phenomenon in network analysis

The decay phenomenon displayed by functions of sparse matrices can be described using graph theory; see, e.g., [9, 11, 12]. Given a sparse $n \times n$ matrix A, the associated

graph (or network) $G(A)$ is the graph with n nodes which has a directed edge from the node k to the node ℓ if and only if the matrix element $A_{k,\ell}$ is nonzero (if the elements of A are either 0 or 1, then A is the adjacency matrix of $G(A)$). Given the nodes k, ℓ in $G(A)$, the number of edges (the *length*) of the shortest path from k to ℓ is known as the *geodesic* (or *shortest-path*) distance and denoted by $\mathsf{d}_G(k, \ell)$. Then, typically, an a priori decay bound for a matrix function $f(A)$ is obtained by replacing $|k - \ell|$ with $\mathsf{d}_G(k, \ell)$ in (1.1), that is,

$$|f(A)_{k,\ell}| \leq K \exp(-\alpha \mathsf{d}_G(k, \ell)), \quad k, \ell = 1, \ldots, n;$$

see, e.g., [11].

In network analysis, the *centrality index* of a node is a value displaying the node's "importance" with respect to the edge structure of the network. In other words, the centrality index shows which node is more likely to be at the center of information flux; see [12, 14, 45–48]. In particular, the so-called *subgraph centrality index* is defined through matrix functions as

$$SC(k) := [f(A)]_{k,k},$$

that is the kth element of the diagonal of a matrix function. Typically, f is either the resolvent $f(A) = (I - \alpha A)^{-1}$ or the exponential $f(A) = \exp(A)$ [46, 47]. The off-diagonal elements of $f(A)$ define the so-called *subgraph communicability*.

Computing $SC(k)$ can be costly when the size of A is huge, as often happens in network analysis. This becomes even more problematic when the network is subjected to frequent changes or perturbation since, in principle, the nodes' centrality index must be updated. In [C1], with Francesco Tudisco, we showed that the centrality index of a node k is not substantially affected by edge perturbation if the perturbed edges are far away from k. More in detail, suppose a few edges are added, removed, or modified in $G(A)$. Then, we get the perturbed network $G(\widetilde{A})$ with adjacency matrix $\widetilde{A} = A + \Delta A$, where ΔA is a very sparse matrix. By exploiting the decay phenomenon, in [C1], we derived bounds of the kind:

$$|f(A)_{k,k} - f(\tilde{A})_{k,k}| \leq K \left(\frac{1}{\tau}\right)^{\delta+2}, \quad k = 1, \ldots, n,$$

for a certain $\tau > 1$ and $K > 0$, where δ is the geodesic distance between k and the set of nodes whose edges have been perturbed. These bounds explain why peripheral changes in the network $G(A)$ do not significantly influence the subgraph centrality index of the most important nodes. The paper [C1] also extends these results to communicability indexes.

1.3 Arnoldi's residual estimates and decay phenomenon

We start briefly recalling the basic concepts of Arnoldi's method; the reader can refer to the monographs [89, 107] for more information. Given a matrix $A \in \mathbb{R}^{N \times N}$ and a vector $v \neq 0$, Arnoldi's method constructs an orthogonal matrix $U_m = [u_1, \ldots, u_m]$ whose columns are a basis of the (polynomial) Krylov subspace

$$\mathcal{K}_m(A, v) := \operatorname{span}\left\{v, A\,v, \ldots, A^{m-1}\,v\right\}.$$

Arnoldi's method is a Gram-Schmidt orthogonalization process defined by the recurrences

$$h_{j+1,j}u_{j+1} = Au_j - \sum_{i=1}^{j} h_{i,j}u_i, \quad j = 1, \ldots, m, \quad u_1 = v/\|v\|, \tag{1.2}$$

$$h_{i,j} = u_i^H A u_j, \quad h_{j+1,j} = \|u_{j+1}\|, \tag{1.3}$$

which are *long recurrences* since the number of terms increases at each iteration. The recurrences can be compactly rewritten in the matrix form

$$AU_m = U_m H_m + h_{m+1,m}u_{m+1}e_m^T, \tag{1.4}$$

where H_m is the $m \times m$ upper Hessenberg matrix with the coefficients $h_{i,j}$ as entries (e_m is the mth vector of the canonical basis). Moreover, by U_m orthogonality, it holds $H_m = U_m^H A U_m$. The matrix H_m plays two roles:

- It represents the orthogonalization process since it contains the coefficients $h_{i,j}$;
- It represents the action of A in the Krylov subspace $\mathcal{K}_m(A, v)$ since $U_m H_m U_m^H = U_m U_m^H A U_m U_m^H$, with $U_m U_m^H$ orthogonal projector onto $\mathcal{K}_m(A, v)$.

When A is a large sparse matrix, the upper Hessenberg matrix H_m can be used for the approximation of matrix functions through the formula

$$f(A)v \approx \|v\| U_m f(H_m) e_1, \tag{1.5}$$

where the problem $f(H_m)e_1$ is of reduced size and hence easier to compute (model reduction); see, e.g., [72, 96]. Note that when $f(z) = z^{-1}$, this approach is mathematically equivalent to the Full Orthogonalization Method (FOM). Since H_m is a banded matrix (in its lower triangular part), $f(H_m)$ elements may decay. In this case, a priori decay bounds can be used, for instance, for devising relaxed approaches and stopping criteria for iterative solvers in matrix function evaluations and matrix equation problems; see, e.g., [70, 85, 113, 123, 124]. In [C2], with Valeria Simoncini, we derived an a priori bound for the residual of the Arnoldi approximation (1.5) of certain matrix functions $f(A)$. The residual bound was derived through a bound for the elements' decay in $f(H_m)$ based on information on the field of values $W(A)$.

The *inexact Arnoldi's method* is a variant of Arnoldi's method in which the matrix-vector product Au_j in the iterations (1.2)–(1.3) is not computed accurately (e.g., because A is not explicitly known). Given a requested tolerance for the norm of the residual, it is possible to increasingly relax the accuracy of the matrix-vector product as Arnoldi's iterations proceed without significantly changing the residual convergence behavior [110, 113]. In [C2], the introduced decay bound for $f(H_m)$ is also used to devise a strategy for setting the iteration accuracy of the inexact Arnoldi's method when approximating non-Hermitian matrix functions by (1.5).

1.3.1 Rational Krylov subspace method

The model-reduction approach for matrix-function approximation in (1.5) corresponds to a polynomial approximation of $f(A)v$ since $\|v\|U_m f(H_m)e_1$ is a vector in the (polynomial) Krylov subspace. When the polynomial approximation is insufficient (for example, the number of Arnoldi iterations is too large), a rational Krylov approximation may work. Hereafter, we summarize the basics of rational Krylov approximation.

More information can be found in [68, 69, 105]. Analogously to the Arnoldi's algorithm, the *rational Krylov subspace method* (RKSM) produces the orthogonal matrix $V_m = [v_1, \dots, v_m]$ basis of the the rational Krylov subspace

$$\mathcal{R}_m(A, v, \boldsymbol{\sigma}) := \text{span}\left\{ v, (A-\sigma_1 I)^{-1} v, \dots, \prod_{j=1}^{m-1} (A-\sigma_j I)^{-1} v \right\},$$

for the given shifts $\boldsymbol{\sigma} = [\sigma_1, \dots, \sigma_{m-1}]$ not in the spectrum of A. The RKSM is a Gram-Schmidt orthogonalization process defined by the recurrences

$$h_{j+1,j} v_{j+1} = (A - \sigma_j I)^{-1} v_j - \sum_{i=1}^{j} h_{i,j} v_i, \quad v_1 = v/\|v\|, \tag{1.6}$$

$$h_{i,j} = v_i^H (A - \sigma_j I)^{-1} v_j, \quad h_{j+1,j} = \|v_{j+1}\|, \tag{1.7}$$

for $j = 1, \dots, m$. By defining H_m as the upper Hessenberg matrix with entries $h_{i,j}$, and $K_m = (I + H_m \, \text{diag}(\sigma_1, \dots, \sigma_m))$, these long recurrences can be rewritten in the matrix form

$$A V_m H_m = V_m K_m - h_{m+1,m} (A - \sigma_m I) v_{m+1} e_m^T. \tag{1.8}$$

In RKSM, the matrix H_m represents the orthogonalization process (it contains the coefficients $h_{i,j}$). On the other side, the so-called *reduced-order matrix* is defined as

$$J_m := V_m^H A V_m = K_m \, H_m^{-1} - h_{m+1,m} V_m^H (A - \sigma_m I) \, v_{m+1} e_m^T \, H_m^{-1},$$

and it represents the action of A in $\mathcal{R}_m(A, v, \sigma)$. As such, the approximation of matrix functions is given by the formula

$$f(A)v \approx \|v\| V_m f(J_m) e_1,$$

analogously to (1.5); e.g., [68]. Contrary to the Hessenberg matrix in (1.5), the matrix J_m is generally dense. Despite this, the entries' magnitude of J_m and of matrix functions $f(J_m)$ typically decay in their lower triangular part. In [C3], with Valeria Simoncini, we mathematically described the decay phenomenon in J_m and $f(J_m)$ by introducing a priori decay bounds, so proving that the matrix J_m is numerically banded and opening the way to improvements, such as new residual estimates and novel inexact approaches. The results in [C3] also helped us to analyze RKSM finite precision convergence behavior in [C7]; see Section 2.4. The a priori bounds were obtained by exploiting i) the hidden sparsity structure of J_m (a consequence of the orthogonalization process) ii) rational function approximation iii) the domain of analyticity of f iv) the field of values of A.

Another related application is the Lyapunov matrix equations of the kind

$$AX + XA^H = cc^H,$$

with c a vector of norm 1. Its solution can be approximated by solving the reduced-order equation

$$J_m Y_m + Y_m \, J_m^H = e_1 e_1^T, \quad X \approx V_m Y_m V_m^H; \tag{1.9}$$

see [68, 82, 111, 112]. Interestingly, the matrix Y_m is also localized in the upper left corner, as we proved in [C3] by providing a further a priori decay bound.

Chapter 2

Matrix functions, Krylov subspace methods, and the complex Gauss quadrature

2.1 The Gauss quadrature and Lanczos algorithm

We begin summarizing well-known results on the connection between matrix function approximation, the Gauss quadrature, and the Lanczos algorithm, as presented in the monograph [60]. Consider the Riemann-Stieltjes integral with respect to a non-decreasing distribution μ

$$\mathcal{I}_\mu(f) := \int_{\mathbb{R}} f(\lambda) d\mu(\lambda),$$

and assume that the moments $m_0, m_1, m_2, \dots$ of $\mathcal{I}_\mu$ are finite, i.e.,

$$m_j := \int_{\mathbb{R}} \lambda^j d\mu(\lambda) < \infty, \quad j = 0, 1, 2, \dots.$$

Then, the integral $\mathcal{I}_\mu(f)$ is well-defined for every f in $\mathcal{P}$, the space of the polynomials with complex coefficients. Given a function f for which $I_\mu(f)$ is well-defined, the integral can be approximated by an n-node quadrature

$$\int_{\mathbb{R}} f(\lambda)\, d\mu(\lambda) \approx \sum_{j=1}^{n} f(\theta_j)\, \omega_j =: \mathcal{G}_n(f),$$

determined by the nodes θ_j and weights ω_j. In particular, $\mathcal{G}_n$ is a *Gauss quadrature* when its *degree of exactness* is $2n-1$, the maximal one (it is exact for every polynomial f of degree smaller or equal to $2n-1$). We recall that the nodes θ_j of the Gauss quadrature $\mathcal{G}_n$ are the roots of the n-degree orthogonal polynomial p_n with respect to the distribution μ (once the nodes are determined, so are the weights ω_j since the Gauss quadrature is interpolatory). We refer to [37, 55] for more information.

Consider an $N \times N$ Hermitian matrix A and a vector v such that $\|v\| = 1$. For simplicity, assume that A's eigenvalues $\lambda_1 < \cdots < \lambda_N$ are distinct, and denote the

associated eigenvectors with $q_1, \ldots, q_N$. Let us define the discrete distribution

$$\mu(\lambda) := \begin{cases} 0, & \text{if } \lambda < \lambda_1 \\ \sum_{j=1}^{i} |q_j^H v|^2, & \text{if } \lambda_i \leq \lambda < \lambda_{i+1},\ i = 1, \ldots, N-1, \\ \sum_{j=1}^{N} |q_j^H v|^2, & \text{if } \lambda \geq \lambda_N \end{cases} . \tag{2.1}$$

Then, for every function f defined on the spectrum of A, the bilinear form can be rewritten as

$$v^H f(A) v = \int_{\lambda_1}^{\lambda_N} f(\lambda) \mathrm{d}\mu(\lambda) = \sum_{i=1}^{N} f(\lambda_i) |q_i^H v|^2. \tag{2.2}$$

Thanks to (2.2), we can approximate $v^H f(A) v$ by a quadrature rule. Moreover, since μ is the distribution (2.1), the Gauss quadrature $\mathcal{G}_n(f)$ can be evaluated using the Lanczos algorithm [86, 87] which computes the orthogonal matrix $V_n = [v_1, \ldots, v_n]$ whose columns form a basis of the Krylov subspaces $\mathcal{K}_n(A, v)$. The matrix $J_n = V_n^H A V_n$, is a tridiagonal Hermitian matrix known as *Jacobi matrix*. As explained in [60], it holds

$$v^H f(A) v = \int_{\mathbb{R}} f(\lambda)\, d\mu(\lambda) \approx \sum_{j=1}^{n} f(\theta_j)\, \omega_j = e_1^T f(J_n) e_1. \tag{2.3}$$

Therefore, it is possible to compute the Gauss quadrature $\mathcal{G}_n(f)$ by running $n-1$ iterations of the Lanczos algorithm and then compute the bilinear form $e_1^T f(J_n) e_1$. Note that this is a model reduction approach since J_n has a size smaller than A's one. This quadrature-based approach has been successfully applied to matrix function approximation; see, e.g., [6, 10, 12–15, 48, 49, 60, 114].

The following sections show how to generalize the Gauss quadrature to approximate general linear functionals $\mathcal{L} : \mathcal{P} \to \mathbb{C}$, presenting the connections with formal orthogonal polynomials, complex Jacobi matrices, and the non-Hermitian Lanczos algorithm. This allows extending the approximation (2.3) to non-Hermitian matrices.

2.2 Extending the Gauss quadrature

Let $\mathcal{L} : \mathcal{P} \to \mathbb{C}$ be the linear functional determined by the sequence of moments

$$\mathcal{L}(\lambda^j) = m_j, \quad j = 0, 1, \ldots,$$

and let us define the bilinear form $[\cdot, \cdot] : \mathcal{P} \times \mathcal{P} \to \mathbb{C}$ as $[p, q] := \mathcal{L}(pq)$. We say that a sequence of polynomials $p_0, \ldots, p_n$ with degrees respectively $0, \ldots, n$ is a sequence of *formal orthonormal polynomials* (FOPs) when $[p_i, p_j] = \mathcal{L}(p_i p_j) = \delta_{ij}$, with δ_{ij} the Kronecker delta. If such a sequence exists, then $\mathcal{L}$ is said to be *quasi-definite over* $\mathcal{P}_n \subset \mathcal{P}$, the subspace of polynomials with degree at most n; see [37]. Formal orthonormal polynomials satisfy the three-term recurrences

$$\beta_j p_j(\lambda) = (\lambda - \alpha_{j-1}) p_{j-1}(\lambda) - \beta_{j-1} p_{j-2}(\lambda), \quad j = 1, \ldots, n, \tag{2.4}$$

with $p_{-1}(\lambda) = 0$, $p_0(\lambda) \neq 0$, $\beta_j \neq 0$. The recurrences can be rewritten in matrix form:

$$\lambda p(\lambda) = J_n\, p(\lambda) + \beta_n p_n(\lambda)\, e_n$$

with

$$J_n = \begin{pmatrix} \alpha_0 & \beta_1 & & & \\ \beta_1 & \alpha_1 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{n-2} & \alpha_{n-2} & \beta_{n-1} \\ & & & \beta_{n-1} & \alpha_{n-1} \end{pmatrix}, \quad p = \begin{pmatrix} p_0(\lambda) \\ p_1(\lambda) \\ \vdots \\ p_{n-2}(\lambda) \\ p_{n-1}(\lambda) \end{pmatrix}. \tag{2.5}$$

The matrix J_n is known as *complex Jacobi matrix* [7] (note that it is symmetric but not generally Hermitian). For more information about FOPs, refer to [31, 37]. Contrary to the (classical) orthogonal polynomials [55], formal orthogonal polynomials can have roots with multiplicity larger than 1. Therefore, they generally take the form

$$p_n(\lambda) = \gamma(\lambda - \lambda_1)^{s_1}(\lambda - \lambda_2)^{s_2} \cdots (\lambda - \lambda_t)^{s_t}, \quad s_1 + \cdots + s_t = n.$$

Nevertheless, given a regular enough function f, it is possible to use the roots of p_n to define the so-called n-weight *complex Gauss quadrature* $\mathcal{G}_n(f)$ for approximating $\mathcal{L}(f)$, that is

$$\mathcal{G}_n(f) := \sum_{i=1}^{t} \sum_{j=0}^{s_i-1} \omega_{i,j}\, f^{(j)}(\lambda_i), \quad n = s_1 + \cdots + s_t, \tag{2.6}$$

with s_i the multiplicity of the node λ_i. Note that the weights are determined by λ_i and s_i since the quadrature is interpolatory (in the Hermite sense).

The complex Gauss quadrature has been introduced in different forms (equivalent under certain assumptions); see [44, 52, 95] and [C4]. In [C4], with Miroslav Pranić and Zdeněk Strakoš, we described the complex Gauss quadrature from the FOP point of view, assuming $\mathcal{L}$ to be quasi-definite. Moreover, we identified the following basic properties that make (2.6) a useful generalization of the Gauss quadrature in numerical linear algebra.

- G1. $\mathcal{G}_n$ has maximal degree of exactness $2n-1$.
- G2. $\mathcal{G}_n$ is well-defined and unique. Moreover, Gauss quadratures with a smaller number of weights also exist and are unique.
- G3. $\mathcal{G}_n(f) = m_0 e_1^T f(J_n)\, e_1$, where J_n is the complex Jacobi matrix.

Among other results, in [C4], we proved that $m_0 e_1^T f(J_n)\, e_1 = \mathcal{L}(f)$, for every polynomial f of degree at most $2n-1$. Other proofs of this property have been given in different forms and under more restrictive assumptions [38, 52, 62, 89, 115]. We demonstrated Property G3, i.e., that $\mathcal{G}(f) = m_0 e_1^T f(J_n) e_1$ for every $f \in \mathcal{P}$ (in other words, $m_0 e_1^T f(J_n) e_1$ is the matrix form of the complex Gauss quadrature). We further proved that $\mathcal{L}$ is quasi-definite on $\mathcal{P}_n$ if and only if G1 and G2 hold. The results in [C4] opened the way to the connection between the complex Gauss quadrature and the non-Hermitian Lanczos algorithm described in the following section.

2.3 The complex Gauss quadrature and Krylov subspace methods

Let A be a generally non-Hermitian matrix and w, v be vectors such that $w^H v = 1$. The linear functional

$$\mathcal{L} : \mathcal{P} \to \mathbb{C}, \quad \mathcal{L}(f) := w^H f(A)\, v$$

can be approximated by the complex Gauss quadrature (2.6) assuming that it is quasi-definite on $\mathcal{P}_n$. By Property G3, the quadrature can be computed through its matrix representation

$$e_1^T f(J_n) e_1 = \sum_{i=1}^{t} \sum_{j=0}^{s_i - 1} \omega_{i,j}\, f^{(j)}(\lambda_i), \tag{2.7}$$

note that $m_0 = 1$. In the survey [C5], with Miroslav Pranić and Zdeněk Strakoš, we examined the connections between the complex Gauss quadrature for $\mathcal{L}(f) := w^H f(A)\, v$, and the non-Hermitian Lanczos algorithm. Assuming no breakdowns, the non-Hermitian Lanczos algorithm computes the matrices $V_n = [v_1, \ldots, v_n]$ and $W_n = [w_1, \ldots, w_n]$ whose columns are bases, respectively, of the Krylov subspaces $\mathcal{K}_n(A, v)$ and $\mathcal{K}_n(A^H, w)$. The matrices satisfy the *biorthogonality* conditions $W_n^H V_n = I$, but they are not, generally, orthogonal. Since $\mathcal{L}$ is quasi-definite on $\mathcal{P}_n$, the associated FOPs $p_0, p_1, \ldots, p_n$ exist. Then, the non-Hermitian Lanczos algorithm bases can be expressed as

$$v_{j+1} = p_j(A) v \quad \text{and} \quad w_{j+1}^H = w^H p_j(A), \quad j = 0, \ldots, n-1.$$

As a consequence, the non-Hermitian Lanczos recurrences

$$\beta_j v_j = (A - \alpha_{j-1}) v_{j-1} - \beta_{j-1} v_{j-2}. \tag{2.8}$$

$$\bar{\beta}_j w_j = (A^H - \bar{\alpha}_{j-1}) w_{j-1} - \bar{\beta}_{j-1} w_{j-2}, \tag{2.9}$$

are obtained by the FOPs recurrences (2.4); see, e.g., [31, 60, 65, 66, 89, 107]. Opposite to Arnoldi's method, which is based on the *long recurrences* (1.2)–(1.3), the non-Hermitian Lanczos algorithm has *short recurrences*, meaning that they have a small fixed number of terms per iteration. The reduced-order (or projected) matrix $J_n = W_n^H A V_n$ is the complex Jacobi matrix (2.5). Note that it is possible to build it by simply storing the coefficients of the recurrences (2.8)–(2.9) while running the non-Hermitian Lanczos algorithm. Consequently, the non-Hermitian Lanczos algorithm corresponds to computing the complex Gauss quadrature (2.7). The survey [C5] presents the connections between the non-Hermitian Lanczos algorithm and the complex Gauss quadrature. To our knowledge, it provides the first comprehensive overview on the matter by putting together the many results scattered in the literature and thus presenting the extension of the well-known results for Hermitian matrices in Section 2.1 to the non-Hermitian ones.

2.3.1 General linear functionals and Lanczos breakdowns

In the survey [C6], with Miroslav Pranić, we completed our study on the Gauss quadrature extension addressing the case of linear functionals that are not quasi-definite. A

regular formal orthogonal polynomial p_n is defined as the polynomial of degree n satisfying

$$\mathcal{L}(p_n \lambda^j) = 0, \quad j = 0, \ldots, n-1.$$

Note that p_n is unique up to nonzero rescaling and that it is not generally orthonormal since $\mathcal{L}((p_n)^2)$ can be equal to zero. If $\mathcal{L}$ is not quasi-definite, then some FOPs do not exist. In this case, we denote with $\nu(k)$, $k = 0, 1, \ldots$, the degrees for which a regular formal orthogonal polynomial $p_{\nu(k)}$ exists. In order to have a complete basis for $\mathcal{P}$, the gaps in the sequence $p_{\nu(0)}, p_{\nu(1)}, p_{\nu(2)}, \ldots$ are filled with the so-called *quasi-orthogonal* polynomials. Given $p_{\nu(k)}$, a quasi-orthogonal polynomial p_n of degree $\nu(k) < n < \nu(k+1)$ is defined as $p_n := q_{n-\nu(k)} p_{\nu(k)}$, where $q_{n-\nu(k)}$ is a polynomials of degree $n - \nu(k)$.

The sequence of regular formal orthogonal polynomials and quasi-orthogonal polynomials $p_0, \ldots, p_n$ satisfy the recurrence relation [44, 62]

$$\lambda p(\lambda) = T_n p(\lambda) + e_n p_n(\lambda)$$

where T_n is the block tridiagonal matrix

$$T_n = \begin{bmatrix} A_0 & \beta_{\nu(1)} & & & \\ \gamma_{\nu(2)} & A_1 & \ddots & & \\ & \ddots & \ddots & & \\ & & & \beta_{\nu(\ell)} & \\ & & \gamma_n & A_\ell^n \end{bmatrix}, \quad p(\lambda) = \begin{bmatrix} p_0(\lambda) \\ \vdots \\ \\ \\ \\ \vdots \\ p_{n-2}(\lambda) \\ p_{n-1}(\lambda) \end{bmatrix}$$

When the linear functional is quasi definite, the blocks $A_0, \ldots, A_\ell$ have size 1; that is, T_n is tridiagonal. When the polynomials $p_0, \ldots, p_n$ are orthonormal, T_n is also symmetric; hence, it is the complex Jacobi matrix (2.5).

The roots of $p_{\nu(k)}$ determine the quadrature $\mathcal{G}_{\nu(k)}$ as in (2.6); see [44]. In [C6], we identified the following fundamental properties of $\mathcal{G}_{\nu(k)}$ (weaker forms of Properties G1–G3).

1. The complex Gauss quadrature $\mathcal{G}_{\nu(k)}$ has the maximal degree of exactness $\nu(k) + \nu(k+1) - 2 \geq 2\nu(k) - 1$.

2. The quadrature $\mathcal{G}_{\nu(k)}$ is well-defined and unique.

3. The quadrature can be written in the form $\mathcal{G}_{\nu(k)}(f) = \mu m_{\nu(1)-1}\, e_1^T f(T_{\nu(k)})\, e_{\nu(1)}$, with $\mu^{-1} = \beta_1 \cdots \beta_{\nu(n-1)-1}$ for $\nu(1) > 1$ or $\mu = 1$ for $\nu(1) = 1$.

In virtue of such properties, we refer to $\mathcal{G}_{\nu(k)}$ as an extension of the *Gauss quadrature*.

The vectors $v_{j+1} := p_j(A)\, v$ and $w_{j+1}^H := w^H p_j(A)$, $j = 0, \ldots, n-1$, are respectively bases of the Krylov subspaces $\mathcal{K}_n(A, v)$ and $\mathcal{K}_n(A^H, w)$. When the linear functional $\mathcal{L}(f) = w^H f(A) v$ is quasi-definite, we saw in the previous section that these bases are mathematically equivalent to the ones computed by the non-Hermitian Lanczos algorithm (up to rescaling). However, for a general linear functional $\mathcal{L}(f) = w^H f(A) v$,

if $\mathcal{L}(p_j^2) = 0$, then the non-Hermitian Lanczos algorithm has a *breakdown* at the jth iteration. In this case, if $p_0, \ldots, p_{n-1}$ is the sequence of regular formal orthogonal polynomials and quasi-orthogonal polynomials described above, the bases $V_n = [v_1, \ldots, v_n]$ and $W_n = [w_1, \ldots, w_n]$ are equivalent to the ones obtained by employing the *look-ahead strategy*; refer to [32, 50, 51, 99]. The non-Hermitian Lanczos algorithm with a look-ahead strategy then produces the approximation:

$$w^H f(A)v \approx \mu m_{\nu(1)-1}\, e_1^T f(T_n)\, e_{\nu(1)} = \mathcal{G}_{\nu(k)}(f), \quad n = \nu(k), \ldots, \nu(k+1) - 1. \tag{2.10}$$

The look-ahead Lanczos algorithm corresponds to computing the extended Gauss quadrature.

A so-called *incurable breakdown* [99, 116] appears at the nth iteration when p_n is the last regular formal orthogonal polynomial. In this case, the approximation (2.10) is exact for every $f \in \mathcal{P}$. In [C6], we used this fact to give a new proof of the *Mismatch Theorem*, i.e., if an incurable breakdown occurs, then each of T_n eigenvalues are A eigenvalues [98, 99, 116].

The survey [C6] provides a unified description of the connections between the Gauss quadrature, orthogonal polynomials' extensions, and the look-ahead non-Hermitian Lanczos algorithm. To our knowledge, the rest of the literature on this topic is fragmented, and the results are explained from many diverse perspectives (linear algebra, control theory, matrix computations, approximation theory), often without showing the connections. The survey also clarifies the connection with the minimal realization problem [62].

2.4 Rational Lanczos algorithm

To be consistent with the notation of paper [C7], in this section, we consider the following definition of the rational Krylov subspace

$$\mathcal{R}_m(A, v, \boldsymbol{\xi}) := \text{span}\left\{ v, \left(I - \xi_1^{-1}A\right)^{-1} v, \ldots, \prod_{j=1}^{m-1} \left(I - \xi_j^{-1}A\right)^{-1} v \right\},$$

which is mathematically equivalent to the one in Section 1.3.1 assuming $\xi_j = \sigma_j \neq 0$. The Rational Krylov subspace method is characterized by the long recurrences (1.6)–(1.7). When A is Hermitian, a reformulation of RKSM was devised so that the matrices H_m, K_m in (1.8) are tridiagonal [36, 41, 67], leading to the so-called *rational Lanczos algorithm* characterized by the *three-term recurrences*

$$\beta_j \left(I - \xi_j^{-1}A\right) q_{j+1} = Aq_j - \alpha_j \left(I - \xi_{j-1}^{-1}A\right) q_j - \beta_{j-1} \left(I - \xi_{j-2}^{-1}A\right) q_{j-1}, \quad j = 1, \ldots n;$$

see [67]. Setting

$$\begin{aligned} r &:= (I - \xi_j^{-1}A)^{-1}(Aq_j + \beta_{j-1}\xi_{j-2}^{-1}Aq_{j-1}) - \beta_{j-1}q_{j-1}, \\ s &:= (I - \xi_j^{-1}A)^{-1}(I - \xi_{j-1}^{-1}A)q_j, \end{aligned}$$

the coefficients α_j and β_j are given by

$$\alpha_j = (r^T q_j)/(s^T q_j), \quad \beta_j = \|r - \alpha_j s\|.$$

Note that, for Hermitian matrices, the basis $q_1, \ldots, q_n$ is mathematically equivalent to the basis $v_1, \ldots, v_n$ produced by the long RKSM recurrences (1.6)–(1.7). About short recurrences in rational Krylov subspace computations, we refer the reader also to [54, 64, 103, 118].

The rational Lanczos algorithm requires solving two linear systems per iteration. This appears to make the short recurrences not competitive compared to the RKSM algorithm, which only requires the solution of one linear system per iteration. In [C7], with Davide Palitta and Valeria Simoncini, we proposed a new implementation of the rational Lanczos algorithm that i) reduces the computational costs of the two systems combining them into a single one with two right-hand sides ii) computes the entries of the reduced order matrix

$$R_n := Q_n^T A Q_n$$

as the iterations proceed, without storing the whole basis $Q_m = [q_1, \ldots, q_n]$. Moreover, [C7] illustrates the advantages of the new implementation with several applications and discusses the loss of orthogonality of the basis Q_m in finite precision arithmetic. We obtained preliminary considerations on finite precision arithmetic computations indicating that the rational Lanczos algorithm behaves similarly to its polynomial counterpart thanks also to the results on the decay phenomenon of functions of the matrix R_n (Section 1.3.1 and [C2]).

Among other applications, [C7] deals with the approximation of the bilinear form $v^T f(A)\, v$, with $\|v\| = 1$, by the formula

$$v^T f(A)\, v \approx e_1^T f(R_n) e_1.$$

As we have seen in Section 2.1, the classical (polynomial) Lanczos algorithm produces the Jacobi matrix J_m for which the approximant $e_1^T f(J_n) e_1$ is exact for every polynomial f of degree at most $2n-1$. Correspondingly, in the rational case, the following property holds for every polynomial p of degree at most $2n-1$.

$$v^T p(A) q(A)^{-2}\, v \approx e_1^T p(R_n) q(R_n)^{-2} e_1, \quad q(x) = \prod_{j=1}^{n-1} \left(1 - \frac{x}{\xi_j}\right). \tag{2.11}$$

To our knowledge, this result was first stated in [68, Remark 3.2]. However, it is also a consequence of Theorem 2 in [54]. In [C7], we gave a new proof based on ideas borrowed from Vorobjev's problem of moments [89]; see also [104]. Since $v^T p(A) q(A)^{-2}\, v$ can be expressed as a Riemann-Stieltjes integral $I_\mu(pq^{-2})$ as in (2.2), the approximation (2.11) can be seen as a rational Gauss quadrature; see [40, 42, 68, 81, 90, 103].

Chapter 3

Computing the time-ordered exponential

3.1 Introduction

Consider the matrix-valued function $\tilde{A}(t) \in \mathbb{C}^{N\times N}$ analytic on the bounded interval $\mathcal{I} = [a, b]$ and let $\tilde{U}(t) \in \mathbb{C}^{N\times N}$ be the solution of the non-autonomous ordinary differential equation

$$\frac{\partial}{\partial t}\tilde{U}(t) = \tilde{A}(t)\tilde{U}(t), \quad \tilde{U}(a) = I, \quad t \in \mathcal{I} = [a, b]. \tag{3.1}$$

When $\tilde{A}(t)$ commutes with itself at different times, i.e., $\tilde{A}(t_1)\tilde{A}(t_2) = \tilde{A}(t_2)\tilde{A}(t_1)$, the solution is given through the matrix exponential as

$$\tilde{U}(t) = \exp\left(\int_a^t \tilde{A}(\tau)\mathrm{d}\tau\right), \quad t \in [a, b].$$

However, in the general case, there is no explicit formula for $\tilde{U}(t)$ in terms of usual matrix functions. The solution $\tilde{U}(t)$ instead is known as *time-ordered exponential* (TOE), denoted $\mathcal{T}\exp(\tilde{A})$, and can be seen as a generalization of the matrix function concept to the time-dependent case. Among the possible applications, we focus on quantum chemistry where the quantum spin dynamics are often modeled by non-autonomous ODEs (Schrödinger equation); see, e.g., [84]. As no general explicit expression of $\tilde{U}(t)$ is accessible, analytic approaches are typically based on Floquet formalism [79, 109], Magnus series [26, 91], or hybrids of these with ad-hoc approximate/numerical methods [33, 92–94, 119]. These analytic approaches rarely provide exact solutions in a finite number of steps, might suffer from convergence issues [26], and be intractable [39]. There is a perception in the physics community that no exact solutions are achievable [61]. This also influences the development of numerical solvers since the most advanced numerical methods are typically built on analytical approaches [26, 73, 79]. As noted by M. Grifoni and P. Hänggi in [63]: "Solving the time-dependent Schrödinger equation necessitates the development of novel analytic and computational schemes [...] in a nonperturbative manner" – a remark still relevant today.

In the following sections, we first present the results obtained in collaboration with Pierre-Louis Giscard. Together, we derived an entirely original symbolic method that

allows expressing the time-ordered exponential in a finite number of steps. The method is based on a combination of two mathematical novelties. The first is the $\star$-algebra (Section 3.2), a Frechet-Lie algebra on bivariate distributions equipped with the so-called $\star$-product. The second one is the $\star$-Lanczos algorithm, a symbolic method able to transform the matrix $\tilde{A}(t)$ into a tridiagonal matrix in the $\star$-algebra (Section 3.3). Then, with Niel Van Buggenhout, we used these results to devise a new class of numerical ODE solvers (Section 3.4). In Section 3.4.1, we conclude with some remarks on the decay phenomenon of time-ordered exponentials.

3.2 The $\star$-product

For the scope of this thesis, we restrict the definition of the $\star$-product to a product defined on the set $\mathcal{A}(\mathcal{I})$ of the bivariate distributions for which there exists a finite k so that

$$f(t,s) = \tilde{f}_{-1}(t,s)\Theta(t-s) + \tilde{f}_0(t,s)\delta(t-s) + \cdots + \tilde{f}_k(t,s)\delta^{(k)}(t-s),$$

where $\tilde{f}_{-1}(t,s), \ldots, \tilde{f}_k(t,s)$ are functions analytic both in t and s over the interval $\mathcal{I} = [a,b]$, $\Theta(t-s)$ is the Heaviside function ($\Theta(t-s) = 1$ for $t \geq s$, and 0 otherwise), and $\delta(t-s), \delta'(t-s), \delta^{(2)}(t-s), \ldots$ are the Dirac delta and its derivatives. Note that in [106], the $\star$-product is defined over a larger group of distributions. We also define the subset $\mathcal{A}_\Theta(\mathcal{I}) \subset \mathcal{A}(\mathcal{I})$ composed of the distributions of the kind $f(t,s) = \tilde{f}_{-1}(t,s)\Theta(t-s)$.

The $\star$-product of $f_1, f_2 \in \mathcal{A}(\mathcal{I})$ is the non-commutative product defined as

$$(f_1 \star f_2)(t,s) := \int_{\mathcal{I}} f_1(t,\tau) f_2(\tau,s)\,\mathrm{d}\tau \in \mathcal{A}(\mathcal{I}).$$

The product is an extension of Volterra compositions [120–122]. A similar extension to distributions of the Volterra compositions appeared in [58], and the first results on the $\star$-product and its inverse appeared in [59] in connection with the works [C8, C9]. These works led to the formalization of the $\star$-product by Manon Ryckebusch (University of the Littoral Opal Coast) in [106]. The *$\star$-resolvent* of $x \in \mathcal{A}_\Theta(\mathcal{I})$ is defined as

$$R^\star(x) := \sum_{j=0}^{\infty} x^{\star j}.$$

Note that $R^\star(x)$ is well-defined (i.e., convergent) for every $x \in \mathcal{A}_\Theta(\mathcal{I})$ [58]. In Table 3.1, we list essential properties of the $\star$-product and the definition of related objects.

The $\star$-product straightforwardly extends to a matrix-matrix (matrix-vector) $\star$-product for given matrices with compatible sizes, composed of elements from $\mathcal{A}(\mathcal{I})$. We denote with $\mathcal{A}^{N\times M}(\mathcal{I})$ the space of the $N \times M$ matrices with elements from $\mathcal{A}(\mathcal{I})$ (we define analogously the subset $\mathcal{A}_\Theta^{N\times M}(\mathcal{I})$). Note that $I_\star = I\delta(t-s)$ is the identity matrix in $\mathcal{A}^{N\times N}(\mathcal{I})$. As shown in [58], the solution $\tilde{U}(t)$ of the ODE (3.1) can then be expressed as

$$\tilde{U}(t) = U(t,a), \quad U(t,s) = \Theta(t-s) \star R^\star\left(\tilde{A}(t)\Theta(t-s)\right), \quad t \in \mathcal{I} = [a,b]. \tag{3.2}$$

see also [C8, C9]. In particular, for a given vector $v \in \mathbb{C}^N$, consider the ODE

$$\frac{\partial}{\partial t}\tilde{u}(t) = \tilde{A}(t)\tilde{u}(t), \quad \tilde{u}(a) = v, \quad t \in \mathcal{I} = [a,b]. \tag{3.3}$$

Table 3.1: Main properties of the $\star$-product and related definitions ($f, g, x \in \mathcal{A}(\mathcal{I})$).

Name	Symbol	Definition	Comments / Properties
$\star$-identity	δ	$f \star \delta = \delta \star f = f$	
$\star$-inverse	$f^{-\star}$	$f \star f^{-\star} = f^{-\star} \star f = \delta$	Existence [59, 106]
Dirac 1st derivative	δ'	$\delta'(t-s)$	$\delta' \star \Theta = \Theta \star \delta' = \delta$
Dirac derivatives	$\delta^{(j)}$	$\delta^{(j)}(t-s)$	$\delta^{(j)} \star \delta^{(i)} = \delta^{(i+j)}$
$\star$-powers	$f^{\star j}$	$f \star f \star \cdots \star f$, j times	$f^{\star 0} := \delta$, by convention
$\star$-resolvent	$R^\star(x)$	$\sum_{j=0}^{\infty} x^{\star j}$, $x \in \mathcal{A}_\Theta(\mathcal{I})$	$R^\star(x) = (\delta - x)^{-\star}$

Since

$$R^\star\left(\tilde{A}(t)\Theta(t-s)\right) = \left(I_\star(t-s) - \tilde{A}(t)\Theta(t-s)\right)^{-\star},$$

the solution $\tilde{u}(t) = \tilde{U}(t)v$ can be given through the solution of the $\star$-linear equation

$$\tilde{u}(t) = u(t,a), \quad u = \Theta \star x, \quad \left(I_\star - \tilde{A}\Theta\right) \star x = \tilde{v}\delta, \quad t \in [a,b]; \tag{3.4}$$

here and in the following, when needed, we do not explicitly write the arguments t, s when they are clear from the context.

3.3 The $\star$-Lanczos algorithm

In [C9], we introduced the *$\star$-Lanczos algorithm*, a (symbolic) algorithm working in the $\star$-algebra of matrices from $\mathcal{A}^{N\times N}(\mathcal{I})$. The algorithm is a generalization of the non-Hermitian Lanczos algorithm (see Section 2.3). Given $\tilde{A}(t)$ an $N \times N$ time-dependent matrix analytic over $\mathcal{I}$, v and w time-independent vectors ($w^H v \neq 0$), and assuming no breakdown, the nth step of the $\star$-Lanczos algorithm gives the matrices $V_n(t,s), W_n(t,s) \in \mathcal{A}^{N\times n}(\mathcal{I})$ satisfying the $\star$-biorthogonality condition

$$W_n(t,s)^H \star V_n(t,s) = I_\star.$$

Moreover, it produces the tridiagonal matrix

$$T_n(t,s) := \begin{bmatrix} \alpha_0(t,s) & \delta(t-s) & & \\ \beta_1(t,s) & \alpha_1(t,s) & \ddots & \\ & \ddots & \ddots & \delta(t-s) \\ & & \beta_{n-1}(t,s) & \alpha_{n-1}(t-s) \end{bmatrix} \in \mathcal{A}^{n\times n}(\mathcal{I}),$$

such that, denoting $A(t,s) := \tilde{A}(t)\Theta(t-s)$, it holds

$$T_n(t,s) = W_n^H(t,s) \star A(t,s) \star V_n(t,s).$$

The "coefficients" $\alpha_0, \ldots, \alpha_{n-1}$ and $\beta_1, \ldots, \beta_{n-1}$ are (scalar) distributions appearing in the $\star$-Lanczos algorithm short recurrences:

$$\widehat{v}_n = A \star v_{n-1} - v_{n-1} \star \alpha_{n-1} - v_{n-2}, \tag{3.5}$$

$$w_n^H = w_{n-1}^H \star A - \alpha_{n-1} \star w_{n-1}^H - \beta_{n-1} \star w_{n-2}^H, \tag{3.6}$$

which are a generalization of the non-Hermitian Lanczos recurrences (2.8)-(2.9). Most importantly, T_n satisfies the $\star$-*matching moment* property

$$w^H \left(A(t,s)\right)^{\star k} \, v = e_1^H \left(T_n(t,s)\right)^{\star k} \, e_1, \quad k = 0, \ldots, 2n-1;$$

proved in [C9, Theorem 2.1].

Remark 3.1 *In [C9], we defined a $\star$-polynomial $p(\lambda)(t,s)$ as*

$$p(\lambda)(t,s) := \gamma_0(t,s) + \lambda \star \gamma_1(t,s) + \lambda^{\star 2} \star \gamma_2(t,s) + \cdots + \lambda^{\star k} \star \gamma_k(t,s),$$

with λ the variable and $\gamma_0, \ldots, \gamma_k \in \mathcal{A}(\mathcal{I})$ the coefficients of the polynomial. Then, the $\star$-matching moment property was proved by extending the concept of formal orthogonal polynomials *(Section 2.2) to $\star$-polynomials. The short recurrences* (3.5)–(3.6) *can also be derived by applying the strategy presented in Section 2.3 to the $\star$-polynomials case. Furthermore, note that the $\star$-matching moment property has the same degree of exactness as a Gauss quadrature (in terms of the degree of a $\star$-polynomial). Hence, it might be possible to extend the Gauss quadrature to the $\star$-framework. This requires proving the existence of the roots of $\star$-polynomials, a non-trivial problem.*

As a consequence of the $\star$-matching moment property, in [C9] we proposed the following approximation

$$w^H R^\star(A) v = \sum_{k=0}^{\infty} w^H A^{\star k} v \approx \sum_{k=0}^{\infty} e_1^H (T_n)^{\star k} e_1 = e_1^H R^\star(T_n) e_1.$$

Therefore, the $\star$-Lanczos algorithm can be used to approximate the bilinear form

$$\tilde{y}(t) := w^H \tilde{u}(t),$$

with $\tilde{u}(t)$ the solution of Equation (3.3), by

$$\tilde{y}(t) \approx y_n(t,a), \quad y_n = \Theta \star e_1^H x_n, \quad (I_\star - T_n) \star x_n = e_1 \delta, \quad t \in [a,b], \tag{3.7}$$

that is, by solving a reduced order $\star$-linear system. Moreover, in [C8] we combined the tridiagonal structure of T_n with the *Path-sum method* [58] obtaining the expression:

$$e_1^H x_n = e_1^H R^\star(T_n) e_1 = R^\star(\alpha_0 + R^\star(\alpha_1 + R^\star(\cdots + R^\star(\alpha_{n-1}) \star \beta_{n-1}) \star \cdots) \star \beta_1),$$

which is a continued fraction in the $\star$-algebra.

Assuming no breakdowns, in [C8] we proved that for $n = N$ the $\star$-Lanczos algorithm tridiagonalizes $A(t,s)$ producing the $\star$-factorization

$$A(t,s) = V_N(t,s) \star T_N(t,s) \star W_N^H(t,s),$$

implying

$$R^\star(A) = V_N \star R^\star(T_N) \star W_N^H,$$

thus showing that $\tilde{y}(t) = y_N(t, a)$, $t \in [a, b]$, with

$$y_N = \Theta \star R^\star(\alpha_0 + R^\star(\alpha_1 + R^\star(\cdots + R^\star(\alpha_{N-1}) \star \beta_{n-1}) \star \cdots) \star \beta_1). \tag{3.8}$$

This proves that it is possible to express the solution $\tilde{y}(t) = y_N(t, a)$ in a finite number of $\star$-operations. In particular, the expression in (3.8) requires solving an order N number of integro-differential equations. The first expression for the solution of Equation (3.3) given in finitely many integro-differential equations was derived in [58]. Unfortunately, it requires solving a #P-complete problem. Assuming no breakdowns, the $\star$-Lanczos algorithm solves this issue as the algorithm's complexity is of order nN^2 (it is smaller when $\tilde{A}(t)$ is sparse) in terms of the number of usual algebraic sums of functions, $\star$-products, and $\star$-inverses, i.e., the $\star$-operations. Therefore, for $n = N$, the solution $\tilde{y}(t) = y_N(t, a)$ is obtained in an order N^3 of $\star$-operations. Furthermore, [C8] also discusses the breakdown issue, relating it with the breakdown of the usual non-Hermitian Lanczos algorithm with input $\tilde{A}(t), v, w$, where $t \in \mathcal{I}$ is fixed.

We remark that rounding errors deeply affect the (usual) Lanczos algorithm by loss of orthogonality. We expect an analogous behavior in any numerical implementation of the $\star$-Lanczos algorithm. Therefore, this aspect must be investigated before relying on any implementation in finite precision arithmetic.

3.4 Numerical methods for the time-ordered exponential

The $\star$-Lanczos algorithm and the expression in (3.7) require computing $\star$-operations corresponding to computing integrals and solving integro-differential equations; see [59]. Explicit analytic solutions are, hence, rare. Moreover, in many applications the size N of the matrix $\tilde{A}(t)$ in Equation (3.3) can be huge, making symbolic computations unachievable. For these reasons, in [C11], we introduced a numerical approach to compute $\star$-operations.

Numerical solvers for non-autonomous ODEs are commonly used in quantum chemistry, e.g., in spin simulations [84]. Among them, we find: Runge-Kutta and Runge-Kutta-Nyström approaches [22, 29, 117]; splitting and composition approaches (symplectic [24], Magnus [5, 26, 75–77], commutator-free [1, 2, 4, 23, 27, 28], and symmetric [18, 20, 21]); polynomial-based methods [74, 97]. Many of these and other methods are classified as *geometric integrators*, i.e., able to preserve geometrical properties (symmetry, unitarity, symplecticity) [19, 71, 78, 83]. Other classes are the global methods (without interval discretization) [100], and methods characterized by large discretization steps [3, 34, 35, 108]. Nevertheless, the computational cost needed to solve ODE systems is still a bottleneck in many quantum chemistry problems. As stated in [84], "*Time propagation commonly dominates the wall clock time of spin dynamics simulations*".

The numerical approach described below has proved highly competitive in the solution of ODEs related to a specific model, the *generalized Rosen-Zener model*. In [30], with Christian Bonhomme (Sorbonne University) and Niel Van Buggenhout, we devised a method whose computational cost scales linearly with the model size (Fig 3.1,

left). Its cost is also competitive for increasing interval sizes (Fig 3.1, right). These first results might open the way to more general efficient methods for spin simulations.

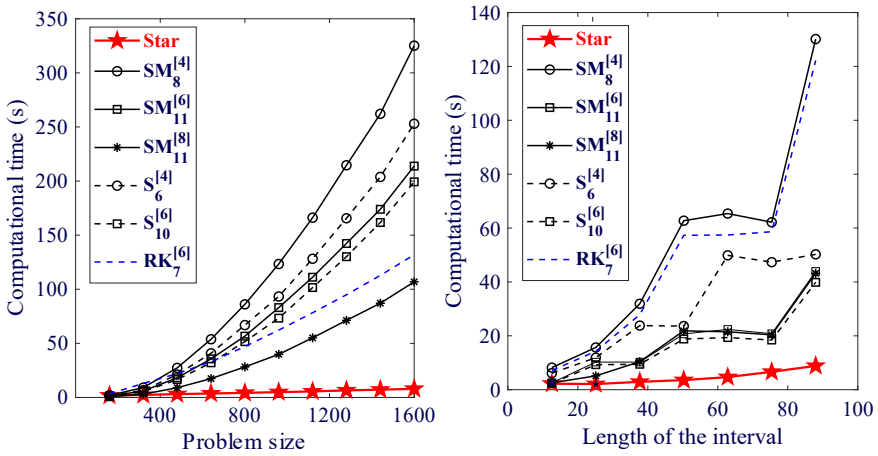


Figure 3.1: Computational time comparison on the generalized Rosen-Zener model of increasing size between the $\star$-method [30] and the methods: $\mathrm{SM}_8^{[4]}$, $\mathrm{SM}_{11}^{[6]}$, $\mathrm{SM}_{11}^{[8]}$ [24], $\mathrm{S}_6^{[4]}$, $\mathrm{S}_{10}^{[6]}$ [29], and Runge-Kutta $\mathrm{RK}_7^{[6]}$ [24]. Left: Increasing size of the problem. Right: Increasing interval length.

In [C10, C11], we used Legendre polynomials to transform $\star$-algebra operations into operations in the usual matrix algebra. First, in [C10], we showed that it is possible to map a subalgebra of the $\star$-algebra into a subalgebra of infinite matrices. For simplicity, let us consider the interval $\mathcal{I} = [-1, 1]$, and let $p_0(\tau), p_1(\tau), p_2(\tau), \ldots$ be the orthonormal Legendre polynomials, i.e., p_k, p_ℓ are polynomials of dregree respectively k, ℓ satisfying the orthogonality condition:

$$\int_{-1}^{1} p_k(\tau) p_\ell(\tau) d\tau = \delta_{k\ell}.$$

Then, a bivariate distribution $f(t, s) \in \mathcal{A}_\Theta(\mathcal{I})$ can be expanded into the series

$$f(t, s) = \tilde{f}(t)\Theta(t - s) = \sum_{k=0}^{\infty} \sum_{\ell=0}^{\infty} \alpha_{k,\ell} p_k(t) p_\ell(s), \quad t, s \in [-1.1], \quad t \neq s,$$

with coefficients

$$\alpha_{k,l} = \int_{-1}^{1} p_k(\rho) \left(\int_{-1}^{1} f(\tau, \rho) p_\ell(\tau) d\tau \right) d\rho.$$

By defining the following infinite-size matrix and vector

$$F := \begin{bmatrix} \alpha_{0,0} & \alpha_{0,1} & \cdots \\ \alpha_{1,0} & \alpha_{1,1} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \qquad \boldsymbol{\varphi}(\tau) := \begin{bmatrix} p_0(\tau) \\ p_1(\tau) \\ \vdots \end{bmatrix}, \tag{3.9}$$

we get the matrix representation of the series:

$$f(t, s) = \boldsymbol{\varphi}(t)^T F \, \boldsymbol{\varphi}(s), \quad t, s \in [-1.1], \quad t \neq s.$$

The matrix F is known as *coefficient matrix*. We denote with $\mathcal{D}_t(\mathcal{I}) \subset \mathcal{A}(\mathcal{I})$ the subring generated by the identity δ and all the distributions $f \in \mathcal{A}_\Theta(\mathcal{I})$ such that $f(t,s) = \tilde{f}(t)\Theta(t-s)$. Given $f, g, p \in \mathcal{D}_t(\mathcal{I})$ so that $p = f \star g$, and let F, G, P be the respective coefficient matrices, in [C10], we showed that

$$P = FG, \tag{3.10}$$

i.e., the infinite matrix-matrix product FG is well-defined and represents the $\star$-product.

Remark 3.2 *The well-definiteness of the product* (3.10) *was proved by exploiting the fact that the element's magnitude of F and G decay exponentially moving away from the diagonal, that is, the matrices display an off-diagonal decay phenomenon (Chapter 1).*

Consequently, other matrix algebra operations are well-defined and correspond to related $\star$-operations. We summarize them in Table 3.2. In particular, the $\star$-resolvent $R^\star(f)$ corresponds to the matrix resolvent $(I-F)^{-1}$ since the infinite matrix $I-F$ is invertible [C10].

$\star$-operations/objects	matrix operations/objects
$p = f \star g$	$P = FG$
$p = f + g$	$P = F + G$
$1_\star = \delta(t-s)$	I, identity matrix
$f^{-\star}$	F^{-1}, inverse
$R^\star(f) = (1_\star - f)^{-\star}$	$(I-F)^{-1}$, resolvent

Table 3.2: Left: $\star$-algebra operations and related objects. Right: The corresponding infinite matrix algebra operations and objects. F, G, P are the coefficient matrices (3.9) of the distributions $f, g, p \in \mathcal{D}_t(\mathcal{I})$ respectively.

The relations in Table 3.2 can be extended to $\mathcal{D}_t^{N\times N}(\mathcal{I})$. Following [C10], let $A(t,s) = [f_{ij}(t,s)]_{i,j=1}^N$ be an $N \times N$ matrix with elements $f_{ij}(t) \in \mathcal{D}_t^{N\times N}(\mathcal{I})$. Each element f_{ij} can be mapped into the related coefficient matrices $F^{(i,j)}$ obtaining the following $N \times N$ matrix composed of infinite-size blocks

$$\mathbf{A} := \begin{bmatrix} \boxed{F^{(1,1)}} & \dots & \boxed{F^{(1,N)}} \\ \vdots & \ddots & \vdots \\ \boxed{F^{(N,1)}} & \dots & \boxed{F^{(N,N)}} \end{bmatrix}. \tag{3.11}$$

Then, given $A(t,s), B(t,s), C(t,s) \in \mathcal{D}_t^{N\times N}(\mathcal{I})$ so that $C(t,s) = A(t,s) \star B(t,s)$, the respective *coefficient matrices* (3.11), denoted $\mathbf{A}, \mathbf{B}, \mathbf{C}$, satisfy $\mathbf{C} = \mathbf{AB}$. Therefore, the relations in Table 3.2 are generalized as in Table 3.3.

Denoting with H the coefficient matrix of $\Theta(t-s)$, we can finally express the solution of Equation (3.3) on the interval $\mathcal{I} = [-1, 1]$ by transforming the expression in (3.4) into the infinite matrix formula:

$$\tilde{u}(t) = (I_N \otimes \boldsymbol{\varphi}(t)^T H)(I - \mathbf{A})^{-1}(v \otimes \boldsymbol{\varphi}(-1)), \tag{3.12}$$

$\star$-operations/objects	matrix operations/objects
$C = A \star B$	$\mathbf{C} = \mathbf{AB}$
$C = A + B$	$\mathbf{C} = \mathbf{A} + \mathbf{B}$
$I_\star = \delta(t-s)I_N$	I, identity matrix
$A^{-\star}$	$\mathbf{A}^{-1}$, inverse
$R^\star(f) = (I_\star - A)^{-\star}$	$(I - \mathbf{A})^{-1}$, resolvent

Table 3.3: Left: $\star$-algebra operations and related objects. Right: The corresponding infinite matrix algebra operations and objects. $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are the coefficient matrices (3.11) of the distribution matrices $A, B, C \in \mathcal{D}_t^{N\times N}(\mathcal{I})$ respectively.

with $\mathbf{A}$ the coefficient matrix of $A(t,s) = \tilde{A}(t)\Theta(t-s)$ and $\otimes$ the Kronecker product. More in general, the time-ordered exponential $\tilde{U}(t)$ (solution of the Equation (3.1)) can be expressed by

$$\tilde{U}(t) = (I_N \otimes \boldsymbol{\varphi}(t)^T)\,\mathbf{U}\,(I_N \otimes \boldsymbol{\varphi}(-1)^T), \quad \mathbf{U} = (I_N \otimes H)(I - \mathbf{A})^{-1}, \tag{3.13}$$

for $t \in \mathcal{I} = [-1, 1]$.

In [C11], with Niel Van Buggenhout, we obtained a new computational approach for ODEs by truncating the coefficient matrices involved in the expression in (3.12). For simplicity, the paper restricts the analysis to the scalar case, i.e., the case in which the matrix $\tilde{A}(t)$ has size 1. Given $f \in \mathcal{D}_t(\mathcal{I})$, it introduces an efficient method to compute the $M \times M$ leading submatrix F_M of the coefficient matrix (3.9). Moreover, it shows that the truncation error can be kept under control by defining the matrix $\underline{F}_M$ by setting to zero the last L rows of F_M, with L the numerical bandwidth of F_M. Then, the solution of the (scalar) ODE

$$\frac{\partial}{\partial t}\tilde{u}(t) = \tilde{f}(t)\tilde{u}(t), \quad \tilde{u}(-1) = 1, \quad t \in [-1, 1] \tag{3.14}$$

is approximated by the formula

$$\tilde{u}(t) \approx \tilde{u}_M(t) := \boldsymbol{\varphi}_M(t)^T \underline{H}_M (I_M - \underline{F}_M)^{-1} \boldsymbol{\varphi}_M(-1),$$

where $\boldsymbol{\varphi}_M(-1)$ is the first M elements of $\boldsymbol{\varphi}(-1)$ and $\underline{H}_M$ is the $M \times M$ leading submatrix of H with the last rows set to zero. Then, the solution is computed by solving the linear system

$$(I_M - \underline{F}_M)x = \boldsymbol{\varphi}_M(-1), \quad \tilde{u}_M(t) = \boldsymbol{\varphi}_M(t)^T \underline{H}_M x.$$

The approximated solution is then $\tilde{u}_M(t) = \sum_{i=0}^{M-1} \hat{c}_i p_i(t)$, where the coefficients $\hat{c}_i$ are the components of the vector $\hat{c} = \underline{H}_M x$. Assuming the linear system can be computed exactly, in [C11], we proved that the truncation error $|\tilde{u}(t) - \tilde{u}_M(t)|$ can be kept at a desired precision as long as M is large enough, for $t \in [-1, 1]$. We remark that $\hat{c}_0, \ldots, \hat{c}_{M-1}$ are the approximations of the first M Legendre coefficients of the solution $\tilde{u}(t) = \sum_{i=0}^{\infty} c_i p_i(t)$. Therefore, the described approach is a spectral method we named *$\star$-method*. It is summarized as follows.

Algorithm 3.3 ($\star$-method – scalar version) *Given an analytic function $\tilde{f}(t)$, the method computes the approximated Legendre coefficients $\hat{c}_0, \ldots, \hat{c}_{M-1}$ of $\tilde{u}(t)$, the solution of Equation* (3.14)*.*
Procedure:

1. *Determine an appropriate value for the truncation parameter M.*
2. *Construct the truncated coefficient matrix $\underline{F}_M$;*
3. *Solve the linear system of equations $(I_M - \underline{F}_M)x = \phi_M(-1)$.*
4. *Compute $\hat{c} = \underline{H}_M x$, obtaining the coefficients $\hat{c} = [\hat{c}_0, \ldots, \hat{c}_{M-1}]^T$.*

As numerically shown in [30, 102], the $\star$-method can be extended to solve the system of ODEs (3.3) as follows.

Algorithm 3.4 ($\star$-method – matrix version) *Given an analytic $N \times N$ matrix-valued function $\tilde{A}(t)$, a vector v, and setting the interval $\mathcal{I} = [-1, 1]$, the method computes the approximated Legendre coefficients $\hat{c}_{(i-1)M}, \ldots, \hat{c}_{iM-1}$ of the ith component of Equation (3.3) solution for $i = 1, \ldots, N$, i.e.,*

$$\tilde{u}_i(t) \approx \sum_{j=0}^{M-1} \hat{c}_{(i-1)M+j} p_j(t), \quad t \in [-1, 1], \quad i = 1, \ldots, N.$$

Procedure:

1. *Determine an appropriate value for the truncation parameter M.*
2. *Construct the truncated coefficient matrix $\mathbf{A}_M = \left[\underline{F}_M^{(i,j)}\right]_{i,j=1}^{N}$.*
3. *Solve the linear system of equations $(I_{MN} - \mathbf{A}_M)x = v \otimes \phi_M(-1)$.*
4. *Compute $\hat{c} = (I_N \otimes \underline{H}_M)x$, obtaining the coefficients $\hat{c} = [\hat{c}_0, \ldots, \hat{c}_{MN-1}]^T$.*

In [30], using an ad hoc solver for the equation $(I_{MN} - \mathbf{A}_M)x = v \otimes \phi_M(-1)$, we were able to obtain the results in Fig. 3.1.

The numerical results in [30, 102] suggest that the analysis of the truncation error of Algorithm 3.3 done in [C11] extends to Algorithm 3.4. While a thorough analysis is needed, the partial results in [C11] are a fundamental step towards it. The analysis of the computational error of the $\star$-method is an ongoing work that requires i) extending the truncation error analysis done in [C11] ii) analyzing the error of the algorithm in finite precision arithmetic, in particular when solving the linear system.

3.4.1 Time-ordered exponential and decay phenomenon

Many of the results presented in [C10, C11] are based on the fact that the coefficient matrices (3.9) is characterized by an off-diagonal decay; see Remark 3.2. Consider the matrix composed of infinite-size blocks $\mathbf{U} = (I_N \otimes \underline{H})(I - \mathbf{A})^{-1}$ in (3.13). Each block $U^{(i,j)}$ of the matrix $\mathbf{U}$ is the coefficient matrix of a distribution from $\mathcal{D}_t(\mathcal{I})$. Hence, $U^{(i,j)}$ is characterized by an off-diagonal decay.

The decay phenomenon does not appear only in each block of $\mathbf{U}$. When the time-dependent matrix $\tilde{A}(t)$ in (3.1) is sparse, we also expect an overall decay phenomenon since $\mathbf{U}$ is obtained from the inverse $(I - \mathbf{A})^{-1}$. This means that many blocks $U^{(i,j)}$ are expected to be nearly zero blocks. See Fig. 3.2 for a numerical example. This is

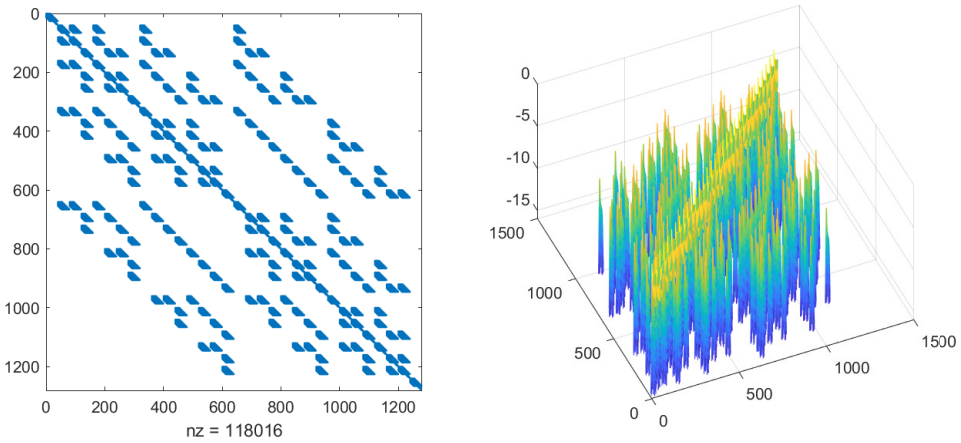


Figure 3.2: Left: The sparsity pattern of the matrix $\mathbf{A}_M$ taken from the numerical experiments in [88, Section 4.2], with $M = 40$ and $N = 32$. Right: Magnitude of the elements of $(I_N \otimes \underline{H}_M)(I - \mathbf{A}_M)^{-1}$ in logarithmic scale.

not surprising, since $\tilde{U}(t)$ and $\mathbf{U}$ are connected by the formula in (3.13) and $\tilde{U}(t)$ is known to be localized, i.e., its elements' magnitudes decay with the geodesic distance in the associated graph [9, 58]; see Section 1.2.

One open issue in the $\star$-method is how to a priori estimate the truncation parameter M, which must be large enough to well approximate both $\mathbf{A}$ and $\mathbf{U}$. The main problem is with the matrix $\mathbf{U}$ since, obviously, it is not accessible before running the method. A possible approach we are investigating is using a priori bounds like the ones described in Chapter 1. The a priori bounds might also be used to devise numerical algorithms for the approximation of $\tilde{U}(t)$, with a numerical cost that is linear in the matrix size N (under appropriate conditions); see [16].

3.5 Conclusions and outlook

As explained in this chapter, a fundamental connection exists between the $\star$-algebra and the algebra of matrices. In [C10], this link has enabled us to leverage results on the decay phenomenon, as introduced in Chapter 1, demonstrating the existence of a map between a subalgebra of the $\star$-algebra and a corresponding decay algebra. Decay algebras are algebras of infinite-size matrices characterized by off-diagonal decay [9, 80]. On the one hand, this map allowed us to approximate the $\star$-product by a matrix-matrix product. On the other hand, we proved that the matrix $I - \mathbf{A}$ in (3.12) is invertible by exploiting the fact that $I_\star(t-s) - \tilde{A}(t)\Theta(t-s)$ is $\star$-invertible [C10].

Moreover, we described how the formal orthogonal polynomials from Chapter 2 can be generalized within the $\star$-product framework. This led to introducing the $\star$-Lanczos algorithm, a generalization of the non-Hermitian Lanczos algorithm. This result paves the way for extending other numerical linear algebra algorithms, such as the BiCG algorithm [107], to the $\star$-algebra setting. Conversely, investigating the $\star$-Lanczos algorithm and other $\star$-based techniques may offer insights into improving their

matrix algebra counterparts. For instance, it may be possible to develop preconditioners within the $\star$-algebra and then map them to the decay algebra of matrices, potentially yielding effective preconditioners for solving (3.12).

The novel numerical approach described in Section 3.4 has shown strong performance in solving ODEs related to a specific model, the generalized Rosen-Zener model. In [30], we introduced a new algorithm for these ODEs, which demonstrates a computational cost scaling linearly with the model size (Fig 3.1, left) and maintains high efficiency for increasing interval sizes (Fig 3.1, right). These results might open the way to more general efficient methods for spin simulations and optimal control. However, the spectral properties and structural complexity of other quantum systems present challenges when solving (3.12). For example, we are currently working on a system that considers a Nuclear Magnetic Resonance (NMR) application with dipolar interactions [88]. In this case, the strategies used in [30] prove insufficient. To address these challenges, we are testing randomized approaches (in collaboration with Lorenzo Lazzarino, University of Oxford) and tensor methods, which show promising results. The development of numerical methods with a computational cost that scales linearly with the problem size could enable the solution of previously intractable NMR problems (e.g., systems with ten thousand spins or diffusion in multiphasic materials). This progress would significantly advance quantum optimal control techniques and the broader field of quantum simulation.

Bibliography

[1] A. Alvermann and H. Fehske, *High-order commutator-free exponential time-propagation of driven quantum systems*, J. Comput. Phys., 230 (2011), pp. 5930–5956.

[2] A. Alvermann, H. Fehske, and P. B. Littlewood, *Numerical time propagation of quantum systems in radiation fields*, New J. Phys., 14 (2012), p. 105008.

[3] P. Amodio, L. Brugnano, and F. Iavernaro, *Analysis of spectral Hamiltonian boundary value methods (SHBVMs) for the numerical solution of ODE problems*, Numer. Algorithms, 83 (2019), pp. 1489–1508.

[4] P. Bader, S. Blanes, and N. Kopylov, *Exponential propagators for the Schrödinger equation with a time-dependent potential*, J. Chem. Phys., 148 (2018), p. 244109.

[5] P. Bader, A. Iserles, K. Kropielnicka, and P. Singh, *Efficient methods for linear Schrödinger equation in the semiclassical regime with time-dependent potential*, Proc. R. Soc. A, 472 (2016), pp. 689–720.

[6] J. Baglama, C. Fenu, L. Reichel, and G. Rodriguez, *Analysis of directed networks via partial singular value decomposition and Gauss quadrature*, Linear Algebra Appl., 456 (2014), pp. 93–121.

[7] B. Beckermann, *Complex Jacobi matrices*, J. Comput. Appl. Math., 127 (2001), pp. 17–65.

[8] M. Benzi, *Some uses of the field of values in numerical analysis*, Boll. Unione Mat. Ital., 14 (2020), pp. 159–177.

[9] M. Benzi, D. Bini, D. Kressner, H. Munthe-Kaas, and C. Van Loan, *Exploiting Hidden Structure in Matrix Computations: Algorithms and Applications*, Springer International Publishing, 2016.

[10] M. Benzi and P. Boito, *Quadrature rule-based bounds for functions of adjacency matrices*, Linear Algebra Appl., 433 (2010), pp. 637–652.

[11] ——, *Decay properties for functions of matrices over C^*-algebras*, Linear Algebra Appl., 456 (2014), pp. 174–198.

[12] ——, *Matrix functions in network analysis*, GAMM-Mitteilungen, 43 (2020), p. e202000012.

[13] M. Benzi, I. Daidone, C. Faccio, and L. Zanetti-Polzi, *Structural analysis of water networks*, J. Complex Netw., 11 (2023), p. cnad001.

[14] M. Benzi, E. Estrada, and C. Klymko, *Ranking hubs and authorities using matrix functions*, Linear Algebra Appl., 438 (2013), pp. 2447–2474.

[15] M. Benzi and G. H. Golub, *Bounds for the entries of matrix functions with application to preconditioning*, BIT, 39 (1999), pp. 417–438.

[16] M. Benzi and N. Razouk, *Decay bounds and $O(n)$ algorithms for approximating functions of sparse matrices*, Electron. Trans. Numer. Anal., 28 (2007), pp. 16–39.

[17] M. Benzi and V. Simoncini, *Decay bounds for functions of Hermitian matrices with banded or Kronecker structure*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 1263–1282.

[18] J. Bernier, S. Blanes, F. Casas, and A. Escorihuela-Tomàs, *Symmetric-conjugate splitting methods for linear unitary problems*, BIT, 63 (2023).

[19] S. Blanes and F. Casas, *A concise introduction to geometric numerical integration*, CRC Press, 2017.

[20] S. Blanes, F. Casas, P. Chartier, and A. Escorihuela-Tomàs, *On symmetric-conjugate composition methods in the numerical integration of differential equations*, Math. Comp., 91 (2022), pp. 1739–1761.

[21] S. Blanes, F. Casas, and A. Escorihuela-Tomàs, *Applying splitting methods with complex coefficients to the numerical integration of unitary problems*, J. Comput. Dyn., 9 (2022), pp. 85–101.

[22] ——, *Runge–Kutta–Nyström symplectic splitting methods of order 8*, Appl. Numer. Math., 182 (2022), pp. 14–27.

[23] S. Blanes, F. Casas, C. González, and M. Thalhammer, *Convergence analysis of high-order commutator-free quasi-Magnus exponential integrators for nonautonomous linear Schrödinger equations*, IMA J. Numer. Anal., 41 (2021), pp. 594–617.

[24] S. Blanes, F. Casas, and A. Murua, *Symplectic time-average propagators for the Schrödinger equation with a time-dependent Hamiltonian*, J. Chem. Phys., 146 (2017), p. 114109.

[25] S. Blanes, F. Casas, and A. Murua, *Splitting methods for differential equations*, Acta Numer., 33 (2024), p. 1–161.

[26] S. Blanes, F. Casas, J. A. Oteo, and J. Ros, *The Magnus expansion and some of its applications*, Phys. Rep., 470 (2009), pp. 151 – 238.

[27] S. Blanes, F. Casas, and M. Thalhammer, *High-order commutator-free quasi-Magnus exponential integrators for non-autonomous linear evolution equations*, Comput. Phys. Commun., 220 (2017), pp. 243–262.

[28] ——, *Convergence analysis of high-order commutator-free quasi-Magnus exponential integrators for nonautonomous linear evolution equations of parabolic type*, IMA J. Numer. Anal., 38 (2018), pp. 743–778.

[29] S. Blanes and P. C. Moan, *Practical symplectic partitioned Runge–Kutta and Runge–Kutta–Nyström methods*, J. Comput. Appl. Math., 142 (2002), pp. 313–330.

[30] C. Bonhomme, S. Pozza, and N. Van Buggenhout, *A new fast numerical method for the generalized Rosen-Zener model*, *arXiv:2311.04144 [math.NA]*, (2023).

[31] C. Brezinski, *Padé-type approximation and general orthogonal polynomials*, Internat. Ser. Numer. Math., Birkhäuser, 1980.

[32] C. Brezinski, M. Redivo Zaglia, and H. Sadok, *A breakdown-free Lanczos type algorithm for solving linear systems*, Numer. Math., 63 (1992), pp. 29–38.

[33] A. Brinkmann, *Introduction to average Hamiltonian theory. I. Basics*, Concepts in Magnetic Resonance Part A: Bridging Education and Research, 45A (2016), p. e21414.

[34] L. Brugnano, F. Iavernaro, and D. Trigiante, *A simple framework for the derivation and analysis of effective one-step methods for ODEs*, Appl Math Comput., 218 (2012), pp. 8475–8485.

[35] L. Brugnano, J. I. Montijano, and L. Rández, *On the effectiveness of spectral methods for the numerical solution of multi-frequency highly oscillatory Hamiltonian problems*, Numer. Algorithms, 81 (2018), pp. 345–376.

[36] A. Bultheel, P. Gonzalez-Vera, E. Hendriksen, and O. Njastad, *Orthogonal rational functions and tridiagonal matrices*, J. Comput. Appl. Math., 153 (2003), pp. 89–97.

[37] T. S. Chihara, *An Introduction to Orthogonal Polynomials*, vol. 13 of Mathematics and its Applications, Gordon and Breach Science Publishers, New York, 1978.

[38] G. Cybenko, *An explicit formula for Lanczos polynomials*, Linear Algebra Appl., 88 (1987), pp. 99–115.

[39] M. Dalgaard and F. Motzoi, *Fast, high precision dynamics in quantum optimal control theory*, J. Phys. B, 55 (2022), p. 085501.

[40] K. Deckers, *Orthogonal rational functions: Quadrature, recurrence and rational Krylov*, PhD thesis, Katholieke Universiteit, Leuven, 2009.

[41] K. Deckers and A. Bultheel, *Rational Krylov sequences and orthogonal rational functions*, tech. rep., Department of Computer Science, Katholieke Universiteit, Leuven, 2007.

[42] K. Deckers and A. Bultheel, *The existence and construction of rational Gauss-type quadrature rules*, Appl Math Comput., 218 (2012), pp. 10299–10320.

[43] S. Demko, *Inverses of band matrices and local convergence of spline projections*, SIAM J. Numer. Anal., 14 (1977), pp. 616–619.

[44] A. Draux, *Polynômes Orthogonaux Formels*, vol. 974 of Lecture Notes in Math., Springer-Verlag, Berlin, 1983.

[45] E. Estrada, *The Structure of Complex Networks: Theory and Applications*, Oxford University Press, 2011.

[46] E. Estrada and D. J. Higham, *Network properties revealed through matrix functions*, SIAM Rev., 52 (2010), pp. 696–714.

[47] E. Estrada and J. A. Rodríguez-Velázquez, *Subgraph centrality in complex networks*, Phys. Rev. E, 71 (2005), p. 056103.

[48] C. Fenu, D. Martin, L. Reichel, and G. Rodriguez, *Block Gauss and anti-Gauss quadrature with application to networks*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1655–1684.

[49] ——, *Network analysis via partial spectral factorization and Gauss quadrature*, SIAM J. Sci. Comput., 35 (2013), pp. A2046–A2068.

[50] R. W. Freund, *The Look-Ahead Lanczos Process for Large Nonsymmetric Matrices and Related Algorithms*, in Linear Algebra for Large Scale and Real-Time Applications, M. S. Moonen, G. H. Golub, and B. L. R. de Moor, eds., vol. 232 of NATO ASI Series, Kluwer Acad. Publ., Dordrecht, 1993, pp. 137–163.

[51] R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal, *An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices*, SIAM J. Sci. Comput., 14 (1993), pp. 137–158.

[52] R. W. Freund and M. Hochbruck, *Gauss quadratures associated with the Arnoldi process and the Lanczos algorithm*, in Linear Algebra for Large Scale and Real-Time Applications, M. S. Moonen, G. H. Golub, and B. L. R. de Moor, eds., vol. 232 of NATO ASI Series, Kluwer Acad. Publ., Dordrecht, 1993, pp. 377–380.

[53] A. Frommer, C. Schimmel, and M. Schweitzer, *Bounds for the decay of the entries in inverses and Cauchy–Stieltjes functions of certain sparse, normal matrices*, Numer. Linear Algebra Appl., 25 (2018), p. e2131.

[54] K. Gallivan, G. Grimme, and P. V. Dooren, *A rational Lanczos algorithm for model reduction*, Numer. Algorithms, 12 (1996), pp. 33–63.

[55] W. Gautschi, *Orthogonal Polynomials: Computation and Approximation*, Numer. Math. Sci. Comput., Oxford University Press, New York, 2004.

[56] P.-L. Giscard and C. Bonhomme, *Dynamics of quantum systems driven by time-varying Hamiltonians: Solution for the Bloch-Siegert Hamiltonian and applications to NMR*, Phys. Rev. Res., 2 (2020), p. 023081.

[57] P.-L. Giscard and M. Foroozandeh, *Exact solutions for the time-evolution of quantum spin systems under arbitrary waveforms using algebraic graph theory*, Comput. Phys. Commun., 282 (2023), p. 108561.

[58] P.-L. Giscard, K. Lui, S. J. Thwaite, and D. Jaksch, *An exact formulation of the time-ordered exponential using path-sums*, J. Math. Phys., 56 (2015), p. 053503.

[59] P.-L. Giscard and S. Pozza, *Lanczos-like algorithm for the time-ordered exponential: The $\star$-inverse problem*, Appl. Math., 65 (2020), pp. 807–827.

[60] G. H. Golub and G. Meurant, *Matrices, Moments and Quadrature with Applications*, Princeton Ser. Appl. Math., Princeton University Press, Princeton, 2010.

[61] C. Graf, A. Rund, C. S. Aigner, and R. Stollberger, *Accuracy and performance analysis for Bloch and Bloch-McConnell simulation methods*, J. Magn. Reson., 329 (2021), p. 107011.

[62] W. B. Gragg and A. Lindquist, *On the partial realization problem*, Linear Algebra Appl., 50 (1983), pp. 277–319.

[63] M. Grifoni and P. Hänggi, *Driven quantum tunneling*, Phys. Rep., 304 (1998), pp. 229–354.

[64] E. J. Grimme, *Krylov projection methods for model reduction*, PhD thesis, University of Illinois at Urbana-Champaign, Urbana, 1997.

[65] M. H. Gutknecht, *A completed theory of the unsymmetric Lanczos process and related algorithms. I*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 594–639.

[66] ——, *A completed theory of the unsymmetric Lanczos process and related algorithms. II*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 15–58.

[67] S. Güttel, *Rational Krylov methods for operator functions*, PhD thesis, TU Bergakademie Freiberg, Germany, 2010.

[68] S. Güttel, *Rational Krylov approximation of matrix functions: Numerical methods and optimal pole selection*, GAMM-Mitteilungen, 36 (2013), pp. 8–31.

[69] S. Güttel and L. Knizhnerman, *A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions*, BIT, 53 (2013), pp. 595–616.

[70] S. Güttel and M. Schweitzer, *A comparison of limited-memory Krylov methods for Stieltjes functions of Hermitian matrices*, SIAM J. Matrix Anal. Appl., 42 (2021), pp. 83–107.

[71] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, Springer-Verlag, 2006.

[72] N. J. Higham, *Functions of Matrices. Theory and Computation*, SIAM, Philadelphia, 2008.

[73] M. Hochbruck and C. Lubich, *On Magnus integrators for time-dependent Schrödinger equations*, SIAM J. Numer. Anal., 41 (2003), pp. 945–963.

[74] H. J. Hogben, M. Krzystyniak, G. T. P. Charnock, P. J. Hore, and I. Kuprov, Spinach – *A software library for simulation of spin dynamics in large spin systems*, J. Magn. Reson., 208 (2011), pp. 179–194.

[75] A. Iserles, K. Kropielnicka, and P. Singh, *Magnus–Lanczos methods with simplified commutators for the Schrödinger equation with a time-dependent potential*, SIAM J. Numer. Anal., 56 (2018), pp. 1547–1569.

[76] ——, *Compact schemes for laser–matter interaction in Schrödinger equation based on effective splittings of Magnus expansion*, Comput. Phys. Commun., 234 (2019), pp. 195–201.

[77] ——, *Solving Schrödinger equation in semiclassical regime with highly oscillatory time-dependent potentials*, J. Comput. Phys., 376 (2019), pp. 564–584.

[78] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna, *Lie-group methods*, Acta Numer., 9 (2000), pp. 215–365.

[79] K. L. Ivanov, K. R. Mote, M. Ernst, A. Equbal, and P. K. Madhu, *Floquet theory in magnetic resonance: Formalism and applications*, Prog. Nucl. Magn. Reson. Spectrosc., 126-127 (2021), pp. 17–58.

[80] S. Jaffard, *Propriétés des matrices "bien localisées" près de leur diagonale et quelques applications*, Ann. Inst. H. Poincaré C Anal. Non Linéaire, 7 (1990), pp. 461–476.

[81] C. Jagels and L. Reichel, *Recursion relations for the extended Krylov subspace method*, Linear Algebra Appl., 434 (2011), pp. 1716–1732.

[82] L. Knizhnerman and V. Simoncini, *Convergence analysis of the extended Krylov subspace method for the Lyapunov equation*, Numer. Math., 118 (2011), pp. 567–586.

[83] N. Kopylov, *Magnus-based geometric integrators for dynamical systems with time-dependent potentials*, PhD thesis, Universitat Politècnica de València, 2019.

[84] I. Kuprov, *Spin: From Basic Symmetries to Quantum Optimal Control*, Springer International Publishing, 2023.

[85] P. Kürschner and M. A. Freitag, *Inexact methods for the low rank solution to large scale lyapunov equations*, BIT, 60 (2020), pp. 1221–1259.

[86] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Research Nat. Bur. Standards, 45 (1950), pp. 255–282.

[87] ——, *Solution of systems of linear equations by minimized iterations*, J. Research Nat. Bur. Standards, 49 (1952), pp. 33–53.

[88] L. Lazzarino, *Numerical approximation of the time-ordered exponential for spin dynamic simulation*, PhD thesis, Univerzita Karlova, Prague, 2023.

[89] J. Liesen and Z. Strakoš, *Krylov subspace methods: principles and analysis*, Numer. Math. Sci. Comput., Oxford University Press, Oxford, 2013.

[90] G. López Lagomasino, L. Reichel, and L. Wunderlich, *Matrices, moments, and rational quadrature*, Linear Algebra Appl., 429 (2008), pp. 2540–2554.

[91] W. Magnus, *On the exponential solution of differential equations for a linear operator*, Comm. Pure Appl. Math., 7 (1954), pp. 649–673.

[92] E. Mananga, *Theoretical perspectives of spin dynamics in solid-state Nuclear Magnetic Resonance and physics*, J. Mod. Phys., 9 (2018), pp. 1645–1659.

[93] E. S. Mananga and T. Charpentier, *Introduction of the Floquet-Magnus expansion in solid-state Nuclear Magnetic Resonance spectroscopy*, J. Chem. Phys., 135 (2011), p. 044109.

[94] ——, *On the Floquet–Magnus expansion: Applications in solid-state Nuclear Magnetic Resonance and physics*, Phys. Rep., 609 (2016), pp. 1–49.

[95] G. V. Milovanović and A. S. Cvetković, *Complex Jacobi matrices and quadrature rules*, Filomat, 17 (2003), pp. 117–134.

[96] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Rev., 45 (2003), pp. 3–49.

[97] M. Ndong, H. Tal-Ezer, R. Kosloff, and C. P. Koch, *A Chebychev propagator with iterative time ordering for explicitly time-dependent Hamiltonians*, J. Chem. Phys., 132 (2010), p. 064105.

[98] B. N. Parlett, *Reduction to tridiagonal form and minimal realizations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 567–593.

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

[100] U. Peskin, R. Kosloff, and N. Moiseyev, *The solution of the time dependent Schrödinger equation by the (t,t') method: The use of global polynomial propagators for time dependent Hamiltonians*, J. Chem. Phys., 100 (1994), pp. 8849–8855.

[101] A. N. Pettitt, I. S. Weir, and A. G. Hart, *A conditional autoregressive Gaussian process for irregularly spaced multivariate data with application to modelling large sets of binary data*, Stat. Comput., 12 (2002), pp. 353–367.

[102] S. Pozza and N. Van Buggenhout, *A new matrix equation expression for the solution of non-autonomous linear systems of ODEs*, PAMM, 22 (2023), p. e202200117.

[103] M. S. Pranić and L. Reichel, *Rational Gauss quadrature*, SIAM J. Numer. Anal., 52 (2014), pp. 832–851.

[104] M. Rinelli, *Approximation of matrix functions arising in physics and network science: Theoretical and computational aspects*, PhD thesis, Scuola Normale Superiore, Pisa, 2024.

[105] A. Ruhe, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra Appl., 58 (1984), pp. 391–405.

[106] M. Ryckebusch, *A Fréchet-Lie group on distributions*, *arXiv:2307.09037 [math.FA]*, (2023).

[107] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, 2003.

[108] I. Schaefer, H. Tal-Ezer, and R. Kosloff, *Semi-global approach for propagation of the time-dependent Schrödinger equation for time-dependent and nonlinear problems*, J. Comput. Phys., 343 (2017), pp. 368–413.

[109] J. H. Shirley, *Solution of the Schrödinger equation with a Hamiltonian periodic in time*, Phys. Rev., 138 (1965), pp. B979–B987.

[110] V. Simoncini, *Variable accuracy of matrix-vector products in projection methods for eigencomputation*, SIAM J. Numer. Anal., 43 (2005), pp. 1155–1174.

[111] V. Simoncini, *The Lyapunov matrix equation. Matrix analysis from a computational perspective*, in Quaderno UMI - Topics in Mathematics, vol. 55, UMI, 2015, pp. 157–174.

[112] V. Simoncini, *Computational methods for linear matrix equations*, SIAM Rev., 58 (2016), pp. 377–441.

[113] V. Simoncini and D. B. Szyld, *Theory of inexact Krylov subspace methods and applications to scientific computing*, SIAM J. Sci. Comput., 25 (2003), pp. 454–477.

[114] M. Stoll, *A literature survey of matrix methods for data science*, GAMM-Mitteilungen, 43 (2020).

[115] Z. Strakoš, *Model reduction using the Vorobyev moment problem*, Numer. Algorithms, 51 (2009), pp. 363–379.

[116] D. R. Taylor, *Analysis of the look ahead Lanczos algorithm*, PhD thesis, University of California, Berkeley, 1982.

[117] J. C. Tremblay and T. Carrington, *Using preconditioned adaptive step size Runge-Kutta methods for solving the time-dependent Schrödinger equation*, J. Chem. Phys., 121 (2004), pp. 11535–11541.

[118] N. Van Buggenhout, M. Van Barel, and R. Vandebril, *Biorthogonal rational Krylov subspace methods*, Electron. Trans. Numer. Anal., 51 (2019), pp. 451–468.

[119] M. Vogl, P. Laurell, A. D. Barr, and G. A. Fiete, *Flow equation approach to periodically driven quantum systems*, Phys. Rev. X, 9 (2019), p. 021037.

[120] V. Volterra, *Sopra una proprietà generale delle equazioni integrali ed integrodifferenziali*, Atti della Reale Accademia dei Lincei, Rendiconti - Classe di Scienze Fisiche Matematiche e Naturali, 20 (1911), pp. 79–88.

[121] ——, *Teoria delle potenze, dei logaritmi e delle funzioni di composizione*, Atti della Reale Accademia dei Lincei, Meomorie - Classe di Scienze Fisiche Matematiche e Naturali, 11 (1916), pp. 167–249.

[122] V. Volterra and J. Pérès, *Leçons sur la composition et les fonctions permutables*, vol. 29, Gauthier-Villars, 1924.

[123] H. Wang and Q. Ye, *Error bounds for the Krylov subspace methods for computations of matrix exponentials*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 155–187.

[124] Q. Ye, *Error bounds for the Lanczos methods for approximating matrix exponentials*, SIAM J. Numer. Anal., 51 (2013), pp. 68–87.

Reprints of papers

In this final part, we include the 11 papers listed hereafter in the same order as in the preface:

[C1] S. Pozza, and F. Tudisco, *On the stability of network indices defined by means of matrix functions*, SIAM J. Matrix Anal. Appl. 39 (4) (2018), pp. 1521–1546. DOI: 10.1137/17M1133920

[C2] S. Pozza, and V. Simoncini, *Inexact Arnoldi residual estimates and decay properties for functions of non-Hermitian matrices*, BIT 59 (2019), pp. 969–986. DOI: 10.1007/s10543-019-00763-6

[C3] S. Pozza, and V. Simoncini, *Functions of rational Krylov space matrices and their decay properties*, Numer. Math. 148 (2021), pp. 99–126. DOI: 10.1007/s00211-021-01198-4

[C4] S. Pozza, M. S. Pranić, and Z. Strakoš, *Gauss quadrature for quasi-definite linear functionals*, IMA J. Numer. Anal. 37 (3) (2017), pp. 1468–1495. DOI: 10.1093/imanum/drw032

[C5] S. Pozza, M. S. Pranić, and Z. Strakoš, *The Lanczos algorithm and complex Gauss quadrature*, Electron. Trans. Numer. Anal. 50 (2018), pp. 1–19. DOI: 10.1553/etna_vol50s1

[C6] S. Pozza, and M. S. Pranić, *The Gauss quadrature for general linear functionals, Lanczos algorithm, and minimal partial realization*, Numer. Algorithms 88 (2021), pp. 647–678. DOI: 10.1007/s11075-020-01052-y

[C7] D. Palitta, S. Pozza, and V. Simoncini, *The short-term rational Lanczos method and applications*, SIAM J. Sci. Comput. 44 (4) (2022), pp. A2843–A2870. DOI: 10.1137/21M1403254

[C8] P-L. Giscard, and S. Pozza, *Tridiagonalization of systems of coupled linear differential equations with variable coefficients by a Lanczos-like method*, Linear Algebra Appl. 624 (2021), pp. 153–173.
DOI: 10.1016/j.laa.2021.04.011

[C9] P-L. Giscard, and S. Pozza, *A Lanczos-like method for non-autonomous linear ordinary differential equations*, Boll. Unione Mat. Ital. 16 (2023), pp. 81–102.
DOI: 10.1007/s40574-022-00328-6

[C10] S. Pozza, *A new closed-form expression for the solution of ODEs in a ring of distributions and its connection with the matrix algebra*, Linear Multilinear Algebra (2024), online.
DOI: 10.1080/03081087.2024.2303058

[C11] S. Pozza, and N. Van Buggenhout, *A new Legendre polynomial-based approach for non-autonomous linear ODEs*, Electron. Trans. Numer. Anal. 60 (2024), pp. 292–326
DOI: 10.1553/etna_vol60s292