

Article

Shared-Pole Carathéodory–Fejér Approximations for Linear Combinations of φ -Functions

Awad H. Al-Mohy 

Department of Mathematics, King Khalid University, Abha 61421, Saudi Arabia; ahalmohy@kku.edu.sa

Abstract

We develop a shared denominator Carathéodory–Fejér (CF) method for efficiently evaluating linear combinations of φ -functions for matrices whose spectrum lies in the negative real axis, as required in exponential integrators for large stiff ODE systems. This entire family is approximated with a single set of poles (a common denominator). The shared pole set is obtained by assembling a stacked Hankel matrix from Chebyshev boundary data for all target functions and computing a single SVD; the zeros of the associated singular-vector polynomial, mapped via the standard CF slit transform, yield the poles. With the poles fixed, per-function residues and constants are recovered by a robust least squares fit on a suitable grid of the negative real axis. For any linear combination of resolvent operators applied to right-hand sides, the evaluation reduces to one shifted linear solve per pole with a single combined right-hand side, so the dominant cost matches that of computing a single φ -function action. Numerical experiments indicate geometric convergence at a rate consistent with Halphen’s constant, and for highly stiff problems our algorithm outperforms existing Taylor and Krylov polynomial-based algorithms.

Keywords: matrix functions; Carathéodory–Fejér approximation; shared poles; exponential integrators; φ -functions

MSC: 15A16; 65F60; 65L05



Academic Editors: Ching-Sung Liu and Xiaobing Feng

Received: 1 November 2025

Revised: 9 December 2025

Accepted: 11 December 2025

Published: 14 December 2025

Citation: Al-Mohy, A.H. Shared-Pole Carathéodory–Fejér Approximations for Linear Combinations of φ -Functions. *Mathematics* **2025**, *13*, 3985. <https://doi.org/10.3390/math13243985>

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Over the last two decades, the φ -functions have become central objects in the design of exponential integrators for stiff systems; see, e.g., the survey by Hochbruck and Ostermann [1], the review by Minchev and Wright [2], and the references therein. Typical target problems include semilinear diffusion–reaction equations, Schrödinger-type equations, advection–diffusion–reaction systems, and more general evolution equations obtained by spatial discretization of parabolic or highly oscillatory PDEs in physics and engineering leading to a system of ODEs:

$$u'(t) = A u(t) + g(u(t), t), \quad u(t_0) = u_0, \quad u(t) \in \mathbb{C}^d, \quad (1)$$

where $A \in \mathbb{C}^{d \times d}$ is obtained from spatial discretization and g is a nonlinear vector function. In such applications, the matrix A is usually large and sparse and in many cases it is negative semidefinite.

Exponential integrators have emerged as a successful class of numerical methods for systems of ODEs. A broad class of exponential integrators reduces each stage to a linear combination of φ -function actions of the form (see, e.g., [1,3–10]):

$$w = \sum_{j=0}^p \varphi_j(A) v_j. \quad (2)$$

For a scalar argument $z \in \mathbb{C}$, the functions φ_k admit a convenient integral representation

$$\varphi_0(z) = e^z, \quad \varphi_k(z) = \frac{1}{(k-1)!} \int_0^1 e^{(1-\tau)z} \tau^{k-1} d\tau, \quad k \geq 1,$$

which, by the holomorphic functional calculus, extends directly to matrices:

$$\varphi_k(A) = \frac{1}{(k-1)!} \int_0^1 e^{(1-\tau)A} \tau^{k-1} d\tau, \quad k \geq 1.$$

Equivalently, one can use the power series representation $\varphi_k(A) = \sum_{m=0}^{\infty} \frac{A^m}{(m+k)!}$.

As an illustration of the pivotal role of the φ -functions in exponential integrators, consider the problem (1) with step size $h > 0$. The classical fourth-order exponential time differencing Runge–Kutta method (ETDRK4) of Cox and Matthews [9], modified later by Kassam and Trefethen [11] for stability, reads as follows. Given $u_n \approx u(t_n)$, define the stage values

$$\begin{aligned} a_n &= e^{\frac{h}{2}A} u_n + A^{-1} (e^{\frac{h}{2}A} - I) g(u_n, t_n), \\ b_n &= e^{\frac{h}{2}A} u_n + A^{-1} (e^{\frac{h}{2}A} - I) g(a_n, t_n + \frac{h}{2}), \\ c_n &= e^{\frac{h}{2}A} a_n + A^{-1} (e^{\frac{h}{2}A} - I) (2g(b_n, t_n + \frac{h}{2}) - g(u_n, t_n)). \end{aligned}$$

Then, the step from t_n to $t_{n+1} = t_n + h$ is given by

$$\begin{aligned} u_{n+1} &= e^{hA} u_n + h^{-2} A^{-3} \left\{ [-4I - hA + e^{hA} (4I - 3hA + (hA)^2)] g(u_n, t_n) \right. \\ &\quad + 2[2I + hA + e^{hA} (-2I + hA)] (g(a_n, t_n + \frac{h}{2}) + g(b_n, t_n + \frac{h}{2})) \\ &\quad \left. + [-4I - 3hA - (hA)^2 + e^{hA} (4I - hA)] g(c_n, t_n + h) \right\}. \end{aligned}$$

Using the fact that ([12], Section 10.7.4)

$$\varphi_0(z) = e^z, \quad \varphi_1(z) = \frac{e^z - 1}{z}, \quad \varphi_2(z) = \frac{e^z - 1 - z}{z^2}, \quad \varphi_3(z) = \frac{e^z - 1 - z - \frac{1}{2}z^2}{z^3},$$

the ETDRK4 scheme can be written in the form

$$\begin{aligned} a_n &= \varphi_0\left(\frac{h}{2}A\right) u_n + \frac{h}{2} \varphi_1\left(\frac{h}{2}A\right) g(u_n, t_n), \\ b_n &= \varphi_0\left(\frac{h}{2}A\right) u_n + \frac{h}{2} \varphi_1\left(\frac{h}{2}A\right) g(a_n, t_n + \frac{h}{2}), \\ c_n &= \varphi_0\left(\frac{h}{2}A\right) a_n + \frac{h}{2} \varphi_1\left(\frac{h}{2}A\right) (2g(b_n, t_n + \frac{h}{2}) - g(u_n, t_n)). \end{aligned}$$

The step from t_n to $t_{n+1} = t_n + h$ is then given by

$$\begin{aligned} u_{n+1} = & \varphi_0(hA) u_n \\ & + h\varphi_1(hA) g(u_n, t_n) \\ & + h\varphi_2(hA) \left[-3g(u_n, t_n) + 2g(a_n, t_n + \frac{h}{2}) + 2g(b_n, t_n + \frac{h}{2}) - g(c_n, t_n + h) \right] \\ & + 4h\varphi_3(hA) \left[g(u_n, t_n) - g(a_n, t_n + \frac{h}{2}) - g(b_n, t_n + \frac{h}{2}) + g(c_n, t_n + h) \right]. \end{aligned}$$

This formulation is algebraically equivalent to the original ETDRK4 scheme of Cox and Matthews, but all matrix coefficients are expressed as linear combinations of the φ -functions φ_j , $j = 0, 1, 2, 3$, making the scheme stable and less prone to subtractive cancellation [11].

The demand of computing such a combination in (2) has led to a substantial body of work devoted specifically to the numerical evaluation of matrix φ -functions. Niesen and Wright [8] proposed a Krylov subspace algorithm for computing $\varphi_k(A)v$ that is now widely used in exponential integrator codes. Their algorithm has been improved and extended to several Krylov-based algorithms that simultaneously evaluate several linear combinations of the form (2); see, e.g., Luan et al. [13], Gaudreault et al. [5], and Caliarì et al. [14]. Recently, Al-Mohy [3] proposed an algorithm based on the Taylor series that simultaneously calculates several linear combinations of the form (2). For the implementation of rational Krylov subspaces, see, e.g., Moret [15], Bergermann and Stoll [16], and the references therein.

For algorithms of φ -functions of medium size, Berland, Skaflestad, and Wright [17] developed the EXPINT MATLAB package, emphasizing that the stability and efficiency of exponential integrators hinge on the accurate evaluation of the underlying φ -functions. A more recent contribution includes the scaling and recovering algorithm by Al-Mohy and Liu [18], which extends the work of Al-Mohy and Higham [19] for the matrix exponential.

In this context, our goal in the present work is to develop a CF-based rational approximation framework that exploits a shared set of poles for the family $\{\varphi_j\}_{j=0}^p$. By constructing near-best scalar rational approximants with a common denominator on the negative real axis, and then lifting them to the matrix level, we obtain an efficient mechanism for evaluating general linear combinations of the form (2) using only one set of shifted factorizations $(A - \theta_\ell I)$ across all indices j and all vectors v_j .

In this manuscript, we focus on approximating the linear combination (2) simultaneously by constructing a shared-pole CF rational approximation for each φ_j of the form [20]

$$r_j(x) = r_\infty^{(j)} + \sum_{\ell=1}^n \frac{\eta_\ell^{(j)}}{x - \theta_\ell}, \quad x \in \mathbb{R}^- := (-\infty, 0], \quad j = 0, \dots, p, \quad (3)$$

where the pole set $\{\theta_\ell\}_{\ell=1}^n$ is common to all j , while the residues $\eta_\ell^{(j)}$ and constants $r_\infty^{(j)}$ depend on j . Equivalently,

$$r_j(x) = \frac{p_{j,n}(x)}{q_n(x)}, \quad q_n(x) = \prod_{\ell=1}^n (x - \theta_\ell),$$

so that all r_j share the same denominator q_n and differ only in their numerators $p_{j,n}$.

The CF approach proposed by Trefethen [21,22] and Trefethen and Gutknecht [23] constructs near-best real rational approximants to scalar functions from boundary data using singular structure of a Hankel matrix. Its roots trace back to the early 20th-century work of Carathéodory and Fejér on the relation between the extrema of harmonic functions and their coefficients [24]; an extensive historical review is given in [23]. The use of CF approximants for matrix functions goes back to Trefethen, Weideman, and Schmelzer ([20], Section 4) for the matrix exponential. Schmelzer and Trefethen [25] subse-

quently used CF approximants to evaluate actions $\varphi_j(A)v$, typically with distinct pole sets for each j . They also advocate a common-pole strategy exploiting block-matrix identities among the φ -functions ([25], Section 4), yielding rational approximants that share a single denominator and thereby allowing reuse of the same shifted factorizations $A - \theta_\ell I$ across multiple right-hand sides. They did not develop a general framework for arbitrary linear combinations of the form (2) with heterogeneous right-hand sides, which we provide below. Moreover, they explicitly remark that these approximants are far from optimal.

A key advantage of our approach is that the required approximation degree (and hence the accuracy up to the conditioning of the problems) of the CF rational approximants is its tendency to be independent of the spectral radius of A : the same shared pole set yields geometric decay uniformly on $\mathbb{R}^-$, so large spectral radius does not force a higher degree n . By contrast, the algorithms based on Taylor series, like that of Al-Mohy [3] and based on the standard Krylov like those of [5,8,14] typically require degrees that grow with $\|A\|$ [16].

This paper is organized as follows: In Section 2, we show how the shared poles, residues, and constants can be computed and propose an algorithm for their computation. In Section 3, we present several results showing that the shared-pole rational approximants retain the exponential accuracy. Section 4 presents the main algorithm for computing the linear combination in (2). Next we present our numerical experiments in Section 5. Finally, we draw some concluding remarks in Section 6.

In the next section, we describe how to construct the shared pole set $\{\theta_\ell\}_{\ell=1}^n$ together with the per-function residues $\eta_\ell^{(j)}$ and constants $r_\infty^{(j)}$ that define the CF approximants r_j in (3).

2. Constructing Shared Poles and Per-Function Residues and Constants

This section builds on the CF analysis and practice of [20,23,25]. Let $f_0, \dots, f_p$ be analytic in $\mathbb{C} \setminus \mathbb{R}^-$ and continuous up to $\mathbb{R}^-$. Assume further that $|f_j(z)| \rightarrow 0$ as $|z| \rightarrow \infty$ in a left sector containing $\mathbb{R}^-$, for $j = 0, \dots, p$.

We consider the conformal map

$$z(w) = \sigma \left(\frac{w-1}{w+1} \right)^2, \quad (4)$$

where $\sigma > 0$ is a suitably chosen scaling parameter. It maps the interior of the unit disk onto $\mathbb{C} \setminus \mathbb{R}^-$ and the unit circle $w = e^{i\theta}$, $0 \leq \theta \leq 2\pi$, onto $\mathbb{R}^-$, on which

$$z(e^{i\theta}) = \sigma \left(\frac{e^{i\theta} - 1}{e^{i\theta} + 1} \right)^2 = \sigma \frac{t-1}{t+1}, \quad t = \cos \theta.$$

Since $(t-1)/(t+1) \rightarrow -\infty$ as $t \rightarrow -1^+$ and $f_j(z) \rightarrow 0$ as $z \rightarrow -\infty$ along $\mathbb{R}^-$ by assumption, we set

$$F_j(-1) := \lim_{t \rightarrow -1^+} f_j \left(\sigma \frac{t-1}{t+1} \right) = 0,$$

so that F_j is continuous on $[-1, 1]$. For each $j = 0, \dots, p$ we define the boundary function

$$F_j(t) = f_j(z(w)) = f_j \left(\sigma \frac{t-1}{t+1} \right), \quad t \in [-1, 1].$$

Since F_j is continuous on $[-1, 1]$, it has the Chebyshev expansion

$$F_j(t) = \sum_{k=0}^{\infty} c_k^{(j)} T_k(t),$$

where $c_0^{(j)} = a_0^{(j)}$ and $c_k^{(j)} = 2a_k^{(j)}$ for $k \geq 1$, with

$$a_k^{(j)} = \frac{1}{\pi} \int_{-1}^1 \frac{F_j(t) T_k(t)}{\sqrt{1-t^2}} dt = \frac{1}{2\pi} \int_0^{2\pi} F_j(\cos \theta) \cos(k\theta) d\theta,$$

since the Chebyshev polynomial $T_k(t) = \cos(k \arccos t)$. Numerically these integrals are approximated using the composite trapezoidal rule in θ , which converges geometrically for 2π -periodic analytic functions ([26], Theorem 3.1). The fast Fourier transform (FFT) can then be used to compute the coefficients efficiently because of periodicity ([27], Section 5.5).

Trefethen and Gutknecht ([23], Section 1) truncate F_j as

$$F_j^{(K)}(t) = \sum_{k=0}^K c_k^{(j)} T_k(t),$$

and write $t = (w + \bar{w})/2$ for $|w| = 1$, so that $T_k(t) = (w^k + \bar{w}^k)/2$ for $k \geq 1$, yielding

$$F_j^{(K)}(t) = c_0^{(j)} + \frac{1}{2}(g_j(w) + g_j(\bar{w})), \quad g_j(w) = \sum_{k=1}^K c_k^{(j)} w^k. \quad (5)$$

The key consequence of the CF theorem ([21], Theorem 3) is as follows: The polynomial g_j in (5) has a unique best rational approximation of type (n, n) , denoted $r_{nn}^{*(j)}$, with all poles lying outside the unit disk, and

$$g_j(w) - r_{nn}^{*(j)}(w) = s_{n+1} w^K B_j(w), \quad B_j(w) = \frac{u_1 + u_2 w + \dots + u_K w^{K-1}}{v_K + v_{K-1} w + \dots + v_1 w^{K-1}}, \quad (6)$$

where $B_j(w)$ is a finite Blaschke product of degree n and $v = [v_1, v_2, \dots, v_K]^T$ and $u = [u_1, u_2, \dots, u_K]^T$ are the right and left singular vectors, respectively, associated with the singular value s_{n+1} of the Hankel matrix

$$H^{(j)} = \begin{bmatrix} c_1^{(j)} & c_2^{(j)} & \dots & c_{K-1}^{(j)} & c_K^{(j)} \\ c_2^{(j)} & c_3^{(j)} & \dots & c_K^{(j)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{K-1}^{(j)} & c_K^{(j)} & \dots & 0 & 0 \\ c_K^{(j)} & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (7)$$

That is, $H^{(j)}v = s_{n+1}u$; see ([22], Proposition 3.1) and ([23], Theorem 1).

In our applications the functions f_j are real-valued on $\mathbb{R}^-$ (for example, the scalar φ -functions), and, hence, the boundary functions F_j are real-valued on $[-1, 1]$. Consequently, all Chebyshev coefficients $c_k^{(j)}$ are real and the Hankel matrices $H^{(j)}$ are real. It follows that we may take the singular vectors u and v in the relation $H^{(j)}v = s_{n+1}u$ to be real, so that the associated Blaschke product B_j and the type (n, n) rational approximant $r_{nn}^{*(j)}$ have real coefficients and poles occurring in complex conjugate pairs. For simplicity, we adopt this real-valued setting throughout the paper. The construction extends in a straightforward way to complex-valued functions f_j , at the expense of working with complex Hankel matrices and singular vectors; we do not pursue this generality here.

Trefethen ([22], p. 303) shows that the best rational approximant $r_{nn}^{*(j)}$ has n poles counted with multiplicity and they are the zeros of the denominator polynomial of $B(w)$ in (6) that lie *outside* the unit circle. Thus, the residues and constants associated with partial fraction form of $r_{nn}^{*(j)}$ are also encoded in the vector u representing the numerator of $B(w)$;

see Trefethen and Gutknecht ([23], Section 1) for further details. After obtaining the poles and residues in the w -plane, the conformal map $z(w)$ defined in (4) is applied to transplant them in the z -plane and construct a *near-best* rational approximation for f_j .

Our novel approach in this paper is to construct a family of rational approximants $r_{nn}^{(j)}$ in a near-best sense for the functions f_j such that $r_{nn}^{(j)}$, $j = 0, \dots, p$, have the same denominator. That is, they share the same set of poles. We begin by forming the Hankel matrices $H^{(j)}$, $j = 0 \dots, p$, as prescribed above and then build the block-row matrix

$$H = \begin{bmatrix} \sqrt{w_0} H^{(0)} \\ \sqrt{w_1} H^{(1)} \\ \vdots \\ \sqrt{w_p} H^{(p)} \end{bmatrix},$$

which we refer to as the *stacked weighted* Hankel matrix with positive weights w_j . We then compute the thin singular value decomposition (SVD) $H = USV^T$ and let $v_{n+1} \in \mathbb{R}^K$ denote the right singular vector associated with the $(n+1)$ st singular value s_{n+1} (singular values ordered nonincreasingly). Equivalently, v_{n+1} minimizes the quadratic form

$$\sum_{j=0}^p w_j \|H^{(j)} v\|_2^2 \quad \text{over unit vectors } v \text{ orthogonal to } \text{span}\{v_1, \dots, v_n\}. \quad (8)$$

The vector v_{n+1} is regarded as the coefficients of the polynomial

$$Q(q) = \sum_{i=1}^K (v_{n+1})_i q^{K-i}.$$

We evaluate the zeros $\{q_\ell\}$ of Q lying outside the unit disk and retain the n zeros nearest to the unit circle. Finally, we map them to the slit plane via the conformal map $z(w)$ in (4) as

$$\theta_\ell = z(q_\ell), \quad \ell = 1, \dots, n.$$

The resulting $\{\theta_\ell\}_{\ell=1}^n$ constitute a shared pole set (a common denominator) for all functions $\{f_j\}$. With these poles fixed, we approximate each f_j by a shared denominator partial fraction

$$r_j(x) = r_\infty^{(j)} + \sum_{\ell=1}^n \frac{\eta_\ell^{(j)}}{x - \theta_\ell}, \quad x \in \mathbb{R}^-.$$

To compute the residues $\eta_\ell^{(j)}$ and constants $r_\infty^{(j)}$, we choose suitable grid points $\{x_k\}_{k=1}^\tau \subset [-M, 0] \subset \mathbb{R}^-$ (for some truncation parameter $M > 0$ proportional to σ in practice) and solve a robust least squares problem

$$\min_{r_\infty^{(j)}, \eta^{(j)}} \|L \beta^{(j)} - f_j(x)\|_2, \quad L = \begin{bmatrix} \mathbf{1} & ((x_k - \theta_\ell)^{-1})_{k,\ell} \end{bmatrix}, \quad \beta^{(j)} = \begin{bmatrix} r_\infty^{(j)} \\ \eta_1^{(j)} \\ \vdots \\ \eta_n^{(j)} \end{bmatrix}, \quad (9)$$

where $f_j(x) = [f_j(x_1), f_j(x_2), \dots, f_j(x_\tau)]^T$ and $\mathbf{1}$ denotes the column vector whose entries are all ones. Another way to compute the residues $\eta_\ell^{(j)}$ and constants $r_\infty^{(j)}$ is to use the $(n+1)$ st vector of U , denote it u . By blocking u in accordance with the rows of H ,

$$u = [u^{(0)}, u^{(1)}, \dots, u^{(p)}]^T,$$

we have

$$H^{(j)} v_{n+1} = \frac{s_{n+1}}{\sqrt{w_j}} u^{(j)}, \quad j = 0, \dots, p.$$

This shows that the blocks $u^{(j)}$ play, for each f_j , a role analogous to the left singular vector in the one-function CF construction. In principle, one could recover residue information from these vectors along the lines of [22,23]. In Algorithm 1, however, we opt for the simpler and more robust least squares procedure (9); we do not exploit this alternative in our implementation.

Algorithm 1 Shared Denominator CF on $\mathbb{R}^-$: poles $\{\theta_\ell\}$, residues $\{\eta_\ell^{(j)}\}$, and constants $r_\infty^{(j)}$.

Input: Functions $f_0, \dots, f_p$ satisfying the assumptions in Section 2; target degree n ; CF parameters: scale $\sigma > 0$, truncation $K > n$, number of Chebyshev points $N > K$; positive weights $w_0, \dots, w_p$ (default $w_j \equiv 1$); residue fit parameters: LS grid size d ; far-left bound M (e.g., $M \simeq 100\sigma$).

Output: Shared poles $\{\theta_\ell\}_{\ell=1}^n$; per-function constants $r_\infty^{(j)}$ and residues $\{\eta_\ell^{(j)}\}_{\ell=1}^n$.

1: **Map and sample on the unit circle**

2: Define $x(t) = \sigma \frac{t-1}{t+1}$ and sample $t_k = \cos(2\pi i k/N)$, $k = 0, \dots, N-1$.

3: **for** $j = 0, \dots, p$ **do**

4: $F_j(t_k) \leftarrow f_j(x(t_k))$;

5: Compute the Cheb. coef. $c_0^{(j)}, \dots, c_K^{(j)}$ exploiting FFT.

6: Form $K \times K$ Hankel matrices $H^{(j)}$.

▷ see (7)

7: **end for**

8: **Stacked-weighted Hankel SVD and pole extraction**

9: Build $H = \begin{bmatrix} \sqrt{w_0} H^{(0)} \\ \sqrt{w_1} H^{(1)} \\ \vdots \\ \sqrt{w_p} H^{(p)} \end{bmatrix}$.

10: Compute the thin SVD $H = USV^T$.

▷ singular values nonincreasing.

11: Extract v_{n+1} , the $(n+1)$ st column of V .

12: Compute the roots of the polynomial $\sum_{i=1}^K (v_{n+1})_i q^{K-i}$.

13: Select the n roots nearest to the unit circle with $|q| > 1$, denote $\{q_\ell\}_{\ell=1}^n$.

14: $\theta_\ell = \sigma \left(\frac{q_\ell - 1}{q_\ell + 1} \right)^2$, $\ell = 1, \dots, n$.

▷ shared poles

15: **Per-function residues and constants via x -plane least squares on $[-M, 0]$**

16: Build a log-dense grid $\{x_k\}_{k=1}^\tau$ on $[-M, 0]$.

17: Construct the matrix:

$$L = \begin{bmatrix} \mathbf{1} & ((x_k - \theta_\ell)^{-1})_{k,\ell} \end{bmatrix} \in \mathbb{C}^{\tau \times (n+1)}.$$

18: Compute the thin QR factorization $L = QR$.

19: **for** $j = 0, \dots, p$ **do**

20: $y \leftarrow [f_j(x_k)]_{k=1}^\tau$

21: Solve $R\beta^{(j)} = Q^*y$

▷ least squares for $\beta^{(j)} \in \mathbb{C}^{n+1}$

22: Set $r_\infty^{(j)} = \beta_1^{(j)}$ and $\eta_\ell^{(j)} = \beta_{\ell+1}^{(j)}$ for $\ell = 1, \dots, n$.

23: **end for**

24: **return** shared poles $\{\theta_\ell\}$, constants $\{r_\infty^{(j)}\}$, residues $\{\eta_\ell^{(j)}\}$.

The quadratic form in (8) makes clear how the single denominator is optimized across the family $\{f_j\}$. Larger w_j bias the pole set toward features of f_j (improving its fit, potentially at the expense of others). A simple and effective choice is $w_0 = \dots = w_p$, so each f_j contributes equally to the shared poles. In the terminology of [25], the “common poles” used in practice correspond to taking the denominator from a *single base function* (often e^x); in our framework, this is the extreme weight choice $w_0 = 1$ and $w_j = 0$ for $1 \leq j \leq p$. This perspective aligns with their comment that such approximations are “far from optimal.”

3. Exponential Accuracy of Shared-Denominator Rational Approximants

Let $D := \mathbb{C} \setminus \mathbb{R}^-$. Write $\mathcal{A}(D)$ for the class of functions analytic in D and continuous up to $\mathbb{R}^-$ in the nontangential sense. The results below show that shared denominator (shared-pole) CF rational approximants retain *exponential* accuracy. Trefethen, Weideman, and Schmelzer [20] observed that quadrature formulas can be interpreted as rational approximations (and conversely). Consequently, it suffices to establish exponential convergence of the periodic trapezoidal rule for the boundary integral representation (after mapping D to the exterior disk); this holds whenever the integrand extends analytically to a fixed annulus about the unit circle, and thus yields exponential accuracy for the CF approximants; see, e.g., ([26], Theorem 2.1).

Theorem 1. Fix a finite family $F = \{f_0, \dots, f_p\} \subset \mathcal{A}(D)$. There exist constants $C > 0$ and $\rho > 1$, depending only on F and on $\mathbb{R}^-$, such that for each $n \in \mathbb{N}$ there are polynomials

$$q_n \text{ (deg } q_n \leq n), \quad p_{j,n} \text{ (deg } p_{j,n} \leq n, 0 \leq j \leq p),$$

with a single denominator q_n (independent of j) satisfying

$$\max_{0 \leq j \leq p} \sup_{x \leq 0} |f_j(x) - p_{j,n}(x)/q_n(x)| \leq C \rho^{-n}.$$

Proof. Let $\Phi : D \rightarrow \{w : |w| > 1\}$ be a conformal bijection with $\Phi(\infty) = \infty$ and $\Phi'(\infty) > 0$, and let $g_D(z, \infty) = \log |\Phi(z)|$. For any $\sigma > 1$ define the level curve $\Gamma_\sigma := \{z \in D : |\Phi(z)| = \sigma\}$ and parameterize it by $\zeta(\theta) = \Phi^{-1}(\sigma e^{i\theta})$, $0 \leq \theta < 2\pi$. For $x \in \mathbb{R}^-$,

$$f_j(x) = \frac{1}{2\pi} \int_0^{2\pi} \frac{f_j(\zeta(\theta)) \zeta'(\theta)}{\zeta(\theta) - x} d\theta = \frac{1}{n} \sum_{m=0}^{n-1} \frac{\omega_m f_j(\zeta_m)}{\zeta_m - x} + \mathcal{E}_{j,n}(x),$$

with $\zeta_m = \zeta(2\pi m/n)$ and $\omega_m = \zeta'(2\pi m/n)$. The integrand is analytic and bounded in a strip $|\operatorname{Im} \theta| < a$, so by the exponentially convergent trapezoidal rule, $\sup_{x \in K} |\mathcal{E}_{j,n}(x)| \leq C_1 \eta^{-n}$ for some $\eta > 1$ independent of j and x ; see ([26], Theorem 2.1). Clearing denominators gives a rational $r_{j,n}(x) = p_{j,n}(x)/q_n(x)$ with $q_n(x) = \prod_{m=0}^{n-1} (\zeta_m - x)$, common to all j . Taking $\rho = \eta$ proves the claim. $\square$

This leads to two important corollaries. The first shows that any linear combination admits shared denominator approximants with exponential accuracy; the second treats matrix arguments.

Corollary 1. For any $\alpha \in \mathbb{C}^{p+1}$, there exist $C_\alpha > 0$ and $\rho > 1$ such that the combination $F_\alpha := \sum_{j=0}^p \alpha_j f_j$ admits type (n, n) shared denominator approximants with

$$\sup_{x \leq 0} |F_\alpha(x) - P_n(x)/q_n(x)| \leq C_\alpha \rho^{-n}.$$

Proof. Let $r_{j,n}(x) = p_{j,n}(x)/q_n(x)$ be the shared denominator approximants from Theorem 1, so that

$$\sup_{x \leq 0} |f_j(x) - r_{j,n}(x)| \leq C_j \rho^{-n}, \quad 0 \leq j \leq p.$$

For $\alpha = (\alpha_0, \dots, \alpha_p)$ define $P_n(x) := \sum_{j=0}^p \alpha_j p_{j,n}(x)$ and $R_n(x) := P_n(x)/q_n(x)$. Then,

$$|F_\alpha(x) - R_n(x)| = \left| \sum_{j=0}^p \alpha_j (f_j(x) - r_{j,n}(x)) \right| \leq \sum_{j=0}^p |\alpha_j| C_j \rho^{-n};$$

hence, $\sup_{x \leq 0} |F_\alpha(x) - P_n(x)/q_n(x)| \leq C_\alpha \rho^{-n}$ with $C_\alpha := \sum_{j=0}^p |\alpha_j| C_j$. $\square$

Corollary 2. Let A be normal with spectrum $\sigma(A) \subset \mathbb{R}^-$. Then there exist $C > 0$ and $\rho_1 > 1$ such that

$$\|f_j(A) - p_{j,n}(A)/q_n(A)\|_2 = \sup_{x \leq 0} |f_j(x) - p_{j,n}(x)/q_n(x)| \leq C \rho_1^{-n}.$$

If A is nonnormal and the field of values $W(A) \subset \mathbb{C} \setminus \mathbb{R}^-$, Crouzeix's theorem [28] yields

$$\|f_j(A) - p_{j,n}(A)/q_n(A)\| \leq C' \rho_2^{-n},$$

for some $C' > 0$ and $\rho_2 > 1$.

Proof. First, suppose A is normal and $\sigma(A) \subset \mathbb{R}^-$. By the spectral theorem, there is a unitary U and a real diagonal Λ with entries in $\mathbb{R}^-$ such that $A = U\Lambda U^*$. For any rational r without poles on $\mathbb{R}^-$ and since the 2-norm is unitarily invariant, we have

$$\begin{aligned} \|f_j(A) - r(A)\|_2 &= \|U(f_j(\Lambda) - r(\Lambda))U^*\|_2 \\ &= \|f_j(\Lambda) - r(\Lambda)\|_2 = \max_{\lambda \in \sigma(A)} |f_j(\lambda) - r(\lambda)|. \end{aligned}$$

Apply Theorem 1 with a contour Γ_σ enclosing $\mathbb{R}^-$; the same construction provides a uniform bound on any compact subset inside Γ_σ , hence on $\sigma(A) \subset \mathbb{R}^-$:

$$\|f_j(A) - p_{j,n}(A)/q_n(A)\|_2 \leq \sup_{x \leq 0} |f_j(x) - p_{j,n}(x)/q_n(x)| \leq C \rho_1^{-n},$$

for some $C > 0$ and $\rho_1 > 1$.

If A is nonnormal and $W(A) \subset \mathbb{C} \setminus \mathbb{R}^-$, pick $\sigma > 1$ so that the level curve Γ_σ encloses the compact set $W(A)$ at positive distance. By the same Cauchy-trapezoidal construction, the rational $r_n = p_{j,n}/q_n$ satisfies

$$\sup_{z \in W(A)} |f_j(z) - r_n(z)| \leq \widehat{C} \rho_2^{-n},$$

for some $\widehat{C} > 0$ and $\rho_2 > 1$. Therefore, we have

$$\|f_j(A) - r_n(A)\| \leq C_{\text{Crx}} \sup_{z \in W(A)} |f_j(z) - r_n(z)| \leq C' \rho_2^{-n}$$

by Crouzeix's theorem with the universal constant $C_{\text{Crx}} \leq 1 + \sqrt{2}$. $\square$

In view of classical potential theoretic results on best type- (n, n) rational approximation of e^x on $\mathbb{R}^-$, the geometric rate ρ in Theorem 1 can be identified with the reciprocal of Halphen's constant $H \approx 1/9.28903$, i.e., $\rho = 1/H \approx 9.28903$; see ([20], Section 4). Motivated by this prediction, in the numerical sweep of Section 5.1 we vary the CF scale parameter $\sigma \in \{5, 7, 9, 11, 13\}$ and measure the familywise worst error E_n and fitted rates ρ^* . As reported in Table 1, for all $p = 0, \dots, 4$ the fitted rates lie between 9 and 10 and $\sigma = 9$ consistently yields both the largest ρ^* and the smallest worst error E_{14} . Thus, $\sigma = 9$ is a robust, near-optimal choice for the CF scale parameter, and we adopt this fixed default in Algorithm 1.

Table 1. Fitted rate ρ_* and familywise worst error E_{14} on $[-M, 0]$ with $M = 100\sigma$, $5 \leq \sigma \leq 13$.

p	ρ_*	E_{14}
0	9.663	4.11×10^{-14}
1	9.581	6.60×10^{-14}
2	9.691	7.15×10^{-14}
3	9.959	4.54×10^{-14}
4	9.788	5.51×10^{-14}

4. Evaluating the Linear Combination of the φ -Functions with Shared Poles

The functions of interest $\{\varphi_j\}_{j=0}^p$ are evaluated on the slit plane $\mathbb{C} \setminus \mathbb{R}^-$, with $\mathbb{R}^-$ playing the role of a branch cut, and they satisfy the decay property $\varphi_j(z) \rightarrow 0$ as $|z| \rightarrow \infty$ in a sector containing the negative real axis. A shared denominator captures the common structure across the family. If r_j denotes a CF rational approximant to φ_j as in (3), then for matrix arguments with spectra lying in the negative real axis we have

$$r_j(A) = r_\infty^{(j)} I + \sum_{\ell=1}^n \eta_\ell^{(j)} (A - \theta_\ell I)^{-1}.$$

Consequently, the linear combination (2) is approximated by

$$\begin{aligned} \sum_{j=0}^p \varphi_j(A) v_j &\approx \sum_{j=0}^p \left(r_\infty^{(j)} v_j + \sum_{\ell=1}^n \eta_\ell^{(j)} (A - \theta_\ell I)^{-1} v_j \right) \\ &= \sum_{j=0}^p r_\infty^{(j)} v_j + \sum_{\ell=1}^n (A - \theta_\ell I)^{-1} v_{\text{rhs}}^{(\ell)}, \end{aligned}$$

where

$$v_{\text{rhs}}^{(\ell)} := \sum_{j=0}^p \eta_\ell^{(j)} v_j, \quad \ell = 1, \dots, n.$$

Thus, we solve only n shifted linear systems—one per pole—so the cost scales with the pole count n , rather than with $(p+1)n$ as it would if each r_j had a different pole set.

A substantial computational saving is possible when the matrix A and the vectors $\{v_j\}_{j=0}^p$ are real, which is the most common situation in practice. Because each φ_j is real-valued on $\mathbb{R}^-$, the CF approximants r_j can be chosen with real coefficients, so all non-real poles and residues occur in complex conjugate pairs (apart from any real poles). In this case, we order the poles so that for some even degree n we have

$$\theta_{n+1-\ell} = \overline{\theta_\ell}, \quad \ell = 1, \dots, \frac{n}{2},$$

and we reorder the residues accordingly:

$$\eta_{n+1-\ell}^{(j)} = \overline{\eta_{\ell}^{(j)}}, \quad j = 0, \dots, p.$$

Since A and the v_j are real, this implies

$$v_{\text{rhs}}^{(n+1-\ell)} = \sum_{j=0}^p \eta_{n+1-\ell}^{(j)} v_j = \overline{\sum_{j=0}^p \eta_{\ell}^{(j)} v_j} = \overline{v_{\text{rhs}}^{(\ell)}}, \quad \ell = 1, \dots, \frac{n}{2}.$$

Hence, the corresponding solutions satisfy

$$x^{(n+1-\ell)} := (A - \theta_{n+1-\ell} I)^{-1} v_{\text{rhs}}^{(n+1-\ell)} = \overline{(A - \theta_{\ell} I)^{-1} v_{\text{rhs}}^{(\ell)}} = \overline{x^{(\ell)}},$$

and, therefore,

$$\sum_{\ell=1}^n (A - \theta_{\ell} I)^{-1} v_{\text{rhs}}^{(\ell)} = 2 \operatorname{Re} \left(\sum_{\ell=1}^{n/2} (A - \theta_{\ell} I)^{-1} v_{\text{rhs}}^{(\ell)} \right).$$

In the real case, it thus suffices to solve only $n/2$ shifted linear systems.

We are now in a position to present our algorithm for computing linear combinations of several φ -function actions; it is given in Algorithm 2. It is worth noticing that the m shifted linear systems in line 11 of Algorithm 2 need not be solved sequentially. Since each system corresponds to a distinct pole and has an independent right-hand side, they can be solved in parallel.

Algorithm 2 Linear combination of φ -functions with shared poles.

Input:

- ◇ $A \in \mathbb{C}^{d \times d}$ and vectors $v_1, \dots, v_{\tau} \in \mathbb{C}^d$
- ◇ Degree n (default $n = 14$) of the rational approximants r_j (for real data, n is even so that poles occur in conjugate pairs)
- ◇ A family $\{\varphi_{j_k}\}_{k=1}^{\tau}$, where $j_k \in \mathcal{I} \subset \mathbb{N} \cup \{0\}$ and $|\mathcal{I}| = \tau$
- ◇ Shared CF data: poles $\{\theta_{\ell}\}_{\ell=1}^n$, residues $\eta_{\ell}^{(j_k)}$, and constants $r_{\infty}^{(j_k)}$ constructed via Algorithm 1 for the scalar functions $\varphi_{j_1}, \dots, \varphi_{j_{\tau}}$

Output: $w \approx \sum_{k=1}^{\tau} \varphi_{j_k}(A) v_k$.

- 1: Order $\{\theta_{\ell}\}_{\ell=1}^n$ so that complex-conjugate pairs $(\theta_{\ell}, \theta_{n+1-\ell})$ are adjacent, and reorder the corresponding $\eta_{\ell}^{(j_k)}$ accordingly.
 - 2: $m = n$
 - 3: **if** realData **then**
 - 4: $m \leftarrow n/2$
 - 5: **end if**
 - 6: $V = [v_1, v_2, \dots, v_{\tau}]$ ▷ $d \times \tau$ matrix
 - 7: $w = 0$ ▷ $d \times 1$ zero vector
 - 8: **for** $\ell = 1$ to m **do**
 - 9: $\gamma^{(\ell)} \leftarrow [\eta_{\ell}^{(j_1)}, \eta_{\ell}^{(j_2)}, \dots, \eta_{\ell}^{(j_{\tau})}]^T$
 - 10: $v_{\text{rhs}}^{(\ell)} \leftarrow V \gamma^{(\ell)}$ ▷ one combined right-hand side
 - 11: Solve $(A - \theta_{\ell} I) x^{(\ell)} = v_{\text{rhs}}^{(\ell)}$
 - 12: **if** realData **then**
 - 13: $x^{(\ell)} \leftarrow 2 \operatorname{Re}(x^{(\ell)})$
 - 14: **end if**
 - 15: $w \leftarrow w + x^{(\ell)}$
 - 16: **end for**
 - 17: $w \leftarrow w + V [r_{\infty}^{(j_1)}, r_{\infty}^{(j_2)}, \dots, r_{\infty}^{(j_{\tau})}]^T$
-

5. Numerical Experiments

This section presents three numerical experiments. The first investigates the algorithmic parameters and the geometric convergence rate of the CF approximation while the second experiment implements Algorithm 2 for a highly nonnormal matrix. The third experiment involves the 2D Poisson matrix.

All runs were performed in MATLAB R2022b on a single desktop (Intel® Core™ i7–7700T @ 2.90 GHz, 16 GB RAM, Intel Corporation, Santa Clara, CA, USA). To contextualize performance and accuracy, we compare the following five routines:

- cfphimv: our MATLAB routine for Algorithm 2 (<https://github.com/aalmohy/cfphimv> (accessed on 9 December 2025)).
- phimv: Al-Mohy’s algorithm ([3], Algorithm 2) (<https://github.com/aalmohy/phimv> (accessed on 20 October 2025)).
- phi_funm: Al-Mohy and Liu’s algorithm ([18], Algorithm 5.1) (https://github.com/xiaoboliu/phi_funm (accessed on 20 October 2025)). This algorithm evaluates several φ -functions of a moderate size matrix jointly via a scaling and recovering strategy. We use this routine to compute a reference solution for medium-sized problems.
- bamphi: Caliari, Cassini, and Živković’s routine [14], combining Newton form polynomial interpolation at special nodes with Krylov techniques (<https://github.com/francozivicovich/bamphi> (accessed on 20 October 2025)).
- kiops: The adaptive Krylov solver of Gaudreault, Rainwater, and Tokman [5] based on the incomplete orthogonalization procedure (<https://gitlab.com/stephane.gaudreault/kiops> (accessed on 20 October 2025)).

All `phimv`, `bamphi`, and `kiops` natively accept block right-hand sides and, in one call, evaluate several linear combinations of the form (2). We use each routine with its default settings; for `kiops`, we set the tolerance to the double-precision machine epsilon for consistency with the other routines.

5.1. Numerical Sweep for Shared-Denominator CF Tables

The purpose of this experiment is to (i) empirically verify geometric convergence of the shared denominator CF approximants for the family $\{\varphi_j\}_{j=0}^p$ on $\mathbb{R}^-$, and (ii) identify practical defaults for the CF scale σ and degree n that deliver double precision accuracy uniformly across j .

For $p \in \{0, 1, 2, 3, 4\}$ we build type- (n, n) shared-pole tables at $n \in \{6, 8, 10, 12, 14\}$ and $\sigma \in \{5, 7, 9, 11, 13\}$. Pole extraction via Algorithm 1 uses $K = 100$ Chebyshev coefficients and $N = 1024$ Chebyshev points. With the shared poles $\{\theta_\ell\}$ fixed, per-function residues and constants are fitted by least squares (9) on a log-dense training grid of size $\tau = 8000$ over $[-M, 0]$ with $M = 100\sigma$. Accuracy is measured on an independent testing grid of size $\tau = 20,000$ over $[-M, 0]$. For each configuration, we compute the familywise worst error

$$E_n = \max_{0 \leq j \leq p} \max_{x \in [-M, 0]} |\varphi_j(x) - r_j(x)|.$$

We report the geometric rate as ρ_* , defined by a least squares fit of $\log E_n \approx a + bn$ over the interior degrees $\{8, 10, 12\}$ to reduce endpoint bias, yielding $\rho_* = e^{-b}$ and $C = e^a$. Across all p , the fitted rates lie near the Halphen’s constant and the worst errors at $n = 14$ are between 10^{-14} and 10^{-13} . Similar behavior holds for $\sigma \in \{5, 7, 11, 13\}$, with $\sigma = 9$ consistently optimal by rate and end error.

We conclude that choosing $\sigma \approx 9$ is sufficient to achieve a nearly optimal geometric convergence rate across $p = 0, \dots, 4$. This default aligns with the scales reported in [20,25]. Accordingly, we recommend $\sigma = 9$ as the CF scale parameter in Algorithm 1.

5.2. Chebyshev Spectral Laplacian (Dirichlet)

We use a Chebyshev collocation discretization on the interval $[0, L]$ with homogeneous Dirichlet boundary conditions ([3], Section 5). Starting from the Chebyshev first derivative matrix D on $\xi \in [-1, 1]$, the change in variables $x = \frac{L}{2}(\xi + 1)$ implies $\frac{d}{dx} = \frac{2}{L} \frac{d}{d\xi}$, so the discrete second derivative on $[0, L]$ is $(2/L)^2 D^2$. Enforcing Dirichlet conditions by deleting the first and last rows/columns yields a dense, strongly nonnormal matrix $A \in \mathbb{R}^{(N-1) \times (N-1)}$ whose spectrum lies on the negative real axis and whose spectral radius grows with N , making it a stiff, representative test for exponential integrators. The code in Table 2 (adapted from [29]) constructs A for general N and L .

Table 2. MATLAB code for the discrete second derivative on $[0, L]$ with Dirichlet conditions.

```
k = (0:N)'; % node indices
xi = cos(pi*k/N); % Chebyshev points on [-1,1]
x = (xi + 1)*(L/2); % map to [0,L]
c = [2; ones(N-1,1); 2].*(-1).^k; % barycentric weights
X = repmat(x,1,N+1);
dX = X - X.'; % pairwise differences
D = (c*(1./c)')./(dX + eye(N+1)); % first-derivative matrix on [-1,1]
D = D - diag(sum(D,2)); % enforce exact column sums = 0
Lh = (2/L)^2 * (D*D); % scaled second derivative on [0,L]
A = Lh(2:N, 2:N); % remove boundary rows/cols (Dirichlet)
```

We consider $N \in \{50, 100, 150, 200, 250\}$ and $L = 2$. The aim is to demonstrate the speed of our shared denominator CF rational approximant (with $n = 14$) relative to Taylor- and Krylov polynomial-based routines. As $\|A\|$ and nonnormality grow, polynomial approaches typically require higher degrees (or smaller steps) to control the error, so their cost escalates. By contrast, the CF rational approximants employ a fixed set of poles that capture the branch cut of the φ -family, making accuracy essentially insensitive to the spectral radius of A ; the dominant work reduces to a small number of shifted linear solves. This is therefore a highly stiff test where rational approximants should retain both accuracy and speed as N increases. For $p = 3$ and each N , we generate a random set of vectors $\{v_0, v_1, \dots, v_p\} \in \mathbb{R}^{N-1}$ and evaluate $w_N = \sum_{j=0}^p \varphi_j(A)v_j$ using `cfphimv`, `phimv`, `bamphi`, and `kiops`. The reference solution is computed using `phi_funm`.

The data in Table 3 highlight three key points. (i) The cost of `cfphimv` is essentially flat in N : even as $\|A\|_2$ increases from 3.0×10^5 to 1.9×10^8 and the matrix becomes more nonnormal, the runtime stays below 0.5 s and the relative error remains in the 10^{-11} – 10^{-10} range. (ii) Krylov/Taylor-based methods (`phimv`, `bamphi`, `kiops`) degrade rapidly with N : `phimv` and `bamphi` become slower by one to two orders of magnitude and eventually time out, and `kiops` either times out or returns unusably large errors (up to 10^8). (iii) Beyond $N \simeq 100$, only `cfphimv` continues to deliver both accuracy and subsecond turnaround. This supports the main claim of the paper: once the shared pole set is precomputed, evaluating the full linear combination of φ -functions reduces to solving a small number of shifted systems, and this remains stable and fast even for highly stiff, strongly nonnormal matrices.

Table 3. Runtimes and relative errors for the linear combination $w_N = \sum_{j=0}^p \varphi_j(A)v_j$ with a hard per-call timeout of 300 s. “TO” = timed out; “ER” = routine error.

N	$\ A\ _2$	phimv		bamphi		kiops		cfphimv	
		time (s)	rel. err	time (s)	rel. err	time (s)	rel. err	time (s)	rel. err
50	3.0×10^5	3.300	2.4×10^{-13}	3.108	1.6×10^{-11}	2.066	9.0×10^5	0.076	3.7×10^{-12}
100	4.8×10^6	35.202	1.3×10^{-11}	126.170	1.3×10^{-10}	32.038	2.7×10^8	0.073	1.4×10^{-11}
150	2.4×10^7	TO	—	TO	—	TO	—	0.462	1.9×10^{-11}
200	7.7×10^7	TO	—	TO	—	TO	—	0.452	1.7×10^{-11}
250	1.9×10^8	TO	—	ER	—	TO	—	0.450	2.7×10^{-10}

5.3. Two-Dimensional Poisson Matrix

In this experiment we use the two-dimensional Poisson matrix $P \in \mathbb{R}^{N^2 \times N^2}$ obtained from the standard five-point finite difference discretization of $-\Delta u = f$ on $(0,1)^2$ with homogeneous Dirichlet boundary conditions. With lexicographic ordering of the N^2 interior grid points, P can be written as a sum of Kronecker products

$$P = I_N \otimes T + T \otimes I_N,$$

where $T \in \mathbb{R}^{N \times N}$ is the tridiagonal Toeplitz matrix $\text{tridiag}(-1, 2, -1)$ corresponding to the one-dimensional second-difference operator. The matrix A can be generated by the MATLAB command `P = gallery('poisson', N)`. Poisson matrices of this type arise ubiquitously in the finite difference and finite element discretization of diffusion and heat equations, electrostatics and potential problems, pressure Poisson equations in incompressible flow, and in image processing and graph-based models where discrete Laplacians are used for smoothing and regularization.

The spectral structure of P can be characterized explicitly. The one-dimensional matrix T is diagonalized by the discrete sine transform matrix S with entries

$$S_{jk} = \sqrt{\frac{2}{N+1}} \sin\left(\frac{jk\pi}{N+1}\right), \quad j, k = 1, \dots, N,$$

yielding $T = SAS^T$ with eigenvalues

$$\lambda_k = 2 - 2\cos\left(\frac{k\pi}{N+1}\right) = 4\sin^2\left(\frac{k\pi}{2(N+1)}\right), \quad k = 1, \dots, N.$$

By standard properties of Kronecker products,

$$P = (S \otimes S) \Lambda_{2D} (S \otimes S)^T, \quad \Lambda_{2D} = \text{diag}(\lambda_{k,\ell}), \quad \lambda_{k,\ell} = \lambda_k + \lambda_\ell, \quad k, \ell = 1, \dots, N,$$

so the eigenvectors of P are tensor products $s_k \otimes s_\ell$ of the one-dimensional sine modes. Detailed expositions of this construction and its use in fast Poisson solvers and spectral analysis can be found; for example, in Strang ([27], Section 5.5) and Golub and Van Loan ([30], Section 4.8).

To evaluate $f(P)b$ efficiently for a scalar matrix function f analytic on the spectrum of P and a vector $b \in \mathbb{C}^{N^2}$, we exploit the explicit diagonalization

$$P = (S \otimes S) \Lambda_{2D} (S \otimes S)^T, \quad \Lambda_{2D} = \text{diag}(\lambda_{k,\ell}), \quad \lambda_{k,\ell} = \lambda_k + \lambda_\ell.$$

We first reshape b into an $N \times N$ matrix B such that $\text{vec}(B) = b$, where the vec operator stacks the columns of a matrix on top of each other. Using the identity

$$(S \otimes S)^T \text{vec}(B) = \text{vec}(S^T B S),$$

we compute the spectral coefficients $\widehat{B} = S^T B S$. Defining

$$F_{k,\ell} = f(\lambda_{k,\ell}), \quad k, \ell = 1, \dots, N,$$

we apply $f(\Lambda_{2D})$ by pointwise multiplication,

$$Z_{k,\ell} = F_{k,\ell} \widehat{B}_{k,\ell}, \quad k, \ell = 1, \dots, N,$$

and transform back via $B_f = S Z S^T$, so that

$$f(P)b = \text{vec}(B_f).$$

In this procedure, we never form $S \otimes S$ or $f(P)$ explicitly; the dominant cost is a small number of $N \times N$ matrix–matrix products (or fast sine transforms), i.e., $\mathcal{O}(N^3)$ flops with an explicit S or $\mathcal{O}(N^2 \log N)$ flops with FFT-based discrete sine transforms ([30], Section 4.8). We use this spectral machinery, combined with multiprecision arithmetic using the Multiprecision Computing Toolbox (ver. 5.1.0) [31], to generate highly accurate reference solutions.

We now use this eigenvalue decomposition as a high-accuracy reference to evaluate the linear combination (2). Let $h = 1/(N+1)$ be the mesh size and $N = 2^k$ for $k = 6, \dots, 10$. We define the scaled operator

$$A = -\frac{1}{h^2} P.$$

Thus, A is negative definite and increasingly stiff as N grows (indeed, $\|A\|_1 = \mathcal{O}(N^2)$). For each N and four ($p = 3$) randomly generated vectors $\{v_j\}_{j=0}^p \subset \mathbb{R}^{N^2}$, we evaluate the linear combination

$$w_N = \sum_{j=0}^p \varphi_j(A) v_j$$

using the routines `cfphimv`, `phimv`, `kiops`, and `bamphi`. The reference solution is computed via the spectral diagonalization of A described above, with $\varphi_j(-\lambda_{k,\ell}/h^2)$ applied to the eigenvalues in multiprecision arithmetic, and the relative error is measured in the 1-norm against this reference.

Table 4 presents the results. For the smallest sizes ($N = 64, 128$), all four algorithms reach very small relative errors, typically between 10^{-14} and 10^{-11} , confirming that they all resolve the highly stiff operator A on this model problem. However, the runtimes differ markedly: `cfphimv` is already more than an order of magnitude faster than `phimv` and substantially faster than `kiops` and `bamphi`. As N increases, the stiffness and dimension grow rapidly ($\|A\|_1$ increases by roughly two orders of magnitude over the tested range). For $N = 256$, `phimv` already hits the time limit of 600 s, while `kiops` and `bamphi` remain accurate but require tens of seconds. For $N \geq 512$, all three polynomial- or Krylov-based routines time out, whereas `cfphimv` continues to deliver accurate results with runtimes ranging from a fraction of a second (for $N = 64$) to about 80 s (for $N = 1024$). The modest growth of the `cfphimv` error with N (remaining around 10^{-9} at $N = 1024$) indicates that the rational approximation error and the effect of the reference solver are both well under control. Overall, this experiment shows that the shared denominator CF approach can handle very stiff, large-scale diffusion operators for linear combinations of φ -functions at a cost comparable to a small number of shifted linear solves, while competing Taylor and rational Krylov methods become prohibitively slow or fail to complete within the time limit.

Table 4. Runtimes and relative errors for the linear combination $w_N = \sum_{j=0}^p \varphi_j(A)v_j$ with A the scaled 2D Poisson matrix and a hard per-call timeout of 600 s. “TO” = timed out.

N	$\ A\ _1$	phimv		bamphi		kiops		cfphimv	
		time (s)	rel. err	time (s)	rel. err	time (s)	rel. err	time (s)	rel. err
64	3.4×10^4	8.340	1.0×10^{-14}	0.770	3.0×10^{-13}	0.450	5.2×10^{-15}	0.180	3.9×10^{-11}
128	1.3×10^5	104.470	2.3×10^{-14}	5.490	5.2×10^{-12}	3.280	1.7×10^{-14}	0.540	5.2×10^{-11}
256	5.3×10^5	TO	—	68.650	1.7×10^{-11}	47.950	9.6×10^{-14}	2.320	1.2×10^{-10}
512	2.1×10^6	TO	—	TO	—	TO	—	13.410	8.1×10^{-10}
1024	8.4×10^6	TO	—	TO	—	TO	—	81.180	1.2×10^{-9}

6. Conclusions

We presented a shared denominator CF framework for evaluating linear combinations of φ -functions with stable matrices, a core task in exponential integrators. The key idea is to approximate $\{\varphi_j\}_{j=0}^p$ on $\mathbb{R}^-$ with rational functions that share a single pole set $\{\theta_\ell\}_{\ell=1}^n$, while allowing per-function residues $\{\eta_\ell^{(j)}\}$ and constants $\{r_\infty^{(j)}\}$. We obtain the poles by a single SVD of a stacked weighted Hankel built from Chebyshev boundary data of all functions, and then recover the residues and constants via a robust least squares fit on a log-dense grid of $\mathbb{R}^-$. With these ingredients fixed, any linear combination $w = \sum_{j=0}^p \varphi_j(A)v_j$ is reduced to solving only $n/2$ (for real data and even n) shifted linear systems *independent of* p —one per shared pole—against a single combined right-hand side per shift as described in Algorithm 2. Thus, the dominant cost matches that of evaluating a single φ -function action.

On the theoretical side, we proved a “no assumptions” shared denominator approximation theorem on $\mathbb{R}^-$ that yields a geometric rate $C\rho^{-n}$ for a finite family of analytic functions and their linear combinations, and we lifted these bounds to matrix arguments for normal matrices (via the spectral theorem) and for nonnormal matrices whose field of values avoids the cut (via Crouzeix’s theorem). This places our construction on the same exponential convergence footing as CF/contour-trapezoid approaches, with the observed rates closely tied to the classical constants for slit domains.

Our numerical sweeps corroborate the theory: for $p = 0, \dots, 4$ and moderate degrees $n \leq 14$, the worst case scalar errors across $\{\varphi_j\}$ decay geometrically to $\sim 10^{-14}$, with a near-uniformly effective CF scale around $\sigma \approx 9$. The experiments also clarify the distinct roles of the parameters used in practice. Importantly, because the rational tables are constructed on the continuous negative real axis and then applied to A via shifted solves, their effectiveness is largely insensitive to the spectral radius of A , in contrast to Taylor and polynomial Krylov approaches whose difficulty grows directly with $\|A\|$.

Because our evaluation reduces to solves with the shifted operators $(A - \theta_\ell I)$ against a single combined right-hand side per pole, the method is immediately amenable to matrix-free implementations.

Funding: This research was funded by the Deanship of Scientific Research at King Khalid University through the Research Groups Program under Grant No. RGP.1/318/45.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: We thank the reviewers for their insightful comments and suggestions that helped to improve the presentation of this paper.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Hochbruck, M.; Ostermann, A. Exponential integrators. *Acta Numer.* **2010**, *19*, 209–286. [\[CrossRef\]](#)

2. Minchev, B.V.; Wright, W.M. *A Review of Exponential Integrators for First Order Semi-Linear Problems*; Technical Report 2/05; Norwegian University of Science and Technology: Trondheim, Norway, 2005.
3. Al-Mohy, A.H. Computing Linear Combinations of φ -Function Actions for Exponential Integrators. *arXiv* **2025**, arXiv:2509.26475.
4. Al-Mohy, A.H.; Higham, N.J. Computing the Action of the Matrix Exponential, with an Application to Exponential Integrators. *SIAM J. Sci. Comput.* **2011**, *33*, 488–511. [[CrossRef](#)]
5. Gaudreault, S.; Rainwater, G.; Tokman, M. KIOPS: A Fast Adaptive Krylov Subspace Solver for Exponential Integrators. *J. Comput. Phys.* **2018**, *372*, 236–255. [[CrossRef](#)]
6. Hochbruck, M.; Lubich, C.; Selhofer, H. Exponential Integrators for Large Systems of Differential Equations. *SIAM J. Sci. Comput.* **1998**, *19*, 1552–1574. [[CrossRef](#)]
7. Koskela, A.; Ostermann, A. Exponential Taylor Methods: Analysis and Implementation. *Comput. Math. Appl.* **2013**, *65*, 487–499. [[CrossRef](#)]
8. Niesen, J.; Wright, W.M. Algorithm 919: A Krylov Subspace Algorithm for Evaluating the φ -Functions Appearing in Exponential Integrators. *ACM Trans. Math. Softw.* **2012**, *38*, 22. [[CrossRef](#)]
9. Cox, S.; Matthews, P. Exponential Time Differencing for Stiff Systems. *J. Comput. Phys.* **2002**, *176*, 430–455. [[CrossRef](#)]
10. Luan, V.T. Efficient Exponential Runge–Kutta Methods of High Order: Construction and Implementation. *BIT Numer. Math.* **2021**, *61*, 535–560. [[CrossRef](#)]
11. Kassam, A.K.; Trefethen, L.N. Fourth-Order Time-Stepping for Stiff PDEs. *SIAM J. Sci. Comput.* **2005**, *26*, 1214–1233. [[CrossRef](#)]
12. Higham, N.J. *Functions of Matrices: Theory and Computation*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2008; p. xx+425. [[CrossRef](#)]
13. Luan, V.T.; Pudykiewicz, J.A.; Reynolds, D.R. Further Development of Efficient and Accurate Time Integration Schemes for Meteorological Models. *J. Comput. Phys.* **2019**, *376*, 817–837. [[CrossRef](#)]
14. Caliari, M.; Cassini, F.; Zivcovich, F. BAMPHI: Matrix-free and Transpose-free Action of Linear Combinations of φ -Functions from Exponential Integrators. *J. Comput. Appl. Math.* **2023**, *423*, 114973. [[CrossRef](#)]
15. Moret, I. On RD-rational Krylov approximations to the core-functions of exponential integrators. *Numer. Linear Algebra Appl.* **2007**, *14*, 445–457. [[CrossRef](#)]
16. Bergermann, K.; Stoll, M. Adaptive Rational Krylov Methods for Exponential Runge–Kutta Integrators. *SIAM J. Matrix Anal. Appl.* **2024**, *45*, 744–770. [[CrossRef](#)]
17. Berland, H.; Skaflestad, B.; Wright, W.M. EXPINT—A MATLAB Package for Exponential Integrators. *ACM Trans. Math. Softw.* **2007**, *33*, 4-es. [[CrossRef](#)]
18. Al-Mohy, A.H.; Liu, X. A Scaling and Recovering Algorithm for the Matrix φ -Functions. *arXiv* **2025**, arXiv:2506.01193.
19. Al-Mohy, A.H.; Higham, N.J. A New Scaling and Squaring Algorithm for the Matrix Exponential. *SIAM J. Matrix Anal. Appl.* **2009**, *31*, 970–989. [[CrossRef](#)]
20. Trefethen, L.N.; Weideman, J.A.C.; Schmelzer, T. Talbot quadratures and rational approximations. *BIT Numer. Math.* **2006**, *46*, 653–670. [[CrossRef](#)]
21. Trefethen, L.N. Near-circularity of the error curve in complex Chebyshev approximation. *J. Approx. Theory* **1981**, *31*, 344–367. [[CrossRef](#)]
22. Trefethen, L.N. Rational Chebyshev approximation on the unit disk. *Numer. Math.* **1981**, *37*, 297–320. [[CrossRef](#)]
23. Trefethen, L.N.; Gutknecht, M.H. The Carathéodory–Fejér Method for Real Rational Approximation. *SIAM J. Numer. Anal.* **1983**, *20*, 420–436. [[CrossRef](#)]
24. Carathéodory, C.; Fejér, L. Über den Zusammenhang der Extremen von harmonischen Funktionen mit ihren Koeffizienten und über den Picard–Landau’schen Satz. *Rend. Circ. Mat. Palermo* **1911**, *32*, 218–239. [[CrossRef](#)]
25. Schmelzer, T.; Trefethen, L.N. Evaluating Matrix Functions for Exponential Integrators via Carathéodory–Fejér Approximation and Contour Integrals. *Electron. Trans. Numer. Anal.* **2007**, *29*, 1–18.
26. Trefethen, L.N.; Weideman, J.A.C. The Exponentially Convergent Trapezoidal Rule. *SIAM Rev.* **2014**, *56*, 385–458. [[CrossRef](#)]
27. Strang, G. *Introduction to Applied Mathematics*; Wellesley–Cambridge Press: Wellesley, MA, USA, 1986.
28. Crouzeix, M.; Palencia, C. The Numerical Range is a $(1 + \sqrt{2})$ -Spectral Set. *SIAM J. Matrix Anal. Appl.* **2017**, *38*, 649–655. [[CrossRef](#)]
29. Trefethen, L.N. *Spectral Methods in MATLAB*; SIAM: Philadelphia, PA, USA, 2000. [[CrossRef](#)]
30. Golub, G.H.; Van Loan, C.F. *Matrix Computations*; Johns Hopkins University Press: Baltimore, MD, USA, 2013.
31. Advanpix. *Multiprecision Computing Toolbox*; Advanpix: Tokyo, Japan, 2025. Available online: <http://www.advanpix.com> (accessed on 15 August 2023).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.