

## SPECTRAL APPROXIMATION OF CONVOLUTION OPERATORS OF FREDHOLM TYPE\*

XIAOLIN LIU<sup>†</sup>, KUAN DENG<sup>†</sup>, AND KUAN XU<sup>†</sup>

**Abstract.** We have developed a method for constructing spectral approximations for convolution operators of Fredholm type. The algorithm we propose is numerically stable and takes advantage of the recurrence relations satisfied by the entries of such a matrix approximation. When used for computing the Fredholm convolution of two given functions, such approximations produce the convolution more rapidly than the state-of-the-art methods. The proposed approximation also leads to a spectral method for solving the Fredholm convolution integral equations and makes the operator analogue of the Krylov subspace methods efficient for computing the eigenvalues and pseudospectra of Fredholm convolution operators.

**Key words.** convolution, Fredholm convolution integral, operator approximation, Legendre polynomial, spectral method, pseudospectra

**MSC codes.** 44A35, 45E10, 45B05, 65R10, 47A58, 33C45, 65Q30

**DOI.** 10.1137/24M1652532



See reproducibility of  
computational results  
at end of the article.

**1. Introduction.** Convolution is a fundamental operation which abounds in science and engineering. The convolution of two smooth signals or functions typically results in a piecewise smooth function, which consists of three pieces. The two end pieces are known as Volterra convolution integral because each has a variable integration limit, while the middle piece, which has constant limits, is a Fredholm convolution integral. See [5, 12] for detailed discussion and illustrative diagrams. In this paper, we are concerned with the construction of approximations to the convolution operators of Fredholm type and the applications of such approximations, including the calculation of Fredholm convolution integrals and the spectral method for solving Fredholm convolution integral equations. This paper can be considered a companion work to [13], which addresses the spectral approximation of the convolution operators of Volterra type.

Let  $f(x) : [a, b] \rightarrow \mathbb{R}$  and  $g(x) : [c, d] \rightarrow \mathbb{R}$  be two continuous integrable compactly supported functions with  $b - a > d - c$ . The Fredholm convolution of  $f(x)$  and  $g(x)$  is

$$h(x) = \int_c^d f(x - t)g(t)dt, \quad x \in [a + d, b + c].$$

Since it is convenient to discuss in terms of functions defined on canonical intervals centered at the origin, we define the following mapped versions of  $f$  and  $g$ :

$$\begin{aligned} \tilde{f}(y) &= f\left(\frac{d-c}{2}y + \frac{a+b}{2}\right), \quad y \in [-(r+1), r+1], \\ \tilde{g}(y) &= g\left(\frac{d-c}{2}y + \frac{d+c}{2}\right), \quad y \in [-1, 1], \end{aligned}$$

\*Submitted to the journal's Numerical Algorithms for Scientific Computing section April 22, 2024; accepted for publication (in revised form) February 20, 2025; published electronically June 11, 2025.  
<https://doi.org/10.1137/24M1652532>

<sup>†</sup>School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, Anhui, China (xiaolin@mail.ustc.edu.cn, dengkuan@mail.ustc.edu.cn, kuanxu@ustc.edu.cn).

where  $r = (b - a)/(d - c) - 1 > 0$ . Thus, the convolution of  $\tilde{f}(y)$  and  $\tilde{g}(y)$  is

$$(1.1) \quad \tilde{h}(y) = \int_{-1}^1 \tilde{f}(y - t)\tilde{g}(t)dt, \quad y \in [-r, r],$$

which is related to  $h(x)$  by

$$h(x) = \frac{d - c}{2} \tilde{h} \left( \frac{2x - (a + b + c + d)}{d - c} \right), \quad x \in [a + d, b + c].$$

Thus, in the remainder of this paper we consider only the convolution  $\tilde{h}$  given by (1.1) with  $\tilde{f}$  and  $\tilde{g}$  defined on  $[-(r + 1), r + 1]$  and  $[-1, 1]$ . To further facilitate the discussion, we slightly abuse the notations by dropping the tildes in  $\tilde{f}$ ,  $\tilde{g}$ , and  $\tilde{h}$ , i.e.,  $f(x) : [-(r + 1), r + 1] \rightarrow \mathbb{R}$ ,  $g(x) : [-1, 1] \rightarrow \mathbb{R}$ , and  $h(x) : [-r, r] \rightarrow \mathbb{R}$ . Hence, we consider the convolution operator  $\mathcal{F}$  of Fredholm type that is defined by  $f(x)$  and acts on  $g(x)$ ,

$$\mathcal{F}_f[g](x) = h(x) = \int_{-1}^1 f(x - t)g(t)dt, \quad x \in [-r, r].$$

Here,  $f(x)$  is usually called the kernel function or, simply, the kernel. Note that unlike in the convolution of Volterra type, the kernel function  $f(x)$  and the function to be convolved, i.e.,  $g(x)$ , do not commute, i.e.,  $\mathcal{F}_f[g](x) \neq \mathcal{F}_g[f](x)$ . As we shall see, the value  $r$  plays a pivotal role throughout our discussion.

If  $f(x)$  and  $g(x)$  have certain regularity, they can be approximated by polynomials  $f_M(x)$  and  $g_N(x)$  of sufficiently high degree so that  $\|f(x) - f_M(x)\|_\infty$  and  $\|g(x) - g_N(x)\|_\infty$  are on the order of machine precision. Such polynomial approximations can be constructed as, e.g., Legendre series

$$(1.2) \quad f_M(x) = \sum_{m=0}^M a_m P_m \left( \frac{x}{r + 1} \right) \quad \text{and} \quad g_N(x) = \sum_{n=0}^N b_n P_n(x),$$

where  $P_m$  is the Legendre polynomial of degree  $m$ . This way,  $h(x)$  is a polynomial of degree  $M$ , denoted by  $h_M(x)$ , since  $h(x)$  and  $f(x)$  should have the same degree. It follows then that  $h_M(x)$  is the product of a  $[-r, r] \times (N + 1)$  quasi-matrix  $\underline{R}$  and the coefficient vector  $\underline{b} = (b_0, b_1, \dots, b_N)^T$ . That is,

$$(1.3) \quad h_M(x) = \underline{R}\underline{b} = \left( \mathcal{F}_{f_M}[P_0] \left| \mathcal{F}_{f_M}[P_1] \right| \cdots \left| \mathcal{F}_{f_M}[P_N] \right| \right) \underline{b},$$

where the  $n$ th column of  $\underline{R}$  is the convolution of  $f_M(x)$  and  $P_n(x)$ . For convenience, we denote by  $\underline{R}_n$  the  $n$ th column of  $\underline{R}$ , with the index  $n$  starting from 0. Since  $h_M(x)$  and  $\underline{R}_n$  are both polynomials of degree not exceeding  $M$ , they can be written as Legendre series, i.e.,

$$(1.4a) \quad h_M(x) = \sum_{m=0}^M c_m P_m \left( \frac{x}{r} \right),$$

$$(1.4b) \quad \underline{R}_n(x) = \sum_{m=0}^M R_{m,n} P_m \left( \frac{x}{r} \right),$$

where  $\{R_{m,n}\}_{m=0}^M$  are the Legendre coefficients of  $\underline{R}_n(x)$ . Substituting (1.4b) into (1.3) gives

$$\sum_{m=0}^M c_m P_m\left(\frac{x}{r}\right) = \sum_{n=0}^N b_n \sum_{m=0}^M R_{m,n} P_m\left(\frac{x}{r}\right),$$

or, equivalently,

$$(1.5) \quad \underline{c} = R\underline{b},$$

where  $\underline{c} = (c_0, \dots, c_M)^T$  and  $R$  is an  $(M+1) \times (N+1)$  matrix whose  $(m, n)$ th entry is  $R_{m,n}$ .

In [3], Hale proposes constructing  $R$  via the approximation of two related convolution operators of Volterra type but limited to the case of  $r = 1$ . The primary goal of the current work is to propose a new method for the construction of  $R$  for general  $r$ . One might think that the approximation of Volterra convolution operator (see (3.1)) is more difficult than that of the Fredholm one, since the former has a variable limit of integration. However, as shown by (1.2), the approximation of the latter is probably more challenging due to the fact that one can never map the domains of both  $f_M$  and  $g_N$  to the canonical domain  $[-1, 1]$  simultaneously if  $r \neq 0$ .

The outline of this paper is as follows. In section 2, we propose a novel approach to constructing the matrix approximation  $R$  of the Fredholm convolution operator. Section 3 relates the proposed approximation to that of the Volterra convolution operator. We demonstrate in section 4 the speed and accuracy when the proposed approximations are employed to calculate convolution integrals, followed by various applications of such approximations. We close in section 5 with a summary.

**2. Constructing the convolution matrices.** We start off by the lemma below which collects some basic results of Legendre polynomials.

LEMMA 2.1. For  $n \geq 1$ ,

$$(2.1a) \quad (2n+1)P_n(x) = \frac{d}{dx} (P_{n+1}(x) - P_{n-1}(x)),$$

$$(2.1b) \quad \int P_n(x) dx = \frac{1}{2n+1} (P_{n+1}(x) - P_{n-1}(x)) + C,$$

$$(2.1c) \quad (2n+1)xP_n(x) = (n+1)P_{n+1}(x) - nP_{n-1}(x),$$

where  $C$  is the integration constant.

*Proof.* The proof can be found in many standard texts on Legendre polynomials, e.g., [9].  $\square$

Given a Legendre series  $S(x) = \sum_{l=0}^L \alpha_l P_l(x)$ , it follows immediately from (2.1b) that

$$\int S(x) dx = \sum_{l=0}^{L+1} \hat{\alpha}_l P_l(x) + C,$$

where  $(\hat{\alpha}_0, \hat{\alpha}_1, \dots, \hat{\alpha}_{L+1})^T = I_L (\alpha_0, \alpha_1, \dots, \alpha_L)^T$  and

$$I_L = \begin{pmatrix} 0 & -\frac{1}{3} & & & \\ 1 & 0 & -\frac{1}{5} & & \\ & \frac{1}{3} & 0 & -\frac{1}{7} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{(L+2) \times (L+1)}$$

is the integration matrix. Similarly, it follows from (2.1c) that the product of  $x$  and  $S(x)$ ,

$$xS(x) = \sum_{l=0}^{L+1} \tilde{\alpha}_l P_l(x),$$

where  $(\tilde{\alpha}_0, \tilde{\alpha}_1, \dots, \tilde{\alpha}_{L+1})^T = X_L (\alpha_0, \alpha_1, \dots, \alpha_L)^T$  and

$$X_L = \begin{pmatrix} 0 & \frac{1}{3} & & & & \\ 1 & 0 & & & & \\ & \frac{2}{3} & & & & \\ & & 0 & & & \\ & & & \frac{3}{7} & & \\ & & \ddots & \ddots & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{(L+2) \times (L+1)}$$

is the  $x$ -multiplication matrix. In  $I_L$  and  $X_L$ , the subscript  $L$  indicates the matrix dimension.

The following theorem shows that the matrix  $R$  is skew upper triangular, whose dimension is solely determined by  $M$ .

**THEOREM 2.2** (skew upper triangularity). *For  $R$  given in (1.5),  $R_{m,n} = 0$  for  $m + n > M$ .*

*Proof.* Since  $f_M$  is a polynomial of degree  $M$ ,

$$f_M(x - t) = \sum_{l=0}^M Q_l(x) P_l(t),$$

where  $\deg(Q_l) + \deg(P_l) \leq M$ . For  $m + n > M$ ,

$$\begin{aligned} R_{m,n} &= \frac{2m+1}{2r} \int_{-r}^r \int_{-1}^1 f_M(x-t) P_n(t) dt P_m\left(\frac{x}{r}\right) dx \\ &= \frac{2m+1}{2r} \sum_{l=0}^M \int_{-r}^r P_m\left(\frac{x}{r}\right) Q_l(x) dx \int_{-1}^1 P_l(t) P_n(t) dt = 0. \end{aligned}$$

To see why this must be zero, we assume that this is not the case. Then we must have  $l = n$  and  $\deg(Q_l) \geq m$  for some  $l$ . Thus,  $M \geq \deg(Q_l) + l \geq m + n > M$ , which is a contradiction. □

Hence, we consider mainly the case of  $M = N$  in the rest of this section. In case of  $N < M$ , we prolong  $\underline{b}$  by zeros. This theorem also justifies the employment of the Legendre polynomials in approximating the Fredholm convolution operators in that the triangularity is lost if we switch to other classical orthogonal polynomials.

What follows next is our first main result which shows a four-term recurrence relation satisfied by the entries of  $R$ .

**THEOREM 2.3** (four-term recurrence relation). *For  $m, n \geq 1$ ,*

$$(2.2a) \quad R_{m,n+1} = R_{m,n-1} + r(2n+1) \left( \frac{R_{m-1,n}}{2m-1} - \frac{R_{m+1,n}}{2m+3} \right),$$

$$(2.2b) \quad R_{m+1,n} = \frac{2m+3}{2m-1} R_{m-1,n} - \frac{1}{r} \frac{2m+3}{2n+1} (R_{m,n+1} - R_{m,n-1}),$$

$$(2.2c) \quad R_{m,n-1} = R_{m,n+1} - r(2n+1) \left( \frac{R_{m-1,n}}{2m-1} - \frac{R_{m+1,n}}{2m+3} \right),$$

$$(2.2d) \quad R_{m-1,n} = \frac{2m-1}{2m+3} R_{m+1,n} + \frac{1}{r} \frac{2m-1}{2n+1} (R_{m,n+1} - R_{m,n-1}).$$

*Proof.* We show (2.2a) only, for (2.2b)–(2.2d) are just rearrangements of (2.2a). By a change of variable  $z = y - t$ , we have

$$\underline{R}_{n+1}(y) = \int_{-1}^1 f_M(y-t)P_{n+1}(t)dt = \int_{y-1}^{y+1} f_M(z)P_{n+1}(y-z)dz.$$

Differentiating the last equation on both sides gives

$$\begin{aligned} \frac{d}{dy}\underline{R}_{n+1}(y) &= \int_{y-1}^{y+1} f_M(z) \frac{d}{dy}P_{n+1}(y-z)dz + f_M(y+1)P_{n+1}(-1) \\ &\quad - f_M(y-1)P_{n+1}(1) \\ &= \int_{y-1}^{y+1} f_M(z) \left[ (2n+1)P_n(y-z) + \frac{d}{dy}P_{n-1}(y-z) \right] dz \\ &\quad + f_M(y+1)P_{n-1}(-1) - f_M(y-1)P_{n-1}(1) \\ &= (2n+1)\underline{R}_n(y) + \frac{d}{dy}\underline{R}_{n-1}(y), \end{aligned}$$

where we have used (2.1a) and  $P_{n+1}(-1) = P_{n-1}(-1)$  and  $P_{n+1}(1) = P_{n-1}(1)$ . Integrating on both sides over  $[-r, x]$  yields

$$\underline{R}_{n+1}(x) = (2n+1) \int_{-r}^x \underline{R}_n(y)dy + \underline{R}_{n-1}(x) + C.$$

Substituting (1.4b) into the last equation and noting the triangularity of  $R$ , we have

$$\begin{aligned} \sum_{m=0}^{M-n-1} R_{m,n+1}P_m\left(\frac{x}{r}\right) &= (2n+1) \sum_{m=0}^{M-n} R_{m,n} \int_{-r}^x P_m\left(\frac{y}{r}\right) dy \\ &\quad + \sum_{m=0}^{M-n+1} R_{m,n-1}P_m\left(\frac{x}{r}\right) + C. \end{aligned}$$

Using (2.1b) and matching the Legendre coefficients of  $P_m\left(\frac{x}{r}\right)$  for  $m \geq 1$  leads to (2.2a).  $\square$

The four identities in (2.2d) suggest that the entries in  $R$  can be calculated recursively in four ways: columnwise rightward, rowwise downward, columnwise leftward, and rowwise upward. It may not be immediately obvious how to initiate the recursion with (2.2c) or (2.2d), and we leave it aside for the moment. To recurse with (2.2a) or (2.2b), it is apparent that we have to first figure out the first two columns and the zeroth row or the first two rows and the zeroth column, respectively. To this end, we first give the following lemma on representing  $P_j\left(\frac{x+1}{r+1}\right) - P_j\left(\frac{x-1}{r+1}\right)$  in terms of  $\{P_k\left(\frac{x}{r}\right)\}_{k=0}^{j-1}$ . For notational convenience, we let  $\underline{P}_j = (P_0\left(\frac{x}{r}\right), P_1\left(\frac{x}{r}\right), \dots, P_j\left(\frac{x}{r}\right))$ .

LEMMA 2.4. *Suppose that*

$$P_j\left(\frac{x+1}{r+1}\right) - P_j\left(\frac{x-1}{r+1}\right) = \sum_{k=0}^{j-1} w_{k,j}P_k\left(\frac{x}{r}\right).$$

*If the coefficients  $w_{k,j}$  are collected in the matrix  $W_L = (w_{k,j}) \in \mathbb{R}^{(L+1) \times (L+2)}$ ,  $W_L$  is strictly upper triangular with*

$$W_L(0:2, 0:3) = \begin{pmatrix} 0 & \frac{2}{r+1} & 0 & \frac{2r^2-6r+2}{(r+1)^3} \\ 0 & 0 & \frac{6r}{(r+1)^2} & 0 \\ 0 & 0 & 0 & \frac{10r^2}{(r+1)^3} \end{pmatrix}.$$

With the further assumption that  $w_{k,j} = 0$  for  $k < 0$  for all  $j$ , the entries of  $W_L$  satisfy a 9-term recurrence relation

$$(2.3) \quad w_{k,j+2} = B_{k-1,j+1}w_{k-1,j+1} + C_{k+1,j+1}w_{k+1,j+1} + E_{k-2,j}w_{k-2,j} + F_{k,j}w_{k,j} \\ + G_{k+2,j}w_{k+2,j} + H_{k-1,j-1}w_{k-1,j-1} + J_{k+1,j-1}w_{k+1,j-1} + K_{k,j-2}w_{k,j-2},$$

where

$$A_{k,j+2} = \frac{2j+3}{(j+2)(r+1)}, \quad B_{k-1,j+1} = \frac{2rkA_{k,j+2}}{2k-1}, \quad C_{k+1,j+1} = \frac{2r(k+1)A_{k,j+2}}{2k+3}, \\ E_{k-2,j} = -\frac{r^2k(k-1)(2j+1)A_{k,j+2}}{(r+1)(2k-3)(2k-1)(j+1)}, \quad F_{k,j} = F_{k,j}^1 + F_{k,j}^2 + F_{k,j}^3, \\ F_{k,j}^1 = \frac{(2j+1)A_{k,j+2}}{(j+1)(r+1)}, \quad F_{k,j}^2 = -\frac{(r+1)(j+1)A_{k,j+2}}{2j+3} - \frac{(r+1)j^2A_{k,j+2}}{(j+1)(2j-1)}, \\ F_{k,j}^3 = -\frac{r^2(2j+1)(2k^2-2k-1)A_{k,j+2}}{(r+1)(j+1)(4k^2-4k-3)}, \\ G_{k+2,j} = -\frac{r^2(2j+1)(k+1)(k+2)A_{k,j+2}}{(r+1)(j+1)(2k+5)(2k+3)}, \quad H_{k-1,j-1} = \frac{2rjkA_{k,j+2}}{(j+1)(2k-1)}, \\ J_{k+1,j-1} = \frac{2rj(k+1)A_{k,j+2}}{(j+1)(2k+3)}, \quad K_{k,j-2} = -\frac{j(j-1)(r+1)A_{k,j+2}}{(j+1)(2j-1)}.$$

Additionally,  $w_{k,j} = 0$  when  $k + j$  is even.

*Proof.* Let  $\phi_j(x) = P_j(\frac{x+1}{r+1}) - P_j(\frac{x-1}{r+1})$  and  $\psi_j(x) = P_j(\frac{x+1}{r+1}) + P_j(\frac{x-1}{r+1})$ . It is easy to see that  $\phi_j(x)$  is a polynomial of degree  $j-1$  for  $j \geq 1$ , implying that  $W_L$  is strictly upper triangular. The first few  $\phi_j(x)$  can be easily obtained with a bit of algebraic manipulation:

$$\phi_0(x) = 0, \quad \phi_1(x) = \frac{2}{r+1}, \quad \phi_2(x) = \frac{6r}{(r+1)^2}P_1\left(\frac{x}{r}\right), \\ \phi_3(x) = \frac{2r^2-6r+2}{(r+1)^3} + \frac{10r^2}{(r+1)^3}P_2\left(\frac{x}{r}\right).$$

It follows from (2.1c) that

$$\phi_{j+1}(x) = \frac{2j+1}{j+1} \left[ \frac{x+1}{r+1}P_j\left(\frac{x+1}{r+1}\right) - \frac{x-1}{r+1}P_j\left(\frac{x-1}{r+1}\right) \right] \\ - \frac{j}{j+1} \left[ P_{j-1}\left(\frac{x+1}{r+1}\right) - P_{j-1}\left(\frac{x-1}{r+1}\right) \right] \\ = \frac{2j+1}{(j+1)(r+1)}(x\phi_j(x) + \psi_j(x)) - \frac{j}{j+1}\phi_{j-1}(x).$$

Similarly,

$$\psi_{j+1}(x) = \frac{2j+1}{(j+1)(r+1)}(x\psi_j(x) + \phi_j(x)) - \frac{j}{j+1}\psi_{j-1}(x).$$

Combining the last two equations, we have

$$(2.4) \quad \phi_{j+2}(x) = \frac{2j+3}{(j+2)(r+1)} \left[ 2x\phi_{j+1}(x) + \frac{2jx}{j+1}\phi_{j-1}(x) - \frac{j(j-1)(r+1)}{(j+1)(2j-1)}\phi_{j-2}(x) \right] \\ + \left( \frac{2j+1}{(j+1)(r+1)}(1-x^2) - \frac{(j+1)(r+1)}{2j+3} - \frac{j^2(r+1)}{(j+1)(2j-1)} \right) \phi_j(x),$$

which relates  $\phi_{j-2}(x), \dots, \phi_{j+2}(x)$  together. To get rid of the multiplying factors of  $x$  and  $x^2$ , we note

$$(2.5) \quad \begin{aligned} \left(\frac{x}{r}\right) \phi_j(x) &= \underline{P}_j X_{j-1}(w_{0,j}, w_{1,j} \dots w_{j-1,j})^T, \\ \left(\frac{x}{r}\right)^2 \phi_j(x) &= \underline{P}_{j+1} X_j X_{j-1}(w_{0,j}, w_{1,j} \dots w_{j-1,j})^T. \end{aligned}$$

Substituting (2.5) into (2.4) and matching up the Legendre coefficients in the resulting equation gives (2.3). In case of  $k + j$  being even,  $w_{k,j} = 0$  follows from the 9-term recurrence relation.  $\square$

We are now in a position to construct the zeroth column of  $R$ .

**THEOREM 2.5** (construction of the zeroth column). *The elements of the zeroth column of  $R$  are*

$$(2.6) \quad (r+1)W_M I_M \underline{a},$$

where  $\underline{a} = (a_0, a_1, \dots, a_M)^T$ .

*Proof.* The zeroth column of  $R$  is the vector collecting the coefficients of  $\int_{-1}^1 f_M(x-t) dt$  in the basis  $\{P_m(\frac{x}{r})\}_{m=0}^M$ . Using (2.1b), we have

$$(2.7) \quad \begin{aligned} \int_{-1}^1 f_M(x-t) dt &= \int_{-1}^1 \sum_{m=0}^M a_m P_m\left(\frac{x-t}{r+1}\right) dt \\ &= (r+1) \sum_{m=0}^{M+1} \hat{a}_m \left[ P_m\left(\frac{x+1}{r+1}\right) - P_m\left(\frac{x-1}{r+1}\right) \right], \end{aligned}$$

where  $\hat{\underline{a}} = (\hat{a}_0, \dots, \hat{a}_{M+1})^T = I_M \underline{a}$ . It follows from Lemma 2.4 that

$$\sum_{m=0}^{M+1} \hat{a}_m \left[ P_m\left(\frac{x+1}{r+1}\right) - P_m\left(\frac{x-1}{r+1}\right) \right] = \underline{P}_M W_M \hat{\underline{a}},$$

which, combined with (2.7), gives

$$\int_{-1}^1 f_M(x-t) dt = (r+1) \underline{P}_M W_M I_M \underline{a}. \quad \square$$

The following theorem shows the construction of the first column of  $R$ .

**THEOREM 2.6** (construction of the first column). *The elements of the first column of  $R$  are*

$$(2.8) \quad (r(r+1)X_{M+1}W_M I_M - (r+1)^2 W_{M+1} I_{M+1} X_M) \underline{a}.$$

*Proof.* Since the entries in the first column of the matrix  $R$  are the coefficients of  $\int_{-1}^1 f_M(x-t) t dt$ ,

$$\begin{aligned} \int_{-1}^1 f_M(x-t) t dt &= (r+1) \sum_{m=0}^M a_m \int_{-1}^1 P_m\left(\frac{x-t}{r+1}\right) \left(\frac{x}{r+1} - \frac{x-t}{r+1}\right) dt \\ &= x \sum_{m=0}^M a_m \int_{-1}^1 P_m\left(\frac{x-t}{r+1}\right) dt - (r+1) \sum_{m=0}^M a_m \int_{-1}^1 \frac{x-t}{r+1} P_m\left(\frac{x-t}{r+1}\right) dt. \end{aligned}$$

By (2.1c) and Theorem 2.5, we have

$$\sum_{m=0}^M a_m \int_{-1}^1 \frac{x-t}{r+1} P_m \left( \frac{x-t}{r+1} \right) dt = (r+1) \underline{P}_{M+1} W_{M+1} I_{M+1} X_M \underline{a}.$$

For the first term, up to a constant  $r$ ,

$$\frac{x}{r} \sum_{m=0}^M a_m \int_{-1}^1 P_m \left( \frac{x-t}{r+1} \right) dt = (r+1) \underline{P}_{M+1} X_{M+1} W_M I_M \underline{a},$$

which follows from a similar argument. The last three equations gives (2.8).  $\square$

The length of (2.8) is  $M + 2$ , whose last two entries are exactly zero, since  $\underline{R}_1(x)$  is a polynomial of degree  $M - 1$ .

The zeroth and first rows can be constructed analogously. Lemma 2.7 and Theorem 2.8 below are the results in parallel for rows, and we omit the proofs.

LEMMA 2.7. *Suppose that*

$$P_j \left( \frac{r-t}{r+1} \right) - P_j \left( -\frac{r+t}{r+1} \right) = \sum_{k=0}^{j-1} \hat{w}_{k,j} P_k(t),$$

If the coefficients  $\hat{w}_{k,j}$  are collected in the matrix  $\hat{W}_L = (\hat{w}_{k,j}) \in \mathbb{R}^{(L+1) \times (L+2)}$ ,  $\hat{W}_L$  is strictly upper triangular with

$$\hat{W}_L(0:2, 0:3) = \begin{pmatrix} 0 & \frac{2r}{r+1} & 0 & \frac{r(2r^2-6r+2)}{(r+1)^3} \\ 0 & 0 & -\frac{6r}{(r+1)^2} & 0 \\ 0 & 0 & 0 & \frac{10r}{(r+1)^3} \end{pmatrix}.$$

With the further assumption that  $\hat{w}_{k,j} = 0$  for  $k < 0$  for all  $j$ , the entries of  $\hat{W}_L$  satisfy a 9-term recurrence relation

$$\hat{w}_{k,j+2} = \hat{B}_{k-1,j+1} \hat{w}_{k-1,j+1} + \hat{C}_{k+1,j+1} \hat{w}_{k+1,j+1} + \hat{E}_{k-2,j} \hat{w}_{k-2,j} + \hat{F}_{k,j} \hat{w}_{k,j} \\ + \hat{G}_{k+2,j} \hat{w}_{k+2,j} + \hat{H}_{k-1,j-1} \hat{w}_{k-1,j-1} + \hat{J}_{k+1,j-1} \hat{w}_{k+1,j-1} + \hat{K}_{k,j-2} \hat{w}_{k,j-2},$$

where

$$\hat{B}_{k-1,j+1} = -B_{k-1,j+1}/r, \quad \hat{C}_{k+1,j+1} = -C_{k+1,j+1}/r, \quad \hat{E}_{k-2,j} = E_{k-2,j}/r^2, \\ \hat{F}_{k,j} = \hat{F}_{k,j}^1 + \hat{F}_{k,j}^2 + \hat{F}_{k,j}^3, \quad \hat{F}_{k,j}^1 = r^2 F_{k,j}^1, \quad \hat{F}_{k,j}^2 = F_{k,j}^2, \quad \hat{F}_{k,j}^3 = F_{k,j}^3/r^2, \\ \hat{G}_{k+2,j} = G_{k+2,j}/r^2, \quad \hat{H}_{k-1,j-1} = -H_{k-1,j-1}/r, \\ \hat{J}_{k+1,j-1} = -J_{k+1,j-1}/r, \quad \hat{K}_{k,j-2} = K_{k,j-2}.$$

Additionally,  $\hat{w}_{k,j} = 0$  when  $k + j$  is even.

For the first two rows, we also need the diagonal matrix  $D_M = \text{diag}(1, 1/3, \dots, 1/(2M + 1)) \in \mathbb{R}^{(M+1) \times (M+1)}$  which comes from the integration constant.

THEOREM 2.8. *The elements of the zeroth and the first rows of  $R$  are*

$$(2.9) \quad \frac{r+1}{r} D_M \hat{W}_M I_M \underline{a},$$

$$(2.10) \quad \frac{3(r+1)}{r^2} D_{M+1} \left( (r+1) \hat{W}_{M+1} I_{M+1} X_M + X_{M+1} \hat{W}_M I_M \right) \underline{a},$$

respectively.

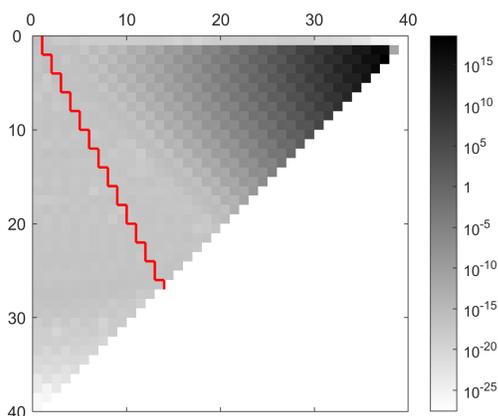


FIG. 1. Error growth when recursing for  $R$  corresponding to  $f_{39}$  using (2.2a), which is stable in the region that is to the left of the red borderline. (Figure in color online.)

The calculation of  $R$  could have been as easy as suggested by (2.2a) and (2.2b)—generate the first two columns and the first row or the first two rows and the first column followed by recursing the entries columnwise or rowwise, respectively. Unfortunately, (2.2a) and (2.2b) are numerically unstable. In fact, recursing with (2.2a) and (2.2b) is only stable when  $\frac{r(2n+1)}{2m-1} \leq 1$  and  $\frac{2m+3}{r(2n+1)} \leq 1$ , respectively. Otherwise, the error in  $R_{m\pm 1, n}$  and  $R_{m, n\pm 1}$  are to be amplified repeatedly in the course of recursion. In the worst-case scenario, the error in certain entries could be amplified approximately by a factor of  $M^{\mathcal{O}(n)}$ . To see the instability, consider  $R \in \mathbb{R}^{40 \times 40}$  defined by  $f_{39}(x)$  and  $r = 2$ , which is a Legendre series of degree 39 with all the coefficients set to 1. Figure 1 shows the entrywise error in  $R$  obtained by recursing columnwise with (2.2a). The largest error is  $2.7098 \times 10^{30}$  in this example, occurring at  $R_{1,37}$ .

To identify the entries for which each of (2.2d) is stable, we first safely ignored the amplification effect caused by  $\frac{2m+3}{2m-1}$  in (2.2b) since it is close enough to 1 for large  $m$ . Thus, the four recurrence relations in (2.2d) are stable, respectively, when

$$(2.11) \quad r \frac{2n+1}{2m-1} \leq 1, \quad \frac{2m+3}{r(2n+1)} \leq 1, \quad r \frac{2n+1}{2m-1} \leq 1, \quad \frac{2m-1}{r(2n+1)} \leq 1.$$

If we further ignore the constants in the fractions in (2.11), (2.2a) and (2.2c) are stable when  $m/n \geq r$  whereas (2.2b) and (2.2d) are stable for  $m/n \leq r$ . The stability regions for (2.2d) are marked by red lines in Figure 2. For  $r \geq 1$ , (2.2a) and (2.2c) are stable in  $S_1$ , whereas (2.2b) and (2.2d) are stable in  $S_{23} = S_2 + S_3$ . For  $0 < r \leq 1$ , (2.2a) and (2.2c) are stable in  $\hat{S}_{12} = \hat{S}_1 + \hat{S}_2$ , while (2.2b) and (2.2d) are stable in  $\hat{S}_3$ .

Now it should be clear that we have four possible recursing strategies for entries of  $R$  when  $r \geq 1$ : (1) recursing with (2.2a) rightward in  $S_1$  followed by recursing with (2.2d) upward in  $S_{23}$ , (2) recursing with (2.2b) downward in  $S_{23}$  followed by recursing with (2.2c) leftward in  $S_1$ , (3) recursing with (2.2a) rightward in  $S_1$  followed by recursing with (2.2b) downward in  $S_{23}$ , and (4) recursing with (2.2b) downward in  $S_{23}$  followed by recursing with (2.2a) rightward in  $S_1$ . We would not prefer (3) and (4) since we have to precompute both the first two columns and rows, which incurs unnecessary cost. If we take (2), we have to know certain entries in  $S_1$  to recurse downward for the entries in  $S_2$ . Hence, when  $r > 1$  we recurse following strategy (1) (Figure 3a). Following an analogous argument, we recurse with (2.2b) downward in  $\hat{S}_3$  followed by recursing with (2.2c) leftward in  $\hat{S}_{12}$  for  $0 \leq r \leq 1$  (Figure 3b).

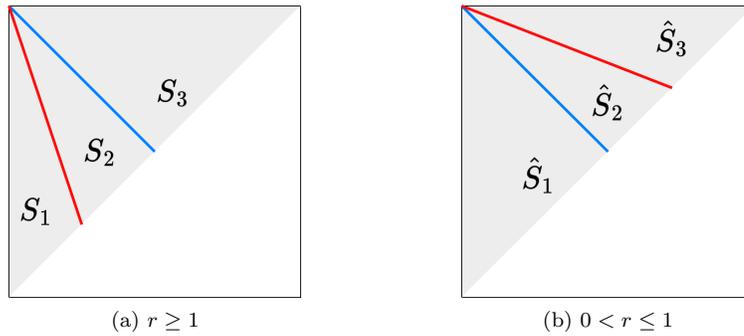


FIG. 2. Partition of the nonzero region by the borderline of stability (red) and the line of dependence (blue). (Figure in color online.)

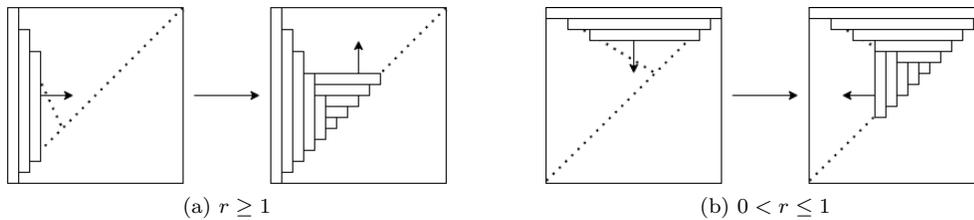


FIG. 3. Stable recursing strategies.

We summarize the stable recursions in Algorithm 2.1. If we recalculate the example in Figure 1 using Algorithm 2.1, the largest entrywise error is about  $2.3 \times 10^{-16}$ .

Since each nonzero entry of  $W_M$  or  $\hat{W}_M$  is obtained at a cost of approximately 145 flops and only  $M^2/4$  entries of  $W_M$  or  $\hat{W}_M$  are nonzero, the total complexity of constructing  $W_M$  or  $\hat{W}_M$  is about  $36M^2$ . If we ignore the  $\mathcal{O}(M)$  operations, the dominant cost in evaluating (2.6) and (2.8) or (2.9) and (2.10) is the three multiplication of  $W_M$  or  $\hat{W}_M$  with vectors, totaling  $6M^2$  flops. The cost for generating the first two columns or rows is thus  $42M^2$ . For the  $M^2/(2(r+1))$  entries in  $S_1$  or  $\hat{S}_{12}$ , recursing for each entry with (2.2a) or (2.2c) requires 12 flops, leading to a total cost of  $6M^2/(r+1)$  flops. Analogously, each entry in  $S_{23}$  or  $\hat{S}_3$  can be obtained following (2.2d) and (2.2b), respectively with 15 flops. For the  $rM^2/(2(r+1))$  entries in  $S_{23}$  or  $\hat{S}_3$ , the total cost is  $7rM^2/(r+1)$ . Thus, the total complexity in the recursing part of Algorithm 2.1 is roughly  $7M^2$ , independent of  $r$ . We list the stepwise complexities in Algorithm 2.1 and note that the overall cost of Algorithm 2.1 is roughly  $49M^2$  flops.

If only the coefficients of the convolution  $h_M$  are what we need, the cost of storage can be reduced from  $\mathcal{O}(M^2)$  to  $\mathcal{O}(M)$  by allocating only three  $M$ -vectors, two for the recursion and one for accumulating  $c$ .

**3. Relation to the convolution operator of Volterra type.** For  $\check{f}, \check{g} : [-1, 1] \rightarrow \mathbb{R}$ , the convolution operator of Volterra type is defined as

$$(3.1) \quad \check{h} = \mathcal{V}_{\check{f}}[\check{g}](x) = \int_{-1}^x \check{f}(x-t)\check{g}(t)dt,$$

where the resulting function  $\check{h} : [-2, 0] \rightarrow \mathbb{R}$ . Suppose that  $\check{f}(x)$  and  $\check{g}(x)$  are approximated by the Legendre series  $\check{f}_M(x) = \sum_{m=0}^M \check{a}_m P_m(x)$  and  $\check{g}_N(x) = \sum_{n=0}^N \check{b}_n P_n(x)$ , respectively. Then  $\check{h}_{M+N+1}(x) = \mathcal{F}_{\check{f}_M}[\check{g}_N](x)$  is a Legendre series of length  $M+N+2$ ,

---

**Algorithm 2.1.** Stable construction of the convolution matrix  $R$ .
 

---

```

if  $r \geq 1$  then
  Construct the first two columns by (2.6) and (2.8).  $\triangleright 42M^2$ 
  Recursing with (2.2a) rightward for the entries in  $S_1$ .  $\triangleright 6M^2/(r+1)$ 
  Recursing with (2.2d) upward for the entries in  $S_{23}$ .  $\triangleright 7rM^2/(r+1)$ 
else
  Construct the first two rows by (2.9) and (2.10).  $\triangleright 42M^2$ 
  Recursing with (2.2b) downward for the entries in  $\hat{S}_3$ .  $\triangleright 7rM^2/(r+1)$ 
  Recursing with (2.2c) leftward for the entries in  $\hat{S}_{12}$ .  $\triangleright 6M^2/(r+1)$ 
end if

```

---

and the  $(M+N+2) \times (N+1)$  matrix that relates  $\check{h}_{M+N+1}(x)$  and  $\check{g}_N(x)$  serves as an approximation of  $\mathcal{V}_{\hat{f}}$ . If we denote by  $V_{\hat{f}}$  this matrix approximation of  $\mathcal{V}_{\hat{f}}$ ,  $V_{\hat{f}}$  is also obtainable by a four-term recurrence relation similar to (2.2d) at a cost of  $\mathcal{O}((M+N)N)$ . Particularly,  $V_{\hat{f}}$  is banded with the upper and lower bandwidths both being exact  $M+1$ . For details, see [5, 13].

For a function  $\varphi$ , let  $\varphi_{\{a,b\}}$  denote the restriction of  $\varphi$  to the interval  $[a, b]$  and  $\hat{\varphi} = \varphi(-x)$ . It is shown in [3] that when  $r = 1$  a convolution operator of Fredholm type can be written as the combination of two Volterra convolution operators

$$(3.2) \quad \mathcal{F}_f[g](x) = \mathcal{V}_{f_1}[g](x) + \mathcal{V}_{\hat{f}_2}[\hat{g}](\hat{x}),$$

where  $f_1 = f_{\{0,2\}}$  and  $f_2 = f_{\{-2,0\}}$ . Let  $\hat{I} = \text{diag}(1, -1, 1, -1, \dots)$  be the alternating diagonal matrix. Suppose that both  $f_1$  and  $\hat{f}_2$  can be approximated by Legendre series of degree  $M$  and  $g$  by a Legendre series of degree  $N$ . Hence, the  $(M+N+2) \times (N+1)$  matrix approximation of the Fredholm convolution operator

$$(3.3) \quad \hat{R} = V_{f_1} + \hat{I}V_{\hat{f}_2}\hat{I}.$$

Note that  $\hat{R}$  may have different dimensions from  $R$ , as the latter is assumed square throughout. Obtaining  $\hat{R}$  via (3.3) incurs the construction of both  $V_f$  and  $V_{\hat{f}}$ , the cost being  $\mathcal{O}((M+N)N)$ . On the contrary, the cost of constructing  $R$  following Algorithm 2.1 is  $\mathcal{O}(M^2)$ , independent of  $N$ .

Since the dimensions of  $V_f$  and  $V_{\hat{f}}$  are usually determined by the length of  $g_N(x)$ , a great deal of the computation might be wasteful when  $N$  is much larger than  $M$ . When  $\hat{R}$  is constructed via (3.3), cancellations are expected for entries in the band except those in the top-left triangular part. Furthermore, the cancellation is never perfect—entrywise residuals are usually on the order of machine epsilon or greater. When such a  $\hat{R}$  is used to calculate convolutions or solve convolution integral equations, the cancellation errors may lead to inaccurate, if not totally erroneous, results. This is particularly the case if certain entries in the nonzero triangular part are on the same order of or smaller than the cancellation errors.

Let  $f_{17}(x)$  be the Legendre series of degree 17 which approximates  $e^x$  on  $[-2, 2]$  and consider the Fredholm convolution operator defined by  $f_{17}(x)$ . Figure 4a shows the entrywise magnitude of  $\hat{R}$  obtained by (3.3), where the nonzero entries that could be obtained directly by Algorithm 2.1 are enclosed by the borderline in red. Comparing with Figure 4b which displays the entrywise magnitude of  $R$  constructed using Algorithm 2.1, we can see that the cancellation errors are of the same magnitude as or even larger than many nonzero entries, making the skew upper-triangular structure

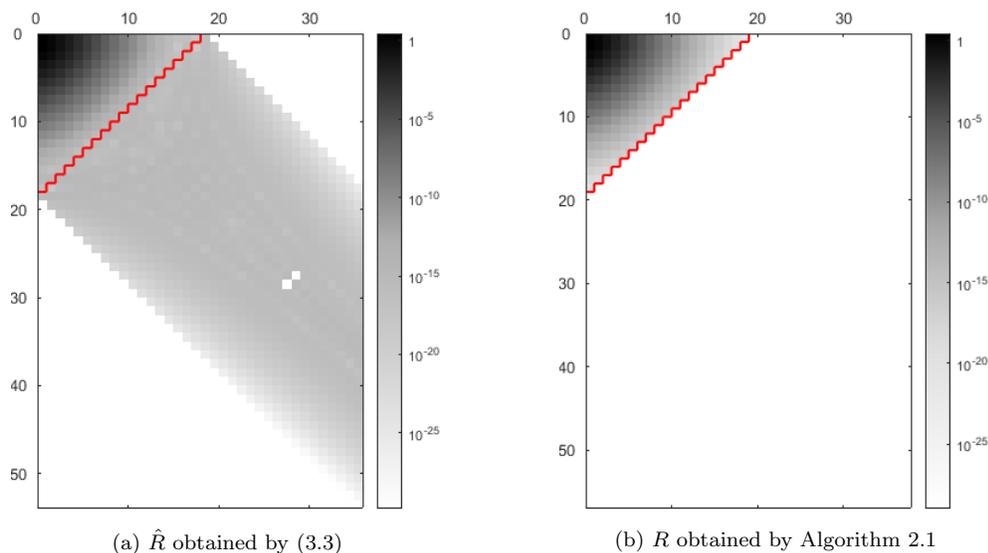


FIG. 4. The entrywise magnitude of the matrix approximations  $\hat{R}$  and  $R$  to the Fredholm convolution operator defined by the Legendre series that approximates  $e^x$  for  $x \in [-2, 2]$ . When (3.3) is used, cancellation errors are introduced so that the triangularity (marked by the red borderline) is lost and we can only see bandedness since the matrix approximation to the Volterra convolution operator (3.1) is strictly banded when both  $\hat{f}(x)\hat{g}(x)$  are Legendre series [5, 13]. In contrast, the triangularity is respected by Algorithm 2.1. (Figure in color online.)

invisible. This may explain why the triangularity is overlooked by Hale and Townsend in their works [3, 5].

For  $r \neq 1$ , there is unfortunately no direct relation like (3.2). Hence, Algorithm 2.1 is so far the only way to construct coefficient-based matrix approximations to the Fredholm convolution operator for  $r \neq 1$  (see subsection 4.3). However, the Fredholm convolution of two functions can still be calculated via convolutions of Volterra type [5] but in a less efficient way. To see this, consider the case of  $r < 1$ . It is shown by Hale and Townsend that

$$(3.4) \quad \mathcal{F}_f[g](x) = \mathcal{V}_{f_{\{r-1, r+1\}}} [g](x) + \mathcal{V}_{\hat{f}_{\{-r-1, r-1\}}} [\hat{g}_{\{-2r+1, 1\}}] (-x), \quad x \in [-r, r].$$

Thus, we need to make two restrictions of  $f$  and one of  $g$ . These are done by evaluating  $f$  and  $g$  at Chebyshev grids of  $M$  and  $N$  points, respectively, via the Clenshaw algorithm at a cost of  $\mathcal{O}(M^2 + N^2)$ , followed by Chebyshev transforms from values to coefficients via FFT in  $\mathcal{O}(M \log M + N \log N)$  flops and Chebyshev–Legendre transforms to recover the Legendre coefficients for each restriction in  $\mathcal{O}(M(\log M)^2 + N(\log N)^2)$  flops. Finally, constructing two Volterra convolution matrices costs  $\mathcal{O}((M + N)M)$  flops.

For  $r > 1$ ,

$$(3.5) \quad \mathcal{F}_f[g](x) = \sum_{j=1}^{\lfloor r \rfloor} \mathcal{F}_{f^{[j]}} [g](x) + \mathcal{F}_{f_{\{2\lfloor r \rfloor - r - 1, r + 1\}}} [g](x), \quad x \in [-r, r],$$

where  $f^{[j]} = f_{\{-(r+1)+2(j-1), -(r+1)+2(j+1)\}}$  and the summation and the plus sign should be understood as concatenations. Each summand  $\mathcal{F}_{f^{[j]}} [g](x)$  satisfies  $r = 1$

since the length of the domain of  $f^{[j]}$  is twice that of  $g$ . When  $r$  is an integer, the last term in (3.5) vanishes. In such a case, we need to make  $r$  restrictions of  $f$  and construct  $2r$  approximations of the Volterra convolution operators, incurring  $\mathcal{O}(rM(M+N+(\log M)^2))$  flops. If  $r$  is not an integer, the last term  $\mathcal{F}_{f_{\{2\lfloor r \rfloor - r - 1, r + 1\}}}[g](x)$  corresponds to  $r < 1$  and can be treated following (3.4), adding an extra  $\mathcal{O}(N^2 + N(\log N)^2)$  term to the total complexity.

In contrast to the complexities we just worked out, the cost of Algorithm 2.1 is independent of  $N$  and  $r$ . This immediately suggests the advantage of the proposed approximation in calculating the convolution for which the domain of  $f$  is much more sizable than that of  $g$  or  $N$  is much greater than  $M$ . See subsection 4.1 for numerical experiments.

**4. Numerical examples.** In this section, we first test the use of the approximation to the Fredholm convolution operator for calculating the convolution of two given functions (subsections 4.1 and 4.2), followed by its applications in solving convolution integral equations (subsection 4.3) and the calculation of pseudospectra (subsection 4.4). All the numerical experiments are performed in JULIA v1.9.3 on a laptop with a 14 core 4.7 GHz Intel Core i7 CPU and 16 GB RAM. The execution times are obtained using `BenchmarkTools.jl`.

**4.1. Speed in computing Fredholm convolutions.** We evaluate the Fredholm convolution of two given Legendre series with randomly generated coefficients using the proposed approximation and compare the computational time with that of the Hale–Townsend method [5, section 5]. Since the Hale–Townsend method depends on  $M$ ,  $N$ , and  $r$ , we time the computation by varying one of them and fixing the other two.

Each panel of Figure 5 has fixed value of  $r$  and shows the dependence of the execution time on  $M$  for various values of  $N/M$ . It is shown that the time grows quadratically with  $M$  for both methods, as expected. The dependence on  $N$  of the Hale–Townsend method is suggested by the curves corresponding different ratios of  $N/M$ , whereas the proposed method is  $N$ -independent. Since the proposed method is independent of  $r$ , its execution times remain consistent and are thus indistinguishable across the panels. On the other hand, the time for the Hale–Townsend method grows proportionally with  $r$ . In the case of  $r = 1$ , the new method based on the proposed approximation is  $5\times$  to  $10\times$  faster than the Hale–Townsend method. For  $r = 100$ , the speed-up is roughly 200 folds.

Figure 6 shows the time for varying  $N/M$  ratio at fixed  $M$  and  $r$ . We could have shown the plots for time versus  $N$ . However, the use of the ratio  $N/M$  helps align the curves for different  $M$  so that the dependence on  $N$  is better illustrated. Since the proposed method is  $N$ -independent the solid curves are largely straight lines, whereas the time for the Hale–Townsend method scales linearly with  $N$ . The greater  $N$  is, the more conspicuous the speed-ups are gained by the new method. Figure 7 displays the (in)dependence on  $r$  of the two methods. This is again where the proposed method is favored as its computational time is constant despite the length of the intervals. In Figures 5 to 7,  $M$ ,  $N$ , and  $r$  are chosen to be typical and representative, and the new method is faster in all these tests with clear advantages for  $r \gg 1$  or  $N \gg M$ .

**4.2. An example for demonstrating the Weierstrass approximation theorem.** Our second example is the evaluation of the Fredholm convolution of  $f(x) = \sin(1/x)\sin(1/(\sin(1/x)))$  with  $x \in [0.2885554757, 0.3549060246]$  and  $g(x) = \exp(-x^2/(4t))/\sqrt{4\pi t}$  with  $t = 10^{-7}$  and  $x \in [-3 \times 10^{-3}, 3 \times 10^{-3}]$ . This example

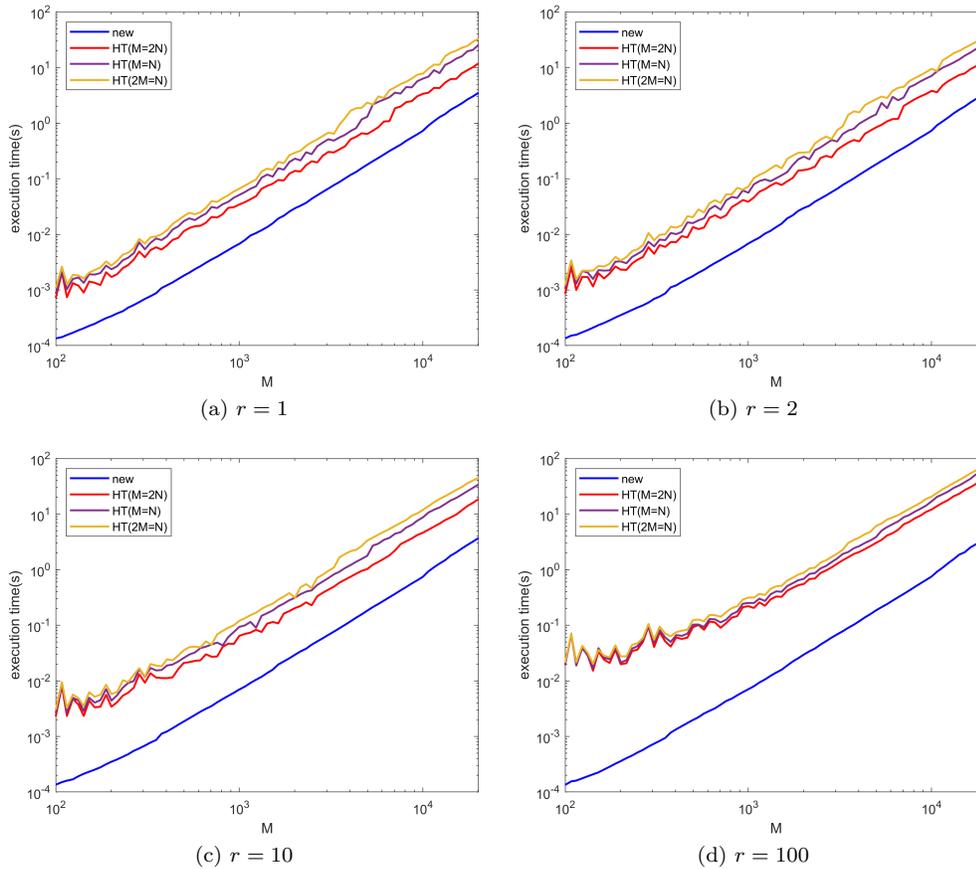


FIG. 5. Comparison between the proposed method (blue) and the Hale–Townsend method (other): computational time versus  $M$  for various choices of  $N$  and  $r$ . The proposed method is significantly faster than the Hale–Townsend method for various choices of  $N$  and  $r$ . (Figure in color online.)

is borrowed from [10, section 6], where it is used to demonstrate the Weierstrass approximation theorem. As in [10, section 6], we approximate  $f(x)$  and  $g(x)$  by polynomials of degree 2000 and 65, respectively. Figure 8 shows the errors obtained by comparing the computed convolution using the Hale–Townsend method and the proposed method with that obtained in octuple precision.<sup>1</sup> The execution times of the proposed method and the Hale–Townsend method are 38 and 98 milliseconds, respectively. That is, the new method is  $2.5\times$  faster than the Hale–Townsend one even though  $N$  is much smaller than  $M$  in this example.

**4.3. Convolution integral equation.** The approximation of Fredholm convolution operators is the key in solving convolution integral equations of Fredholm type [3]. For instance, Love’s integral equation [6]

<sup>1</sup>The 256-bit octuple precision is the default format of JULIA’s `BigFloat` type of floating point number. `BigFloat`, based on the GNU MPFR library, is the arbitrary precision floating point number type in JULIA.

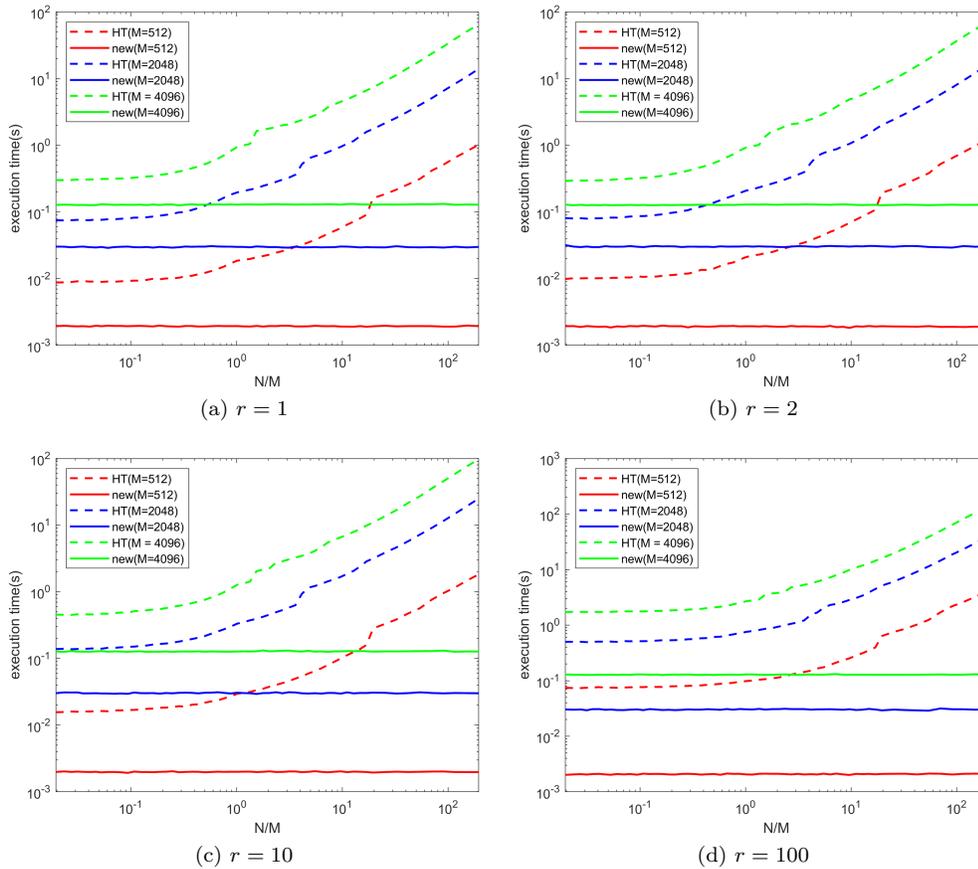


FIG. 6. Comparison between the proposed method (solid) and the Hale-Townsend method (dashed): computational time versus  $N/M$  for various choices of  $M$  and  $r$ . Unlike the Hale-Townsend method, the execution time for the proposed method does not depend on  $N$ .

$$(4.1) \quad y(t) = f(t) - \frac{\delta}{\pi} \int_{-1}^1 \frac{y(s)}{\delta^2 + (t-s)^2} ds, \quad -1 \leq t \leq 1,$$

is a Fredholm convolution integral equation of the second kind that arises in electrostatics. We choose  $\delta = 1$  and  $f(t) = 1/(1/4 + 4t^2)$  for our test. The resulting linear system constructed using the proposed approximation is an arrow-shaped one of infinite dimensions. This suggests a spectral method for Fredholm convolution integral equations in the framework of infinite-dimensional linear algebra [7], though in practice we need only solve the  $M \times M$  subsystem that encloses the nonzero triangular part. The error in the computed solution is shown in Figure 9a by taking the solution obtained in octuple precision as the exact. Alongside, the error in the solution obtained using (3.3) is also plotted for comparison.

If we modify (4.1) so that  $r \neq 1$ , the method of [3] ceases to work while the proposed approximation continues to produce accurate solutions. Figure 9b shows the error in the solution to (4.1) but with  $-5 \leq t \leq 5$ .

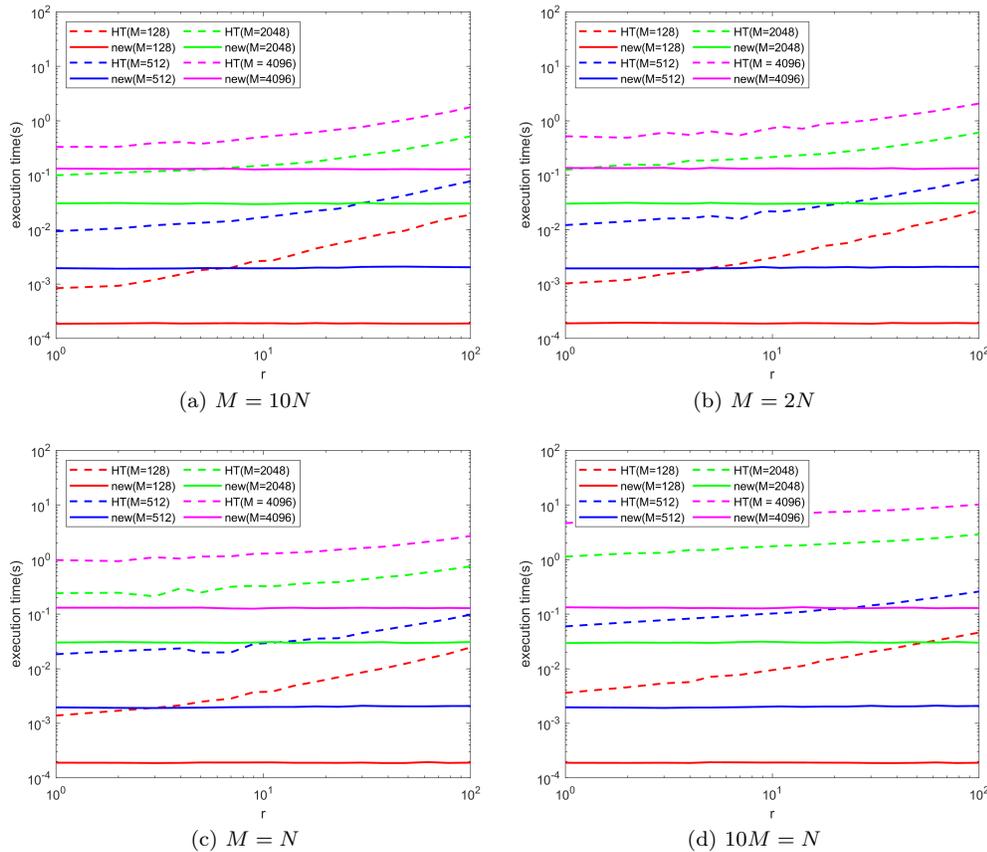


FIG. 7. Comparison between the proposed method (solid) and the Hale–Townsend method (dashed): execution time versus  $r$  for various choices of  $M$  and  $N/M$ . Unlike the Hale–Townsend method, the execution time for the proposed method does not depend on  $r$ .

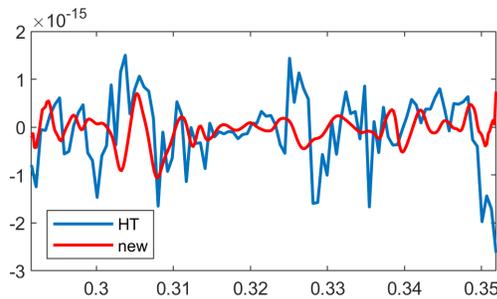


FIG. 8. The error in the computed convolution using the Hale–Townsend method and the proposed method by taking the convolution computed in octuple precision as the exact.

**4.4. Pseudospectrum.** As for solving Fredholm convolution integral equations, the calculation of the pseudospectra of the Fredholm convolution operator relies on the approximation to the operator. Consider the Huygens–Fresnel operator

$$(\mathcal{H}u)(s) = \sqrt{\frac{iF}{\pi}} \int_{-1}^1 e^{-iF(s-t)^2} u(t) dt,$$

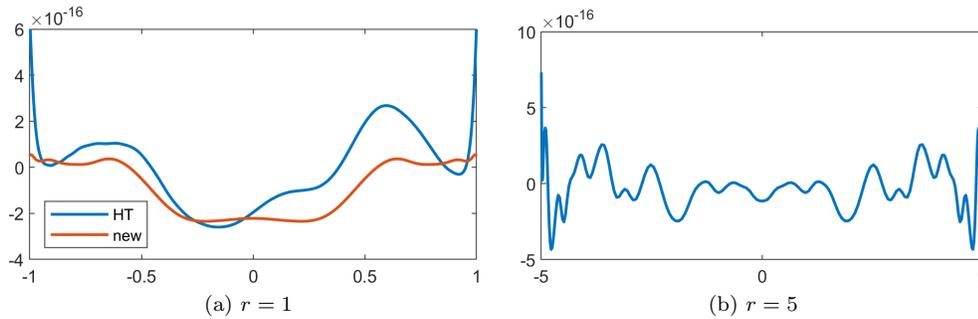
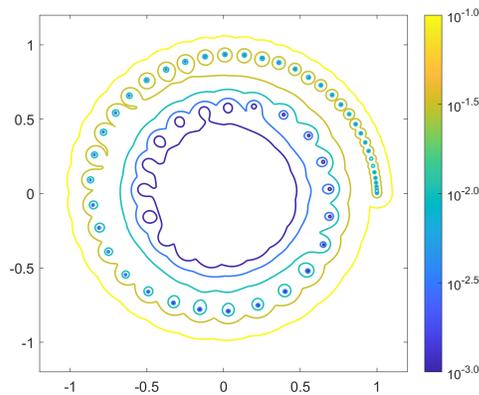


FIG. 9. Error in the computed solution to Love's equation.

FIG. 10.  $\varepsilon$ -pseudospectra of the Huygens-Fresnel operator.

which comes from the modeling of the laser problem [11, section 60]. This integral operator acts on  $u \in L_2([-1, 1])$  and corresponds to stable resonators.

Figure 10 displays the pseudospectra of the Huygens-Fresnel operator with the Fresnel number  $F = 64\pi$ . These boundaries of  $\varepsilon$ -pseudospectra at  $\varepsilon = 10^{-1}, 10^{-1.5}, \dots, 10^{-3}$  are computed using the operator analogue of the Lanczos iteration [2], employing the “solve-then-discretize” strategy. In each iteration of the Lanczos process, we solve in succession two Fredholm convolution integral equations of the second kind  $(z\mathcal{I} - \mathcal{H})v = u$  and  $(z^*\mathcal{I} - \mathcal{H}^*)w = v$ , where the asterisk denotes the complex conjugate for  $z$  and the adjoint of  $\mathcal{H}$ . The proposed spectral approximation plays a key role in solving these two equations and, as a consequence, obtaining the pseudospectra accurately and efficiently. While the contours in Figure 10 are visually indistinguishable from those in the final panel of Figure 60.2 from [11, section 60], which Trefethen and Embree obtained following the conventional “discretize-then-solve” method, the traditional approach is at high risk of *spectral pollution* and *spectral invisibility*. See [2] for more details.

**5. Conclusion.** We have presented a novel  $\mathcal{O}(M^2)$  method for constructing the spectral approximation of a convolution operator of Fredholm type defined by a Legendre series. In contrast to existing methods, the method is applicable to problems where the two convolved Legendre series are defined in intervals with an arbitrary length ratio. When such an approximation is used to evaluate the Fredholm

convolution of two given Legendre series, the computational cost depends only on the length of the kernel, and is independent of the Legendre series being convolved and the interval length ratio. In all the cases that we test, the method is faster than the existing methods. In cases where the Legendre series being convolved is of large degree or the interval length ratio is far greater than 1, the new method is significantly faster. Moreover, such approximations lead to spectral methods for solving Fredholm convolution integral equations, and make the operator analogue of the Krylov subspace methods efficient for computing the eigenvalues and pseudospectra of Fredholm convolution operators.

This work complements the previous ones on the spectral approximation of the Volterra convolution operators [5, 13]. An exciting generalization of this work is to kernels with singularities following a similar approach in [13] and this paper via recurrence relations. This may lead to new methods for the evaluation of fractional integrals and the solution to fractional integral and differential equations, joining the existing spectral methods for these problems, e.g., [14, 1, 4, 8].

**Reproducibility of computational results.** This paper has been awarded the “SIAM Reproducibility Badge: Code and data available” as a recognition that the authors have followed reproducibility principles valued by SISC and the scientific computing community. Code and data that allow readers to reproduce the results in this paper are available at <https://github.com/xiaolin9907/FredConv> and in the supplementary materials (M165253\_SM.zip [local/web 18.9KB]).

**Acknowledgments.** We would like to thank our team members Lu Cheng and Ouyuan Qin for reading an early draft of this paper. We are also grateful to two most constructive anonymous referees for their careful review of the paper as well as their most valuable comments and suggestions.

## REFERENCES

- [1] S. CHEN, J. SHEN, AND L.-L. WANG, *Generalized Jacobi functions and their applications to fractional differential equations*, *Math. Comp.*, 85 (2016), pp. 1603–1638, <https://doi.org/10.1090/mcom3035>.
- [2] K. DENG, X. LIU, AND K. XU, *A Continuous Approach to Computing the Pseudospectra of Linear Operators*, preprint, <https://arxiv.org/abs/2405.03285>, 2024.
- [3] N. HALE, *An ultraspherical spectral method for linear Fredholm and Volterra integro-differential equations of convolution type*, *IMA J. Numer. Anal.*, 39 (2019), pp. 1727–1746, <https://doi.org/10.1093/imanum/dry042>.
- [4] N. HALE AND S. OLVER, *A fast and spectrally convergent algorithm for rational-order fractional integral and differential equations*, *SIAM J. Sci. Comput.*, 40 (2018), pp. A2456–A2491, <https://doi.org/10.1137/16M1104901>.
- [5] N. HALE AND A. TOWNSEND, *An algorithm for the convolution of Legendre series*, *SIAM J. Sci. Comput.*, 36 (2014), pp. A1207–A1220, <https://doi.org/10.1137/140955835>.
- [6] E. R. LOVE, *The electrostatic field of two equal circular co-axial conducting disks*, *Quart. J. Mech. Appl. Math.*, 2 (1949), pp. 428–451, <https://doi.org/10.1093/qjmam/2.4.428>.
- [7] S. OLVER AND A. TOWNSEND, *A fast and well-conditioned spectral method*, *SIAM Rev.*, 55 (2013), pp. 462–489, <https://doi.org/10.1137/120865458>.
- [8] T. PU AND M. FASONDINI, *The numerical solution of fractional integral equations via orthogonal polynomials in fractional powers*, *Adv. Comput. Math.*, 49 (2023), 7, <https://link.springer.com/article/10.1007/s10444-022-10009-9>.
- [9] G. SZEGÖ, *Orthogonal Polynomials*, American Mathematical Society, Providence, RI, 1939.
- [10] L. N. TREFETHEN, *Approximation Theory and Approximation Practice*, SIAM, Philadelphia, 2013.
- [11] L. N. TREFETHEN AND M. EMBREE, *Spectra and Pseudospectra: The Behaviour of Non-normal Matrices and Operators*, Princeton University Press, Princeton, NJ, 2005.

- [12] K. XU, A. P. AUSTIN, AND K. WEI, *A fast algorithm for the convolution of functions with compact support using Fourier extensions*, SIAM J. Sci. Comput., 39 (2017), pp. A3089–A3106, <https://doi.org/10.1137/17M1114764>.
- [13] K. XU AND A. F. LOUREIRO, *Spectral approximation of convolution operators*, SIAM J. Sci. Comput., 40 (2018), pp. A2336–A2355, <https://doi.org/10.1137/17M1149249>.
- [14] M. ZAYERNOURI AND G. E. KARNIADAKIS, *Exponentially accurate spectral and spectral element methods for fractional ODEs*, J. Comput. Phys., 257 (2014), pp. 460–480, <https://doi.org/10.1016/j.jcp.2013.09.039>.