

# A Fast Solver for Boundary Integral Equations of the Modified Helmholtz Equation

Rui Wang · Xiangling Chen

Received: 31 July 2014 / Revised: 4 December 2014 / Accepted: 10 December 2014 /  
Published online: 19 December 2014  
© Springer Science+Business Media New York 2014

**Abstract** The main purpose of this paper is to develop a fast fully discrete Fourier–Galerkin method for solving the boundary integral equations reformulated from the modified Helmholtz equation with boundary conditions. We consider both the nonlinear and the Robin boundary conditions. To tackle the difficulties caused by the two types of boundary conditions, we provide an improved version of the Galerkin method based on the Fourier basis. By employing a matrix compression strategy and efficient numerical quadrature schemes for oscillatory integrals, we obtain fully discrete nonlinear or linear system. Finally, we use the multilevel augmentation method to solve the resulting systems. We point out that the proposed method enjoys an optimal convergence order and a nearly linear computational complexity. The theoretical estimates are confirmed by the performance of this method on several numerical examples.

**Keywords** Modified Helmholtz equation · Fourier–Galerkin methods · Multilevel augmentation methods

**Mathematics Subject Classification** 65M38 · 45L05

## 1 Introduction

In this paper, we are concerned with the modified Helmholtz equation (or the linearized Poisson–Boltzmann equation) with associated boundary conditions, which arises in many areas of science and engineering [4, 26, 27, 30, 40]. The heat equation is of fundamental impor-

---

R. Wang  
School of Mathematics, Jilin University, Changchun 130012, People’s Republic of China  
e-mail: rwang11@jlu.edu.cn

X. Chen (✉)  
College of Mathematics and Computer Science, Hunan Normal University,  
Changsha 410081, Hunan, People’s Republic of China  
e-mail: chenxiangling828@163.com

tance in fluid dynamics, pattern formation, and variational problems. Discretizing the time derivative of the heat equation, which is known as the Rothe's method, gives rise to the modified Helmholtz equation. The Poisson–Boltzmann theory aims at describing the electrostatic interaction of biological systems as long as the electrostatic coupling strength is weak. The linearized Poisson Boltzmann equation has often been used to approximate ionic solutions.

Specifically, we consider the following boundary value problem:

$$\begin{cases} \Delta u(x) - \alpha^2 u(x) = 0, & x \in D, \\ \frac{\partial u(x)}{\partial \mathbf{n}_x} = -g(x, u(x)) + f(x), & x \in \Gamma, \end{cases} \quad (1.1)$$

where  $\alpha$  is a positive constant and  $g, f$  are both given functions. Here, we make the assumption that  $D$  is a simply connected bounded domain in  $\mathbb{R}^2$  and has a  $C^2$  boundary  $\Gamma$  with a parametrization

$$x(t) = (x_1(t), x_2(t)), \quad t \in I := [0, 2\pi].$$

Let  $\mathbf{n}_x$  denote the exterior unit normal to  $\Gamma$  at  $x$ . In this paper, we focus on two types of boundary conditions: the nonlinear boundary condition with  $g$  being nonlinear with respect to  $u$  and the linear boundary condition with  $g(\cdot, u) = u$ , which is called the Robin boundary condition. In any case, we always assume that the boundary value problem (1.1) is uniquely solvable. In fact, the unique solvability can be obtained by imposing suitable assumptions on the functions  $g, f$  and the boundary  $\Gamma$  [3, 34, 41].

Before solving the above boundary value problem, we make a few remarks about our assumptions on the boundary. Although most boundaries of interest in some branches of science are not smooth, there are many applications of smooth boundaries in the fields of biology and medicine and so on. Moreover, the boundaries in practical applications are usually so complex that one can not find a suitable parametrization. For this case, we should consider interpolating the complex boundary by a parametric curve by making use of a sequence of data, which are sampled from the original curve. Then by replacing the original boundary by the interpolating parametric curve, we solve the resulting boundary value problem by the fast algorithm proposed in this paper. Specifically, we suppose that the original curve passes through a sequence of points. We first choose parameter values corresponding to the interpolation points and then find a parametric curve such as polynomial or spline curve, matching position and/or derivatives at the same parameter values.

We note that efforts to address the above issue have been made by number of researchers in the field of computer aided geometric designed. It has been shown that the choice of parametrization has a significant effect on the approximation performance of the resulting interpolating curve. The common choices of parameter include uniform, chordal [1], centripetal [28], Foley–Nielsen [19] parametrizations and parametrizations based on optimization. In [17, 18], the authors study the effect of parametrization on the rate of convergence of parametric curve interpolation. The traditional approach to approximation of parametric curves is to consider the curve as a vector-valued function and then apply a suitable scheme for approximation of functions to each component [16]. To improve on the classical schemes, parametric approximation schemes have been developed, which in the approximation of one component uses information about the other components and so give better accuracy, [13–15, 22, 32, 33, 35].

Among all methods for numerically solving the boundary value problem (1.1), the boundary integral equation approach is one of the most fundamental treatments [31]. It transforms the boundary problem into an integral equation defined on the boundary. To review the boundary integral equation reformulation, we introduce the modified Bessel function of order  $n$  as

$$\begin{aligned}
 K_n(z) := & \frac{1}{2} \sum_{m \in \mathbb{Z}_n} \frac{(-1)^m (n - m - 1)!}{m!} \left(\frac{z}{2}\right)^{2m-n} \\
 & + (-1)^{n+1} \sum_{m \in \mathbb{N}_0} \frac{1}{m!(m+n)!} \left(\log \frac{z}{2} - \frac{1}{2}(\psi(m+n+1) + \psi(m+1))\right) \left(\frac{z}{2}\right)^{2m+n},
 \end{aligned}
 \tag{1.2}$$

where  $\gamma$  is the Euler’s constant and for  $l \in \mathbb{N}$

$$\psi(l) := -\frac{1}{l} - \gamma - \sum_{n \in \mathbb{N}} \left(\frac{1}{l+n} - \frac{1}{n}\right).$$

Here, we let  $\mathbb{N}$  denote the set of all positive integers and set  $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ . For each  $n \in \mathbb{N}$ , we also set  $\mathbb{Z}_n := \{0, 1, \dots, n - 1\}$  and  $\mathbb{Z}_0 = \emptyset$ . Then the fundamental solution of the modified Helmholtz equation is given by

$$\Phi(x, y) := \frac{1}{2\pi} K_0(\alpha|x - y|), \quad x, y \in \mathbb{R}^2.$$

By the potential theory, the solution of the boundary value problem (1.1) can be expressed as

$$u(x) = \int_{\Gamma} \frac{\partial u(y)}{\partial \mathbf{n}_y} \Phi(x, y) ds_y - \int_{\Gamma} u(y) \frac{\partial}{\partial \mathbf{n}_y} \Phi(x, y) ds_y, \quad x \in D.
 \tag{1.3}$$

Letting  $x$  tend to a point on the boundary  $\Gamma$  and using the boundary condition, we obtain the boundary integral equation

$$u(x) + 2 \int_{\Gamma} u(y) \frac{\partial}{\partial \mathbf{n}_y} \Phi(x, y) ds_y + 2 \int_{\Gamma} g(y, u(y)) \Phi(x, y) ds_y = 2 \int_{\Gamma} f(y) \Phi(x, y) ds_y, \quad x \in \Gamma.
 \tag{1.4}$$

Once the solution of the boundary integral equation is achieved, one can get the normal derivative on the boundary and then obtain the solution of (1.1) by calculating the expression (1.3). This motivates one to solve the boundary integral equation (1.4) numerically. We note that Eq. (1.4) is a linear boundary integral equation for the Robin boundary condition and a nonlinear one for the nonlinear boundary condition.

In the literature, there have been various numerical methods developed for solving the boundary integral equations reformulated from the modified Helmholtz equation with linear or nonlinear boundary conditions. A Galerkin method with one-periodic B-splines as basis functions [36] and a spectral collocation method [37] were proposed for the mixed boundary value problem. Fast multipole-accelerated integral equation methods were discussed in [7, 26, 27] for solving the modified Helmholtz equation with different kinds of linear boundary conditions. For the nonlinear boundary value problem, the mechanical quadrature methods and extrapolation were developed in [8]. In this paper, we shall consider the Galerkin method using the Fourier basis for solving the boundary integral equation (1.4). Our interest in the Fourier–Galerkin method is motivated by the decomposition of the weakly singular integral operator arising in the integral equation (1.4). To be more specific, we split the operator into the form  $\mathcal{A} + \mathcal{B}$ , where the operator  $\mathcal{A}$  carries the main singularity characteristic of the original operator and the operator  $\mathcal{B}$ , however, is a compact operator with a smooth kernel. It is worth to point out that the operator  $\mathcal{A}$  has the Fourier basis functions as its eigenfunctions. Taking this advantage, we get the matrix representation of  $\mathcal{A}$  under the Fourier basis as a diagonal matrix with the entries being obtained directly.

In fact, solving the discrete system resulting from the Fourier–Galerkin method requires large computational costs, which are caused for two reasons. One reason is that we have to face computing double singular integrals resulting from the boundary condition. In the case of nonlinear boundary condition, these integrals arise from evaluating and updating the Jacobian matrix in each iteration step of Newton’s method for solving the nonlinear system. And in the case of the Robin boundary condition, setting up the coefficient matrix in the resulting linear system also leads to these integrals. To overcome this difficulty, we first adopt a technique, which was used in [9, 10] for solving the Laplace’s equation with nonlinear boundary condition. Specifically, we project the term, involving the boundary function  $g$ , onto the approximate subspace. By doing so, we transform the evaluation of double singular integrals into establishing the matrix representations of operators and multiplying these matrices by vectors. The matrix representation of the operator  $\mathcal{A}$  has a simple structure and needs no computational efforts to set up. However, the matrix representation of other operators are usually dense and so we shall apply appropriate matrix truncation strategy. The truncation strategy allows us not only to compress the dense matrices to sparse ones but also to retain enough entries to represent critical information encoded in the original matrices. Because of the smoothness of the kernel functions, we shall take the hyperbolic cross compression proposed in [6] and employ the numerical quadrature formula developed in [24] to compute the nonzero entries in resulting sparse matrices. Moreover, the Robin boundary condition also need us to set up another dense matrix with the entries involving single oscillation integrations. We shall adopt the trapezoid quadrature rule and the fast Fourier transform to build this matrix. The other reason for the large computational costs is to invert the linear or nonlinear operator in the entire approximate subspace, which usually has a large dimension so that one can get good approximation accuracy. The multilevel augmentation method developed in [9, 11, 12] can help us to deal with this issue efficiently. It is worth to point out that the fast algorithm based upon the above techniques preserves the optimal convergence order and at the same time enjoys a nearly linear computational complexity.

This paper is organized as follows. We describe the Fourier–Galerkin method for solving the resulting boundary integral equation in the next section. Through introducing an additional projection operator, we present an improved version of the Fourier–Galerkin method. In Sect. 3, we establish the fast solvers for the linear and nonlinear system resulting from the improved Fourier–Galerkin method, respectively. We also give the convergence analysis as well as the computational complexity estimate of the proposed algorithm for the linear system. The analysis of the algorithm for the nonlinear system can be obtained by arguments similar to those in [10]. In Sect. 4, to demonstrate the computational efficiency and the approximate accuracy of the proposed algorithms, we present numerical examples for both the nonlinear boundary condition and the Robin boundary condition.

## 2 The Fourier–Galerkin Method for Boundary Integral Equations

In this section, we describe the Fourier–Galerkin method for solving the boundary integral equation (1.4). We also provide an improved version of this method, which is more suitable for solving the nonlinear boundary value problem as well as the Robin boundary value problem.

To this end, we begin with presenting Eq. (1.4) in an operator form. With respect to the parametrization of the boundary  $\Gamma$ :

$$x(t) = (x_1(t), x_2(t)), \quad t \in I = [0, 2\pi],$$

we set  $r(s, t) := |x(s) - x(t)|$ , for any  $s, t \in I$ . Let  $\mathbb{Z}$  be the set of all integers. By using the modified Bessel functions defined as in (1.2), we define two kernels at  $s, t \in I$  by

$$\mathcal{L}(s, t) := \frac{1}{\pi} K_0(\alpha r(s, t))$$

and

$$\mathcal{K}(s, t) := \begin{cases} \frac{\alpha K_1(\alpha r(s, t))}{r(s, t)} \frac{x_2'(t)(x_1(t) - x_1(s)) - x_1'(t)(x_2(t) - x_2(s))}{\pi}, & s - t \neq 2k\pi, k \in \mathbb{Z}, \\ \frac{1}{\pi} \frac{x_1'(t)x_2''(t) - x_2'(t)x_1''(t)}{2(x_1'(t)^2 + x_2'(t)^2)}, & \text{otherwise.} \end{cases}$$

Then by introducing three functions on  $I$  as

$$u(t) := u(x(t)), \quad g(t, u(t)) := g(x(t), u(x(t))), \quad f(t) := f(x(t)), \quad t \in I,$$

we can now rewrite Eq. (1.4) as

$$u(s) - \int_I \mathcal{K}(s, t)u(t)dt + \int_I \mathcal{L}(s, t)g(t, u(t))|x'(t)|dt = \int_I \mathcal{L}(s, t)f(t)|x'(t)|dt, \quad s \in I. \tag{2.1}$$

Associated with the kernels  $\mathcal{K}$  and  $\mathcal{L}$ , we define two linear integral operators respectively, by

$$(\mathcal{K}u)(s) := \int_I \mathcal{K}(s, t)u(t)dt, \quad s \in I,$$

and

$$(\mathcal{L}u)(s) := \int_I \mathcal{L}(s, t)u(t)dt, \quad s \in I.$$

Properties of the operators  $\mathcal{K}$  and  $\mathcal{L}$  are inherited from those of the kernels  $\mathcal{K}$  and  $\mathcal{L}$ , respectively. Specifically, observing from the definition, we have that the kernel  $\mathcal{K}$  is of  $C^{\mu-2}$  when the boundary  $\Gamma$  is of  $C^\mu$ ,  $\mu \geq 2$ . The smoothness of the kernel  $\mathcal{K}$  leads directly to the compactness of the operator  $\mathcal{K}$ . By Eq. (1.2) for  $n = 0$ , we get

$$K_0(z) = -\log \frac{z}{2} - \gamma, \quad z \rightarrow 0,$$

which shows that  $K_0$  has a logarithmic singularity at  $z = 0$ , so is the kernel  $\mathcal{L}$ . Thus, the operator  $\mathcal{L}$  is a weakly singular integral operator. We next decompose the integral operator  $\mathcal{L}$  into the sum of two integral operators, where one carries the main singularity characteristic of  $\mathcal{L}$  and the other is a compact operator with a smooth kernel. This can be done by splitting the kernel  $\mathcal{L}$  into the form

$$\mathcal{L}(s, t) = \mathcal{A}(s, t) + \mathcal{B}(s, t), \quad s, t \in I, \tag{2.2}$$

where for any  $s, t \in I$

$$\mathcal{A}(s, t) := -\frac{1}{\pi} \log \left| 2e^{-1/2} \sin \frac{s-t}{2} \right|,$$

and

$$\mathcal{B}(s, t) = \begin{cases} \mathcal{L}(s, t) - \mathcal{A}(s, t), & t - s \neq 2k\pi, k \in \mathbb{Z}, \\ -\frac{1}{\pi} \log \frac{\alpha e^{1/2} |x'(t)|}{2} - \frac{\gamma}{\pi}, & \text{otherwise,} \end{cases}$$

and defining two operators by

$$(\mathcal{A}u)(s) := \int_I \mathcal{A}(s, t)u(t)dt, \quad (\mathcal{B}u)(s) := \int_I \mathcal{B}(s, t)u(t)dt, \quad s \in I.$$

Among all the decompositions of the operator  $\mathcal{L}$ , Eq. (2.2) has two advantages. Firstly, the singular operator  $\mathcal{A}$  has the Fourier basis functions as its eigenfunctions. Specifically, for any  $k \in \mathbb{Z}$  there holds  $\mathcal{A}(e^{ik(\cdot)}) = \lambda_k e^{ik(\cdot)}$  with  $\lambda_k = 1/\max\{1, |k|\}$ . Then its matrix representation under the Fourier basis is diagonal and the entries on the diagonal can also be obtained exactly. Secondly, although the matrix representation of the compact operator  $\mathcal{B}$  under the Fourier basis is usually dense, it can be compressed to a sparse matrix without loss of critical information encoded in the matrix. We note that the compression strategy can also be used to deal with the matrix representation of the compact operator  $\mathcal{K}$  under the same basis.

We also define an operator by

$$(\Psi u)(t) := g(t, u(t))|x'(t)|, \quad t \in I, \tag{2.3}$$

which is nonlinear for the case that  $g(x, u)$  is nonlinear with respect to  $u$  and is linear for the case that  $g(x, u) = u$ . With the operators defined above, Eq. (2.1) is represented in an operator form:

$$u - \mathcal{K}u + (\mathcal{A} + \mathcal{B})\Psi u = h, \tag{2.4}$$

where  $h := (\mathcal{A} + \mathcal{B})(f|x'|)$ .

The features of the kernels motivate us to use the Galerkin method based on the Fourier basis for solving the kernel equation (2.4). We now turn to describing this method. For each  $n \in \mathbb{N}$ , we introduce a finite dimensional subspace of  $L^2(I)$ , spanned by the Fourier basis  $e_k(t) := e^{ikt}/\sqrt{2\pi}$ ,  $t \in I, k \in \mathbb{Z}$ , as

$$\mathbb{X}_n := \text{span}\{e_k : |k| \in \mathbb{Z}_n\}.$$

Let  $\mathcal{P}_n$  be the orthogonal projection from  $L^2(I)$  onto  $\mathbb{X}_n$ . The Fourier–Galerkin method for solving Eq. (2.4) is described as finding  $u_n$  in  $\mathbb{X}_n$  such that

$$u_n - \mathcal{P}_n \mathcal{K}u_n + \mathcal{P}_n(\mathcal{A} + \mathcal{B})\Psi u_n = \mathcal{P}_n h. \tag{2.5}$$

To describe the unique solvability and the error analysis of Eq. (2.5), we introduce the Fréchet derivative of the operator  $\mathcal{L}\Psi$  at  $u$  as an integral operator defined by

$$((\mathcal{L}\Psi)'(u)h)(s) := \int_I \mathcal{L}(s, t)\Psi'(u)h(t)dt.$$

Under certain hypotheses imposed on the linear operator  $\mathcal{K} - (\mathcal{L}\Psi)'(u)$ , we obtain in the following theorem the existence of the solution of Eq. (2.5) and its convergence property. We note that the theorem can be proved by making use of Theorem 3.3.2 in [11] for the case of the Robin boundary condition and Theorem 2 in [38,39] for the case of the nonlinear boundary condition.

**Theorem 2.1** *Suppose that  $u \in L^2(I)$  is an isolated solution of (2.4) and 1 is not an eigenvalue of the linear operator  $\mathcal{K} - (\mathcal{L}\Psi)'(u)$ . Then for sufficiently large  $n$ , Eq. (2.5) has a unique solution  $u_n \in B(u, \delta)$  for some  $\delta > 0$  and there exist positive constants  $c_1$  and  $c_2$  such that*

$$c_1 \|u - \mathcal{P}_n u\| \leq \|u - u_n\| \leq c_2 \|u - \mathcal{P}_n u\|.$$

We next consider the convergence order of the solution of Eq. (2.5). For this purpose, we introduce the Sobolev spaces of  $2\pi$ -periodic functions, which are subspaces of  $L^2(I)$  and are required for their elements a certain decay of their Fourier coefficients. To be more specific, for each  $\mu \geq 0$ ,

$$H^\mu(I) := \left\{ \phi \in L^2(I) : \sum_{k \in \mathbb{Z}} (1 + k^2)^\mu |\phi_k|^2 < \infty \right\},$$

where  $\phi_k$  is the  $k$ -th Fourier coefficient of  $\phi$  defined by  $\phi_k := \int_I \phi(x) e_k(x) dx$ . The space  $H^\mu(I)$  is a Hilbert space with the inner product defined for  $\phi, \psi \in H^\mu(I)$  by

$$\langle \phi, \psi \rangle_\mu := \sum_{k \in \mathbb{Z}} (1 + k^2)^\mu \phi_k \overline{\psi_k},$$

and its norm is given by  $\|\phi\|_\mu := \langle \phi, \phi \rangle_\mu^{\frac{1}{2}}$ . We note that the space  $H^0(I)$  coincides with  $L^2(I)$ . It is well-known [2,25] that for any  $\phi \in H^\mu(I)$ , there holds

$$\|\phi - \mathcal{P}_n \phi\| \leq n^{-\mu} \|\phi\|_\mu. \tag{2.6}$$

With the help of estimate (2.6), Theorem 2.1 shows that the Fourier–Galerkin method has convergence of the optimal order.

**Corollary 2.2** *If the solution  $u \in H^\mu(I)$ ,  $\mu \geq 0$ , there exists a constant  $c > 0$  such that*

$$\|u - u_n\| \leq cn^{-\mu} \|u\|_\mu.$$

In terms of the Fourier basis, the Fourier Galerkin method (2.5) is equivalent to a system of equations. Specifically, it suffices to seek a function  $u_n \in \mathbb{X}_n$  with the form

$$u_n(t) = \sum_{|l| \in \mathbb{Z}_n} a_l e_l(t), \quad t \in I,$$

satisfying the system

$$\langle u_n - \mathcal{K} u_n + (\mathcal{A} + \mathcal{B}) \Psi u_n, e_k \rangle = \langle h, e_k \rangle, \quad |k| \in \mathbb{Z}_n. \tag{2.7}$$

We note that if the operator  $\Psi$  defined in (2.3) is linear, so is the system (2.7). Otherwise, the system is nonlinear, which is usually solved by Newton’s method. However, for solving any kind of systems, we have to face computing the quantities

$$\langle e_l, e_k \rangle - \langle \mathcal{K} e_l, e_k \rangle + \langle (\mathcal{A} + \mathcal{B}) \Psi'(u_n^*) e_l, e_k \rangle, \quad |k|, |l| \in \mathbb{Z}_n, \tag{2.8}$$

where  $u_n^*$  denotes  $u_n$  or  $u_n^{(m)}$  according to the type of the operator  $\Psi$ . To be more specific, if the operator  $\Psi$  is linear, then we choose  $u_n^* = u_n$  and the items in (2.8) arise from establishing the Galerkin matrix. If the operator  $\Psi$  is nonlinear and  $u_n^{(m)}$  denotes the solution at the  $m$ th iteration of Newton’s method, then we choose  $u_n^* = u_n^{(m)}$  and the items in (2.8) coincide with the entries of the Jacobian matrix at the  $m$ th iteration. Since the items involve double integrals, solving (2.7) demands a large amount of computational efforts.

We observe from (2.8) that the first item need not be evaluated because of the orthogonality of  $e_k$ ’s. The discussion of the evaluation of the second item will be postponed until the next section, where the quantities  $\langle \mathcal{K} e_l, e_k \rangle, |k|, |l| \in \mathbb{Z}_n$ , can be evaluated efficiently by compressing the matrix representation of the operator  $\mathcal{K}$ . In this section, we are only interested in the third item which carries the main computational difficulties. Motivated by the fact that the Fourier basis functions are the eigenfunctions of the operator  $\mathcal{A}$ , we consider projecting

the item  $\Psi'(u_n^*)$  onto the subspace  $\mathbb{X}_n$ . By doing so, we will transform the computation of double singular integrals into the establishment of the matrix representations of  $\mathcal{A}$  and  $\mathcal{B}$  and the multiplication of matrices and vectors.

To implement the above idea, we introduce the improved Fourier–Galerkin method for solving (2.4), which finds  $\hat{u}_n \in \mathbb{X}_n$  such that

$$\hat{u}_n - \mathcal{P}_n \mathcal{K} \hat{u}_n + \mathcal{P}_n (\mathcal{A} + \mathcal{B}) \mathcal{P}_n \Psi \hat{u}_n = \mathcal{P}_n h. \tag{2.9}$$

The following theorem is concerned with the unique solvability and the convergence order of the improved Fourier–Galerkin method. This result can be proved by arguments similar to those for Theorem 2.1 and Corollary 2.2.

**Theorem 2.3** *Suppose that  $u \in L^2(I)$  is an isolated solution of (2.4) and 1 is not an eigenvalue of the linear operator  $\mathcal{K} - (\mathcal{L}\Psi)'(u)$ . Then for sufficiently large  $n$ , Eq. (2.9) has a unique solution  $\hat{u}_n \in B(u, \delta)$  for some  $\delta > 0$  and there exists a constant  $c > 0$  such that*

$$\|u - \hat{u}_n\| \leq c \|u - \mathcal{P}_n u\|.$$

Furthermore, if the solution  $u \in H^\mu(I)$ ,  $\mu \geq 0$ , then there holds

$$\|u - \hat{u}_n\| \leq cn^{-\mu} \|u\|_\mu.$$

### 3 A Fast Solver for the Improved Fourier–Galerkin Method

The main purpose of this section is to develop a fast fully discrete algorithm for solving Eq. (2.9). This will be done by employing a matrix truncation strategy, quadrature formulas for single and double oscillatory integrals and the multilevel augmentation method.

#### 3.1 Fast Algorithm for Solving the Nonlinear System

When we consider the boundary value problem (1.1) with the nonlinear boundary condition, Eq. (2.9) is equivalent to the following nonlinear system:

$$\langle \hat{u}_n - \mathcal{K} \hat{u}_n + (\mathcal{A} + \mathcal{B}) \mathcal{P}_n \Psi \hat{u}_n, e_k \rangle = \langle h, e_k \rangle, \quad |k| \in \mathbb{Z}_n, \tag{3.1}$$

with  $\hat{u}_n := \sum_{|l| \in \mathbb{Z}_n} \hat{a}_l e_l$ . To rewrite (3.1) in a matrix form, we now present the matrix representations of the operators  $\mathcal{K}$ ,  $\mathcal{A}$  and  $\mathcal{B}$  under the Fourier basis. To this end, we require some necessary notations. For each  $n \in \mathbb{N}$ , we set  $\mathbb{Z}_n^+ := \{1, 2, \dots, n - 1\}$ . We denote by  $\mathbf{w} = [w_k, w_{-k} : k \in \mathbb{Z}_n^+]$  the vector  $\mathbf{w} = [w_1, w_{-1}, \dots, w_{n-1}, w_{-(n-1)}]$  and by  $\mathbf{w} = [w_k, w_{-k} : k \in \mathbb{Z}_n]$  the vector  $\mathbf{w} = [w_0, w_1, w_{-1}, \dots, w_{n-1}, w_{-(n-1)}]$ . For each  $|k|, |l| \in \mathbb{Z}_n$ , we set

$$K_{k,l} := \int_{I^2} \mathcal{K}(s, t) e_k(s) e_l(t) ds dt$$

and introduce the square matrix of order 2

$$\mathbf{K}_{k,l} := \begin{bmatrix} K_{-k,l} & K_{-k,-l} \\ K_{k,l} & K_{k,-l} \end{bmatrix}.$$

By defining the matrix blocks

$$\mathbf{K} := [\mathbf{K}_{k,l} : k, l \in \mathbb{Z}_n^+], \quad \mathbf{K}' := [K_{0,0}], \quad \mathbf{K}'' := [K_{0,k}, K_{0,-k} : k \in \mathbb{Z}_n^+],$$



$$\mathbf{K}''' := [K_{-k,0}, K_{k,0} : k \in \mathbb{Z}_n^+]^T,$$

we build the matrix representation of the operator  $\mathcal{K}$  as

$$\mathbf{K}_n := \begin{bmatrix} \mathbf{K}' & \mathbf{K}'' \\ \mathbf{K}''' & \mathbf{K} \end{bmatrix}.$$

In a similar manner, we can define the matrices  $\mathbf{A}_n$  and  $\mathbf{B}_n$  as the matrix representations of the operators  $\mathcal{A}$  and  $\mathcal{B}$ , respectively. The projection of  $\Psi \hat{u}_n$  can be expressed in the form

$$\mathcal{P}_n \Psi \hat{u}_n(t) = \sum_{|k| \in \mathbb{Z}_n} \hat{b}_k e_k(t), \quad t \in I,$$

and the coefficients  $\hat{b}_k, |k| \in \mathbb{Z}_n$ , are determined by  $\hat{b}_k = \langle \Psi \hat{u}_n, e_k \rangle$ . Associated with  $\hat{u}_n = \sum_{|l| \in \mathbb{Z}_n} \hat{a}_l e_l$ , we introduce two vectors

$$\hat{\mathbf{u}}_n := [\hat{a}_k, \hat{a}_{-k} : k \in \mathbb{Z}_n]^T \text{ and } \hat{\mathbf{v}}_n := [\hat{b}_k, \hat{b}_{-k} : k \in \mathbb{Z}_n]^T.$$

By setting  $h_k := \int_I h(t) e_k(t) dt, |k| \in \mathbb{Z}_n$ , we also define the vector

$$\mathbf{h}_n := [h_{-k}, h_k : k \in \mathbb{Z}_n]^T.$$

Let  $\mathbf{I}_n$  denote the identity matrix of order  $2n - 1$ . By using the matrix and vector notations, Eq. (3.1) takes the equivalent matrix form

$$(\mathbf{I}_n - \mathbf{K}_n) \hat{\mathbf{u}}_n + (\mathbf{A}_n + \mathbf{B}_n) \hat{\mathbf{v}}_n = \mathbf{h}_n. \tag{3.2}$$

To develop a fast solver for (3.2), we shall apply appropriate matrix truncation strategy to the dense matrices  $\mathbf{K}_n$  and  $\mathbf{B}_n$ . Using this strategy, we not only compress a dense matrix to a sparse one but also retain enough entries to represent critical information encoded in the original matrix. It is the smoothness of the kernel functions  $\mathcal{K}$  and  $\mathcal{B}$  that allows us to carry out the hyperbolic cross compression technique to the matrices  $\mathbf{K}_n$  and  $\mathbf{B}_n$ . In fact, this type of compression comes from the hyperbolic cross sparse approximation of functions by the Fourier basis, which remove the unnecessary high frequency terms and retain only low frequency and necessary high frequency terms in the Fourier expansion. Specifically, we introduce the index set  $\mathbb{L}_n := \{[k, l] \in \mathbb{Z}_n^2 : kl \leq n\}$  and define the truncation matrix of  $\mathbf{K}_n$  by setting

$$\tilde{\mathbf{K}}_{k,l} := \begin{cases} \mathbf{K}_{k,l}, & [k, l] \in \mathbb{L}_n \cap (\mathbb{Z}_n^+)^2, \\ 0_{2 \times 2}, & \text{otherwise,} \end{cases} \quad \tilde{\mathbf{K}} := [\tilde{\mathbf{K}}_{k,l} : k, l \in \mathbb{Z}_n^+],$$

and letting

$$\tilde{\mathbf{K}}_n := \begin{bmatrix} \mathbf{K}' & \mathbf{K}'' \\ \mathbf{K}''' & \tilde{\mathbf{K}} \end{bmatrix}.$$

The dense matrix  $\mathbf{B}_n$  can be similarly handled. If we use  $\mathcal{N}(\mathbf{M})$  to denote the number of nonzero entries in the matrix  $\mathbf{M}$ , an estimation in [6] gives that  $\mathcal{N}(\tilde{\mathbf{K}}_n) = \mathcal{N}(\tilde{\mathbf{B}}_n) = \mathcal{O}(n \log n)$ , which shows the sparsity of the matrices  $\tilde{\mathbf{K}}_n$  and  $\tilde{\mathbf{B}}_n$ . This truncation strategy admits a fast algorithm to set up the matrix representations and to solve (3.2) subsequently.

Numerical implementation of the fast method requires efficient computation of the nonzero entries of the matrices  $\tilde{\mathbf{K}}_n$  and  $\tilde{\mathbf{B}}_n$ . These quantities involve oscillatory integrals, which should be treated by efficient quadrature formula. For recent development of numerical integration of oscillatory integrals, see [20, 21, 23, 29]. We will use an efficient quadrature formula developed

in [23] for computing the nonzero entries of the truncated matrix  $\tilde{\mathbf{K}}_n$ , which are defined by the integrals

$$K_{k,l} := \int_{J^2} \mathcal{K}(s, t) e_k(s) e_l(t) ds dt, \quad |k|, |l| \in \mathbb{L}_n.$$

In this method, one first constructs a multi-scale Lagrange interpolation on sparse grids as a good approximation of the kernel function  $\mathcal{K}$ . Replacing the kernel function  $\mathcal{K}$  by the multi-scale Lagrange interpolation, one can obtain a quadrature formula of the original integrations. With the help of the product integration method, the resulting oscillation integrations can be evaluated exactly. Readers are referred to [23, 24] for a detailed description of the algorithm. In the following, we denote by  $\tilde{\mathbf{K}}_{n,N}$  the matrix  $\tilde{\mathbf{K}}_n$  with  $K_{k,l}$  being evaluated by the quadrature formula in [23], where  $N$  is the number of partition level of the multi-scale Lagrange interpolation for the kernel  $\mathcal{K}$ . We deal with the entries of the matrix  $\tilde{\mathbf{B}}_n$  in a similar manner and obtain the matrix  $\tilde{\mathbf{B}}_{n,N}$ .

Up to now, by replacing the dense matrices  $\mathbf{K}_n$  and  $\mathbf{B}_n$  by  $\tilde{\mathbf{K}}_{n,N}$  and  $\tilde{\mathbf{B}}_{n,N}$ , respectively, we obtain the fully discrete truncated nonlinear system

$$(\mathbf{I}_n - \tilde{\mathbf{K}}_{n,N})\tilde{\mathbf{u}}_{n,N} + (\mathbf{A}_n + \tilde{\mathbf{B}}_{n,N})\tilde{\mathbf{v}}_{n,N} = \mathbf{h}_n, \tag{3.3}$$

where  $\tilde{\mathbf{u}}_{n,N} := [\tilde{a}_{k,N}, \tilde{a}_{-k,N} : k \in \mathbb{Z}_n]^T$  is the corresponding solution to be determined and  $\tilde{\mathbf{v}}_{n,N} := [\tilde{b}_{k,N}, \tilde{b}_{-k,N} : k \in \mathbb{Z}_n]^T$  with

$$\tilde{b}_{k,N} := \left\langle \Psi \left( \sum_{|l| \in \mathbb{Z}_n} \tilde{a}_{l,N} e_l \right), e_k \right\rangle, \quad |k| \in \mathbb{Z}_n.$$

Finally, we consider solving the nonlinear system (3.3) by the multilevel augmentation method. The key idea of the multilevel augmentation method is to solve the system by inverting the nonlinear or linear operator in a subspace of a much lower dimension with a compensation of high frequency by matrix-vector multiplications. To apply this idea to Eq. (3.3), we let  $k$  be a fixed positive integer and denote the initial level of approximation. Assume that we solve (3.3) with  $n := 2^{k+m}$ ,  $m \in \mathbb{N}$ . We write the matrix  $\tilde{\mathbf{K}}_{n,N}$  in the block form

$$\tilde{\mathbf{K}}_{n,N} = \begin{bmatrix} \overline{K}_{00} & \overline{K}_{01} & \cdots & \overline{K}_{0m} \\ \overline{K}_{10} & \overline{K}_{11} & \cdots & \overline{K}_{1m} \\ \cdots & \cdots & \cdots & \cdots \\ \overline{K}_{m0} & \overline{K}_{m1} & \cdots & \overline{K}_{mm} \end{bmatrix},$$

where  $\overline{K}_{00}$  has order  $(2^{k+1} - 1) \times (2^{k+1} - 1)$ ,  $\overline{K}_{0q}$  has order  $(2^{k+1} - 1) \times 2^{k+q}$ ,  $\overline{K}_{q0}$  has order  $2^{k+q} \times (2^{k+1} - 1)$  for  $q = 1, 2, \dots, m$  and  $\overline{K}_{pq}$  has order  $2^{k+p} \times 2^{k+q}$  for  $p, q = 1, 2, \dots, m$ . For each  $l = 0, 1, \dots, m$ , we denote by  $\tilde{\mathbf{K}}_{k,l,N}$  the submatrix of  $\tilde{\mathbf{K}}_{n,N}$  that has the form

$$\tilde{\mathbf{K}}_{k,l,N} = \begin{bmatrix} \overline{K}_{00} & \overline{K}_{01} & \cdots & \overline{K}_{0l} \\ \overline{K}_{10} & \overline{K}_{11} & \cdots & \overline{K}_{1l} \\ \cdots & \cdots & \cdots & \cdots \\ \overline{K}_{l0} & \overline{K}_{l1} & \cdots & \overline{K}_{ll} \end{bmatrix},$$

and split  $\tilde{\mathbf{K}}_{k,l,N}$  into its “low frequency” and “high frequency” components:

$$\tilde{\mathbf{K}}_{k,l,N}^L = [\overline{K}_{00} \quad \overline{K}_{01} \quad \cdots \quad \overline{K}_{0l}] \quad \text{and} \quad \tilde{\mathbf{K}}_{k,l,N}^H = \begin{bmatrix} \overline{K}_{10} & \overline{K}_{11} & \cdots & \overline{K}_{1l} \\ \cdots & \cdots & \cdots & \cdots \\ \overline{K}_{l0} & \overline{K}_{l1} & \cdots & \overline{K}_{ll} \end{bmatrix}.$$

By the same way, we also introduce the submatrices and corresponding “low frequency” components and “high frequency” components for the matrices  $\mathbf{I}_n, \mathbf{A}_n$  and  $\tilde{\mathbf{B}}_{n,N}$ . With the help of these notations, we establish the following fast algorithm for solving the nonlinear system (3.3).

**Algorithm 1: The fast multilevel augmentation method for the nonlinear system**

Let  $k$  be a fixed positive integer. Set  $\mathbf{h}_{2^{k+m}} := [\langle h, e_j \rangle, \langle h, e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+m}}]^T$ .

**Step 1:** According to Newton’s method, solve the nonlinear system

$$(\mathbf{I}_{2^k} - \tilde{\mathbf{K}}_{k,0,N})\mathbf{u}_{k,0,N} + (\mathbf{A}_{2^k} + \tilde{\mathbf{B}}_{k,0,N})\mathbf{v}_{k,0,N} = \mathbf{h}_{2^k},$$

for the solution  $\mathbf{u}_{k,0,N} = [(\mathbf{u}_{k,0,N})_j, (\mathbf{u}_{k,0,N})_{-j} : j \in \mathbb{Z}_{2^k}]^T$ , where

$$\mathbf{v}_{k,0,N} := [\langle \Psi(u_{k,0,N}), e_j \rangle, \langle \Psi(u_{k,0,N}), e_{-j} \rangle : j \in \mathbb{Z}_{2^k}]^T$$

and  $u_{k,0,N} := \sum_{|j| \in \mathbb{Z}_{2^k}} (\mathbf{u}_{k,0,N})_j e_j$ . Set  $l := 1$ .

**Step 2:** Set  $\mathbf{h}_{k,l} := [\langle h, e_j \rangle, \langle h, e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+l}} \setminus \mathbb{Z}_{2^k}]^T$ .

- Compute  $\hat{\mathbf{v}}_{k,l,N} = [\langle \Psi(u_{k,l-1,N}), e_j \rangle, \langle \Psi(u_{k,l-1,N}), e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+l}}]^T$ .
- Augment the matrices  $\tilde{\mathbf{K}}_{k,l-1,N}^H, \mathbf{A}_{k,l-1}^H$  and  $\tilde{\mathbf{B}}_{k,l-1,N}^H$  to form  $\tilde{\mathbf{K}}_{k,l,N}^H, \mathbf{A}_{k,l}^H$  and  $\tilde{\mathbf{B}}_{k,l,N}^H$ , respectively.
- Augment the vector  $\mathbf{u}_{k,l-1,N}$  by setting  $\hat{\mathbf{u}}_{k,l,N} = \begin{bmatrix} \mathbf{u}_{k,l-1,N} \\ \mathbf{0}_{2^{k+l} \times 1} \end{bmatrix}$ .
- Compute

$$\mathbf{u}_{k,l,N}^H := \mathbf{h}_{k,l} + \tilde{\mathbf{K}}_{k,l,N}^H \hat{\mathbf{u}}_{k,l,N} - (\mathbf{A}_{k,l}^H + \tilde{\mathbf{B}}_{k,l,N}^H) \hat{\mathbf{v}}_{k,l,N}.$$

Let  $u_{k,l,N}^H := \sum_{|j| \in \mathbb{Z}_{2^{k+l}} \setminus \mathbb{Z}_{2^k}} (\mathbf{u}_{k,l,N}^H)_j e_j$ .

**Step 3:** Augment the matrices  $\tilde{\mathbf{K}}_{k,l-1,N}^L, \mathbf{A}_{k,l-1}^L$  and  $\tilde{\mathbf{B}}_{k,l-1,N}^L$  to form  $\tilde{\mathbf{K}}_{k,l,N}^L, \mathbf{A}_{k,l}^L$  and  $\tilde{\mathbf{B}}_{k,l,N}^L$ , respectively. According to Newton’s method, solve the nonlinear system

$$(\mathbf{I}_{k,l}^L - \tilde{\mathbf{K}}_{k,l,N}^L) \begin{bmatrix} \mathbf{u}_{k,l,N}^L \\ \mathbf{u}_{k,l,N}^H \end{bmatrix} + (\mathbf{A}_{k,l}^L + \tilde{\mathbf{B}}_{k,l,N}^L) \bar{\mathbf{v}}_{k,l,N} = \mathbf{h}_{2^k}$$

for the solution  $\mathbf{u}_{k,l,N}^L = [(\mathbf{u}_{k,l,N}^L)_j, (\mathbf{u}_{k,l,N}^L)_{-j} : j \in \mathbb{Z}_{2^k}]^T$ , where

$$\bar{\mathbf{v}}_{k,l,N} := [\langle \Psi(u_{k,l,N}^L + u_{k,l,N}^H), e_j \rangle, \langle \Psi(u_{k,l,N}^L + u_{k,l,N}^H), e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+l}}]^T$$

and  $u_{k,l,N}^L := \sum_{|j| \in \mathbb{Z}_{2^k}} (\mathbf{u}_{k,l,N}^L)_j e_j$ . Define  $\mathbf{u}_{k,l,N} := \begin{bmatrix} \mathbf{u}_{k,l,N}^L \\ \mathbf{u}_{k,l,N}^H \end{bmatrix}$  and  $u_{k,l,N} :=$

$$u_{k,l,N}^L + u_{k,l,N}^H.$$

**Step 4:** Set  $l \leftarrow l + 1$  and go back to Step 2 until  $l = m$ .

To close this subsection, we will show in the following theorem that the proposed method is indeed a fast algorithm and preserves the optimal order of convergence. It can be proved by arguments similar to those in [10]. To present this result, following [10], we introduce two spaces of functions. For any  $\alpha := [\alpha_0, \alpha_1] \in \mathbb{N}_0^2$ , let  $|\alpha|_\infty := \max\{\alpha_0, \alpha_1\}$  and  $|\alpha| := |\alpha_0| + |\alpha_1|$ . For a function  $f \in C^\sigma(I^2)$  and for  $\alpha := [\alpha_0, \alpha_1] \in \mathbb{N}_0^2$ , we write

$$f^{(\alpha)}(s, t) := \left( \frac{\partial^{|\alpha|}}{\partial s^{\alpha_0} \partial t^{\alpha_1}} f \right) (s, t), \quad (s, t) \in I^2,$$

and define the space

$$X^\sigma(I^2) := \{f : I^2 \rightarrow \mathbb{R} : f^{(\alpha)} \in C(I^2), |\alpha|_\infty \leq \sigma\}.$$

For  $\mu \geq 0$ , we also define the space  $H^\mu(I^2)$  of functions  $\phi \in L^2(I^2)$  whose Fourier coefficients  $\{\phi_{k,l} : k, l \in \mathbb{Z}\}$  satisfy

$$\sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} (1 + k^2)^\mu (1 + l^2)^\mu |\phi_{k,l}|^2 < \infty.$$

**Theorem 3.1** *If the kernel functions  $\mathcal{K}$  and  $\mathcal{B} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon$ ,  $\mu \geq 0$ ,  $\epsilon > 0$  and  $u \in H^\mu(I)$ , then there exist a positive constant  $c$  and a positive integer  $n_0$ , such that for all  $k \geq n_0$  and  $m \in \mathbb{N}_0$ , there holds*

$$\|u_{k,m,N} - u\| \leq c2^{-\mu(k+m)} \|u\|_\mu,$$

where  $N = k + m$ . In addition, the total number of multiplications required for obtaining  $u_{k,m,N}$  in Algorithm 1 is  $\mathcal{O}((k + m)^3 2^{k+m})$ .

### 3.2 Fast Algorithm for Solving the Linear System

In the case of the Robin boundary value problem, the operator  $\Psi$  is given by

$$(\Psi u)(t) = u(t)|x'(t)|, \quad t \in I.$$

This leads the system (3.1), which is equivalent to (2.9), to be a linear system. We now write it in a matrix form. In this case, we also need the matrix representations of the operators  $\mathcal{H}$ ,  $\mathcal{A}$  and  $\mathcal{B}$ , which have been described in the last subsection. The only difference is the projection of  $\Psi \hat{u}_n$ . Since  $(\Psi \hat{u}_n)(t) = \hat{u}_n(t)|x'(t)|$ ,  $t \in I$ , the coefficients  $\hat{b}_k$ 's can be represented by

$$\hat{b}_k = \sum_{|l| \in \mathbb{Z}_n} \langle |x'|e_l, e_k \rangle \hat{a}_l, \quad |k| \in \mathbb{Z}_n.$$

For each  $|k|$ ,  $|l| \in \mathbb{Z}_n$ , we set  $\psi_{k,l} := \int_I |x'(t)|e_k(t)e_l(t)dt$  and define square matrix of order 2

$$\mathbf{F}_{k,l} := \begin{bmatrix} \psi_{-k,l} & \psi_{-k,-l} \\ \psi_{k,l} & \psi_{k,-l} \end{bmatrix}.$$

Using the matrix blocks defined by

$$\begin{aligned} \mathbf{F} &:= [\mathbf{F}_{k,l} : k, l \in \mathbb{Z}_n^+], \quad \mathbf{F}' := [\psi_{0,0}], \\ \mathbf{F}'' &:= [\psi_{0,k}, \psi_{0,-k} : k \in \mathbb{Z}_n^+], \quad \mathbf{F}''' := [\psi_{-k,0}, \psi_{k,0} : k \in \mathbb{Z}_n^+]^T, \end{aligned}$$

we build the matrix

$$\mathbf{F}_n := \begin{bmatrix} \mathbf{F}' & \mathbf{F}'' \\ \mathbf{F}''' & \mathbf{F} \end{bmatrix}.$$

With the help of this matrix and the representation matrices of operators, (3.1) reduces to the linear system

$$(\mathbf{I}_n - \mathbf{K}_n)\hat{u}_n + (\mathbf{A}_n + \mathbf{B}_n)\mathbf{F}_n\hat{u}_n = \mathbf{h}_n. \tag{3.4}$$

By similar arguments with those in the last subsection, we compress  $\mathbf{K}_n$  and  $\mathbf{B}_n$  to sparse matrices and evaluate the nonzero entries of the resulting matrices by the quadrature formula

in [23]. In addition, for computing the entries of  $\mathbf{F}_n$ , we use the trapezoid quadrature formula. That is, for each  $|k|, |l| \in \mathbb{Z}_n$ , we set

$$\tilde{\psi}_{k,l} := \frac{\pi}{n} \sum_{j \in \mathbb{Z}_{2n}} \left| x' \left( \frac{j\pi}{n} \right) \right| e_k \left( \frac{j\pi}{n} \right) e_l \left( \frac{j\pi}{n} \right). \tag{3.5}$$

We then denote  $\tilde{\mathbf{F}}_n$  the matrix  $\mathbf{F}_n$  with the entries being evaluated by formula (3.5). We note that the number of multiplications used to set up  $\tilde{\mathbf{F}}_n$  by the fast Fourier transform is  $\mathcal{O}(n \log_2 n)$ . Consequently, we obtain the fully discrete truncated linear system

$$(\mathbf{I}_n - \tilde{\mathbf{K}}_{n,N})\tilde{\mathbf{u}}_{n,N} + (\mathbf{A}_n + \tilde{\mathbf{B}}_{n,N})\tilde{\mathbf{F}}_n\tilde{\mathbf{u}}_{n,N} = \mathbf{h}_n. \tag{3.6}$$

In the light of the idea of the multilevel augmentation methods, we establish in the following a fast algorithm for solving (3.6) with  $n = 2^{k+m}$ . Here, we also define the submatrices and corresponding ‘‘low frequency’’ components and ‘‘high frequency’’ components for the matrix  $\tilde{\mathbf{F}}_n$  as we have defined for  $\tilde{\mathbf{K}}_{n,N}$ .

**Algorithm 2: The fast multilevel augmentation method for the linear system**

Let  $k$  be a fixed positive integer. Set  $\mathbf{h}_{2^{k+m}} := [\langle h, e_j \rangle, \langle h, e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+m}}]^T$ .

**Step 1:** Solving the linear system

$$(\mathbf{I}_{2^k} - \tilde{\mathbf{K}}_{k,0,N})\mathbf{u}_{k,0,N} + (\mathbf{A}_{2^k} + \tilde{\mathbf{B}}_{k,0,N})\tilde{\mathbf{F}}_{2^k}\mathbf{u}_{k,0,N} = \mathbf{h}_{2^k}, \tag{3.7}$$

for the solution  $\mathbf{u}_{k,0,N} = [(\mathbf{u}_{k,0,N})_j, (\mathbf{u}_{k,0,N})_{-j} : j \in \mathbb{Z}_{2^k}]^T$ . Define  $u_{k,0,N} := \sum_{|j| \in \mathbb{Z}_{2^k}} (\mathbf{u}_{k,0,N})_j e_j$ . Set  $l := 1$ .

**Step 2:** Set  $\mathbf{h}_{k,l} := [\langle h, e_j \rangle, \langle h, e_{-j} \rangle : j \in \mathbb{Z}_{2^{k+l}} \setminus \mathbb{Z}_{2^k}]^T$ .

- Augment the matrices  $\tilde{\mathbf{K}}_{k,l-1,N}^H, \mathbf{A}_{k,l-1}^H, \tilde{\mathbf{B}}_{k,l-1,N}^H$  and  $\tilde{\mathbf{F}}_{k,l-1}$  to form  $\tilde{\mathbf{K}}_{k,l,N}^H, \mathbf{A}_{k,l}^H, \tilde{\mathbf{B}}_{k,l,N}^H$  and  $\tilde{\mathbf{F}}_{k,l}$ , respectively.
- Augment the vector  $\mathbf{u}_{k,l-1,N}$  by setting  $\hat{\mathbf{u}}_{k,l,N} = \begin{bmatrix} \mathbf{u}_{k,l-1,N} \\ \mathbf{0}_{2^{k+l} \times 1} \end{bmatrix}$ .
- Compute

$$\mathbf{u}_{k,l,N}^H := \mathbf{h}_{k,l} + \tilde{\mathbf{K}}_{k,l,N}^H \hat{\mathbf{u}}_{k,l,N} - (\mathbf{A}_{k,l}^H + \tilde{\mathbf{B}}_{k,l,N}^H)\tilde{\mathbf{F}}_{k,l} \hat{\mathbf{u}}_{k,l,N}. \tag{3.8}$$

Let  $u_{k,l,N}^H := \sum_{|j| \in \mathbb{Z}_{2^{k+l}} \setminus \mathbb{Z}_{2^k}} (\mathbf{u}_{k,l,N}^H)_j e_j$ .

**Step 3:** Augment the matrices  $\tilde{\mathbf{K}}_{k,l-1,N}^L, \mathbf{A}_{k,l-1}^L$  and  $\tilde{\mathbf{B}}_{k,l-1,N}^L$  to form  $\tilde{\mathbf{K}}_{k,l,N}^L, \mathbf{A}_{k,l}^L$  and  $\tilde{\mathbf{B}}_{k,l,N}^L$ , respectively. Solve the linear system

$$(\mathbf{I}_{k,l}^L - \tilde{\mathbf{K}}_{k,l,N}^L) \begin{bmatrix} \mathbf{u}_{k,l,N}^L \\ \mathbf{u}_{k,l,N}^H \end{bmatrix} + (\mathbf{A}_{k,l}^L + \tilde{\mathbf{B}}_{k,l,N}^L)[(\tilde{\mathbf{F}}_{k,l}^L)^* \mathbf{u}_{k,l,N}^L + (\tilde{\mathbf{F}}_{k,l}^H)^* \mathbf{u}_{k,l,N}^H] = \mathbf{h}_{2^k} \tag{3.9}$$

for the solution  $\mathbf{u}_{k,l,N}^L = [(\mathbf{u}_{k,l,N}^L)_j, (\mathbf{u}_{k,l,N}^L)_{-j} : j \in \mathbb{Z}_{2^k}]^T$ . Define  $u_{k,l,N}^L := \sum_{|j| \in \mathbb{Z}_{2^k}} (\mathbf{u}_{k,l,N}^L)_j e_j$ ,  $\mathbf{u}_{k,l,N} := \begin{bmatrix} \mathbf{u}_{k,l,N}^L \\ \mathbf{u}_{k,l,N}^H \end{bmatrix}$  and  $u_{k,l,N} := u_{k,l,N}^L + u_{k,l,N}^H$ .

**Step 4:** Set  $l \leftarrow l + 1$  and go back to Step 2 until  $l = m$ .

*Remark* In step 2 of Algorithm 2, for computing  $\mathbf{u}_{k,l,N}^H$  according to formula (3.8), we have to face the multiplication of the dense matrix  $\tilde{\mathbf{F}}_{k,l}$  and the vectors  $\hat{\mathbf{u}}_{k,l,N}$ . This will bring to us the

large computational costs. To deal with this issue, we rewrite the vector  $\hat{\mathbf{v}}_{k,l,N} := \tilde{\mathbf{F}}_{k,l} \hat{\mathbf{u}}_{k,l,N}$  by the following formulas:

$$u_{k,l-1,N} \left( \frac{j\pi}{n} \right) = \sum_{|q| \in \mathbb{Z}_{2^{k+l-1}}} (\mathbf{u}_{k,l-1,N})_q e_q \left( \frac{j\pi}{n} \right), \quad j \in \mathbb{Z}_{2n},$$

and

$$(\hat{\mathbf{v}}_{k,l,N})_p := \frac{\pi}{n} \sum_{j \in \mathbb{Z}_{2n}} u_{k,l-1,N} \left( \frac{j\pi}{n} \right) \left| x' \left( \frac{j\pi}{n} \right) \right| e_p \left( -\frac{j\pi}{n} \right), \quad |p| \in \mathbb{Z}_{2^{k+l}}.$$

Applying the fast Fourier transform to obtain  $u_{k,l-1,N} \left( \frac{j\pi}{n} \right)$ ,  $j \in \mathbb{Z}_{2n}$ , and then  $(\hat{\mathbf{v}}_{k,l,N})_p$ ,  $|p| \in \mathbb{Z}_{2^{k+l}}$ , we require only  $\mathcal{O}(n \log_2 n)$  number of multiplications. We shall compute the vector  $(\tilde{\mathbf{F}}_{k,l}^H)^* \mathbf{u}_{k,l,N}^H$  in (3.9) in the same way.

In the rest of this subsection, we analyze the proposed method. We first gives an estimation of the computational cost required for Algorithm 2.

**Theorem 3.2** *The total number of multiplications required for obtaining  $\mathbf{u}_{k,m,N}$  in Algorithm 2 is  $\mathcal{O}((k+m)^3 2^{k+m})$ .*

*Proof* The total number of multiplications required for obtaining  $\mathbf{u}_{k,m,N}$  equals to the sum of numbers of multiplications used to generate the matrices  $\tilde{\mathbf{K}}_{2^{k+m},N}$ ,  $\tilde{\mathbf{B}}_{2^{k+m},N}$  and  $\tilde{\mathbf{F}}_{2^{k+m}}$  and that used to carry out the computing steps listed in Algorithm 2.

It is known [24] that the number of multiplications for generating  $\tilde{\mathbf{K}}_{2^{k+m},N}$  and  $\tilde{\mathbf{B}}_{2^{k+m},N}$  is  $\mathcal{O}((k+m)^3 2^{k+m})$ . Applying the fast Fourier transform to compute the entries of  $\tilde{\mathbf{F}}_n$ , we require  $\mathcal{O}((k+m)2^{k+m})$  number of multiplications. To carry out the computing steps listed in Algorithm 2, we need to compute only once of step 1 and update for  $l \in \mathbb{Z}_{m+1}$  in steps 2 and 3. The computational cost of step 1, used to solve a linear system of a much smaller order, can be considered as a constant. In step 2, as has been pointed out, the number of multiplications used for computing the vector  $\tilde{\mathbf{F}}_{k,l} \hat{\mathbf{u}}_{k,l,N}$  is  $\mathcal{O}((k+m)2^{k+m})$ . Since the truncated matrices  $\tilde{\mathbf{K}}_{k,l,N}^H$  and  $\tilde{\mathbf{B}}_{k,l,N}^H$  both have  $\mathcal{O}((k+l)2^{k+l})$  numbers of nonzero entries, the number of multiplications used to multiply these matrices with corresponding vectors is also  $\mathcal{O}((k+l)2^{k+l})$ . In step 3, the number of multiplications for evaluating the vector  $(\tilde{\mathbf{F}}_{k,l}^H)^* \mathbf{u}_{k,l,N}^H$  is also  $\mathcal{O}((k+m)2^{k+m})$ . Moreover, since the linear system (3.9) has the same size as (3.7), we consider the computational cost for solving it as a constant.

Consequently, we get the desired result by adding up the above estimates together.  $\square$

We next establish the convergence order of the approximate solution  $u_{k,m,N}$  by a similar analysis used in the nonlinear case. To this end, we represent Algorithm 2 in an operator form. Following [10], we introduce the following operators. Set  $n := 2^{k+m}$  and  $N := k+m$ . We denote by  $\mathcal{K}_n$  and  $\tilde{\mathcal{K}}_{n,N}$  the linear operators from  $\mathbb{X}_n$  to  $\mathbb{X}_n$ , which have  $\mathbf{K}_n$  and  $\tilde{\mathbf{K}}_{n,N}$  as the matrix representations under the Fourier basis, respectively. For each  $l \in \mathbb{Z}_{m+1}$ , we also introduce the operator  $\tilde{\mathcal{K}}_{k,l,N} : \mathbb{X}_{2^{k+l}} \rightarrow \mathbb{X}_{2^{k+l}}$  with the matrix representation  $\tilde{\mathbf{K}}_{k,l,N}$  and split the operator into its “lower frequency” and “higher frequency” components as  $\tilde{\mathcal{K}}_{k,l,N} = \tilde{\mathcal{K}}_{k,l,N}^L + \tilde{\mathcal{K}}_{k,l,N}^H$ , with

$$\tilde{\mathcal{K}}_{k,l,N}^L := \mathcal{P}_{2^k} \tilde{\mathcal{K}}_{k,l,N}, \quad \tilde{\mathcal{K}}_{k,l,N}^H := (\mathcal{P}_{2^{k+l}} - \mathcal{P}_{2^k}) \tilde{\mathcal{K}}_{k,l,N}.$$

It is clear that the matrices  $\tilde{\mathbf{K}}_{k,l,N}^L$  and  $\tilde{\mathbf{K}}_{k,l,N}^H$  are just the matrix representations of the operators  $\tilde{\mathcal{K}}_{k,l,N}^L$  and  $\tilde{\mathcal{K}}_{k,l,N}^H$ , respectively. In a similar manner, we also introduce the corresponding operators for the kernel functions  $\mathcal{A}$  and  $\mathcal{B}$ .

Moreover, we define the interpolating projection operator  $\mathcal{Q}_n : L^2(I) \rightarrow \mathbb{X}_n$  by

$$\mathcal{Q}_n(\phi) := \sum_{|k| \in \mathbb{Z}_n} \tilde{\phi}_k e_k$$

with

$$\tilde{\phi}_k := \frac{\pi}{n} \sum_{j \in \mathbb{Z}_{2n}} \phi \left( \frac{j\pi}{n} \right) e_{-k} \left( \frac{j\pi}{n} \right), \quad |k| \in \mathbb{Z}_n.$$

It is known [2,25] that there exists a positive constant  $c$  such that for any  $\phi \in H^\mu(I)$  with  $\mu > 1/2$ , there holds

$$\|\phi - \mathcal{Q}_n \phi\| \leq cn^{-\mu} \|\phi\|_\mu.$$

Let  $\mathcal{F}_n := \mathcal{P}_n \Psi|_{\mathbb{X}_n}$  and  $\tilde{\mathcal{F}}_n := \mathcal{Q}_n \Psi|_{\mathbb{X}_n}$ . Then the matrices  $\mathbf{F}_n$  and  $\tilde{\mathbf{F}}_n$  are the matrix representations of operators  $\mathcal{F}_n$  and  $\tilde{\mathcal{F}}_n$ , respectively. For each  $l \in \mathbb{Z}_{m+1}$ , we use  $\tilde{\mathcal{F}}_{k,l}$  to denote the linear operator from  $\mathbb{X}_{2^{k+l}}$  to  $\mathbb{X}_{2^{k+l}}$  with the matrix representation  $\tilde{\mathbf{F}}_{k,l}$ .

With these notations, for each  $l \in \mathbb{Z}_{m+1}$ , Eq. (3.8) is equivalent to

$$u_{k,l,N}^H = (\mathcal{P}_{2^{k+l}} - \mathcal{P}_{2^k})h + [\tilde{\mathcal{K}}_{k,l,N}^H - (\mathcal{A}_{k,l}^H + \tilde{\mathcal{B}}_{k,l,N}^H)\tilde{\mathcal{F}}_{k,l}]u_{k,l-1,N}, \tag{3.10}$$

and linear system (3.9) is equivalent to

$$[\mathcal{P}_{2^k} - \tilde{\mathcal{K}}_{k,l,N}^L + (\mathcal{A}_{k,l}^L + \tilde{\mathcal{B}}_{k,l,N}^L)\tilde{\mathcal{F}}_{k,l}](u_{k,l,N}^L + u_{k,l,N}^H) = \mathcal{P}_{2^k}h. \tag{3.11}$$

Combining (3.10) with (3.11), we have for each  $l \in \mathbb{Z}_{m+1}$

$$\begin{aligned} & [\mathcal{I} - \tilde{\mathcal{K}}_{k,l,N}^L + (\mathcal{A}_{k,l}^L + \tilde{\mathcal{B}}_{k,l,N}^L)\tilde{\mathcal{F}}_{k,l}]u_{k,l,N} \\ &= \mathcal{P}_{2^{k+l}}h + [\tilde{\mathcal{K}}_{k,l,N}^H - (\mathcal{A}_{k,l}^H + \tilde{\mathcal{B}}_{k,l,N}^H)\tilde{\mathcal{F}}_{k,l}]u_{k,l-1,N}. \end{aligned} \tag{3.12}$$

To give the convergence analysis, we need some useful lemmas. The first lemma given in [10,24] is concerned with the differences between  $\mathcal{K}_n$  and  $\tilde{\mathcal{K}}_{n,N}$ .

**Lemma 3.3** *If  $N := \lceil \log_2 n \rceil$  and  $\mathcal{K} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon$ ,  $\mu \geq 0$ ,  $\epsilon > 0$ , then there exist positive constants  $c$  and  $n_0 \in \mathbb{N}$  such that for all  $n \in \mathbb{N}$  with  $n \geq n_0$  and for all  $w \in L^2(I)$ ,*

$$\|(\mathcal{K}_n - \tilde{\mathcal{K}}_{n,N})\mathcal{P}_n w\| \leq c\|w\|n^{-\mu}.$$

As a direct consequence of Lemma 4.3 in [10], we have the following estimation of the differences  $(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\mathcal{F}_n$ .

**Lemma 3.4** *If  $N := \lceil \log_2 n \rceil$  and  $\mathcal{B} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon$ ,  $\mu \geq 0$ ,  $\epsilon > 0$ , then there exist positive constants  $c$  and  $n_0 \in \mathbb{N}$  such that for all  $n \in \mathbb{N}$  with  $n \geq n_0$  and for all  $w \in L^2(I)$ ,*

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\mathcal{F}_n \mathcal{P}_n w\| \leq c\|w\|n^{-\mu}.$$

The next result is devoted to estimate the difference between the operators  $\mathcal{F}_n$  and  $\tilde{\mathcal{F}}_n$ , which can be proved by similar arguments to those of Lemma 4.1 in [5].

**Lemma 3.5** *If the function  $|x'| \in H^{\mu+1}(I)$ ,  $\mu \geq 0$ , then there exists a positive constant  $c$  such that for all  $n \in \mathbb{N}$  and  $w \in L^2(I)$*

$$\|(\mathcal{F}_n - \tilde{\mathcal{F}}_n)\mathcal{P}_n w\| \leq c\|w\|n^{-\mu}. \tag{3.13}$$

*Proof* Replacing the function  $a$  by  $|x'|$ , Lemma 4.1 in [5] leads directly the desired result.  $\square$

Combining the above three lemmas, we obtain the following estimation.

**Lemma 3.6** *If  $N := \lceil \log_2 n \rceil$ ,  $\mathcal{B} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon$ ,  $\mu \geq 0$ ,  $\epsilon > 0$ , and  $|x'| \in H^{\mu+1}(I)$ , then there exist positive constants  $c$  and  $n_0 \in \mathbb{N}$  such that for all  $n \in \mathbb{N}$  with  $n \geq n_0$  and for all  $w \in L^2(I)$ ,*

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\tilde{\mathcal{F}}_n \mathcal{P}_n w\| \leq c \|w\| n^{-\mu}.$$

*Proof* By the triangle inequality, we get for all  $n \in \mathbb{N}$  and  $w \in L^2(I)$

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\tilde{\mathcal{F}}_n \mathcal{P}_n w\| \leq \|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\mathcal{F}_n \mathcal{P}_n w\| + \|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})(\mathcal{F}_n - \tilde{\mathcal{F}}_n)\mathcal{P}_n w\|. \tag{3.14}$$

On one hand, by Lemma 3.4, there exist  $c_1 > 0$  and  $n_1 \in \mathbb{N}$  such that for all  $n \geq n_1$  and  $w \in L^2(I)$ ,

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})\mathcal{F}_n \mathcal{P}_n w\| \leq c_1 \|w\| n^{-\mu}. \tag{3.15}$$

On the other hand, applying Lemma 3.3 to the kernel  $\mathcal{B}$ , we have that there exist positive constants  $c_2$  and  $n_2 \in \mathbb{N}$  such that for all  $n \geq n_2$  and  $w \in L^2(I)$ ,

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})(\mathcal{F}_n - \tilde{\mathcal{F}}_n)\mathcal{P}_n w\| \leq c_2 \|(\mathcal{F}_n - \tilde{\mathcal{F}}_n)\mathcal{P}_n w\| n^{-\mu}.$$

Together with (3.13) the above inequality shows that there exists positive constant  $c_3$  such that

$$\|(\mathcal{B}_n - \tilde{\mathcal{B}}_{n,N})(\mathcal{F}_n - \tilde{\mathcal{F}}_n)\mathcal{P}_n w\| \leq c_3 \|w\| n^{-2\mu}. \tag{3.16}$$

Substituting (3.15) and (3.16) into the inequality (3.14) and letting  $c := \max\{c_1, c_3\}$ ,  $n_0 := \max\{n_1, n_2\}$ , we get the desired result.  $\square$

A standard argument in [2,25] tells us that there exist a positive integer  $n_0$  and a positive constant  $\rho$  such that for all  $n \geq n_0$  and all  $w \in X_n$ ,

$$\|(\mathcal{I} - \mathcal{K}_n + (\mathcal{A}_n + \mathcal{B}_n)\mathcal{F}_n)w\| \geq \rho \|w\|, \tag{3.17}$$

which shows the operator  $\mathcal{I} - \mathcal{K}_n + (\mathcal{A}_n + \mathcal{B}_n)\mathcal{F}_n$  is stable. We are now ready to present the stability of the operator  $\mathcal{I} - \tilde{\mathcal{K}}_{n,N} + (\mathcal{A}_n + \tilde{\mathcal{B}}_{n,N})\tilde{\mathcal{F}}_n$ .

**Theorem 3.7** *If  $N := \lceil \log_2 n \rceil$ ,  $\mathcal{K}, \mathcal{B} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon$ ,  $\mu \geq 0$ ,  $\epsilon > 0$ , and  $|x'| \in H^{\mu+1}(I)$ , then there exist a positive integer  $n_0$  such that for all  $n \geq n_0$  and for all  $w \in X_n$ ,*

$$\|(\mathcal{I} - \tilde{\mathcal{K}}_{n,N} + (\mathcal{A}_n + \tilde{\mathcal{B}}_{n,N})\tilde{\mathcal{F}}_n)w\| \geq \frac{\rho}{2} \|w\|.$$

*Proof* By Lemmas 3.3 and 3.6, we conclude that there exists a positive integer  $n_1$  such that for all  $n \geq n_1$  and for all  $w \in X_n$ ,

$$\|(\mathcal{K}_n - \tilde{\mathcal{K}}_{n,N})w\| \leq \frac{\rho}{6} \|w\|, \quad \|(\tilde{\mathcal{B}}_{n,N} - \mathcal{B}_n)\tilde{\mathcal{F}}_n w\| \leq \frac{\rho}{6} \|w\|.$$

It follows from Lemma 3.5 and the uniformly boundedness of the operators  $\mathcal{A}_n$  and  $\mathcal{B}_n$ ,  $n \in \mathbb{N}$ , that there exists a positive integer  $n_2$  such that for all  $n \geq n_2$  and for all  $w \in X_n$ ,

$$\|(\mathcal{A}_n + \mathcal{B}_n)(\tilde{\mathcal{F}}_n - \mathcal{F}_n)w\| \leq \frac{\rho}{6} \|w\|.$$

Combining the above inequalities with (3.17), we get for all  $n \geq n_0 := \max\{n_1, n_2\}$  and for all  $w \in X_n$ ,

$$\|(\mathcal{I} - \tilde{\mathcal{K}}_{n,N} + (\mathcal{A}_n + \tilde{\mathcal{B}}_{n,N})\tilde{\mathcal{F}}_n)w\|$$



$$\begin{aligned} &\geq \|(\mathcal{I} - \mathcal{K}_n + (\mathcal{A}_n + \mathcal{B}_n)\mathcal{F}_n)w\| - \|(\mathcal{K}_n - \tilde{\mathcal{K}}_{n,N})w\| \\ &\quad - \|(\tilde{\mathcal{B}}_{n,N} - \mathcal{B}_n)\tilde{\mathcal{F}}_n w\| - \|(\mathcal{A}_n + \mathcal{B}_n)(\tilde{\mathcal{F}}_n - \mathcal{F}_n)w\| \geq \frac{\rho}{2}\|w\|. \end{aligned}$$

□

**Theorem 3.8** *If  $\mathcal{K}, \mathcal{B} \in X^\sigma(I^2) \cap H^\mu(I^2)$  with  $\sigma \geq \mu + 1/2 + \epsilon, \mu \geq 0, \epsilon > 0$  and  $u \in H^\mu(I), |x'| \in H^{\mu+1}(I)$ , then there exist a positive constant  $c$  and a positive integer  $n_0$ , such that for all  $k \geq n_0$  and for all  $m \in \mathbb{N}_0$ ,*

$$\|u_{k,m,N} - u\| \leq c2^{-\mu(k+m)}\|u\|_\mu, \tag{3.18}$$

where  $N = k + m$ .

*Proof* We prove this theorem by induction on  $m$ . The proof for the case  $m = 0$  is trivial. We assume that (3.18) holds for the case of  $m - 1$  and consider the case of  $m$ .

Let  $\hat{u}_{2^{k+m}}$  be the solution of Eq. (2.9) with  $n = 2^{k+m}$ . Using the triangle inequality, we get

$$\|u_{k,m,N} - u\| \leq \|\hat{u}_{2^{k+m}} - u\| + \|u_{k,m,N} - \hat{u}_{2^{k+m}}\|.$$

According to Theorem 2.3, the first term on the right-hand side of the above inequality can be estimated as

$$\|u - \hat{u}_{2^{k+m}}\| \leq c_02^{-\mu(k+m)}\|u\|_\mu \tag{3.19}$$

for some positive constant  $c_0$  and sufficiently large integer  $k$ . It suffices to estimate the second term. Combining (2.9) with (3.12), we represent the difference  $u_{k,m,N} - \hat{u}_{2^{k+m}}$  as

$$u_{k,m,N} - \hat{u}_{2^{k+m}} = [\mathcal{I} - \tilde{\mathcal{K}}_{k,m,N} + (\mathcal{A}_{2^{k+m}} + \tilde{\mathcal{B}}_{k,m,N})\tilde{\mathcal{F}}_{2^{k+m}}]^{-1}(u_1 + u_2), \tag{3.20}$$

where

$$\begin{aligned} u_1 := &(\tilde{\mathcal{K}}_{k,m,N} - \mathcal{K}_{2^{k+m}})\hat{u}_{2^{k+m}} + [(\mathcal{B}_{2^{k+m}} - \tilde{\mathcal{B}}_{k,m,N})\tilde{\mathcal{F}}_{2^{k+m}}]\hat{u}_{2^{k+m}} \\ &+ (\mathcal{A}_{2^{k+m}} + \mathcal{B}_{2^{k+m}})(\mathcal{F}_{2^{k+m}} - \tilde{\mathcal{F}}_{2^{k+m}})\hat{u}_{2^{k+m}}, \end{aligned}$$

and

$$u_2 := [\tilde{\mathcal{K}}_{k,m,N}^H + (\mathcal{A}_{2^{k+m}}^H + \tilde{\mathcal{B}}_{k,m,N}^H)\tilde{\mathcal{F}}_{2^{k+m}}](u_{k,m,N} - u_{k,m-1,N}).$$

From Theorem 3.7, there holds for sufficiently large integer  $k$

$$\|u_{k,m,N} - \hat{u}_{2^{k+m}}\| \leq \frac{2}{\rho}(\|u_1\| + \|u_2\|). \tag{3.21}$$

We now turn to estimating  $u_1$  and  $u_2$ . It follows from Lemmas 3.3, 3.5 and 3.6 and the uniformly boundedness of the operators  $\mathcal{A}_{2^{k+m}}$  and  $\mathcal{B}_{2^{k+m}}$ , that there exists a constant  $c_1$  such that for sufficiently large integer  $k$  and for all  $m$

$$\|u_1\| \leq c_12^{-\mu(k+m)}\|\hat{u}_{2^{k+m}}\|.$$

Since  $\|\hat{u}_{2^{k+m}} - u\| \rightarrow 0$  uniformly for all  $m$ , as  $k \rightarrow \infty$ , and  $\|u\| \leq \|u\|_\mu$ , we conclude that there exists a constant  $c_2$  such that for sufficiently large integer  $k$  and for all  $m$

$$\|u_1\| \leq c_22^{-\mu(k+m)}\|u\|_\mu. \tag{3.22}$$

To estimate  $u_2$ , we first prove the uniformly boundedness of the operators  $\tilde{\mathcal{F}}_n$ ,  $n \in \mathbb{N}$ , on  $\mathbb{X}_n$ . By Lemma 3.5 and with the fact that there holds  $\|\mathcal{F}_n w\| \leq c' \|w\|$  for all  $n \in \mathbb{N}$  and  $w \in \mathbb{X}_n$ , we have

$$\|\tilde{\mathcal{F}}_n w\| \leq \|(\tilde{\mathcal{F}}_n - \mathcal{F}_n)w\| + \|\mathcal{F}_n w\| \leq (c''n^{-\mu} + c')\|w\|,$$

for all  $n \in \mathbb{N}$  and  $w \in \mathbb{X}_n$ . That is to say, there exists a positive constant  $c_3$  such that for all  $n \in \mathbb{N}$  and  $w \in \mathbb{X}_n$ ,  $\|\tilde{\mathcal{F}}_n w\| \leq c_3 \|w\|$ . This leads to the following estimation of  $u_2$

$$\|u_2\| \leq (\|\tilde{\mathcal{K}}_{k,m,N}^H\| + c_3\|\mathcal{A}_{2^{k+m}}^H\| + c_3\|\tilde{\mathcal{B}}_{k,m,N}^H\|)\|u_{k,m,N} - u_{k,m-1,N}\|.$$

We note that  $\|\tilde{\mathcal{K}}_{k,m,N}^H\| + c_3\|\mathcal{A}_{2^{k+m}}^H\| + c_3\|\tilde{\mathcal{B}}_{k,m,N}^H\| \rightarrow 0$  uniformly for all  $m$ , as  $k \rightarrow \infty$ . Hence, there holds for sufficiently large integer  $k$  and for all  $m$ ,

$$\|u_2\| \leq \frac{\rho}{4}(\|u_{k,m-1,N} - \hat{u}_{2^{k+m}}\| + \|\hat{u}_{2^{k+m}} - u_{k,m,N}\|). \tag{3.23}$$

Substituting (3.22) and (3.23) into (3.21), we obtain

$$\|u_{k,m,N} - \hat{u}_{2^{k+m}}\| \leq \frac{4c_2}{\rho}2^{-\mu(k+m)}\|u\|_{\mu} + \|u_{k,m-1,N} - \hat{u}_{2^{k+m}}\|.$$

Together with the induction hypothesis, the above inequality yields

$$\begin{aligned} \|u_{k,m,N} - \hat{u}_{2^{k+m}}\| &\leq \frac{4c_2}{\rho}2^{-\mu(k+m)}\|u\|_{\mu} + \|u_{k,m-1,N} - u\| + \|u - \hat{u}_{2^{k+m}}\| \\ &\leq c_4 2^{-\mu(k+m)}\|u\|_{\mu}, \end{aligned} \tag{3.24}$$

for some positive constant  $c_4$ . By inequalities (3.19) and (3.24), we prove (3.18) for  $m$ .  $\square$

### 4 Numerical Examples

In this section, we present numerical examples to demonstrate the approximate accuracy and computational complexity of the proposed method. All computer programs for the numerical examples are run on a personal computer with a 2.8GHz CPU and 8G memory.

As has been pointed out in [10], if the exact solution  $u$  of the boundary integral equation is analytic, the improved Fourier–Galerkin method (2.9) enjoys the exponential convergence order. Hence, for a small  $n$  the approximate solution  $\hat{u}_n$  will achieve enough accuracy and then we solve the nonlinear system (3.2) directly by Newton’s method or solve the linear system (3.4) directly by Gaussian elimination. In this case, we denote by  $d_n$  the dimensions of the approximate subspaces and define the relative error  $\text{Err}$  of the approximate solution by

$$\text{Err} := \frac{\|u - \hat{u}_n\|}{\|u\|}.$$

In the case when the exact solution  $u$  has a general regularity, we will use the fast algorithms proposed in section 3 for solving (2.9). To demonstrate the numerical results, we let  $k$  denote the initial level and  $m$  denote the number of levels to be augmented. We also use  $d_{2^{k+m}}$  to denote the dimensions of the approximate subspaces. The relative error “Err” and the approximation order “AO” are defined respectively by

$$\text{Err} := \frac{\|u - u_{k,m,k+m}\|}{\|u\|}, \quad \text{and} \quad \text{AO} := \log_2 \frac{\|u - u_{k,m,k+m}\|}{\|u - u_{k,m+1,k+m+1}\|},$$

**Table 1** Numerical results for Example 4.1

$n$	$d_n$	Err
2	3	8.0990e-03
8	15	1.1300e-06
16	31	1.1913e-11

where  $u_{k,m,k+m}$  is the approximate solution obtained by Algorithm 1 or Algorithm 2. We denote by “CT” the computing time measured in seconds, which is used for generating the sparse matrices  $\tilde{\mathbf{K}}_{n,N}$ ,  $\tilde{\mathbf{B}}_{n,N}$  and solving the nonlinear system (3.3) by Algorithm 1 or solving the linear system (3.6) by Algorithm 2. To confirm the quasi-linear computational complexity, we also define

$$CO := \log_2 \frac{CT(2^{k+m+1})}{CT(2^{k+m})}.$$

We first present three numerical examples for solving the boundary integral equation reformulated from the nonlinear boundary value problem:

$$\begin{cases} \Delta u(x) - 2u(x) = 0, & x \in D, \\ \frac{\partial u(x)}{\partial \mathbf{n}_x} = -u(x) - \sin u(x) + f(x), & x \in \Gamma, \end{cases} \tag{4.1}$$

where  $D$  is a circle plate region with the boundary  $\Gamma$  having the parametrization

$$x(t) = (\cos t, \sin t), \quad t \in I.$$

*Example 4.1* We choose the right-hand side function  $f = e^{\cos t + \sin t}(\cos t + \sin t + 1) + \sin(e^{\cos t + \sin t})$ ,  $t \in I$ , so that the function

$$u(t) = e^{\cos t + \sin t}, \quad t \in I,$$

is the exact solution of the integral equation reformulated from (4.1). Since the solution  $u$  is analytic, we obtain the approximation solution by solving (3.2) directly by Newton’s method. The numerical results for this example are presented in Table 1.

We observe from the numerical results that the approximation error decays quite fast so that for  $n = 16$  the approximate solution  $\hat{u}_n$  has already got enough accuracy. Hence, we need not to consider a fast solver for the improved Fourier–Galerkin method.

To confirm the theoretical order of approximation and computational complexity of Algorithm 1, the following two examples are concerned with the exact solutions with different regularities.

*Example 4.2* In this example, the right-hand side function  $f$  is chosen such that

$$u(t) = \begin{cases} t, & 0 \leq t < \pi, \\ 2\pi - t, & \pi \leq t < 2\pi, \end{cases}$$

is the exact solution of the integral equation. We note that  $u \in H^{0.5-\epsilon}(I)$  with  $\epsilon > 0$  being an arbitrary number. We solve (3.2) by Algorithm 1 with the initial level  $k = 4$  and list the numerical results in Table 2. It is known from the regularity of  $u$  that the convergence order of the approximation solution is  $0.5 - \epsilon$ . This is confirmed by the numerical results listed in Table 2. The numerical results also show that the computing time is consistent with the quasi-linear computational complexity estimate.

**Table 2** Numerical results of Algorithm 1 for Example 4.2

$m$	$d_{2^{4+m}}$	Err	AO	CT	CO
0	31	9.9015e−02		0.008	
1	63	6.9683e−02	0.5068	0.021	1.3923
2	127	4.8969e−02	0.5045	0.061	1.5384
3	255	3.4533e−02	0.5037	0.163	1.4183
4	511	2.4391e−02	0.5017	0.444	1.4457
5	1,023	1.7238e−02	0.5007	1.167	1.3742
6	2,047	1.2186e−02	0.5003	3.156	1.4353

**Table 3** Numerical results of Algorithm 1 for Example 4.3

$m$	$d_{2^{4+m}}$	Err	AO	CT	CO
0	31	9.0850e−03		0.008	
1	63	3.1511e−03	1.5276	0.023	1.5236
2	127	1.0977e−03	1.5214	0.061	1.4072
3	255	3.8554e−04	1.5095	0.165	1.4356
4	511	1.3589e−04	1.5044	0.437	1.4052
5	1,023	4.7973e−05	1.5021	1.168	1.4183
6	2,047	1.6949e−05	1.5010	3.162	1.4368

*Example 4.3* We choose the right-hand side function  $f$  such that

$$u(t) = \frac{1}{12}(3t^2 - 6\pi t + 2\pi^2), \quad t \in I,$$

is the exact solution of the integral equation. It can be seen that  $u \in H^{1.5-\epsilon}(I)$  with  $\epsilon > 0$  being an arbitrary number. We also solve (3.2) by Algorithm 1 with the initial level  $k = 4$  and report the the numerical results for this example in Table 3.

The next three numerical examples are devoted to solving the boundary integral equation reformulated from the Robin boundary value problem:

$$\begin{cases} \Delta u(x) - 2u(x) = 0, & x \in D, \\ \frac{\partial u(x)}{\partial \mathbf{n}_x} = -u(x) + f(x), & x \in \Gamma, \end{cases} \quad (4.2)$$

where  $D$  is a kite-shaped region with the boundary  $\Gamma$  having the parametrization

$$x(t) = (x_1(t), x_2(t)) = (\cos t + 0.65 \cos 2t + 0.65, 1.5 \sin t), \quad t \in I.$$

*Example 4.4* In this example, we consider the analytic solution  $u(t) = e^{-x_1(t)-x_2(t)}$ ,  $t \in I$ . Then the right-hand side function has the form  $f(t) = e^{-x_1(t)-x_2(t)}[(-1.5 \cos t - \sin t - 1.3 \sin 2t)/\sqrt{w(t)}]$  with  $w(t) = (1.5 \cos t)^2 + (\sin t + 1.3 \sin 2t)^2$ ,  $t \in I$ . We solve (3.4) directly by Gaussian elimination. The numerical results presented in Table 4 confirm that the improved Fourier–Galerkin method attains the exponential convergence order.

*Example 4.5* We choose the right-hand side function  $f$  such that

$$u(t) = \begin{cases} t, & 0 \leq t < \pi, \\ 2\pi - t, & \pi \leq t < 2\pi, \end{cases}$$

**Table 4** Numerical results for Example 4.4

$n$	$d_n$	Err
2	3	1.0415e-01
8	15	1.5918e-03
16	31	2.7569e-08
32	63	4.5270e-11

**Table 5** Numerical results of Algorithm 2 for Example 4.5

$m$	$d_{2^{4+m}}$	Err	AO	CT	CO
0	31	9.9005e-02		0.008	
1	63	6.9543e-02	0.5096	0.023	1.5236
2	127	4.8937e-02	0.5070	0.063	1.4537
3	255	3.4528e-02	0.5032	0.173	1.4573
4	511	2.4390e-02	0.5015	0.462	1.4171
5	1,023	1.7238e-02	0.5007	1.234	1.4174
6	2,047	1.2186e-02	0.5004	3.690	1.5803

**Table 6** Numerical results of Algorithm 2 for Example 4.6

$m$	$d_{2^{4+m}}$	Err	AO	CT	CO
0	31	9.0847e-03		0.008	
1	63	3.1695e-03	1.5192	0.023	1.5236
2	127	1.0996e-03	1.5273	0.064	1.4764
3	255	3.8570e-04	1.5114	0.175	1.4512
4	511	1.3591e-04	1.5049	0.466	1.4130
5	1,023	4.7974e-05	1.5023	1.225	1.3944
6	2,047	1.6949e-05	1.5011	3.741	1.6106

is the exact solution of the integral equation reformulated from (4.2). Solving the linear system (3.6) by Algorithm 2 with the initial level  $k = 4$ , we obtain the approximation solution. The desired theoretical approximation order is  $0.5 - \epsilon$  since  $u \in H^{0.5-\epsilon}(I)$  with  $\epsilon > 0$  being an arbitrary number. We list the numerical results of this example in Table 5, which are consistent with the approximate accuracy and computational complexity of Algorithm 2.

*Example 4.6* In this example, we also consider the exact solution of the integral equation as

$$u(t) = \frac{1}{12}(3t^2 - 6\pi t + 2\pi^2), \quad t \in I,$$

by choosing the right-hand side function  $f$ . As has been pointed out in Example 4.3, there holds  $u \in H^{1.5-\epsilon}(I)$  with  $\epsilon > 0$  being an arbitrary number. In Table 6, we present the numerical results for this example when we solve (3.6) by Algorithm 2 with the initial level  $k = 4$ .

**Acknowledgments** This research is partially supported by Guangdong Provincial Government of China through the ‘‘Computational Science Innovative Research Team’’ program, by the Natural Science Foundation

of China under Grants 11401207, 11301208, 11271370, by Hunan Provincial Natural Science Foundation of China under Grant 2015JJ6069.

## References

- Ahlberg, J.H., Nilson, E.N., Walsh, J.L.: *The Theory of Splines and Their Applications*. Academic Press, New York (1967)
- Atkinson, K.E.: *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, Cambridge (1997)
- Atkinson, K.E., Chandler, G.: Boundary integral equation methods for solving Laplace's equation with nonlinear boundary conditions: the smooth boundary case. *Math. Comput.* **55**, 451–472 (1990)
- Bazant, M.Z., Thornton, K., Ajdari, A.: Diffuse-charge dynamics in electrochemical systems. *Phys. Rev. E* **70**, 021506 (2004)
- Cai, H.: A fast solver for the Hilbert-type singular integral equations based on the direct Fourier spectral method. *J. Comput. Appl. Math.* **250**, 83–95 (2013)
- Cai, H., Xu, Y.: A fast Fourier–Galerkin method for solving singular boundary integral equations. *SIAM J. Numer. Anal.* **46**, 1965–1984 (2008)
- Cheng, H., Huang, J., Leiterman, T.: An adaptive fast solver for the modified Helmholtz equation in two dimensions. *J. Comput. Phys.* **211**, 616–637 (2006)
- Cheng, P., Huang, J., Wang, Z.: Mechanical quadrature methods and extrapolation for solving nonlinear boundary Helmholtz integral equations. *Appl. Math. Mech. (English Ed.)* **32**, 1505–1514 (2011)
- Chen, X., Chen, Z., Wu, B., Xu, Y.: Fast multilevel augmentation methods for nonlinear boundary integral equation. *SIAM J. Numer. Anal.* **49**, 2231–2255 (2011)
- Chen, X., Wang, R., Xu, Y.: Fast Fourier-Galerkin methods for nonlinear boundary integral equations. *J. Sci. Comput.* **56**, 494–514 (2013)
- Chen, Z., Micchelli, C.A., Xu, Y.: *Multiscale Methods for Fredholm Integral Equations*. Cambridge University Press, Cambridge (2014)
- Chen, Z., Wu, B., Xu, Y.: Fast multilevel augmentation methods for solving Hammerstein equations. *SIAM J. Numer. Anal.* **47**, 2321–2346 (2009)
- de Boor, C., Höllig, K., Sabin, M.: High accuracy geometric Hermite interpolation. *Comput. Aided Geom. Des.* **4**, 269–278 (1988)
- Degen, W.L.F.: Best approximation of parametric curves by splines. In: Lyche, T., Schumaker, L.L. (eds.) *Mathematical Methods in Computer Aided Geometric Design II*, pp. 171–184. Academic Press, Waltham (1992)
- Eisele, E.F.: Chebychev approximation of plane curves by splines. *J. Approx. Theory* **76**, 133–148 (1994)
- Farin, G.: *Curves and Surfaces for CAGD: A Practical Guide*, 5th edn. Academic Press, San Diego (2002)
- Floater, M.S.: Chordal cubic spline interpolation is fourth order accurate. *IMA J. Numer. Anal.* **26**, 25–33 (2006)
- Floater, M.S., Surazhsky, T.: Parameterization for curve interpolation. In: Jetter, K., et al. (eds.) *Topics in Multivariate Approximation and Interpolation*, pp. 39–54. Elsevier, Amsterdam (2006)
- Foley, T.A., Nielson, G.M.: Knot selection for parametric spline interpolation. In: Lyche, T., Schumaker, L.L. (eds.) *Mathematical Methods in Computer Aided Geometric Design*, pp. 261–272. Academic Press, Waltham (1989)
- Iserles, A.: On the numerical quadrature of highly-oscillating integrals I: Fourier transforms. *IMA J. Numer. Anal.* **24**, 365–391 (2004)
- Iserles, A.: On the numerical quadrature of highly-oscillating integrals II: irregular oscillators. *IMA J. Numer. Anal.* **25**, 25–44 (2005)
- Jaklič, G., Kozak, J., Krajnc, M., Žagar, E.: On geometric interpolation by planar parametric polynomial curves. *Math. Comput.* **76**, 1981–1993 (2007)
- Jiang, Y., Xu, Y.: Fast discrete algorithms for sparse Fourier expansions of high dimensional functions. *J. Complex.* **26**, 51–81 (2010)
- Jiang, Y., Xu, Y.: Fast Fourier-Galerkin methods for solving singular boundary integral equations: numerical integration and precondition. *J. Comput. Appl. Math.* **234**, 2792–2807 (2010)
- Kress, R.: *Linear Integral Equations*. Springer, New York (1989)
- Kropinski, M.C., Quaife, B.: Fast integral equation methods for the modified Helmholtz equation. *J. Comput. Phys.* **230**, 425–434 (2011)
- Kropinski, M.C., Quaife, B.: Fast integral equation methods for Rothe's method applied to the isotropic heat equation. *Comput. Math. Appl.* **61**, 2436–2446 (2011)

28. Lee, E.T.Y.: Choosing nodes in parametric curve interpolation. *Comput. Aided Des.* **21**, 363–370 (1989)
29. Levin, D.: Procedures for computing one-and-two dimensional integrals of functions with rapid irregular oscillations. *Math. Comput.* **38**, 531–538 (1982)
30. Lin, H., Xu, Z., Tang, H., Cai, W.: Image approximations to electrostatic potentials in layered electrolytes/dielectrics and an ion-channel model. *J. Sci. Comput.* **53**, 249–267 (2012)
31. Lu, B.Z., Zhou, Y.C., Holst, M.J., McCammon, J.A.: Recent progress in numerical methods for the Poisson-Boltzmann equation in biophysical applications. *Commun. Comput. Phys.* **3**, 973–1009 (2008)
32. Mørken, K.: On geometric interpolation of parametric surfaces. *Comput. Aided Geom. Des.* **22**, 838–848 (2005)
33. Mørken, K., Scherer, K.: A general framework for high-accuracy parametric interpolation. *Math. Comput.* **66**, 237–260 (1997)
34. Ruotsalainen, K., Wendland, W.: On the boundary element method for some nonlinear boundary value problems. *Numer. Math.* **53**, 299–314 (1988)
35. Schaback, R.: Planar curve interpolation by piecewise conics of arbitrary type. *Constr. Approx.* **9**, 373–389 (1993)
36. Steinbach, O., Tchoualag, L.: Fast Fourier transform for efficient evaluation of Newton potential in BEM. *Appl. Numer. Math.* **81**, 1–14 (2014)
37. Smitheman, S.A., Spence, E.A., Fokas, A.S.: A spectral collocation method for the Laplace and modified Helmholtz equations in a convex polygon. *IMA J. Numer. Anal.* **30**, 1184–1205 (2010)
38. Vainikko, G.M.: Perturbed Galerkin method and general theory of approximate methods for nonlinear equations. *Zh. Vychisl. Mat. Fiz.* **7**, 723–751 (1967)
39. Vainikko, G.M.: Galerkin’s perturbation method and the general theory of approximate methods (English translation). *USSR Comput. Math. Methods Phys.* **7**, 1–41 (1967)
40. Xu, Z., Liang, Y., Xing, X.: Mellin transform and image charge method for dielectric sphere in an electrolyte. *SIAM J. Appl. Math.* **73**, 1396–1415 (2013)
41. Yan, Y., Sloan, I.H.: On integral equations of the first kind with logarithmic kernels. *J. Integral Equ. Appl.* **1**, 549–579 (1988)